



ISSN 2954-2757 (Online)

ESENSI: Jurnal Manajemen Bisnis

<https://esensijournal.com/index.php/esensi>

Vol. 28 Issue 3, Desember (2025)

doi.org/10.56943/esensi.v28i3.444

Application Control Effectiveness in Cloud-Based Accounting Software: Comparative Analysis of Mekari and Xero

Cornelius Jevon Tirta Salim^{1*}, Gregorius Rudy Antonio²

¹jevon.lim22@gmail.com, ²gregorius@staff.ubaya.ac.id

Universitas Surabaya

*Corresponding Author: Cornelius Jevon Tirta Salim

Email: jevon.lim22@gmail.com

ABSTRACT

This study evaluates application control effectiveness in two prominent cloud-based accounting platforms, Mekari Jurnal ID and Xero, through systematic black box testing of revenue cycle controls. Employing a qualitative explanatory research design, the investigation examined input controls (missing data checks, numeric-alphabetic validation, limit checks, range checks, validity checks, sign checks, sequence checks) and output controls (transaction logging, report accuracy) across quotation, sales order, and invoice modules. Testing was conducted between October 2024 and January 2025 using trial accounts with comprehensive working paper documentation. The findings revealed significant control weaknesses in both platforms including inadequate field validation permitting single-character entries, insufficient inventory limit enforcement enabling overselling, inconsistent discount percentage validation, absence of sequential document numbering protocols, and deficient transaction logging capabilities in Mekari Jurnal ID. Both platforms demonstrated accurate master data validation and sign check controls but failed to enforce sequence integrity or prevent commercially unreasonable value entries. Comparative analysis indicated that Mekari Jurnal ID exhibited stronger stage-based preventive controls while Xero demonstrated superior transaction logging granularity. The research contributes empirical evidence to the limited literature on cloud-based accounting controls and provides actionable recommendations for software developers, business users, accounting professionals, and auditors to enhance internal control effectiveness in cloud computing environments.

Keywords: *Application Controls, Cloud-Based Accounting Software, Information System Audit, Internal Control Effectiveness, Revenue Cycle Management*

Efektivitas Kontrol Aplikasi pada Perangkat Lunak Akuntansi Berbasis Cloud: Analisis Komparatif Mekari dan Xero

ABSTRAK

Penelitian ini mengevaluasi efektivitas kontrol aplikasi pada dua platform akuntansi berbasis cloud yang terkemuka, yaitu Mekari Jurnal ID dan Xero, melalui pengujian black box sistematis terhadap kontrol siklus pendapatan. Dengan menggunakan desain penelitian kualitatif eksplanatori, investigasi ini mengkaji kontrol input (pemeriksaan data yang hilang, validasi numerik-alfabetik, pemeriksaan batas, pemeriksaan rentang, pemeriksaan validitas, pemeriksaan tanda, pemeriksaan urutan) dan kontrol output (pencatatan transaksi, akurasi laporan) di seluruh modul penawaran harga, pesanan penjualan, dan faktur. Pengujian dilakukan antara Oktober 2024 dan Januari 2025 menggunakan akun percobaan dengan dokumentasi kertas kerja yang komprehensif. Temuan mengungkapkan kelemahan kontrol yang signifikan pada kedua platform, termasuk validasi bidang yang tidak memadai yang memungkinkan entri karakter tunggal, penegakan batas persediaan yang tidak memadai yang memungkinkan penjualan berlebih, validasi persentase diskon yang tidak konsisten, tidak adanya protokol penomoran dokumen berurutan, dan kemampuan pencatatan transaksi yang kurang pada Mekari Jurnal ID. Kedua platform menunjukkan validasi data master dan kontrol pemeriksaan tanda yang akurat, tetapi gagal menegakkan integritas urutan atau mencegah entri nilai yang tidak masuk akal secara komersial. Analisis komparatif menunjukkan bahwa Mekari Jurnal ID menunjukkan kontrol preventif berbasis tahap yang lebih kuat, sementara Xero menunjukkan granularitas pencatatan transaksi yang superior. Penelitian ini berkontribusi pada bukti empiris terhadap literatur terbatas mengenai kontrol akuntansi berbasis cloud dan memberikan rekomendasi yang dapat ditindaklanjuti bagi pengembang perangkat lunak, pengguna bisnis, profesional akuntansi, dan auditor untuk meningkatkan efektivitas kontrol internal dalam lingkungan komputasi cloud.

Kata Kunci: *Audit Sistem Informasi, Efektivitas Kontrol Internal, Kontrol Aplikasi, Manajemen Siklus Pendapatan, Perangkat Lunak Akuntansi Berbasis Cloud*

INTRODUCTION

Contemporary business organizations increasingly allocate substantial resources to technological infrastructure alongside traditional tangible assets to enhance operational efficiency and competitive advantage (Syarifuddin & As'ad, 2024). The digital transformation of business processes has fundamentally altered organizational structures, with information technology serving as a critical enabler of enhanced productivity, stakeholder accountability, informed decision-making, and accurate financial reporting (Alsharari & Ikem, 2023; Wijayanti et al., 2024). This technological evolution is particularly evident in the accounting domain, where cloud-based accounting software has emerged as a predominant solution for financial management across organizations of varying scales and industries (Herdiana & Nugrahanti, 2025). The migration from manual to computerized accounting systems represents a paradigmatic shift that extends beyond mere automation, fundamentally transforming the nature of internal controls, audit trails, and financial oversight mechanisms (Kolisnyk & Skliarov, 2025).

The adoption of cloud-based accounting software has accelerated globally due to its accessibility, scalability, and cost-effectiveness compared to traditional on-premise solutions (Lutfi, 2022; Ma et al., 2021). Organizations increasingly favor commercially available Software-as-a-Service (SaaS) accounting platforms over developing proprietary systems, as custom development demands extensive time, technical expertise, and financial investment (Pinasti & Pramuka, 2020). However, this widespread adoption has introduced new complexities in ensuring data integrity, transaction accuracy, and control effectiveness. Unlike manual accounting systems that relied heavily on human oversight and segregation of duties, automated systems integrate multiple business processes and provide real-time transaction processing, necessitating robust application-level controls to prevent, detect, and correct errors (Hall, 2016). The increasing sophistication of these platforms has simultaneously enhanced operational capabilities while introducing potential vulnerabilities in data validation, authorization workflows, and audit trail maintenance that require systematic evaluation.

Empirical research has demonstrated the critical importance of information system auditing in ensuring the effectiveness of internal controls within computerized accounting environments. Nusa (2020) found that the implementation of information system audit procedures resulted in a 36.7% improvement in internal control effectiveness, highlighting the substantial impact of systematic control evaluations on organizational risk management. Similarly, Wijaya and Yulyona (2017) identified strong correlations between audit task complexity (45.9%) and information system characteristics, emphasizing the need for specialized audit methodologies tailored to automated environments. These findings underscore the necessity of rigorous control assessments to maintain data integrity and financial reporting reliability in technology-dependent accounting systems. Furthermore, research in Malaysian enterprises confirmed that accounting

software implementation significantly enhances efficiency, reliability, data quality, and reporting accuracy when accompanied by appropriate control mechanisms (Chong & Nizam, 2018; Gunawan et al., 2023).

Previous studies have explored various dimensions of cloud-based accounting software controls, yet significant gaps remain in the empirical literature. Wahyudi and Antonio (2024) conducted a comparative analysis of application controls in Accurate 5 and Zahir 6 accounting software, establishing a foundation for evaluating control effectiveness in locally developed platforms. Atabayeva et al. (2024) compared 1C and Xero platforms, focusing on feature functionality and user adoption patterns but providing limited insight into systematic control weaknesses. Ganapathy (2024) examined audit trail transparency in cloud-based auditing contexts, identifying challenges in transaction logging and historical data retrieval. However, these studies primarily focused on feature comparisons or general control frameworks without systematically evaluating specific application control weaknesses through rigorous testing methodologies. The lack of comprehensive, empirically grounded assessments of input, processing, and output controls in widely adopted international platforms represents a critical knowledge gap that limits practitioners' ability to identify and mitigate control deficiencies effectively.

Despite the proliferation of cloud-based accounting platforms and their increasing adoption across diverse organizational contexts, systematic empirical evaluations of application controls remain scarce. This research addresses this gap by conducting a comprehensive comparative analysis of application controls in two prominent cloud-based accounting platforms: Mekari Jurnal ID and Xero. Mekari Jurnal ID represents a well-established domestic solution serving over 35,000 businesses across multiple industries in Indonesia as of June 2024, while Xero, established in 2006, has achieved global prominence with 4.2 million active users, particularly in New Zealand, Australia, and the United Kingdom. The selection of these platforms enables an international comparative framework that examines both emerging-market and developed-market solutions, providing broader insights into control implementation patterns across different regulatory and operational contexts. This study employs black box testing methodology to systematically evaluate application controls within the revenue cycle, specifically examining quotation, sales order, and invoice processing stages.

The primary objective of this research is to assess whether information system audit procedures applied to application controls can enhance internal control effectiveness in Mekari Jurnal ID and Xero platforms. This investigation addresses several specific research questions: (1) What are the features and functionality characteristics of each platform? (2) How effective are the input controls embedded in sales transaction processing? (3) How reliable are the output controls governing financial reporting functions? (4) What control enhancements can be recommended to software developers, accounting professionals, and business users? The research

gap addressed by this study lies in the absence of empirical, systematic evaluations of specific control mechanisms using standardized testing protocols across internationally recognized platforms. The novelty of this research manifests in three dimensions: first, it provides empirical evidence of control strengths and weaknesses through systematic black box testing rather than theoretical analysis; second, it establishes a replicable comparative framework applicable to evaluating controls in other cloud-based accounting platforms; third, it extends the limited body of empirical research on application controls in cloud environments by examining platforms with substantial market presence across different geographical and regulatory contexts.

This study contributes to both academic knowledge and professional practice by providing actionable insights for multiple stakeholder groups. For software developers, the findings identify specific control deficiencies requiring remediation to enhance product reliability and market competitiveness. For accounting professionals and auditors, the research offers evidence-based guidance on compensating controls and risk assessment procedures necessary when relying on cloud-based accounting systems. For business users, the study highlights critical control considerations for system selection, configuration, and user acceptance testing processes. By systematically documenting control weaknesses in widely adopted platforms, this research advances the emerging body of empirical literature on information system auditing in cloud-based accounting environments and establishes a foundation for future comparative studies across additional platforms and accounting cycles.

LITERATURE REVIEW

Information System Auditing in Cloud Environments

Information system auditing constitutes a systematic process of gathering and evaluating evidence to determine whether information systems adequately protect organizational assets, maintain data integrity, achieve organizational objectives effectively, and utilize resources efficiently (Hall, 2016). The audit process encompasses comprehensive assessments of whether information systems provide adequate protection against unauthorized access, modification, or destruction of organizational assets while ensuring the provision of accurate, complete, and timely information for decision-making processes. Information system audits encompass two primary control categories: general controls, which apply to the overall information technology environment and include physical security, access controls, system development procedures, backup and recovery processes, and business continuity planning; and application controls, which relate to specific applications and ensure accurate and complete transaction processing (Sangkala, 2024).

The necessity for rigorous information system auditing has intensified as organizations increasingly depend on information technology to perform critical

business functions. The evolution from manual to automated systems has fundamentally transformed control requirements, as manual processes historically relied on human oversight and segregation of duties, whereas automated systems demand robust application-level controls to ensure processing accuracy and completeness. Cloud-based environments introduce additional complexities including data security concerns, vendor dependency, system integration challenges, and reduced visibility into underlying control structures (Alavi & Weiss, 1986). Contemporary cloud environments necessitate formal audit methodologies to establish that application controls adequately mitigate inherent risks in automated transaction processing. Ganapathy (2024) emphasized that audit trail transparency remains a critical challenge in cloud-based auditing, as traditional audit trail mechanisms may be insufficient in distributed computing environments where data processing occurs across multiple geographic locations and vendor-controlled infrastructure.

Application Controls Framework

Application controls represent specific procedures embedded within application software to ensure transaction processing accuracy, completeness, and authorization throughout the transaction lifecycle (Bobbyansyah & Kosasih, 2019). Modern accounting information systems theory advocates for incorporating automated preventive controls into transaction workflows at the point of entry, ensuring compliance and consistency between data preparers and reviewers while minimizing downstream correction and exception handling requirements (Romney et al., 2021). This point-of-entry prevention concept aligns with regulatory frameworks such as the Sarbanes-Oxley Act, which mandates strong internal controls over financial reporting, and enterprise resource planning architectures that incorporate continuous control monitoring across organizational functions (Boddu & Goel, 2025). Automated rule enforcement at the point of entry enables organizations to apply business rules consistently across all transaction inputs, support automated reconciliations, and identify exceptions in high-volume transaction environments.

Application controls are categorized into three fundamental types, each addressing distinct risks within the transaction processing lifecycle (Mascha & Miller, 2024). Input controls prevent or detect errors and irregularities during data entry, employing techniques such as field interrogation (missing data checks, numeric-alphabetic verification, limit checks, range checks, validity checks) and record interrogation (reasonableness checks, sign checks, sequence checks) to ensure data quality before processing commences. Processing controls verify that data manipulation, calculations, and file updates occur accurately during transaction processing, maintaining logical consistency and computational integrity. Output controls ensure that processing results are not lost, misstated, or altered, employing mechanisms such as report distribution controls for batch systems and real-time

validation procedures for immediate processing environments. The theoretical foundation of preventive controls, particularly those embedded at the point of data entry, distinguishes them from detective controls that identify issues retrospectively and corrective controls that address previously detected weaknesses (Romney et al., 2021). Preventive controls preclude errors or irregularities before occurrence, thereby reducing rework, lessening exception handling burdens, and enhancing overall data quality.

Cloud-Based Accounting Software Adoption and Challenges

Cloud-based accounting software represents a paradigmatic shift from traditional on-premises systems to Software-as-a-Service (SaaS) delivery models, fundamentally transforming how organizations manage financial data and processes (Phu et al., 2025). This transformation offers substantial advantages including reduced infrastructure costs, automatic software updates, remote accessibility, scalability, and integration capabilities with complementary business applications. The global adoption of cloud-based accounting solutions has accelerated particularly among small and medium-sized enterprises seeking cost-effective alternatives to traditional systems requiring substantial capital investment and technical expertise (Pinasti & Pramuka, 2020). Platforms such as Mekari Jurnal ID and Xero exemplify this evolution, with Mekari Jurnal ID serving over 35,000 businesses across Indonesia with features customized to local regulatory requirements, while Xero has achieved global prominence with 4.2 million active users across 180 countries through user-friendly interfaces and extensive third-party integrations.

Despite widespread adoption, cloud-based accounting systems introduce unique control challenges that differentiate them from traditional on-premises solutions. These challenges include heightened data security concerns due to data storage on vendor-controlled infrastructure, dependency on vendor stability and service continuity, limitations in system customization capabilities, and critically, reduced visibility into underlying control structures due to the proprietary nature of SaaS platforms (Atabayeva et al., 2024). The black-box nature of cloud-based systems, wherein users cannot access or modify underlying code and control structures, necessitates structured testing methodologies to assess control effectiveness empirically. Wahyudi and Antonio (2024) demonstrated the applicability of systematic audit frameworks to cloud-based accounting platforms through their evaluation of Accurate 5 and Zahir 6 software, establishing a precedent for rigorous control assessments in Indonesian contexts. However, empirical research evaluating application control effectiveness in widely adopted international platforms remains limited, creating a substantial knowledge gap regarding the adequacy of controls in systems processing critical financial data for millions of users globally.

Revenue Cycle Controls in Accounting Information Systems

The revenue cycle constitutes a critical component of accounting information systems, encompassing recurring business activities and information processing operations associated with providing products or services to customers and collecting payment for those deliverables. Romney et al. (2021) identified the fundamental objective of the revenue cycle as contributing to organizational success through revenue generation that exceeds associated costs while ensuring delivery of correct products to customers at appropriate times and prices. The revenue cycle comprises four primary business functions: sales order entry, which processes customer orders received by the sales department; shipping, which involves physical delivery of products to customers; billing, which encompasses invoice creation and transmission to customers; and cash collection, which records payment receipt in the accounting system (Manley & Satiani, 2009). Each stage generates specific documents and data flows requiring proper collection and processing to maintain transaction integrity and financial reporting accuracy.

Effective revenue cycle management demands robust controls ensuring transaction accuracy, fraud prevention, data integrity maintenance, and accurate financial reporting capabilities (Chandawarkar et al., 2024). Common threats within the sales order entry process include order entry errors that diminish operational efficiency and effectiveness, as well as order legitimacy concerns that may facilitate fraudulent transactions. These risks can be mitigated through effective utilization of enterprise resource planning systems incorporating validation controls and digital signature mechanisms for authenticating customer orders. The sales order document serves as the foundation for all subsequent revenue cycle processing, containing essential elements including order numbers, customer identification, transaction dates, purchase order references, and detailed quantity and pricing information for each product. The complexity of modern cloud-based accounting systems processing revenue cycle transactions necessitates systematic identification of control vulnerabilities to prevent adverse effects on transaction accuracy, fraud prevention, data integrity, and financial reporting reliability (Widyadhana & Kirana, 2025). Contemporary research has identified gaps in understanding how cloud-based platforms implement fundamental revenue cycle controls, particularly regarding input validation, inventory management, and sequential documentation protocols that form the cornerstone of effective revenue cycle control frameworks.

Black Box Testing Methodology for Control Assessment

Black box testing methodology represents an empirical approach to evaluating system behavior and control effectiveness by examining inputs and outputs without access to internal code structures, algorithms, or implementation details (Mintarsih, 2023). This methodology aligns with the practical circumstances faced by most accounting software users and auditors, who lack access to

proprietary source code and must assess control reliability based on observable system responses to various input conditions. Black box testing enables systematic evaluation of whether systems enforce stated control objectives by submitting test transactions designed to violate control parameters and observing whether the system appropriately prevents, detects, or permits such violations (Narizzano et al., 2020; Zhang et al., 2018). This approach proves particularly valuable in cloud-based environments where the SaaS delivery model inherently restricts user access to underlying system architecture and control implementation mechanisms.

The application of black box testing to accounting information systems involves developing comprehensive test cases addressing specific control objectives across input, processing, and output domains (Utomo et al., 2020). Typical test scenarios include attempting to save transactions with incomplete required fields (missing data checks), entering alphabetic characters in numeric fields (data type validation), exceeding logical limits such as inventory quantities or discount percentages (limit and range checks), referencing non-existent master file records (validity checks), and evaluating output accuracy through transaction log examination and unique identifier verification (Menghi et al., 2025). Working papers documenting each test case serve as audit evidence, capturing test objectives, scope definitions, detailed procedures, findings with supporting documentation, and pass/fail determinations. This structured approach to control testing enables replicable assessments that can be applied consistently across different platforms, facilitating comparative analyses of control effectiveness. The methodology's emphasis on observable system behavior rather than theoretical control design aligns with the practical requirements of auditors and users who must assess control reliability in operational environments where compensating controls and procedural safeguards may be necessary to address identified deficiencies.

RESEARCH METHODOLOGY

This study employed a qualitative explanatory research design to assess application controls in two cloud-based accounting platforms: Mekari Jurnal ID and Xero (Creswell & Creswell, 2022). Mekari Jurnal ID was selected as the domestic platform due to its widespread adoption across Indonesia, serving over 35,000 businesses with certifications from Fintech Indonesia and the Indonesian Ministry of Communication and Information Technology. Xero was chosen as the international comparison platform, representing a global market leader with 4.2 million active users across 180 countries. The research was conducted between October 16, 2024, and January 22, 2025, utilizing free trial accounts with access periods of seven days for Mekari Jurnal ID and thirty days for Xero.

The study applied black box testing methodology to evaluate application controls within the revenue cycle, specifically examining quotation, sales order, and invoice modules (Hassija et al., 2024). This approach was selected because it

reflects the practical experience of typical users and auditors who lack access to proprietary source code and must assess control effectiveness based on observable system behavior. The testing framework comprised nine distinct control tests: seven input control tests (missing data check, numeric-alphabetic check, limit check, range check, validity check, date check, and currency check) and two output control tests (transaction log verification and unique identifier validation). Each test was designed to evaluate whether the systems appropriately prevented, detected, or permitted control violations under various input conditions.

Comprehensive working papers were developed for each test case, documenting the software name, test objective, scope of the revenue cycle module tested, detailed testing procedures, empirical findings with supporting screenshots, and pass or fail determinations. The working papers served as audit evidence supporting the findings and recommendations presented in this research. The testing procedures and documentation protocols followed established information system auditing methodologies and control objectives derived from accounting information systems literature (Romney et al., 2018). This systematic approach enabled replicable assessments and facilitated comparative analysis of control effectiveness across both platforms.

RESULT AND DISCUSSION

Platform Overview and Revenue Cycle Architecture

Mekari Jurnal ID, developed by PT Mid Solusi Nusantara, represents Indonesia's leading cloud-based accounting solution, serving over 35,000 businesses with certifications from Fintech Indonesia and the Indonesian Ministry of Communication and Information Technology. The platform holds ISO 27001 certification for data security standards and maintains 800,000 active users across diverse industries. The revenue cycle architecture comprises three primary modules: Sales Quotation, which enables businesses to provide product information and pricing proposals to influence customer purchasing decisions; Sales Order, which processes confirmed orders and allocates inventory; and Sales Invoice, which generates billing documents and records accounts receivable transactions.

Xero, founded in 2006 in New Zealand, has achieved global market leadership with 4.2 million active users across 180 countries, particularly dominating markets in New Zealand, Australia, and the United Kingdom. The platform is recognized for user-friendly interfaces, extensive third-party application integrations, and comprehensive accounting capabilities tailored to small and medium-sized businesses. The revenue cycle implementation mirrors Mekari Jurnal ID's modular structure through Quotes, Sales Orders, and Invoices modules, supplemented by robust customer relationship management features, multi-currency support, diverse tax regime compatibility, and integration with global payment processors. Both platforms provide cloud-based access, automatic data

backup, real-time financial reporting, and mobile application support, yet demonstrate significant variations in regional focus, regulatory compliance features, user interface design, and specific control implementation approaches.

Input Control Assessment

1. Missing Data Check Results

Comparative testing of missing data controls revealed both similarities and critical differences between the platforms. Table 1 presents the comprehensive findings.

Table 1 Missing Data Check Control Comparison

Control Dimension	Mekari Jurnal ID Result	Xero Result	Comparative Assessment
Customer/Counterparty Field	Customer field mandatory; system blocks processing when empty across all modules	"To" field mandatory; system blocks saving when empty	Equal - both systems consistently enforce counterparty requirements
Product/Item Field	Product field mandatory; cannot save transactions when empty	Item field mandatory; cannot save when empty	Equal - identical preventive enforcement
Quantity When Left Blank	Quotation: auto-set to 0; Sales order/invoice: must be >0 or blocked	Auto-set to 1 when blank, but can remain 0 without blocking	Mekari Jurnal ID - stricter validation at financially binding stages
Unit Price When Left Blank	Mandatory; auto-set to 0 when blank	Mandatory; auto-set to 0 when blank	Equal - identical default behavior
Date Fields	Transaction and expiration dates mandatory and auto-generated	Issue and due dates mandatory and auto-generated	Equal - both ensure date completeness

Source: Author’s Analysis (2025)

2. Limit Check and Range Check Findings

Limit checking revealed critical control failures across both platforms, particularly regarding inventory management and discount validation. Both systems permit sales transactions that exceed available inventory quantities, creating substantial risks for revenue recognition accuracy, inventory management integrity, customer satisfaction, and financial reporting reliability. Organizations may inadvertently commit to product deliveries they cannot fulfill, resulting in backorder

complications, customer complaints, and potential revenue recognition violations under accounting standards requiring reasonable assurance of collectability and deliverability.

Discount percentage validation demonstrated inconsistent enforcement patterns across transaction types and entry methods. While both platforms prevent discount percentages exceeding 100% in certain modules, testing identified scenarios where unreasonable discount amounts were accepted, creating vulnerabilities for fraudulent transaction processing and pricing errors. Credit limit enforcement presented additional concerns, as both systems permitted transaction processing for customers exceeding credit limits when payment instruments involved post-dated checks or promissory notes. The systems incorrectly treated future-dated payment instruments as immediate payment, marking invoices as paid and bypassing credit limit restrictions. This deficiency creates elevated credit risk exposure and potential bad debt losses.

Table 2 Limit Check Control Comparison

Control Dimension	Mekari Jurnal ID Result	Xero Result	Comparative Assessment
Quantity Limit at Quotation	Unlimited quantity permitted	Not tested at quotation stage	Context-dependent - acceptable for non-binding quotations
Quantity Limit at Invoice	Mixed results: blocks quantities exceeding sales order but allows exceeding physical stock	Allows quantities exceeding physical stock	Mekari Jurnal ID (slightly) - partial order linkage provides constraint
Credit Limit at Sales Order	System rejects transactions exceeding credit limit	Not tested	Mekari Jurnal ID - preventive control before billing occurs
Credit Limit at Invoice	Inconsistent: bypassed when post-dated payment instruments used	Similar bypass behavior with pending instruments	Equal (both weak) - insufficient payment instrument integration

Source: Author's Analysis (2025)

Range check testing revealed that both platforms technically accept values within system-defined boundaries but lack controls preventing commercially unreasonable entries. Both systems permit prices as low as 1 Rupiah, which while technically valid, may indicate data entry errors or fraudulent activities. The absence of warning messages or

confirmation requirements for extreme values represents a control weakness requiring compensating manual review procedures. Negative value testing demonstrated that both platforms appropriately prevent negative quantities and prices in standard input fields; however, adjustment and correction features permitted negative values without adequate audit trail documentation, potentially facilitating unauthorized transaction reversals or financial statement manipulation.

3. Sign Check and Sequence Check Results

Sign check testing demonstrated robust preventive controls in both platforms, with comprehensive rejection of negative values for quantities, prices, and discount percentages across all tested transaction stages. Both systems effectively prevent invalid sign entries through preventive blocking mechanisms rather than corrective adjustments, ensuring data integrity at the point of entry. Mekari Jurnal ID demonstrated broader testing coverage across quotation, order, and invoice stages, providing evidence of consistent sign enforcement throughout the sales cycle, while Xero's testing scope focused on the invoicing module with equivalent strictness within that narrower scope.

Table 3 Sign Check Control Comparison

Control Dimension	Mekari Jurnal ID Result	Xero Result	Comparative Assessment
Quantity Sign Validation	Negative quantities rejected across all modules	Negative quantities rejected	Equal - effective prevention
Unit Price Sign Validation	Negative prices rejected across all modules	Negative prices rejected	Equal - identical enforcement
Discount Sign Validation	Negative discount percentages rejected	Negative discount percentages rejected	Equal - consistent prevention
Consistency Across Stages	Consistent behavior across quotation, order, invoice	Observed at invoicing stage only	Mekari Jurnal ID - multi-stage consistency

Source: Author's Analysis (2025)

Sequence check testing revealed significant control weaknesses in both platforms, representing one of the most critical deficiencies identified in this study. Both systems permit users to create and process documents using non-sequential numbering without issuing warnings or implementing automated sequence integrity enforcement. Users can skip large ranges of document numbers, creating substantial risks for

transaction completeness verification and audit trail integrity. Neither platform provides automated exception reporting when document number sequences contain gaps, placing heavy reliance on manual monitoring and procedural controls to detect missing, duplicate, or modified transactions. This deficiency fundamentally undermines auditors' ability to verify that all transactions have been captured and processed in chronological order, a cornerstone principle of effective internal control frameworks.

Output Control Assessment

1. Transaction Logging and Audit Trail Capabilities

Transaction log examination revealed substantial differences in audit trail comprehensiveness between the platforms. Mekari Jurnal ID maintains transaction logs that record only the most recent modification to each transaction, lacking detailed field-level change history. When multiple sequential modifications occur, only the final state is preserved, eliminating visibility into intermediate changes and the evolution of transaction attributes over time. This limitation significantly impairs forensic investigation capabilities, accountability establishment, and regulatory compliance demonstrations for organizations subject to audit requirements.

Conversely, Xero implements comprehensive transaction logging that captures detailed field-level change history, preserving complete modification sequences with timestamps and user attribution. Auditors and management can reconstruct the complete chronology of transaction modifications, identifying what changed, who made the change, and when the modification occurred. This granular audit trail substantially enhances post-transaction review capabilities, accountability mechanisms, and traceability of all modifications throughout the transaction lifecycle.

Table 4 Transaction Log Control Comparison

Control Dimension	Mekari Jurnal ID Result	Xero Result	Comparative Assessment
Existence of Transaction Logs	Logs exist and accessible at transaction forms	Logs exist and accessible at invoice forms	Equal - both provide visible logs
Level of Detail in Logs	Records only most recent change without field-level history	Records detailed change history with specific fields modified	Xero - superior granularity and audit utility
Coverage of Data Changes	Additions and changes logged but historical	Additions and successive changes logged with traceable details	Xero - comprehensive change coverage

Control Dimension	Mekari Jurnal ID Result	Xero Result	Comparative Assessment
	sequences not preserved		
Support for Accountability	Limited due to lack of detailed change history	Strong support through detailed chronological reconstruction	Xero - enhanced accountability mechanisms
Overall Pass/Fail Outcome	Failed - insufficient detail in transaction logs	Passed - detailed logs satisfy control objectives	Xero - meets audit trail requirements

Source: Author's Analysis (2025)

The comparative results clearly demonstrate that while both systems maintain transaction logs, effectiveness varies substantially based on granularity and completeness of captured information. The transaction log's value for internal control purposes depends not merely on existence but on the detail level and comprehensiveness of change histories recorded. Systems capable of retaining detailed, field-level transaction logs provide superior assurance regarding auditability and governance compared to systems recording only final transaction states.

2. Report Accuracy and Distribution Controls

End-user control testing revealed significant differences in report accuracy between platforms. Both systems correctly generated accurate total sales reports when compared to independent transaction records, confirming that top-level aggregation logic functions equivalently. However, substantial divergence emerged in lower-level sales reporting accuracy. Mekari Jurnal ID demonstrated superior performance by passing end-user control tests for both total sales reports and customer-level sales reports, failing only at product-level reporting. Xero failed end-user control tests for both customer-level and product-level sales reports, indicating broader deficiencies in segmentation logic congruence.

The primary source of reporting inaccuracies in both platforms stemmed from inconsistent treatment of sales returns and discounts in segmentation reports. Both systems treated credit memos as gross amounts rather than net of discounts when disaggregating sales by customer or product dimensions, creating mathematical discrepancies between aggregated totals and segmented subtotals. Additional inconsistencies included rounding differences between detail and summary reports and periodic discrepancies between screen display and report export functionality. Although many identified errors involved immaterial amounts, they reflect underlying weaknesses in report

generation control logic that may manifest differently depending on transaction volume and complexity.

Table 5 Summary of Application Control Effectiveness

Control Type	Mekari Jurnal ID Key Outcome	Xero Key Outcome	Overall Assessment
Missing Data Check	Strong stage-based enforcement; quantity >0 at binding stages	Baseline enforcement; quantity may remain 0	Mekari Jurnal ID - stronger preventive control
Limit Check	Credit limit enforced at sales order; weaknesses at invoice with post-dated instruments	Similar invoice-stage weaknesses; no earlier enforcement tested	Mekari Jurnal ID (slightly) - earlier-stage enforcement
Range Check	Discounts bounded; chronological date logic enforced; price lower bound weak	Discounts bounded; weaker date sequence; price lower bound weak	Mekari Jurnal ID - stronger temporal consistency
Sign Check	Negative values blocked across all sales stages	Negative values blocked at invoice stage	Equal - equivalent strictness
Sequence Check	Non-sequential document numbers allowed across all stages	Non-sequential invoice numbers allowed	Equal (both weak) - no automated sequence integrity
Transaction Log	Logs exist but only last change recorded; limited detail	Detailed, field-level change history recorded	Xero - superior audit trail depth
Report Accuracy	2 of 3 reports accurate; weakness at product-level only	1 of 3 reports accurate; weaknesses at customer and product levels	Mekari Jurnal ID - higher accuracy

Source: Author's Analysis (2025)

Report distribution controls demonstrated mature implementation in both platforms, with no evidence of unauthorized access or control failures during testing. Mekari Jurnal ID employs role-based access control through customizable user roles defined by administrators, while Xero implements permission-based access control through staff invitation and explicit permission assignment mechanisms. Despite architectural differences, both approaches effectively establish authorization boundaries and support segregation of duties requirements. Residual risk in report distribution controls depends primarily on administrative governance quality rather than system design limitations.

Analysis of Control Patterns and Weaknesses

The examination of application controls in Mekari Jurnal ID and Xero reveals complex patterns of control strengths and weaknesses characteristic of contemporary cloud-based accounting software. The identified inadequacies span multiple control categories and affect fundamental aspects of transaction processing, data integrity maintenance, and financial reporting reliability. The high frequency of input control weaknesses, particularly regarding numeric-alphabetic validation and limit checks, suggests that software developers prioritize user experience and functional flexibility over control rigor. The acceptance of single-character entries in critical fields, while preventing completely blank entries, fails to establish adequate data quality standards for business operations.

This pattern aligns with Alavi and Weiss's (1986) foundational observation that end-user computing introduces new risks requiring systematic management approaches. Cloud-based platforms, designed for diverse user populations with varying technical expertise, intentionally implement permissive controls to avoid frustrating users through strict validation processes. However, this design philosophy creates material control gaps that transfer risk from the software layer to organizational procedures and user discipline. The inability to enforce inventory limits before completing sales transactions represents a particularly serious control weakness with direct implications for revenue recognition compliance, inventory management accuracy, and customer relationship maintenance.

The identified control deficiencies suggest that market dynamics prioritizing user-friendly interfaces and feature richness exert stronger influence on purchasing decisions than control effectiveness, which remains less evident to non-technical users during evaluation processes. Software vendors responding to competitive pressures may consciously or unconsciously sacrifice control rigor to enhance perceived usability and feature comprehensiveness. This market reality necessitates that organizations implementing cloud-based accounting systems recognize the limitations of embedded controls and proactively develop compensating manual controls, supervisory reviews, and independent reconciliation procedures to establish effective internal control frameworks.

The findings of this study demonstrate substantial alignment with previous research examining application controls in cloud-based accounting environments while extending the empirical evidence base to additional platforms and international contexts. Wahyudi and Antonio (2024) identified similar control weaknesses in their comparative analysis of Accurate 5 and Zahir 6 accounting software, including inadequate field validation, insufficient inventory limit enforcement, and inconsistent discount percentage validation. Their research documented that both platforms permitted single-character entries in critical fields and failed to enforce inventory constraints during sales order processing, findings that closely parallel the deficiencies identified in Mekari Jurnal ID and Xero. This consistency across multiple platforms suggests that the observed control

weaknesses represent systemic challenges in cloud-based accounting software development rather than platform-specific deficiencies.

The convergence of findings across different studies indicates that fundamental tensions between user experience optimization and control rigor pervade the cloud-based accounting software industry. Wahyudi and Antonio (2024) attributed these patterns to developers' prioritization of interface usability and feature accessibility over validation strictness, a conclusion reinforced by the current study's findings. Additionally, both studies identified transaction logging deficiencies as significant audit trail concerns, with varying degrees of severity across platforms. The current study's finding that Xero maintains superior transaction logging capabilities compared to Mekari Jurnal ID contrasts with Wahyudi and Antonio's observation that both Accurate and Zahir demonstrated limited change history preservation, suggesting that international platforms serving multiple regulatory jurisdictions may face stronger audit trail requirements than domestic-focused solutions.

Ganapathy (2024) emphasized audit trail transparency as a critical challenge in cloud-based auditing contexts, noting that traditional audit trail mechanisms may prove insufficient in distributed computing environments where data processing occurs across multiple geographic locations and vendor-controlled infrastructure. The current study's findings regarding sequence check deficiencies and transaction logging limitations in both Mekari Jurnal ID and Xero substantiate Ganapathy's concerns. The inability of both platforms to enforce sequential document numbering or automatically report sequence gaps fundamentally compromises audit trail completeness verification, a cornerstone requirement for effective internal control frameworks. Furthermore, Ganapathy's identification of limited visibility into vendor-controlled infrastructure as an audit challenge aligns with this study's application of black box testing methodology, which reflects the practical constraints auditors face when evaluating proprietary cloud-based systems. The systematic control weaknesses identified across multiple studies collectively demonstrate that current cloud-based accounting platforms, despite their operational advantages, require substantial compensating controls to achieve internal control effectiveness comparable to traditional on-premises systems with greater user visibility and customization capabilities.

CONCLUSION

This study conducted a comprehensive evaluation of application controls in two prominent cloud-based accounting platforms, Mekari Jurnal ID and Xero, through systematic black box testing of input and output controls across revenue cycle modules. The empirical findings revealed significant control weaknesses in both platforms affecting data integrity, transaction processing accuracy, and financial reporting reliability. Critical deficiencies identified include inadequate

field validation permitting single-character entries in essential fields, insufficient inventory limit enforcement enabling sales exceeding available stock, inconsistent discount percentage validation creating fraud vulnerabilities, absence of sequential document numbering protocols compromising audit trail integrity, and inaccurate segmentation reporting logic potentially misleading management decision-making. These findings demonstrate that widely adopted cloud-based accounting platforms contain material control gaps requiring compensating manual controls and organizational diligence to ensure effective internal control frameworks. The research contributes empirical evidence to the limited body of literature examining application control effectiveness in cloud-based accounting environments and establishes a replicable testing framework applicable to evaluating additional platforms and accounting cycles.

The practical implications of these findings extend to multiple stakeholder groups requiring differentiated responses to identified control weaknesses. Software developers should implement enhanced validation rules, enforce inventory constraints, improve transaction logging mechanisms capturing detailed modification histories, and provide configurable control parameters enabling organizations to balance user experience with control rigor according to their risk tolerance. Business users must recognize that embedded software controls prove insufficient for establishing effective internal control systems and should implement compensating controls including supervisory review procedures, periodic data quality assessments, independent reconciliation processes, and comprehensive user training emphasizing data integrity importance. Accounting professionals and auditors should apply heightened professional skepticism when relying on system-generated information, conduct enhanced substantive testing procedures when control weaknesses are identified, utilize data analytics to validate report accuracy and detect unusual transaction patterns, and develop platform-specific audit programs addressing known control deficiencies in commonly deployed cloud-based accounting systems.

This research acknowledges several limitations presenting opportunities for future investigation. The study focused exclusively on revenue cycle controls without evaluating other accounting cycles including purchasing, payroll, and fixed asset management, suggesting future research should extend the audit framework to provide comprehensive control assessments across all accounting functions. The evaluation encompassed only two platforms among numerous globally available cloud-based accounting solutions, indicating future studies should examine additional platforms to determine whether identified weaknesses represent systemic industry challenges or platform-specific deficiencies. Methodological constraints included reliance on free-trial accounts with time-limited access potentially restricting exposure to advanced features, application of black box testing methodology precluding evaluation of underlying code and development processes, and absence of user behavior and organizational compensating control assessments.

Future research employing mixed methodologies incorporating developer interviews, full platform access, and field studies examining organizational implementation practices would provide deeper insights into control weakness causation and effective remediation strategies, ultimately advancing both theoretical understanding and practical guidance for achieving reliable financial reporting in cloud-based accounting environments.

REFERENCES

- Alavi, M., & Weiss, I. R. (1986). Managing the Risks Associated with End-User Computing on JSTOR. *Journal of Management Information Systems*, 2(3), 5–20.
- Alsharari, N. M., & Ikem, F. (2023). Digital accounting systems and information technology in the public sector: mutual interaction. *Journal of Systems and Information Technology*, 25(1), 53–73. <https://doi.org/10.1108/JSIT-09-2021-0190>
- Atabayeva, A. ., Akynov, D. ., Ovcharenko, I. ., Agabekova, G. ., & Agabekova, S. N. (2024). Using cloud platforms in accounting: comparison of 1C and Xero. *Bulletin of the Karaganda University Economy Series*, 29(1), 216–231. <https://doi.org/10.31489/2024ec1/216-231>
- Bobbyansyah, B., & Kosasih, E. (2019). Analisis Kontrol Aplikasi Untuk Menghasilkan Keakuratan Dalam Kelengkapan Informasi Pada Siklus Pendapatan Adorable Projects. *Jurnal Akuntansi Maranatha*, 11(2), 230–247. <https://doi.org/10.28932/jam.v11i2.1914>
- Boddu, N., & Goel, O. (2025). SOX Compliance in Oracle ERP Systems. *International Journal of Research in Modern Engineering & Emerging Technology*, 13(4), 160–171. <https://doi.org/10.63345/ijrmeet.org.v13.i4.9>
- Chandawarkar, R., Nadkarni, P., Barmash, E., Thomas, S., Capek, A., Casey, K., & Carradero, F. (2024). Revenue Cycle Management: The Art and the Science. *Plastic and Reconstructive Surgery - Global Open*, 12(7), e5756. <https://doi.org/10.1097/GOX.0000000000005756>
- Chong, Y., & Nizam, I. (2018). THE IMPACT OF ACCOUNTING SOFTWARE ON BUSINESS PERFORMANCE. *ISSN: 22897615 Page 1 International Journal of Information System and Engineering* , 6(1).
- Creswell, J. W., & Creswell, J. D. (2022). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (Sixth Edition). SAGE Publications, Inc. <https://uk.sagepub.com/en-gb/eur/research-design/book270550>
- Ganapathy, V. (2024). COLLABORATIVE AUDITING IN OSMOTIC COMPUTING ENVIRONMENTS: OPPORTUNITIES AND CHALLENGES. *Shodh Sari-An International Multidisciplinary Journal*, 03(02), 138–166. <https://doi.org/10.59231/SARI7693>
- Gunawan, S., Wibowo, E. W., Wahyu Wibowo, E., Siswardhani Wahjono, D., & Harahap, S. (2023). PROSEDUR QUALITY CONTROL PADA PT JST INDONESIA. *ESENSI: Jurnal Manajemen Bisnis*, 26(1), 27–33. <https://esensijournal.com/index.php/esensi/article/view/4>
- Hall, J. A. (2016). Information Technology Auditing. In *Kesehatan Masyarakat*

- (Fourth Edition). Cengage.
- Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., Scardapane, S., Spinelli, I., Mahmud, M., & Hussain, A. (2024). Interpreting Black-Box Models: A Review on Explainable Artificial Intelligence. *Cognitive Computation*, 16(1), 45–74. <https://doi.org/10.1007/s12559-023-10179-8>
- Herdiana, K., & Nugrahanti, T. P. (2025). DIGITAL AUDIT TECHNOLOGIES IN ACCOUNTING INFORMATION SYSTEMS: A SYSTEMATIC LITERATURE REVIEW OF CONTEMPORARY TRENDS AND PROFESSIONAL IMPLICATIONS. *Journal of Entrepreneurship*, 4(2), 87–101. <https://doi.org/10.56943/joe.v4i2.778>
- Kolisnyk, O., & Skliarov, D. (2025). TRANSFORMATION OF FINANCIAL ACCOUNTING: FROM TRADITIONAL METHODS TO DIGITALIZATION. *Business Navigator*, 2(79). <https://doi.org/10.32782/business-navigator.79-36>
- Lutfi, A. (2022). Understanding the Intention to Adopt Cloud-based Accounting Information System in Jordanian SMEs. *The International Journal of Digital Accounting Research*, 47–70. https://doi.org/10.4192/1577-8517-v22_2
- Ma, D., Fisher, R., & Nesbit, T. (2021). Cloud-based client accounting and small and medium accounting practices: Adoption and impact. *International Journal of Accounting Information Systems*, 41, 100513. <https://doi.org/10.1016/j.accinf.2021.100513>
- Manley, R., & Satiani, B. (2009). Revenue cycle management. *Journal of Vascular Surgery*, 50(5), 1232–1238. <https://doi.org/10.1016/j.jvs.2009.07.065>
- Mascha, M. F., & Miller, C. L. (2024). Applying Internal Control Concepts Using Database Design: An Educational Case. *AIS Educator Journal*, 19(1), 1–16. <https://doi.org/10.3194/1935-8156-19.1.1>
- Menghi, C., Balai, E., Valovcin, D., Stickssel, C., & Rajhans, A. (2025). Completeness and Consistency of Tabular Requirements: An SMT-Based Verification Approach. *IEEE Transactions on Software Engineering*, 51(2), 595–620. <https://doi.org/10.1109/TSE.2025.3530820>
- Mintarsih, M. (2023). Pengujian Black Box Dengan Teknik Transition Pada Sistem Informasi Perpustakaan Berbasis Web Dengan Metode Waterfall Pada SMC Foundation. *Jurnal Teknologi Dan Sistem Informasi Bisnis*, 5(1), 33–35. <https://doi.org/10.47233/jteksis.v5i1.727>
- Narizzano, M., Pulina, L., Tacchella, A., & Vuotto, S. (2020). *Automated Requirements-Based Testing of Black-Box Reactive Systems* (pp. 153–169). https://doi.org/10.1007/978-3-030-55754-6_9
- Nusa, I. B. S. (2020). Quality of Audit System Information for Internal Control Effectiveness. *Advances in Economics, Business and Management Research*. <https://doi.org/10.2991/aebmr.k.200108.046>
- Phu, G. N., Thi, T. H., & Bich, H. T. N. (2025). The impact of cloud computing technology on cloud accounting adoption and financial management of businesses. *Humanities and Social Sciences Communications*, 12(1), 851. <https://doi.org/10.1057/s41599-025-05190-3>
- Pinasti, M., & Pramuka, B. A. (2020). Does Cloud-Based Accounting Information System Harmonize the Small Business Needs? *Journal of Information and Organizational Sciences*, 44(1), 141–156.

- <https://doi.org/10.31341/jios.44.1.6>
- Romney, : Marshall B, Steinbart, P. J., L.Summers, S., & Wood, D. A. (2021). Accounting Information Systems. In *ICB Research Reports* (Issue 9). Pearson.
- Sangkala, M. (2024). CONCEPT AND APPLICATION OF AUDIT IN INFORMATION SYSTEMS. *MARGINAL JOURNAL OF MANAGEMENT ACCOUNTING GENERAL FINANCE AND INTERNATIONAL ECONOMIC ISSUES*, 3(3), 770–781. <https://doi.org/10.55047/marginal.v3i3.1193>
- Syarifuddin, S., & As'ad, A. (2024). The Role of Technology and Infrastructure in Improving Operational Efficiency. *Bata Ilyas Educational Management Review*, 4(2), 14–29. <https://doi.org/10.37531/biemr.v4i2.2414>
- Utomo, A., Sutanto, Y., Tiningrum, E., & Susilowati, E. M. (2020). PENGUJIAN APLIKASI TRANSAKSI PERDAGANGAN MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS. *Jurnal Bisnis Terapan*, 4(2), 133–140. <https://doi.org/10.24123/jbt.v4i2.2170>
- Wahyudi, F., & Antonio, G. R. (2024). IMPLEMENTATION OF INFORMATION SYSTEM AUDIT TO IMPROVE INTERNAL CONTROL ON ACCURATE 5 AND ZAHIR 6. *Journal of Entrepreneurship*, 3(1), 55–71. <https://doi.org/10.56943/joe.v3i1.474>
- Widyadhana, K. A., & Kirana, R. T. (2025). Cloud-Based Accounting Systems : Solusi Transparansi Informasi Keuangan atau Hanya Ilusi Kecanggihan Teknologi ? *Jurnal Ekonomi, Bisnis Dan Manajemen*, 4(1), 144–157. <https://doi.org/10.58192/ebismen.v4i1.3169>
- Wijaya, I. A., & Yulyona, M. T. (2017). Does Complexity Audit Task, Time Deadline Pressure, Obedience Pressure, and Information System Expertise Improve Audit Quality? *International Journal of Economics and Financial Issues*, 7(3), 398–403. <https://www.econjournals.com/index.php/ijefi/article/view/4643>
- Wijayanti, P., Mohamed, I. S., & Daud, D. (2024). Computerized accounting information systems: An application of task technology fit model for microfinance. *International Journal of Information Management Data Insights*, 4(1), 100224. <https://doi.org/10.1016/j.jjime.2024.100224>
- Zhang, H., Feng, L., Wu, N., & Li, Z. (2018). Integration of Learning-Based Testing and Supervisory Control for Requirements Conformance of Black-Box Reactive Systems. *IEEE Transactions on Automation Science and Engineering*, 15(1), 2–15. <https://doi.org/10.1109/TASE.2017.2693995>