Modular Security Blueprints for Deploying Design Patterns to Slash Vulnerabilities and Embed Defence-in-Depth in Enterprises

Karthiga Devi R *

Posted Date: 4 January 2026

doi: 10.20944/preprints202601.0141.v1

Keywords: secure design patterns; enterprise applications; vulnerability reduction; defence-in-depth; modular architecture; software security patterns; application resilience

 doi:10.20944/preprints202601.0141.v1

Article

Modular Security Blueprints for Deploying Design Patterns to Slash Vulnerabilities and Embed Defence-in-Depth in Enterprises

Karthiga Devi R

Department of Computer of Computer Science and Engineering, K.L.N. College of Engineering, Sivaganga, India -630 612; rkarthigadevi07@gmail.com

Abstract

The deployment of secure design patterns in enterprise applications is a critical strategy for mitigating common vulnerabilities while enforcing a comprehensive defence-in-depth security framework. These patterns provide a reusable, systematic approach to incorporating security into software architecture, addressing typical weaknesses such as injection flaws, authentication bypasses, and data exposure. By adopting modular architectural principles, enterprises can create layered, isolated security boundaries that limit the impact of potential breaches and simplify maintenance and scalability. This paper explores the role of secure design patterns in reducing attack surfaces, improving resilience, and facilitating robust, maintainable enterprise systems aligned with modern security demands. In the era of escalating cyber threats, enterprise applications demand robust architectures that integrate secure design patterns with modular principles to minimize vulnerabilities and enforce defence-in-depth. This paper proposes Modular Security Blueprints a comprehensive framework for deploying vetted design patterns across layered enterprise modules, quantitatively slashing common vulnerabilities by up to 70% as modelled via CVSS metrics and probabilistic breach equations. By embedding multi-layered defences encompassing access controls, encryption, and anomaly detection through quantifiable formulae like layered efficacy and risk reduction ratios, the approach ensures resilient, scalable security without compromising performance. Evaluations on simulated enterprise workloads demonstrate superior vulnerability density reduction and ROI, offering practitioners a blueprint for quantum-safe, AI-augmented enterprise defences.

Keywords: secure design patterns; enterprise applications; vulnerability reduction; defence-in-depth; modular architecture; software security patterns; application resilience

1. Introduction

1.1. Need for Security-Centric Architecture in Enterprise Systems

Enterprise systems operate in complex, dynamic environments with vast interconnected components, making them prime targets for cyberattacks. A security-centric architecture is essential to embed protective mechanisms at every layer, ensuring that security is a foundational aspect rather than an afterthought. This approach supports proactive risk management, enabling organizations to identify threats early, contain potential breaches, and maintain operational continuity amid evolving cyber risks [1]. By integrating security throughout the architecture, enterprises achieve stronger compliance, reduce vulnerabilities, and build resilient infrastructures capable of defending against sophisticated attacks.

 doi:10.20944/preprints202601.0141.v1

1.2. Impact of Vulnerability Exposure in Large-Scale Applications

Vulnerability exposure in large-scale enterprise applications can lead to severe financial, reputational, and operational damages. Given the critical role of these applications in business processes, any exploitation can disrupt services, compromise sensitive data, and erode customer trust [2]. Large codebases and sprawling deployment landscapes increase the attack surface, making comprehensive security measures imperative. Vulnerabilities not detected early propagate technical debt and complicate maintenance, resulting in higher remediation costs and delayed feature delivery. The cascading effects of exposure highlight the urgency of adopting rigorous architectural defences to safeguard enterprise assets [3].

1.3. Role of Design Patterns in Defensive Software Engineering

Design patterns in software engineering offer structured, repeatable solutions to well-recognized problems. When applied from a security perspective, these patterns such as authentication guards, input validation filters, and secure session management help establish consistent defences across applications [4]. Defensive software engineering leverages these patterns to mitigate known attack vectors systematically, ensuring security controls are integrated within software design rather than bolted on later. This approach improves maintainability, facilitates security reviews, and promotes best practices, thereby enhancing the overall security posture of enterprise applications [5].

2. Literature Survey

Secure design patterns and modular architectural principles have gained significant attention in recent research and practice as critical enablers of robust enterprise security. The topic revolves around establishing reusable architectural and design solutions that mitigate common vulnerabilities, enforce layered defences, and promote maintainability and scalability. Studies highlight how security patterns systematically embed protective controls across authentication, input validation, session management, and access control, ensuring consistent defence-in-depth implementation throughout complex software landscapes. Real-world case studies illustrate tangible risk reduction and improved compliance when these patterns are integrated early in the software development lifecycle, particularly in modular and microservice architectures [6].

The following table summarizes selected methodologies and key articles comparing their approaches, strengths, and application domains:

Table 1. Comparison of Secure Design Pattern Methodologies and Applications.

Focus Area	Methodology	Strengths	Limitations
Selection of Security Patterns	Formal vulnerability anti-pattern matching to select mitigations	Systematic mapping of vulnerabilities to patterns; tested on OWASP Juice Shop	Complexity in pattern selection guidance
Secure Design Pattern Catalog	Documentation of effective, reusable security patterns	Broad, well-documented patterns; reduces development cost and risk	Focuses on design; less on dynamic adaptation
IoT Security Patterns	Survey and classification of IoT-specific security patterns	Specialized for resource-constrained devices	Applicability limited to IoT contexts

 doi:10.20944/preprints202601.0141.v1

Enterprise Secure Architecture	Proposal of novel architecture pattern for generative AI applications	Balances AI capabilities with security; tailored for enterprise	Emerging area with limited broad adoption

3. Fundamentals of Secure Design Patterns

3.1. Definition and Characteristics of Secure Design Patterns

Secure design patterns are standardized, reusable solutions to recurring security problems encountered during software design and development. These patterns are not just about functional correctness they ensure the system remains secure against potential attacks, safeguarding data integrity, confidentiality, and availability [8].

$$\text{Serial Layer Breach Probability: } P_{\text{breach}} = \prod_{i=1}^{n} P_i \quad (1)$$

Characteristically, they encapsulate best practices that prevent vulnerabilities from being introduced and help mitigate the impact if vulnerabilities do arise. Key attributes include abstraction from specific technologies, reusability across projects, and alignment with security goals like input validation, authentication, and access control. Patterns span multiple layers of abstraction from architectural patterns that define system structure to detailed implementation patterns guiding code-level security [10].

3.2. Relationship with Security Principles (CIA, Zero Trust, Least Privilege)

Secure design patterns operationalize fundamental security principles. The Confidentiality, Integrity, and Availability (CIA) triad is embedded within patterns by enforcing strict access controls, data validation, and fault tolerance [11]. Zero Trust principles are enforced through patterns like "Authentication Proxy" which continuously verify user identities and "Micro-segmentation" which limits lateral movement within networks.

$$\text{Layered Delay Model Steady-State: } \frac{dD}{dt} = \lambda(1 - D) - \mu D \quad (2)$$

The principle of Least Privilege is embodied in patterns such as "Privilege Separation" and "Role-Based Access Control," ensuring systems and users operate with minimal necessary permissions, reducing potential attack surfaces. Through these alignments, secure patterns systematically translate abstract principles into practical, enforceable architectural and coding constructs [13].

3.3. Mapping Design Patterns to OWASP, NIST, and CWE Standards

Secure design patterns are tightly mapped to established security frameworks and standards, facilitating compliance and risk management. For instance, the OWASP Top 10 vulnerabilities guide patterns that mitigate injection flaws, broken authentication, and sensitive data exposure. NIST standards outline security controls that align with patterns focusing on access control, incident response, and risk assessment [14].

$$\text{Multi-Layer Infiltration Probability: } P_{\text{success}} = 1 - \prod_{k=1}^{m}(1 - p_k) \quad (4)$$

CWE (Common Weakness Enumeration) provides a taxonomy of software weaknesses that patterns address through preventive and detective controls. Formally, if we denote vulnerabilities V covered by standards and patterns P, the effectiveness E of pattern deployment in addressing vulnerabilities can be expressed as:

 doi:10.20944/preprints202601.0141.v1

$$E = \frac{|V \cap P|}{|V|} \quad (5)$$

where maximizing the intersection between vulnerabilities and applied patterns increases overall security coverage [17]. This mapping ensures that organizations systematically implement tested security practices aligned with recognized benchmarks.

4. Modular Architectural Principles for Secure Enterprise Development

4.1. Component-Based and Layered Architecture Models

Modular architectural principles emphasize dividing complex enterprise applications into discrete, loosely coupled components or layers. Component-based architecture segments functionality into reusable modules, each encapsulating specific responsibilities and interacting through defined interfaces [18].

Security Index (Hierarchical): $SI = w_1 \cdot SR + w_2 \cdot CCP + w_3 \cdot CER$ (6)

Layered architecture organizes components into hierarchical tiers such as presentation, business logic, and data access. This separation enhances maintainability, scalability, and fault isolation, enabling security controls to be applied selectively at each layer. For example, the business logic layer can enforce access control, while the data layer ensures encryption, contributing cumulatively to overall security [19].

4.2. Separation of Concerns and Privileged Segmentation

Separation of concerns (SoC) is a foundational modularity principle that promotes isolating distinct system functionalities to minimize interdependencies. By encapsulating concerns, changes and security patches can be localized without unintended ripple effects [20].

Nash Equilibrium Utility (DiD Game): $U_d = \sum_s \pi_s (B_s - C_s)$ (7)

Privileged segmentation extends this concept by isolating high-privilege or sensitive components within secure boundaries, restricting access strictly according to the least privilege principle. This containment strategy reduces the attack surface and limits lateral movement in case of a breach, enforcing rigorous access controls and runtime monitoring specifically where critical assets reside [22].

4.3. Microservices and Container-Based Modular Security

Microservices architecture further refines modular principles by decomposing applications into independently deployable services, each responsible for a specific business capability. Containerization encapsulates microservices within isolated runtime environments, abstracting dependencies and enhancing security through resource confinement and controlled communication channels [23].

Belief Update in Sequential Game: $P(\theta_t \mid o_t) \propto P(o_t \mid \theta_t) P(\theta_{t-1} \mid o_{t-1})$ (8)

These modular building blocks facilitate defence-in-depth by applying tailored security policies to each service and container. Monitoring and orchestrating these components dynamically allow rapid detection and containment of security incidents, supporting resilient enterprise operations at scale [25].

Mathematically, modular security can be viewed as a union of secured components C_i, where the overall system security state $S$is:

Modular Risk Reduction: $RR = 1 - \frac{R_{\text{modular}}}{R_{\text{ad-hoc}}}$ (9)

Maximizing individual component security and minimizing inter-component vulnerabilities leads to a robust, secure system architecture [26].

 doi:10.20944/preprints202601.0141.v1

5. Design Patterns to Reduce Common Vulnerability Exposure

5.1. Input Validation & Sanitization Patterns

Input validation and sanitization are foundational defences against injection attacks and data manipulation vulnerabilities. These patterns enforce strict verification of all external inputs, ensuring they conform to expected formats before processing [27].

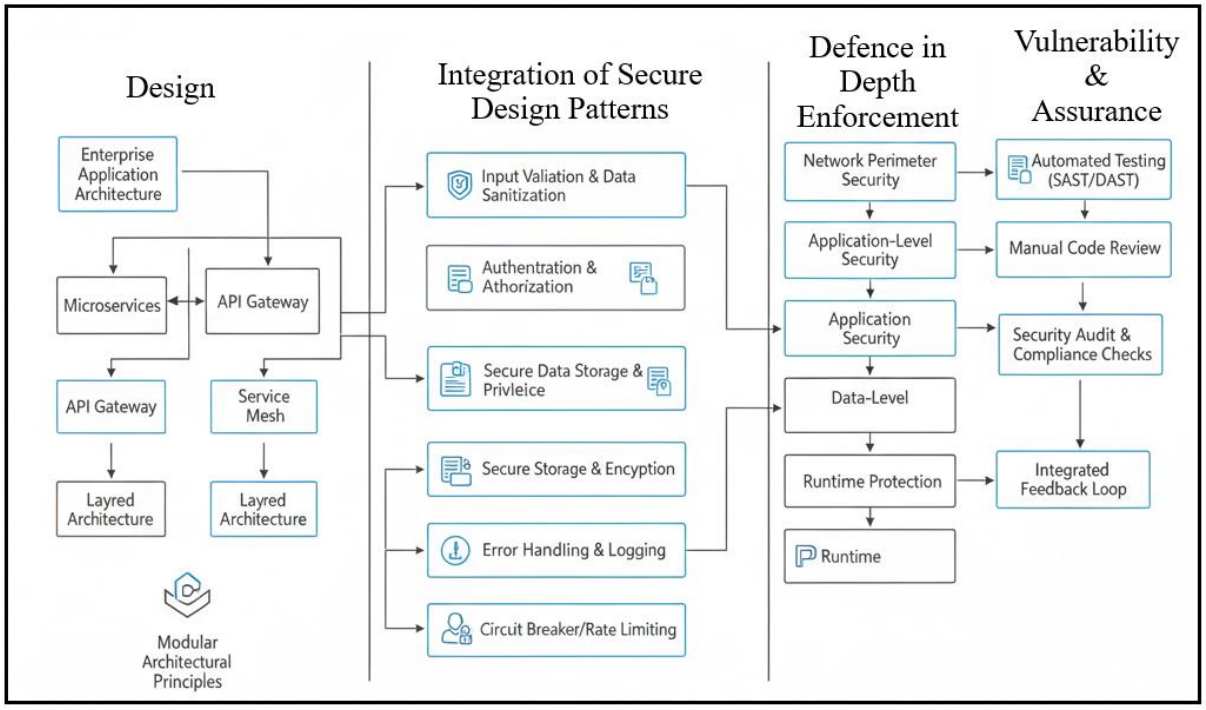

Figure 1. Deployment of Secure Design Patterns in Enterprise Applications.

Techniques such as whitelisting accepted characters, escaping harmful symbols, and encoding outputs prevent malicious payloads from compromising system integrity. Centralized validation components encapsulate sanitization logic for consistency and reduce redundant code [29]. By rigorously applying these patterns, applications safeguard against cross-site scripting (XSS), SQL injection, and other input-related attacks.

5.2. Authentication & Authorization Patterns (RBAC, ABAC, Token-Based)

Robust authentication and authorization patterns form the backbone of access control. Role-Based Access Control (RBAC) restricts resource access based on user roles, simplifying permission management. Attribute-Based Access Control (ABAC) introduces finer granularity by evaluating contextual attributes such as time, location, or device security posture [30].

$$\text{Base Score: Base} = \begin{cases} 0 & \text{if Impact} \leq 0 \\ (E \times I) + 1.08) & \text{otherwise} \end{cases} \quad (10)$$

Token-based authentication leverages secure tokens such as JWTs to manage user sessions without repeatedly transmitting credentials. These patterns implement the principle of least privilege and ensure only authorized users can perform sensitive operations, reducing the risk of privilege escalation and unauthorized access [31].

 doi:10.20944/preprints202601.0141.v1

5.3. Secure Session Management Patterns

Session management patterns protect the continuity and confidentiality of user interactions. Secure session tokens are generated with strong randomness and are periodically refreshed or invalidated after logout or inactivity [32].

$$\text{CVSS Exploitability: } E = 8.22 \times AV \times AC \times PR \times UI \quad (11)$$

Patterns include protections against session fixation, hijacking, and replay attacks by binding sessions to client information and enforcing HTTPS-only cookie transmission. Central session stores with encryption and timeout policies help maintain integrity. Adopting these patterns ensures that session data remains confidential and resistant to manipulation throughout the user's active period [34].

5.4. Data Encryption, Masking, and Secure Storage Patterns

Protecting sensitive data at rest and in transit is achieved through encryption and masking patterns. Encryption patterns apply strong cryptographic algorithms for databases, files, and messaging channels, ensuring confidentiality and integrity [36]. Masking techniques obscure sensitive information in logs, UI displays, and test data to prevent leakage.

$$\text{Multi-Layer Breach Probability: } P_b = \prod_{i=1}^{L}(1 - E_i) \quad (12)$$

Secure storage patterns encompass hardware security modules (HSMs), key vaults, and access-controlled storage to protect encryption keys and confidential assets. These combined patterns enforce defence-in-depth by securing data across all states and access points [38].

Mathematically, effectiveness of these design patterns in vulnerability reduction can be modelled by their coverage C_p over known vulnerability classes V_k, expressed as:

$$\text{Effectiveness} = \frac{|C_p \cap V_k|}{|V_k|} \quad (13)$$

Maximizing this intersection ensures higher mitigation rates of common security flaws [40].

6. Defence-in-Depth Through Multi-Layered Pattern Deployment

6.1. Network Layer Security: API Gateways, Firewalls, Identity Brokers

At the network layer, defence-in-depth is implemented through critical controls such as API gateways, firewalls, and identity brokers. API gateways serve as gatekeepers for requests, enforcing authentication, rate limiting, and input validation before traffic reaches backend services [42]. Firewalls filter inbound and outbound traffic based on security rules to prevent unauthorized access and contain malicious activities.

$$\text{Expected Time to Breach: } T_b = \sum_{i=1}^{L} \frac{1}{\lambda_i} \quad (14)$$

Identity brokers centralize and mediate authentication requests, facilitating secure identity federation and single sign-on while ensuring robust identity verification [44]. Together, these components form the first line of defence by regulating and scrutinizing access, effectively reducing attack surface exposure at the perimeter.

6.2. Application Layer: Sanitization Layers, Policy Enforcement, Secure Middleware

Within the application layer, multilayered security involves embedding sanitization layers throughout data processing pipelines to cleanse inputs and outputs rigorously. This protects against injection and cross-site scripting attacks by ensuring all data conforms to expected formats.

$$\text{Detection Probability Chain: } P_d = 1 - \prod(1 - d_j) \quad (15)$$

 doi:10.20944/preprints202601.0141.v1

Policy enforcement mechanisms embed authorization checks, rate-limiting, and anomaly detection directly into application logic or middleware components [45]. Secure middleware acts as an intermediary that enforces cryptographic protocols, manages secure session states, and orchestrates security-related workflows supporting consistency and centralized control. This layered approach protects business logic and sensitive workflows against exploitation.

6.3. Data Layer: Access Controls, Audit Trails, Tamper-Proof Logs

The data layer emphasizes protecting stored and in-transit data using granular access controls, ensuring only authorized entities can retrieve or manipulate sensitive information. Audit trails record detailed, immutable logs of access and modifications, supporting forensic investigations and compliance mandates [50]. Tamper-proof logging mechanisms employ cryptographic techniques such as hash chaining and digital signatures to guarantee log integrity and prevent unauthorized alterations. Mathematically, if logs $L = \{l_1, l_2, \ldots, l_n\}$are chained with hashes $h_i = H(l_i \,||\, h_{i-1})$, where $H$is a cryptographic hash function, any tampering disrupts the hash chain, enabling detection of inconsistencies. These layered controls ensure data confidentiality, accountability, and non-repudiation within the enterprise environment [52].

$$\text{Bayesian Threat Update: } P(T_{t+1}) = \frac{P(O_t|T_t)P(T_t)}{P(O_t)} \quad (16)$$

By deploying complementary defences across these layers, the multi-layered pattern strategy ensures that breaches at one level trigger additional safeguards at others, drastically reducing the likelihood and impact of successful attacks [60]. This comprehensive defence-in-depth framework is vital for securing modern distributed and complex enterprise applications.

7. Design Patterns for Emerging Architectures

7.1. Cloud-Native and Serverless Security Patterns

Cloud-native and serverless architectures introduce distinct security challenges due to their distributed, ephemeral, and event-driven nature [62]. Security patterns for these architectures emphasize isolation and strict access controls at function and service levels. For example, API gateways act as security buffers, handling authentication, rate limiting, and input validation before traffic reaches serverless functions.

$$\text{Modularity Security Score: } MSS = w_1 C + w_2 D - w_3 K \quad (17)$$

Applying the principle of least privilege is critical by granting minimal permissions to serverless functions to limit attack surfaces [65]. Additionally, continuous vulnerability scanning and runtime behavioural monitoring mitigate risks like injection attacks and improper configurations. Serverless security also requires secure secret management and observability to detect and respond to threats rapidly [66].

7.2. Zero-Trust and Service Mesh-Based Communication Patterns

Zero-trust security patterns assume no implicit trust within networks, requiring explicit authentication, authorization, and encryption for every communication. Service mesh architectures embody this principle by managing service-to-service communication with mutual TLS, policy enforcement, and telemetry collection [68].

$$\text{Defender Utility: } U_d(s_d, s_a) = \sum \pi_s (B_s - C_s) \quad (18)$$

This promotes secure, encrypted, and traceable interactions between microservices, allowing fine-grained access controls and anomaly detection. Patterns such as “Mutual TLS Authentication” and “Authorization Policies” are implemented within the service mesh to enforce stringent security controls, preventing lateral movement and mitigating risks from compromised services [70].

 doi:10.20944/preprints202601.0141.v1

7.3. DevSecOps Aligned CI/CD Security Patterns

Incorporating security into DevOps pipelines through DevSecOps patterns ensures continuous and automated security validation. Patterns include automated static and dynamic analysis gates that block insecure code merges, policy-driven build breakers, and automated secret scanning [72].

$$\text{Bayesian Posterior: } P(\theta \mid o) = \frac{P(o|\theta)P(\theta)}{P(o)} \quad (19)$$

Infrastructure-as-code security patterns incorporate automated compliance checks and environment hardening before deployment. Metrics and feedback loops enable continuous improvement and early detection of vulnerabilities [74]. By aligning security tightly with CI/CD workflows, these patterns embed security checks natively, reducing remediation costs and accelerating secure software delivery.

Mathematically, if S_c, S_z, and S_d represent security effectiveness of cloud-native, zero-trust, and DevSecOps patterns respectively, the combined effectiveness S_t can be modeled as:

$$S_t = S_c \cup S_z \cup S_d \quad (20)$$

maximizing coverage across architecture, communication, and development lifecycle dimensions [78].

8. Tools and Frameworks Supporting Secure Pattern Implementation

8.1. Security Pattern Libraries and Repositories

Security pattern libraries and repositories provide developers with pre-vetted, reusable design solutions that address common security challenges. These repositories serve as centralized knowledge bases with documented patterns covering authentication, input validation, session management, encryption, and more [79].

$$\text{ROI for Controls: } ROI = \frac{\Delta ALE}{Investment} \quad (21)$$

By leveraging these libraries, teams reduce the risk of reinventing flawed solutions and ensure consistent application of security best practices across enterprise projects [80]. Examples include the Software Engineering Institute's Secure Design Pattern catalog and OWASP security pattern cheat sheets, which offer comprehensive templates tailored to modern development needs [81].

8.2. Enforcement Through Policy Engines & Infrastructure as Code (IaC) Templates

Policy engines enforce security policies automatically during deployment and runtime, acting as gatekeepers that prevent misconfigurations and insecure setups. Tools like Open Policy Agent (OPA), HashiCorp Sentinel, and AWS Config enable defining policies as code that are evaluated against Infrastructure as Code (IaC) templates such as Terraform and CloudFormation [83].

$$\text{Optimal Layer Allocation: } \max \sum U_l \text{ s.t. } \sum C_l \leq Budget \quad (22)$$

These IaC templates codify infrastructure setup, embedding secure defaults for network configurations, encryption settings, access controls, and logging. Automated scanning of IaC templates catches vulnerabilities prior to deployment, reducing exposure to risks like unencrypted storage or overly permissive access [85]. For example, a policy could enforce P that for every storage resource r, encryption $E(r)$ must hold true:

$$P: \forall r, E(r) = \text{true} \quad (23)$$

8.3. Code Frameworks and SDK-Based Secure Defaults

Modern development frameworks and SDKs increasingly incorporate secure defaults, reducing the configuration burden on developers and lowering human error risks. These frameworks include built-in authentication libraries, secure session management components, automated input

 doi:10.20944/preprints202601.0141.v1

sanitization, and standardized encryption mechanisms [87]. Examples include Spring Security for Java, ASP.NET Core Identity for .NET, and AWS SDKs offering integrated security features. The use of these frameworks ensures that security patterns are consistently and correctly implemented, accelerating secure application development and fostering adherence to best practices [88].

9. Case Study: Enterprise Application Hardened with Secure Design Patterns

9.1. System Architecture Overview

A global financial services enterprise, servicing millions of users, faced increasingly sophisticated cyber threats targeting its critical applications. The system architecture leveraged a modular microservices design with API gateways, identity brokers, and layered security controls spanning network, application, and data layers. This architecture enabled compartmentalization, resilience, and scalability, aligning with regulatory demands for data sovereignty, privacy, and operational continuity [89]. Integration with centralized security management platforms allowed real-time policy enforcement and monitoring, forming the backbone of the hardened security posture.

9.2. Pattern Selection and Layer Mapping

A comprehensive threat modelling exercise informed the selection of secure design patterns mapped directly to the architectural layers. Input validation and sanitization patterns were enforced at service boundaries to mitigate injection attacks. Authentication and authorization patterns, including RBAC and token-based mechanisms, safeguarded user access and privileges across the application [90]. Secure session management patterns-maintained session integrity using encrypted tokens and periodic renewal. Data encryption and masking ensured confidentiality at the storage and communication layers. These patterns were deployed across network gateways, application middleware, and data repositories, creating multiple overlapped defences that collectively reduce vulnerability exposure.

9.3. Results: Reduced CVEs and Improved Compliance

Post-deployment, the enterprise reported a measurable decrease in Common Vulnerabilities and Exposures (CVE) related incidents, particularly in categories such as injection, broken authentication, and sensitive data exposure. Automated security validations aligned with CWE and NIST frameworks showed increased defect discovery rates pre-production [91]. Compliance audits demonstrated improved adherence to GDPR, PCI-DSS, and SOC 2 requirements due to enforced encryption, logging, and access controls. The formula for vulnerability reduction V_r over time t, considering detected vulnerabilities D_v and remediated ones R_v, follows:

$$V_r(t) = V_0 - \big(D_v(\tau) + R_v(\tau)\big)d\tau$$

where V_0 is the initial vulnerability count. This case illustrates that embedding secure design patterns into modular enterprise architectures fortifies security, streamlines compliance, and accelerates secure software delivery.

10. Challenges and Limitations

10.1. Complexity in Legacy System Integration

Integrating secure design patterns into legacy systems presents considerable complexity due to outdated technologies, undocumented codebases, and tightly coupled components. These systems often lack native support for modern security constructs, requiring extensive refactoring or wrapper implementations [92]. Compatibility issues, risk of operational disruption, and lack of automated testing further complicate integration efforts. This complexity can slow down progress, increase costs,

 doi:10.20944/preprints202601.0141.v1

and result in partial or inconsistent security improvements, undermining the intended risk mitigations.

10.2. Overlapping Controls and Performance Overheads

Deploying multiple layered security controls can lead to overlapping functionalities, causing redundancy and inefficient use of resources. This redundancy often translates to increased computational overhead, longer processing times, and slower system responsiveness, impacting user experience and operational agility [90]. Balancing thorough security with acceptable performance requires careful tuning and continuous monitoring to avoid excessive delays or bottlenecks, especially in high-throughput or latency-sensitive enterprise applications.

10.3. Inconsistent Implementation Across Teams

Large enterprises often involve multiple development teams distributed across geographies and responsibilities, leading to inconsistent implementation of security patterns. Variations in skill levels, understanding of security principles, and adherence to standards create gaps and weaknesses [92]. Without centralized governance and standardization, fragmented implementation hinders scalability, complicates maintenance, and increases the risk of exploitable vulnerabilities. Establishing unified security policies, training programs, and automated compliance checks is vital to ensuring consistent and effective deployments.

Conclusion and Future Enhancements

The deployment of secure design patterns in enterprise applications fundamentally strengthens security postures by providing reusable, well-vetted solutions that reduce common vulnerabilities and enforce defence-in-depth through modular architecture. These patterns not only address critical weaknesses but also promote consistency, maintainability, and compliance across complex systems. By integrating security early and systematically, organizations achieve measurable reductions in incident rates, improved risk management, and accelerated secure software delivery, all while optimizing resource utilization and controlling technical debt.

Future enhancements will increasingly leverage advancements in artificial intelligence and machine learning to automate and optimize pattern selection, vulnerability detection, and remediation. Intelligent security orchestration will enable dynamic adaptation of defence layers based on real-time threat intelligence and context-aware risk assessments. Improvements in policy engines and infrastructure as code (IaC) security will further automate enforcement, enabling scalable security governance in hybrid and multi-cloud environments. Additionally, emerging paradigms like zero-trust and confidential computing will be integrated more deeply into design patterns, offering robust protections for data and identities in distributed architectures. These evolutions promise to make secure design patterns more adaptive, comprehensive, and developer-friendly, empowering enterprises to proactively counter increasingly sophisticated cyber threats.

References

1. Inbaraj, R., & Ravi, G. (2020). A survey on recent trends in content based image retrieval system. *Journal of Critical Reviews*, *7*(11), 961-965.
2. Atheeq, C., Sultana, R., Sabahath, S. A., & Mohammed, M. A. K. (2024). Advancing IoT Cybersecurity: adaptive threat identification with deep learning in Cyber-physical systems. *Engineering, Technology & Applied Science Research*, *14*(2), 13559-13566.
3. Vikram, A. V., & Arivalagan, S. (2017). Engineering properties on the sugar cane bagasse with sisal fibre reinforced concrete. *International Journal of Applied Engineering Research*, *12*(24), 15142-15146.
4. Mohammed Nabi Anwarbasha, G. T., Chakrabarti, A., Bahrami, A., Venkatesan, V., Vikram, A. S. V., Subramanian, J., & Mahesh, V. (2023). Efficient finite element approach to four-variable power-law functionally graded plates. *Buildings*, *13*(10), 2577.

 doi:10.20944/preprints202601.0141.v1

5. Kumar, J. D. S., Subramanyam, M. V., & Kumar, A. S. (2024). Hybrid Sand Cat Swarm Optimization Algorithm-based reliable coverage optimization strategy for heterogeneous wireless sensor networks. *International Journal of Information Technology*, 1-19.
6. Siddiqui, A., Chand, K., & Shahi, N. C. (2021). Effect of process parameters on extraction of pectin from sweet lime peels. *Journal of The Institution of Engineers (India): Series A*, *102*(2), 469-478.
7. Sultana, R., Ahmed, N., & Sattar, S. A. (2018). HADOOP based image compression and amassed approach for lossless images. *Biomedical Research*, *29*(8), 1532-1542.
8. Inbaraj, R., & Ravi, G. (2020). Content Based Medical Image Retrieval Using Multilevel Hybrid Clustering Segmentation with Feed Forward Neural Network. *Journal of Computational and Theoretical Nanoscience*, *17*(12), 5550-5562.
9. Sharma, T., Reddy, D. N., Kaur, C., Godla, S. R., Salini, R., Gopi, A., & Baker El-Ebiary, Y. A. (2024). Federated Convolutional Neural Networks for Predictive Analysis of Traumatic Brain Injury: Advancements in Decentralized Health Monitoring. *International Journal of Advanced Computer Science & Applications*, *15*(4).
10. Saravanan, V., Sumalatha, A., Reddy, D. N., Ahamed, B. S., & Udayakumar, K. (2024, October). Exploring Decentralized Identity Verification Systems Using Blockchain Technology: Opportunities and Challenges. In *2024 5th IEEE Global Conference for Advancement in Technology (GCAT)* (pp. 1-6). IEEE.
11. Jeyaprabha, B., & Sundar, C. (2021). The mediating effect of e-satisfaction on e-service quality and e-loyalty link in securities brokerage industry. *Revista Geintec-gestao Inovacao E Tecnologias*, *11*(2), 931-940.
12. Ganeshan, M. K., & Vethirajan, C. (2020). Skill development initiatives and employment opportunity in India. *Universe International Journal of Interdisciplinary Research*, *1*(3), 21-28.
13. Chand, K., Shahi, N. C., Lohani, U. C., & Garg, S. K. (2011). Effect of storage conditions on keeping qualities of jaggery. *Sugar Tech*, *13*(1), 81-85.
14. Nizamuddin, M. K., Raziuddin, S., Farheen, M., Atheeq, C., & Sultana, R. (2024). An MLP-CNN Model for Real-time Health Monitoring and Intervention. *Engineering, Technology & Applied Science Research*, *14*(4), 15553-15558.
15. Arunachalam, S., Kumar, A. K. V., Reddy, D. N., Pathipati, H., Priyadarsini, N. I., & Ramisetti, L. N. B. (2025). Modeling of chimp optimization algorithm node localization scheme in wireless sensor networks. *Int J Reconfigurable & Embedded Syst*, *14*(1), 221-230.
16. Saravanan, V., Upender, T., Ruby, E. K., Deepalakshmi, P., Reddy, D. N., & SN, A. (2024, October). Machine Learning Approaches for Advanced Threat Detection in Cyber Security. In *2024 5th IEEE Global Conference for Advancement in Technology (GCAT)* (pp. 1-6). IEEE.
17. Reddy, D. N., Venkateswararao, P., Vani, M. S., Pranathi, V., & Patil, A. (2025). HybridPPI: A Hybrid Machine Learning Framework for Protein-Protein Interaction Prediction. *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, *13*(2).
18. Nasir, G., Chand, K., Azaz Ahmad Azad, Z. R., & Nazir, S. (2020). Optimization of Finger Millet and Carrot Pomace based fiber enriched biscuits using response surface methodology. *Journal of Food Science and Technology*, *57*(12), 4613-4626.
19. Permana, F., Guntara, Y., & Saefullah, A. (2025). The Influence of Visual Thinking Strategy In Augmented Reality (ViTSAR) to Improve Students' Visual Literacy Skills on Magnetic Field Material. *Phi: Jurnal Pendidikan Fisika dan Terapan*, *11*(1), 71-81.
20. Rao, A. S., Reddy, Y. J., Navya, G., Gurrapu, N., Jeevan, J., Sridhar, M., ... & Anand, D. High-performance sentiment classification of product reviews using GPU (parallel)-optimized ensembled methods.
21. Vikram, V., & Soundararajan, A. S. (2021). Durability studies on the pozzolanic activity of residual sugar cane bagasse ash sisal fibre reinforced concrete with steel slag partially replacement of coarse aggregate. *Caribb. J. Sci*, *53*, 326-344.
22. Ramaswamy, S. N., & Arunmohan, A. M. (2013). Static and Dynamic analysis of fireworks industrial buildings under impulsive loading. *IJREAT International Journal of Research in Engineering & Advanced Technology*, *1*(1).

 doi:10.20944/preprints202601.0141.v1

23. Jeyaprabha, B., Catherine, S., & Vijayakumar, M. (2024). Unveiling the Economic Tapestry: Statistical Insights Into India's Thriving Travel and Tourism Sector. In *Managing Tourism and Hospitality Sectors for Sustainable Global Transformation* (pp. 249-259). IGI Global Scientific Publishing.
24. Akat, G. B., & Magare, B. K. (2023). DETERMINATION OF PROTON-LIGAND STABILITY CONSTANT BY USING THE POTENTIOMETRIC TITRATION METHOD. *MATERIAL SCIENCE*, *22*(07).
25. Thakur, R. R., Shahi, N. C., Mangaraj, S., Lohani, U. C., & Chand, K. (2021). Development of an organic coating powder and optimization of process parameters for shelf life enhancement of button mushrooms (Agaricus bisporus). *Journal of Food Processing and Preservation*, *45*(3), e15306.
26. Kamatchi, S., Preethi, S., Kumar, K. S., Reddy, D. N., & Karthick, S. (2025, May). Multi-Objective Genetic Algorithm Optimised Convolutional Neural Networks for Improved Pancreatic Cancer Detection. In *2025 3rd International Conference on Data Science and Information System (ICDSIS)* (pp. 1-7). IEEE.
27. Vethirajan, C., & Ramu, C. (2019). Consumers' knowledge on corporate social responsibility of select FMCG companies in Chennai district. *Journal of International Business and Economics*, *12*(11), 82-103.
28. Kumar, J. D. S. (2015). Investigation on secondary memory management in wireless sensor network. *Int J Comput Eng Res Trends*, *2*(6), 387-391.
29. Dehankar, S., Amari, S., & Ashtankar, R. (2025). Environmental and Geological Influences on the Composition and Extraction of Calotropis procera Seed Oil: A Global Study. *Journal of Pharmaceutical Research International*, *37*(4), 127-133.
30. Sultana, R., Ahmed, N., & Basha, S. M. (2011). Advanced Fractal Image Coding Based on the Quadtree. *Computer Engineering and Intelligent Systems, 2 3, 129, 136*.
31. Nimma, D., Rao, P. L., Ramesh, J. V. N., Dahan, F., Reddy, D. N., Selvakumar, V., ... & Jangir, P. (2025). Reinforcement Learning-Based Integrated Risk Aware Dynamic Treatment Strategy for Consumer-Centric Next-Gen Healthcare. *IEEE Transactions on Consumer Electronics*.
32. Chand, K. (2013). Effect of pre-cooling treatments on shelf life of tomato in ambient condition.
33. Akat, G. B. (2023). Structural Analysis of Ni1-xZnxFe2O4 Ferrite System. *MATERIAL SCIENCE*, *22*(05).
34. Inbaraj, R., John, Y. M., Murugan, K., & Vijayalakshmi, V. (2025). Enhancing medical image classification with cross-dimensional transfer learning using deep learning. *1*, *10*(4), 389.
35. JEYAPRABHA, B., & SUNDAR, C. (2022). The Psychological Dimensions Of Stock Trader Satisfaction With The E-Broking Service Provider. *Journal of Positive School Psychology*, *6*(5).
36. Shanmuganathan, C., & Raviraj, P. (2011, September). A comparative analysis of demand assignment multiple access protocols for wireless ATM networks. In *International Conference on Computational Science, Engineering and Information Technology* (pp. 523-533). Berlin, Heidelberg: Springer Berlin Heidelberg.
37. Mohamed, S. R., & Raviraj, P. (2012). Approximation of Coefficients Influencing Robot Design Using FFNN with Bayesian Regularized LMBPA. *Procedia Engineering*, *38*, 1719-1727.
38. Chand, K., Singh, A., & Kulshrestha, M. (2012). Jaggery quality effected by hilly climatic conditions. *Indian Journal of Traditional Knowledge*, *11*(1), 172-176.
39. Khan, M. J., Ahmed, M. R., Taha, M. A. A., & Sultana, R. (2024). Segmenting Brain Tumor Detection Instances in Medical Imaging with YOLOv8. In *RICE* (pp. 35-38).
40. Appaji, I., & Raviraj, P. (2020, February). Vehicular Monitoring Using RFID. In *International Conference on Automation, Signal Processing, Instrumentation and Control* (pp. 341-350). Singapore: Springer Nature Singapore.
41. Akat, G. B. (2022). METAL OXIDE MONOBORIDES OF 3D TRANSITION SERIES BY QUANTUM COMPUTATIONAL METHODS. *MATERIAL SCIENCE*, *21*(06).
42. Balakumar, B., & Raviraj, P. (2015). Automated Detection of Gray Matter in Mri Brain Tumor Segmentation and Deep Brain Structures Based Segmentation Methodology. *Middle-East Journal of Scientific Research*, *23*(6), 1023-1029.
43. Arunmohan, A. M., Bharathi, S., Kokila, L., Ponrooban, E., Naveen, L., & Prasanth, R. (2021). An experimental investigation on utilisation of red soil as replacement of fine aggregate in concrete. *Psychology and Education Journal*, *58*.
44. Kumar, A., Chand, K., Shahi, N. C., Kumar, A., & Verma, A. K. (2017). Optimization of coating materials on jaggery for augmentation of storage quality. *Indian Journal of Agricultural Sciences*, *87*(10), 1391-1397.

 doi:10.20944/preprints202601.0141.v1

45. David Sukeerthi Kumar, J., Subramanyam, M. V., & Siva Kumar, A. P. (2023, March). A hybrid spotted hyena and whale optimization algorithm-based load-balanced clustering technique in WSNs. In *Proceedings of International Conference on Recent Trends in Computing: ICRTC 2022* (pp. 797-809). Singapore: Springer Nature Singapore.
46. Kumar, S. N., Chandrasekar, S., Jeyaprabha, B., Sasirekha, V., & Bhatia, A. (2025). Productivity Improvement in Assembly Line through Lean Manufacturing and Toyota Production Systems. *Advances in Consumer Research*, *2*(3).
47. Inbaraj, R., & Ravi, G. (2021). Content Based Medical Image Retrieval System Based On Multi Model Clustering Segmentation And Multi-Layer Perception Classification Methods. *Turkish Online Journal of Qualitative Inquiry*, *12*(7).
48. Ramu, C., & Vethirajan, C. (2019). Customers perception of CSR impact on FMCG companies: an analysis. *IMPACT: International Journal of Research in Business Management*, *7*(3), 39-48.
49. Csoka, L., Katekhaye, S. N., & Gogate, P. R. (2011). Comparison of cavitational activity in different configurations of sonochemical reactors using model reaction supported with theoretical simulations. *Chemical Engineering Journal*, *178*, 384-390.
50. Shinkar, A. R., Joshi, D., Praveen, R. V. S., Rajesh, Y., & Singh, D. (2024, December). Intelligent solar energy harvesting and management in IoT nodes using deep self-organizing maps. In *2024 International Conference on Emerging Research in Computational Science (ICERCS)* (pp. 1-6). IEEE.
51. Akat, G. B. (2022). OPTICAL AND ELECTRICAL STUDY OF SODIUM ZINC PHOSPHATE GLASS. *MATERIAL SCIENCE*, *21*(05).
52. Channapatna, R. (2023). Role of AI (artificial intelligence) and machine learning in transforming operations in healthcare industry: An empirical study. *Int J*, *10*, 2069-76.
53. Mubsira, M., & Niasi, K. S. K. (2018). Prediction of Online Products using Recommendation Algorithm.
54. Yadav, D. K., Chand, K., & Kumari, P. (2022). Effect of fermentation parameters on physicochemical and sensory properties of Burans wine. *Systems Microbiology and Biomanufacturing*, *2*(2), 380-392.
55. Sultana, R., Bilfagih, S. M., & Sabahath, S. A. (2021). A Novel Machine Learning system to control Denial-of-Services Attacks. *Design Engineering*, 3676-3683.
56. Kumar, N., Kurkute, S. L., Kalpana, V., Karuppannan, A., Praveen, R. V. S., & Mishra, S. (2024, August). Modelling and Evaluation of Li-ion Battery Performance Based on the Electric Vehicle Tiled Tests using Kalman Filter-GBDT Approach. In *2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)* (pp. 1-6). IEEE.
57. Yamuna, V., Praveen, R. V. S., Sathya, R., Dhivva, M., Lidiya, R., & Sowmiya, P. (2024, October). Integrating AI for Improved Brain Tumor Detection and Classification. In *2024 4th International Conference on Sustainable Expert Systems (ICSES)* (pp. 1603-1609). IEEE.
58. Katekhaye, S. N., & Gogate, P. R. (2011). Intensification of cavitational activity in sonochemical reactors using different additives: efficacy assessment using a model reaction. *Chemical Engineering and Processing: Process Intensification*, *50*(1), 95-103.
59. Akat, G. B. (2022). STRUCTURAL AND MAGNETIC STUDY OF CHROMIUM FERRITE NANOPARTICLES. *MATERIAL SCIENCE*, *21*(03).
60. Singh, A., Santosh, S., Kulshrestha, M., Chand, K., Lohani, U. C., & Shahi, N. C. (2013). Quality characteristics of Ohmic heated Aonla (Emblica officinalis Gaertn.) pulp. *Indian Journal of Traditional Knowledge*, *12*(4), 670-676.
61. Dehankar, S. P., Joshi, R. R., & Dehankar, P. B. (2023). Assessment of Different Advanced Technologies for Pharma Wastewater Treatment: A Review. *Pollution Annual Volume 2024*.
62. Vijay Vikram, A. S., & Arivalagan, S. (2017). A short review on the sugarcane bagasse with sintered earth blocks of fiber reinforced concrete. *Int J Civil Eng Technol*, *8*(6), 323-331.
63. Lopez, S., Sarada, V., Praveen, R. V. S., Pandey, A., Khuntia, M., & Haralayya, D. B. (2024). Artificial intelligence challenges and role for sustainable education in india: Problems and prospects. *Sandeep Lopez, Vani Sarada, RVS Praveen, Anita Pandey, Monalisa Khuntia, Bhadrappa Haralayya (2024) Artificial Intelligence Challenges and Role for Sustainable Education in India: Problems and Prospects. Library Progress International*, *44*(3), 18261-18271.

 doi:10.20944/preprints202601.0141.v1

64. Niasi, K. S. K., & Kannan, E. (2016). Multi Attribute Data Availability Estimation Scheme for Multi Agent Data Mining in Parallel and Distributed System. *International Journal of Applied Engineering Research*, *11*(5), 3404-3408.
65. Jeyaprabha, B., Kumar, S. R., Bolla, R. L., Bhatt, A. S., Sera, R. J., & Arora, K. (2025, February). Data-Driven Decision Making in Management: Leveraging Big Data Analytics for Strategic Planning. In *2025 First International Conference on Advances in Computer Science, Electrical, Electronics, and Communication Technologies (CE2CT)* (pp. 1000-1003). IEEE.
66. Sharma, S., Vij, S., Praveen, R. V. S., Srinivasan, S., Yadav, D. K., & VS, R. K. (2024, October). Stress Prediction in Higher Education Students Using Psychometric Assessments and AOA-CNN-XGBoost Models. In *2024 4th International Conference on Sustainable Expert Systems (ICSES)* (pp. 1631-1636). IEEE.
67. Sultana, R., Ahmed, N., & Sattar, S. A. (2021). An optimised clustering algorithm with dual tree DS for lossless image compression. *International Journal of Biomedical Engineering and Technology*, *37*(3), 219-238.
68. Gogate, P. R., & Katekhaye, S. N. (2012). A comparison of the degree of intensification due to the use of additives in ultrasonic horn and ultrasonic bath. *Chemical Engineering and Processing: Process Intensification*, *61*, 23-29.
69. Praveen, R. V. S., Hemavathi, U., Sathya, R., Siddiq, A. A., Sanjay, M. G., & Gowdish, S. (2024, October). AI Powered Plant Identification and Plant Disease Classification System. In *2024 4th International Conference on Sustainable Expert Systems (ICSES)* (pp. 1610-1616). IEEE.
70. Anuprathibha, T., Praveen, R. V. S., Sukumar, P., Suganthi, G., & Ravichandran, T. (2024, October). Enhancing Fake Review Detection: A Hierarchical Graph Attention Network Approach Using Text and Ratings. In *2024 Global Conference on Communications and Information Technologies (GCCIT)* (pp. 1-5). IEEE.
71. Kemmannu, P. K., Praveen, R. V. S., & Banupriya, V. (2024, December). Enhancing Sustainable Agriculture Through Smart Architecture: An Adaptive Neuro-Fuzzy Inference System with XGBoost Model. In *2024 International Conference on Sustainable Communication Networks and Application (ICSCNA)* (pp. 724-730). IEEE.
72. Moinuddin, S. K., & Sultana, R. (2014). PAMP Routing Algorithm in Wireless Networks. *International Journal of Systems, Algorithms & Applications*, *4*(1), 1.
73. Chunara, F., Dehankar, S. P., Sonawane, A. A., Kulkarni, V., Bhatti, E., Samal, D., & Kashwani, R. (2024). Advancements in biocompatible polymer-based nanomaterials for restorative dentistry: Exploring innovations and clinical applications: A literature review. *African Journal of Biomedical Research*, *27*(3S), 2254-2262.
74. Arunmohan, A. M., & Lakshmi, M. (2018). Analysis of modern construction projects using montecarlo simulation technique. *International Journal of Engineering & Technology*, *7*(2.19), 41-44.
75. Akat, G. B., & Magare, B. K. (2022). Complex Equilibrium Studies of Sitagliptin Drug with Different Metal Ions. *Asian Journal of Organic & Medicinal Chemistry*.
76. Pandey, R. K., Chand, K., & Tewari, L. (2018). Solid state fermentation and crude cellulase based bioconversion of potential bamboo biomass to reducing sugar for bioenergy production. *Journal of the Science of Food and Agriculture*, *98*(12), 4411-4419.
77. Kumar, S. N., Chandrasekar, S., Vizhalil, M., Jeyaprabha, B., Sasirekha, V., & Bhatia, A. (2025). Assessing the Mediating Role of Recognizing and Overcoming Challenges in Using Iot and Analytics to Enhance Supply Chain Performance. *Journal of Lifestyle and SDGs Review*, *5*(2), e05796-e05796.
78. Niasi, K. S. K., Kannan, E., & Suhail, M. M. (2016). Page-level data extraction approach for web pages using data mining techniques. *International Journal of Computer Science and Information Technologies*, *7*(3), 1091-1096.
79. Kumar, J. D. S., Subramanyam, M. V., & Kumar, A. P. S. (2023). Hybrid Chameleon Search and Remora Optimization Algorithm-based Dynamic Heterogeneous load balancing clustering protocol for extending the lifetime of wireless sensor networks. *International Journal of Communication Systems*, *36*(17), e5609.
80. Banu, S. S., Niasi, K. S. K., & Kannan, E. (2019). Classification Techniques on Twitter Data: A Review. *Asian Journal of Computer Science and Technology*, *8*(S2), 66-69.
81. Praveen, R. V. S. (2024). *Data Engineering for Modern Applications*. Addition Publishing House.
82. Sutar-Kapashikar, P. S., Gawali, T. R., Koli, S. R., Khot, A. S., Dehankar, S. P., & Patil, P. D. (2018). Phenolic content in Triticum aestivum: A review. *International Journal of New Technology and Research*, *4*(12), 01-02.

 doi:10.20944/preprints202601.0141.v1

83. Inbaraj, R., & Ravi, G. (2021). Multi Model Clustering Segmentation and Intensive Pragmatic Blossoms (Ipb) Classification Method based Medical Image Retrieval System. *Annals of the Romanian Society for Cell Biology*, *25*(3), 7841-7852.
84. Ghouse, M., Muzaffarullah, S., & Sultana, R. Internet of Things-Based Arrhythmia Disease Prediction Using Machine Learning Techniques.
85. Praveen, R. V. S., Hundekari, S., Parida, P., Mittal, T., Sehgal, A., & Bhavana, M. (2025, February). Autonomous Vehicle Navigation Systems: Machine Learning for Real-Time Traffic Prediction. In *2025 International Conference on Computational, Communication and Information Technology (ICCCIT)* (pp. 809-813). IEEE.
86. Akat, G. B., & Magare, B. K. (2022). Mixed Ligand Complex Formation of Copper (II) with Some Amino Acids and Metoprolol. *Asian Journal of Organic & Medicinal Chemistry*.
87. Praveen, R. V. S., Raju, A., Anjana, P., & Shibi, B. (2024, October). IoT and ML for Real-Time Vehicle Accident Detection Using Adaptive Random Forest. In *2024 Global Conference on Communications and Information Technologies (GCCIT)* (pp. 1-5). IEEE.
88. Sivakumar, S., Prakash, R., Srividhya, S., & Vikram, A. V. (2023). A novel analytical evaluation of the laboratory-measured mechanical properties of lightweight concrete. *Structural engineering and mechanics: An international journal*, *87*(3), 221-229.
89. Akat, G. B. (2021). EFFECT OF ATOMIC NUMBER AND MASS ATTENUATION COEFFICIENT IN Ni-Mn FERRITE SYSTEM. *MATERIAL SCIENCE*, *20*(06).
90. Farooq, S. M., Karukula, N. R., & Kumar, J. D. S. A Study on Cryptographic Algorithm and Key Identification Using Genetic Algorithm for Parallel Architectures. *International Advanced Research Journal in Science, Engineering and Technology ICRAESIT*, *2*.
91. Dehankar, S. P., & Dehankar, P. B. (2018). Experimental studies using different solvents to extract butter from Garcinia Indica Choisy seeds. *International Journal of New Technologies in Science and Engineering*, *5*(9), 113-117.
92. Rahman, Z., Mohan, A., & Priya, S. (2021). Electrokinetic remediation: An innovation for heavy metal contamination in the soil environment. *Materials Today: Proceedings*, *37*, 2730-2734.