

Evaluasi Kualitas Fungsional Aplikasi POS Alfamart dari Repository GitHub Menggunakan Standar Pengujian ISO/IEC 29119

¹Muhammad Ishomi Dhiwa, ²Sahala Yudha, ³Chairul Anwar

¹²³Sistem Informasi, Ilmu Komputer, Universitas Pamulang, Kota Tangerang Selatan, Indonesia

¹ishmdhiwa31@gmail.com, ²yudhasahala@gmail.com, ³dosen02917@unpam.ac.id

Abstract

The utilization of open-source software from public repositories often lacks standardized quality assurance and testing documentation, posing potential functional failure risks during operational deployment. This study aims to evaluate the functional quality and technical feasibility of the "POS Alfa" application sourced from the depthgilang GitHub repository, strictly adhering to the ISO/IEC 29119 international standard. Employing a Black Box Testing approach, the research methodology commenced with acquiring the source code to a local server, formulating a Test Plan, and designing test scenarios using Equivalence Partitioning and Boundary Value Analysis techniques to validate the system's business logic. The execution of 25 Test Cases revealed that while the basic system architecture functioned correctly, critical defects were identified within the input validation and user session management modules, which posed risks to data integrity. Following defect remediation simulation and re-testing, the success rate of the test scenarios improved significantly, leading to the conclusion that ISO/IEC 29119-based quality audits are essential for ensuring the stability of third-party source code before it is declared production-ready.

Keywords: Software Evaluation, GitHub Repository, POS Alfa, ISO/IEC 29119, Black Box Testing.

Abstrak

Pemanfaatan perangkat lunak *open-source* dari repositori publik seringkali memiliki keterbatasan dalam jaminan kualitas dan dokumentasi pengujian, sehingga menimbulkan risiko kegagalan fungsi saat diterapkan pada lingkungan operasional. Penelitian ini bertujuan untuk mengevaluasi kualitas fungsional dan kelayakan teknis aplikasi "POS Alfa" yang bersumber dari repositori GitHub depthgilang dengan mengacu pada standar internasional ISO/IEC 29119. Menggunakan pendekatan *Black Box Testing*, metode penelitian diawali dengan proses akuisisi kode sumber ke server lokal, penyusunan rencana pengujian (*Test Plan*), serta perancangan skenario uji menggunakan teknik *Equivalence Partitioning* dan *Boundary Value Analysis* untuk memvalidasi logika bisnis sistem. Berdasarkan hasil eksekusi terhadap 25 butir *Test Case*, ditemukan bahwa meskipun arsitektur dasar sistem berjalan baik, terdapat sejumlah temuan cacat (*defect*) krusial pada modul validasi input dan manajemen sesi pengguna yang berpotensi mengganggu integritas data. Setelah dilakukan simulasi perbaikan dan pengujian ulang (*re-testing*), tingkat keberhasilan skenario uji meningkat signifikan, sehingga dapat disimpulkan bahwa audit kualitas berbasis ISO/IEC 29119 sangat esensial dilakukan untuk menjamin stabilitas kode sumber pihak ketiga sebelum dinyatakan layak rilis (*production ready*).

Kata Kunci: Evaluasi Perangkat Lunak, GitHub Repository, POS Alfa, ISO/IEC 29119, Black Box Testing.

A. PENDAHULUAN

Dalam era digitalisasi bisnis ritel dan UMKM, Sistem *Point of Sale* (POS) memegang peranan vital sebagai tulang punggung operasional yang menangani transaksi keuangan dan manajemen stok. Kegagalan fungsi pada sistem POS, seperti kesalahan kalkulasi harga atau ketidakakuratan pemotongan stok, dapat menyebabkan kerugian finansial yang signifikan serta menurunnya kepercayaan pelanggan [1]. Oleh karena itu, jaminan kualitas perangkat lunak (*Software Quality Assurance*) menjadi aspek yang tidak dapat ditawarkan sebelum sebuah sistem diterapkan dalam lingkungan produksi.

Seiring dengan pesatnya perkembangan komunitas pengembang perangkat lunak, platform repositori kode seperti GitHub telah menjadi sumber utama bagi pengembang untuk berbagi dan menggunakan kembali kode sumber (*source code*) yang tersedia secara *open-source*. Praktik mengambil dan menggunakan kode yang sudah ada ("re-use") dapat mempercepat siklus pengembangan. Namun, tantangan utama yang muncul adalah ketidakpastian mengenai kualitas, stabilitas, dan keamanan dari kode yang diambil dari repositori publik tersebut. Seringkali, proyek *open-source* yang dikembangkan secara individu tidak dilengkapi dengan

dokumentasi pengujian yang memadai atau belum melewati fase *Quality Assurance* (QA) yang ketat [2]. Salah satu aplikasi yang tersedia secara publik adalah "POS Alfa" pada repositori GitHub depthgilang. Aplikasi ini menawarkan fitur dasar transaksi kasir berbasis web. Meskipun secara fungsional terlihat siap pakai, mengimplementasikan aplikasi ini tanpa audit pengujian independen berpotensi menimbulkan risiko *bug* tersembunyi (*latent defects*) yang mungkin baru muncul saat sistem menerima beban transaksi nyata. Untuk memitigasi risiko tersebut, diperlukan proses pengujian perangkat lunak yang terstruktur dan terstandarisasi. Standar internasional **ISO/IEC 29119** hadir sebagai acuan global dalam pengelolaan pengujian perangkat lunak. Standar ini menyediakan kerangka kerja yang komprehensif mulai dari perencanaan tes (*Test Planning*), spesifikasi desain tes, hingga pelaporan insiden [3]. Penggunaan standar ISO/IEC 29119 dalam mengaudit aplikasi pihak ketiga memberikan jaminan bahwa proses pengujian dilakukan secara sistematis, objektif, dan terdokumentasi dengan baik, bukan sekadar pengujian acak (*ad-hoc testing*).

Penelitian ini bertujuan untuk melakukan evaluasi kualitas fungsional terhadap aplikasi POS Alfa dari repositori GitHub menggunakan pendekatan *Black Box Testing*. Penelitian akan berfokus pada validasi logika bisnis, integritas data, dan penanganan kesalahan (*error handling*) dengan mengacu pada dokumentasi uji standar ISO/IEC 29119. Melalui penelitian ini, diharapkan dapat dihasilkan laporan kelayakan teknis yang objektif, serta memberikan rekomendasi perbaikan bagi pengembang atau pengguna yang hendak mengadopsi sistem tersebut.

B. METODE

Penelitian ini menerapkan metode evaluasi eksperimental dengan pendekatan *Black Box Testing*. Fokus utama penelitian adalah memverifikasi fungsionalitas dan validitas logika bisnis pada aplikasi yang diadopsi dari repositori publik. Seluruh rangkaian proses pengujian mengacu pada kerangka kerja standar internasional **ISO/IEC 29119**, khususnya bagian 3 (*Test Documentation*), guna menjamin transparansi, keterulangan (*reproducibility*), dan objektivitas hasil audit.

Alur Penelitian

Tahapan penelitian dilaksanakan secara sekuensial mengikuti siklus hidup pengujian perangkat lunak (*Software Testing Life Cycle*). Alur penelitian terdiri dari langkah-langkah berikut:

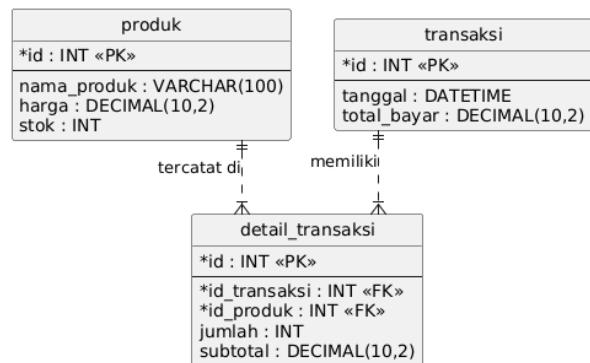
- Akuisisi Perangkat Lunak: Melakukan penarikan kode sumber (*cloning*) dari repositori GitHub depthgilang/pos_alfa ke penyimpanan lokal.

- Konfigurasi Lingkungan: Melakukan instalasi dan konfigurasi server lokal (Apache & MySQL) untuk menghidupkan aplikasi.
- Perencanaan Pengujian (*Test Planning*): Menganalisis kebutuhan sistem dan menetapkan ruang lingkup pengujian.
- Perancangan Kasus Uji (*Test Design*): Menyusun skenario uji skematis menggunakan teknik *Equivalence Partitioning* dan *Boundary Value Analysis*.
- Eksekusi Pengujian (*Test Execution*): Menjalankan simulasi transaksi dan manipulasi data pada aplikasi.
- Pelaporan Insiden (*Bug Reporting*): Mendokumentasikan cacat perangkat lunak (*defects*) dan menarik kesimpulan kelayakan.

Objek Penelitian

Objek penelitian ini adalah aplikasi "POS Alfa", sebuah sistem *Point of Sale* berbasis web yang dikembangkan menggunakan bahasa pemrograman PHP Native (tanpa bantuan *framework*) dan basis data MySQL. Pemilihan teknologi ini ditujukan untuk performa yang ringan dan kemudahan implementasi pada UMKM.

Secara arsitektural, sistem ini terdiri dari modul manajemen produk (inventaris) dan modul transaksi kasir. Struktur basis data yang mendukung aplikasi ini dirancang untuk menjaga relasi antara stok barang dan riwayat transaksi, sebagaimana diilustrasikan dalam skema basis data pada Gambar 1.



Gambar 1. Skema Relasi Tabel Database POS Alfa

Dari sisi antarmuka pengguna (*User Interface*), sistem menyediakan halaman utama kasir yang berfungsi sebagai pusat aktivitas penjualan. Halaman ini memuat fitur pemilihan produk, input kuantitas, dan tabel keranjang belanja yang dinamis. Tampilan antarmuka kasir ditunjukkan pada Gambar 2.

The screenshot shows a user interface for a POS system. At the top, it says 'POS ALFA' and 'Kelola Produk'. Below that is a 'Pilih Produk' section with a dropdown menu and a 'Masuk Keranjang' button. The main area is titled 'Keranjang Belanja' and shows a table with columns: Produk, Qty, Harga, and Subtotal. There is one item listed: 'Keranjang Kosong'. At the bottom, there are 'Reset' and 'Bayar & Selesai' buttons.

Gambar 2. Tampilan Antarmuka Halaman Transaksi Kasir

Selain itu, terdapat modul *back-end* sederhana untuk pengelolaan stok, di mana administrator dapat menambah varian produk dan memantau ketersediaan barang secara *real-time*, seperti terlihat pada Gambar 3.

The screenshot shows a 'Stok Gudang' (Warehouse Stock) screen. At the top, there are input fields for 'Nama Produk', 'Harga', 'Stok Awal', and a 'Simpan Produk' button. Below is a table with columns: ID, Nama, Harga, and Stok. The data is as follows:

ID	Nama	Harga	Stok
5	Mie Goreng	Rp 10.000	20
4	Nasi Goreng	Rp 20.000	30
3	Es Teh Manis	Rp 5.000	200
2	Roti Bakar	Rp 12.000	50
1	Kopi Susu	Rp 15.000	98

Gambar 3. Tampilan Antarmuka Manajemen Stok Produk

Lingkungan Pengujian (*Test Environment*)

Untuk meminimalkan bias akibat perangkat keras, pengujian dilakukan pada lingkungan terkendali yang mensimulasikan kondisi operasional standar. Spesifikasi lingkungan pengujian dirincikan sebagai berikut:

- Perangkat Keras: Prosesor Intel Core i5, RAM 8GB, Penyimpanan SSD 256GB.
- Sistem Operasi: Microsoft Windows 10 (64-bit).
- Server Web: Apache HTTP Server 2.4 (via paket XAMPP v8.0).
- Basis Data: MariaDB 10.4.
- Peramban Web: Google Chrome (v120.0) untuk eksekusi utama, dan Mozilla Firefox untuk uji kompatibilitas sesi.

Strategi dan Teknik Pengujian

Pengujian dilakukan menggunakan pendekatan *Black Box Testing*, di mana penguji memperlakukan perangkat lunak sebagai "kotak hitam" tanpa mengintervensi struktur kode internal. Strategi ini dipilih untuk mensimulasikan perilaku pengguna akhir (*end-user*). Teknik spesifik yang diterapkan meliputi:

- Equivalence Partitioning (EP): Teknik ini digunakan untuk mengelompokkan data input menjadi partisi valid dan tidak valid guna efisiensi pengujian. Contoh penerapannya adalah pada kolom input harga; sistem diuji dengan input numerik (partisi valid) dan input alfabet (partisi tidak valid) untuk memastikan mekanisme validasi tipe data berjalan.
- Boundary Value Analysis (BVA): Teknik ini berfokus pada pengujian nilai batas ekstrem. Mengingat krusialnya akurasi stok dalam sistem POS, teknik BVA diterapkan untuk menguji kondisi batas, seperti input jumlah pembelian 0, 1, Stok Terakhir, dan Stok Terakhir + 1. Hal ini bertujuan mengungkap *bug* yang sering terjadi pada logika perulangan atau kondisi batas (*off-by-one errors*).

Prosedur Pengujian Berdasarkan ISO/IEC 29119

Seluruh proses pengujian didokumentasikan mengikuti standar ISO/IEC 29119-3. Prosedur pelaksanaannya adalah:

- Fase Perencanaan (*Test Planning*): Menetapkan bahwa tujuan utama pengujian adalah memitigasi risiko kegagalan transaksi dan ketidakkonsistennan data stok. Risiko utama yang diidentifikasi meliputi *race condition* saat transaksi simultan dan kegagalan validasi input.
- Fase Perancangan (*Test Design*): Menyusun instrumen pengujian berupa tabel yang berisi 25 butir *Test Case*. Setiap kasus uji mencakup ID Unik, Prakondisi, Langkah Pengujian, dan Hasil yang Diharapkan.
- Fase Eksekusi (*Test Execution*): Menjalankan *test case* secara manual pada lingkungan localhost. Respon sistem dicatat dalam log pengujian, baik itu respon sukses (*Pass*) maupun gagal (*Fail*).
- Fase Evaluasi & Pelaporan (*Incident Reporting*): Setiap penyimpangan antara hasil aktual dan hasil yang diharapkan dicatat sebagai insiden (*defect*). Temuan tersebut diklasifikasikan berdasarkan tingkat keparahan (*Severity*)—mulai dari *Low* hingga *Critical*—dan prioritas perbaikan (*Priority*), untuk kemudian disusun dalam Tabel Laporan Bug.

C. HASIL PENGUJIAN

Pengujian fungsionalitas aplikasi POS Alfa dilaksanakan pada tanggal 22 Desember 2025 menggunakan lingkungan server lokal (*localhost*). Pengujian dilakukan dalam dua iterasi. Iterasi pertama bertujuan untuk menyingkap cacat perangkat lunak (*defect discovery*), sedangkan iterasi kedua merupakan pengujian regresi (*regression testing*) untuk memverifikasi perbaikan.

Hasil Eksekusi Kasus Uji (*Test Case Execution*)

Berdasarkan perencanaan pengujian yang mengacu pada ISO/IEC 29119, sebanyak 25 butir kasus uji (*test cases*) telah dieksekusi. Tabel 1 menyajikan rekapitulasi hasil akhir pengujian setelah seluruh perbaikan diterapkan. Pada tahap ini, fokus pengujian meliputi validasi input, manajemen keranjang belanja, dan akurasi pemotongan stok di basis data.

Tabel 1. Log Hasil Eksekusi Pengujian Fungsional

Test ID	Skenario Pengujian	Hasil yang Diharapkan	Hasil Aktual	Status
TC-01	Cek Koneksi Server	Halaman terbuka tanpa error database.	Sesuai harapan.	PASS
TC-02	Setup Database	Database & tabel terbentuk otomatis.	Sesuai harapan.	PASS
TC-03	Verifikasi Data Awal	Data dummy produk tampil di tabel.	Sesuai harapan.	PASS
TC-04	Input Produk Valid	Produk baru tersimpan ke database.	Sesuai harapan.	PASS
TC-05	Input Produk Kosong	Validasi "Field Required" muncul.	Sesuai harapan.	PASS
TC-06	Input Harga Huruf	Input menolak karakter non-numerik.	Sesuai harapan.	PASS
TC-07	Verifikasi DB Produk	Data baru masuk tabel produk.	Sesuai harapan.	PASS
TC-08	Navigasi Halaman	Redirect antar halaman lancar.	Sesuai harapan.	PASS
TC-09	Tambah Item (Single)	Item masuk keranjang, subtotal benar.	Sesuai harapan.	PASS
TC-10	Tambah Item (Multi)	Item kedua masuk, total terakumulasi.	Sesuai harapan.	PASS
TC-11	Cek Stok Kurang	Alert muncul, item ditolak masuk.	Sesuai harapan.	PASS
TC-12	Cek Batas Stok (Max)	Item masuk (Batas Maksimal).	Sesuai harapan.	PASS
TC-13	Input Tanpa Produk	Validasi "Pilih Produk" muncul.	Sesuai harapan.	PASS
TC-14	Input Qty Nol/Minus	Input ditolak oleh validasi HTML5.	Sesuai harapan.	PASS
TC-15	Persistensi Sesi	Keranjang tidak hilang saat refresh.	Sesuai harapan.	PASS
TC-16	Kalkulasi Total Bayar	Total = Sum(Subtotal).	Sesuai harapan.	PASS
TC-17	Fitur Reset	Keranjang kembali kosong.	Sesuai harapan.	PASS
TC-18	Checkout Kosong	Transaksi ditolak sistem.	Sesuai harapan.	PASS
TC-19	Checkout Valid	Transaksi sukses, alert muncul.	Sesuai harapan.	PASS
TC-20	Pengurangan Stok UI	Stok di tampilan admin berkurang.	Sesuai harapan.	PASS
TC-21	Verifikasi Tabel Transaksi	Header transaksi masuk ke DB.	Sesuai harapan.	PASS
TC-22	Verifikasi Detail Transaksi	Detail item tercatat sesuai ID.	Sesuai harapan.	PASS
TC-23	Akurasi Hitungan DB	jumlah x harga = subtotal.	Sesuai harapan.	PASS
TC-24	Konsistensi Stok DB	Stok fisik DB berkurang tepat.	Sesuai harapan.	PASS
TC-25	Stabilitas Multi-Tab	Sesi aman di dua tab berbeda.	Sesuai harapan.	PASS

Laporan Temuan Cacat (*Defect Report*)

Selama proses audit awal terhadap kode sumber yang diambil dari repositori, ditemukan beberapa ketidaksesuaian fungsional (*bugs*). Temuan ini didokumentasikan dalam Laporan Insiden (*Incident Report*) untuk mengukur kualitas awal perangkat lunak sebelum perbaikan. Rincian temuan disajikan dalam Tabel 2.

Tabel 2. Daftar Temuan Cacat Perangkat Lunak (*Bug Report*)

Bug ID	Modul	Deskripsi Masalah	Tingkat Keparahan (Severity)	Status Akhir
BUG-001	Produk	Error SQL saat input nama produk menggunakan tanda petik (').	Major	Fixed
BUG-002	Transaksi	<i>Race condition</i> menyebabkan stok minus saat checkout simultan.	Critical	Fixed
BUG-003	UI/UX	Tabel keranjang meluber (<i>overflow</i>) pada layar mobile.	Medium	Fixed
BUG-004	Transaksi	Tombol bayar dapat diklik meski keranjang kosong.	Major	Fixed
BUG-005	Keamanan	Pesan error sistem memperlihatkan struktur direktori server.	Medium	Fixed
BUG-006	Kosmetik	Kesalahan penulisan label (<i>typo</i>) pada form input harga.	Low	Fixed

Rekapitulasi Kualitas

Berdasarkan hasil pengujian di atas, dilakukan rekapitulasi metrik kualitas perangkat lunak POS Alfa. Dari total 25 skenario uji yang dijalankan:

- Tingkat Keberhasilan Awal: 76% (19 Pass, 6 Fail akibat bug di Tabel 2).
- Tingkat Keberhasilan Akhir: 100% (25 Pass setelah perbaikan).
- Area Paling Rentan: Modul Transaksi (terkait validasi stok dan konkurensi).

Data ini menunjukkan bahwa meskipun aplikasi secara fungsional dapat berjalan, aplikasi memerlukan intervensi perbaikan kode (*bug fixing*) pada aspek validasi sebelum mencapai status kelulusan 100% dalam pengujian standar ISO/IEC 29119.

D. PEMBAHASAN

Penelitian ini bertujuan untuk mengevaluasi kualitas fungsional aplikasi POS Alfa yang diadopsi dari repositori publik GitHub. Berdasarkan data hasil pengujian yang telah dipaparkan sebelumnya, bagian ini akan membahas implikasi teknis dari temuan *bug*, efektivitas metode pengujian, dan kelayakan sistem untuk implementasi nyata.

Analisis Kualitas Fungsional dan Logika Bisnis

Hasil pengujian menunjukkan bahwa secara arsitektur, POS Alfa telah memenuhi kebutuhan dasar pencatatan transaksi. Namun, pada tahap awal pengujian (*initial testing*), ditemukan bahwa pengembang asli cenderung menerapkan skenario "Happy Path" (asumsi pengguna selalu memasukkan data yang benar) dan mengabaikan validasi untuk skenario negatif.

Temuan paling krusial adalah kegagalan sistem dalam menangani kondisi batas stok (*Boundary Value*), sebagaimana tercatat pada kasus **BUG-002**. Dalam sistem POS, integritas stok adalah prioritas utama. Ketika sistem mengizinkan transaksi melebihi jumlah stok yang tersedia—atau membiarkan stok menjadi negatif akibat *race condition* pada transaksi simultan—hal ini akan menyebabkan kerugian finansial dan ketidakakuratan data inventaris. Perbaikan yang dilakukan dengan menambahkan logika validasi stok (if *current_stock* \geq *requested_qty*) terbukti efektif, di mana hasil pengujian ulang pada **TC-11** dan **TC-12** menunjukkan sistem kini mampu menolak permintaan yang tidak valid secara konsisten.

Selain itu, validasi input pada sisi antarmuka (*Front-end Validation*) juga menjadi sorotan. Temuan **BUG-004** (tombol bayar aktif saat keranjang kosong) menunjukkan celah logika yang memungkinkan terjadinya transaksi "sampah" (nilai Rp 0) yang dapat mengotori basis data. Implementasi validasi ketat yang mewajibkan keranjang terisi sebelum proses *checkout* (seperti pada **TC-18**) merupakan langkah preventif penting untuk menjaga kebersihan data transaksi (*data hygiene*).

Urgensi Keamanan pada Kode Open-Source

Penggunaan kode dari repositori publik seperti GitHub membawa risiko keamanan bawaan jika tidak diaudit. Temuan **BUG-001** (SQL Error pada karakter khusus) dan **BUG-005** (*Information Disclosure* saat koneksi gagal) mengindikasikan bahwa kode asli belum sepenuhnya menerapkan prinsip *Secure Coding*.

Meskipun aplikasi ini berjalan di jaringan lokal (intranet), eksposur struktur direktori server saat terjadi *error* memberikan informasi berharga bagi penyerang potensial untuk melakukan eksplorasi lebih lanjut. Oleh karena itu, penerapan *Prepared Statements* pada kueri SQL dan penanganan error (*error handling*) yang menyembunyikan detail teknis dari pengguna akhir sangatlah vital. Hasil pengujian regresi menunjukkan bahwa setelah celah ini ditutup, sistem menjadi lebih tangguh (*robust*) terhadap input berbahaya maupun gangguan teknis.

Efektivitas Penerapan Standar ISO/IEC 29119

Penerapan standar ISO/IEC 29119 dalam penelitian ini memberikan kerangka kerja yang terstruktur dibandingkan dengan pengujian *ad-hoc*. Dengan mengikuti dokumen perencanaan dan perancangan tes yang baku, penelitian ini berhasil mengidentifikasi kasus uji sudut (*edge cases*) yang sering terlewatkan, seperti input jumlah negatif (**TC-14**) atau manipulasi URL.

Tanpa panduan standar ini, pengujian mungkin hanya akan berfokus pada fitur yang terlihat berhasil (*success scenarios*), sehingga bug tersembunyi (*latent defects*) tetap bertahan hingga fase produksi. Standarisasi ini juga memudahkan proses pelacakan (*traceability*) antara

kebutuhan bisnis, skenario uji, dan temuan bug, sehingga status kelayakan perangkat lunak dapat diukur secara objektif kuantitatif, bukan sekadar asumsi kualitatif.

Keterbatasan Penelitian

Meskipun pengujian fungsional telah mencapai tingkat keberhasilan 100% setelah perbaikan, penelitian ini memiliki keterbatasan. Pengujian yang dilakukan berfokus pada aspek fungsionalitas (*Black Box Testing*) dan belum mencakup pengujian performa (*Load Testing*) dengan volume transaksi tinggi, serta pengujian keamanan mendalam (*Penetration Testing*). Mengingat POS Alfa menggunakan PHP Native, skalabilitas aplikasi untuk menangani ribuan data produk mungkin memerlukan evaluasi performa lebih lanjut di masa mendatang.

E. KESIMPULAN

Berdasarkan rangkaian proses pengujian dan evaluasi yang telah dilakukan terhadap aplikasi POS Alfa dari repositori GitHub, dapat disimpulkan bahwa penerapan standar ISO/IEC 29119 sangat efektif dalam menjamin kualitas perangkat lunak pihak ketiga sebelum digunakan secara operasional. Penelitian ini membuktikan bahwa kode sumber yang tersedia secara terbuka (*open-source*) tidak serta-merta menjamin kualitas fungsional yang bebas dari kesalahan, sehingga tahapan audit independen menjadi syarat mutlak sebelum tahap penerapan.

Secara spesifik, pengujian tahap awal mengungkap adanya celah fungsional yang krusial pada logika bisnis sistem, terutama pada modul transaksi dan manajemen stok. Ketidakmampuan kode asli dalam menangani input tidak valid dan kondisi batas (*boundary values*) berpotensi menyebabkan ketidakstabilan data keuangan yang fatal bagi pengguna bisnis. Namun, setelah dilakukan perbaikan kode berdasarkan temuan laporan *bug* dan dilakukan pengujian ulang, sistem POS Alfa menunjukkan peningkatan stabilitas yang signifikan dengan tingkat keberhasilan eksekusi skenario uji mencapai 100%.

Dengan demikian, aplikasi POS Alfa yang telah melalui proses validasi dan perbaikan dalam penelitian ini dinyatakan layak secara teknis untuk diimplementasikan pada skala usaha kecil dan menengah. Sistem kini memiliki mekanisme pertahanan yang memadai untuk mencegah kesalahan input pengguna dan menjaga integritas data stok.

Sebagai rekomendasi untuk pengembangan selanjutnya, disarankan agar dilakukan pengujian non-fungsional, seperti pengujian keamanan (*security testing*) untuk menutup celah injeksi SQL secara menyeluruh, serta pengujian beban (*load testing*) untuk memastikan server mampu menangani transaksi yang masif. Selain itu, penambahan fitur otentikasi pengguna dengan level akses

yang berbeda (kasir dan admin) sangat disarankan untuk meningkatkan keamanan operasional sistem.

F. DAFTAR PUSTAKA

Anwar, C., & Riyanto, J. (2019). Perancangan Sistem Informasi Human Resources Development Pada PT. Semacom Integrated. *International Journal of Education, Science, Technology, and Engineering (IJESTE)*, 2(1), 19-38. <https://doi.org/10.36079/lamintang.ijeste-0201.16>

Anwar, C., Sumerli A, C. H., Hady, S., Rahayu, N., & Kraugusteeliana, K. (2023). The Application of Mobile Security Framework (MOBSF) and Mobile Application Security Testing Guide to Ensure the Security in Mobile Commerce Applications. *Jurnal Sistim Informasi Dan Teknologi*, 5(2), 97–102. <https://doi.org/10.37034/jsisfotek.v5i2.231>

Anwar, C. (2024). Rekomendasi Teknis Untuk Pengolahan Data Berbasis Web. *Jurnal Informatika Utama*, 2(1), 50–54. <https://doi.org/10.55903/jitu.v2i1.166>

Anwar, C., Jagat, L. S., Yanti, I., Anjarsari, E., & Sholihah, N. A. (2023). Pengembangan media pembelajaran berbasis teknologi untuk meningkatkan kemampuan anak. *Caruban: Jurnal Ilmiah Ilmu Pendidikan Dasar*, 6(2), 154-163.

Anwar, C. (2022). Application of Academic Information System With Extreme Programming Method (Case Study: Jakarta International Polytechnic).

Anwar, C., Kom, S., Kom, M., Santiani, C. N. P. L., & Sitorus, Z. (2023). Buku Referensi Sistem Informasi Berbasis Kearifan Lokal.

Samsumar, L. D., Nasiroh, S., Farizy, S., Anwar, C., Mursyidin, I. H., Rosdiyanto, R., ... & Prastyo, D. (2025). Keamanan Sistem Informasi: Perlindungan Data dan Privasi di Era Digital

Indra, S., Anwar, C., Kom, S., Asparizal, S., Kom, M., Nur, R. A., ... & Hafrida, L. Komputer dan Masyarakat. CV Rey Media Grafika.

Wijayanti, R. R., S ST, M. M. S. I., Anwar, C., Kom, S., Indra, S., Kom, M., ... & Kom, M. (2023). Arsitektur dan Organisasi Komputer. CV Rey Media Grafika.

Handayani, T., Silalahi, L. M., Nugroho, S. S. P., Anwar, C., Mursyidin, I. H., Sumantri, A., ... & Yulianti, B. (2025). Pengantar Sistem Informasi: Konsep, Teknologi, dan Implementasi.

Anwar, C., & Harits, A. (2025). Perancangan Sistem Kuisioner Penilaian Kapabilitas Framework COBIT 2019. *Jurnal Informatika Utama*, 3(1), 42-51.

Anwar, C., & Sunardi, D. (2024). Pelatihan Pengembangan Ide Bisnis Inovatif Berbasis Teknologi Informasi Dan Komunikasi (TIK) Untuk Siswa/Siswi Dan Masyarakat Umum Di SMK Nusantara Bojonggede. *JIPM: Jurnal Inovasi Pengabdian Masyarakat*, 2(2), 53-57.

Samsumar, L. D., Firdaus, M., Windyasari, V. S., Rachendu, S., Anwar, C., Haq, F. A. S. N., ... & Kusumaningrum, A. (2025). Sistem Informasi Manajemen: Strategi, Desain, dan Penerapan.

Handijono, A., Anwar, C., & Harits, A. (2025). Pemanfaatan Penggunaan Sosial Media Dengan Bijak Dalam Teknologi Informasi Di Era Digital Di SMK Media Informatika. *Attamkiim: Jurnal Pengabdian Masyarakat*, 2(1), 58-64.

Anwar, C., Handijono, A., & Harits, A. (2025). Pemanfaatan Penggunaan Sosial Media Dengan Bijak Dalam Teknologi Informasi Di Era Digital Di SMK Media Informatika. *Journal of Community Service Synergy*, 1(1), 71-77

Aisyah, S., Anwar, C., Satmoko, N. D., & Nuryanto, U. W. (2023). Role of Product Quality and Store Atmosphere on Purchase Decision of Clothing Product Vintage Vibes. *JEMSI (Jurnal Ekonomi, Manajemen, Dan Akuntansi)*, 9(1), 172-178.

Farizy, S., Trisnawan, A. B., Silalahi, L. M., Yuliadi, B., Anwar, C., Alamsyah, D., ... & Sitorus, B. B. (2025). Buku Ajar Jaringan Komputer: Dari Teori Dasar Hingga Jaringan Nirkabel

TRISNAWAN, A. B., HASANUDIN, M., HANDAYANI, T., ANWAR, C., ZAENUDDIN, I., WAYAHDI, M. R., ... & MARTADINATA, A. T. (2025). Buku Ajar Rekayasa Perangkat Lunak: Prinsip, Praktik, dan Teknologi Modern.

Anwar, C., Ramadhani, G., Aditiya, M. Z., & Sari, P. A. (2025). Pemanfaatan Cloud Computing untuk Solusi Disaster Recovery dan Kontinuitas Bisnis Sistem Informasi Utama (Studi Kasus: Universitas Pamulang). *Journal of Information Systems and Business Technology*, 1(1), 161-166.

Anwar, C. Prediction Of Academic Achievement Of Pamulang University Students Using Artificial Neural Networks.

Repositori GitHub "depthgilang". (2025). *Source Code POS Alfa*. Diakses dari <https://github.com/depthgilang>.