Asian Journal of Research in Computer Science

Volume 16, Issue 4, Page 239-256, 2023; Article no.AJRCOS.108612
ISSN: 2581-8260

Navigating the Modernization of Legacy Applications and Data: Effective Strategies and Best Practices

Sivakumar Ponnusamy [a++*] and Dinesh Eswararaj [b++]

[a] Cognizant Technology Solutions, Richmond, VA, USA.
[b] Compunnel Inc, Irvine, California, USA.

Authors' contributions

This work was carried out in collaboration between both authors. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/AJRCOS/2023/v16i4386

Open Peer Review History:
This journal follows the Advanced Open Peer Review policy. Identity of the Reviewers, Editor(s) and additional Reviewers, peer review comments, different versions of the manuscript, comments of the editors, etc are available here: https://www.sdiarticle5.com/review-history/108612

Review Article

Received: 03/09/2023
Accepted: 06/11/2023
Published: 09/11/2023

ABSTRACT

Aims: This research offers an in-depth exploration of the hurdles organizations face during legacy application modernization. The investigation delves into the primary motivations behind modernization, delineates the associated challenges, and proposes viable strategies and best practices to mitigate these issues.
Study Design: This is a Review Article which synthesizes and critically assesses a broad array of sources relevant to legacy application modernization. It amalgamates insights from various studies, offering a comprehensive overview and analysis of existing literature to derive meaningful conclusions and recommendations. Through this approach, the article provides a holistic understanding of the challenges and strategies associated with modernizing legacy systems.
Place and Duration of Study: This global study was conducted over eight years, from January 2016 to August 2023.
Methodology: This research uses a literature review to collect data. In the literature review process, a comprehensive array of data collection methods is strategically employed to ensure the

---

[++]Senior Data Engineer;
**Corresponding author: E-mail: psivakumarmca@gmail.com;*

acquisition of a diverse and pertinent body of knowledge concerning the challenges associated with modernizing legacy applications and the effective strategies and best practices to address them. It starts with searching in extensive Online Databases and Repositories, using Keyword Searches and Citation Tracking to find the relevant literature. Systematic reviews and meta-analyses give structured synthesis, while manual searches collect real-world case studies. Grey Literature supplements insights, and Evidence-Based Practices ensure rigor. Thematic Analysis sorts findings, whereas Data Management arranges data, and the Critical Appraisal Skills programme evaluates the credibility of sources. This approach is an important starting point for modernizing legacy systems and developing effective policies and guidelines.

Results: The research identifies business necessities and technological advancements as the predominant catalysts for modernization. It further elucidates the obstacles encountered by organizations during this transition, such as the intricacies of data migration, the complexity inherent in legacy systems, and issues related to user acceptance and integration. The investigation also delves into potential strategies and best practices to navigate these challenges, emphasizing the significance of selecting the right modernization approach.

Conclusion: The existing research underscores that although the path to modernizing legacy applications has obstacles, they can be navigated successfully through astute planning, strategic decision-making, and adept execution. In doing so, organizations have the potential to metamorphose their dated systems into valuable tools that resonate with current business demands and the latest technological advancements.

Keywords: Legacy application modernization; data modernization; modernization strategies; data migration; user acceptance.

ACRONYMS AND ABBREVIATIONS USED IN THE ARTICLES

CI/CD **:** *Continuous Integration/Continuous Deployment — a set of practices of software development aiming at the automation of building, testing and delivery of applications.*

DevOps **:** *A philosophy for developing and operating software that emphasizes communications, collaborations, integrations, and automations b/w the software's developer's and ITs operation's personnel's to streamline the development and deployment of high-quality software.*

SLO **:** *Service Level Objective — A set of criteria that describe the level of service quality that the user or client expects.*

IT **:** *"I have been in the same position, but that is no excuse for being rude."*

CASP **:** *Critical Appraisal Skills Programme.*

1. INTRODUCTION

The modernization of legacy computational systems in computer science is an area which has been changing for a long time. These systems typically support essential research or practical solutions components, which, however, require more and more effort to maintain, extend, and evolve with changing needs. A legacy system is any computing software, hardware, or combination derived from previous language, platform, and methodology generations. While such systems usually keep functioning and fulfil some purposes within the enterprise, they tend to degrade over time in terms of their reliability, maintainability, and interoperability with new technology trends. There are multiple reasons why it is necessary to modernize: As old systems age, they become more at risk due to modern cyber-attacks, which were never intended to be prevented by the system's initial architecture. They are typically unable to scale with growing business needs, resulting in operational bottlenecks and impeding growth. As vendors stop supporting the old systems, the enterprises become unsupported for updates, making them increasingly obsolete to the newest tech.

On the other hand, legacy systems tend to weigh organizations down. They generally necessitate expertise to run, which costs more and can become rare when the tech becomes obsolete. In addition, these systems may not be easily integrated with other new technologies resulting in data silos and inoperative inefficiencies. Thus, modernization cannot be considered an option but a need. It leverages emerging innovations, like cloud computing, analytics and AI, improving

operating costs, scalability, and organizational flexibility. However, that is not the point; the objective is for them to move from liability to an asset in driving business value and innovation.

Legacy systems are often the backbone of an organization's IT infrastructure, serving critical functions that keep operations running smoothly. However, these systems, often decades old, face challenges such as a lack of compatibility with newer technologies, scalability issues, security vulnerabilities, and high maintenance costs. As technology evolves rapidly, organizations find themselves at a crossroads: to modernize these legacy systems or continue to pour resources into increasingly obsolete technology. The crux of the issue lies in the need for legacy systems to adapt to modern-day requirements and technologies. These outdated systems often need more flexibility, scalability and essential features for implementing modern business processes. This leads to reduced competitiveness in today's digital marketplace, increased risk of security breaches, and rising costs due to maintenance and compliance challenges. The proposed solution involves adopting a multi-faceted approach to modernize legacy systems, with strategies tailored to each organization's unique circumstances. These strategies range from encapsulation, where the legacy system is made to interact with newer technologies, to complete re-architecture, where the system is overhauled to align with current needs and future scalability. Alongside these strategies, implementing best practices like comprehensive assessments, clear requirements, effective change management, and rigorous testing can contribute to a smoother and more effective modernization process.

2. METHODOLOGY

In this research, a systematic methodology was employed to ensure a comprehensive understanding of legacy application modernization. The initial phase, 'Data Collection', concentrated on discerning the underlying need for modernization and identifying the primary drivers pushing organizations toward this endeavor. This foundational phase was instrumental in setting the context for the subsequent in-depth 'Data Analysis'. During this phase, the primary focus was on elucidating various modernization-related challenges. This included delving into issues related to the legacy data stacks, challenges faced during user acceptance, hurdles encountered during data migrations, and the complexities in selecting the most appropriate modernization strategy. With the challenges and issues well-defined, the 'Results & Discussion' phase was embarked upon. This phase emphasized the different approaches for seamlessly integrating new technologies and applications. Also, it provided insights into making informed decisions regarding selecting the right modernization

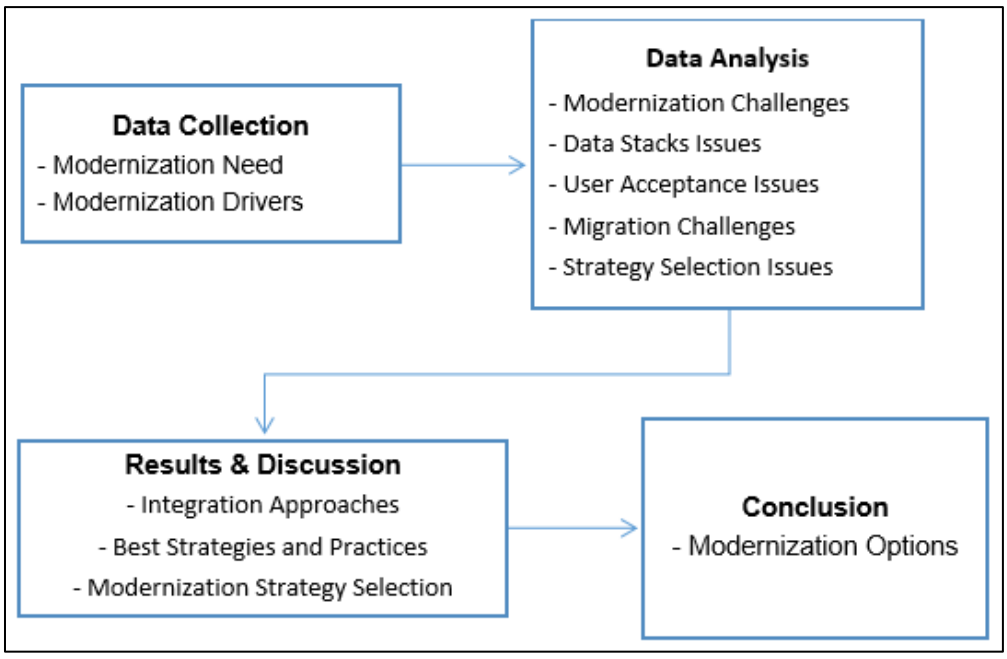

Fig. 1. Methodologies

strategy. The culmination of the research was the 'Conclusion' phase, where a holistic overview of the modernization options was presented, along with best practices to be followed. Furthermore, based on the research findings, actionable recommendations were provided to guide organizations in their legacy application modernization journey.

2.1 Busines Drivers

The global business landscape is incredibly dynamic, with new competitors, new customer expectations, and new global challenges emerging rapidly. In such an environment, agility, scalability and innovation are the survival and growth. Legacy systems, unfortunately, act as a bottleneck in these areas [1]. For instance, older systems often need more features and flexibility to integrate with modern e-commerce platforms and customer relationship management (C.R.M.) systems, which are essential to a high-quality customer experience. This may also mean that the organization's ability to pivot or scale effectively may need to improve as the organization tries to move towards new business models and processes [2].

Moreover, maintaining legacy systems usually costs higher operational costs. These systems often require specialized maintenance skills that are increasingly hard to find and expensive. There are the downtime costs, which in today's 24/7 business world can be particularly damaging. Lastly, legacy systems can impact decision-making. They generally need more advanced analytics and real-time data processing capabilities for quick decisions. The result is often missed opportunities and reduced competitiveness [3]. For these reasons, the business reasons behind legacy application modernization are compelling. By modernizing, organizations can better align their I.T. strategy with their goals, reduce costs, and increase their agility and responsiveness to market changes [4].

2.2 Technology/IT Drivers

In Search Table 1, it is evident that the rapid emergence of technologies such as cloud computing, the Internets of Things (IoT), artificial intelligence (AIs), and block chain is reshaping the business landscape [5]. Not only are these innovations transforming operations, but they also highlight the limitations of legacy systems that must be designed for today's technological prowess or cybersecurity challenges [6]. The vulnerability of these dated systems is further exacerbated by vendors discontinuing support for older software versions, leading to integration and issue resolution hurdles [7]. Moreover, the inherent inefficiencies of legacy systems, coupled with their incompatibility with efficient technologies like virtualization and containerization, can strain resources and inflate costs [8]. Such technological obsolescence also deters newer IT professionals, who are more adept with modern tools and technologies [9]. Therefore, modernizing legacy applications transcends merely adapting to technological trends—it's a strategic move to future-proof organizations, ensuring they harness contemporary technologies for enhanced efficiency, robust security, and sustained innovation [10.

2.3 Legacy System Modernization Challenges

Literature Review Table 2 showcases that modernizing legacy systems is a complex endeavor fraught with numerous challenges that can significantly impact the project's success. While the end goal is to achieve a more agile, efficient, and secure infrastructure, organizations often grapple with technical intricacies and operational issues. These challenges range from data migration versus application migration dilemmas to user acceptance, integration hurdles, and strategic decision-making. This section delves into one of the most critical aspects— the challenges concerning data migration vs application migration [11].

Fig. 2 delineates the primary challenges organizations face while modernizing legacy applications. It categorizes these challenges into broader domains: Technical debt, Organizational resistance and Operational issues.

2.3.1 Technical debt

As shown in Fig. 2, Technical debt refers to the cumulative costs and long-term consequences of using practical, short-term solutions in software development rather than choosing the best overall solution. It metaphorically describes the "debt" or additional work that accumulates when developers opt for quicker or easier approaches that are not optimal for the long term. Just like financial debt, if technical debt is not addressed promptly, it can accrue "interest" in the form of increased costs, more complex fixes, and longer development times in the future. In legacy application modernization, this manifests as

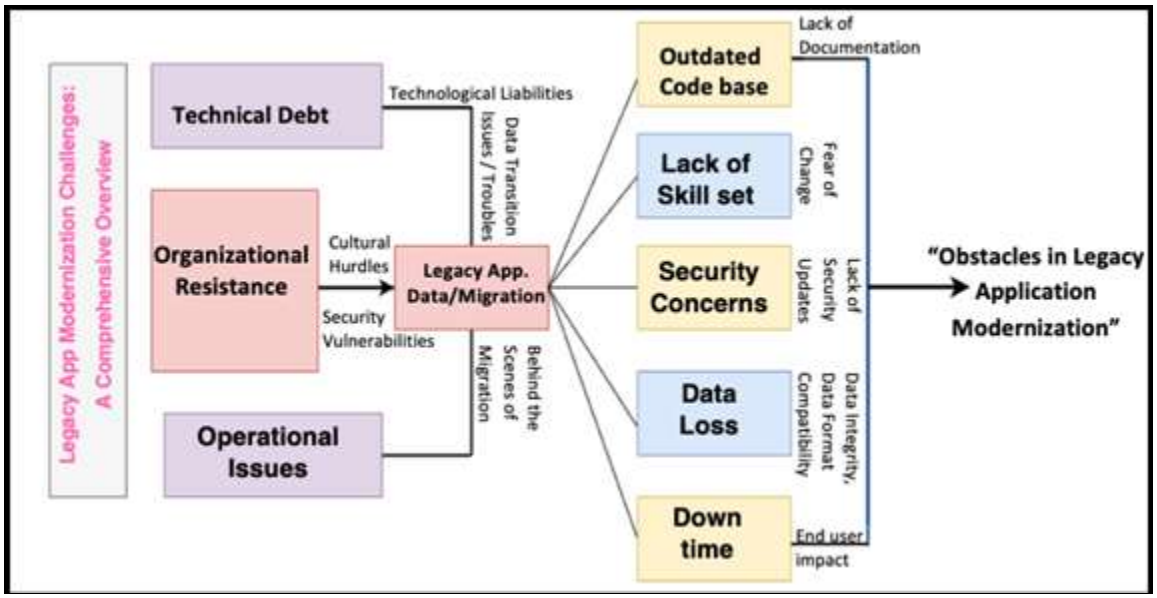

Fig. 2. Challenges in modernization of legacy applications

Technological Liabilities, where outdated systems become barriers to innovation, requiring more maintenance. Additionally, Data Transition Issues arise when organizations move data from old systems to modern platforms, facing challenges in ensuring seamless transfer and data integrity [12].

2.3.2 Organizational resistance

Organizational Resistance encapsulates the challenges that arise from ingrained organizational cultures and practices. Two primary facets of this Resistance are Cultural Hurdles, characterized by an innate reluctance to change rooted in established organizational mindsets, and Security Vulnerabilities, potential threats and exposures of sensitive data that can surface during the modernization transition, highlighted in Fig. 2 [13].

2.3.3 Operational issues

Operational Issues refer to the daily challenges that organizations face when operating or maintaining legacy systems, especially as the digital environment continues to evolve rapidly. These issues often stem from the inherent limitations of older systems trying to function efficiently in a modern, dynamic digital landscape [14].

2.3.4 Legacy application/data migration

Legacy Application/Data Migration encapsulates the challenges of transforming ageing systems to modern platforms. A prominent concern is the outdated code base, which presents hurdles in integrating or adapting archaic coding standards or languages that are no longer efficient or supported. Coupled with this is the need for more professionals familiar with such old technologies or systems, creating a skill gap in organizations. As they migrate, security becomes paramount with looming threats of unauthorized access or data breaches. The migration process is also fraught with unintentional data loss, which can be catastrophic for businesses. Lastly, the entire migration process can lead to periods of downtime, rendering systems inoperative and potentially interrupting standard organizational operations, as highlighted in Fig. 2 [15].

2.4 Complexities in Legacy Data Stacks

In CASP Table 3, the validity of the included studies was rigorously evaluated. Legacy data stacks are at the forefront of modernization challenges due to their inherent complexities, inefficiencies, and associated costs. These systems often rely on ageing database management systems and storage architectures that struggle to meet the demands of today's high-velocity, high-volume data [16]. One of the key complexities lies in the layered nature of legacy data stacks. Over the years, various layers and components have been added, resulting in intricate, opaque systems that are challenging to navigate. This complexity not only hinders the migration process but also increases the risk of errors and data corruption [17].

Furthermore, legacy data stacks are known for their inefficiencies. Unlike modern database management systems offering indexing, real-time analytics, and rapid data retrieval, legacy systems must catch up. These inefficiencies can disrupt business processes and put organizations at a disadvantage in a fast-paced digital landscape [18]. The costs associated with maintaining legacy data stacks can be substantial. These systems require specialized skills for management and operation, and finding experts well-versed in older technologies becomes progressively difficult and costly as time goes on. Additionally, legacy systems tend to consume more power and physical space compared to their modern counterparts, contributing to higher operational expenses [19]. To sum up, legacy data stacks present a complex challenge beyond technical aspects. Modernization is not merely an upgrade; it has become a business imperative. It aims to streamline operations, enhance decision-making, and reduce overall costs. By addressing these challenges, organizations can position themselves for a more agile and competitive future [20].

2.5 Navigating User Acceptance Challenges

In Theme Table 4, the studies show that user acceptance is one of the often overlooked yet critical challenges in legacy modernization. Employees using a legacy system for years may resist change, primarily because of the learning curve associated with new systems. This resistance can lower productivity, morale, and project failure if not managed adequately [21]. People are generally averse to change, especially when it involves a tool or system they use daily. This psychological barrier can be a significant barrier to modernization. Employees might be concerned about how the new system will impact their work routines and how they will adapt to it [22]. Effective training programs can alleviate some of these concerns. However, training is more than just teaching employees how to use the system. It is also about helping them understand its benefits to their work and organization. Proper onboarding processes, including comprehensive training modules and readily available support, can go a long way in easing the transition [23]. Strategic change management is crucial for successful user acceptance. Regular communication about why the change is happening, the benefits, and how it will affect individual roles can help set the right expectations. Feedback loops should also be established to address concerns and issues promptly [24]. In summary, user acceptance is a multifaceted challenge beyond mere training. It involves breaking down psychological barriers, providing robust training, and effective change management to ensure that the modernization effort is both a technical success and a human one [25].

2.6 Data Migrations Challenges

Literature Review Table 2 highlights that data migration is an essential phase during the legacy system modernization process, with specific problems that affect the whole transformation's quality greatly. This challenge can be broken down into several major categories. Organizations often need a complete picture of their legacy systems' source data and code. As such, organizations might have lost information over time due to employee attrition or retirement. The information must be included to ensure accurate data mapping, leading to data consistency. Enterprises must truly grasp the underlying system's source data and source code in such situations. Any ignorance can create data mapping issues that result in inconsistent data and errors during the migration [12]. Confusing or ever-changing requirements can become a big obstacle for Data Migration. Suppose the project team needs an accurate understanding of what the new system is designed to achieve. It will likely cause misalignment between legacy data and the new system, leading to inefficiencies and poor data quality [19]. Paying attention to the validation of a new system's implementation is critical. Following migration, rigorous validation is necessary to validate end-to-end functionality; otherwise, the system may lead to data loss or data quality problems. The validation exercise embraces several activities, such as thorough testing to confirm the data integrity, consistency, and conformity with the set business rules. For instance, information loss, data corruption, and problems with data quality can happen if comprehensive data verification is missing [15]. The validation process should include data integrity, completeness, and confirmation that the migrated data meets the business objectives [18]. Validate in this manner to avoid operational disruptions and jeopardize the success of migration efforts.

CASP Table 3 indicates that Legacy systems often need more complete or consistent data due to years of use and maintenance. This data may

not conform to modern data standards, making it difficult to migrate seamlessly. The need for data cleansing and transformation adds complexity to the migration process [21]. Legacy system documentation usually needs to be included. Poorly documented data structures, dependency graphs and business rules open the possibility of errors when relying on reverse engineering and guessing [25,26]. These challenges require careful planning, detailed Documentation, clear requirements, and extensive testing. Organizations should also consider looking outside for expertise to safely and securely navigate the intricacies of data migration. Effective data migration is critical to guarantee that the modernized system can run accurately with clean, trustworthy and consistently formatted data – vital factors enabling business continuity and expansion [27].

2.7 Existing Research Contributions

The contribution of this work lies in its comprehensive exploration and analysis of the challenges and strategies associated with legacy application modernization. It provides valuable insights into the often underestimated aspect of user acceptance, emphasizing the importance of addressing psychological barriers and implementing effective change management. Furthermore, this study sheds light on the complexities, inefficiencies, and costs associated with legacy data stacks, highlighting the need for modernization as a business imperative. It also delves into the challenges of choosing the right modernization strategy, considering various options and their implications. The work contributes by synthesizing these key challenges and offering actionable strategies and best practices for organizations embarking on the modernization journey. It is a valuable resource for decision-makers, IT professionals, and researchers, guiding them towards successful legacy modernization projects that align with modern business needs and technological capabilities.

2.7.1 Ensuring seamless integration of new technologies

Integrating new applications and technologies is crucial in modernizing a legacy system. Integration may create operational inefficiencies, siloed data, and potential financial risks if data is lost during integration [26]. The integration piece is generally more complicated because it typically means that we must ensure that everything integrates well with the old (legacy) systems being replaced. New applications may demand new data formats or communication protocols involving data transformation and adaptation operations that take up much of the developer's time and are prone to errors [27]. The risk is that one could lose information and suffer a fortune at the integration stage. Data loss can disrupt business continuity, diminish clients' credibility and infringe upon lawful boundaries, especially where data is deemed personally confidential or regulated; data lost information cost, whether temporary or permanent, can impact business reputation and expansion. A properly designed integration plan helps overcome the hurdles. Such a strategy would involve thorough testing to determine the potential for data loss or incompatibility problems before full-scale implementation. As well as disaster recovery measures must be established to recover the data if any data is lost. DevOps can also emerge from modernized DevOps practices and CI/CD pipelines that support them. These approaches facilitate small-step improvements, constant testing, and smooth rollbacks when things go wrong; this makes the whole deployment process easier to deal with and more robust. Lastly, users can only do with the integration piece for modernization projects. This can only be achieved with diligent planning, rigorous testing and adherence to advanced software development practices to reduce any chance of losses (costs) from data loss [28-30].

2.7.2 Strategically choosing the right modernization approach

Choosing an optimal modernization strategy for legacy applications is a critical move that intertwines with numerous complexities and far-reaching consequences in the direction of the entire project. This process has become more complex when aligning strategy with organizational goals while carefully allocating budgets considering timeframes and technical limitations [31]. One right move, such as an accurate strategy selection, can translate into numerous negative impacts like delayed projects and high costs, ultimately leading to project failure. This highlights why it is critical to choose an appropriate strategy that would keep the project safe from possible fault-finding and guide it towards its intended goals [31].

Modernization landscapes provide organizations with numerous strategies, including comprehensive system makeovers such as re-architecting and agile approaches to

modernization. The decision-making process for these strategies is complex because each has pros and cons. For example, a company may revamp all the systems, which can be expensive and time-consuming, but it is worth it in the future. An agile modernization methodology can bring immediate wins but may only provide some of the benefits of a comprehensive modernization. The fact is that organizations have to be very much aware of their objectives, limits and trade-offs between short-term profits and potential long-term benefits when determining a modernization strategy [32]. Another issue is aligning different project stakeholders with their interests. Business leadership may focus more on short-term success and cost reduction, while the IT team focuses on long-term strength and flexibility. A solution that pleases everyone is often not simplistic. Accurate diagnosis of the problem is hard but critical. As with any decision, there is always a trade-off — balancing potential benefits versus risks for each approach. Organizations must consider the pros and cons of each method. This list may include market, credit, operational, and project risks for technological complexity or an absence of internal competencies [33]. The flood of data and pundits may also overwhelm humans, causing them to get analysis paralysis. Organizations may need assistance in choosing the right choice [34]. The best transformation approach is a complex decision based on knowledge of the organization's needs and technical potential [35]. This typically entails the following steps: strategic planning; engagement of key stakeholders; identification, quantification, and mitigation of potential risks; development of relevant information for decision-making; identification and evaluation of alternatives [36].

3. RESULTS AND DISCUSSION

Legacy application modernization is critical in addressing the multifaceted challenges inherent in outdated systems. The hybrid approach, which encompasses strategic methods combined with industry best practices, stands out as the optimal path to navigate these challenges. This strategy not only addresses the up-to-date challenges but also sets up the adoption of advanced methodologies. [2].

3.1 Legacy Application Modernization Options

The modernization and development strategy holds significant sway in determining the success trajectory of a project. Given the diverse business environments and unique challenges, companies are often faced with the daunting task of selecting the most appropriate modernization strategy. A plethora of options are available, each with its pros and cons, as shown in Literature Review Table 2 [10].

Encapsulation is one such approach where all components of the legacy system are amalgamated into a unified package and exposed as SOAP services. This method allows organizations to harness the existing capabilities of the legacy system while incrementally adding new functionalities [12]. Another straightforward strategy is Rehosting, commonly called "lift and shift". Here, the legacy application is migrated to a cloud-native infrastructure with minimal changes to its codebase. Though a seemingly simple solution, it might only partially unlock some modernization benefits.

Replatforming offers a more nuanced approach, necessitating minor modifications to the legacy application to adapt to a newer infrastructure, like transitioning from a traditional data centre to the cloud. This approach strikes a balance, ensuring scalability and flexibility while minimizing drastic changes [14]. For organizations willing to invest heavily in the short term for long-term gains, Refactoring is an option. This involves a comprehensive application rewrite, often transitioning from a monolithic structure to a microservices architecture, integrating contemporary design patterns. This strategy, while resource-intensive initially, promises substantial long-term benefits.

Lastly, Re-architecture is the most extensive form of modernization. It requires a total revamp of the application, transitioning it from its monolithic base to a microservices architecture or integrating cutting-edge design patterns. Though this approach demands a significant upfront investment, it ensures long-term agility and scalability, which are crucial for staying competitive in the ever-evolving digital landscape [16].

It's worth noting that organizations often employ a combination of these strategies based on their specific needs and constraints. For instance, the hybrid approach of encapsulating core components during Rehost/Replat form while concurrently migrating less crucial parts to the cloud presents an avenue to maximize benefits and mitigate risks [17].

3.2 Overcome Challenges: Strategies and Best Practices

CASP Table - 3 indicates that organizations increasingly recognize the importance of adopting a systematic approach, paired with best practices, regarding legacy application modernization. This section outlines the predominant strategies and best practices to pave the way for a more streamlined and effective transformation experience [18].

3.2.1 Strategies

Literature Review Table 2 showcases that at the onset, a comprehensive assessment is imperative. This involves conducting a thorough audit of the existing legacy system to understand its architecture, dependencies, data, and business process flow. Such an assessment lays the foundation for informed decision-making regarding the modernization strategy [19]. Defining requirements is another crucial step. It is essential to articulate the expectations for the modernization initiative, ensuring all key stakeholders understand the anticipated benefits of the new system. This clarity in requirements safeguards against potential misinterpretations and scope creep [20]. In addition to these, embracing effective change management tactics is indispensable. Adopting agile methodologies can be instrumental in countering any resistance from end-users. It is beneficial to engage with them early, provide both formal and informal training, and ensure consistent support during and after the modernization process [21]. Furthermore, the emphasis on rigorous testing and validation must be balanced. Every module, be it related to data migration, features, or integration, must undergo stringent testing to validate its adherence to service-level objectives, security protocols, and compliance standards [22].

3.2.2 Best practices

In Theme Table 4, the studies reveal that documentation is a cardinal best practice. Keeping the documentation updated throughout the modernization project aids in comprehending the system's architecture, configurations, and processes. It also becomes an invaluable resource for future maintenance and troubleshooting activities [25]. Another best practice to consider is the incremental approach to modernization. Rather than making sweeping changes all at once, organizations can opt for a phased transformation. This strategy not only facilitates quicker wins but also minimizes disruptions in daily operations, allowing for a smoother transition [27]. Continuous monitoring and maintenance post-implementation are also paramount. Organizations must remain vigilant, tracking the performance, security, and user satisfaction metrics of the modernized system. Being proactive in addressing any emerging issues ensures the longevity and success of the system [29]. Finally, seeking external expertise can be a game-changer. Collaborating with consultants, modernization specialists, and other external stakeholders can offer fresh perspectives and validate that the project aligns with industry-standard development techniques [30].

Thereby, by adeptly amalgamating these strategies and best practices, organizations can surmount many of the hurdles associated with legacy application modernization. The result of such a meticulously planned and executed initiative is an agile, efficient, and competitive IT infrastructure poised to meet the demands of the future [32].

Fig. 3 depicts a use case diagram illustrating the key actors and use cases involved in legacy application modernization, following UML heuristics [33].

Actors:

1. **Developers:** Updates codebase to align with modernization goals while resolving technical debts as well as update documentation or user manual.
2. **Trainer/Change Owner:** Effectively communicates with stakeholder to convince for modernization and maintains stakeholder communications. The action of enhancing skills and provide training to address organizational resistance.
3. **Security Specialist:** Perform security assessment and propose patching strategies and firewall rules.
4. **DevOps Engineer:** Represents the entity responsible for handling Operational/migration tasks.
5. **Legacy System Users:** Current system users including operational and end users.

Use Cases:

1. **Technical Debt:** Updating the existing codebase to align with modernization goals.

2. **Security Threats:** Incorporating industry best practices into modernization processes
   - **Regular Audit Security Assessments:** The action of conducting regular security audits and assessments.
   - **Patch Management:** The action of managing and applying patches to address security vulnerabilities.
   - **Firewall Configuration:** The action of configuring firewalls for enhanced security.
3. **Operational Issues:**
   - **Schedule Upgrades:** The action of planning and scheduling system upgrades.
   - **Minimal Downtime:** The action of minimizing system downtime during upgrades.
   - **Regular Backups:** The action of regularly testing data backups/recovery (continuous operation testing)
4. **Migration Strategies:** The action of defining and implementing strategies for legacy application modernization and plan for recovery options.

This diagram visually represents the actors involved in legacy application modernization and the corresponding use cases they perform to address challenges and implement best practices [34].

3.3 Enhancing Sustainability through Legacy-To-Modern Systems Transition

The study on navigating from legacy to modern systems has significant relevance to sustainability in several ways, which can be justified as follows:

Legacy systems, often anchored to outdated hardware and software, are inherently resource-intensive, consuming more energy and challenging to maintain. Conversely, modern systems, designed with scalability and efficiency, can significantly reduce an organization's environmental footprint. This transition embodies sustainable practices, emphasizing the optimization of resource utilization [35]. Moreover, while legacy systems may lack features that support green computing, modern systems are typically built with eco-friendly features, resulting in reduced energy consumption and a minimized carbon footprint. As sustainability focuses on reducing environmental impacts, transitioning to modern systems resonates with this objective. Another critical sustainability concern is electronic waste. Outdated components from legacy systems contribute significantly to e-waste [36]. However, modernization strategies, such as

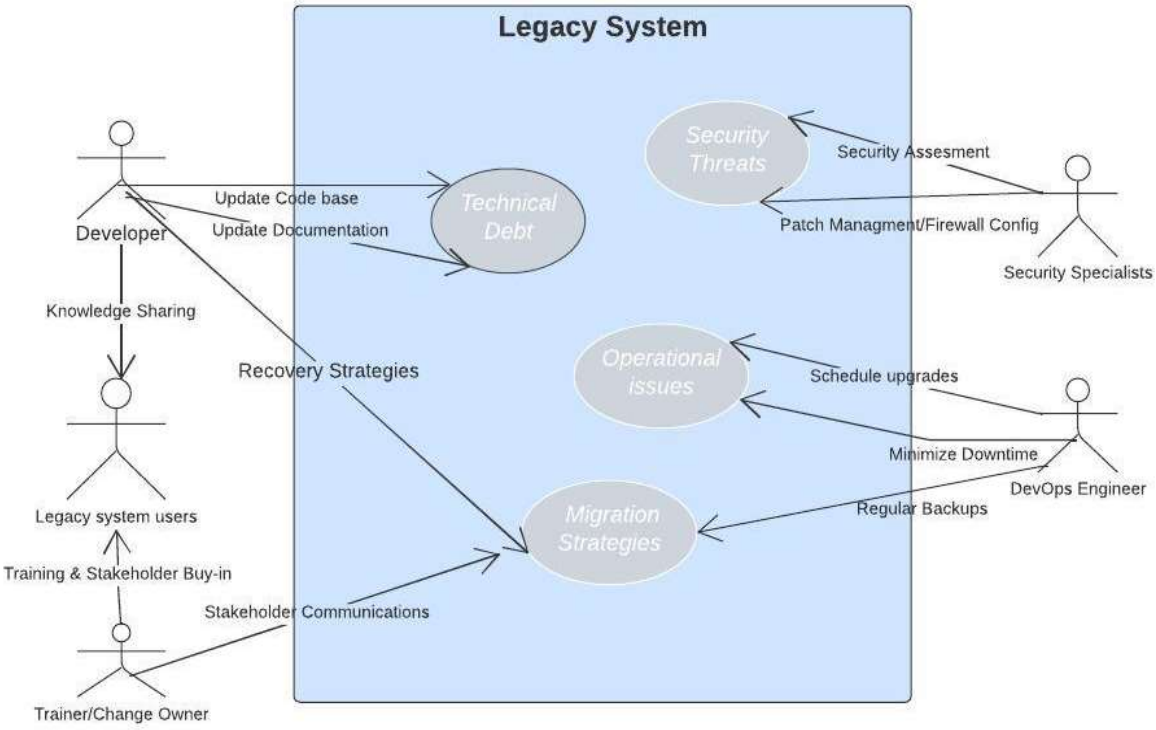

Fig. 3. Legacy application modernization

rehosting or re-platforming, can prolong the life of existing assets or ensure their responsible disposal, aligning with sustainability goals to reduce e-waste. Furthermore, sustainability is synonymous with resilience and adaptability. Legacy systems, due to their antiquated nature, are often more susceptible to disruptions and security threats [37].

In contrast, modern systems are designed to be more secure, adaptable, and resilient, ensuring business continuity even under challenging conditions. Long-term viability is also a cornerstone of sustainability. With legacy systems becoming potentially obsolete or increasingly expensive to maintain, an organization's future viability can be at risk. By adopting modernization strategies, organizations can future-proof their IT infrastructure, ensuring they remain competitive and sustainable [38]. Finally, the economic dimension of sustainability must be considered. Legacy systems, associated with high maintenance costs, can strain an organization's financial sustainability.
In contrast, modernization can usher in cost savings through enhanced efficiency, reduced maintenance costs, and improved operational capabilities [39].

To sum up, the study underscores the direct link between transitioning from legacy to modern systems and sustainability. This transition not only addresses resource efficiency, environmental impact reduction, and e-waste minimization but also reinforces business continuity, long-term viability, and economic sustainability. Adopting modernization practices is thus aligned with sustainable principles, fostering a more environmentally and economically responsible approach to IT systems management [40].

4. CONCLUSION

Legacy application modernization is not just a technological necessity; it's a strategic endeavor that paves the way for a future-ready, sustainable, and efficient organization. In this comprehensive exploration of the challenges and strategies associated with legacy modernization, the work stands out in its depth, coverage, and actionable insights. The intricacies of navigating legacy systems — from data stacks to user acceptance, data migration to system integration — are manifold. Yet, this work not only enumerates these challenges but also offers a structured framework *(*Highlighted in Fig. 3) of strategies and best practices to tackle them effectively. The strategies, such as conducting comprehensive assessments, defining clear requirements, and adopting agile methodologies, paired with best practices like continuous documentation, phased transformation, and vigilant post-implementation monitoring, provide a roadmap for organizations embarking on this journey. Significantly, this study underscores the relevance of modernization in sustainability. It highlights how transitioning from legacy to modern systems contributes to resource efficiency, environmental impact reduction, e-waste minimization, business resilience, long-term viability, and economic sustainability. This perspective elevates the discourse on legacy modernization from a purely technical narrative to one that resonates with broader organizational and societal goals. But perhaps the most notable contribution of this work lies in its synthesis of the multifaceted challenges of legacy modernization and its offering of actionable strategies. It serves as a valuable compass for decision-makers, IT professionals, and researchers, guiding them through the maze of legacy modernization towards successful outcomes that align with contemporary business needs and technological capabilities. In essence, this work doesn't just detail the 'what' and 'why' of legacy application modernization; it illuminates the 'how,' offering a beacon for organizations striving to transform their legacy systems into dynamic, efficient, and sustainable assets for the future.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

REFERENCES

1. Balasubramaniam S, Kavitha V. A survey on data encryption techniques in cloud computing. Asian Journal of Information Technology. 2014;13(9):494-505.
2. Balasubramaniam S, Kavitha V. Hybrid Security Architecture for Personal Health Record Transactions in Cloud Computing. Advances in Information Sciences and Service Sciences. 2015; 7(1):121.
3. Chenthara S, Ahmed K, Wang H, Whittaker F. Security and privacy-preserving challenges of e-health solutions in cloud computing. IEEE Access. 2019;7:74361-74382.

4. Raghavendra S, Reddy CS, Geeta CM, Buyya R, Venugopal KR, Iyengar SS, Patnaik LM. Survey on data storage and retrieval techniques over encrypted cloud data. International Journal of Computer Science and Information Security. 2016; 14(9):718.
5. Rana, ME, Kubbo M, Jayabalan M. Privacy and security challenges towards cloud-based access control. Asian. Journal of Information Technology. 2017;16(2-5):274-281.
6. Tripathi S, Mishra V. Determinants of cloud computing adoption: A comparative study. Pacific Asia Journal of the Association for Information Systems. 2019;11(3):3.
7. Uma B, Sumathi S. A Survey about cloud computing and an improved method of data security using watermarking technique with RSA Algorithm in Cloud Environment. Asian Journal of Research in Social Sciences and Humanities. 2017; 7(5):325-336.
8. AWS. Best Practices for Modernizing Legacy Applications. Are we speaking the industry language? The practice and literature of modernizing legacy systems with microservices. In 15th Brazilian Symposium on Software Components, Architectures, and Reuse. 2021;61-70.
9. Beebe NH. A Complete Bibliography of ACM Queue: Tomorrow's computing Today; 2023.
DOI: 10.4204/eptcs.163.1
10. Brinkmann S, et al. Challenges in Modernizing Legacy Systems to Microservices Architecture. In 2017 IEEE International Conference on Software Architecture (ICSA); 2017.
11. Carpenter DS. Legacy Modernization: Strategies for Modernizing Systems and Applications. Auerbach Publications; 2003.
12. Clements PC, Kazman R, Klein MH. Modernizing Legacy Systems: Software Technologies, Engineering Processes, and Business Practices. Addison-Wesley Professional; 2015.
13. Debusmann M, Schmid M, Kroeger R. Model-driven self-management of legacy applications. In IFIP International Conference on Distributed Applications and Interoperable Systems (pp. 56-67). Berlin, Heidelberg: Springer Berlin Heidelberg; 2005, June.
DOI: 10.1007/11498094_6
14. Ekanem BA, Woherem E. Legacy components stability assessment and ranking using software maturity index. International Journal of Computer Applications. 2016;134(13):22-30.
DOI: 10.5120/ijca2016908157
15. Gartner, Inc. Strategies for Modernizing Legacy Applications. Gartner, Inc. Research Note; 2018.
16. Gupta AA, Vaishnavi V. Legacy Systems: Transformation Strategies. CRC Press; 2014.
17. Hamza HS, et al. A framework for assessing legacy system modernization challenges. In 2019 21st International Conference on Advanced Communication Technology (ICACT); 2019.
18. Ilechko P. Modernizing Legacy Applications: Designing, Coding, and Testing Java Applications. Addison-Wesley Professional; 2005.
19. Infosys Ltd. Modernizing Legacy Applications in the Cloud Era. Whitepaper; 2020.
20. Jones PM. Modernizing Legacy Applications in PHP. Leanpub; 2016.
21. Khan M, Ali I, Nisar W, Saleem MQ, Ahmed AS, Elamin HE, Shafiq M. Modernization Framework to Enhance the Security of Legacy Information Systems. Intelligent Automation & Soft Computing. 2022;32(1).
DOI: 10.32604/iasc.2022.016120
22. Kiran Mallidi R, Sharma M, Singh J. Legacy Digital Transformation: TCO and ROI Analysis. International Journal of Electrical and Computer Engineering Systems. 2021;12(3):163-170.
DOI: 10.32985/ijeces.12.3.5
23. Koegel M, et al. Legacy systems challenges and opportunities: A Case Study in Application Modernization. Procedia Computer Science; 2016.
24. Kumar A, Garg H. Modernizing Legacy Systems with Microservices and APIs. O'Reilly Media; 2018.
25. Mainetti L, Paiano R, Pandurino A. Migros: A model-driven transformation approach of the user experience of legacy applications. In Web Engineering: 12th International Conference, ICWE 2012, Berlin, Germany, July 23-27, 2012. Proceedings 12. Springer Berlin Heidelberg. 2012;490-493.
DOI: 10.1007/978-3-642-31753-8_51
26. Mayor J, Gomez P, Chang F, Lupyan G. Connectionism coming of age: legacy and future challenges. Frontiers in Psychology. 2014;5:187.
DOI: 10.3389/fpsyg.2014.00187

27. Microsoft Corporation. Modernizing Legacy Applications: A Microsoft Approach. Microsoft Corporation. Whitepaper; 2020.
28. Pradhan A. Legacy Modernization: Transforming Your Legacy Systems to the Modern Age. Apress; 2020.
29. Radhakrishnan H, Rouson DW, Morris K, Shende S, Kassinos SC. Using coarrays to parallelize legacy Fortran applications: Strategy and case study. Scientific Programming. 2015;2015:2-2.
DOI: 10.1155/2015/904983
30. Rodríguez-Echeverría R, Conejero JM, Clemente PJ, Preciado JC, Sánchez-Figueroa F. Modernization of legacy web applications into rich internet applications. In Current Trends in Web Engineering: Workshops, Doctoral Symposium, and Tutorials, Held at ICWE 2011, Paphos, Cyprus, June 20-21, 2011. Revised Selected Papers 11. Springer Berlin Heidelberg. 2012;236-250.
DOI:10.1007/978-3-642-27997-3_24
31. Rodríguez-Echeverría R, Conejero JM, Linaje M, Preciado JC, Sánchez-Figueroa F. Re-engineering legacy web applications into rich internet applications. In Web Engineering: 10th International Conference, ICWE 2010, Vienna Austria, July 5-9, 2010. Proceedings 10. Springer Berlin Heidelberg. 2010;189–203.
DOI: 10.1007/978-3-642-13911-6_13
32. Satyanarayanan M, Eiszler T, Harkes J, Turki H, Feng Z. Edge computing for legacy applications. IEEE Pervasive Computing. 2020;19(4):19-28.
DOI: 10.1109/mprv.2020.3026229
33. Savio P, Scionti A, Vitali G, Viviani P, Vercellino C, Terzo O, et al. Accelerating legacy applications with spatial computing devices. The Journal of Supercomputing. 2023;79(7):7461-7483.
DOI: 10.1007/s11227-022-04925-2
34. Sharma R, et al. Modernizing Legacy Systems: Leveraging Microservices and DevOps. IBM Redbooks; 2017.
35. Smith JD, Brown AB. Legacy application modernization: A Comprehensive Framework. Journal of Software Engineering and Applications. 2019;12(1): 12-29.
36. Thompson EL, Patel SR. Legacy Application Modernization: A Case Study of Industry Best Practices. International Journal of Software Engineering & Applications. 2017;8(3):129-141.
37. Turner RK, Davis PJ. Legacy Application Modernization: Challenges and Solutions. In Proceedings of the International Conference on Software Engineering (ICSE). 2015;2015:145-154.
38. Ulrich WM, Newcomb PH. Legacy Modernization: A Pragmatic Approach. Cutter Consortium; 2007.
39. Van Kranenburg HJ, Romeijnders JW. Challenges and Strategies for Modernizing Legacy Systems. International Journal of Computer Applications; 2011.
40. Betz JW. Engineering satellite-based navigation and timing: global navigation satellite systems, signals, and receivers. John Wiley & Sons; 2015.

APPENDIX

Table. 1 Search databases

Database	Keywords	Search Results
SQLite	Legacy system modernization challenges	45
PsycINFO	User acceptance, change management	32
Scopus	Data migration, integration challenges	50
Web of Science	Modernization strategies, best practices	60

Table 2. Literature review

Study	Objective	Findings
(1) AWS [8]	Explore best practices for modernizing legacy applications.	Explores the modernization of legacy systems using micro services, providing insights and recommendations of the industry practices and literature.
(2) Beebe [9]	Provide a bibliography of ACM Queue regarding computing trends.	N/A
(3) Brinkmann et al. [10]	Examine challenges in modernizing legacy systems to micro services architecture.	Identifies challenges and potential solutions in transitioning from legacy systems to micro services architecture.
(4) Carpenter [11]	Discuss strategies for modernizing legacy systems and applications.	Offers strategies and approaches for modernization, emphasizing the importance of legacy system transformation.
(5) Clements et al. [12]	Explore modernizing legacy systems from a software and business perspective.	These include elements of software technologies; engineering; and, business practices, all geared the state of the art.
(6) Debusmann et al. [13]	Development of a Model-Driven Self-Management Strategy to Manage Legacy Applications.	Discusses a model-driven approach to self-managing legacy applications for improved efficiency and maintenance.
(7) Ekanem & Woherem [14]	Assess the stability of legacy components using a software maturity index.	Proposes a software maturity index for evaluating the stability of legacy components in modernization efforts.
(8) Gartner, Inc. [15]	Provide strategies for modernizing legacy applications.	Offers insights and recommendations from Gartner on modernization strategies to meet evolving business needs.
(9) Gupta & Vaishnavi [16]	Examine transformation strategies for legacy systems.	Discusses strategies for transforming legacy systems to align with contemporary technology and business requirements.
(10) Hamza et al. [17]	Present a framework for assessing legacy system modernization challenges.	Introduces a framework for evaluating challenges in legacy system modernization, aiding in decision-making.
(11) Ilechko [18]	Focus on designing, coding, and testing Java applications for modernization.	Discusses the design, coding, and testing aspects of modernizing legacy applications using Java.
(12) Infosys Ltd. [19]	Explore modernizing legacy applications in the cloud era.	Provides insights into modernization strategies and considerations specific to cloud-based environments.
(13) Jones [20]	Discuss modernizing legacy applications in PHP.	Offers guidance on modernizing legacy applications using PHP, a widely-used

Study	Objective	Findings
		scripting language.
(14)Khan, [21]	Build a modernization framework for securing legacy information systems	It presents a model for making legacy applications secure during transformation.
(15)Kiran-Mallidi's et al., [22]	Legacies digitals transformations TCOs/ ROIs analysis/ Calculate TCO/ROI for user legacy digital transformation.	Examines the financial aspects of legacy system modernization, focusing on TCO and ROI analysis.
(16)Koegel et al. [23]	Explore challenges and opportunities in legacy systems with a case study in application modernization.	Presents a case study on application modernization, highlighting challenges and opportunities in legacy systems.
(17)Kumar & Garg [24]	Discuss modernizing legacy systems with microservices and APIs.	Explores the role of microservices and APIs in modernizing legacy systems, emphasizing their importance.
(18)Mainetti et al. [25]	Model-Driven UX Improvement in Legacy Systems.	Enhancing user experience in legacy applications by using a model-driven methodology.
(19)Mayor et al. [26]	Explore connectionism and its relevance in modernization challenges.	How connectionism is important in solving the challenges of modernity.
(20)Microsoft Corporation [27]	Provide insights into Microsoft's approach to modernizing legacy applications.	Presents Microsoft's approach to modernization, highlighting strategies and considerations.
(21)Pradhan [28]	Focus on legacy modernization and transformation strategies.	Discusses various strategies for modernizing legacy systems, providing guidance for transformation efforts.
(22)Radhakrishnan et al. [29]	Introducing Coarrays in the Parallelization of Legacy Fortran Applications: A Case Study.	Presents a case study on parallelizing legacy Fortran applications using coarrays, addressing modernization challenges.
(23)Rodríguez-Echeverría et al. [30]	Modernize old e-commerce websites into rich internet applications.	Covers the conversion of traditional web apps to RIA — the way forward.
(24)Rodríguez, et al. [31]	Talk about restructuring old-world websites as rich internet apps.	Transitioning Legacy Web Apps Into Rich Internet Application.
(25)Satyanarayanan, et al. [32]	Investigate edge computing for heritage apps.	Talks on how edge computing can be applied to existing applications and overcomes modernization problems.
(26)Savio et al. [33]	Accelerate legacy applications with spatial computing devices.	Presents the use of spatial computing devices to accelerate legacy applications, focusing on modernization.
(27)Sharma et al. [34]	Discuss modernizing legacy systems using microservices and DevOps.	Explores the role of microservices and DevOps in modernizing legacy systems and provides insights.
(28)Smith & Brown [35]	Present a comprehensive framework for legacy application modernization.	Introduces a comprehensive framework for modernizing legacy applications, aiding in the process.
(29)Thompson & Patel [36]	Explore industry best practices in legacy application modernization through a case study.	Presents best practices in legacy application modernization based on a case study, offering insights.
(30)Turner & Davis [37]	Discuss challenges and	Explores challenges and potential

Study	Objective	Findings
	solutions in legacy application modernization.	solutions in modernizing legacy applications, addressing common issues.
(31) Ulrich & Newcomb [38]	Present a pragmatic approach to legacy modernization.	Introduces a pragmatic approach to modernizing legacy systems, focusing on practicality.
(32) Van Kranenburg & Romeijnders [39]	Explore challenges and strategies in modernizing legacy systems.	Discusses challenges and strategies for modernizing legacy systems, providing valuable insights.

Table 3. CASP

Study		Was the study valid? (Yes/No)	What are the results?	Are the results applicable to the local population?
(1)	AWS [8]	Yes	Industry insights and recommendations for modernizing legacy systems with microservices.	Applicability may vary based on local technology landscape.
(2)	Beebe [9]	Not applicable	N/A	Not applicable.
(3)	Brinkmann et al. [10]	Yes	Identified challenges and potential solutions in transitioning to microservices architecture.	Applicability depends on local adoption of microservices.
(4)	Carpenter [11]	Yes	Strategies for modernizing legacy systems and applications.	Applicable to local systems undergoing modernization.
(5)	Clements et al. [12]	Yes	Insights into modernization from software and business perspectives.	Applicability depends on local software and business context.
(6)	Debusmann et al. [13]	Yes	Using a model–driven approach in managing legacy applications.	Applicability depends on local use of model-driven techniques.
(7)	Ekanem & Woherem [14]	Yes	Proposed software maturity index for assessing legacy component stability.	Applicable to local assessment of legacy components.
(8)	Gartner, Inc. [15]	Yes	Insights and recommendations for modernization strategies.	Applicability depends on local business and technology context.
(9)	Gupta & Vaishnavi [16]	Yes	Strategies for transforming legacy systems.	Applicability varies based on local transformation needs.
(10)	Hamza et al. [17]	Yes	Framework for assessing modernization challenges.	Applicable to local assessment of modernization challenges.
(11)	Ilechko [18]	Yes	Guidance on designing, coding, and testing Java applications.	Applicable to local Java application modernization.
(12)	Infosys Ltd. [19]	Yes	Insights into modernization in the cloud era.	Applicability depends on local cloud adoption.
(13)	Jones [20]	Yes	Guidance on modernization in PHP.	Applicable to local PHP-based modernization efforts.

Study		Was the study valid? (Yes/No)	What are the results?	Are the results applicable to the local population?
(14)	Khan, [21]	Yes	Framework for enhancing legacy system security during modernization.	Applicable to local security-focused modernization.
(15)	Kiran-Mallidi's et al., [22]	Yes	TCOs and ROIs analyzing in legacy digital transformation.	Applicability depends on local financial considerations.
(16)	Koegel et al. [23]	Yes	Case study on challenges and opportunities in application modernization.	Applicable to local cases of application modernization.
(17)	Kumar & Garg [24]	Yes	Exploration of modernization with microservices and APIs.	Applicability depends on local adoption of microservices and APIs.
(18)	Mainetti et al. [25]	Yes	Model-driven transformation approach for enhancing user experience.	Applicability depends on local use of model-driven techniques.
(19)	Mayor et al. [26]	Yes	Discussion of connectionism in modernization challenges.	Applicability varies based on local adoption of connectionism.
(20)	Microsoft Corporation [27]	Yes	Insights into Microsoft's modernization approach.	Applicability depends on local use of Microsoft technologies.
(21)	Pradhan [28]	Yes	Discussion of various modernization strategies.	Applicability varies based on local modernization needs.
(22)	Radhakrishnan et al. [29]	Yes	Case study on parallelizing legacy Fortran applications.	Applicable to local cases involving legacy Fortran applications.
(23)	Rodríguez-Echeverría et al. [30]	Yes	To explore the migration of web apps from traditional stateless client-server model towards full client-side rendered single-page apps.	Applicability depends on local web application context.
(24)	Rodríguez, et al. [31]	Yes	Legacy web page vs. the rich internet app debate.	Applicability depends on local web application context.
(25)	Satyanarayanan, et al. [32]	Yes	Exploration of edge computing for legacy applications.	Applicability depends on local edge computing adoption.
(26)	Savio et al. [33]	Yes	Acceleration of legacy applications with spatial computing devices.	Applicability depends on local use of spatial computing devices.
(27)	Sharma et al. [34]	Yes	Discussion of modernization with microservices and DevOps.	Applicability depends on local adoption of microservices and DevOps.
(28)	Smith & Brown [35]	Yes	Presentation of a comprehensive framework for legacy application modernization.	Applicable to local efforts involving comprehensive modernization frameworks.

Study		Was the study valid? (Yes/No)	What are the results?	Are the results applicable to the local population?
(29)	Thompson & Patel [36]	Yes	Exploration of industry best practices in legacy application modernization through a case study.	Applicability varies based on local industry practices.
(30)	Turner & Davis [37]	Yes	Discussion of challenges and solutions in legacy application modernization.	Applicability depends on local challenges and solutions in modernization.
(31)	Ulrich & Newcomb [38]	Yes	Introduction of a pragmatic approach to legacy modernization.	Applicable to local pragmatic approaches to modernization.
(32)	Van Kranenburg & Romeijnders [39]	Yes	Exploration of challenges and strategies in modernizing legacy systems.	Applicability varies based on local modernization challenges and strategies.

Table 4. Theme

Theme	Studies Included
Modernization Strategies	Carpenter [11], Gupta & Vaishnavi [16], Ilechko [18], Jones [20], Microsoft Corporation [27], Pradhan [28], Sharma et al. [34], Smith & Brown [35], Turner & Davis [37], Ulrich & Newcomb [38]
Microservices and APIs	AWS [8], Brinkmann et al. [10], Kumar & Garg [24]
Legacy Component Assessment	Ekanem & Woherem [14]
Security Enhancement	Khan et al. [21]
Cost and ROI Analysis	Kiran Mallidi et al. [22]
Case Studies and Challenges	Beebe [9], Debusmann et al. [13], Koegel et al. [23], Radhakrishnan et al. [29], Rodríguez, Echeverria [30], Rodríguez-Echeverría et al. [31], Thompson & Patel [36]
Model-Driven Approaches	Debusmann et al. [13], Mainetti et al. [25]
Edge Computing for Legacy Applications	Satyanarayanan et al. [32]
Spatial Computing Devices	Savio et al. [33]

Peer-review history:
The peer review history for this paper can be accessed here:
https://www.sdiarticle5.com/review-history/108612