

Design and Implementation of RPA for Financial Audit and Notification Automation

Jovita Junilla, Jessica Selistio, Qabil Putra Sandi and Ali Gunawan

Abstract— This study deals with the inefficiencies found in manual financial audit and payment notification processes at educational institutions, where the administrative staff have to manually reconcile outstanding payment and overdue unpaid records from multiple systems. These activities are repetitive, time-consuming and prone to human error resulting in delayed payment reminders and increased operational workload. To address this limitation, this study proposes a non-invasive audit-to-action RPA workflow that combines rule-based financial reconciliation, dynamic parent-contact synchronization from Google Sheets, and a hierarchical WhatsApp notification mechanism into a coordinated cross-platform process without modifying the institution's existing information systems. Unlike conventional RPA implementations that automate isolated administrative tasks, the proposed workflow enables validated audit outcomes to directly trigger operational actions without modifying the institution's existing information systems. A case study was conducted at XYZ School which involved 898 student payment records. The results show that the proposed workflow reduced from 1,897 minutes to 387 minutes, achieving approximately a 4.9-fold improvement over the manual process. Moreover, the recurring direct operational cost was reduced by approximately 46-fold, while the system reached 100% validation accuracy during User Acceptance Testing and a 93.0% notification delivery success rate in real operational conditions. The findings demonstrate that the proposed audit-to-action workflow provides a reusable and non-invasive approach for coordinating financial validation and automated follow-up across existing educational systems.

Index Terms—Automation, Business Process Management, Workflow Management Software, Decision Support Systems

I. INTRODUCTION

THE digital transformation of educational institutions leaves many administrative tasks, particularly financial audits and payment monitoring [1] are still carried out manually, causing inefficiencies and the risk of human error when dealing with large amounts of student data. Although automation technologies have improved administrative efficiency, repetitive financial processes remain challenging to automate [2]. Many institutions also lack an integrated mechanism to separate overdue from non-overdue arrears,

requiring staff to manually match records to determine payment status and reminder priority.

Robotic Process Automation (RPA) is a possible solution that uses software robots to imitate human interaction with digital systems to automatically perform rule-based tasks [3], [4]. RPA has been found to improve accuracy and efficiency of processing in accounting and finance [5] and to support administrative processes such as validating student data [6], [7]. It is also non-invasive, and can be integrated with existing systems, without changing the underlying architecture [8]. Most previous work, however, automates isolated tasks rather than integrating financial validation, dynamic contact synchronization, and follow-up notification into a continuous workflow [9]. Many educational systems also record outstanding and overdue balances separately, requiring staff to determine which payments require a reminder.

To validate the proposed audit-to-action workflow in a real operational environment, XYZ School was selected as the case study. Although the school uses a web-based administration system, its financial audit and payment reminder process remains manual. Staff compare outstanding balances with unpaid records in Excel before sending reminder messages through WhatsApp Desktop. This process delays reminders, increases administrative workload, and may result in payment penalties. The institution therefore requires a non-invasive automation approach without modifying the existing portal.

To address this research gap, this paper proposes a non-invasive audit-to-action RPA workflow consisting of three processes: data extraction and consolidation, financial validation and synchronization, and automated notification and logging. Unlike conventional RPA implementations that automate isolated tasks, the workflow reconciles outstanding and overdue payment records, synchronizes the latest parent contact information from Google Sheets, and delivers WhatsApp reminders through a hierarchical notification logic across the School Portal, Excel, Google Sheets, and WhatsApp Desktop without modifying existing information systems.

The main contributions of this paper are threefold: (1) a reusable, non-invasive RPA workflow that integrates financial reconciliation, dynamic contact synchronization, and automated notification across existing educational information systems; (2) a dynamic parent-contact synchronization and hierarchical WhatsApp notification mechanism that enables validated audit results to trigger follow-up actions; and (3)

¹Manuscript received July 25, 2026.

J. Junilla, J. Selistio, Q. P. Sandi, and A. Gunawan are with the School of Information Systems, Information Systems Department, Bina Nusantara University, Jakarta 11480, Indonesia (e-mail: jovita.junilla@binus.ac.id, jessica.selistio@binus.ac.id, qabil.sandi@binus.ac.id, gunlee77@binus.ac.id).

empirical validation of the proposed workflow through a real-world case study, demonstrating significant improvements in processing efficiency and operational cost.

II. LITERATURE REVIEW

A. Digital Transformation Theory

Digital transformation refers to organizational change driven by digital technologies to create new business models and generate additional value [10]. In educational institutions, digital transformation can improve administrative processes by reducing manual work and enabling more efficient information flow and coordination across organizational activities. In this context, automation technologies such as RPA can support the transformation of repetitive administrative processes without requiring major changes to existing information systems [11].

B. Robotic Process Automation (RPA)

RPA uses software bots to imitate human interactions with digital systems to perform repetitive and rule-based tasks [12]. It is particularly suitable for standardized, high-volume processes involving structured data and predictable activities.

Unlike conventional system integration, RPA operates through existing user interfaces without modifying the underlying information systems [9]. This non-invasive characteristic enables organizations to automate legacy systems without major system changes. However, RPA is less suitable for processes involving unstructured data, frequent exceptions, or high variability.

C. Rule-Based Workflow Automation

Rule-based workflow automation coordinates activities using predefined business rules and explicit decision logic, allowing the output of one activity to trigger subsequent processes automatically. This approach is suitable for repetitive administrative processes involving structured data and predictable decisions. According to Bédard et al. [3], RPA performs best when business rules are clearly defined, processes are stable, and exceptions are limited, enabling reliable automation without modifying systems.

Unlike isolated task automation, rule-based workflow automation integrates multiple activities into a single coordinated process. Software bots interact with applications through their user interfaces, allowing workflows to span multiple systems while preserving the existing system architecture. Siderska [13] notes that this approach improves process consistency, reduces manual effort, and minimizes human error in repetitive business processes.

Within the Business Process Management (BPM) domain, current RPA implementations primarily rely on predefined rules and manual configurations to automate predictable and repetitive processes. Agostinelli et al. [14] found that existing RPA tools are mainly effective for simple and predictable routine processes, while the composition of multiple routines into larger workflows still requires manual support. Building on these concepts, this study proposes a rule-based audit-to-

action workflow in which validated financial audit results directly trigger synchronization and notification activities through predefined business rules. The workflow coordinates multiple existing systems without modifying their underlying applications, providing a practical and reusable approach for educational administrative processes.

D. RPA Performance Evaluation

Evaluating process characteristics is essential to determine whether an operational activity is suitable for automation and can deliver measurable benefits. Wellmann et al. [15] link the viability of RPA to several process characteristics: the extent of task standardization, the frequency of task execution, the extent of data structure, system stability, and the amount of human judgment required. The safest bets are repetitive, rule-based tasks built on structured digital data, because they can run consistently with very low need for exception handling. Farinha et al. [15], [16] identify process standardization, process stability, number of transactions, and repetitiveness as relevant criteria for selecting processes for RPA automation.

TABLE I
RPA EVALUATION DIMENSIONS AND INDICATORS

Evaluation Dimension	Indicator	Evaluation Purpose
Efficiency	Processing Time	Measure execution speed improvement
Cost	Operational Cost Reduction	Measure reduction of manual operational effort
Quality	Validation Accuracy	Measure correctness and consistency of process execution
Reliability	Notification Success Rate	Measure execution reliability and delivery consistency
Scalability	Easy reconfiguration	Measure ability to handle addition of class and branch
Availability	Unattended scheduled execution	Measure ability to run when on schedule without attendance
Interoperability	Cross-System Integration	Measure capability to interact with multiple systems

This study evaluates the extent to which the implemented RPA solution supports audit and payment monitoring activities along these dimensions. Efficiency and cost are measured by processing time and reduction in operational cost, and reliability is measured by validation accuracy and notification success rate. Scalability, availability and interoperability are evaluated to analyze the system's ability to maintain continuous operations over multiple platforms, providing a basis for assessing the operational impact of the automation.

E. Total Cost of Ownership (TCO)

Evaluating RPA solely from operational savings does not always reflect the investment required throughout its lifecycle. Traditional cost-benefit analysis often focuses on recurring operational expenses while overlooking implementation, maintenance, human, and organizational costs, that significantly affect the overall cost of an automation project.

To provide a more comprehensive economic evaluation, Total Cost of Ownership (TCO) considers the costs incurred across the entire lifecycle of an automation solution. These include not only recurring operational expenses, but also implementation, technical, project, maintenance, and human-

related costs. By accounting for these components, TCO provides a more realistic basis for assessing the long-term financial impact of RPA adoption.

Following this perspective, the proposed workflow is evaluated using two complementary cost measures. Direct operational cost is used to quantify recurring execution savings after deployment, while TCO is used to assess the broader financial impact by incorporating implementation and lifecycle-related costs. This combined evaluation provides a more balanced assessment of the economic benefits of the proposed audit-to-action workflow.

III. RESEARCH METHOD

The study applies RPA to automate the financial audit and payment notification at XYZ School using a case study approach. The solution consists of three coordinated, independently scheduled subprocesses: (1) data extraction and consolidation, (2) financial validation and synchronization, and (3) automated notification and logging. The subprocess 1, data is extracted at the beginning of each semester, and the subprocess 2 and 3, validation and notification are run monthly before the payment due date.

The development is based on the RPA implementation framework proposed by El-Gharib and Amyot [17], which consists of four phases: assessment, program and test, implementation, and monitoring. The solution is developed in UiPath Studio. Subprocess 1 extracts student master data from the school's web portal into an Excel file, separated by branch, with each worksheet representing a class. Subprocess 2 validates financial records by comparing outstanding and past-due payments, recording the differences in an audit file to identify accounts requiring reminders. Subprocess 3 uses parent contact information from Google Sheets to send WhatsApp Desktop reminders.

WhatsApp Desktop was chosen over the official WhatsApp Business API to accommodate the institution's financial and technical constraints. The official API incurs recurring per-conversation costs and requires complex backend management that exceeds current administrative resources. Furthermore, processing fewer than 1,000 records once a month falls safely within the standard application's operational limits.

However, UI-based automation introduces inherent risks. System stability relies on the graphical layout, meaning structural UI updates can break RPA visual selectors and require manual script maintenance. The workflow is also vulnerable to loading latency, session disconnections, and account suspension if execution speeds are not regulated to mimic human interaction. Despite these risks, the task's low volume and frequency make this approach a highly pragmatic, accessible, and cost-effective solution.

The notification mechanism has a fallback in case the mother's contact number is not available and the father's number is called. If both numbers are invalid, the record is

updated as "Invalid Number". Functional testing was performed for 15 predefined scenarios including incomplete data, invalid contacts, inconsistent synchronization.

During implementation, the bot was deployed to the operational environment of XYZ School and configured to execute audits and send notifications on a schedule. The configuration included scheduling, synchronization, exception handling, and local folder management. The monitor-and-sustain phase addresses changes to portal structure, data inconsistencies, synchronization failures, and notification delivery failures to ensure long-term operational stability.



Fig. 1. Proposed RPA Development Workflow

Data on operational inefficiencies of the payment audit were gathered using semi-structured interviews and direct observation of administrative staff [14], [18]. The existing (As-Is) and proposed (To-Be) workflows were modeled using Business Process Model and Notation (BPMN) [19], [20] and baseline processing times for post-automation comparison were established through time-based observations [21].

A User Acceptance Testing (UAT) process was conducted on 100 randomly sampled records from the total 898-record dataset. The sampled records covered both Meruya and Teluk Gong branches and several classes. Each sampled record was compared with ground-truth data obtained through manual verification on the school portal. Two finance staff and the development team validated the RPA output. A record was considered correct only when the arrears, payment records, discrepancies, and notification status exactly matched the ground-truth data. The validation also included predefined edge cases, such as incomplete data, invalid contacts, and inconsistent synchronization. All 100 evaluated records matched the ground-truth data, resulting in 100% validation agreement within the evaluated sample.

The performance evaluation was based on the RPA evaluation criteria of Wellmann et al. [15] and used a mix of quantitative and qualitative methods. The criteria include efficiency, cost, quality, reliability, scalability, availability and interoperability. Efficiency was calculated based on processing time before and after automation. Cost was evaluated from two perspectives. Direct operational cost was calculated based on labor cost and execution duration, while Total Cost of Ownership (TCO) additionally incorporated implementation, consultancy, support, and maintenance costs. Reliability was calculated based on validation accuracy, delivery success rate and error rate. A 95% confidence interval for the contact delivery success rate was estimated using the Wilson score interval for a binomial proportion [22], [23], which provides more reliable interval coverage than the Wald interval, particularly for moderate sample sizes or proportions near the boundaries of 0 and 1. Scalability was indicated through the system's adaptability to changes or additions in branches and

classes. Availability was based on the system's ability to run automatically on scheduled time. Interoperability was based on its integration with school portal, Google Sheets, Excel and WhatsApp Desktop without altering existing systems.

The RPA architecture consists of four layers: (1) client side (UiPath Studio and automation components), (2) presentation layer (school portal and WhatsApp Desktop), (3) process execution layer (workflow orchestration, reconciliation, synchronization, and notification logic), and (4) data storage layer (master data, audit files, logs, and notification results), which enable scalable, cross-platform automation.

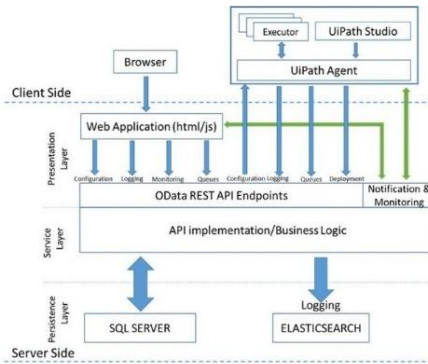


Fig. 2. Architecture Layer of RPA System [24]

This study received authorization from the school owner to conduct the research and access student and parent data, valid until completion of the implementation period. To protect Personally Identifiable Information (PII), development and testing used fictitious datasets. For future operational deployment, actual data will be used based on standard administrative procedures and informed consent through a Google Form with process disclosure and an opt-out mechanism. Operational processing will be confined to access-controlled, on-premise computers.

Architecturally, the RPA bot operates as a volatile pipeline, processing PII entirely in memory without maintaining a backend database. While the system outputs a local Excel execution log, this file is not a historical archive of sensitive data. It serves strictly as a temporary operational status report (e.g., logging "Delivered to mother" or "Failed"). School staff will retain these logs locally only for short-term auditing and manually delete them once the auditing purpose has been fulfilled, ensuring that no permanent, unmanaged historical records are retained post-execution [25].

IV. RESULTS

A. As-Is Process Analysis

The review of existing business process is needed to identify an inefficient pattern within the manual process, and to see if automation can be an option to solve it [17], [26]. The financial administration employees of XYZ School are required to process student payment data and audit records of

all branches and classes regularly. The existing (As-Is) workflow, described in Figure 3, involves staff manually accessing the system, filtering by branch, searching for records one by one, and checking each student's payment status.

This requires a second verification step. The outstanding payments amount is the total balance that is not paid, regardless of the due date. The unpaid list only shows payments that are already overdue. Staff have to manually compare the two to determine which balances are not yet overdue and which students need to be reminded, and then enter the results into Excel. It takes time, often more than a whole working day per audit cycle, and the repetitive manual verification raises the risk of human error across large volumes of data. Without an integrated follow-up mechanism, reminders have to be sent manually via WhatsApp Desktop, which delays reminders. These attributes repetitive, rule-based execution, high transaction volume, structured digital data, and the need for uniform validation [27] indicate that the current process is a good candidate for automation.

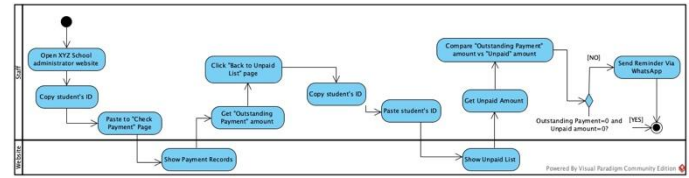


Fig. 3. As-Is Process of Student Payment Audit and Reminder Workflow

B. To-Be Process Design

To address this, an automated To-Be process was designed to reduce manual workload by integrating audit validation and notification into a single workflow. As shown in Figure 4, the workflow consists of three independently scheduled subprocesses: (1) data extraction and consolidation, (2) financial validation and synchronization, and (3) automated notification and recording. Subprocess 1 runs semi-annually at the beginning of each semester, whereas Subprocesses 2 and 3 run monthly before the payment due date.

C. Data Extraction and Consolidation

Subprocess 1 is the automatic extraction of the student master data from the school portal to Excel. The bot selects a branch and a class and then extracts the name, ID, class and branch details of the enrolled students and saves it in an Excel file specific to the branch with one worksheet for each class as shown in Figure 4. This architecture enables scalable processing, eliminates manual copy-paste work and forms the main input to the financial validation subprocess.

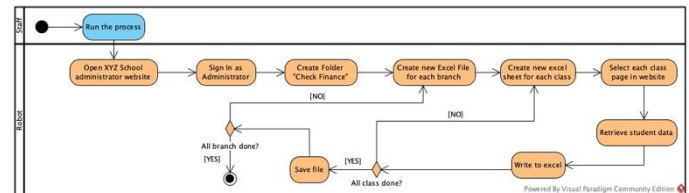


Fig. 4. To-Be Process of Student Master Data Extraction and Consolidation

D. Financial Validation and Synchronization

Subprocess 2 checks payment data. The bot looks up each student's outstanding and unpaid amounts on the portal using the master data files, as shown in Figure 5.

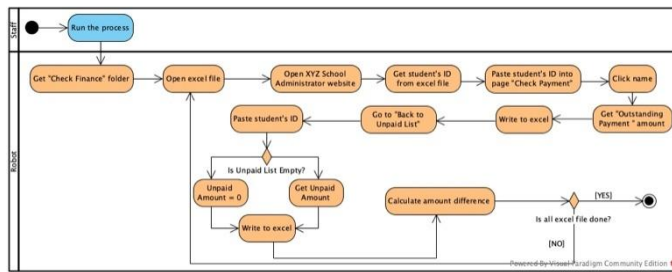


Fig. 5. To-Be Process of Financial Data Validation and Synchronization

The first amount is the total due to date, while the second is the amount past due; the bot calculates the difference to confirm each record before sending reminders. This reduces manual verification effort and improves consistency and accuracy for large datasets. The subprocess was run on 100 randomly selected records and compared with manually verified data, demonstrating consistent and accurate reconciliation under real conditions.

E. Automated Notification and Status Log

Subprocess 3 notifies and logs delivery status. The bot tries the mother's WhatsApp number first and logs "Delivered to Mother" if successful. If it is invalid or unregistered, it falls back to the father's number and logs "Delivered to Father", and logs "Invalid Number" if both fail, as shown in Figure 6 for students with outstanding payments. This shifts the audit workflow from passive reporting to a coordinated process where audit findings directly trigger reminder actions.

It is important to distinguish workflow accuracy from contact reachability. The bot executed the intended mother-then-father fallback sequence in all evaluated cases, while successful delivery depended on the availability and validity of parent contact information maintained in Google Sheets.

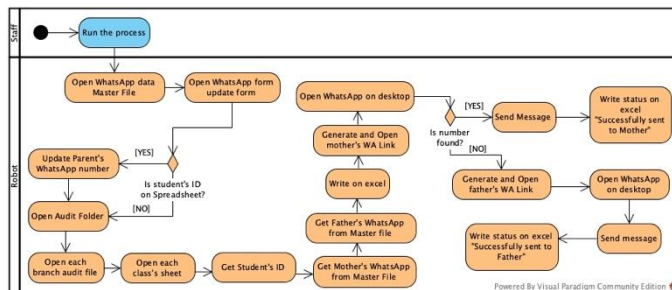


Fig. 6. To-Be Process of Automated Notification and Logging

V. SYSTEM IMPLEMENTATION

The system implementation makes the design an automation workflow for XYZ School, orchestrating data extraction, validation, synchronization, and notification without manual intervention. The workflow is divided into

subprocesses, allowing each to be scheduled. The system uses structured rule-based logic to manage audit and notification execution. The RPA workflow was developed and executed using UiPath Studio 26.0.201 on a machine running Windows 11. The hardware configuration utilized for deployment was an Intel Core i3 processor with 8GB of RAM, representing standard administrative computer specifications. Figure 7 illustrates how the workflow validates outstanding payment values (unpaid balance regardless of due date) against overdue unpaid records (unpaid balance past due) to identify records within the payment deadline that require reminders. Records with no outstanding payment are filtered out to improve efficiency.

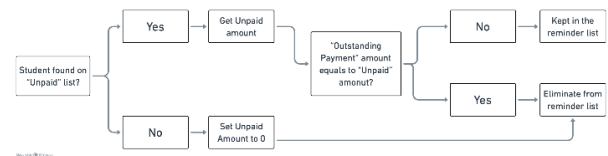


Fig. 7. Flowchart of Validation, Synchronization, and Paid Payment Data Elimination

When notification is required, the system attempts to find parent contact information based on the hierarchical fallback shown in Figure 8. First it checks the mother's contact number, then the father's number if the mother's number is unavailable or invalid. If neither contact is available, the system records logs "No Parent Contact Available", so staff can follow up manually.

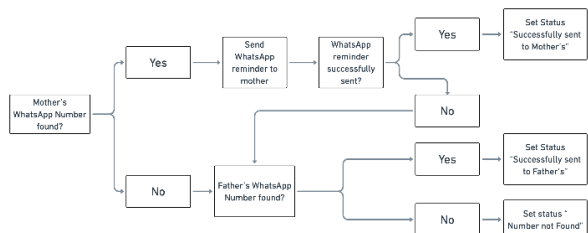


Fig. 8. Flowchart of Contact Fallback Mechanism and System Exception Handling

The implementation has three main processes based on the proposed system design that will be run in UiPath Studio. In the data extraction process (Process 1), UI element recognition is used to log into the school portal and extract student names, IDs, branch, and class data branches and classes, as shown in Figure 9. Results are arranged in Excel by branch and class.

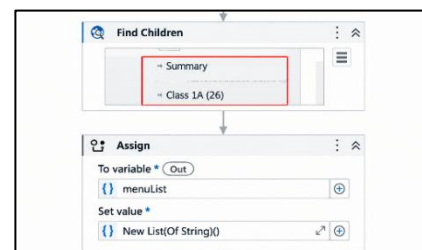


Fig. 9. Dynamic Navigation and Element Recognition

The workflow for the extraction is shown in Figure 10.

Figure 11 shows the folder structure for the data storage by timestamps for better organization and audit traceability.

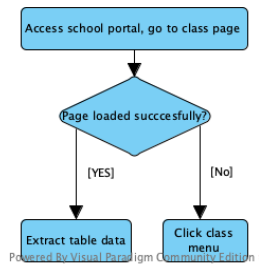


Fig. 10. Data Master Extraction

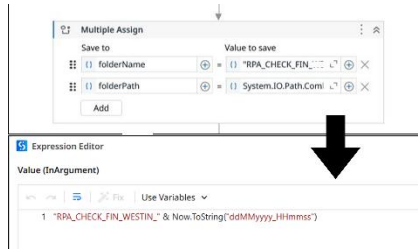


Fig. 11. Timestamp Based Folder Management

The financial validation (sub-process 2), performs data transformation and rule-based reconciliation. As shown in Figure 12, financial values are parsed and standardized before further processing.

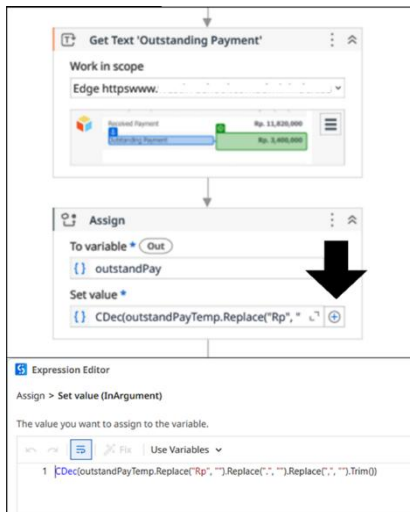


Fig. 12. Data Conversion Implementation (Parsing)

As illustrated in Figure 13 (a) Filtration Logic, the conditional logic compares outstanding balances and filters relevant records so that only records requiring action proceed to the next stage.

The notification phase (sub-process 3) merges data in Excel files with Google Sheets (Figure 13 (b) Contact Number Retrieval Flow) to retrieve and update the newest parent contact information into the audit dataset before sending out notifications.

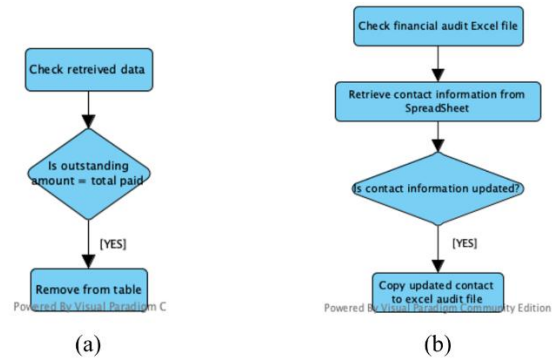


Fig. 13. Data Processing Flows: (a) Filtration Logic and (b) Contact Number Retrieval Flow

As shown in Figure 14, phone numbers are validated and standardized for generating link to WhatsApp Desktop before reminder delivery.

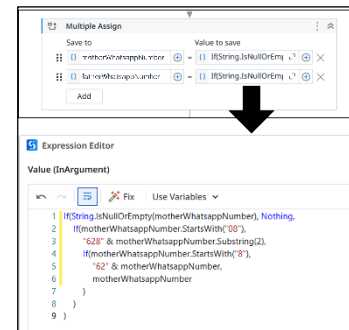


Fig. 14. Phone Number Format Validation Logic

The workflow is encapsulated in a Try-Catch error-handling framework, presented in Figure 15, to address exceptions including network failure, synchronization problems, missing interface elements, and unexpected responses, logging outcomes ("Delivered to Mother", "Delivered to Father", "Invalid Number", "No Parent Contact Available") for traceability.

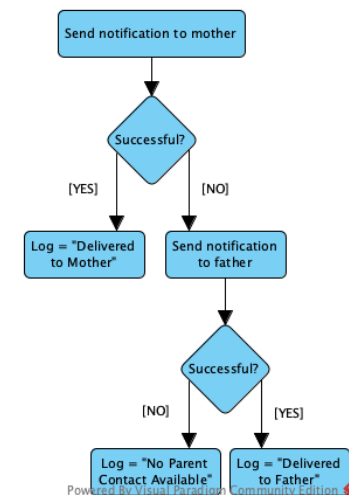


Fig. 15. Configuration of Error Handling Mechanism

Overall, the implementation demonstrates how the use of

structured rule-based logic, fallback mechanisms and exception handling enable RPA to orchestrate multi-step extraction, reconciliation, synchronization and notification workflows with improved consistency, scalability, interoperability and reliability in a real educational administrative environment.

VI. PERFORMANCE ANALYSIS

The performance analysis tests the proposed RPA solution for solving the problem faced by XYZ School in operational efficiency. To better assess the operational and financial impact of this automation project [28], this analysis takes a single automation project into account. The institution does not have formal KPIs on this process, however, it is expected that automation will improve efficiency, reduce processing time and to standardize the execution of monthly audit and reminder. The evaluation is carried out within the RPA assessment framework of Wellmann et al. [15] which considers efficiency, dependability, scalability, availability, and interoperability. The evaluation also includes a financial impact study based on Time-Driven Activity-Based Costing (TDABC) to compare time activities and resource consumption before and after automation.

The manual operational cost was estimated assuming a monthly salary of approximately USD \$468 for administrative personnel, reflecting the finance administrative staff at the case-study institution. Assuming 20 working days per month, this corresponds to a labor cost of USD 23.4 per workday.

Manual processing time was measured through one direct testing session for each activity using a stopwatch and screen recording. The measured duration was used to calculate the processing time per student. The per-student time was then extrapolated to the full workload of 898 student payment records. RPA execution time was obtained from system logs for the same workload. Thus, both approaches were compared using the same workload of 898 records. The manual baseline was obtained through direct measurement and extrapolation, while the RPA time was obtained from actual execution logs. Table V compares the two in minutes over 898 student payment records. The notification subprocess was tested separately during a 28-minute session with 100 records, a different sample than the UAT validation set.

TABLE II
REPEATED EXECUTION

Iteration	Sub-process 1	Status	Sub-process 2	Status	Sub-process 3	Status
1	05m 43s	Succeed	03h 35m 02s	Succeed	02h 44m 38s	Succeed
2	06m 07s	Succeed	03h 37m 30s	Succeed	02h 44m 11s	Succeed
3	05m 17s	Succeed	03h 39m 01s	Succeed	02h 43m 26s	Succeed
Average	05m 42s	Succeed (100%)	03h 37m 11s	Succeed (100%)	02h 44m 05s	Succeed (100%)

Table II details the performance of repeated executions across three iterations, demonstrating a reliable workflow with a consistent 100% success rate for all three sub-processes.

TABLE III

NOTIFICATION DELIVERY PERFORMANCE EVALUATION

Evaluation Period	Total Records Processed	Successful Deliveries	Failed Deliveries (Both Contacts)	Delivery Success Rate (%)	Avg. Processing Time per Record (s)
Quartile 1 (Min 1-7)	25	23	2	92.0%	16.8
Quartile 2 (Min 8-14)	25	24	1	96.0%	16.8
Quartile 3 (Min 15-21)	25	22	3	88.0%	16.8
Quartile 4 (Min 22-28)	25	24	1	96.0%	16.8
Total / Average	100	93	7	93.0% (95% CI: 86.3-96.6)	16.8

Notification Execution Accuracy, defined as the correct execution of the mother-then-father fallback logic, was 100% across the 100 evaluated UAT records. In contrast, the Contact Delivery Success Rate was 93.0%. To properly distinguish system execution accuracy from contact reachability, it is crucial to note that the remaining 7.0% failure rate was attributable entirely to external data-quality issues rather than automation defects. Specifically, the unreached cases consisted of 2 unregistered WhatsApp accounts and 5 instances of unavailable contact information. In all 7 instances, the RPA bot's exception-handling protocol functioned correctly by identifying the unreachability, skipping the delivery, and logging the failure without halting the overall workflow. To reduce this contact reachability shortfall in future operations, periodic data cleansing of the contact database is highly recommended.

Processing 100 records manually would require an estimated 300 minutes (3 min/record), compared with 28 minutes using RPA, saving over 4.5 hours per cycle. The average processing time was 16.8 seconds per record. This pacing reduces the risk of triggering WhatsApp anti-spam mechanisms and account suspension. Consequently, system scalability depends not only on third-party API rate limits and network latency but also on local computing resources.

TABLE IV
WILSON SCORE CONFIDENCE INTERVAL FOR RPA CONTACT DELIVERY SUCCESS RATE

Sample Size (n)	Successful Deliveries	Success Rate	95% Wilson CI
100	93	93.0%	[86.3%, 96.6%]

Total time to process the 898 records dropped from 1,897 minutes (manual) to 387 minutes (automated) for approximately 4.9-fold improvement in efficiency ($1,897 \div 387 = 4.90$). But the real benefit is not speed, but the elimination of human error and delayed notifications. The contact delivery success rate was evaluated using the Wilson score interval for a binomial proportion ($n = 100$), yielding a 95% confidence interval of [86.3%, 96.6%] (Table IV). The Wilson score interval was selected because it provides more reliable coverage than the normal-approximation (Wald) interval for moderate sample sizes and proportions close to 1 [24], [25].

TABLE V

BEFORE AND AFTER AUTOMATION

Activity	Sub-Activity	Before (min)	After (min)
Process 1 (Data Master)	Retrieve student data from web and compile into Excel file	168	6
Process 2 (Audit)	Check outstanding payment details and calculate balance differences	854	215
	Update audit report based on validation results	10	2
Process 3 (Notification)	Synchronize contact numbers from Google Sheets	20	4
	Send WhatsApp messages (mother/father fallback logic)	815	153
	Record delivery status into report	30	7
Total		1,897	387

Table VI summarizes the daily and annual costs before and after automation based on the schedule of each subprocess: Subprocess 1 is run once per semester, whereas Subprocesses 2 and 3 are run monthly.

The manual process of 898 records took 1,897 minutes per cycle, or 21,084 minutes per year $((168 \times 2) + [(854 + 10 + 20 + 815 + 30) \times 12]$; 336 minutes from the twice-yearly Subprocess 1 and 20,748 minutes for the monthly Subprocesses 2–3), which is approximately 43.93 workdays per year at a rate of 8 hours per day. The annual cost was \$1,027.85, or \$5,139.25 over five years.

Axmann et al. [29] identify implementation, maintenance, infrastructure, licensing, and other cost drivers that may arise across different RPA project scopes beyond direct operational costs. Since the proposed workflow uses an open-source RPA platform and existing institutional infrastructure, the recurring direct operational cost was estimated at 10% of the equivalent onshore employee cost (\$2.34 per workday). Based on an annual processing time of 4,584 minutes $((6 \times 2) + [(215 + 2 + 4 + 153 + 7) \times 12]$, or 9.55 workdays), the annual recurring direct operational cost was \$22.35 $(9.55 \times \$2.34)$, representing approximately a 46-fold reduction $(\$1,027.85 \div \$22.35)$ compared with the manual operational cost.

TABLE VI
DIRECT COST SPENT BEFORE AND AFTER AUTOMATION

Sub-Activity	Before Automation		After Automation	
	Duration Annually (days)	Cost Annually	Duration Annually (days)	Cost Annually
Retrieve student data and write to Excel Sheets	0.70	\$16.38	0.03	\$0.06
Get Student ID, navigate to payment page, and get amount	21.35	\$499.59	5.37	\$12.58
Calculate amount difference and update audit file	0.25	\$5.85	0.05	\$0.12
Sync contacts from GSuite and update Master file	0.50	\$11.70	0.10	\$0.23
Generate WA Link, send message (Mother/Father fallback)	20.38	\$476.78	3.82	\$8.95

Write delivery status to Excel report	0.75	\$17.55	0.18	\$0.41
Total	43.93	\$1,027.85	9.55	\$22.35
Total 5 years		\$5,139.25		\$111.75

In addition to recurring direct operational costs, the Total Cost of Ownership (TCO) was estimated using TDABC based on activity duration, required resources, and ongoing maintenance [18], this included one-time costs (implementation and consulting, in year one) and recurring costs (licensing, support, and maintenance).

The proposed workflow was implemented using an open-source RPA platform with no annual licensing cost, leaving support and maintenance as the primary recurring expenses.

The TCO analysis includes one-time implementation and consultancy costs, together with recurring support, maintenance, and direct operational costs. Since the proposed workflow was deployed using the institution's existing hardware, network, and software environment, no additional infrastructure cost was incurred. Human oversight was limited to periodic monitoring and was therefore included as part of the annual support and maintenance cost.

Table VII summarizes the Total Cost of Ownership (TCO) of the proposed workflow over a five-year period. The first-year total cost (including implementation and operational expenses) was \$443.77 with no licensing fee. The cost for subsequent years was \$92.59 per year for support and maintenance for a cumulative cost of \$814.13 over 5 years.

This represents approximately a 2.3-fold reduction in total cost during the first year $(\$1,027.85 \div \$443.77 = 2.32)$ and 6.3-fold reduction over five years $(\$5,139.25 \div \$814.13 = 6.31)$.

TABLE VII
TOTAL COST OF OWNERSHIP (TCO) AFTER AUTOMATION

Year	Cost component	Time Unit	Cost per automation
1	Implementation	One-Time	\$58.53
	Consultancy	One-Time	\$292.65
	Support & Maintenance	Annually	\$70.24
	Direct Cost	Annually	\$22.35
Total Year 1			\$443.77
2	Support & Maintenance	Annually	\$70.24
	Direct Cost	Annually	\$22.35
3	Support & Maintenance	Annually	\$70.24
	Direct Cost	Annually	\$22.35
4	Support & Maintenance	Annually	\$70.24
	Direct Cost	Annually	\$22.35
5	Support & Maintenance	Annually	\$70.24
	Direct Cost	Annually	\$22.35
Total 5 Years			\$814.13

Unlike the recurring direct operational cost reported in Table VI, the TCO analysis includes one-time implementation and consultancy costs together with recurring support and maintenance costs. Accordingly, the 46-fold reduction reflects recurring operational savings, whereas the 6.3-fold reduction represents the overall economic benefit when total ownership cost is considered. The operational performance of the proposed workflow, evaluated using quantitative and qualitative criteria, is summarized in Table VIII.

VII. DISCUSSION

The results show that RPA improves financial audit and payment notification by converting manual sequential activities into rule-based automation. The processing time was reduced from 1,897 to 387 minutes (≈ 4.9 -fold improvement), while the recurring direct operational cost was reduced by approximately 46-fold. These findings demonstrate that integrating rule-based financial reconciliation, dynamic parent-contact synchronization and hierarchical WhatsApp notification provides an effective automation approach for repetitive financial administration without modifying existing information systems.

The system was successful in the User Acceptance Testing (UAT) with high validation accuracy and 93.0% success rate for field notification delivery. The other 7.0% of failures were based on data quality issues (e.g., invalid or unregistered contact numbers) but not the automation workflow itself, and batch testing showed stable execution with no workflow terminations. This hierarchical fallback mechanism contributed to operational reliability, by automatically redirecting failed primary notifications to secondary contacts. These findings also highlight the importance of accurate operational data for maximizing automation performance.

The implemented automation demonstrated stable performance during the workload, processing 100 notification records in 28 minutes (16.8 seconds per record). The workflow supports scheduled unattended execution and interoperates with the School Portal, Google Sheets, and WhatsApp Desktop without requiring modifications to existing systems.

A. Comparative Analysis

These findings build on prior RPA literature. Moffitt et al. [5] identified accuracy improvement and workload reduction as the main benefits of RPA for audit functions, but their work is conceptual and does not provide performance benchmarks. Our study extends their framework by quantifying these benefits in a live educational setting. Our findings demonstrate that a validated audit outcome can independently trigger downstream operational action, transforming a static audit report into an instrument for coordinated follow-up.

Enriquez et al. [9] systematically mapped 53 RPA studies and found that most implementations focus on isolated task-level automation, while cross-functional workflows integrating validation, synchronization, and communication remain under-represented. This study addresses this gap through three subprocesses that integrate data extraction, financial reconciliation, and automated notification into an end-to-end workflow across a web portal, Excel, Google Sheets, and WhatsApp Desktop without API-level changes..

In the field of education, Wijaya and Gunawan [7] used RPA in the context of internship reporting administration in a similar setting in Indonesia, this research tackles a more difficult and repetitive financial reconciliation process, with a comparison of students' outstanding payment and overdue

records for 898 students across branches and classes. This added complexity is reflected in performance gains including 4.9-fold improvement in processing efficiency, an approximately 46-fold reduction in recurring direct operational cost, and a hierarchical fallback notification mechanism that achieved a 93.0% delivery success rate.

Finally, Ribeiro et al. [30] focusing on industrial manufacturing environments, suggest that integrating RPA with Artificial Intelligence (AI) can further enhance automation capabilities. In contrast, the proposed workflow demonstrates that structured administrative processes can be effectively automated using a rule-based workflow architecture, achieving 100% validation accuracy in UAT and a 93.0% notification delivery success rate in the evaluated implementation. These findings indicate that rule-based RPA provides a practical and cost-efficient solution for structured administrative processes in educational institutions.

B. Research Limitations

These benefits are not without limitations. This automation relies on the integrity of external platforms and is therefore sensitive to interface modifications that may break the workflow execution. Furthermore, due to its rule-based nature, RPA is limited in its ability to handle highly complex or ambiguous cases and still requires human intervention in certain situations. So, continuous monitoring, workflow adjustment from time to time and data quality management are still needed to ensure long-term operational reliability.

Overall, the results summarized in Table VIII indicate that the proposed audit-to-action workflow improves operational efficiency, reduces processing costs, and demonstrates stable operational performance during the evaluated implementation. These findings suggest that the workflow provides a practical and reusable automation approach for educational institutions managing repetitive administrative processes.

TABLE VIII
OPERATIONAL PERFORMANCE EVALUATION OF RPA IMPLEMENTATION

Dimension	Metric/ Indicator	Measurement Result	Interpretation
Efficiency	Processing time reduction	1,897 to 387 minutes (898 records)	~ 4.9 -fold faster, indicating significant efficiency improvement
Cost	Operational cost reduction	\$1,027.85 to \$22.35 per year	~ 46 -fold lower cost, indicating strong financial impact
Quality	Validation accuracy	100% agreement within the evaluated UAT sample	Ensures consistent and accurate financial audit results
Reliability	Notification success rate	93.0% (93/100 records)	High delivery effectiveness with minimal failure
	Error rate	7.0% (7/100 records)	Failures mainly caused by data quality issues, not system logic
	Execution stability	Stable workflow completion during testing	No workflow termination failures observed
Scalability	Recovery mechanism	Hierarchical fallback execution	Supports automated notification recovery process
	Easy reconfiguration	Will adapt to addition of class and branch	Easy to reconfigure when adaptation is needed

Availability	Unattended scheduled execution	Can be run automatically on schedule	Reduce dependency on human intervention
Interoperability	System integration	School Portal + Google Sheets + WhatsApp	Enables seamless cross-platform automation

VIII. CONCLUSION

This study demonstrated how RPA can address manual financial audits and delayed notifications by integrating financial reconciliation, dynamic parent-contact synchronization, and hierarchical WhatsApp notification into a rule-based audit-to-action workflow without modifying existing information systems. The implementation achieved processing speeds of up to 4.9-fold faster, high reliability, and significant cost savings, with 100% accuracy during UAT and a 93.0% notification delivery success rate in live operations. These findings indicate that RPA can improve operational efficiency, while reducing administrative costs, and support scalable, and coordinated administrative processes.

In practice, this approach reduces administrative workload and allows staff to focus on higher-value activities. From a theoretical perspective, it introduces a rule-based workflow model that combines financial reconciliation, dynamic parent-contact synchronization, and hierarchical notification through predefined business rules. It also provides a non-invasive, interoperable, and reusable reference model for other educational institutions facing similar challenges.

However, several limitations should be considered. The workflow depends on external platforms and input data quality, both of which may affect performance. Moreover, because it is rule-based, the workflow cannot effectively handle complex or ambiguous cases and still requires periodic monitoring and human intervention.

Future work may focus on extending the proposed workflow to support more complex business rules and exception handling beyond the capabilities of the current rule-based automation. Further studies may also evaluate the applicability of the proposed workflow across different educational institutions to strengthen its generalizability.

ACKNOWLEDGMENT

The authors would like to thank the XYZ school for providing access to data and supporting the implementation of this study.

REFERENCES

- [1] V. Bhardwaj and M. Kumar, "Transforming higher education with robotic process automation: enhancing efficiency, innovation, and student-centered learning," Dec. 01, 2025, *Springer Nature*. doi: 10.1007/s43621-025-01198-6.
- [2] S. A. Khairullah, S. Harris, H. J. Hadi, R. A. Sandhu, N. Ahmad, and M. A. Alshara, "Implementing artificial intelligence in academic and administrative processes through responsible strategic leadership in the higher education institutions," *Front. Educ. (Lausanne)*, vol. 10, 2025, doi: 10.3389/FEDUC.2025.1548104.
- [3] M. Bédard, A. Leshob, I. Benzarti, H. Mili, R. Rab, and O. Hussain, "A rule-based method to effectively adopt robotic process automation," *Journal of Software: Evolution and Process*, vol. 36, no. 11, Nov. 2024, doi: 10.1002/SMR.2709.
- [4] D. Andrade, "Challenges of Automated Software Testing with Robotic Process Automation RPA-A Comparative Analysis of UiPath and Automation Anywhere," *International Journal of Intelligent Computing Research*, vol. 11, no. 1, Dec. 2020, doi: 10.20533/ijicr.2042.4655.2020.0129.
- [5] K. C. Moffitt, A. M. Rozario, and M. A. Vasarhelyi, "Robotic process automation for auditing," *Journal of Emerging Technologies in Accounting*, vol. 15, no. 1, pp. 1–10, Mar. 2018, doi: 10.2308/JETA-10589.
- [6] A. Gunawan, M. I. Wijaya, N. Andika, and A. Lius, "Empirical Study of Robotic Process Automation: Implementation and Evaluation," *Hum. Behav. Emerg. Technol.*, vol. 2025, no. 1, 2025, doi: 10.1155/hbe2/2876164.
- [7] M. I. Wijaya and A. Gunawan, "Robotic Process Automation to Improve Education's Administration Process: Case of Students' Internship Reporting," in *Proceedings of 2023 International Conference on Information Management and Technology, ICIMTech 2023*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 155–160. doi: 10.1109/ICIMTech59029.2023.10277719.
- [8] M. König, L. Bein, A. Nikaj, and M. Weske, "Integrating Robotic Process Automation into Business Process Management," *Lecture Notes in Business Information Processing*, vol. 393 LNBP, pp. 132–146, 2020, doi: 10.1007/978-3-030-58779-6_9.
- [9] J. G. Enriquez, A. Jimenez-Ramirez, F. J. Dominguez-Mayo, and J. A. Garcia-Garcia, "Robotic Process Automation: A Scientific and Industrial Systematic Mapping Study," *IEEE Access*, vol. 8, pp. 39113–39129, 2020, doi: 10.1109/ACCESS.2020.2974934.
- [10] P. C. Verhoef *et al.*, "Digital transformation: A multidisciplinary reflection and research agenda," *J. Bus. Res.*, vol. 122, pp. 889–901, Jan. 2021, doi: 10.1016/J.JBUSRES.2019.09.022.
- [11] M. Delagrammatikas, S. Stelios, and P. Tzavaras, "The Role of RPA and Data Analysis in the Transformation of the Insurance and Banking Industries," *Encyclopedia 2025*, vol. 5, no. 4, p. 155, Sep. 2025, doi: 10.3390/ENCYCLOPEDIA5040155.
- [12] R. Syed *et al.*, "Robotic Process Automation: Contemporary themes and challenges," *Comput. Ind.*, vol. 115, Feb. 2020, doi: 10.1016/J.COMPIND.2019.103162.
- [13] J. Siderska, "Robotic Process Automation-a driver of digital transformation?," *Engineering Management in Production and Services*, vol. 12, no. 2, pp. 21–31, Jun. 2020, doi: 10.2478/EMJ-2020-0009.
- [14] S. Agostinelli, A. Marrella, and M. Mecella, "Towards Intelligent Robotic Process Automation for BPMers," *AAAI-20 Workshop on Intelligent Process Automation (IPA'20)*, Jan. 2020, doi: 10.48550/arXiv.2001.00804.
- [15] C. Wellmann, M. Stierle, S. Dunzer, and M. Matzner, "A framework to evaluate the viability of robotic process automation for business process activities," Sep. 2020, doi: 10.1007/978-3-030-58779-6_14.
- [16] D. Farinha, R. Pereira, and R. Almeida, "A framework to support Robotic process automation," *Journal of Information Technology*, vol. 39, no. 1, 2024, doi: 10.1177/02683962231165066.
- [17] N. M. El-Gharib and D. Amyot, "Robotic process automation using process mining — A systematic literature review," *Data Knowl. Eng.*, vol. 148, p. 102229, Nov. 2023, doi: 10.1016/J.DATAK.2023.102229.
- [18] W. M. P. van der Aalst, M. Bichler, and A. Heinzl, "Robotic Process Automation," *Business and Information Systems Engineering*, vol. 60, no. 4, pp. 269–272, Aug. 2018, doi: 10.1007/S12599-018-0542-4.
- [19] D. M. D. U. Putra and W. Welda, "Business Process Analysis and Modeling Using the Business Process Improvement Framework at the Internal Quality Assurance STMIK STIKOM Indonesia," *ACSIE (International Journal of Application Computer Science and Informatic Engineering)*, vol. 1, no. 2, pp. 75–86, Nov. 2019, doi: 10.33173/ACSIE.53.
- [20] W. M. P. van der Aalst, "Business Process Management: A Comprehensive Survey," *Int. Sch. Res. Notices*, vol. 2013, no. 1, p. 507984, Jan. 2013, doi: 10.1155/2013/507984.
- [21] C. Flechsig, J. Lohmer, and R. Lasch, "Realizing the Full Potential of Robotic Process Automation Through a Combination with BPM," *Lecture Notes in Logistics*, pp. 104–119, 2019, doi: 10.1007/978-3-030-29821-0_8.
- [22] E. B. Wilson, "Probable Inference, the Law of Succession, and Statistical Inference," *J. Am. Stat. Assoc.*, vol. 22, no. 158, pp. 209–212, 1927, doi: 10.1080/01621459.1927.10502953.
- [23] P. G. Andersson, "A Note on Confidence Intervals for a Binomial p: Andersson–Nerman vs. Wilson," *Stat*, vol. 13, no. 4, Dec. 2024, doi: 10.1002/sta4.70027.

- [24] UiPath Documentation, "About Deployment," <https://docs.uipath.com/orchestrator/standalone/2025.10/installation-guide/about-deployment#about-deployment>.
- [25] UiPath Documentation, "Protecting Sensitive Information," <https://docs.uipath.com/studio/standalone/latest/user-guide/protecting-sensitive-information#protecting-sensitive-information>.
- [26] D. Schlegel, B. Rosenberg, O. Fundanovic, and P. Kraus, "How to conduct successful business process automation projects? An analysis of key factors in the context of robotic process automation," *Business Process Management Journal*, vol. 30, no. 8, pp. 99–119, Dec. 2024, doi: 10.1108/BPMJ-06-2023-0465.
- [27] R. V. E. Quille *et al.*, "Performance Analysis Method for Robotic Process Automation," *Sustainability (Switzerland)*, vol. 15, no. 4, Feb. 2023, doi: 10.3390/SU15043702.
- [28] M. Lacity and L. Willcocks, "Becoming strategic with intelligent automation," *MIS Quarterly Executive*, vol. 20, no. 2, Jun. 2021, doi: 10.17705/2msqe.00047.
- [29] B. Axmann, H. Harmoko, L.-V. Herm, and C. Janiesch, "A Framework of Cost Drivers for Robotic Process Automation Projects," *Springer Nature Switzerland AG*, pp. 7–22, 2021, doi: 10.1007/978-3-030-85867-4_2.
- [30] J. Ribeiro, R. Lima, T. Eckhardt, and S. Paiva, "Robotic Process Automation and Artificial Intelligence in Industry 4.0 – A Literature review," *Procedia Comput. Sci.*, vol. 181, pp. 51–58, Jan. 2021, doi: 10.1016/J.PROCS.2021.01.104.

Jovita Junilla born in Jakarta, Indonesia 06 of June 2005. Currently undergoing an undergraduate program of information systems in BINUS University Greater Jakarta, Indonesia. She is currently an IT Project Manager with Mandiri Utama Finance, Jakarta, Indonesia.

Jessica Selistio born in Jakarta, Indonesia 29 of December 2005. Currently undergoing an undergraduate program of information systems in BINUS University Greater Jakarta, Indonesia.

She is currently doing an internship program, studying to be an IOS Developer in Tangerang, Indonesia.

Qabil Putra Sandi born in Jakarta, Indonesia 23 of November 2004. Currently undergoing an undergraduate program of information systems in BINUS University Greater Jakarta, Indonesia. He is currently doing an internship program at PT Inovasi Sukses Sentosa (Indotrading), working as a UI/UX Designer in West Jakarta.

Ali Gunawan received the Bachelor's degree in management information systems from Universitas Bina Nusantara, Jakarta, Indonesia, in 1998, and the Master's degree in marketing management from Universitas Esa Unggul, Jakarta, Indonesia, in 2010. He has been a Lecturer with Bina Nusantara University, West Jakarta, Indonesia, since 2000, and is currently the Director of PT Hyperdata Solusindo Mandiri, Kota Tangerang, Banten, Indonesia. His research and professional interests include robotic process automation (RPA), business analysis, and database management systems. Mr. Gunawan holds several professional certifications, including CSCA, CBDMP, PSM I, CDAP, UiRPA, and CDVP.