

TRANSMISSION OF DIGITIZED INVOICES TO ACCOUNTING SOFTWARE

Sylvia MAŤAŠOVÁ, Martin CHOVANEC, Dušan ČATLOCH

Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics,
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, Tel. +421 55/602 3175,
E-mails: sylvia.matasova@tuke.sk, martin.chovanec@tuke.sk, dusan.catloch@tuke.sk

ABSTRACT

The primary objective of this work was to automate the transfer of digitized invoices into an accounting software system. Traditionally, manual invoice processing is time-consuming, prone to errors, and increases operational costs, hindering timely financial reporting. The work analyzes various solutions for invoice digitization, their export methods, and the technologies utilized for system development and data transmission. The key outcome is the successful implementation of a system for transferring digitized invoices into accounting software. The system leverages a modular design with a microservices-oriented architectural approach, with C# chosen as the primary programming language. To ensure reliable asynchronous communication, AWS Simple Queue Service (SQS) was utilized, and AWS DynamoDB was used for persistent data storage. The system was designed to eliminate manual intervention in the invoice processing workflow, thereby enhancing efficiency, accuracy, and overall operational speed. This work significantly contributes to centralizing invoices in a digital format with a unified design, streamlining accounting processes, and laying the groundwork for advanced financial analytics or machine learning applications in accounting.

Keywords: invoice, accounting software, automation, digitization, data transfer

1. INTRODUCTION

In today's digital age, efficient and streamlined business operations are paramount. One of the key areas ripe for automation is the processing and management of financial documents, particularly invoices. Traditionally, invoice handling has been a manual, time-consuming process prone to human error, involving physical documents, manual data entry, and storage challenges. This approach not only increases operational costs but also hinders timely financial reporting and analysis. Moreover, the environmental impact of paper-based processes and the time-consuming nature of physical archiving are increasingly recognized. Automated solutions offer a pathway to reduced paper consumption, faster financial closures, and enhanced decision-making capabilities for businesses globally.

The objective of this work was to address these challenges by developing an automated system for transferring digitized invoices into an accounting software. The primary goal was to eliminate manual intervention in the invoice processing workflow, thereby enhancing efficiency, accuracy, and overall operational speed. This research aims to explore existing solutions for invoice digitization, analyze various export methods and data formats, and leverage modern technologies to create a robust and reliable data transfer system. The ultimate goal is to provide a seamless integration between digitized invoice sources and accounting software, ensuring that all invoices are centralized in a digital format with a unified design, significantly contributing to the modernization of accounting processes.

2. RELATED WORK

Invoice digitization and automation currently leverage advanced technologies such as Optical Character Recognition (OCR), deep learning, and computer vision. These systems efficiently extract data from various invoice formats. Modern OCR tools, often enhanced with

Convolutional Neural Networks (CNNs), achieve high accuracy even with complex layouts and tables. Object detection algorithms (e.g., YOLO, SSD, R-CNN) identify key fields. Template-free approaches, in turn, help process diverse invoice designs.

Intelligent Document Processing (IDP) for invoices is driven by AI, OCR, deep learning, and Robotic Process Automation (RPA). Charabuddi's research from 2025 [1] and Anchoori's study from 2024 [2] indicate that modern systems now combine AI-powered OCR, RPA, and ERP integration to automate data extraction, validation, and workflow management. This significantly reduces manual intervention, error rates, and improves processing speed and compliance. Deep learning, as stated by Gunaydin and colleagues in 2023 [3], with ResNet-50-based models, achieved text recognition accuracy of up to 97%, enabling rapid invoice digitization. AI solutions process diverse formats, adapt to unstructured data, and utilize Natural Language Processing (NLP) for more precise extraction and anomaly detection. Predictive automation using machine learning, as demonstrated by Charabuddi in 2025 [4], further optimizes invoice approvals by classifying risk and automating low-risk transactions. Desai and colleagues in 2021 [5] found that these technologies showed substantial efficiency gains; RPA systems process invoices twice as fast as humans. Microservice architectures and intelligent validation frameworks enhance scalability and integration with financial systems. Ultimately, these innovations transform financial operations, streamlining invoice processing, reducing costs, and supporting better decision-making.

In the field of automated invoice data extraction, emerging technologies are evolving rapidly. Anchoori in 2024 [2] noted that modern systems increasingly combine OCR with advanced machine and deep learning models (e.g., transformers like LayoutLM) to extract data from various formats, including those from irregular vendors. The integration of Large Language Models

(LLMs) with OCR improves contextual understanding and decision-making, increasing accuracy and reducing manual intervention. Arslan in 2022 [6] emphasized that computer vision techniques (e.g., YOLO, R-CNN) are used to identify key fields and tables. Ramyadevi and colleagues in 2024 [7] mentioned that RPA platforms like UI Path streamline repetitive tasks, minimize errors, and integrate data into systems. These AI-driven architectures offer real-time processing, scalability, and improved compliance. However, challenges remain, including processing a wide variety of invoice formats and ensuring continuous adaptability. The convergence of OCR, NLP, deep learning, and RPA is transforming invoice processing, leading to significant increases in speed, accuracy, and operational efficiency.

Traditional OCR systems for invoice processing convert scanned images into text. However, they often struggle with various formats, handwritten elements, and complex layouts, leading to errors and manual interventions. In contrast, Yarlagadda in 2025 [8] stated that AI-driven solutions (integrating AI-powered OCR, machine learning, RPA, and LLMs) offer significant improvements in accuracy, adaptability, and efficiency. These advanced systems learn from diverse formats, extract data more reliably, and automate validation and workflows. This reduces manual labor and errors while increasing speed and scalability. Arslan in 2022 [9] indicated that AI-enhanced OCR can also process unstructured data, adapt to new document types, and continuously improve. Kerutis and Calneryte in their 2022 work [10] showed that integrating AI with OCR and RPA enables comprehensive automation, including the elimination of unnecessary documents and compliance management. While traditional OCR is limited by templates, AI-driven solutions excel in dynamic environments and are the preferred choice for optimizing invoice processing.

Invoice digitization has undergone significant development with digital technologies. Various methods are used to convert physical invoices into electronic formats. Scanning transforms them into digital images (PDF, JPEG). Subsequently, OCR software extracts text information and converts unstructured data into machine-readable formats. Advanced OCR solutions vary: some are template-based for consistent layouts, while others utilize sophisticated AI-driven OCR that adapts to different structures and extracts data from unstructured text, increasing accuracy and reducing manual review, as pointed out by Holt and Chisholm in their 2018 research [11].

In addition to scanning and OCR, there are also standardized electronic document exchange methods. Klapita in 2021 [12] stated that Electronic Data Interchange (EDI) offers a robust approach for exchanging business documents between partners, although it requires significant setup and is more common for high-volume transactions. Many modern accounting software and invoice management platforms often leverage Application Programming Interfaces (APIs) for programmatic access to their functionalities. Hu and colleagues in 2021 [13] found that APIs can vary in design, requiring attention to data models, authentication, and rate limits. Rossum

documentation [14] clarifies that webhooks automatically notify the accounting system of a newly digitized invoice. For data export, common formats include: XML (Extensible Markup Language) for complex enterprise integrations, JSON (JavaScript Object Notation) for web services and mobile applications, and CSV (Comma Separated Values) for simple tabular data.

Microservices architecture is increasingly used in financial document automation to enhance scalability, modularity, and resilience. Singh in 2024 [15] stated that by dividing complex systems into independent services, institutions can automate document processing, improve risk management, and process data in real-time. The integration of AI-driven monitoring in microservices prevents failures and ensures high availability. Kurakula in 2025 [16] indicated that the transition from monolithic systems to microservices allows for more flexible updates and the integration of technologies like generative AI, which automates customer service and streamlines workflows. Modernization brings challenges such as system decomposition and maintaining data consistency, addressable by strategies like the Strangler Pattern and event-driven architectures. AI tools support this transition by automating code analysis and data synchronization. Microservices combined with AI provide a robust foundation for financial document automation, improving operational efficiency and maintaining regulatory compliance.

Automated data transfer systems are developed using programming languages, frameworks, and cloud services. Rossum [17] suggests that Python is popular for its versatility and libraries for data processing, web applications, and automation. C# and Java are also utilized in enterprise integrations. Web frameworks like FastAPI (Python) or ASP.NET Core (C#) enable robust and efficient APIs for data processing and inter-service communication. These frameworks support microservices, which enhance modularity, scalability, and fault isolation, ideal for distributed systems. Gunasekaran and colleagues in 2009 [18] pointed out that message queuing systems (e.g., RabbitMQ, Apache Kafka) ensure reliable asynchronous data processing and scalability under high loads. Finally, as Amazon Web Services [19] states in its documentation, cloud platforms (Azure, AWS, Google Cloud) provide services such as virtual machines, serverless functions, and databases, crucial for deploying and managing automated systems with the required scalability, reliability, and security. Understanding these technologies is essential for selecting tools for an automated invoice transfer system.

Cloud solutions are transforming automated invoice management. Gold Beulah Patturose and colleagues in 2024 [20] stated that these systems leverage cloud computing to provide accessible and cost-effective platforms that automate invoice generation, processing, and storage. They are particularly attractive to SMEs, who can thus avoid the costs of on-premise software. Zhang in 2018 [21] described advanced features, including QR code recognition for rapid and accurate data entry, integration with cloud-based enterprise information, and real-time access to financial data, significantly reducing

errors and processing time. Kumar and colleagues in 2023 [22] indicated that artificial intelligence and machine learning further improve automation through intelligent data extraction, fraud detection, and predictive analytics, streamlining validation, approval, and integration with ERP systems. These innovations enhance operational efficiency and financial accuracy, supporting paperless and scalable invoice management, thereby enabling businesses to focus on growth.

Machine Learning (ML) is transforming accounting automation by streamlining repetitive tasks, improving accuracy, and enabling sophisticated data analysis. Bavaresco and colleagues in 2023 [23] and Zhang in 2021 [24] stated that key applications include automated data entry, financial forecasting, predictive analytics, fraud detection, and intelligent transaction recognition. All of this reduces manual labor and operational costs and improves decision-making. Zhang in 2021 [24] also found that ML-driven systems link financial and business data, supporting management from financial analysis to performance evaluation. In practice, ML is often integrated with RPA and chatbots, simplifying accounting processes and reducing compliance time. However, successful implementation requires high-quality data, continuous training, and attention to user experience. Smith and Lamprecht in 2024 [25] noted that limitations such as data availability and training time need to be addressed for reliable financial information. Bertomeu in 2020 [26] argued that while ML increases efficiency, human expertise for complex decision-making remains essential. Overall, ML offers significant opportunities for accounting automation, but its full potential depends on overcoming technical, ethical, and organizational challenges.

Overall, the convergence of advanced technologies such as OCR, AI, deep learning, NLP, and RPA, combined with cloud solutions and microservice architectures, is transforming invoice processing and overall accounting automation. These innovations deliver significant gains in speed, accuracy, and operational efficiency, while reducing costs, minimizing manual intervention, and supporting better compliance and decision-making. Although implementation presents challenges, particularly in data quality and system adaptation, the potential for optimizing financial operations is immense.

3. METHODOLOGY

The proposed automated invoice transfer system was designed to create a robust and efficient solution for migrating digitized invoice data into accounting software. The system leverages a combination of modern technologies and architectural patterns to ensure reliability, scalability, and maintainability of operations.

3.1. Overall system architecture

The system's architecture is founded on a modular design, facilitating independent component development

and deployment. The workflow commences with an external Source System (Webhook Producer), responsible for digitizing invoices. This system, specifically integrating with the Rossum platform for intelligent document processing and OCR, dispatches a JSON-formatted webhook request upon invoice completion to our custom solution.

This request is received by a C# Webhook Listener which performs initial validation before enqueueing the message into an AWS Simple Queue Service (SQS) message queue for asynchronous processing. SQS acts as a decoupling intermediary, ensuring message persistence and system resilience. The core logic resides in the C# Invoice Processing Service, which retrieves messages from SQS, parses and transforms invoice data to the accounting software's specific format, and executes the data transfer via the Accounting Software API/Integration. Processed data and logs are persistently stored in AWS DynamoDB.

3.2. Technological architecture

The system utilizes a selected technology stack primarily within the Amazon Web Services (AWS) ecosystem. C# serves as the primary programming language for its robustness and enterprise compatibility, implementing core logic for both the webhook listener and processing service. AWS SQS provides scalable asynchronous messaging [19], while AWS DynamoDB, a NoSQL database, ensures high-performance persistent storage for invoice data [27]. Communication with the accounting software API is managed via an Apollo service or similar integration layer. The entire solution is deployed and managed on the AWS cloud platform, leveraging its comprehensive services for scalability, security, and reliability [28]. Docker can be employed for local development consistency.

3.3. Data flow and processing logic

The data flow follows a structured sequence: a digitized invoice is processed by the external source system, triggering a JSON webhook request to the C# webhook listener. After initial payload validation, the raw invoice data is published to a dedicated AWS SQS queue. Concurrently, the Invoice Processing Service continuously consumes messages from this queue. Upon retrieval, the service parses the JSON data, performs crucial data transformations to align with the target accounting software's API requirements (including field mapping, data type conversion, and application of business logic), and finally executes an API call to create the new invoice record. Successful transfers are logged, and error handling mechanisms, potentially including retry logic or notifications, are triggered for failures. This systematic approach ensures reliable and automated invoice data transfer, significantly reducing manual effort and minimizing errors.

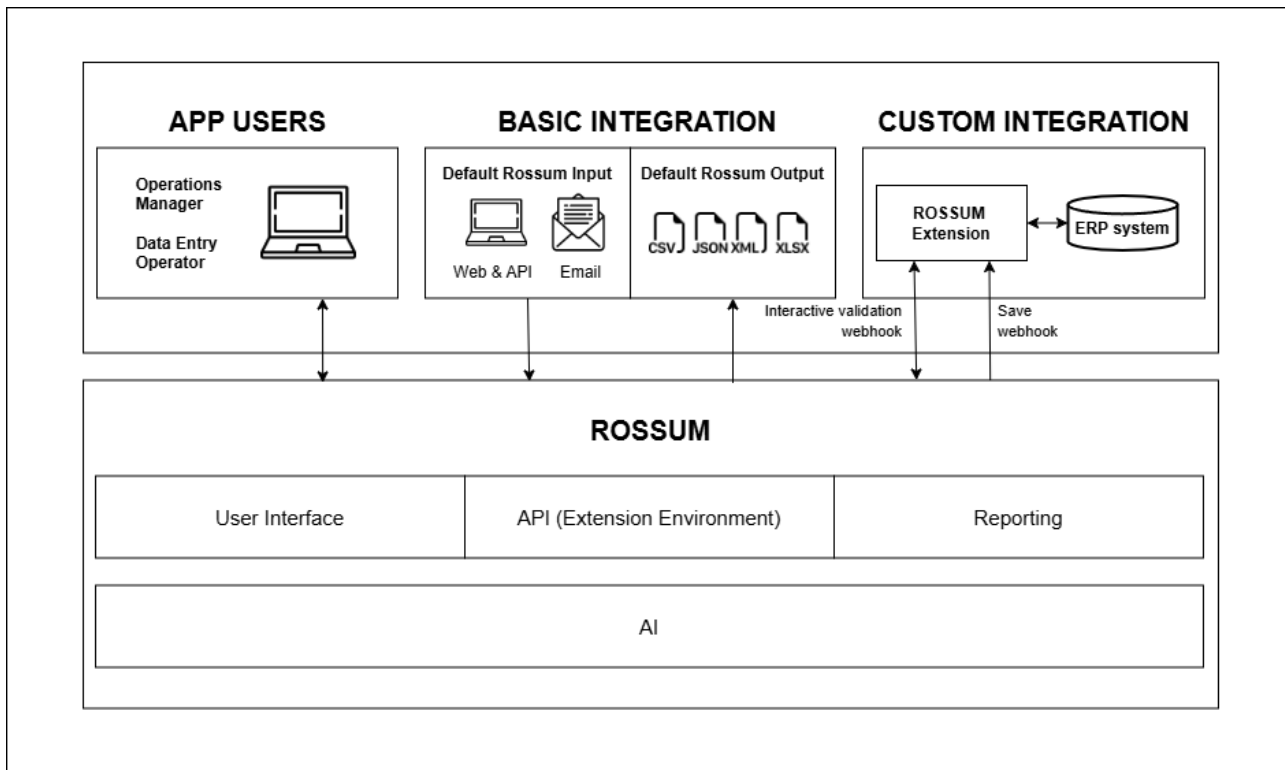


Fig. 1 Rossum Integration Architecture showing initial data flow to the custom solution

3.4. Design of key algorithms and methods

To ensure the robustness and efficiency of the system, several key algorithms and methodological approaches are integrated into each processing phase.

Webhook Validation and Asynchronous Enqueueing

The Webhook Listener implements a rigorous JSON payload structure validation upon receiving incoming invoice data. This validation is critical for ensuring data integrity and preventing the processing of corrupted or incomplete messages. A recommended approach for this is JSON Schema Validation, which allows for a declarative definition of the expected data structure. Following successful validation, the message is then atomically stored in the AWS SQS queue. This crucial step serves to decouple the system's components and concurrently provides resilience against temporary outages of downstream services. Messages remain persistently in the queue until their successful processing, ensuring no data loss during transient system interruptions.

Invoice Processing and Data Transformation The Invoice Processing Service executes a sophisticated transformation of data fields, converting them from the original Rossum format to the precise specifications required by the target accounting software. This comprehensive process encompasses both field mapping (direct translation of field names and their corresponding values) and data type conversion (ensuring correct data types, such as converting strings to numbers or dates). Beyond mere structural changes, business logic specific to invoice processing, such as VAT calculations or checks

for invoice uniqueness, is applied. These rules may necessitate the implementation of more complex algorithms or leverage rule-engine-based approaches to facilitate flexible configuration and adaptation. Communication with the accounting software's API is managed via HTTP calls. To bolster reliability, a retry mechanism with exponential backoff is integrated. This strategy significantly minimizes the risk of permanent transfer failures that might arise from temporary network issues or an overloaded target API. Furthermore, incorporating the Circuit Breaker pattern would substantially enhance the system's resilience to prolonged unavailability of external services, effectively preventing unproductive attempts to communicate with a non-functional API and thereby conserving valuable system resources.

Persistence and Audit All critical states and data pertaining to invoice processing are persistently stored in AWS DynamoDB. This chosen NoSQL database offers rapid data access and inherent scalability, which are indispensable qualities for maintaining comprehensive audit trails and effectively monitoring the processing status. The system's operational transparency is further enhanced through the use of structured logging for recording all operations and events. When combined with performance metrics and a Correlation ID (a unique identifier used to trace each invoice throughout its journey across the entire system), this comprehensive approach enables efficient monitoring, streamlined problem diagnosis, and thorough auditing of the entire invoice processing lifecycle.

4. EXPERIMENTAL ARCHITECTURE AND SYSTEM VERIFICATION

To verify the functionality of the proposed system, an experimental architecture was created that simulated real-world invoice processing conditions. The testing environment consisted of various digitized invoice formats, which were processed through the Rossum platform and transferred into the SmartPablo accounting software. The verification was conducted in a controlled setting, monitoring each step of the workflow—from document digitization and data validation to export into the target system.

Participants performed a defined sequence of tasks that simulated typical operational procedures. Each step was time-tracked and evaluated for reliability and consistency. The results demonstrated that the system could process invoices without manual intervention, with all test cases completed successfully. During the verification, key webhook events were identified that ensure proper initiation of the data export process. These findings confirm that the proposed architecture is suitable for practical deployment in automated accounting environments.

5. RESULTS

The implemented system successfully demonstrated the automated transfer of digitized invoice data into the target accounting software, thereby achieving the primary objective of automating the invoice processing workflow and leading to significant improvements in efficiency and accuracy.

5.1. Functional performance and system usability

Comprehensive testing, utilizing diverse digitized invoice samples, affirmed the system's robust functional performance. The C# webhook listener consistently demonstrated reliable initial data ingestion by accurately receiving webhook payloads from the external digitization service. Subsequently, messages were dependably enqueued in AWS SQS and robustly consumed by the processing service, underscoring the effectiveness and resilience of the asynchronous communication mechanism. Crucially, the invoice processing service consistently performed JSON data parsing, executed all requisite data transformations, and accurately facilitated the transfer of essential invoice metadata and line-item details (e.g., invoice identifier, issuance date, due date, vendor information, itemized services/goods, and total financial amounts) to the accounting software via its designated Application Programming Interface (API). Furthermore, fundamental error handling mechanisms were effectively integrated to log failed transfer attempts, providing critical insights into potential systemic anomalies such as API communication failures or data validation discrepancies. This automation significantly mitigated the manual effort previously associated with data entry, effectively eliminating human intervention from the point of invoice digitization to its final entry into the accounting system. This not only accelerates the overall process throughput

but also substantially minimizes the inherent risk of human transcription errors typically associated with manual data input.

The successful integration critically relies on specific webhook events triggered by the external digitization platform. These events ensure the timely and accurate initiation of the invoice transfer process. A detailed description of key webhook events and their corresponding actions is presented in Table 1.

Table 1 Webhook Event Descriptions

Event and Action	Description
annotation_status.changed	<i>status</i> - Annotation status changed
annotation_content.initialize	Annotation initialized (data extracted)
annotation_content.user_update	User updated annotation
annotation_content.export	Annotation is moving to exported state
email.received	Email with attachments received

Beyond functional verification, a series of usability assessments were conducted to evaluate the system's practical applicability and performance under controlled conditions. These assessments involved multiple participants executing predefined tasks, with their completion times meticulously recorded. The results indicated universal task completion by all participants, validating both the system's operational effectiveness and the clarity of the defined procedural steps. The temporal data for individual tasks, measured in seconds, revealed minimal inter-user variance, confirming the efficiency and temporal consistency of the automated processes. This uniformity in task completion times underscores the robust design and inherent reliability of the system. Upon conclusion of the assessments, participants collaboratively evaluated their requirements, previously identified using the MoSCoW method, affirming their successful fulfillment. This collective positive feedback further substantiates the successful outcome of the system's validation phase.

Specifically, participants were instructed to execute a defined sequence of tasks representative of the invoice processing workflow within the system:

Task 1: User Authentication -Perform the authentication procedure to gain access to the Rossum system using provided credentials.

Task 2: Document Digitization -Execute the process of scanning or photographing the provided physical invoice and storing the digital representation on the local workstation.

Task 3: Digital Asset Ingestion - Utilize the "Import" functionality to upload the digitized invoice file into the Rossum interface, observing its subsequent placement within the processing queue.

Task 4: Digitized Data Verification and Validation - Access the digitized invoice, review the automatically

extracted data fields, perform necessary corrections or additions, and confirm the data accuracy before returning to the processing queue.

Task 5: Data Export Initiation - Navigate to the "Confirmed" tab, select the validated invoice, and initiate its export as a JSON object via the system's control panel.

Task 6: Accounting System Authentication - Perform the authentication procedure to gain access to the SmartPablo accounting software using provided credentials.

Task 7: Exported Record Retrieval - Access the "Sales" module within the accounting software's navigation menu and locate the recently exported invoice, which is expected to be displayed as the primary record.

Task 8: Digital Artifact Retrieval and Validation - Within the detailed view of the exported invoice, initiate

the download of the digital invoice document (e.g., PDF) and subsequently verify its contents against the original physical invoice for data correspondence.

The time expended by each of the three participants to complete these tasks is graphically represented in Figure 1, illustrating the consistency of performance across the user cohort.

The successful implementation of this system validates the chosen architectural approach and technology stack, demonstrating a practical and effective solution for modernizing invoice management processes. This outcome supports the potential for broad applicability in similar document processing automation scenarios and contributes to the knowledge base of distributed system design for enterprise data integration.

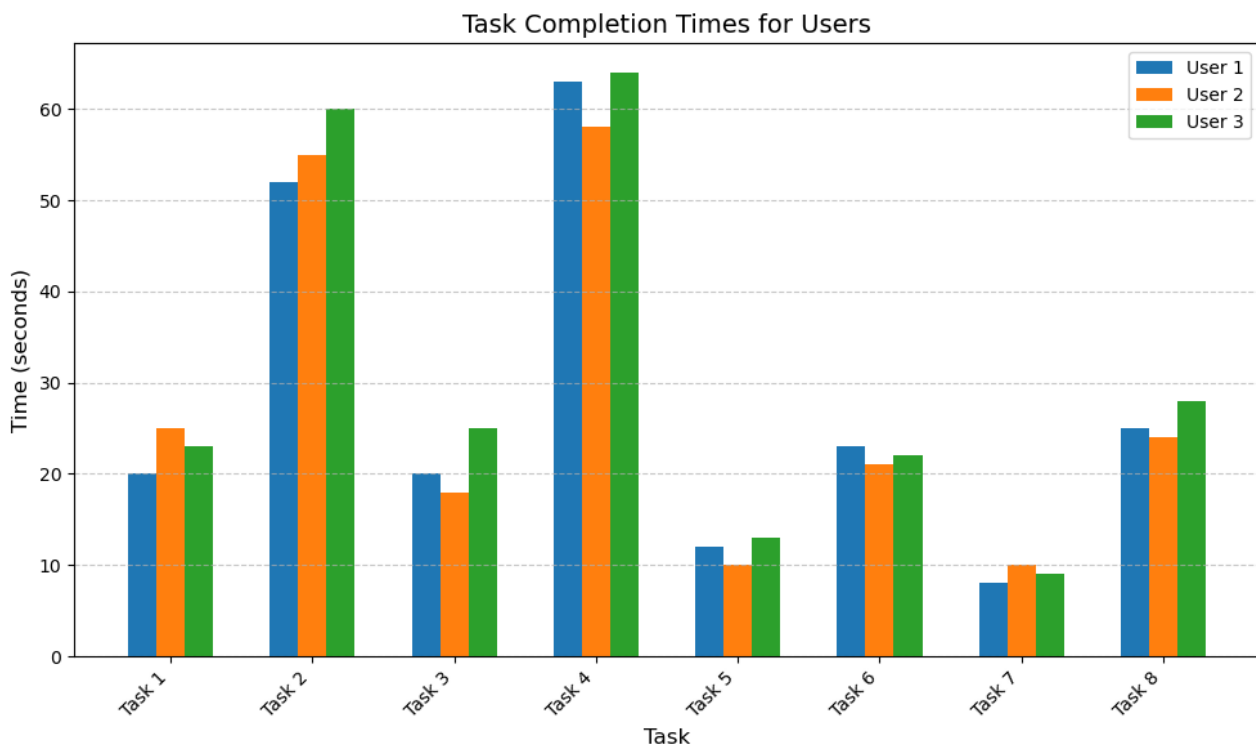


Fig. 2 Task Completion Times (in seconds) for Users

6. DISCUSSION

The automated invoice transfer system developed in this work represents a significant advancement towards streamlining financial document management. By leveraging a microservices-oriented architecture with asynchronous messaging, a design increasingly favored for scalability and resilience as highlighted by Singh in 2024 [15], the solution effectively addresses common challenges associated with traditional manual invoice processing. These challenges, often including high labor costs, prevalent data entry errors, and delays in financial reporting, are mitigated, aligning with the efficiency gains discussed by Desai and colleagues in

2021 [5]. The strategic choice of a C# application for the webhook listener and AWS SQS as a robust message broker, as described in AWS documentation [19], proved particularly effective in ensuring reliable and resilient data ingestion and processing, even under varying loads. C#'s inherent robustness further facilitated rapid development and seamless integration with various APIs, consistent with its utility in enterprise integrations mentioned by Rossum [17].

While the system successfully achieved its primary objective of automating invoice transfer, its impact extends beyond mere operational efficiency. The centralization of invoices in a unified digital format significantly enhances

data accessibility and facilitates future analytical processes, potentially laying the groundwork for advanced financial analytics or sophisticated machine learning applications in accounting, as explored by Bavaresco and colleagues in 2023 [23] and Zhang in 2021 [24]. The system's modular design inherently ensures future scalability and ease of maintenance, allowing for straightforward integration of new invoice sources or diverse accounting software systems as evolving business needs dictate. This aligns with modern trends in financial document automation which emphasize modularity, scalability, and resilience through microservices, as noted by Singh [15] and Kurakula in 2025 [16].

A critical aspect for any such system, especially given the handling of sensitive financial information, is security. Ensuring secure data transmission, robust authentication mechanisms, and comprehensive protection against unauthorized access are paramount for API integrations and webhooks, a general concern for systems utilizing APIs as discussed by Hu and colleagues in 2021 [13]. Furthermore, strict adherence to data privacy regulations, such as GDPR, is integral throughout the entire data processing pipeline.

The implemented system's successful functionality and consistent performance, as demonstrated by the usability assessments, validate the chosen architectural approach and technology stack. The minimal inter-user variance in task completion times indicates a high degree of process automation and reliability, reducing human-induced variability. This is particularly relevant in comparison to traditional OCR systems and manual processes, which, as Yarlagadda in 2025 [8] and Holt and Chisholm in 2018 [11] point out, are inherently more prone to errors and slower execution due to struggles with diverse formats and reliance on manual interventions.

6.1. Limitations and future research

A potential limitation of the current system architecture is its dependency on the external invoice digitization service to provide data in a consistent format via webhooks. Any significant alterations in the external service's output format would necessitate corresponding adjustments in the invoice processing service, potentially requiring maintenance overhead. This highlights a common challenge in integrating with external platforms, even with the flexibility offered by webhooks as detailed by Rossum documentation [14].

Future research and development efforts could explore several avenues to further enhance the system's capabilities and resilience:

Advanced Data Validation and Error Recovery: Incorporating more sophisticated data validation techniques beyond schema checks, possibly utilizing machine learning (ML) for anomaly detection in invoice data, could enhance data quality at an earlier stage, building on the ML applications in accounting discussed by Bavaresco and colleagues [23]. Furthermore, developing more advanced automated error recovery mechanisms, potentially involving AI-driven re-processing or intelligent

routing to human review for complex errors, would improve overall system robustness and reduce the need for constant human oversight, addressing some of the limitations identified by Smith and Lamprecht in 2024 [25].

Broader API Integration and Versatility: Expanding the system's integration capabilities to support a wider range of accounting software APIs would significantly increase its versatility and market applicability. This would involve developing a more abstract and configurable integration layer to minimize the effort required for new API adaptations, drawing inspiration from the varying API designs observed by Hu and colleagues [13].

Enhanced Document Understanding with AI/ML: Moving beyond basic OCR, future work could delve into the application of advanced Artificial Intelligence (AI) and Machine Learning (ML) techniques for more profound document understanding. This includes leveraging Large Language Models (LLMs) and transformer networks (e.g., LayoutLM) for improved contextual understanding and semantic extraction from highly unstructured invoice formats, as highlighted by Anchoori in 2024 [2]. Such advancements could enable functionalities like automated account coding, intelligent categorization of expenses, or even fraud detection based on invoice patterns and historical data, aligning with the transformative impact of AI/ML on accounting automation, including predictive analytics and automated approvals, as discussed by Kumar and colleagues in 2023 [22] and Charabuddi in 2025 [4].

Proactive Monitoring and Predictive Analytics: Integrating AI-driven monitoring within the microservices architecture could proactively prevent failures and ensure higher availability, a benefit of microservices architectures combined with AI as stated by Singh in 2024 [15]. Furthermore, leveraging predictive analytics based on historical processing data could optimize resource allocation and anticipate potential bottlenecks, further enhancing operational efficiency and contributing to the broader application of ML in financial forecasting and management information systems, as explored by Zhang in 2021 [24].

Overall, the system delivers a practical and impactful solution for modernizing invoice processing, demonstrating the tangible benefits of automation in financial operations. The outlined future research directions offer significant potential for evolving this system into a more intelligent, adaptable, and comprehensive financial automation platform.

7. CONCLUSION

This paper successfully designed and implemented an automated system for transferring digitized invoices into an accounting software system. The primary goal of eliminating manual intervention in the invoice processing workflow was achieved through a robust architecture utilizing C# applications, AWS SQS, and AWS DynamoDB, deployed on Amazon Web Services (AWS). The system effectively digitizes, processes, and transfers invoice data, leading to enhanced efficiency, accuracy, and centralization of financial records. This

work significantly contributes to streamlining accounting processes and provides a scalable and reliable foundation for future digital transformation in financial document management.

REFERENCES

- [1] CHARABUDDI, Ranadheer Reddy: *Transforming Financial Operations: Integrating SAP OpenText VIM, AI-Powered OCR, and RPA for Advanced Invoice Processing and Junk Document Elimination*, World Journal of Advanced Engineering Technology and Sciences **Vol. 15, No. 02, pp. 828–836, 2025**, No. DOI: 10.30574/wjaets.2025.15.2.0620 (Access date: 20.05.2025) <https://doi.org/10.30574/wjaets.2025.15.2.0620>
- [2] ANCHOORI, S.: *AI-Driven Document Processing: A Novel Framework for Automated Invoice Data Extraction from PDF Documents*, International Journal for Multidisciplinary Research **Vol. 6, No. 6, 2024**, No. DOI: 10.36948/ijfmr.2024.v06i06.32247 (Access date: 20.05.2025) <https://doi.org/10.36948/ijfmr.2024.v06i06.32247>
- [3] GUNAYDIN, E. – GENCTURK, B. – ERGEN, C. – KOKLU, M.: *Digitization and Archiving of Company Invoices using Deep Learning and Text Recognition-Processing Techniques*, Intelligent Methods in Engineering Sciences **2023**, No. DOI: 10.58190/imiens.2023.69 (Access date: 20.05.2025) <https://doi.org/10.58190/imiens.2023.69>
- [4] CHARABUDDI, R.: *Enhancing Financial Approvals with AI-Powered Predictive Automation: Optimizing Invoice Management and Vendor Risk Assessment*, European Journal of Computer Science and Information Technology **Vol. 13, No. 1, 2025**, No. DOI: 10.37745/ejcsit.2013/vol13n172738 (Access date: 20.05.2025) <https://doi.org/10.37745/ejcsit.2013/vol13n172738>
- [5] DESAI, D. – JAIN, A. – NAIK, D. – PANCHAL, N. – SAWANT, D.: *Invoice Processing using RPA & AI*, SSRN Electronic Journal **2021**, No. DOI: 10.2139/ssrn.3852575 (Access date: 20.05.2025) <https://doi.org/10.2139/ssrn.3852575>
- [6] ARSLAN, H.: *End to End Invoice Processing Application Based on Key Fields Extraction*, IEEE Access **Vol. 10, pp. 78398–78413, 2022**, No. DOI: 10.1109/access.2022.3192828 (Access date: 20.05.2025) <https://doi.org/10.1109/access.2022.3192828>
- [7] RAMYADEVI, R. – ANBUSELVAN, A. – JULIAN, A. – SELVI, S. – MURALI DHAR, M. S.: *Invoice Processing Automation Using UI Path Studio*, Proc. 2024 IEEE Int. Conf. on Computing, Power and Communication Technologies (IC2PCT), Greater Noida, India **pp. 486–489, 2024**, No. DOI: 10.1109/IC2PCT60090.2024.10486692 (Access date: 20.05.2025) <https://doi.org/10.1109/IC2PCT60090.2024.10486692>
- [8] YARLAGADDA, K.: *Transforming Financial Operations: The Impact of AI-Driven Invoice Processing Across Industries*, International Journal of Scientific Research in Computer Science, Engineering and Information Technology **2025**, No. DOI: 10.32628/cseit25112794 (Access date: 20.05.2025) <https://doi.org/10.32628/cseit25112794>
- [9] ARSLAN, H.: *End to End Invoice Processing Application Based on Key Fields Extraction*, IEEE Access **Vol. 10, pp. 78398–78413, 2022**, No. DOI: 10.1109/ACCESS.2022.3192828 (Access date: 20.05.2025) <https://doi.org/10.1109/ACCESS.2022.3192828>
- [10] KERUTIS, V. – CALNERYTE, D.: *Intelligent Invoice Documents Processing Employing RPA Technologies*, In: Lopata, A.; Gudonienė, D.; Butkienė, R. (eds) Information and Software Technologies. ICIST 2022. Communications in Computer and Information Science, vol. 1665 **Springer, Cham, 2022**, No. DOI: 10.1007/978-3-031-16302-9_18 (Access date: 20.05.2025) https://doi.org/10.1007/978-3-031-16302-9_18
- [11] HOLT, X. – CHISHOLM, A.: *Extracting structured data from invoices*, Proceedings of the Australasian Language Technology Association Workshop 2018, No. (2018) 53–59 <https://aclanthology.org/U18-1006>
- [12] KLAPITA, V.: *Implementation of Electronic Data Interchange as a Method of Communication Between Customers and Transport Company*, Transportation Research Procedia **53**, No. (2021) 174–179 <https://www.sciencedirect.com/science/article/pii/S2352146521001836>
- [13] HU, T. – ZHANG, Z. – YI, P. – LIANG, D. – LI, Z. – REN, Q. – HU, Y. – LAN, J.: *SEAPP: A secure application management framework based on REST API access control in SDN-enabled cloud environment*, Journal of Parallel and Distributed Computing **147**, No. (2021) 108–123 <https://www.sciencedirect.com/science/article/pii/S0743731520303671>
- [14] ROSSUM: *Webhook vs Connector*, Official Documentation, No. (Access date: 20.05.2025) <https://api.elis.rossum.ai/docs/#webhook-vs-connector>
- [15] SINGH, M.: *Resilient Microservices Architecture with Embedded AI Observability for Financial Systems*, Journal of Electrical Systems **2024**, No. DOI: 10.52783/jes.8596 (Access date: 20.05.2025) <https://doi.org/10.52783/jes.8596>
- [16] KURAKULA, S.: *Architectural Challenges in Modernizing Legacy Financial Systems with Microservices and AI*, World Journal of Advanced Engineering Technology and Sciences **2025**, No. DOI: 10.30574/wjaets.2025.15.2.0651 (Access date: 20.05.2025) <https://doi.org/10.30574/wjaets.2025.15.2.0651>

- [17] ROSSUM: *Getting Started with Rossum*, Official Documentation, No. (Access date: 20.05.2025) <https://developers.rossum.ai/docs/getting-started>
- [18] GUNASEKARAN, A. – MCGAUGHEY, R. E. – NGAI, E. W. T. – RAI, B. K.: *E-Procurement adoption in the Southcoast SMEs*, International Journal of Production Economics **122**, No. 1 (2009) 161–175
- [19] Amazon Web Services (AWS): *Amazon SQS Documentation*, No. (Access date: 20.05.2025) <https://aws.amazon.com/sqs/resources/>
- [20] GOLD BEULAH PATTUROSE, J. – PRISCILLA, R. – ARAVIND SHANKAR, S. – HARIRAM, S. M.: *Remote and Centralized Billing System Using Cloud Computing*, In: Owoc, M. L.; Varghese Sicily, F. E.; Rajaram, K.; Balasundaram, P. (eds) Computational Intelligence in Data Science. ICCIDS 2024. IFIP Advances in Information and Communication Technology, vol. 718 **Springer, Cham, 2024**, No. DOI: 10.1007/978-3-031-69986-3_18 (Access date: 20.05.2025) https://doi.org/10.1007/978-3-031-69986-3_18
- [21] ZHANG, W.: *Online Invoicing System Based on QR Code Recognition and Cloud Storage*, Proc. 2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC) **pp. 2576–2579, 2018**, No. DOI: 10.1109/IMCEC.2018.8469461 (Access date: 20.05.2025) <https://doi.org/10.1109/IMCEC.2018.8469461>
- [22] KUMAR, N. – KATOCH, S. – TRIPATHI, S. – SINGH, B. P. – SAJAL, S. – YADAV, A. L.: *AI Enabled Invoice Management Application*, Proc. 2023 8th Int. Conf. on Communication and Electronics Systems (ICES), Coimbatore, India **pp. 816–820, 2023**, No. DOI: 10.1109/ICES57224.2023.10192802 (Access date: 20.05.2025) <https://doi.org/10.1109/ICES57224.2023.10192802>
- [23] BAVARESCO, R. S. – NESI, L. C. – BARBOSA, J. L. V. – ANTUNES, R. S. – RIGHI, R. – COSTA, C. A. – VANZIN, M. – DORNELLES, D. – JUNIOR, S. C. – GATTI, C. – FERREIRA, M. – SILVA, E. – MOREIRA, C.: *Machine Learning-Based Automation of Accounting Services: An Exploratory Case Study*, International Journal of Accounting Information Systems **Vol. 49, 2023, Article 100618**, No. DOI: 10.1016/j.accinf.2023.100618 (Access date: 30.06.2025) <https://doi.org/10.1016/j.accinf.2023.100618>
- [24] ZHANG, X.: *Application of Data Mining and Machine Learning in Management Accounting Information System*, Journal of Applied Science and Engineering **Vol. 24, No. 5, pp. 813–820, 2021**, No. DOI: 10.6180/jase.202110_24(5).0018 (Access date: 30.06.2025) [https://doi.org/10.6180/jase.202110_24\(5\).0018](https://doi.org/10.6180/jase.202110_24(5).0018)
- [25] SMITH, L. – LAMPRECHT, C.: *Identifying the Limitations Associated with Machine Learning Techniques in Performing Accounting Tasks*, Journal of Financial Reporting and Accounting **2024**, No. DOI: 10.1108/jfra-05-2023-0280 (Access date: 30.06.2025) <https://doi.org/10.1108/jfra-05-2023-0280>
- [26] BERTOMEU, J.: *Machine Learning Improves Accounting: Discussion, Implementation and Research Opportunities*, Review of Accounting Studies **Vol. 25, No. 3, pp. 1135–1155, 2020**, No. DOI: 10.1007/s11142-020-09554-9 (Access date: 30.06.2025) <https://doi.org/10.1007/s11142-020-09554-9>
- [27] Amazon Web Services (AWS): *Amazon DynamoDB Documentation*, No. (Access date: 20.05.2025) <https://aws.amazon.com/dynamodb/resources/>
- [28] Amazon Web Services (AWS): *AWS Cloud Platform Overview*, No. (Access date: 20.05.2025) <https://aws.amazon.com/what-is-aws/>

Received August 19, 2025, accepted September 8, 2025

BIOGRAPHIES

Sylvia Mat’ášová was born in Slovakia. She completed her Bachelor’s and Master’s (Ing.) studies in Informatics at the Technical University of Košice, Faculty of Electrical Engineering and Informatics, from September 2019 to May 2024. Currently, she is pursuing her PhD studies in Informatics at the same university, with her research focusing on "Research and Development of Artificial Intelligence for Document Indexing and Structuring." Her academic background includes a strong foundation in computer science and experience in various programming languages and modern development technologies.

Martin Chovanec received his engineering degree in informatics in 2005 from Faculty of Electrical Engineering and Informatics, Technical University of Kosice. In 2008 he received his Ph.D. degree at the Department of Computers and Informatics of the Faculty of Electrical Engineering and Informatics of the Technical University of Kosice and his scientific research was focused on network security and encryption algorithms. Currently, he is Director of the Institute of Computer Technology of the Technical University of Kosice.

Dušan Čatloch graduated with a Masteros degree in 2024 at the Department of Computers and Informatics, Technical University of Košice, Slovakia. Currently, he is pursuing his Ph.D. at the same department. The main topic of his research is machine learning.