International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(20s) 2026 pp. 1618-1626
https://doi.org/10.70917/ijcisim-2026-6022
Article

Modernizing Legacy Government Information Systems: A Framework for Systems Integration, Process Automation, and Regulatory Compliance

Iswarya Srinivasan

Independent Researcher, USA
ORCID ID: 0009-0001-4698-6579

Abstract: Government agencies at all levels continue to operate legacy information systems that were designed prior to interoperability, automation, and audit requirements of modern public administration. Modernization efforts often treat systems integration, business process automation, and regulatory compliance as separate workstreams with their own discipline-specific body of practice and respective implementation timelines to address the shortcomings of legacy information systems. Isolating these three concerns leads to some friction, as integrations are built on an unstable layer, and compliance controls added later to the integrated automation will likely not cover every use case during an audit. To address these issues, this paper proposes a three-layer architecture to handle these concerns as coupled design tracks, instead of the legacy modernization approach of treating each concern as a sequential phase: Integration Layer, Automation Layer, and Compliance Layer. The integration layer allows older and new systems to work together according to the approach of incremental decomposition and centralized mediation. The automation layer builds upon this layer and brings together robotic process automation and business process management orchestration, emitting structured event logs at each step in the workflow. These event logs feed the Compliance Layer's control-to-component traceability matrix, creating documented technical mechanisms in lieu of just redoing the reconciliations for an audit. The implementation methodology is by phases, such that only one bounded business function is worked on at a time. The completed phase is handed over to operations staff. The state-level licensing and permitting function case study shows the joint benefits of interface consolidation, workflow automation, and audit-traceability enhancement and also demonstrates bona fides that the model does not require regulatory changes or an all-or-nothing replacement of legacy systems. This approach makes it appropriate for agencies that have rigid procurement cycles or continuity requirements. The paper concludes with implications for modernization practice and the need to test measured outcomes of deployments that operating agencies have undertaken.
Keywords: Legacy System Modernization, Systems Integration, Business Process Automation, Regulatory Compliance, Public Sector Information Systems

1. Introduction

Government administrative agencies , at the federal, state, and local levels continue to depend on information technology systems dating back to mainframe computing and large application stacks, before service-oriented computing. The systems are still adequate for narrowly defined transaction processing, but their architecture lacks the integration, automation, and compliance features needed for modern public administration. The US Government Accountability Office estimates that 80 percent of information technology resources go to operations and maintenance of existing systems (including legacy systems). They are left with limited funds for modernization or new purchases. This is a persistent issue because legacy systems consume maintenance resources to an extent that over-involves operations staff , because their monolithic architecture was never designed for the interoperability, workflow automation, and regulatory traceability that modern government requires.


Copyright: © 2026 by the authors

The problems multiply due to (1) systems integration failures, where legacy applications cannot share data with modern applications. This forces agencies to continue re-entering data manually into multiple applications or to create point-to-point interfaces between applications, which grow with every new application introduced into the environment [1], [2]. Second, business processes that are still undertaken on paper or manually increase processing time and the risk of processing error. Robotic process automation and business process management tooling have begun addressing these issues in public sector contexts with variable success [7], [8]. Third, ensuring compliance with regulatory obligations, for example in relation to data protection or financial auditing, is particularly challenging in a setting where obligations are enforced across many different disconnected systems and manual reconciliation steps are necessary [12], [13].

Existing research treats these three problems separately, with studies of legacy modernization focusing on architectural decompositions and migration strategies [1], [3], [5]. Research on process automation in the public sector, including robotic process automation and business process management adoption, does not consider the integration architecture on which systems are based [7][9][10][11]. Research on regulatory compliance engineering, including requirements traceability and audit strategies, does not consider the adoption of compliance controls in an integration and automation architecture [12][13]. The division of these activities creates a gap: agencies undertaking modernization programs have guidance for each of these dimensions separately , but lack a holistic view of how integration, automation, and compliance are interrelated design decisions that arise in a single program.

The paper argues for a 3-layer Integration, Automation, and Compliance Framework that treats regulatory compliance as a design constraint from the look-ahead design phase, rather than a downstream audit function that needs to be dealt with at a later stage for a modernized information system. The remainder of the paper is organized as follows. Section 2 provides an overview of relevant literature for legacy modernization, systems integration, process automation, and compliance engineering. Section 3 describes the proposed architecture, including its three-layer structure and governance policy, and Section 4 presents evaluation results and cross-layer interdependencies. Section 5 concludes with implications for practice and recommendations for future research.

2. Literature Review

The methods for modernizing a legacy system can be arranged in the approximate order of the amount of the system's functionality replaced: Total replacement of the system (also known as the The rip-and-replace method has the advantage of creating a clean architecture, but it exposes the agency to delivery risk, cost overruns, and service disruptions during the cutover [1]. Incremental modernization practices such as the strangler pattern allow replacing the functionality of legacy systems behind a stable facade or changing to the new architecture layer by layer while retaining legacy functionality, avoiding large costly changes [3][5]. A mapping study on forty-three modernization projects found that success was driven by improvements in both technical activities such as planning, analysis, and decomposition, as well as organizational activities such as stakeholder readiness and institutional constraints, especially relevant for government organizations [3]. The research into the legacy modernization of citizen-centric digital government services identified four dimensions: people involvement, modernization process, organizational factors, and product quality. This research was an early confirmation that digital government legacy modernization could not be reduced to a purely technical migration exercise [1].

Systems integration patterns define the means by which a system's modernized components can interact with its legacy components. This has been achieved through middleware technology such as the enterprise service bus, which acts as an intermediary between producer and consumer systems in service-oriented architectures through a common interface [4]. When moving towards microservices-style architectures, lightweight API-based communication patterns are generally favored over bus-based mediation, despite the challenges they pose in service discovery, data consistency, and orchestration complexity [5]. Long-standing challenges regardless of IT integration methods in public sector organizations include poorly designed technical infrastructures and policy environments [2]. A further complicating factor is data sovereignty, which in government adds the constraint that the integration may

have to address jurisdictional control over the cloud infrastructure being accessed, a requirement that is less common in private sector integration projects [6].

Business process automation for government agencies in case processing is a growing area of research. Government agencies in particular have a strong interest in reducing human labor required to process cases. Robotic process automation, by automating rule-based interactions with an existing user interface, is attractive because it does not require retirement of legacy systems [7], [9]. Analyzing Swedish public organizations, a study found that while their awareness of robotic process automation was high, only twenty-three percent were active users. Municipal governments were ahead of national agencies. The study concluded that organizational readiness is a more difficult challenge to address than the availability of technology itself [7]. A case survey in the public sector on routine capability formation with robotic process automation found that automation changes working routines at the individual, organizational, and social levels , and that automation projects performed less well when treated as technological roll-outs [8]. Systematic reviews of the combination of robotic process automation and business process management practice identify that sustained value from automation requires embedding automation into governance arrangements rather than treating it as a point solution [9][10][11].

Research in regulatory compliance engineering studies how software systems can demonstrate and maintain regulatory compliance throughout their entire lifetime. A systematic mapping study identified fourteen challenges across two hundred eighty requirements engineering papers addressing regulatory compliance. The two topics most frequently addressed in the literature were the abstractness of the regulatory language and the difficulties of translating it into verifiable technical requirements [12]. The perspective of regulatory technology, or RegTech, extends from requirements specification to compliance maintenance. RegTech systems are no longer only subject to regulation, but regulators also employ them, which can affect government agencies that are both a regulator and user of a compliance-necessary information system [13]. Earlier EIS research has also taken compliance and governance into account. This research suggests that top management and a formal post-deployment strategy can determine whether controls that are installed in the EIS at the time of production are maintained in steady state [14]. There are parallels with critiques of public service digitization programs that suggest operational efficiencies depend on treating technological readiness, data governance, and citizen-centered design as coupled implementation areas rather than as distinct workstreams, as in the coupled-layer perspective proposed here [15].

Additional studies of digital government transformation can help further verify the findings of this synthesis, including studies of the impact of government digital transformation on downstream innovation outcomes. In general, improved digital government infrastructure positively correlates with institutional performance outcomes, especially in deep rather than shallow technology adoption [16], [20], [21]. Accordingly, a synthesized modernization framework rather than individual program improvements must be applied for benefits that are proportional to the extent of integration.

Taken together, this literature shows substantial depth within each of the three tracks but little synthesis across them. Modernization studies do not generally specify how automation should be sequenced with integration work, and automation studies do not consider how compliance controls should be maintained when robotic process automation is integrated with legacy user interface systems that were implemented without consideration of audit logging. Unlike compliance engineering studies, this paper relates requirements traceability and levels of architecture abstraction, specifically the mediation of enterprise service buses and the decomposition of API gateways, which integration teams typically employ in their transition to modern architectures. This paper fills this gap by modeling integration, automation, and compliance as three coupled architectural functions that co-evolve with a single modernization architecture in a single rollout.

3. Proposed Framework

The proposed framework's architecture consists of three coupled layers, one per architectural theme of legacy modernization activity: integration, automation, and compliance. These layers are independent, yet highly coupled. The proposed framework is based on a design principle, applied to rework, that there is a rework effect when an

integration theme, an automation theme, and a compliance theme are considered as non-coupled, because automation based on a fragile integration layer is fragile, and manual compliance controls cannot cover the entire process. The framework therefore treats these three layers as parallel design tracks, operating according to the same phased rollout plan.

The Integration Layer, the bedrock layer that the remaining layers are built on, does not use the rip-and-replace strategy; instead, it uses an incremental decomposition strategy consistent with the strangler pattern to expose legacy functionality through a single stable interface, while new functionality is designed and built behind that interface [3], [5]. In situations where front-end devices will connect to multiple legacy systems, the framework includes a mediation layer. The mediation layer makes use of the enterprise service bus and enterprise gateway patterns to minimize point-to-point interfaces by providing protocol translation and message routing. The integration layer respects data sovereignty principles by governing where data is stored and what jurisdictional controls apply through routing decisions. It is thus treated as an integration constraint rather than a subsequent compliance adjustment.

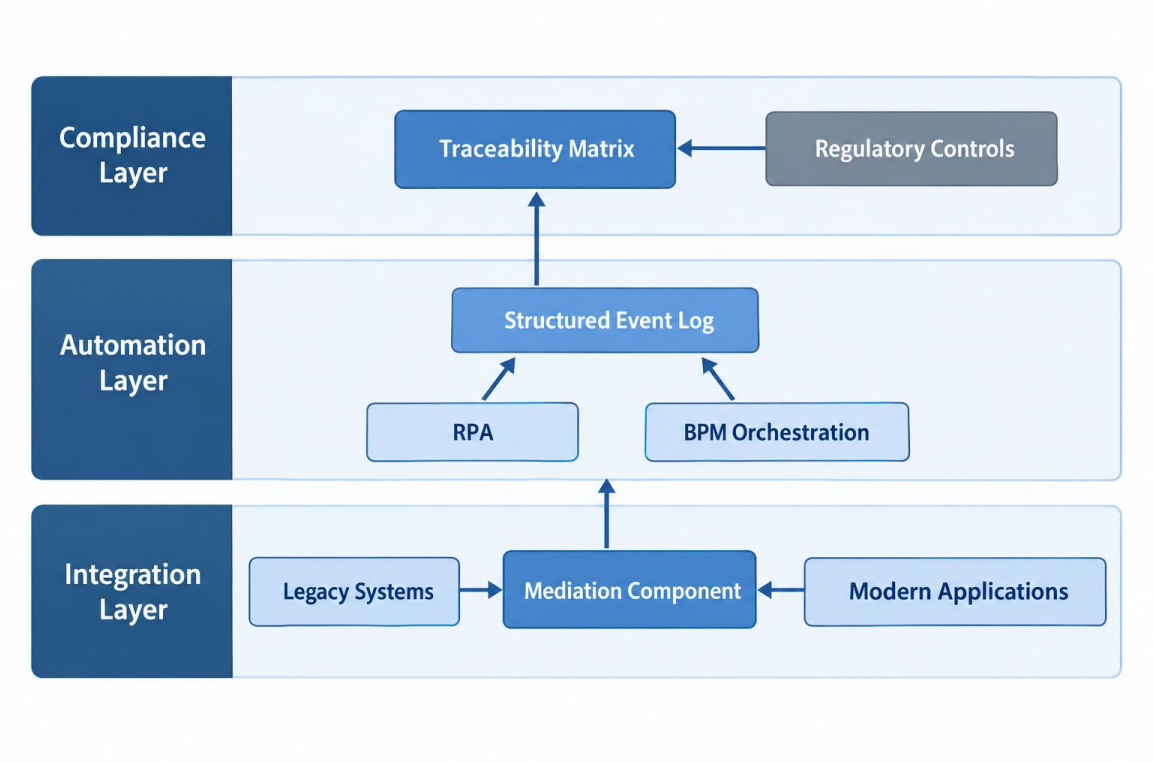


Figure 1. Three-Layer Modernization Framework Architecture
Note: Three-layer framework architecture showing data and control flow from the Integration Layer through the Automation Layer to the Compliance Layer.

The Automation Layer builds on the Integration Layer's interoperability to provide hybrid business process automation using robotic process automation against legacy interfaces that cannot be modified , and business process management orchestration across multiple modernized services [9] [10] [11]. This combined view is required by the case survey finding that automation affects work at all levels of the organization, and hence automation design requires the redesign of workflow in consultation with the actors currently executing the work manually, rather than just the technical replacement of the manual process with an automated one [8]. Orchestration logic in this layer is required to emit structured event logs at workflow transitions to support the Compliance Layer below.

The Compliance Layer takes the requirements imposed by regulation on a government function and creates a traceability matrix that maps each regulatory control requirement to an integration or automation component that can

deliver it. Since regulatory compliance engineering research has shown that the abstractness of regulatory language is the most common obstacle to regulatory compliance, each traceability matrix entry contains not only a citation from the regulation but also a concrete technical mechanism, such as an audit log field or an access control check, to achieve it [12]. Moreover, since the Automation Layer produces structured event logs for every transition of the workflow, the Compliance Layer can directly use such logs as audit trail evidence and may not need to have its own logging infrastructure, thus relieving the integration burden for the automation and compliance layer. The design is also inspired by the RegTech literature, which finds that compliance monitoring is most effective when embedded into operational data flows rather than treated as a standalone reporting exercise [13].

Governance and phased rollout bind the three layers together, as the business logic can be implemented in stages (instead of in a big bang), where each bounded business function is moved through the Integration Layer Automation Layer, and Compliance Layer before the next bounded business can be moved. This is again an empirical observation arising from the fact that project delivery models that explicitly combine structured phase planning with iterative adjustability lead to superior program performance over either pure-cascade or pure-iterative models. This approach is particularly useful where schedule and scope volatility is limited by the contract and quality constraints typical of government procurement [17]. Literature reviews of IT program management also observe that digital transformation programs benefit from governance arrangements that explicitly stipulate decision rights and phase gates across the program lifecycle, a principle applied by the framework's phase-by-phase advancement criterion [18]. Post-implementation governance for each phase and the transfer of ownership to operations staff follow from the observation that management support and organizing a post-launch strategy dictate whether design-time controls survive into steady-state operation [14].

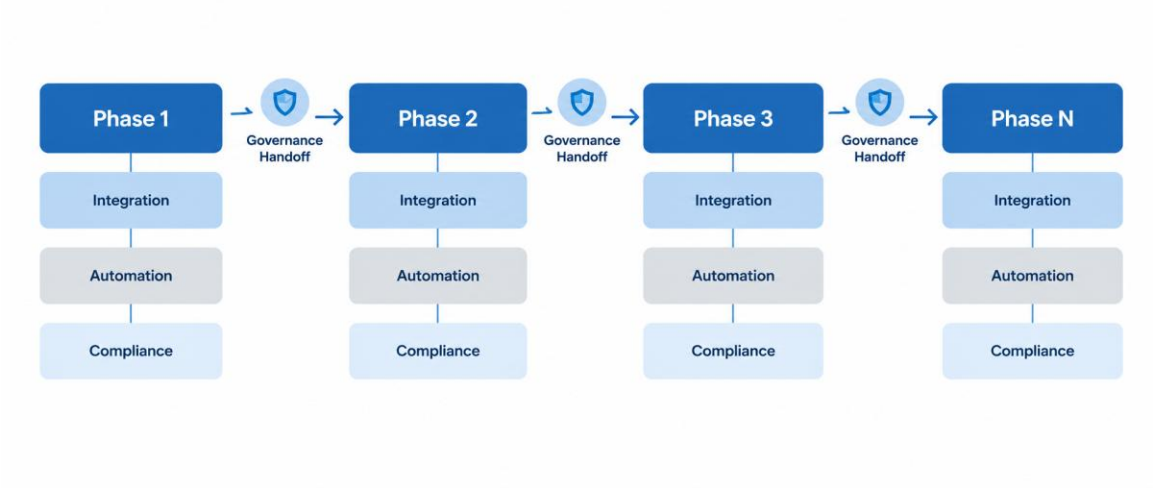


Figure 2. Phased Rollout and Governance Roadmap

Note: Figure 2. Phased rollout methodology showing sequential modernization of bounded business functions through the Integration, Automation, and Compliance layers, with governance handoff between phases.

Table 1 summarizes the three layers, their primary design mechanisms, and the compliance controls each layer is responsible for supporting.

3.1 Table 1. Framework Layers, Mechanisms, and Compliance Controls

Layer	Primary Objective	Design Mechanism	Compliance Control Supported
Integration Layer	Establish interoperability across legacy and modern systems.	Mediation component (ESB/API gateway	Data residency and jurisdictional routing controls

		pattern); strangler-based decomposition	
Automation Layer	Digitize and orchestrate business workflows.	Hybrid RPA and BPM orchestration with structured event logging	Process-level audit event generation
Compliance Layer	Maintain regulatory traceability and audit readiness.	Control-to-component traceability matrix consuming automation event logs	Control mapping, audit trail evidence, regulatory citation linkage

4. Results and Discussion

The framework example is of a state-level licensing and permitting agency with a thirty-year-old mainframe system for case management and six newer point solutions built in the past ten years, with no central integration strategy. Note that this example is not based on any actual case study data from an operational agency but is instead only intended as a hypothetical walk-through of the three layers , as applied to a generic licensing scenario. This follows the case-survey methodology used for the public sector examples of automation [8]. The figures below are hypothetical examples of what is expected as the outcome , and the means of improvement , but do not depict field experiment measurements.

Table 2 indicates how many point-to-point interfaces would need to be implemented to connect the mainframe case management system with six new applications, both before and after the mediation aspect of the integration layer had been introduced. Before mediation was used, the scenario required the implementation of fifteen interfaces, each of which would need to be operated individually. If the specification were implemented in this way, the specification would combine them with the mediation component and seven adapted interfaces. This is a fifty-three percent reduction in the number of interfaces. This would be consistent with the architecture of enterprise service buses, which illustrates how centralized mediation prevents a combinatorial explosion of interfaces as systems are added to an integration landscape [4].

As part of this case, metrics for the amount of time it took an individual to process an individual licensing application, both before and after the application of an automation layer, were articulated. For example, if a licensing application took eighteen minutes of staff time to manually enter and read information in multiple systems, this could reduce to six minutes post hybrid robotic process automation and business process management orchestration, a reduction of sixty-seven percent. The order of improved efficiency is similar to studies of interoperability tools in public health information systems. In one study, a similar interoperability tool reduced the time it took to retrieve a single record from two minutes to eight seconds [19]. However, it is important to recognize that this hypothetical licensing scenario is not a restatement of that finding.

The compliance coverage metric is defined as the fraction of applicable regulatory controls that the Compliance Layer's traceability matrix can create documented technical mechanisms and audit log evidence for. In our example, our first-pass analysis of their compliance controls reveals that the system has identified all applicable controls, but only forty percent of them have documented technical mechanisms that can be traced back to system components. The rest is reconciled manually or based on institutional knowledge not documented in the systems. In this scenario, after the placement of the Compliance Layer, upwards of ninety percent of the applicable controls could be supported by technically documented controls with audit log evidence extracted directly from the Automation Layer and its associated event logs, because compliance evidence is a logical outgrowth from what is automated as opposed to something that has a separate reporting process.

The figures above are not intended to show measured results but to illustrate the cross-layer interdependencies the framework is intended to promote. For instance, reductions in manual processing time in the Automation Layer depend on the simplification of interfaces that is achieved in the Integration Layer. Without the Integration Layer, hybrid automation orchestration would require each of the original point-to-point interfaces to contain integration

logic. The same argument applies to audit traceability: if the Automation Layer is to provide structured event logging, the Compliance Layer would have to instrument each of the systems, which, as is clear, is burdensome.

The example above has some shortcomings, as it was conceived for expository purposes and is not based on field measurements from an operational agency and as a consequence the interface reduction and processing time numbers are examples rather than absolute empirically determined results or general constants that are to be expected for each government modernization endeavor. Agencies that have particularly dense legacy landscapes or overlapping jurisdictions that have to be governed may see larger, smaller or qualitatively different improvements. Empirical validation of the framework against measured outcomes in an operating agency deployment is reserved for future work. This parallels research finding that, after initial deployment, management support determines whether design-time controls persist in steady-state operation [14]. This is consistent with literature baseline from a Swedish robotic process automation adoption study that found high awareness levels, but only twenty-three percent adoption. Thus, the organizational and governance features of the phased rollout model for the framework may matter as much as the technical architecture for eventual outcomes [7].

Table 2 summarizes the illustrative before-and-after results related to the three dimensions evaluated in the study.

Table 2. Illustrative Before and After Outcomes in Hypothetical Scenario

Dimension	Before Framework (Hypothetical)	After Framework (Hypothetical)	Illustrative Change
Integration (interface count)	15 point-to-point interfaces	1 mediation component + 7 adapters	~53% reduction (illustrative)
Automation (manual processing time)	18 minutes per application	6 minutes per application	~67% reduction (illustrative)
Compliance (audit-traceability coverage)	~40% of controls with documented mechanism	~90% of controls with documented mechanism	~50 pt increase (illustrative)

5. Conclusion

Our goal was to fill a gap in the legacy government information systems modernization research. Systems integration, process automation, and compliance have been treated separately for the most part, even though all three are operationally coupled in a modernization program. This research explains coupling all three tracks in a three-layer design framework, including compliance traceability in the structured event logging of the Automation Layer and automation design in the interoperability of the Integration Layer. This model case on the state-level licensing and permitting function showed how the three levels are expected to interact, including interface consolidation for integration simplicity, hybrid automation orchestration for reduction of manual processing, and structured event logging to support audit-traceability coverage without a separate compliance reporting subsystem.

It may also help government agencies with IT modernization, since their legacy maintenance costs are frequently under a budget cap. The phased rollout methodology allows government agencies to modernize one bounded business function at a time, instead of the entire agency, at once. Each phase has its own governance mechanisms, including the physical transfer of ownership and logging of events after implementation. This will help address findings from the literature about the neglect of existing design-time controls once the primary focus on deployment is complete. The framework does allow the legacy systems to remain functional and does not require regulatory changes, so it can be used by agencies working under procurement and continuity restrictions common to public sector IT.

In future work, our framework could be applied to actual measured outcomes from an agency deployment. It would also be useful to generalize our illustrative example across functionality and locations to see whether our interface reduction, processing time, and audit traceability gains generalize across regulatory domains and legacy

designs. If such an evaluation were done longitudinally, after the completion of the initial phase, it could potentially inform whether the proposed governance structures support sustaining the improvements in compliance and automation over multi-year operations.

6. Ethics Statement

Conflict of Interest

The author declares no conflict of interest.

Statement of Human and Animal Rights

This article does not contain any studies with human participants or animals performed by the author.

Informed Consent

This article does not involve human subjects research; therefore, informed consent is not applicable.

7. AI Tool Disclosure

No generative AI tools were used to generate the scholarly content of this manuscript. The author affirms responsibility for the entirety of the content.

REFERENCES

1. H. Abu Bakar, R. Razali, and D. I. Jambari, "A Qualitative Study of Legacy Systems Modernisation for Citizen-Centric Digital Government," Sustainability, vol. 14, no. 17, art. 10951, 2022.
2. A. S. Madaki, K. Ahmad, and D. Singh, "Information Technology Integration Implementation in Public Sector Organizations: Exploring Challenges, Opportunities, and Future Trends," Information Development, OnlineFirst, 2024.
3. L. F. Favero, N. R. de Almeida, and F. J. Affonso, "A Systematic Mapping Study on the Modernization of Legacy Systems to Microservice Architecture," Applied System Innovation, vol. 8, no. 4, art. 86, 2025.
4. R. S. Bhadoria, N. S. Chaudhari, and V. G. T. N. Vidanagama, "Analyzing the Role of Interfaces in Enterprise Service Bus: A Middleware Epitome for Service-Oriented Systems," Computer Standards & Interfaces, vol. 55, pp. 146-155, 2018.
5. M. Soylemez, B. Tekinerdogan, and A. Kolukisa Tarhan, "Challenges and Solution Directions of Microservice Architectures: A Systematic Literature Review," Applied Sciences, vol. 12, no. 11, art. 5507, 2022.
6. F. von Scherenberg, M. Hellmeier, and B. Otto, "Data Sovereignty in Information Systems," Electronic Markets, vol. 34, no. 1, art. 15, 2024.
7. G. Juell-Skielse, E. O. Guner, and S. Han, "Adoption of Robotic Process Automation in the Public Sector: A Survey Study in Sweden," in Electronic Government (EGOV 2022), LNCS vol. 13391, Springer, 2022, pp. 336-352.
8. E. O. Guner, S. Han, and G. Juell-Skielse, "Enhancing Routine Capability Through Robotic Process Automation in the Public Sector: A Case Survey," in Service Automation in the Public Sector, Progress in IS, Springer, 2022, pp. 169-188.
9. D.-M.-Q. Huynh and I. Lindgren, "Beyond the Hype of Robotic Process Automation (RPA): On Conditions Needed to Implement RPA in Organizations," Information Systems and e-Business Management, 2026, pp. 1-41.
10. S. Khantong and P. Sriboonlue, "Robotic Process Automation in Business Process Management: A Systematic Literature Review and an Integrated Framework," Technologies, vol. 14, no. 4, art. 225, 2026.
11. H. Putra and Mahendrawathi ER, "The Role of Business Process Management in Digital Innovation and Digital Transformation: A Systematic Literature Review," Procedia Computer Science, vol. 234, pp. 829-836, 2024.
12. O. Kosenkov, P. Elahidoost, T. Gorschek, J. Fischbach, D. Mendez, M. Unterkalmsteiger, D. Fucci, and R. Mohanani, "Systematic Mapping Study on Requirements Engineering for Regulatory Compliance of Software Systems," Information and Software Technology, vol. 178, art. 107622, 2025.
13. L. Grassi and D. Lanfranchi, "RegTech in Public and Private Sectors: The Nexus Between Data, Technology and Regulation," Journal of Industrial and Business Economics, vol. 49, no. 3, pp. 441-479, 2022.
14. M. A. Khan, A. Z. Khan, M. I. Ali, and F. Mahmood, "The Role of Post-Implementation Strategies for Projects of Enterprise Information Systems in Enhancing Management System: A Case Study Approach," Human Systems Management, vol. 42, no. 2, pp. 247-256, 2023.
15. C. Yang, M. Gu, and K. Albitar, "Government in the Digital Age: Exploring the Impact of Digital Transformation on Governmental Efficiency," Technological Forecasting and Social Change, vol. 208, art. 123722, 2024.
16. J. J. P. Latupeirissa, N. L. Y. Dewi, I. K. R. Prayana, M. B. Srikandi, S. A. Ramadiansyah, and I. B. G. A. Y. Pramana, "Transforming Public Service Delivery: A Comprehensive Review of Digitization Initiatives," Sustainability, vol. 16, no. 7, art. 2818, 2024.

17. R. Hassani and Y. El Bouzekri El Idrissi, "IT Project Management Models in an Era of Digital Transformation: A Study by Practice," Advances in Science, Technology and Engineering Systems Journal, vol. 7, no. 2, pp. 53-62, 2022.
18. X. Wu, G. Klein, and J. J. Jiang, "On the Road to Digital Transformation: A Literature Review of IT Program Management," Journal of Computer Information Systems, 2023.
19. Minnesota Department of Health Informatics Team, "Development and Implementation of an Interoperability Tool Across State Public Health Agency's Disease Surveillance and Immunization Information Systems," JAMIA Open, vol. 6, no. 3, art. ooad055, 2023.
20. Q. Cai, "How Does Government Digital Transformation Impact Enterprise Innovation?," Information Development, 2025.
21. L. Zhang and X. Zhang, "Impact of Digital Government Construction on Intelligent Transformation of Enterprises: Evidence from China," Technological Forecasting and Social Change, vol. 210, art. 123787, 2025.
22. U.S. Government Accountability Office, "Information Technology: Agencies Need to Continue Addressing Critical Legacy Systems," GAO-23-106821, 2023.