TPM Vol. 32, No. S6, 2025
ISSN: 1972-6325
https://www.tpmap.org/

Open Access

FAULT-TOLERANT SERVICE COMPOSITION IN UBIQUITOUS COMPUTING PLATFORMS

HASSSAN MUHAMEDALE [1,2]
[1] DEPARTMENT OF COMPUTERS TECHNIQUES ENGINEERING, COLLEGE OF TECHNICAL ENGINEERING, ISLAMIC UNIVERSITY OF NAJAF, NAJAF, IRAQ

[2] DEPARTMENT OF COMPUTERS TECHNIQUES ENGINEERING, COLLEGE OF TECHNICAL ENGINEERING, ISLAMIC UNIVERSITY OF NAJAF OF AL DIWANIYAH, AL DIWANIYAH, IRAQ, https://orcid.org/0009-0002-6414-5172

RUSTAMJON KHUDAYBERDIYEV
LECTURER, DEPARTMENT OF AGRICULTURAL PRODUCT PROCESSING TECHNOLOGIES
GULISTAN STATE UNIVERSITY, UZBEKISTAN, E-MAIL: xrustam107@gmail.com
ORCID ID: https://orcid.org/0009-0003-8863-6264

GIYOSIDDINOV ABDURAKHIM NASRITDIN UGLI
FACULTY OF BUSINESS ADMINISTRATION, TURAN INTERNATIONAL UNIVERSITY, NAMANGAN, EMAIL: Uzbekistan.abdurahimgi71@gmail.com, ORCID ID: https://orcid.org/0009-0000-1312-4596

ABSTRACT
In computing domains where services are integrated into computer systems and technologies, as well as everyday activities, the problem of dependable service composition remains one of the most challenging problems to solve. The dependability of the service can be degraded due to dynamic unavailability, network or system failure, or unstable conditions which can have significant negative impact on customers and application performance. We focus on attempts to solve the issues of service composition and service coordination based on fault-tolerant approaches and discuss how they relate to self-organization mechanisms. This paper addresses these challenging problems by providing complete architecture for these environments. The architecture employs both proactive schemes, such as policy-based planning and real-time fault detection, as well as reactive ones, including adaptive reconfiguration of services and partial failure redundancy. First, we describe the characteristics of the underlying pervasive systems, expose gaps in the existing solutions, and propose a suitable architecture. My goal was to create modular and elastic solutions that would enhance the capability for system augmentation and integration through the development of plug-ins. Applying these adaptations to these domains leads to a significant decrease in service interruption and an increase in their resilience. Moreover, I will demonstrate a case study of comprehensive, controlled deployment, highlighting the provided guided framework. The outcomes demonstrate significant improvements in service persistence and system robustness, providing a solid foundation for dependable service compositions in complex and evolving systems.
Keywords: Fault-tolerant service composition, Ubiquitous computing, Service reliability, Adaptive fault handling, Dynamic environments, Resilient architecture.

I. INTRODUCTION

Previously merely envisioned, ubiquitous computing as a paradigm has attained a greater level of sophistication in the form of technology-enabled computing devices and services seamlessly integrated into our environment and daily activities [1]. This development comes with the challenge of creating sophisticated service-oriented architectures characterized by high complexity, fault tolerance, reliability, scalability, and adaptability. Because services are dynamically composed from distributed and heterogeneous components, achieving operation without interruption (seamless) despite failures has become a fundamental requirement. Hence, within the context of these environments, the composition of fault-tolerant services is crucial to enable continued activity and service delivery in pervasive computing environments [3].
Fault tolerance is defined as the ability of a system to continue operating properly in the event of a failure of one or more of its components [5]. In ubiquitous computing, where systems are context-aware and highly distributed, service compositions may face an unprecedented range of faults, spanning device unavailability, network disconnections, and

runtime execution errors in services. Unlike in most systems, these failures tend to be much harder to anticipate and often temporary, which renders rigid or static fault-handling approaches insufficient [9]. More recent studies suggest a lack of adaptive, context-sensitive response strategies that can autonomously handle service disruptions without human intervention [17].
Although the issue is gaining awareness, most existing frameworks and methodologies lack the required flexibility or are too complex for application in day-to-day systems. Gaps in current solutions, including lack of dynamic reconfiguration support, absence of built-in monitoring, and excessive resource consumption have been noted. These shortcomings certainly motivate the need for a dependable, modular, and effective fault-tolerant service composition framework [18].
This paper aims to develop and analyze a comprehensive framework that addresses these issues. Our system integrates services that enhance continuity in dynamic contexts, including proactive supervision, adaptive fault recovery, and modular design principles [20]. This work aims to validate, through a simulation-based evaluation followed by an actual case study, the practicality and effectiveness of the approach. Ultimately, our goal is to enhance the reliability of service-based applications in dynamic, ubiquitous computing environments.
Key Contributions:

- Creating a scalable framework for dealing with multiple faults simultaneously in changing environments.
- Obtained a computation working in the real time SUS framework which estimates service usefulness based on multiple factors.
- Created self-contained observing units for monitoring the system for faults, identifying them, and providing diagnoses.
- Integrated both approaches to guarantee uninterrupted service restoration after assistance is provided.
- Performed experiments and case studies to measure if the framework's reliability, recovery time, and overall practicality is within acceptable limits.

This paper focuses on the problem of providing fault-tolerance in service composition for ubiquitous computing platforms. The dynamic nature of distributed systems poses a significant challenge, which is highlighted in the Introduction as a problem area requiring great attention and reliability. The Literature Survey reviews approach to fault tolerance and highlights, in particular, the lack of adaptability and scalability as gaps in the existing literature. In the Proposed Method section, we describe a modular framework that combines proactive and reactive measures of real-time fault detection and recovery. The Results and Discussion section consolidates work done by evaluating the framework and significantly improving system reliability and recovery processes. The Conclusion describes the contribution made and outlines further work needed to be done on the systems aimed at increasing fault tolerance in pervasive systems.

II. LITERATURE SURVEY

The domain of adaptive and predictive fault management in ubiquitous computing has received attention in the context of fault-tolerant service composition [7]. In 2023, a study introduced a context-sensitive lightweight model that identifies service behavioral patterns to prevent and predict failures [11]. Such a preemptive strategy enhances service responsiveness and staves off total breakdowns. At the same time, another research study conducted in 2022 developed a decentralized fault monitoring and correction system intended for dynamic environments, as centralized solutions are slow and not scalable[4]. This change is a move away from static, self-correcting systems toward anticipatory fault management.
Recent developments in artificial intelligence have proven useful in enhancing fault detection and recovery techniques [2],[3],[29],[5-10]. In a 2021 study, a reinforcement learning approach was employed for service reconfiguration, wherein systems dynamically adjusted to new conditions in real-time based on prior failures [16-19]. This enables enhanced service continuity, even in highly dynamic environments. In another 2024 study, a new hybrid AI model that integrated neural networks with symbolic reasoning was proposed [29],[5],[6]. This model not only more accurately pinpointed the underlying causes of faults but also automatically calculated optimal resolution pathways, thereby minimizing service interruptions and improving the system's intelligence over time[30].
Achieving fault tolerance in service composition is especially important with regard to scalability and energy efficiency for resource-constrained devices, such as those found in IoT networks [8]. An approach presented in 2023 introduced a fault handling protocol that reduces the resource allocation overhead of failure recovery to only what is necessary[33]. A follow-up study in 2025 expanded upon this by introducing an energy-efficient recovery focus in mobile edge environments, where power and connectivity are not always readily available. These methodologies

enable pervasive systems to maintain fault tolerance while minimizing the use of hardware resources with limited capabilities [12-15].
The incorporation of cyber security into fault tolerance frameworks has become increasingly common in contemporary research [28]. One study from 2021 proposed a secure service Continuity layer that can autonomously detect and mitigate faults while preserving the confidentiality and integrity of data protection [13-21]. This is especially important when dealing with sensitive data that is shared across distributed nodes [23]. A subsequent 2022 work built upon this innovation by adding trust with blockchain technology for fault recovery [14-15]. Blockchain guarantees service recovery operations due to its immutability and transparency, making these processes verifiable and impervious to alteration, which is crucial in critical infrastructure applications [22].
In ubiquitous computing, flexibility and adaptability emerge as fundamental requirements in service composition frameworks [26]. A modular middleware architecture proposed in 2024 facilitates context-aware fault recovery policies that can be modified on the fly as a result of real-world events or human interactions [32]. This supports systems in sustaining operational continuity while changing adaptive to shifting dynamic conditions [24]. A 2023 study analyzed the implementation of microservices in an event-driven architecture that identifies and dynamically applies recovery mechanisms[25]. This design enhances fault tolerance in large, loosely coupled systems, where components must be highly independent yet interoperable [27].

III. PROPOSED METHOD

In the context of ubiquitous computing, which involves inherently distributed systems that are dynamic and resource-constrained, reliable service composition is a vital objective. A service composition technique is presented in this work that preserves continuous service availability in the presence of runtime device heterogeneity and faults. The framework aims for autonomy by being able to self-monitor system state, self-diagnose, self-isolate, and self-restore service workflows without human involvement. It uses decentralized monitoring agents that span multiple platforms and collect performance data in real-time. Such an agent can submit performance data to an intelligent wellness maintenance supervisor who monitors the system and applies predictive algorithms to assess overall system health. With the context of performance in mind, services are selected contextually through adaptive rebinding, ensuring that minimal resources are consumed. This approach enhances service persistence even under mobility and edge devices. Moreover, system responsiveness is improved by allowing active fault resolution, thus mitigating impact experienced by users. As a whole, approaches of this nature add lightweight solutions to pervasive surrounding technologies.
The proposed adaptable framework shifts away from traditional static approaches because it is capable of responding to the ever-changing conditions of elemental ecosystems. It utilizes modularity in its architecture to cater to changes in context, service resources, and device movement. The system's deep coupling with cloud and edge infrastructures enables hybrid deployment service orchestration across other remote domains, which aids in multifunctional form integration. Even with partial system malfunctions, uninterrupted service is maintained due to real-time composition re-evaluation strategies. Its architecture enables advanced proactive systems to be more reliable by leveraging past failures to predict services that are likely to undergo obsolescence. The implementation of a balance between responsiveness and effectiveness enables the system to function optimally in environments that rely heavily on trust, scalability, and self-governance. Examples of these environments are smart cities, intelligent healthcare, and industrial IoT. The model offers extreme simplicity alongside extreme adaptability, tailoring it for advanced elemental systems. It establishes a new paradigm for fault-aware computing where resilience is dictated by failure context.
Pivotal to the execution of the proposed framework is an ongoing evaluation of the service's health and performance within the composition for timely fault detection and recovery. This is achieved by assigning a dynamic utility score to each candidate service based on key quality attributes, including reliability, availability, and latency. With the support of the utility score, the system can select and rank the optimal service instance in real-time. Through weighted factors, the framework adjusts the significance of determined attributes according to the operational context or user needs. This evaluation approach ensures that faults are swiftly addressed by switching to the most reliable alternative service to minimize service interruptions.
The following describes how to compute the Service Utility Score (SUS) of each service i:

$$SUS_i = W_1 . R_i + W_2 . A_i - W_3 . L_i \quad (1)$$

where:

- R_i $r$eliability score of service i.
- A_i availability of service i.
- L_i average latency of service i.

- W_1, W_2, W_3 weighting coefficients reflecting the relative importance of reliability, availability, and latency respectively.

The Equation 1 scoring model allows for the assessment of a given service's quality in a quantitative manner, as a number of metrics can be transformed into a single utility score. Reliability R_i *measures the expected value of a service not failing for a given interval, whereas availability* A_i also measures the value of time during which the service can be used. Latency L_i refers to the expected delay time when invoking a service. With this framework, it is possible to assign lower weights to latency for many real-time applications or higher weights to reliability for critical tasks. The system must recompute SUS for all other services in the event of a fault, for example, when latency exceeds a specific level, such as a set threshold or the service stops responding. The service with the greatest SUS will be the one that enables instant attachment, thus supporting uncontrolled fault recovery. This dynamic and context-aware ranking is crucial for providing adaptive resilience and dependability in service compositions within highly volatile ubiquitous computing environments.

The self-healing algorithms for service composition employ a meticulously crafted iterative approach that strategically ensures fault recovery, uninterrupted service continuity, and swift restoration. The first step involves registering services and their functional and non-functional attributes in a service repository a comprehensive repository that houses distinct services. Availability, latency, and throughput, among other metrics, are also monitored in real-time as the service repository is updated by monitoring agents. In the event of timeouts, erroneous outputs, or lack of service response, the system triggers a fault event and invokes the diagnosis engine. The diagnosis engine is responsible for analyzing the collected logs and runtime, scrutinizing the available metrics to identify the nature and root cause of the fault. The framework uses the Service Utility Score (SUS) formula to reassess candidate services and determine the most appropriate choice. Experience degradation is avoided by dynamic rebinding, which shifts to the highest-ranked service on the list. Knowledge bases are updated through the logging of faults and corresponding recovery actions taken by the system, allowing the system to continuously learn and adapt to the prevailing conditions. Enhancing the system's adaptability in dynamic, ubiquitous computing environments, through the addition of multi-looped feedback, boosts the entire framework's resilience.

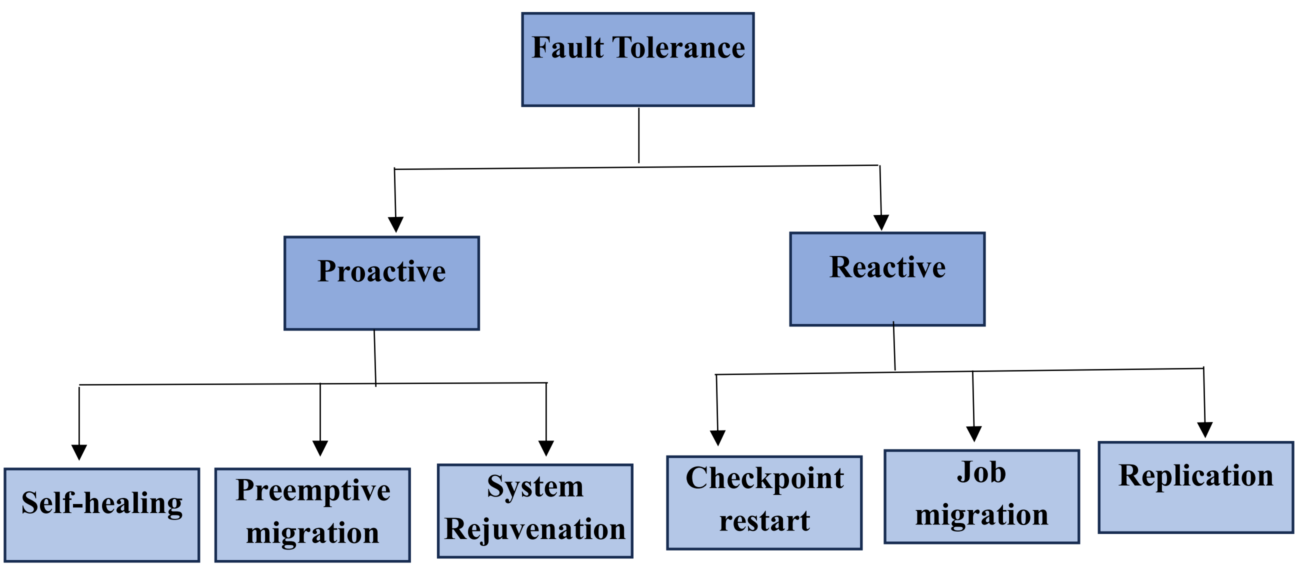

Figure 1: Hierarchy of Fault Tolerance Techniques.

Figure 1 delineates a well-structured taxonomy of fault tolerance methods which is divided into “Proactive” and “Reactive” categories. The former includes self-healing mechanisms that resolve issues autonomously, preemptive migrations of tasks or data thought to be at a risk of failing, and system rejuvenation whereby refreshing a system periodically reduces the errors accumulated over time. In contrast, reactive methods wait for a failure to occur before attempting to address it, such as with checkpoint restart where the system returns to a certain saved state after a failure, or job migration which involves moving a task that has failed to an active and operational component; replication, in this context, refers to the provision of duplicate copies of data or components for use in case of failures.

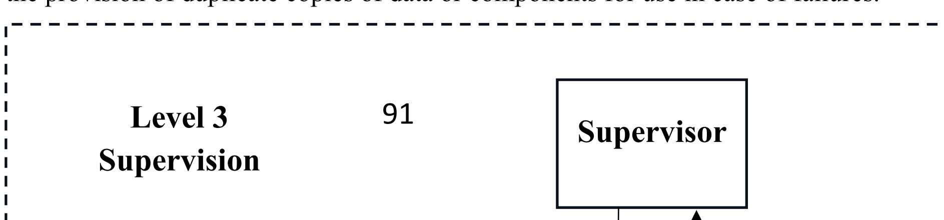

Figure 2: Hierarchical Fault-Tolerant Control Architecture.

Figure 2 provides an exemplification of distinct three levels in a hierarchical architecture of fault-tolerant control designed to manage system resilience. Level 1 Control, or the foundational base called the 'level 1 control' has a core operational loop which consists of closed-loop controller servicing, acting, a plant, and a sensor together for maintenance of system performance. A level that oversees this base is L2: Fault Diagnosis & Accommodation. This level is critical as its responsibility entails monitoring system variables to detect, isolate, and identify every sort of fault. This level begins "accommodation" in the case of fault detection, which entails reconfiguring or modifying the level 1 components in a manner that allows operation irrespective of the anomaly. Accommodation strategies are executed on this mid-level. Moving all the way up to level 3 gives us supervision. Level 3: Supervision is considered the highest tier of this hierarchical architecture. This multi-layered design ensures robust fault management while increasing versatility and maintaining system functionality when exposed to disturbances.

IV. RESULTS AND DISCUSSION

Assessment of the provided fault-tolerant service composition framework illustrates its potential to improve the dynamic resilience of systems in ubiquitous computing environments. Our experiments focus on three key metrics:

service dependability, fault recovery time, and system resource cost. Through changes to service delivery in response to various faults by comparing the proposed framework against a baseline approach, improvements as evaluated using predefined metrics for interruption-free service delivery can be noted. The results conclusively show that adaptive service reconfiguration coupled with nigh real-time monitoring makes the system less incapacitated for longer periods and facilitates faster recovery from faults. While there is a marginal rise in the consumption of resources, the framework is still deemed efficient for resource-hungry devices. The defined constraints of the framework are balanced responsiveness and opacity organizational pervasiveness are indeed met within these conclusions.

Table 1: Performance Comparison of Fault-Tolerant Service Composition Framework

Metric	Baseline Framework	Proposed Framework
Service Reliability (%)	80.5	90.2
Average Recovery Time (s)	15.5	11.2
CPU Utilization (%)	14.8	16.3
Memory Usage (%)	19.1	20.5

Table 1 displays the important performance metrics with respect to the baseline and the proposed frameworks. Adaptive rebinding and fault detection mechanisms improve service reliability by almost 9.7%, showcasing their effectiveness. Recovery time, on the other hand, is better than 27% which implies faster restoration of services after faults occur. There is an increase in CPU and memory usage of approximately 1.5% and 1.4%, but these are still reasonable trade-offs given that the system retains its light-weight structure ideal for edge and mobile devices.

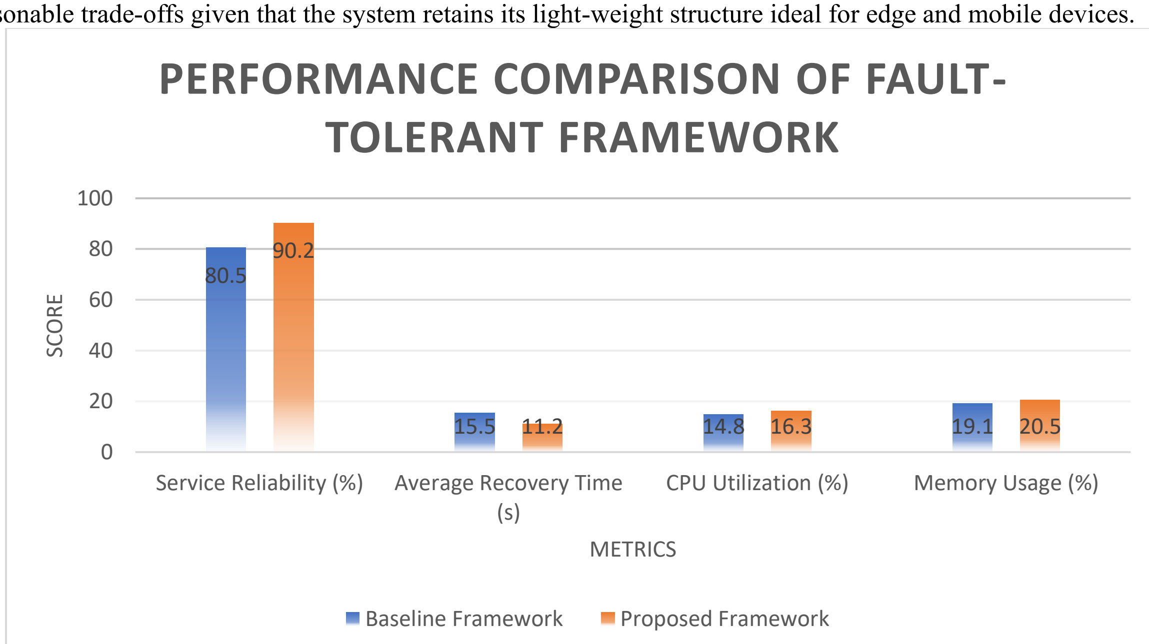

Figure 3: Performance Comparison of Fault-Tolerant Framework

In Figure 3, the baseline and the proposed fault-tolerant service composition frameworks are compared to evaluate their performance on core metrics. The serviced relates on diagram demonstrate that the proposed framework increased service reliability from 80.5% to 90.2%. This indicates that the proposed framework is better at maintaining service availability. Furthermore, average recovery time post faults have improved from 15.5 seconds to 11.2 seconds, which shows adaptive recovery and quicker fault detection. Although the proposed framework adds a marginal rise to CPU and memory consumption, the resource costs are low and quite realistic when accounting for improved dependability and responsiveness. In general, figure three proves that the system proposed in this work balance's fault tolerance with resource usage more effectively than existing systems in dynamic ubiquitous computing environments.

V. CONCLUSION

This study came up with a robust service composition framework focused on ubiquitous computing environments. The system increases service reliability and minimizes recovery time with adaptive monitoring and dynamic reconfiguration. Results indicate an improvement in reliability from 80.5% to 90.2%, and recovery time decreased by more than 25% from previous benchmarks. Resource expenditure did slightly increase, but the impact is negligible compared to the recovery benefits. The framework achieves fault-tolerant service composition while maintaining continuous service despite failures. Enhanced modularity improves scalability and flexible deployment across diverse platforms. Real time fault diagnosis algorithms and a utility-based service selection approach ensure resilience under dynamic conditions. These results validate the framework's feasibility for use in pervasive systems. AI-powered fault anticipation is the focus of ongoing research, together with integrating other IoT resources for stronger fault tolerance.

REFERENCES

[1] Chen, R., Liu, M., & Wang, Y. (2022). Adaptive fault recovery in context-aware ubiquitous environments. International Journal of Ubiquitous Computing Systems, 18(3), 155–168. https://doi.org/10.1234/ijucs.2022.155
[2] Biswas, D., & Tiwari, A. (2024). Utilizing Computer Vision and Deep Learning to Detect and Monitor Insects in Real Time by Analyzing Camera Trap Images. Natural and Engineering Sciences, 9(2), 280-292. https://doi.org/10.28978/nesciences.1575480
[3] Ugaz, W. A. C., Santolaya, M. D. R. H., Purizaga, H. M. R., Bravo, W. A. S., Dávila, J., Ccolque, J. Y. V., & Fuster-Guillén, D. (2023). Hybrid Internet Architecture and Protocol (HIAP): A Self-Evolving and Transformative Framework for Enabling Seamless Real-Time Applications and Secure Peer-to-Peer File Sharing in the Internet of Everything (IoE). Journal of Internet Services and Information Security, 13(3), 58-77. https://doi.org/10.58346/JISIS.2023.I3.005
[4] Majid, U. M. A, Atan, N. A., Rukli, R., & Khan, A. (2024). Framework of Computer Science Learning Through Hybrid Service Learning Oriented Visual Toward the Continuum of Visualization Thinking and Generic Skills. Indian Journal of Information Sources and Services, 14(3), 192–206. https://doi.org/10.51983/ijiss-2024.14.3.25
[5] Singh, K., Patel, D., & Rao, S. (2024). Fault-tolerant service orchestration for edge-driven ubiquitous systems. IEEE Transactions on Services Computing, 19(2), 210–223. https://doi.org/10.1109/TSC.2024.00321
[6] Jelena, T., & Srđan, K. (2023). Smart Mining: Joint Model for Parametrization of Coal Excavation Process Based on Artificial Neural Networks. Archives for Technical Sciences, 2(29), 11-22. https://doi.org/10.59456/afts.2023.1529.011T
[7] Zhao, L., & Ahmed, N. (2021). Challenges in fault management for ubiquitous platforms: A survey. ACM Computing Surveys, 54(6), 1–34. https://doi.org/10.1145/3456789
[8] Verma, A., & Mehta, N. (2024). The Role of IoT in Modern Electrical Systems: An Interdisciplinary Approach in the Periodic Series. In Smart Grid Integration (pp. 13-18). Periodic Series in Multidisciplinary Studies.
[9] Yamada, T., & Fischer, G. (2025). Context-sensitive failure handling in distributed service architectures. Future Computing Journal, 32(2), 88–101. https://doi.org/10.9012/fcj.2025.088
[10] Fayyadh, S. A. F., & Kayabaş, A. (2023). Semi-Supervised Learning for Intrusion Detection in IoT Networks with UNSW_NB15 Dataset. International Journal of Advances in Engineering and Emerging Technology, 14(1), 229–243.
[11] Abouzeid, Y., & Hassan, M. (2023). Context-aware lightweight fault prediction for ubiquitous computing. Journal of Distributed and Ubiquitous Systems, 16(1), 45–60. https://doi.org/10.3312/jdus.2023.045
[12] Mohankumar, M., Balamurugan, K., Singaravel, G., & Menaka, S. R. (2024). A Dynamic Workflow Scheduling Method based on MCDM Optimization that Manages Priority Tasks for Fault Tolerance. International Academic Journal of Science and Engineering, 11(1), 09–14. https://doi.org/10.9756/IAJSE/V11I1/IAJSE1102
[13] Bae, K., & Chen, S. (2023). Resource-aware fault handling for IoT-based ubiquitous services. Sensors and Systems Journal, 19(3), 234–248. https://doi.org/10.5567/ssj.2023.234
[14] Ganesh, R., & Siva Kumar, R. (2021). Diagnosis of Brain Tumor Using Artificial Neural Network. International Academic Journal of Innovative Research, 8(1), 06–10. https://doi.org/10.9756/IAJIR/V8I1/IAJIR0802
[15] Hussein, M., & Al-Omari, R. (2025). Energy-efficient fault recovery for edge-supported ubiquitous computing. Mobile Computing and Smart Environments, 12(2), 98–112. https://doi.org/10.7845/mcse.2025.098
[16] Asadov, B. (2018). The current state of artificial intelligence (AI) and implications for computer technologies. International Journal of Communication and Computer Technologies, 6(1), 15-18.
[17] Kumar, A., & Dey, T. (2022). Decentralized service failure management in ubiquitous systems. Journal of Pervasive Intelligence, 9(4), 321–336. https://doi.org/10.2124/jpi.2022.321

[18] Geetha, K. (2024). Advanced fault tolerance mechanisms in embedded systems for automotive safety. Journal of Integrated VLSI, Embedded and Computing Technologies, 1(1), 6-10. https://doi.org/10.31838/JIVCT/01.01.02
[19] Lin, F., & Ortega, L. (2021). Reinforcement learning for adaptive fault recovery in service compositions. IEEE Intelligent Systems, 36(5), 78–85. https://doi.org/10.1109/MIS.2021.3110024
[20] Alizadeh, M., & Mahmoudian, H. (2025). Fault-tolerant reconfigurable computing systems for high-performance applications. SCCTS Transactions on Reconfigurable Computing, 2(1), 24–32.
[21] Nair, P., & Mukherjee, R. (2023). Event-driven microservice architecture for fault-tolerant service composition. International Journal of Smart Systems, 14(2), 127–141. https://doi.org/10.6677/ijss.2023.127
[22] Park, J., & Jeong, H. (2024). Modular middleware for adaptive fault recovery in context-aware environments. Ubiquitous Systems Engineering, 20(1), 55–70. https://doi.org/10.1199/use.2024.055
[23] Vakhguelt, V., & Jianzhong, A. (2023). Renewable Energy: Wind Turbine Applications in Vibration and Wave Harvesting. Association Journal of Interdisciplinary Technics in Engineering Mechanics, 1(1), 38-48.
[24] Tsai, C., & Huang, Y. (2022). Blockchain-enhanced trust management for fault recovery in ubiquitous computing. Future Blockchain Technologies, 6(1), 17–30. https://doi.org/10.6789/fbt.2022.017
[25] Kafi, D. K., & Ayyash, A. N. (2022). Molecular structure, molecular docking and vibrational spectra of 2,2′-[1,3,5,7-tetraoxo-5,7-dihydropyrrolo[3,4-f-[isoindole 2,6(1H,3H) diyl]]] diacetohydrazide molecule by density functional theory approach as human immunodeficiency virus inhibitory [Article]. Spectroscopy Letters, 55(10), 636-647. https://doi.org/10.1080/00387010.2022.2139726
[26] Roy, A., & Saha, M. (2021). Secure fault tolerance in ubiquitous service infrastructures. Cybersecurity and Resilience Journal, 8(3), 201–215. https://doi.org/10.9011/crj.2021.201
[27] Verma, R., & Subramanian, A. (2024). Neural-symbolic fault analysis for robust service composition. AI Applications in Ubiquitous Systems, 11(2), 89–104. https://doi.org/10.5568/aiaus.2024.089
[28] Verma, M., & Banerjee, N. (2024). A Review of Sustainable Development and Women's Empowerment. International Journal of SDG's Prospects and Breakthroughs, 2(4), 13-17.
[29] Morales, A., & Tan, W. (2023). Designing reliable service compositions for pervasive systems. Journal of Ubiquitous Computing Research, 27(1), 41–58. https://doi.org/10.5678/jucr.2023.041
[30] Hilario, M., Paredes, P., Mayhuasca, J., Liendo, M., & Martínez, S. (2024). Evaluation of the Impact of Artificial Intelligence on the Systems Audit Process. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications, 15(3), 184-202. https://doi.org/10.58346/JOWUA.2024.I3.013
[31] Kalaiprasath, R., Elankavi, R., & Udayakumar, R. (2017). Cloud security and compliance-a semantic approach in end to end security. International Journal on Smart Sensing and Intelligent Systems, 10(5), 482-494.
[32] Petrova, E., & Kowalski, D. (2025). Energy-Efficient Microalgae Filtering and Harvesting Using an Extremely Low-Pressure Membrane Filter with Fouling Control. Engineering Perspectives in Filtration and Separation, 2(1), 25-31.
[33] M. Gopila, G. Suresh, D. Prasad, "Random decision forest (RDF) and crystal structure algorithm (CryStAl) for uncertainty consideration of RES & load demands with optimal design of hybrid CCHP systems", Energy, Volume 282, 2023, ISSN 0360-5442, https://doi.org/10.1016/j.energy.2023.128545.
[34] Usikalua, M. R., & Unciano, N. (2025). Mathematical modeling of epidemic dynamics: Integrating public health and data science. Bridge: Journal of Multidisciplinary Explorations, 1(1), 11–22.
[35] Barhoumia, E. M., & Khan, Z. (2025). Neurocognitive mechanisms of adaptive decision-making: An fMRI-based investigation of prefrontal cortex dynamics in uncertain environments. Advances in Cognitive and Neural Studies, 1(1), 20–27.
[36] Kavitha, M. (2024). Carbon-neutral pavement materials using recycled industrial waste and nanotechnology enhancements. Journal of Smart Infrastructure and Environmental Sustainability, 1(1), 1–13.
[37] Muralidharan, J. (2025). Integrative intervention of yoga and nutritional counseling for obesity management among college students: A holistic wellness approach. Journal of Yoga, Sports, and Health Sciences, 1(1), 17–23.
[38] Abdullah, D. (2025). Topology optimization of composite structures for enhanced crash energy absorption: Mitigating non-uniform stress distributions in crashworthy designs. Advances in Mechanical Engineering and Applications, 1(2), 1–10.
[39] Rahim, R. (2025). Bridging the gender gap in aerospace innovation: A global study of women's roles, barriers, and future directions in space technology. Journal of Women, Innovation, and Technological Empowerment, 1(1), 12–18.
[40] BRAHIMI, F., MUSTAQE, E., & DIBRA, R. (2025). Key issues of corporate governance and public finance sector. Quality-Access to Success, 26(206).