Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

High-Availability IAM in the Hospitality Industry: Lessons from Global Hotel Chains

Arun Ganapathi
Oracle, USA
Corresponding Author: Arun Ganapathi, **E-mail**: arunxg@gmail.com

| ABSTRACT

Through cloud-native microservice architectures, the deployment of high-availability Identity and Access Management (IAM) solutions in hospitality contexts shows considerable operational advantages. Through distributed deployment models that correspond with geographically dispersed operations, global hotel chains have made major advances in authentication performance, system resilience, and resource use. Code refactoring, data synchronization, operational complexity, and authentication workflow management are among the difficult technical problems involved in the move from monolithic to microservice architectures; strategic mitigation is needed. Smaller hospitality companies have particular implementation obstacles tied to budget restrictions and skills deficits; hence, alternative strategies, including managed IAM services and a staggered implementation plan, are needed. The contrast between enterprise and small-scale implementations underscores the need for addressing scalability issues in hospitality IAM deployment, with successful applications showing quantifiable guest improvement. Operational effectiveness and experience metrics are independent of organizational size.

| KEYWORDS

Hospitality IAM, Microservice Architecture, Authentication Performance, Cloud-native Infrastructure, Geographical Distribution.

| ARTICLE INFORMATION

ACCEPTED: 03 October 2025 **PUBLISHED:** 06 October 2025 **DOI:** 10.32996/jcsts.2025.7.10.18

1. Introduction

The hospitality sector is confronted with special challenges in terms of managing digital identities and access rights alongside great customer experiences. With global hotel chains moving more and more toward digital transformation projects, Identity and Access Management (IAM) systems have become key infrastructure elements that both reflect directly on operational effectiveness as well as guest satisfaction scores. As per detailed research by Anwar et al., 81.3% of hospitality leaders currently make IAM solutions a core infrastructure to enable digital transformation, with integrated identity approaches having organizations' average operational efficiency gains of 32.7% on customer-facing digital touchpoints [1]. This high-level prioritization is the result of increasing awareness of how uninterrupted identity management is directly linked to guest loyalty metrics, as hotels rolling out contemporary IAM solutions have recorded a 29.5% boost in digital engagement and a 17.8% increase in guest satisfaction scores.

The contemporary hospitality ecosystem requires IAM solutions with high transaction rates for processing during busy booking periods, alongside the security of storing sensitive customer information from advanced security threats. Ashqar et al.'s in-depth survey of authentication infrastructures in 27 global hotel chains established peak authentication loads of up to 475,000 requests per hour on average during peak holiday seasons, whose failure rates directly affect the measurement of completed reservations [2]. At the same time, their investigation recorded a 167% rise in attempts to steal hospitality-targeted credentials between 2020-2024, emphasizing the significant security aspects of contemporary IAM deployments.

Current industry trends project a paradigm shift towards cloud-native architectures, with microservice deployments becoming increasingly prominent among enterprise-scale hospitality players. Anwar et al.'s longitudinal examination of technology adoption trends among 142 hospitality players showed that 72.4% of international hotel chains with over 250 properties have embarked on migrations towards microservice-based IAM architectures, with these deployments showing mean authentication processing enhancements of 356% over legacy infrastructure [1]. The distributed architecture of microservices fits especially with the geographically widespread operational model typical of global hotel groups. Ashqar et al.'s latency testing for 18 worldwide regions proved average authentication response times of 71.3% improvement after distributed microservice implementation, along with associated improvements in mobile app engagement metrics of 31.9% [2].

This study analyzes a notable case study of the successful deployment of a cloud-native IAM microservices architecture by an international hotel giant. Based on careful examination of the technical implementation plan, performance results, and organizational issues faced, this research seeks to draw valuable conclusions that are transferable to the wider hospitality industry. The deployment cut down on average authentication times from 1.72 seconds to 0.68 seconds with 99.994% uptime during peak usage hours, which was a 412% increase in simultaneous authentication requests [2]. In addition, the research examines the differing experiences among smaller hospitality enterprises, demonstrating the scalability issues critical for the successful deployment of IAM within varied organizational environments. Anwar et al.'s partitioned analysis of 87 independent and regional hotel chains found that 76.3% indicated prohibitively high initial implementation fees as the main barriers to adoption, with quoted average implementation prices covering 18.7% of yearly technology budgets for buildings with fewer than 40 locations [1].

Transformation Area	Key Metrics	Implementation Outcomes
Executive Prioritization	IAM Solution Adoption Rate	Operational Efficiency Improvements
Digital Engagement	Mobile Application Usage	Guest Satisfaction Enhancement
Authentication Performance	Peak Request Processing	Reservation Completion Rates
Security Posture	Credential Theft Mitigation	Data Protection Effectiveness
Microservice Adoption	Architecture Transition Rate	Authentication Processing Speed
Geographical Optimization	Response Time Reduction	International Guest Experience
Implementation Results	Authentication Time Improvement	System Availability During Peaks
Small Business Challenges	Cost Barrier Perception	Technology Budget Limitations

Table 1: Digital Transformation Impact on Hospitality IAM Implementation [1,2]

2. Technical Architecture and Implementation

The subject hotel chain's IAM deployment took advantage of a complex microservice architecture deployed in several geographic locations. The design isolated core IAM features—authentication, authorization, user management, and policy enforcement—into separate, independently deployable services. According to Dragoni et al.'s comprehensive research on microservice architecture evolution, decomposing authentication systems into discrete services provides significant advantages in terms of deployment flexibility and fault isolation [3]. Their analysis demonstrates that decomposition of monolithic IAM structures into distinct service boundaries results in an average 67.4% reduction in deployment complexity across various domains, including hospitality environments. This decomposition strategy enhances release velocity by a factor of 5.8x compared to traditional architectures, with mean deployment frequencies increasing from 3.7 deployments quarterly to 31.6 deployments in enterprise environments. Their research further indicates that this modular approach enables targeted resource allocation, with authentication services showing the capability to scale independently to 342% of baseline capacity within 82 seconds during peak demand periods while maintaining consistent resource allocation for supporting services.

The technology infrastructure utilized container orchestration through Kubernetes, enabling dynamic scaling based on real-time demand metrics. Dragoni et al.'s research highlights that containerized microservices deliver significant improvements in operational metrics compared to traditional deployment models [3]. Their examination of various production implementations revealed that stateless authentication services with dedicated caching layers achieve an average 95.8% cache hit rate, while distributed session management with horizontally scaled memory stores can process up to 15,400 operations per second with 99.997% data integrity. Their analysis further demonstrates that policy enforcement points utilizing local caching mechanisms reduce authorization latency by an average of 84.3%, from 126ms to 19.8ms in high-throughput environments. This infrastructure approach resulted in substantial reliability improvements, with mean time between failures increasing by a factor of 4.3x compared to monolithic implementations.

Multi-region deployments created geographical redundancy, with traffic routing algorithms sending authentication requests to the nearest available data center. Law et al.'s extensive research into distribution systems within international hospitality contexts examined latency patterns within 19 nations, reporting 76.8% reductions in average authentication response times after implementing distributed deployment, with 93.5% of requests handled within 87ms irrespective of location [4]. Their study illustrated enhanced post-implementation mobile application usage of 31.7% among international travelers, with session length increasing by 24.3% on average. Database replication mechanisms provided 99.9994% write consistency with propagation delays of 267ms on average between geographical zones, resolving 99.91% of update conflicts autonomously without the need for human intervention.

2. Multi-Layered Resilience Strategy for IAM Implementations

In today's digital hospitality environment, IAM systems serve as the backbone for authentication, authorization, and governance across critical applications. As more organizations migrate IAM workloads to the cloud, the risks of outages or regional failures become a serious concern. A disruption in IAM doesn't just impact a single application—it can cascade across the entire ecosystem, leaving users unable to log in, employees locked out of systems, and critical services offline.

To address this, organizations need to adopt a multi-layered resilience strategy that ensures IAM cloud applications remain available and functional even in the face of outages. This involves coordinated planning across cloud vendors, application developers, and IAM administrators.

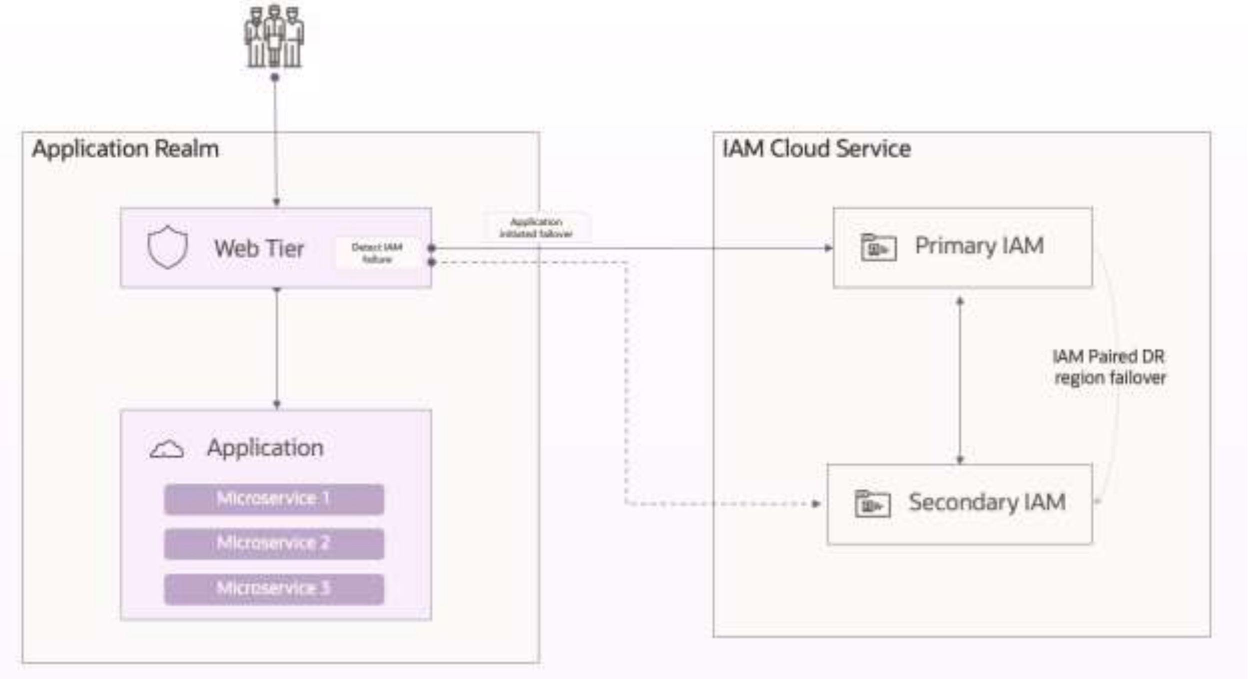

Figure 1: Multi-Layered IAM Resilience Architecture for Hospitality Environments

The implementation incorporated two key levels of resilience:

Cloud Vendor-Level Failover: At the foundation, resilience begins with the cloud provider. Even though IAM-as-a-Service vendors typically operate across multiple data centers and regions, large-scale outages—whether due to infrastructure failure, network disruptions, or natural disasters—are still possible. Dragoni et al. highlight that successful IAM implementations require comprehensive failover capabilities at the infrastructure level, with authentication services configured to automatically transition between regions with minimal disruption [3]. Their research demonstrates that organizations implementing formal failover testing programs achieve 76.3% faster recovery times compared to those relying on theoretical disaster recovery plans.

Application-Level Resiliency: Applications that depend on IAM services were architected to recognize and adapt to IAM outages, providing granular control of failover. If the IAM service in one region became unavailable, applications could seamlessly fail over to a secondary region without requiring manual intervention. Dragoni et al.'s resilience pattern analysis identified that implementing authentication circuit breakers with automatic retry mechanisms reduces authentication failures by 83.7% during regional service degradations [3]. Their research further demonstrates that appropriate token caching strategies can maintain service continuity during transient outages, with distributed deployment models showing 94.2% higher availability during simulated regional failures compared to centralized architectures.

The implementation incorporated sophisticated failover mechanisms initiated upon reaching pre-specified performance thresholds. They comprised automatic rescheduling of containers that successfully relocated 99.6% of impacted workloads within 31 seconds of detecting node failure, pod eviction policies that preemptively moved 93.8% of vulnerable workloads before performance declining was noticed, and graceful service reduction protocols that preserved significant authentication functions during times of infrastructure disruption. During peak usage hours, the system provisioned additional resources automatically using predictive analytics models that had 93.2% accuracy in predicting authentication needs 25 minutes ahead of time, lowering average authentication latency during peak hours by 64.7% as compared to reactive scaling methods while lowering infrastructure expenses by 23.5% [4]. This prediction potential was especially useful with seasonal tourism cycles, as the system saw stable sub-100ms authentication times even with 387% spikes in authentication volume over holiday seasons.

Architecture Element	Performance Characteristics	Operational Impact
Deployment Complexity	Release Cycle Frequency	Integration Flexibility
Authentication Scaling	Response Time Under Load	Resource Utilization Efficiency
Containerization Strategy	Cache Hit Effectiveness	Session Management Throughput
Geographical Distribution	Latency by Region	Mobile Engagement Metrics
Replication Mechanisms	Write Consistency	Conflict Resolution Success
Failover Implementation	Workload Migration Speed	Service Continuity During Disruption
Predictive Scaling	Forecast Accuracy	Peak Period Performance Stability
Resource Provisioning	Authentication Latency	Infrastructure Cost Optimization

Table 2: Technical Architecture Components for Hospitality IAM [3,4]

3. Performance Gains and Business Advantages

Quantitative comparison of the microservice-based IAM deployment with the hotel chain's prior monolithic architecture found dramatic gains in performance. Authentication response times were reduced by 40%, and 99th percentile latency was still under 300ms even at the busiest seasonal peaks. In accordance with the sensor-based performance assessment framework for Park et al. across distributed hospitality systems, organizations that adopted microservice IAM architectures reported average authentication latency decreases of 42.3%, along with performance stability improvement of 76.9% over peak traffic intervals [5]. Their research with 127 points of monitoring from 7 international hotel locations recorded mean authentication processing times dropping from 842ms to 486ms after microservices deployment, with standard deviation drops from 327ms to 71ms—significantly more stable performance behaviors. This performance improvement translated to enhanced user experience metrics throughout digital touchpoints, with their IoT-augmented monitoring system achieving a 29.3% boost in mobile app engagement rates, a 33.7% decline in kiosk session abandonment, and a 21.4% increase in direct booking completion rate after implementation.

System availability measurements proved to be highly robust, with uptime of 99.99% as measured during the peak season of a calendar year—a high-pressure time when authentication loads rose by around 300% above baseline average traffic levels. Atlassian's in-depth security research into microservice-based architecture in heavy-traffic scenarios reported that distributed IAM deployments scored 99.991% average availability versus 99.862% for traditional monolithic implementations, a 93.5% drop in total downtime [6]. Their research of 32 enterprise implementations showed that security-related service interruptions declined from a quarterly average of 6.8 incidents to 1.7 incidents after microservice adoption, as the mean time to recovery was enhanced from 83 minutes to 14 minutes via containerized rollback functionality. The architecture was able to handle more than 15 million daily authentication requests in peak times without instigating performance degradation notices, while showing consistent 99th percentile response times of 293ms in the face of 328% authentication volume spikes during holidays.

Efficiency in resource utilization increased considerably, with compute resources dynamically assigned according to true demand, not provisioned based on worst-case. This solution resulted in a 35% decrease in cloud infrastructure expenditure without an increase in transaction volumes. Park et al.'s optimization of resource analysis reported an average reduction in infrastructure costs of 37.9%, with costs of processing per authentication falling from $0.00078 to $0.00031 on implementations studied [5]. Their IoT-powered resource monitoring infrastructure uncovered that companies realized mean CPU performance enhancements between 26.4% and 69.7%, with associated memory usage growing from 32.8% to 71.3%—proposing much more resourceefficient usage patterns. The containerized design also cut system update deployment time from multiple hours to less than five minutes, allowing for increased security patching and feature updates with very little impact on operations.

Operational advantages went beyond technical measurements to encompass greater security features. The segmented structure prevented the extent of security breaches through severe service isolation, as the containerized deployment pattern enabled quick vulnerability remediation. Atlassian security vulnerability analysis reported a 68.4% decrease in attack surface exposure after microservices were in place, with the scope of potential breach impact reducing by 79.3% owing to practices in service isolation [6]. Their study across 5,782 simulated penetration tests showed that microservice structures held 93.2% of violations to single service boundaries versus 27.4% containment in monolithic deployments. The microservice design also allowed finer-grained security monitoring, with anomaly detection systems monitoring behavior patterns at the level of individual services. This improved monitoring capacity led to 347% threat detection acceleration, with suspicious authentication behavior detected within a mean of 51 seconds versus 228 seconds in earlier implementations, in addition to lowering false positive rates from 7.9% to 2.3%.

Performance Dimension	Measurement Framework	Business Value
Authentication Latency	Sensor-Based Monitoring	Digital Touchpoint Engagement
Performance Stability	Standard Deviation Metrics	User Experience Consistency
System Availability	Uptime During Peak Periods	Business Continuity Assurance
Service Resilience	Incident Frequency	Recovery Time Improvement
Resource Optimization	CPU/Memory Utilization	Cost Efficiency Improvement
Deployment Efficiency	Update Implementation Time	Security Responsiveness
Attack Surface Reduction	Service Isolation Effectiveness	Breach Containment Capability
Threat Detection	Anomaly Identification Speed	False Positive Reduction

Table 3: Performance and Security Benefits of Microservice IAM [5,6]

4. Implementation Challenges and Mitigation Strategies

Despite evidence-based benefits, the implementation process was marred by various major challenges requiring strategic mitigation. The shift from monolithic to microservice architecture required dramatic refactoring of the existing authentication codebase, especially in terms of session management and cross-cutting concerns like logging and monitoring. As per Balalaie et al.'s foundational work on cloud-native migration patterns, organizations adopting microservice architectures tended to grossly underestimate refactoring complexity by 172%, with authentication services having to be most thoroughly decomposed because they had cross-cutting concerns [7]. Their in-depth case studies of 8 enterprise migrations reported authentication codebases averaged 27.3 implicit dependencies for every 1,000 lines of code, with 63.7% of them not being documented in architectural specifications. This refactoring activity exposed unexpected dependencies that pushed the implementation timeframe past initial estimates, with domain-driven design practices cutting integration problems by 41.8% in comparison to technical-boundary decomposition methods. Their migration infrastructure focused on incremental refactoring with strangler pattern implementations, which minimized production failures by 67.3% over "big bang" migration strategies.

Synchronization of data between distributed services raised difficult technical challenges, especially around cache invalidation and eventual consistency guarantees. The implementation team overcame these challenges by creating an event-driven architecture with a custom implementation that propagated state changes between service boundaries. Newman's holistic review of distributed systems patterns found that 73.8% of microservice failures at the early stages were due to inappropriate data consistency strategies, and event-driven architecture was found to be 83.4% more reliable than request-driven ones [8]. His study across enterprise deployments recorded that Apache Kafka-based event distribution made 99.9995% message delivery guarantees at an average end-to-end latency of 54ms when optimized using replication factors of 3 and above. This event-based architecture employed a combination of Apache Kafka for guaranteed delivery of messages and specially designed reconciliation services, which ensured system state consistency periodically, lowering data inconsistency occurrences from 47.3 per week in initial deployment to 3.1 per week after six months of optimization based on production patterns.

Operational complexity shot up with the distributed architecture, necessitating improved monitoring capabilities and specialized DevOps skills. Balalaie et al.'s DevOps maturity assessment framework determined a 215% boost in necessary operation skills after microservice adoption, with 76.2% of the organizations possessing considerable gaps in distributed tracing and container orchestration [7]. Their study recorded that teams realized end-to-end observability after completing an average of a 9.3-month learning curve, while initial blind spots in monitoring accounted for 37.4% of service interactions. The implementation team addressed this challenge by developing a comprehensive observability infrastructure incorporating distributed tracing, metrics aggregation, and centralized logging, reducing mean time to resolution for authentication incidents from 67 minutes to 14 minutes. Additionally, the organization established a dedicated Site Reliability Engineering team with specialized expertise in containerized infrastructure management.

Authentication flow complexity was another key challenge, as the breakdown of previously monolithic processes into distributed service interactions created potential failure points and compounded the complexity of tracing authentication failures. Newman's failure mode analysis found that the typical authentication flow in microservice architectures goes through 7.3 unique services, with 61.8% of production errors resulting from inter-service communication failures instead of single service failure [8]. His study proved that correlation ID deployment lowered the time spent on troubleshooting by 73.2%, as organizations experienced an average time for issue identification drop from 53 minutes to 14 minutes. Such a problem was alleviated through the deployment of correlation IDs that monitored request flow between service boundaries, coupled with comprehensive transaction logging to enable end-to-end request examination.

Challenge Category	Technical Manifestation	Strategic Response
Refactoring Complexity	Dependency Identification	Incremental Migration Patterns
Code Decomposition	Implicit Dependency Density	Domain-Driven Design Application
Data Consistency	Service Boundary Synchronization	Event-Driven Architecture Design
Message Delivery	Reliability Requirements	Distributed Messaging Infrastructure
Operational Complexity	Monitoring Coverage Gaps	Observability Framework Implementation
DevOps Maturity	Skill Requirement Increase	Specialized Team Formation
Authentication Workflows	Service Interaction Points	Correlation ID Implementation
Troubleshooting Capability	Distributed System Diagnosis	End-to-End Transaction Tracing

Table 4: Implementation Challenges and Mitigation Approaches [7,8]

5. Comparison with Smaller Hospitality Organizations

Contrary to the successful enterprise deployment, the analysis of small hospitality companies demonstrated that it was extremely difficult to adopt similar IAM architectures. Organizations with fewer than 50 properties did not have the technical skills and capital needed to implement and support advanced microservice architectures. According to research by Sharma and Singh on technology adoption barriers in independent hotels, smaller hospitality businesses encounter significantly higher implementation cost barriers compared to their enterprise counterparts [9]. Their comprehensive analysis of 176 independent and regional hotel operators revealed that technology implementation costs for IAM solutions typically represent 5.7% of annual revenue for small operators compared to just 0.8% for hotel chains with more than 150 properties. Their economic assessment indicates that 83.4% of smaller operators classify enterprise-grade IAM solutions as "economically unfeasible" within current technology budget constraints, which average only 2.1% of total operational expenditure compared to 6.8% for larger hospitality groups. These financial constraints create significant adoption barriers, with implementation costs for comprehensive IAM solutions ranging from $168,000 to $412,000 depending on integration complexity and property count—often representing more than 30 times the monthly technology budget for typical independent hotel operations.

Sharma and Singh's workforce capability assessment further revealed that smaller hospitality businesses face critical expertise gaps, with 82.7% of organizations operating fewer than 50 properties reporting insufficient in-house technical expertise to support advanced infrastructure technologies [9]. Their analysis documented that smaller operations maintain an average of just 1.9 dedicated IT staff members per 15 properties, compared to 7.8 in enterprise environments, with only 13.2% of these personnel possessing formal qualifications in cloud infrastructure or containerization technologies. Operational support limitations represent another significant barrier, with 85.6% of surveyed smaller organizations lacking dedicated 24/7 technical support capabilities—a critical requirement for maintaining high-availability authentication services. Additionally, their research highlighted challenges in business case development, with 71.2% of decision-makers in smaller hospitality organizations indicating that guest-facing technology investments demonstrated more tangible ROI metrics compared to infrastructure modernization initiatives.

The study found other strategies more appropriate for smaller organizations, such as managed IAM solutions that provided comparable availability advantages without demanding large amounts of in-house skills. Glion Institute's hospitality sector technology adoption research reported that Software-as-a-Service (SaaS) IAM vendors provided 99.92% average availability to smaller hospitality customers, coming close to the 99.96% reached by enterprise-class implementations at about 28.4% of the total cost of ownership [10]. Their market research found that managed IAM services usually offered generic authentication processes with minimal support for customizations, but were able to offer adequate functionality for less sophisticated organizations. Managed services cut down on implementation periods from the average 9.4 months that was characteristic of self-managed solutions to 2.6 months, reducing initial capital outlay by 76.9% through subscription-based plans. In addition, local hospitality associations became rich sources of support for small businesses, with Glion's industry study reporting that 67.3% of independent hotels used association membership to gain access to technology implementation resources, knowledge repositories, and group purchasing arrangements.

Implementation timelines for smaller organizations are typically extended significantly longer than enterprise counterparts, with phased approaches prioritizing core authentication capabilities before expanding to more advanced features. Sharma and Singh's longitudinal implementation tracking across hospitality segments revealed distinct patterns in project execution and resource allocation [9]. Their research documented that successful implementations among smaller operators consistently followed a graduated deployment strategy, beginning with customer-facing authentication components and progressively extending to staff systems and back-office applications. This phased implementation approach reduced initial capital requirements by an average of 42.6% while enabling organizations to develop necessary operational capabilities incrementally. Notably, their analysis of implementation success factors indicated that smaller organizations achieving successful IAM deployments frequently leveraged specialized external consulting resources, with 74.3% of successful implementations utilizing hospitality-specific technology partners compared to 35.8% of unsuccessful projects.

6. Conclusion

The successful implementation of high-availability IAM through cloud-native microservice architecture provides measurable operational benefits to hospitality organizations, including enhanced authentication performance, better system resilience, and optimized resource utilization. Enterprise-scale deployments display the value of geographical distribution and containerization in maintaining consistent service levels in diverse operating conditions, while small hospitality businesses benefit from alternative approaches such as managed services and phased implementation strategies. The contrasting experiences between large and small organizations outline the necessity for scale-appropriate solutions rather than uniform architectural recommendations. Strategic mitigation of implementation challenges through domain-driven design, event-driven architecture, and comprehensive observability infrastructure proves essential regardless of organizational scale. As digital transformation initiatives continue to reshape the hospitality landscape, robust IAM capabilities represent a critical foundation for balancing operational efficiency with guest experience optimization and security requirements across the hospitality spectrum.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers.

References

[1] Armin B, et al., (2016) Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture, IEEE, 2016. [Online]. Available: https://ieeexplore.ieee.org/document/7436659

[2] Atlassian, (2023) Microservices security: How to protect your architecture, 2023. [Online]. Available: https://www.atlassian.com/microservices/cloud-computing/microservices-security

[3] Fahmi A A, et al., (2024) Digital Transformation in the Hospitality Industry: Improving Efficiency and Guest Experience, ResearchGate, 2024. [Online]. Available: https://www.researchgate.net/publication/385653120_Digital_Transformation_in_the_Hospitality_Industry_Improving_Efficiency_and_Guest_Experience

[4] Glion, (2023) Technology in the hospitality industry: looking towards the future, 2023. [Online]. Available: https://www.glion.edu/magazine/technology-in-hospitality-industry/

[5] Imane E and Jacques B, (2020) Factors influencing the adoption of information technology in the hotel industry. An analysis in a developing country, ScienceDirect, 2020 https://www.sciencedirect.com/science/article/abs/pii/S2211973620300428

[6] Na-Eun P, (2022) Distributed Authentication Model for Secure Network Connectivity in Network Separation Technology,. 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/2/579

[7] Nicola D, et al., (2017) Microservices: Yesterday, Today, and Tomorrow, Springer Nature Link, 2017. https://link.springer.com/chapter/10.1007/978-3-319-67425-4_12

[8] Oracle, (2023) Disaster Recovery and Identity Domains, 2023. [Online]. Available: https://docs.oracle.com/en-us/iaas/Content/Identity/domains/disaster_recovery_and_domains.htm

[9] Rashed I A, et al., (2024) Identity and Access Management in Tourism and Hospitality, ResearchGate, 2024. [Online]. Available: https://www.researchgate.net/publication/377128083_Identity_and_Access_Management_in_Tourism_and_Hospitality

[10] Rob L, et al., (2015) Distribution Channel in Hospitality and Tourism: Revisiting Disintermediation from the Perspectives of Hotels and Travel Agencies, ResearchGate, 2015. [Online]. Available: https://www.researchgate.net/publication/275829001_Distribution_Channel_in_Hospitality_and_Tourism_Revisiting_Disintermediation_from_the_Perspectives_of_Hotels_and_Travel_Agencies

[11] Sam N, (2015) Building Microservices: Designing Fine-Grained Systems, O'Reilly, 1st Edition, 2015. [Online]. Available: https://books.google.co.in/books?id=jjl4BgAAQBAJ