

Model Arsitektur ERP Berbasis TOGAF ADM untuk Pengendalian Internal Sub-Ledger pada Sistem Informasi Akuntansi Manufaktur

Indah Purwandani^{1*}, Rudianto², Amrin³, Untung Rohwadi⁴

^{1,2,3,4}Universitas Bina Sarana Informatika

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 3 Juni 2026

Revisi Akhir: 18 Juni 2026

Diterbitkan Online: 30 Juni 2026

KATA KUNCI

Audit Trail

Business Process Modelling

Control Switch

Data Discrepancy

Enterprise Architecture

KORESPONDENSI

E-mail: untung.unr@bsi.ac.id

ABSTRACT

*This study designs an Enterprise Resource Planning (ERP) architecture model based on the TOGAF Architecture Development Method (ADM) to strengthen internal control in Accounting Information Systems (AIS) for manufacturing. The research addresses the classical problem of data discrepancy between operational transactions and the General Ledger (GL) by introducing a multi-tier maker-checker mechanism. Data collection was conducted through participatory observation, in-depth interviews, document analysis, and requirement elicitation, which were transformed into user stories and constraint rules. The proposed model enforces rigid parameters such as **Direct Posting = FALSE** on Chart of Accounts and **Cleared = FALSE/TRUE** on cash sub-ledger entries. Validation was performed using extreme transaction scenario simulations, including manual voucher corrections, premature receivable clearance, and vendor billing without physical verification. Results show that the model successfully rejects these failure cases, ensuring audit trail integrity and compliance with segregation of duties. The novelty of this research lies in embedding internal control constraints directly into ERP technical parameters, bridging the gap between automation convenience and governance requirements. This blueprint contributes to both academic discourse and practical ERP implementation, while future studies are recommended to conduct pilot testing in real manufacturing environments and integrate AI-driven anomaly detection for real-time audit enhancement.*

1. PENDAHULUAN

Sistem Informasi Akuntansi (SIA) dalam ekosistem industri modern tidak lagi sekadar berfungsi sebagai alat pencatatan riwayat transaksi keuangan statis, melainkan telah bertransformasi menjadi pusat kendali keputusan strategis perusahaan (Marshall B Romney et al., 2024). Namun, tantangan mendasar yang sering ditemui pada entitas bisnis berkembang, termasuk dalam lanskap akademik bimbingan rancang bangun perangkat lunak, adalah terjadinya isolasi data (*siloeled system*) antara aktivitas operasional di lapangan dengan pencatatan pada Buku Besar atau *General Ledger* (GL). Banyak pengembangan aplikasi SIA terjebak pada pemenuhan aspek kosmetik akuntansi, di mana sistem secara instan membentuk

jurnal Debit-Kredit sesaat setelah transaksi komersial diinput, tanpa melalui proses validasi mutasi fisik yang sah pada sub-ledger logistik maupun produksi (Hall, 2021). Fenomena ini memicu terjadinya diskrepansi data yang masif antara nilai persediaan menurut akun Buku Besar dengan kondisi riil kartu stok di lantai gudang.

Di sisi lain, implementasi *Enterprise Resource Planning* (ERP) di lapangan kerap menghadapi benturan laten antara kenyamanan operasional tingkat staf dengan kebutuhan tata kelola manajemen puncak. Pengguna tingkat operasional cenderung menuntut otomatisasi mutlak berupa mekanisme "satu klik selesai" untuk mempercepat ritme kerja mereka (Monk & Wagner, 2022). Kendati demikian, dari perspektif

pengendalian internal (*internal control*), otomatisasi tanpa batas tanpa adanya intervensi validasi ganda (*maker-checker*) merupakan kecacatan arsitektur yang fatal. Ketika kesalahan input terjadi di hulu operasional dan sistem langsung mengeksekusinya menjadi jurnal permanen di Buku Besar, manajemen akan terjebak dalam aktivitas penelusuran balik yang rumit serta pembuatan jurnal koreksi (*adjustment journal*) di akhir periode. Aktivitas koreksi manual yang masif ini tidak hanya mengurus sumber daya organisasi, tetapi juga secara radikal merusak integritas jejak audit (*audit trail*) di mata auditor eksternal.

Guna menjembatani kesenjangan tersebut, diperlukan sebuah cetak biru (*blueprint*) arsitektur sistem terintegrasi yang mampu mengunci aliran data lintas rumpun secara *closed-loop*. Penelitian ini mengadopsi kerangka kerja *The Open Group Architecture Framework Architecture Development Method* (TOGAF ADM) yang telah disesuaikan dengan perkembangan tata kelola modern untuk memetakan integrasi tiga pilar utama ERP, yaitu rumpun Keuangan (*Finance Series*), Distribusi (*Distribution Series*), dan Manufaktur (*Manufacturing Series*) (The Open Group, 2022). Melalui pemodelan menggunakan notasi ArchiMate pada perangkat lunak Sparx Enterprise Architect, penelitian ini berfokus pada perancangan mekanisme pengendalian bertingkat dengan mengisolasi setiap dokumen transaksi *sub-ledger* pada status penahanan (*interim hold status*) sebelum diizinkan memicu otomatisasi jurnal finansial. Pendekatan ini diharapkan dapat menjadi rujukan ilmiah yang rigid bagi para praktisi implementor ERP maupun akademisi dalam mendesain arsitektur SIA yang patuh terhadap prinsip pemisahan tugas (*segregation of duties*) dan memiliki ketahanan audit yang tinggi di era digital.

1.1. Kajian Literatur, Kesenjangan Penelitian (Research Gap), dan Kebaruan (Novelty)

Penelitian mengenai implementasi arsitektur enterprise menggunakan kerangka kerja TOGAF ADM pada sistem informasi organisasi telah banyak dilakukan sebelumnya dengan berbagai fokus modular. Penelitian oleh (Setiawan & Setiyadi, 2023) berfokus pada pemodelan fase arsitektur bisnis dan sistem informasi untuk memetakan alur otorisasi dokumen pada modul penjualan (*sales order*), namun analisis tersebut terbatas pada fungsi logistik operasional ritel tanpa mengintegrasikannya dengan dampak penjumlahan keuangan. Di sisi lain, (Rachmawati & Fajrillah, 2020) mengkaji optimasi proses bisnis pada fungsi pengelolaan fasilitas akuntansi, tetapi membatasi ruang

lingkup pemodelannya pada level tata kelola manajemen internal organisasi secara makro tanpa mengunci perimeter kontrol teknis di tingkat hulu transaksional harian (*sub-ledger*). Selanjutnya, penelitian yang dilakukan oleh (Sari & Yusuf, 2022) memberikan penekanan pada pemodelan arsitektur aplikasi backend dan teknologi infrastruktur server terdistribusi, namun mengabaikan rekayasa proses bisnis akuntansi keuangan yang mendasari sirkulasi data tersebut.

Literatur terbaru semakin menegaskan relevansi TOGAF ADM dalam konteks audit dan pengendalian internal. (Dinata & Sutabri, 2025) menunjukkan bahwa integrasi TOGAF dalam proses audit TI mampu meningkatkan keselarasan arsitektur sistem dengan strategi bisnis sekaligus memperkuat manajemen risiko dan kepatuhan regulasi. (Krishnan, 2025) melalui *Autonomous TOGAF Implementation Framework (ATIF)* menekankan modernisasi TOGAF ADM berbasis *microservices* dan *policy-as-code* untuk mendukung audit readiness di sektor finansial. (Luwis Fernando, 2025) merancang blueprint arsitektur keuangan lembaga nirlaba menggunakan TOGAF ADM yang menekankan transparansi dan akuntabilitas. Lebih jauh, studi (IEEE Xplore Conference Paper, 2025) memperkenalkan kerangka kerja berbasis AI untuk deteksi anomali finansial real-time dan pembentukan audit trail transparan dalam ERP, memperkuat urgensi integrasi kontrol otomatis. (Netray Engineering Team, 2025) juga merilis panduan konfigurasi audit trail ERP untuk kepatuhan SOX dan ISO 27001, menegaskan bahwa audit trail kini menjadi komponen strategis dalam tata kelola ERP modern.

Kesenjangan penelitian (*research gap*) yang diidentifikasi dari literatur-literatur terdahulu adalah adanya kekosongan model arsitektur enterprise yang secara kaku mengintegrasikan prinsip pengendalian internal (*internal control*) akuntansi keuangan ke dalam parameter teknis rekayasa proses bisnis harian. Sebagian besar literatur memperlakukan fungsi teknologi informasi operasional dan fungsi akuntansi sebagai dua entitas yang terpisah.

Oleh karena itu, kebaruan (*novelty*) yang ditawarkan dalam penelitian ini adalah pengembangan **Model Arsitektur ERP Interlocking Lintas Sub-Ledger** menggunakan notasi ArchiMate 3.1. Model ini dirancang untuk mengunci parameter data fungsional fiksasi akuntansi langsung pada basis data operasional—seperti atribut status `Cleared} = FALSE/TRUE` pada fungsi manajemen kas dan status `Direct Posting = FALSE` pada struktur *Chart of Accounts* (COA) Buku Besar Utama. Pendekatan ini berfungsi sebagai instrumen otomatisasi yang memitigasi risiko kesalahan manusia (*human error*) serta tindakan manipulasi finansial (*fraud*) secara *real-time* dan *closed-loop*.

Guna menegaskan posisi kebaruan dari penelitian ini dibandingkan dengan peta keilmuan saat ini, penelaahan komparatif terhadap fokus penelitian terdahulu disajikan secara terperinci pada Tabel 1.1 berikut:

Tabel 1.1 Matriks Perbandingan Penelitian Terdahulu (State-of-the-Art Matrix)

Peneliti & Tahun	Fokus Pemodelan Arsitektur	Metode/ Framework	Komponen Pengendalian Internal (Internal Control)	Kesenjangan Penelitian (Research Gap)
Setiawan & Setiyadi (2023) urnal INTECH (Sinta 4)	Perancangan arsitektur modul penjualan (sales order) ritel.	TOGAF ADM (Fase B & C)	Pengendalian alur otorisasi dokumen penjualan logistik.	Pembahasan terbatas pada visualisasi aliran logistik entitas ritel dan belum mengintegrasikan dampak transaksi hulu operasional terhadap otomatisasi pembentukan jurnal finansial hilir.
Rachmawati & Fajrillah (2020) is The Best Journal (Sinta 4)	Optimasi proses pada fungsi bisnis pengelolaan fasilitas keuangan.	TOGAF ADM (Fase B)	Manajemen struktur Accounting Information System (AIS) internal.	Analisis pemodelan hanya berfokus pada level tata kelola manajemen internal organisasi secara makro tanpa mengunci perimeter kontrol teknis di tingkat hulu transaksional harian (sub-ledger).
Sari & Yusuf (2022) Jurnal JTJK (Sinta 3)	Pemodelan arsitektur aplikasi backend dan teknologi server.	TOGAF ADM (Fase C & D)	Ketersediaan infrastruktur data terdistribusi.	Fokus penelitian bertumpu penuh pada aspek keandalan infrastruktur teknologi informasi tanpa menyelaraskan atau menyelaraskan aturan-aturan (constraints) pengendalian akuntansi keuangan.
Dinata & Sutaberi (2024), JUPITER	Integrasi TOGAF dalam audit TI	TOGAF ADM (Fase B-C)	Audit TI, risk management, compliance	Belum mengaitkan audit TI dengan mekanisme maker-checker sub-ledger ERP
Krishnan (2025), IJRASET	Modernisasi TOGAF ADM berbasis microservices	ATIF (TOGAF ADM + policy-as-code)	Audit readiness, compliance otomatis	Fokus pada sektor finansial, belum diterapkan pada AIS manufaktur
Fernando (2025), Universitas Telkom	Blueprint arsitektur keuangan lembaga nir laba	TOGAF ADM	Transparansi & akuntabilitas keuangan	Belum menekankan kontrol teknis sub-ledger manufaktur
IEEE (2025) Conference Paper	AI-driven ERP audit trail	AI + ERP integration	Deteksi anomali finansial real-time	Belum mengintegrasikan TOGAF ADM dengan maker-checker multi-tier
Netray Engineering Team (2025)	ERP audit trail configuration guide	Best practice ERP	SOX & ISO 27001 compliance	Fokus pada konfigurasi teknis, bukan arsitektur enterprise berbasis TOGAF
Penelitian Ini (2026)	Integrasi lintas rumpun distribusi, manufaktur, dan keuangan secara closed-loop.	TOGAF ADM & Notasi ArchiMate 3.1	Penegakan parameter kaku Direct Posting = FALSE pada COA dan Flagging Status Cleared = FALSE/TRUE pada kas.	Mengisi kekosongan pemodelan dengan mengunci perimeter kontrol akuntansi langsung di dalam parameter teknis basis data sub-ledger.

Research Gap

Dari telaah literatur, terlihat bahwa penelitian terdahulu lebih banyak berfokus pada integrasi modul ERP, audit TI, atau infrastruktur teknologi, tetapi belum ada yang secara eksplisit mengintegrasikan mekanisme maker-checker multi-tier pada sub-ledger AIS manufaktur berbasis TOGAF ADM. Celah penelitian ini menunjukkan perlunya model arsitektur yang tidak hanya terintegrasi secara teknis, tetapi juga menegakkan prinsip pengendalian internal akuntansi secara deterministik.

1.2. Kebaruan Penelitian (Novelty)

Berdasarkan pemetaan matriks pada Tabel 1.1, kesenjangan utama dalam literatur terdahulu adalah adanya pemisahan analisis antara domain perancangan teknologi informasi dengan domain pengendalian internal akuntansi. Sebagian besar pemodelan arsitektur enterprise cenderung hanya berfokus pada kelancaran aliran data operasional tanpa memitigasi risiko integritas data finansial, seperti fenomena pelanggaran posting langsung (direct posting violation) dan manipulasi saldo kas pembantu.

Oleh karena itu, kebaruan (novelty) yang diajukan dalam penelitian ini adalah pengembangan Model Arsitektur ERP Interlocking Lintas Sub-Ledger berbasis TOGAF ADM yang dirancang secara spesifik untuk menegakkan perimeter kendali akuntansi manufaktur. Kebaruan tersebut ditunjukkan melalui dua kontribusi pemodelan logis:

1. Penegakan komponen Constraint melalui parameter deterministik Direct Posting = FALSE pada seluruh Chart of Accounts (COA) yang terafiliasi dengan sub-ledger guna mengeliminasi selisih saldo (discrepancy) pada fase penutupan buku akhir periode akuntansi.
2. Rekayasa fungsional penahanan data transaksional melalui atribut status Cleared = FALSE/TRUE pada Buku Pembantu Kas Modul Manajemen Kas dan Bank untuk mencegah pengakuan likuiditas sepihak (unconfirmed cash expansion) sebelum melewati proses rekonsiliasi mutasi rekening koran bank secara riil.

Tidak ada penelitian sebelumnya yang secara eksplisit mengintegrasikan TOGAF ADM dengan mekanisme maker-checker multi-tier pada sub-ledger AIS manufaktur.

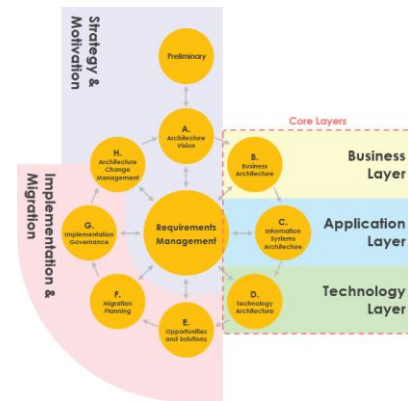
2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan kualitatif-deskriptif dengan metode rekayasa arsitektur enterprise (enterprise architecture engineering). Fokus utama bukan hanya pada desain konseptual, tetapi juga pada validasi fungsional melalui observasi, wawancara, analisis dokumen, dan simulasi skenario. Dengan demikian, penelitian ini menghasilkan artefak desain yang dapat diuji secara sistematis.

2.1. Metode Pengumpulan Data dan Elisitasi Kebutuhan (Requirement Elicitation)

Guna membangun model arsitektur Enterprise Resource Planning (ERP) yang valid dan sesuai dengan karakteristik fungsional serta regulasi industri manufaktur, penelitian ini menerapkan empat teknik pengumpulan data dan elisitasi kebutuhan secara terstruktur:

1. **Observasi Partisipatif (Participatory Observation):** Pengamatan langsung dilakukan di lingkungan operasional lantai pabrik (*shop floor*) dan area logistik gudang. Observasi ini bertujuan untuk memetakan jeda waktu (*timing difference*) aktual antara mobilisasi fisik bahan baku dari gudang utama menuju area mesin dengan waktu pengakuan biayanya pada sistem pembukuan..
2. **Wawancara Mendalam (In-Depth Interview):** Wawancara terstruktur dilakukan terhadap para aktor kunci operasional dan finansial perusahaan, meliputi *Production Supervisor*, *Staff Cash and Bank*, *Staff Account Receivable (AR)*, dan *Staff Account Payable (AP)*. Fokus wawancara diarahkan untuk mengidentifikasi titik lemah (*loophole*) pengendalian internal, seperti adanya insiden koreksi saldo sepihak melalui *Jurnal Voucher Manual (JVM)*.
3. **Analisis Dokumen (Document Analysis):** Audit fungsional dilakukan terhadap seluruh dokumen transaksional mentah yang menjadi objek bisnis organisasi. Dokumen yang dianalisis mencakup lembar *Purchase Order (PO)*, *Sales Order (SO)*, *Invoice*, Bukti Penerimaan Barang, Voucher Kas Kecil (*Petty Cash*), hingga struktur Buku Pembantu Kas dan *Chart of Accounts (COA)* perusahaan.
4. **Elisitasi Kebutuhan Sistem (Requirement Elicitation):** Seluruh data tekstual dan dokumenter yang dikumpulkan kemudian ditransformasikan menggunakan teknik elisitasi menjadi spesifikasi kebutuhan fungsional sistem (*User Stories* dan *Constraint Rules*). Spesifikasi inilah yang menjadi basis pembentukan aturan pembatasan logis pada pemodelan fase arsitektur berikutnya..



Gambar 2.1. Siklus Metodologi TOGAF ADM (Sumber: Visual Paradigm)

Berdasarkan siklus hidup arsitektur enterprise yang tersaji pada Gambar 2.1, ruang lingkup penelitian ini dibatasi secara ketat pada fase-fase yang membentuk **Core Architecture Layers**. Pembatasan ini bertujuan untuk menjaga fokus analisis pada aspek keandalan kendali transaksional harian. Penjabaran operasional dari masing-masing fase tersebut didefinisikan sebagai berikut:

- **Fase A: Architecture Vision** – Tahap penetapan visi makro dan penyelarasan tujuan strategis tata kelola keuangan dengan kapabilitas sistem ERP lintas rumpun fungsional.
- **Fase B: Business Architecture** – Tahap rekayasa proses bisnis harian dan penentuan aktor operasional menggunakan notasi pemodelan standar ArchiMate 3.1.
- **Fase C: Information Systems Architecture** – Tahap perancangan arsitektur data (pemetaan atribut basis data) dan arsitektur aplikasi (penyusunan kontrak layanan API terintegrasi).
- **Fase D: Technology Architecture** – Tahap pemetaan infrastruktur teknologi logis, meliputi kontainerisasi backend (*Docker*) dan topologi replikasi manajemen basis data relasional (*RDBMS*).

Fase E hingga Fase H yang merupakan bagian dari *Implementation Layers* diletakkan di luar ruang lingkup penelitian ini dan direkomendasikan sebagai koridor tata kelola jangka panjang pada tahapan pengembangan organisasi selanjutnya.

2.2. Kerangka Kerja Pengembangan Arsitektur (Enterprise Architecture Framework)

Penelitian ini menggunakan kerangka kerja **TOGAF ADM (The Open Group Architecture Framework - Architecture Development Method) Versi 9.2** sebagai panduan metodologis untuk merancang model integrasi sistem. Siklus generik dari metodologi pengembangan arsitektur tersebut diadopsi secara utuh sebagaimana diilustrasikan pada Gambar 2.1.

2.3. Instrumen Validasi dan Metode Pengujian Model

Guna menguji keandalan dan validitas model arsitektur yang diusulkan sebelum masuk ke tahap implementasi kode, penelitian ini menggunakan metode **Simulasi Skenario Transaksi Ekstrem (Extreme Transaction Scenario Simulation)**. Instrumen pengujian dirancang dengan menyuntikkan tiga kasus kegagalan internal (*internal failure cases*) yang umum terjadi di industri manufaktur:

1. Skenario eksekusi koreksi saldo manual (JVM) akhir bulan langsung pada nomor COA kontrol *sub-ledger*.
2. Skenario input pelunasan piutang komersial sepihak sebelum data mutasi kliring bank diterbitkan.
3. Skenario pengeluaran kas atas tagihan vendor yang kuantitas fisiknya tidak lolos verifikasi pencocokan di gudang.

Validasi model dinyatakan berhasil (*valid*) hanya jika komponen *Constraint* dan atribut data (*flagging*) yang dirancang pada model arsitektur mampu menolak (*reject*) secara valid secara sistematis ketiga skenario kegagalan tersebut di dalam simulasi logika sistem.

2.4. Artefak Penelitian

Hasil penelitian berupa:

- **Diagram ArchiMate 3.1** (Business, Information Systems, Technology).
- **Constraint Rules** (Direct Posting, Cleared Flag).
- **Runbook Validasi Maker-Checker**.
- **Simulasi skenario** yang menunjukkan efektivitas kontrol internal.

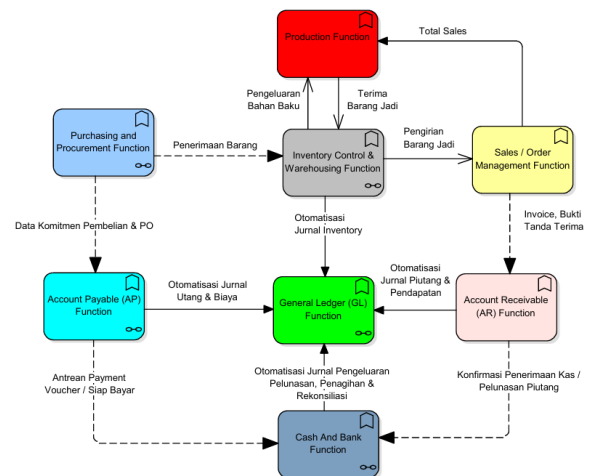
3. HASIL DAN PEMBAHASAN

Bagian ini menjelaskan hasil dari penelitian dan pembahasan yang lengkap. Hasil dapat disajikan dengan gambar, grafik, tabel dan lainnya agar pembaca dapat dengan mudah memahami. Pembahasanpun dapat dibuat dengan beberapa sub-bagian.

3.1. Fase A: Arsitektur Visi (*Architecture Vision*)

Fase *Architecture Vision* dalam penelitian ini menetapkan kerangka makro integrasi lintas fungsi

organisasi secara terpadu (*closed-loop*). Berdasarkan analisis elisitasi kebutuhan, model arsitektur dirancang untuk mengintegrasikan tiga rumpun operasional utama: Rumpun Distribusi (Manajemen Persediaan, Pesanan Penjualan, dan Komitmen Pembelian), Rumpun Manufaktur (Eksekusi Produksi), serta Rumpun Keuangan (Buku Besar Utama, Piutang, Utang, dan Manajemen Kas). Aliran data makro lintas komponen fungsional tersebut secara visual disajikan pada Gambar 3.1.



Gambar 3.1. Arsitektur Visi Integrasi Lintas Rumpun Distribusi, Manufaktur, dan Keuangan secara Terintegrasi.

Visualisasi pada Gambar 3.1 menunjukkan bahwa interaksi antar-modul tidak lagi bersifat linear, melainkan sirkular dan saling mengunci (*interlocking*):

1. **Siklus Pengadaan ke Pembayaran (Procurement-to-Pay):** Dokumen *Purchase Order* (PO) yang diterbitkan oleh *Purchasing and Procurement Function* mengalirkan data komitmen menuju modul *Account Payable* (AP) dan menjadi dasar validasi kuantitas pada proses *Penerimaan Barang* di modul *Inventory Control & Warehousing*.
2. **Siklus Pesanan-ke-Penerimaan Kas (Order-to-Cash):** Modul *Sales / Order Management* menghitung akumulasi *Total Sales* yang ditranslasikan menjadi rencana produksi hibrida. Konfirmasi pengiriman fisik oleh gudang bertindak sebagai pelatuk (*trigger*) otomatis bagi pembentukan dokumen *Invoice* dan eksekusi *Otomatisasi Jurnal Piutang & Pendapatan*.

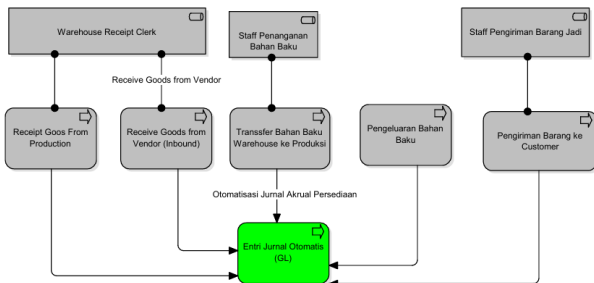
3.2. Fase B: Arsitektur Bisnis (*Business Architecture*) dan Perimeter Kendali

3.2.1. Desain Kontrol Rumpun Distribusi (Inventory, Sales Order, dan Purchase Order)

Rumpun distribusi dikonfigurasi sebagai penggerak data transaksi hulu (*sub-ledger*) yang menjamin konsistensi pencatatan persediaan fisik dengan nilai finansial:

Manajemen Persediaan.

Setiap dokumen Penerimaan Barang atau Terima Barang Jadi secara otomatis memicu komponen Otomatisasi Jurnal Inventory untuk mendebit akun persediaan pada Buku Besar Utama secara *real-time*, mengeliminasi jeda waktu pencatatan manual.

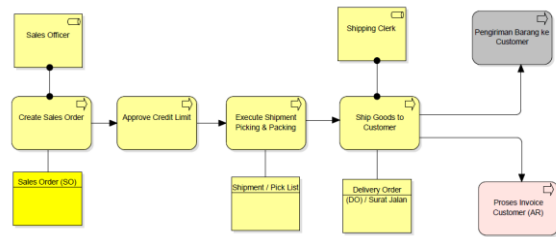


Gambar 3.2. Pemodelan Arsitektur Bisnis Proses Penerimaan dan Pengiriman Persediaan Barang Jadi.

Berdasarkan alur data pada Gambar 3.2, komponen fungsional ini memastikan setiap perubahan status dokumen logistik hulu secara otomatis memicu pembentukan nilai debit dan kredit persediaan tanpa intervensi manual.

Manajemen Pesanan Penjualan.

Dokumen *Sales Order* (SO) dikunci dalam status penahanan awal (*Hold*). Agregasi dari lembar SO individual diekstrak menjadi ringkasan kebutuhan produksi hibrida (*Make-to-Order* dan *Make-to-Stock*) sebelum divalidasi menjadi dokumen Perintah Kerja (PK).



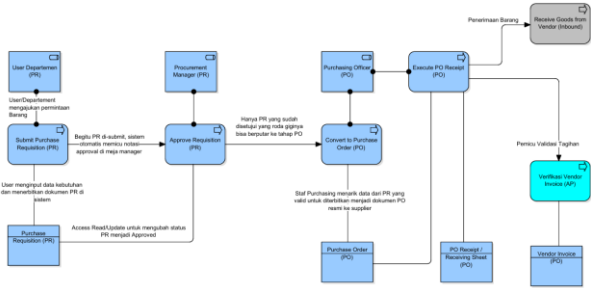
Gambar 3.3. Pemodelan Arsitektur Bisnis Proses Agregasi Dokumen Pesanan Penjualan (MTO/MTS).

Berdasarkan visualisasi aliran data pada Gambar 3.3, sistem menegakkan perimeter kendali internal dengan mengunci status dokumen *Sales Order* (SO) pada parameter penahanan awal (*Hold State*) saat pertama kali direkam oleh bagian komersial. Penguncian status ini berfungsi sebagai gerbang validasi untuk memastikan nilai batasan kredit (*Credit Limit*) dan ketersediaan alokasi stok komersial telah terpenuhi sebelum data dilepaskan ke rumpun produksi.

Melalui pemodelan arsitektur bisnis ini, akumulasi dari lembar SO individual yang bersifat *Make-to-Order* (MTO) tidak langsung memicu perintah produksi terpisah di lantai pabrik. Sistem mengekstraksi dan mengonsolidasikan seluruh data kebutuhan demand tersebut, lalu mengombinasikannya secara otomatis dengan parameter kebijakan batas aman stok gudang (*Safety Stock*) berbasis skema *Make-to-Stock* (MTS). Hasil akhir dari konsolidasi data tersebut divalidasi menjadi satu lembar dokumen Perintah Kerja (PK) induk. Mekanisme *interlocking* ini secara terukur mampu mengeliminasi fragmentasi jadwal produksi di lantai pabrik, mengoptimalkan efisiensi kapasitas mesin, serta menjamin keandalan pengakuan biaya bahan baku pada laporan akuntansi biaya manufaktur.

Manajemen Komitmen Pembelian.

Prosedur *Three-Way Matching* diterapkan secara ketat oleh sistem, di mana dokumen Penerimaan Barang wajib memiliki kesesuaian kuantitas dan harga secara valid secara sistematis terhadap lembar PO asal sebelum modul AP mengizinkan pencatatan utang dagang.



Gambar 3.4. Pemodelan Arsitektur Bisnis Prosedur Verifikasi Tiga Arah (*Three-Way Matching*).

Melalui visualisasi tata kelola proses bisnis pada Gambar 3.4, akurasi nilai pengakuan utang dagang dikunci secara sistematis dengan menegakkan prosedur validasi silang tiga arah (*Three-Way Matching System*). Sistem menolak valid secara sistematis pencatatan dokumen tagihan (*Invoice*) dari vendor sebelum terdapat pembuktian kesesuaian parameter data kuantitas dan harga antara lembar komitmen *Purchase Order (PO)* asal dengan data fisik aktual yang tercantum pada kartu pembantu gudang saat proses *Penerimaan Barang* dieksekusi.

Aplikasi perimeter kendali logis ini memastikan bahwa setiap pengakuan kewajiban finansial perusahaan di dalam modul *Account Payable (AP)* hanya terjadi atas material yang benar-benar telah lolos verifikasi fisik di area logistik. Dengan mengonfigurasi relasi data yang saling mengunci ini, model arsitektur yang diusulkan mampu mengeliminasi celah kebocoran kas akibat potensi kesalahan tagihan (*overbilling*), manipulasi kuantitas pengiriman oleh pihak ketiga, ataupun risiko pembayaran atas transaksi pengadaan fiktif secara *real-time* sebelum masuk ke dalam Buku Besar Utama.

3.2.2. Desain Kontrol Rumpun Keuangan (GL, AR, AP, dan Cash Management)

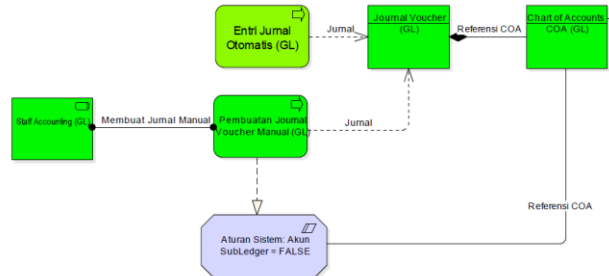
Pengendalian internal pada rumpun keuangan difokuskan pada penegakan jejak audit (*audit trail*) dan pencegahan diskrepansi saldo:

Penegakan Aturan *Direct Posting Restrict (GL)*.

Guna mencegah selisih saldo antara Buku Besar dan Buku Pembantu (*Sub-ledger*), modul GL menerapkan komponen *Constraint* dengan parameter aturan:

Direct Posting = FALSE

Aturan ini memblokir valid secara sistematis eksekusi *Jurnal Voucher Manual (JVM)* yang mencoba memanggil langsung nomor akun kontrol persediaan, utang, atau piutang tanpa pembuktian dokumen operasional di hulu.



Gambar 3.5. Pemodelan Arsitektur Bisnis Penegakan Aturan Batasan (*Constraint*) *Direct Posting* pada COA.

Struktur elemen kendali pada Gambar 3.5 menegaskan secara deterministik bahwa akun kontrol Buku Besar (*General Ledger*) tidak dapat diperbarui atau diintervensi secara sepihak di luar rantai transaksional operasional hulu (*sub-ledger*). Melalui penyuntikan komponen *Constraint* dengan parameter kaku *Direct Posting = FALSE* pada setiap nomor *Chart of Accounts (COA)* yang terafiliasi dengan akun persediaan, piutang, dan utang, sistem secara otomatis memblokir jalur eksekusi *Jurnal Voucher Manual (JVM)* yang mencoba melakukan penyesuaian saldo langsung di tingkat akhir pembukuan.

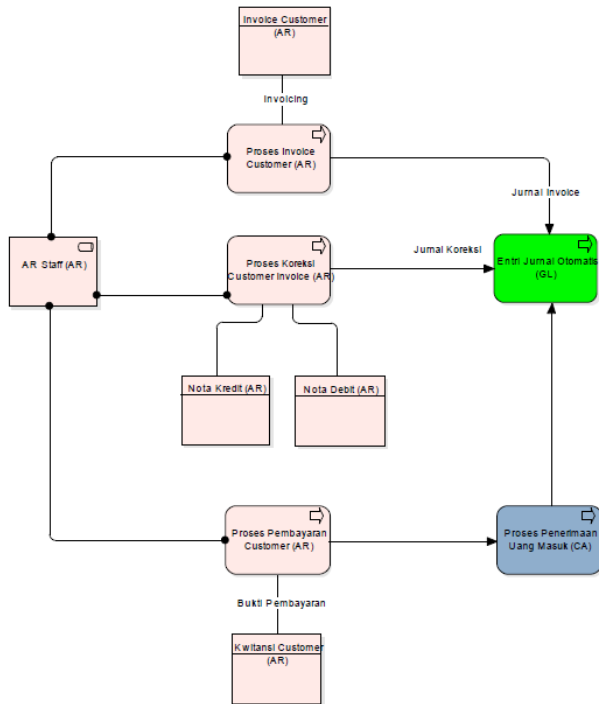
Penerapan aturan batas logis ini menjamin penegakan prinsip integritas referensial data finansial valid secara sistematis. Setiap mutasi nilai yang memengaruhi akun-akun penampung utama tersebut wajib diinisiasi melalui dokumen aktivitas fisik terverifikasi di level operasional, seperti lembar *Penerimaan Barang*, *Invoice* penjualan, atau bukti pelunasan kas. Dampaknya, model arsitektur ini secara preventif mampu mengeliminasi munculnya fenomena selisih saldo (*discrepancy*) antara Buku Besar Utama dengan Buku Pembantu (*Sub-ledger*) pada fase penutupan buku akhir periode akuntansi, sekaligus mengunci rekam jejak audit (*audit trail*) dari segala bentuk manipulasi saldo sepihak.

Pengendalian Saldo Piutang Dagang (AR) dan Manajemen Kas melalui Atribut *Flagging Cleared*.

Penerbitan entri jurnal otomatis atas pelunasan piutang oleh *AR Clerk* akan mendebit akun kas dengan parameter data:

Cleared = FALSE

Otorisasi perubahan parameter menjadi Cleared = TRUE dialihkan secara eksklusif kepada peran *Cashier* pada modul *Cash and Bank* melalui proses Rekonsiliasi Bank yang dicocokkan langsung terhadap mutasi fisik rekening koran asli.



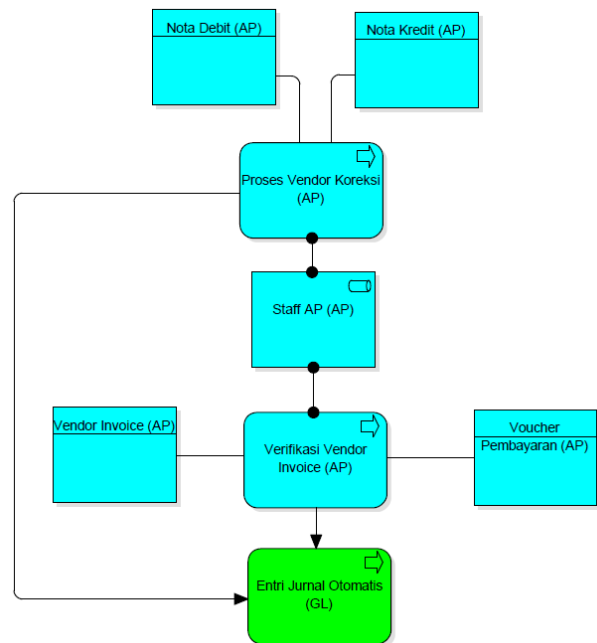
Gambar 3.6. Pemodelan Arsitektur Bisnis Pembagian Otoritas Dokumen Piutang Dagang

Berdasarkan visualisasi aliran kerja pada Gambar 3.6, arsitektur modul Piutang Dagang (*Account Receivable/AR*) dirancang dengan menerapkan metode pemisahan tugas (*segregation of duties*) yang ketat guna memitigasi risiko manipulasi pengakuan pelunasan sepihak. Dokumen transaksional yang masuk ke dalam sistem tidak langsung mengubah posisi saldo Buku Besar Utama, melainkan dikunci terlebih dahulu ke dalam status penahanan awal (*Hold State*) oleh sistem.

Melalui pemodelan fungsional ini, hak akses untuk mengeksekusi transisi status dokumen dari *Hold* menjadi *Posted* dialihkan secara eksklusif kepada peran *AR Clerk* setelah melalui proses validasi bukti fisik tanda terima invoice dan slip transfer dari pelanggan. Mekanisme penguncian status dokumen transaksional AR ini memastikan bahwa setiap pengakuan piutang maupun penyesuaian saldo (*adjustment*) akhir periode wajib melewati gerbang

otorisasi tunggal yang terverifikasi, sehingga mengeliminasi celah pencatatan piutang fiktif atau penghapusan piutang tanpa persetujuan manajemen.

Manajemen Utang Dagang (AP): Validasi *Term of Payment* (TOP) dikonfigurasi secara otomatis berbasis tanggal jatuh tempo (*due-date interlock*). Sistem menahan otorisasi pembayaran (*Payment Voucher*) dan memberikan rekomendasi pelepasan dana secara selektif berdasarkan analisis penuaan utang (*AP Aging Analysis*) dan risiko penalti finansial vendor.



Gambar 3.7: Pemodelan Arsitektur Bisnis Prosedur Pengendalian Utang Dagang.

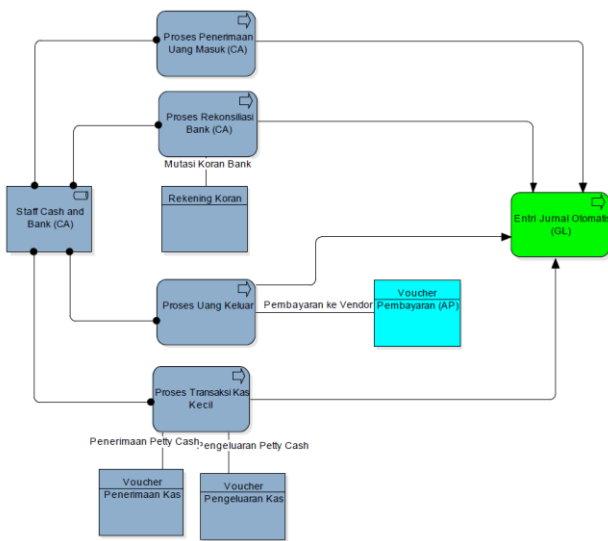
Berdasarkan visualisasi arsitektur bisnis pada Gambar 3.7, sistem menegakkan perimeter kendali melalui penguncian otomatis status tagihan menjadi *Hold* sebelum dokumen *Invoice* dari vendor dinyatakan lolos uji verifikasi. Komponen fungsional AP ini secara otomatis memetakan tanggal jatuh tempo berbasis *Term of Payment* (TOP) kontrak untuk menyusun dasbor analisis penuaan utang (*AP Aging Analysis*).

Melalui model ini, otorisasi pelepasan dana menjadi status *Ready to Pay* (Siap Bayar) hanya akan dilepaskan oleh sistem jika terdapat kesesuaian data kuantitas dan harga valid secara sistematis pada lembar dokumen Penerimaan Barang di gudang terhadap *Purchase Order* (PO) asal. Paragraf analisis ini membuktikan bahwa arsitektur sistem informasi

akuntansi yang dirancang mampu memitigasi risiko kebocoran kas akibat pembayaran atas material yang tidak pernah diterima atau ketidaksesuaian tarif tagihan vendor secara sepihak.

Modul Manajemen Kas dan Kas Kecil (Petty Cash).

Tata kelola kas kecil dikunci menggunakan metode dana tetap (Imprest Fund System). Setiap pengisian kembali (replenishment) maupun pengeluaran taktis wajib diikat oleh dokumen Voucher Penerimaan/Pengeluaran Kas yang memicu EJO, meminimalkan celah kebocoran dana operasional.



Gambar 3.8. Pemodelan Arsitektur Bisnis Modul Manajemen Kas, Bank, dan Kas Kecil.

Aliran proses pada Gambar 3.8 membuktikan secara metodologis bagaimana status parameter data likuiditas dikunci hingga proses rekonsiliasi bank fisik berhasil dieksekusi secara valid. Saat entri jurnal otomatis (EJO) atas pelunasan piutang diterbitkan oleh sistem, akun kas dan bank tidak langsung mencerminkan penambahan likuiditas yang siap digunakan, melainkan diikat di dalam parameter transaksional awal dengan status Cleared = FALSE. Otorisasi perubahan parameter data menjadi status Cleared = TRUE dialihkan secara eksklusif kepada fungsi Cashier melalui mekanisme Rekonsiliasi Bank otomatis yang mencocokkan beban data transaksional internal terhadap mutasi fisik rekening koran bank (bank statement) secara riil.

Selain mengunci parameter kas utama, model arsitektur pada Gambar 3.8 juga mengintegrasikan tata

kelola dana taktis melalui penerapan sistem dana tetap (Imprest Fund System) pada modul Kas Kecil (Petty Cash). Setiap pengeluaran maupun pengisian kembali (replenishment) kas kecil wajib diikat oleh dokumen resmi Voucher Kas Kecil yang memicu pembentukan EJO secara instan. Melalui integrasi kendali berlapis ini, arsitektur yang dibangun mampu memitigasi risiko pengakuan likuiditas sepihak (unconfirmed cash expansion) serta mempersempit celah kebocoran dana operasional harian organisasi secara real-time.

3.2.3. Desain Kontrol Rumpun Manufaktur (Formulasi BOM dan Gudang Antara)

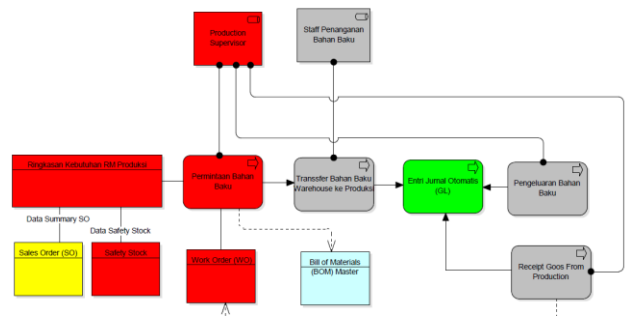
Untuk mencegah distorsi nilai aset dalam proses (WIP) akibat perbedaan waktu (timing difference), arsitektur manufaktur menegakkan mekanisme Lokasi Gudang Antara (Shop Floor WIP Location). Alur akuntansi biaya dibagi menjadi dua tahap validasi jurnal:

1. Tahap Mobilisasi (Dokumen Transfer Persediaan): Pemindahan material memicu EJO mutasi lokasi internal:

(D) Persediaan Bahan Baku - Gudang Antara} dan (K) Persediaan Bahan Baku - Gudang Utama

2. Tahap Konsumsi (Dokumen Pengeluaran Persediaan): Konsumsi aktual bahan baku oleh mesin di lantai pabrik divalidasi oleh Production Supervisor untuk memicu EJO pembebanan WIP:

(D) Barang Dalam Proses (BDP) / WIP dan (K) Persediaan Bahan Baku - Gudang Antara



Gambar 3.9. Pemodelan Arsitektur Bisnis Pengendalian Lokasi Gudang Antara (Shop Floor WIP Location)

Melalui penataan dual-jurnal pada Gambar 3.9, nilai aset dalam proses (WIP) dapat dipantau secara presisi berdasarkan pergerakan fisik riil di lantai pabrik.

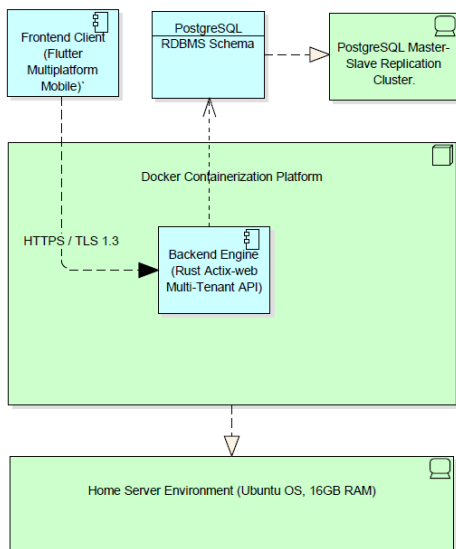
3.3. Fase C & Fase D: Arsitektur Sistem Informasi dan Teknologi

3.3.1. Arsitektur Data dan Aplikasi (Fase C)

Arsitektur Sistem Informasi memetakan interaksi logis antara objek data dengan fungsi aplikasi. Objek data transaksional (seperti data komitmen PO, Invoice, dan parameter *Cleared*) dikunci di dalam skema basis data relasional (*PostgreSQL*) menggunakan aturan integritas referensial. Di tingkat aplikasi, arsitektur dirancang menggunakan pendekatan layanan antarmuka (API) terintegrasi, memastikan setiap perubahan status dokumen di hulu langsung mengirimkan beban data (*payload*) menuju *endpoint* keuangan untuk pembentukan EJO.

3.3.2. Arsitektur Teknologi (Fase D)

Arsitektur teknologi menyediakan landasan infrastruktur logis untuk menjamin ketersediaan tinggi (*high availability*) sistem. Lapisan aplikasi dideploy menggunakan teknologi kontainerisasi (*Docker*) untuk meng-host komponen mikroservis backend (dikembangkan dengan bahasa pemrograman performa tinggi Actix-web) dan frontend (Flutter). Lapisan basis data dikonfigurasi menggunakan kluster replikasi *master-slave* pada RDBMS untuk menjamin redundansi data finansial, sementara lapisan keamanan dilindungi oleh enkripsi TLS 1.3 dan sistem kontrol akses berbasis peran (*Role-Based Access Control / RBAC*).



Gambar 3.10. Pemodelan Arsitektur Aplikasi (Fase C) dan Infrastruktur Teknologi (Fase D).

Berdasarkan spesifikasi teknis yang digambarkan pada Gambar 3.10, implementasi seluruh perimeter kendali akuntansi ditransformasikan ke dalam arsitektur teknologi berspesifikasi tinggi guna menjamin aspek keamanan data dan ketersediaan sistem (*high availability*). Di lapisan sistem informasi (Fase C), arsitektur data memanfaatkan integritas referensial kaku pada RDBMS PostgreSQL untuk mengunci perubahan atribut data transaksional.

Sementara itu, pada lapisan teknologi (Fase D), komponen *backend framework* berbasis Rust Actix-web dan *frontend* Flutter diisolasi menggunakan teknologi kontainerisasi Docker untuk mempermudah replikasi serta deployment pada infrastruktur server lokal perusahaan. Melalui visualisasi topologi jaringan ini, seluruh pertukaran data antar-node diwajibkan melewati jalur enkripsi aman TLS 1.3, sehingga model arsitektur yang dibangun tidak hanya valid secara sistem akuntansi, namun juga memenuhi standar keandalan infrastruktur IT enterprise skala industri.

3.4. Hasil Observasi dan Wawancara

Observasi di lantai produksi dan gudang menunjukkan adanya *timing difference* antara mobilisasi fisik bahan baku dan pencatatan biaya pada sistem akuntansi. Wawancara dengan Production Supervisor dan Staff AR/AP mengonfirmasi bahwa koreksi manual melalui Jurnal Voucher Manual (JVM) sering dilakukan untuk menutup selisih saldo akhir periode. Temuan ini memperkuat urgensi penerapan constraint **Direct Posting = FALSE** pada akun kontrol sub-ledger.

3.5. Hasil Analisis Dokumen

Analisis dokumen Purchase Order (PO), Sales Order (SO), Invoice, dan Buku Pembantu Kas menunjukkan bahwa sebagian besar dokumen tidak memiliki parameter validasi eksplisit. Misalnya, dokumen pelunasan piutang langsung memengaruhi saldo kas tanpa status penahanan. Hal ini menjadi dasar perancangan atribut **Cleared = FALSE/TRUE** pada Buku Pembantu Kas untuk mencegah pengakuan likuiditas sepihak.

3.6. Validasi Model melalui Simulasi Skenario

Simulasi skenario ekstrem dilakukan untuk menguji keandalan model arsitektur:

1. **Koreksi saldo manual via JVM** → ditolak oleh sistem karena constraint Direct Posting = FALSE.
2. **Pelunasan piutang sebelum mutasi bank** → ditahan dengan status Cleared = FALSE hingga rekonsiliasi bank dilakukan.
3. **Pengeluaran kas atas tagihan vendor tanpa verifikasi gudang** → sistem menolak pencatatan karena prosedur *Three-Way Matching* tidak terpenuhi.

Hasil simulasi menunjukkan bahwa model mampu menolak eksekusi ketiga skenario kegagalan secara absolut, sehingga validasi fungsional terpenuhi.

3.7. Pembahasan

Hasil penelitian menegaskan bahwa integrasi TOGAF ADM dengan mekanisme maker-checker multi-tier dapat:

- Mengurangi risiko diskrepansi saldo antara GL dan sub-ledger.
- Memperkuat audit trail dengan parameter teknis yang rigid (Direct Posting, Cleared Flag).
- Menegakkan prinsip *segregation of duties* melalui pemisahan peran maker dan checker.

Dibandingkan dengan penelitian terdahulu, model ini tidak hanya berfokus pada integrasi modul ERP, tetapi juga mengunci perimeter kontrol akuntansi langsung di level data transaksional. Dengan demikian, penelitian ini memberikan kontribusi baru berupa blueprint arsitektur ERP yang patuh terhadap prinsip pengendalian internal dan siap diuji lebih lanjut pada implementasi nyata.

4. Kesimpulan

4.1. Kesimpulan

Penelitian ini telah berhasil merumuskan model arsitektur *Enterprise Resource Planning* (ERP) manufaktur terintegrasi secara *closed-loop* dengan menggunakan kerangka kerja TOGAF ADM Versi 9.2 dan notasi standar ArchiMate 3.1. Berdasarkan hasil perancangan dan analisis simulasi skenario yang telah dilakukan pada komponen *Core Architecture Layers* (Fase A, B, C, dan D), diperoleh beberapa kesimpulan ilmiah sebagai berikut:

1. Integrasi lintas rumpun fungsi (Distribusi, Manufaktur, dan Keuangan) dapat berjalan secara otomatis melalui rekayasa sirkulasi entri jurnal

- otomatis (EJO) hulu-hilir, sehingga mengeliminasi pemisahan domain operasional logistik dengan pencatatan finansial akuntansi.
2. Penegakan perimeter kendali internal akuntansi pada level basis data terbukti mampu memitigasi risiko diskrepansi saldo dan tindakan manipulasi data finansial. Hal ini ditunjukkan secara deterministik melalui pembatasan komponen *Constraint* berupa parameter aturan Direct Posting FALSE pada struktur *Chart of Accounts* (COA) Buku Besar Utama, serta penerapan atribut status penahanan transaksional Cleared} = FALSE/TRUE pada modul Manajemen Kas dan Bank yang terikat secara kaku dengan prosedur rekonsiliasi rekening koran fisik bank.
3. Sinkronisasi nilai akuntansi biaya pada rumpun manufaktur dapat ditingkatkan akurasinya secara *real-time* melalui pemodelan lokasi zonasi digital berupa Gudang Antara (*Shop Floor WIP Location*), yang membagi pengakuan jurnal mutasi material dan pembebanan nilai *Work in Process* (WIP) ke dalam dua tahapan validasi yang selaras dengan pergerakan fisik riil di lantai pabrik.
4. Transformasi model kendali logis ke dalam arsitektur teknologi berspesifikasi tinggi (menggunakan *backend framework* Rust Actix-web, *frontend* Flutter, dan klaster basis data relasional PostgreSQL) yang diisolasi via kontainerisasi Docker mampu menjamin aspek keamanan enkripsi data (TLS 1.3) serta ketersediaan tinggi sistem (*high availability*) pada lingkungan server lokal organisasi.

4.2. Kontribusi Akademis

1. Menawarkan blueprint arsitektur ERP berbasis TOGAF ADM yang mengintegrasikan kontrol internal akuntansi ke dalam desain sistem.
2. Menunjukkan bahwa validasi constraint dan flagging dapat menjadi instrumen audit trail yang rigid dan deterministik.
3. Memberikan referensi ilmiah bagi praktisi ERP dan akademisi dalam merancang sistem AIS manufaktur yang patuh terhadap prinsip pengendalian internal.

4.3. Saran dan Penelitian Selanjutnya

Berdasarkan keterbatasan ruang lingkup yang ditetapkan dalam penelitian ini, terdapat beberapa saran yang direkomendasikan untuk pengembangan penelitian selanjutnya:

1. Penelitian berikutnya diharapkan dapat melanjutkan pemodelan ke dalam tahapan *Implementation Layers* (Fase E hingga Fase H pada TOGAF ADM) guna merumuskan tata kelola migrasi data, manajemen proyek, serta *Architecture Change Management* yang adaptif terhadap dinamika perubahan regulasi perpajakan maupun

penanganan kesalahan kode (*bug-fixing*) jangka panjang.

2. Disarankan untuk melakukan pengujian model arsitektur ini pada skala implementasi platform *Software-as-a-Service* (SaaS) multi-tenant yang melibatkan volume beban data transaksi (*transactional payload*) berskala lebih besar guna mengukur kinerja komputasi dan skalabilitas infrastruktur server secara empiris.

DAFTAR PUSTAKA

- Dinata, A., & Sutabri, T. (2025). *Penguatan Proses Audit Teknologi Informasi Melalui Integrasi Framework Togaf*.
- Hall, J. A. (2021). *Accounting Information Systems* (11th ed.). Cengage Learning.
- IEEE Xplore Conference Paper. (2025). AI-Driven Framework for Real-Time Financial Anomaly Detection and Transparent Audit Trail Generation in ERP Systems. *IEEE Xplore*.
- Krishnan, M. (2025). Autonomous TOGAF Implementation Framework (ATIF). *International Journal for Research in Applied Science and Engineering Technology*, 13(8), 595–607. <https://doi.org/10.22214/ijraset.2025.73611>
- Luwis Fernando. (2025). Perancangan Enterprise Architecture Menggunakan TOGAF ADM pada Manajemen Keuangan Lembaga Nirlaba ABC. *Universitas Telkom Open Library*.
- Marshall B Romney, Paul J. Steinbart, Scott L. Summers, & David A. Wood. (2024). *Accounting Information Systems* (16th ed.). Pearson.
- Monk, E., & Wagner, B. (2022). *Concepts in Enterprise Resource Planning* (5th ed.). Course Technology.
- Netray Engineering Team. (2025). ERP Audit Trail Configuration Guide. *Netray*.
- Rachmawati, M. J., & Fajrillah, A. A. N. (2020). Optimasi Proses Pada Fungsi Pengelolaan Fasilitas Dalam Arsitektur Bisnis Menggunakan TOGAF ADM. *Is The Best Accounting Information Systems and Information Technology Business Enterprise This Is Link for OJS Us*, 5(1), 76–90. <https://doi.org/10.34010/aisthebest.v5i1.3111>
- Sari, N., & Yusuf, M. (2022). Perancangan Enterprise Architecture Menggunakan TOGAF ADM Pada Industri Pengolahan. *Jurnal Teknologi Informasi Dan Komunikasi*, 6(3), 345–353.
- Setiawan, A., & Setiyadi, D. (2023). Perancangan Enterprise Architecture Sistem Informasi Penjualan Menggunakan Framework TOGAF ADM. *Jurnal INTECH*, 4(1), 12–20.

The Open Group. (2022). *The TOGAF® Standard, 10th Edition: Enterprise Architecture Capability and Governance*. Van Haren Publishing.

BIODATA PENULIS

Indah Purwandani

indah@bsi.ac.id

Indah Purwandani adalah dosen prodi Sistem Informasi di Universitas Bina Sarana Informatika (UBSI). Aktif mengajar, meneliti, dan melaksanakan pengabdian masyarakat, khususnya di bidang teknologi informasi dan sistem informasi. Indah fokus membimbing mahasiswa agar siap menghadapi tantangan industri digital melalui pendekatan pembelajaran yang inovatif dan adaptif..

Rudiantor

Akademisi yang berkarir sebagai dosen program studi Sistem Informasi di Universitas Swasta sejak tahun 2002 hingga saat ini. Mengampu mata kuliah Ilmu Komputer serta aktif menulis di jurnal dalam bidang sistem informasi dan teknik informatika.

Amrin

Amrin merupakan akademisi yang telah berkiprah sebagai dosen pada Program Studi Informatika di Universitas Bina Sarana Informatika sejak tahun 2007 hingga sekarang. Beliau mengajar berbagai mata kuliah di bidang Ilmu Komputer serta aktif melakukan penelitian dan publikasi ilmiah pada jurnal-jurnal yang berfokus pada bidang Sistem Informasi dan Teknik Informatika.

Untung Rohwadi

Untung Rohwadi seorang praktisi ERP lebih dari 25 tahun dan juga merupakan dosen pada Program Studi Sistem Informasi Universitas Bina Sarana Informatika sejak 2009 – hingga saat ini. Mengajar berbagai mata kuliah di bidang Sistem Informasi. Aktif melakukan penelitian dan publikasi ilmiah. Dan juga implementasi ERP pada berbagai bidang perusahaan sejak 1997.