MJI
Media Jurnal Informatika
Vol. 17, No. 2, December 2025
p-issn : 2088-2114, e-issn : 2477-2542

STRIDE-Based Threat Analysis and AI-Driven Dataset Design for Securing Educational E-Payment Systems

Doddy Ferdiansyah [a,1,*], Leony Lidya [a,2], Miftahul Fadli Muttaqin [a,3]

[a] Jurusan Teknik Informatika, Fakultas Teknik, Universitas Pasundan Bandung
[1] doddy@unpas.ac.id*; [2] leony.lidya@unpas.ac.id; [3] miftahulfadli@unpas.ac.id

ARTICLE INFO

Article history
Received 2025/11/18
Revised 2025/12/05
Accepted 2025/12/20

Keywords
Threat Modeling
STRIDE
e-payment
Application Security
Threat Dataset

ABSTRACT

The increasing adoption of electronic payment (e-payment) systems in educational settings introduces significant cybersecurity challenges. This study conducts a systematic security analysis of a web-based school canteen e-payment system using the STRIDE threat modeling framework. The methodology involves three stages: system modeling with a Data Flow Diagram (DFD), threat mapping across system components, and qualitative risk assessment based on potential impact and likelihood. The analysis identified six STRIDE threat categories, with high-risk findings in Tampering (balance and price manipulation), Spoofing (account takeover), and Denial of Service (flooding attacks). Recommended mitigation strategies include multi-factor authentication, strict server-side input validation, immutable logging, and secure session management. Beyond manual threat analysis, this research contributes by designing a structured threat dataset as a foundation for artificial intelligence (AI) integration. This dataset enables the development of AI models for automated threat classification, risk prediction, and adaptive mitigation recommendations. The findings highlight the importance of proactive and forward-looking security approaches while opening pathways for future research on data-driven security automation in educational digital infrastructures.

1. Introduction

The global wave of digital transformation has accelerated the adoption of electronic payment (e-payment) systems across various sectors, with educational environments emerging as a noteworthy area of expansion [1]. School canteens, as dynamic ecosystems of daily micro-transactions, stand to gain substantial benefits from the integration of such technologies [2]. Web-based e-payment systems offer undeniable advantages in terms of convenience, speed, and operational efficiency, reducing reliance on physical currency and simplifying financial reconciliation processes [3]. Previous studies have also developed web-based tuition payment recording systems integrated with SMS Gateway to improve payment management efficiency in schools [31].

However, behind these functional benefits lie significant and often underestimated cybersecurity risks that cannot be ignored. Systems designed to handle user data and financial transactions inherently become attractive targets for the evolving landscape of cybercrime [4]. Threats such as credential theft, illegal account balance manipulation, and fraudulent transactions can permanently undermine user trust and cause substantial financial losses [5]. Moreover, systems operating within school environments carry heightened ethical and legal responsibilities to protect user data, much of which belongs to vulnerable minors [6].

Consequently, proactive, in-depth, and systematic security analysis is not optional but an absolute necessity. Identifying potential vulnerabilities before they are exploited by malicious actors has proven to be more effective, efficient, and economical than reactive incident responses, which often entail broader impacts and higher remediation costs [7],[8]. In this context, artificial intelligence (AI) is beginning to play a pivotal

role in strengthening application security, particularly through the automation of threat identification, risk classification, and anomaly detection based on historical data patterns. Integrating AI into security analysis enables systems to become more adaptive and responsive to evolving threats.

This study aims to conduct a comprehensive security analysis of a web-based e-payment system used in school canteens. The analysis employs the STRIDE threat modeling framework, a structured methodology introduced by Microsoft to systematically identify and categorize security threats [9]. By modeling the system architecture and assessing each component against the six STRIDE threat categories, this study produces a structured map of potential vulnerabilities and associated threat scenarios. Based on these findings, it proposes practical and implementable mitigation strategies to strengthen the security posture of educational e-payment systems. In addition, this work outlines a conceptual direction for future AI-assisted security analytics through a structured threat dataset design. However, no AI model is implemented or empirically evaluated in this study.

2. Method

This study uses a qualitative case study approach focused on the security architecture of a web-based school canteen e-payment system. STRIDE threat modeling is applied through three phases: (1) system modeling using data flow diagrams, (2) threat identification by examining each process, data store, and data flow against the STRIDE categories, and (3) analysis and mitigation planning. To prioritize threats, each scenario is rated using a likelihood–impact scale and ranked using a relative risk score (L×I). The proposed mitigations are assessed qualitatively using three explicit criteria: (i) threat-to-control traceability, ensuring that each recommendation addresses a specific threat scenario and affected component; (ii) feasibility, considering implementation complexity and operational constraints in a school canteen environment; and (iii) expected risk reduction, describing whether a control is expected to reduce likelihood and/or impact for higher-priority threats.

2.1 System Modeling

The initial phase involved deconstructing the electronic payment system to understand and visualize its architecture and data flows. This was achieved by developing a Data Flow Diagram (DFD) [10]. The Level 0 DFD, also known as the Context Diagram, was first constructed to map the system's high-level interactions with external entities [11]. This diagram serves as a critical blueprint for accurate and comprehensive threat identification, illustrating the main processes, data flows between components, and data storage. Furthermore, the development of the Level 0 DFD provides an essential high-level overview of system boundaries and external interactions, forming the foundation for more detailed subsequent analysis. This initial visualization is crucial for establishing a shared understanding of the system's scope and operational context, ensuring that the subsequent threat modeling process is based on a clear and agreed conceptual model. Fig 1 illustrates the high-level interactions of the system with external entities. This diagram functions as a vital reference for accurate and comprehensive threat identification, depicting the main processes, data flows, and data storage.

The DFD was refined by explicitly marking the trust boundary of the e-payment system and labeling each communication channel (e.g., HTTPS/TLS and external API calls). This refinement highlights boundary-crossing data flows, which strengthens STRIDE-based threat identification. Subsequently, the Level 1 DFD further elaborates the internal processes, identifying specific sub-processes and their complex interconnections within the e-payment system [12]. Fig 2 presents the Level 1 Data Flow Diagram of the school canteen e-payment system.

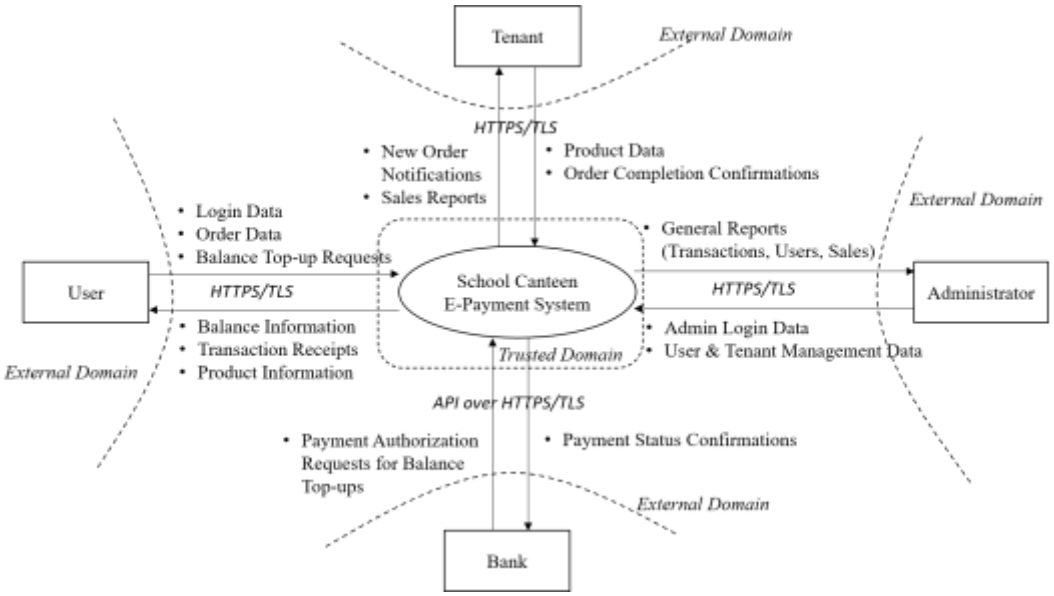

Fig 1. Level 0 DFD of the School Canteen E-Payment System

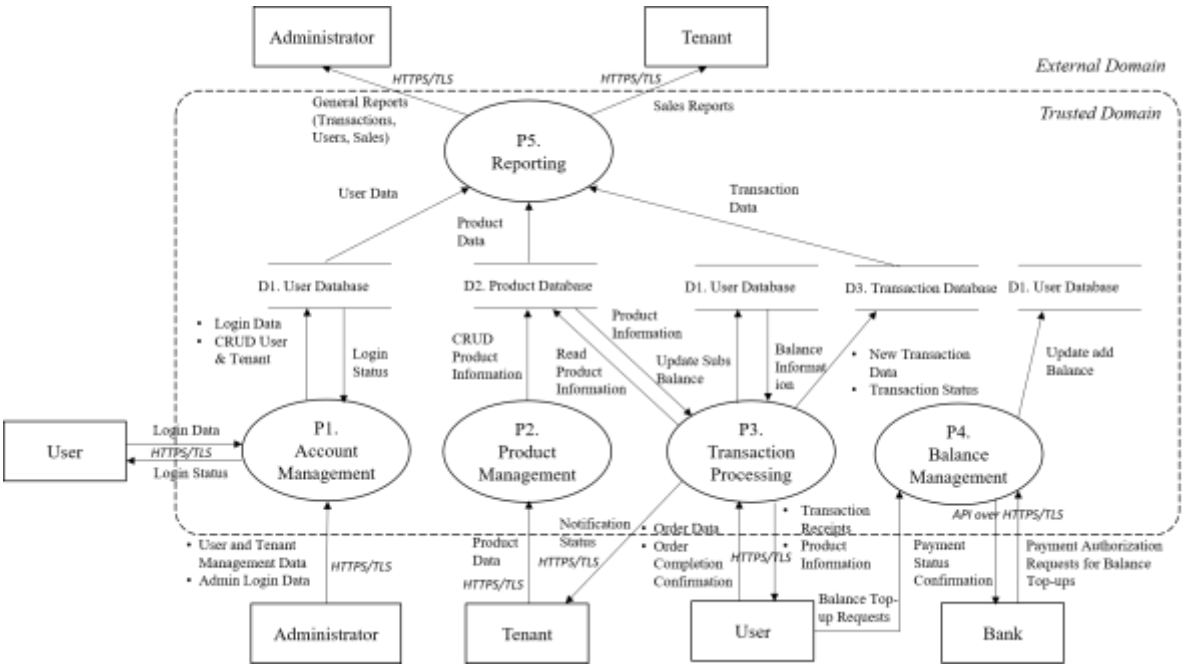

Fig 2. Level 1 DFD of the School Canteen E-Payment System

Fig 2. Level 1 DFD of the school canteen e-payment system with explicit trust boundaries and labeled communication channels (HTTPS/TLS for external access and API calls, internal DB connections for data store access). This diagram details the system's internal processes, outlining specific sub-processes and illustrating the intricate interconnections among components. Such decomposition enables a granular understanding of how data is processed, stored, and transmitted, which is fundamental for effective threat identification. This systematic modeling approach ensures that all potential vulnerability points are considered when applying the STRIDE framework, leading to a stronger and more comprehensive security analysis [13]. Moreover, the standardized DFD structure can also serve as an initial input for AI-based systems to automatically classify threats and support faster, more adaptive mitigation decision-making. By adopting this structured modeling approach, all potential vulnerability points are systematically addressed, thereby reinforcing the robustness and comprehensiveness of the security analysis [14].

2.2 Threat Identification

After the system was modeled in detail, the second phase focused on identifying potential threats inherent in each element of the DFD [15]. This identification process was strictly guided by the six threat categories of the STRIDE framework. Such a systematic approach ensures that all potential vulnerability points are considered, leading to a stronger and more comprehensive security analysis. Table 1 summarizes the threat findings based on the six STRIDE categories, mapping each type of threat to the most vulnerable system components and providing an overview of the application's security posture.

Table 1. System-Specific STRIDE Mapping for the School Canteen E-Payment System

STRIDE	Security Objective	DFD Elements (System-Specific)
S (Spoofing)	Authentication	P1 Account Management, Login data flow
T (Tampering)	Integrity	P3 Balance & Transaction Management, D3 Transaction DB, top-up data flow
R (Repudiation)	Non-repudiation	P3 Transaction Processing, P5 Reporting, D3 Transaction Log/DB
I (Information Disclosure)	Confidentiality	All external data flows (User/Admin/Tenant portals), Bank API flow
D (Denial of Service)	Availability	P1 Account Management, P2 Product Management, search/login endpoints
E (Elevation of Privilege)	Authorization	P1 Account Management, session handling, role management

1. Spoofing: The threat of illegally assuming the identity of another user or system component, potentially leading to unauthorized access. This category evaluates risks associated with identity forgery in the e-payment system [13].
2. Tampering: The threat of deliberately modifying data in transit or at rest, thereby compromising information integrity. This involves assessing vulnerabilities related to unauthorized data alteration [16].
3. Repudiation: The threat in which a user or entity denies performing critical actions without sufficient audit trails, thereby obscuring accountability. This category emphasizes the need for adequate non-repudiation mechanisms [17].
4. Information Disclosure: The threat of unauthorized exposure of sensitive or confidential information, constituting a violation of privacy and data security. This category critically examines how sensitive data, particularly that of minors and financial transactions, is protected from unauthorized access or leakage [13].
5. Denial of Service: The threat aimed at making the system or service unavailable to legitimate users, thereby disrupting operations. This involves assessing vulnerabilities that could lead to degradation or complete shutdown of the e-payment service, affecting the school's operational continuity [18].
6. Elevation of Privilege: The threat in which a user with limited access rights successfully obtains elevated permissions, creating opportunities for exploitation. This category examines vulnerabilities that may allow attackers to bypass security controls and gain higher levels of authority within the system [19].

Each DFD component was systematically and thoroughly analyzed against these six categories to produce a comprehensive inventory of exploitable vulnerabilities. This meticulous process ensures that no potential attack vectors are overlooked, providing a strong foundation for developing targeted mitigation strategies.

Furthermore, the results of this threat identification process open opportunities for the development of AI-based systems, where the classified threat patterns can be used as training datasets for models capable of automatic classification, risk prediction, and real-time anomaly detection. Thus, the STRIDE framework functions not only as a manual analysis tool but also as a foundation for building more adaptive and data-driven security systems. This detailed threat identification, combined with comprehensive system modeling,

forms the basis for prioritizing and addressing the most critical security risks in the school canteen e-payment system. Such granular analysis is crucial for developing robust security measures against the evolving spectrum of cyber threats in contactless transaction systems [20],[21].

2.3 Analysis and Mitigation Recommendations

In the final phase, each identified threat scenario is examined to understand how it could realistically affect the system, its users, and operational continuity. Threats are then prioritized using the likelihood, impact assessment described earlier, enabling higher-risk scenarios to receive attention first.

Mitigation recommendations are derived directly from the STRIDE findings and mapped to the affected DFD elements to ensure clear threat-to-control traceability. The proposed controls reflect widely adopted secure application practices, including authentication hardening, input validation, session security, access control, secure communications, and audit logging [13]. To strengthen practical relevance, each recommendation is also considered in terms of feasibility within a school canteen setting, such as implementation effort, operational overhead, and maintenance requirements.

The expected effectiveness of each mitigation is argued qualitatively by describing whether it primarily reduces likelihood (e.g., rate limiting, MFA, secure session handling) and/or reduces impact (e.g., encryption in transit, least-privilege permissions, tamper-evident logs), especially for higher-priority risks. Finally, the analysis highlights the value of periodic review through logging, monitoring, and repeated threat modeling to keep controls aligned with evolving attack patterns [22].

While this study structures the threat scenarios and mitigation mappings in a way that can support future data-driven security work, it does not implement or evaluate AI-based mechanisms. Any AI-assisted threat analytics remains a planned extension enabled by the structured outputs of this threat modeling process.

2.4 Mitigation Evaluation Criteria

The mitigation recommendations were evaluated using a qualitative validation approach rather than experimental deployment. First, traceability was used to confirm that each mitigation directly addresses the identified STRIDE threat and the corresponding DFD element. Second, feasibility was considered by assessing the practical effort required (configuration changes, development complexity, and administrative overhead) and whether the control is realistic for a school operational setting. Third, expected risk reduction was assessed by determining whether a control primarily reduces threat likelihood (e.g., rate limiting, MFA, secure session handling) and/or reduces impact (e.g., encryption, least-privilege access, immutable logs). These criteria provide a transparent rationale for why certain controls are prioritized for high-risk scenarios.

3. Results and Discussion

The application of the STRIDE threat modeling method to the DFD of the e-payment system successfully identified a series of significant security threats across various system components. This section outlines the findings, analyzes their potential impact in real-world scenarios, and formulates specific mitigation recommendations for each threat category. In addition, the classification results can serve as an initial foundation for training artificial intelligence (AI) models to automate the identification and prediction of application security risks.

3.1 Threat Identification Results

To ensure analytical clarity, the identified threats were mapped onto the six STRIDE categories, namely Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege. This taxonomy enables a comprehensive understanding of potential vulnerabilities across different dimensions of software security. To support prioritization beyond qualitative labels, this study applies a relative risk scoring approach based on Likelihood (L) and Impact (I). Each identified threat scenario is rated on a 1–5 scale for likelihood and impact, and the overall risk score is computed as Risk Score = L × I (range 1–25). The resulting scores are mapped into priority levels to guide mitigation focus (Low: 1–7, Medium: 8–14, High: 15–25). This scoring scheme provides transparent and reproducible prioritization while remaining suitable for qualitative threat modeling contexts. The detailed classification is summarized in Table 2.

Table 2. Threat Identification Using STRIDE

STRIDE Category	Affected DFD Component	Threat Scenario Description	Potential Impact	Likelihood (1–5)	Impact (1–5)	Risk Score	Priority
S (Spoofing – Identity Impersonation)	Process 1: Account Management (Login)	External actors attempt to log in as Users, Tenants, or Admins using brute-force techniques or leaked credential lists (credential stuffing)	Unauthorized account access, balance theft, data manipulation, account takeover.	4	4	16	High
	Data Flow: Login Credentials	Attackers intercept login data over unsecured networks (if unencrypted) to steal passwords and impersonate users.	Complete account compromise.	3	5	15	High
T (Tampering – Data Manipulation)	Process 2: Product Management	Dishonest tenants attempt to alter product prices in the product database after they are displayed to users.	Financial loss for users, unfair gain for tenants.	3	3	9	Medium
	Process 3: Balance & Transaction Management	Technically skilled users intercept top-up requests and alter values before data is sent to the system or bank API.	Illegal balance addition, financial loss for system operators.	3	5	15	High
	Data Store: User Balance DB	Compromised admins or attackers with database access directly modify user balances.	Large-scale balance manipulation, loss of financial data integrity.	2	5	10	Medium
R (Repudiation – Activity Denial)	Process 3: Balance & Transaction Management	Users deny legitimate purchases to request refunds; tenants deny receiving orders or payments.	Transaction disputes, financial losses, audit difficulties	3	3	9	Medium
	Data Store: Transaction Log DB	Compromised admins delete or alter transaction records to conceal fraudulent activity.	Loss of audit trails, invalid accountability.	2	4	8	Medium
I (Information Disclosure – Data Exposure)	Data Flow: All data flows (if unencrypted)	Attackers sniff network traffic to steal user personal data, transaction details, or product information.	Privacy violations, identity theft, exposure of sensitive data.	3	4	12	Medium
	Process 4: Reporting & Audit	Errors in reporting features reveal technical details or database queries (e.g., SQL injection errors), exposing system structure to attackers.	Attackers gain system architecture insights for advanced attacks.	3	3	9	Medium
	Data Flow: Bank API	Credit card or payment details exposed if communication with the bank API is not properly encrypted (e.g., outdated TLS).	Large-scale financial data leakage, PCI DSS compliance violations.	2	5	10	Medium

D (Denial of Service – Service Disruption)	Process 1: Account Management	Attackers flood the login page with thousands of requests per second, overloading the account database and preventing legitimate access.	Complete service outage, system inaccessible to all users.	4	4	16	High
	Process 2: Product Management	Attackers flood product search functions with complex queries, exhausting server resources and slowing the application.	System performance degradation, poor user experience.	3	3	9	Medium
E (Elevation of Privilege – Unauthorized Access Escalation)	Process 1: Account Management	Regular users exploit session management vulnerabilities (e.g., session hijacking) to take over an Admin session.	Complete system compromise, attackers gain full privileges to alter, steal, or delete data.	3	5	15	High
	Process 2: Product Management	Tenants exploit vulnerabilities to access or modify other tenants' product data.	Tenant privacy violations, business sabotage, unfair competition.	3	4	12	Medium

Table 2 comprehensively maps each threat type to the most vulnerable system components, providing a detailed overview of the application's security posture. Each threat category is explained with specific examples relevant to the school e-payment context, offering a deeper understanding of the risks faced.

The STRIDE classification in Table 2 provides a structured representation of threats and affected system elements. To improve reproducibility, the same results are subsequently organized into a dataset representation in Section 3.2, including a defined schema and illustrative instances in both tabular and JSON formats.

3.2 Threat Dataset Design for AI

The STRIDE-based threat identification results can be expressed as a structured dataset to improve reproducibility and to support future security analytics. In this study, the dataset design is presented as a structured representation of the threat modeling outputs, rather than evidence of an implemented AI solution. The dataset records each threat scenario together with the affected DFD element, risk prioritization attributes, and mapped mitigation controls. This representation enables consistent reuse across future studies that may explore data-driven threat classification or prioritization.

3.2.1 Dataset Schema

To make the dataset design explicit, Table 3 defines the proposed fields for each record. The schema is intentionally compact and aligned with the STRIDE mapping and the likelihood–impact risk scoring used for prioritization.

Table 3. Proposed STRIDE-Based Threat Dataset Schema

Field	Type	Description
record_id	string	Unique identifier for each threat record
stride_category	string	STRIDE label (S, T, R, I, D, E)
affected_component	string	Impacted DFD element (process, data flow, or data store)
component_type	string	Process / Data Flow / Data Store
threat_scenario	string	Short description of the threat scenario
potential_impact	string	Summary of consequences (security/financial/operational)
likelihood	integer	Likelihood score (1–5)

impact	integer	Impact score (1–5)
risk_score	integer	Computed as likelihood × impact
priority	string	Low / Medium / High based on risk score mapping
recommended_controls	array of strings	Mitigation controls mapped to the scenario

3.2.2 Sample Dataset Instances

To illustrate how the schema is populated, Table 4 provides sample records derived from the system-specific STRIDE scenarios. These instances are included to support reproducibility and to demonstrate how the threat modeling results can be translated into machine-readable records.

Table 4. Sample Threat Dataset Instances

record_id	stride_category	affected_component	component_type	likelihood	impact	risk_score	priority	recommended_controls
TR-001	S	Process 1: Account Management (Login)	Process	4	4	16	High	Rate limiting; MFA; account lockout; anomaly detection
TR-002	T	Process 3: Balance & Transaction Management	Process	3	5	15	High	Server-side validation; anti-replay token; request signing; TLS
TR-003	D	Process 1: Account Management	Process	4	4	16	High	WAF rules; CAPTCHA; throttling; resource limits

In addition to a tabular representation, the same records can be serialized into JSON to support interoperability across tools and pipelines. Fig 3 shows a simplified JSON example consistent with the schema above.


```
[
  {
    "record_id": "TR-001",
    "stride_category": "S",
    "affected_component": "Process 1: Account Management (Login)",
    "component_type": "Process",
    "threat_scenario": "Credential stuffing using leaked credential lists.",
    "potential_impact": "Unauthorized access, balance theft, and account takeover.",
    "likelihood": 4,
    "impact": 4,
    "risk_score": 16,
    "priority": "High",
    "recommended_controls": [
      "Rate limiting",
      "Multi-factor authentication (MFA)",
      "Account lockout policy",
      "Anomaly detection"
    ]
  }
]
```


Fig 3. Dataset in JSON Format

3.2.3 Future Data Collection Strategy and Dataset Growth

Additionally, the dataset design can be extended with attributes such as occurrence frequency, exploitation probability, or severity levels, enriching the information available for data-driven analysis. Thus, the threat identification results provide not only manual insights for developers and administrators but also a foundation for building adaptive, automated, and machine learning-based security systems. To clarify this design, the study presents two forms of representation:

1. Threat Dataset Table displaying organized data structures with key attributes (see Table 3).
2. Conceptual Flow Diagram illustrating the transformation of STRIDE results into a dataset, which is then used as input for AI models to generate automated threat classification and mitigation recommendations, as shown in Fig 4.

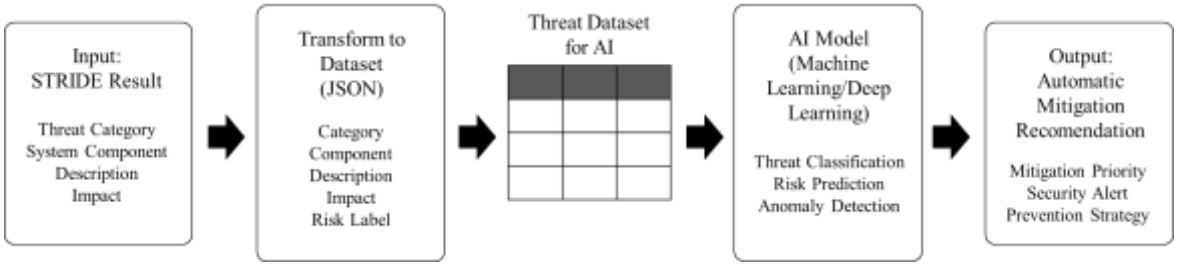

Fig 4. Transformation Flow from STRIDE to AI Model

Fig 4 demonstrates the transformation of STRIDE threat identification results into a structured dataset, which is subsequently used as input for AI models. This dataset enables automated threat classification, risk prediction, and anomaly detection, ultimately producing adaptive and data-driven security mitigation recommendations.

Finally, it is important to note that this paper does not implement or empirically evaluate AI mechanisms. The dataset design is provided to improve reproducibility and to strengthen the feasibility of future AI-oriented extensions based on the structured outputs of the threat modeling process.

3.3 In-Depth Analysis and Mitigation Strategies

This section provides a deeper analysis of each identified threat along with technical recommendations for mitigation, designed to strengthen the overall security posture of the system. The in-depth analysis also serves as manual validation of the threat dataset previously designed, enabling its use as training data for AI models to classify threats and recommend mitigation strategies automatically.

1. Spoofing (Identity Impersonation)
    a. Threat Scenario: Attackers may use automated tools to attempt thousands of username and password combinations (brute force) on the login page. Without restrictions, the system cannot distinguish between legitimate login attempts and automated attacks, allowing user accounts (students or staff) to be compromised.
    b. Mitigation Recommendation: Implement layered defenses. First, enforce strong password policies (minimum length, character combinations). Second, enable account lockout mechanisms to temporarily disable accounts after multiple failed login attempts. Third, require multi-factor authentication (MFA) for high-privilege accounts such as administrators.
    c. AI-Based Mitigation Potential: AI can detect abnormal login patterns (e.g., logins from multiple locations within a short timeframe).
2. Tampering (Data Modification)
    a. Threat Scenario: A technically skilled user may use proxy tools (e.g., Burp Suite) to intercept data sent from the browser to the server during a balance top-up. By modifying the

top-up value from $10 to $100 before reaching the server, the user could illegally gain credit if no server-side validation exists.

b. Mitigation Recommendation: Never trust client-side input. Enforce strict server-side validation for all critical parameters, including price, quantity, and top-up amounts. Additionally, record all financial transactions in append-only or immutable audit logs to prevent post-transaction manipulation.

c. AI-Based Mitigation Potential: AI can monitor transaction anomalies (e.g., unusually high top-up values).

3. Repudiation (Activity Denial)

a. Threat Scenario: A user completes a purchase but later claims the transaction never occurred to obtain a refund. Without detailed logs, administrators lack sufficient digital evidence to verify or refute the claim.

b. Mitigation Recommendation: Implement comprehensive and standardized logging mechanisms. Record all critical actions (login, logout, transactions, top-ups) with details such as timestamp, user ID, action performed, source IP, and status. Logs must be securely stored with restricted access.

c. AI-Based Mitigation Potential: AI can analyze transaction dispute patterns to detect fraudulent behavior.

4. Information disclosure

a. Threat Scenario: Invalid user input triggers error pages displaying technical details such as stack traces, framework versions, or database queries. Although seemingly minor, this information can help attackers map system technologies and plan targeted attacks.

b. Mitigation Recommendation: Configure applications to handle exceptions globally. Display generic error messages to end-users (e.g., “System error occurred”), while recording detailed error logs internally for developers and administrators.

c. AI-Based Mitigation Potential: AI can automatically scan error logs to detect information leakage.

5. Denial of service (DoS)

a. Threat Scenario: Attackers may create scripts to send hundreds of product search requests per second. The high request volume exhausts server resources (CPU and memory), causing severe performance degradation or complete denial of access for legitimate users.

b. Mitigation Recommendation: Apply rate limiting on vulnerable API endpoints (e.g., /login, /search) to restrict the number of requests per IP within a given timeframe. For larger-scale attacks, consider cloud-based DDoS protection services.

c. AI-Based Mitigation Potential: AI can recognize abnormal traffic patterns and trigger automated protection mechanisms.

6. Elevation of privilege

a. Threat Scenario: After an administrator logs in, the system generates a session token stored in browser cookies. If the token is predictable or insufficiently random, attackers may guess valid admin session tokens, gaining full access without knowing the password.

b. Mitigation Recommendation: Use secure session management libraries that generate long, cryptographically random session identifiers. Additionally, enforce strict Role-Based Access Control (RBAC) on the server side, ensuring that every request to administrative features revalidates the user’s role.

c. AI-Based Mitigation Potential: AI can detect user behavior deviating from normal access profiles

With AI integration, mitigation strategies are no longer static but can be dynamically updated through continuous learning from new threat data, ensuring that the system's security posture remains adaptive to evolving attack patterns.

Overall, the implementation of targeted and comprehensive mitigation strategies not only addresses identified vulnerabilities but also establishes a foundation for AI integration. By using the analysis results as validation datasets and leveraging AI's ability to detect anomalies, the school e-payment system can evolve into a resilient, adaptive, and sustainable platform capable of withstanding dynamic cybersecurity threats.

3.4 Discussion

The application of the STRIDE framework for threat modeling has comprehensively outlined the security vulnerabilities within the web-based school canteen e-payment system. This analysis systematically revealed critical threats across six categories: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege [23]. Each identified threat represents a plausible attack vector that could compromise the integrity, confidentiality, or availability of the system [24].

For instance, Spoofing threats, particularly brute-force attacks on login credentials, highlight the tangible risk of unauthorized account access. Similarly, Tampering demonstrates how client-side manipulation, if not strictly validated on the server side, can lead to financial discrepancies and fraudulent activities. Repudiation, where users can easily deny legitimate transactions, underscores the urgent need for strong and immutable audit trails. Information Disclosure risks, often stemming from improperly handled error messages, may inadvertently provide attackers with reconnaissance data, significantly aiding more targeted attacks. Vulnerabilities to Denial of Service attacks expose the system to potential unavailability, disrupting essential services for students and staff. Finally, Elevation of Privilege threats could allow unauthorized individuals to gain full administrative control, resulting in catastrophic system compromise.

Collectively, these findings emphasize that without proactive and integrated security design from the outset, even the most sophisticated systems remain vulnerable to evolving cyber threats [25],[26]. The proposed mitigation strategies, including strong multi-factor authentication [27], strict server-side input validation [28], immutable logging mechanisms [29], and secure session management [30], are essential for building a resilient security posture. Implementing these recommendations is not merely a technical requirement but a strategic necessity to safeguard sensitive financial data and maintain user trust.

Furthermore, the threat identification results converted into dataset form (see subsection 3.2) extend the contribution of this study toward AI integration. By leveraging the threat dataset as training data, AI models can be developed to:

1. Automatically classify threats based on STRIDE patterns.
2. Predict risk levels and potential impacts of threats.
3. Detect transaction or user behavior anomalies in real time.
4. Provide continuously updated adaptive mitigation recommendations.

This integration reinforces the notion that security systems are no longer static but dynamic and adaptive, capable of learning from historical data as well as emerging attack patterns. In the educational context, this is particularly relevant, as school e-payment systems involve sensitive student data that must be protected with the highest security standards.

Thus, this study not only delivers manual STRIDE-based analysis but also opens a strategic pathway toward data-driven security automation. The insights gained are highly valuable for developers, security professionals, and educational institutions seeking to implement secure, reliable, and sustainable e-payment solutions in the digital era.

3.5 Research Limitations

Although this study successfully identified and analyzed security threats using the STRIDE framework, several limitations must be explicitly acknowledged:

1. Limited Case Study Scope

   The analysis was conducted on a web-based school canteen e-payment system with a specific architecture. The findings may not be fully generalizable to other e-payment systems with larger scales or different architectures.

2. Qualitative Approach

   Risk assessment was performed qualitatively based on projected probability and impact. No quantitative testing or real attack simulations were conducted, meaning the accuracy of risk estimation remains dependent on conceptual assumptions

3. No Comparison with Other Frameworks

   This study focused exclusively on STRIDE without comparing its effectiveness to other threat modeling frameworks such as PASTA, OCTAVE, or LINDDUN. This limits the comparative perspective on the strengths and weaknesses of STRIDE.

4. Operational Data Limitations

   The analysis was based on system models and hypothetical threat scenarios. Empirical data from real security incidents or actual attack logs were not utilized, restricting validation against real-world conditions.

5. Focus on Technical Aspects

   The study emphasized technical threat mitigation. Non-technical factors such as organizational policies, user awareness, and legal regulations were not examined in depth, although they may significantly influence the effectiveness of mitigation strategies.

6. Conceptual AI Integration

   While this study designed a threat dataset as a foundation for AI integration, the application of AI models remains conceptual. The dataset was derived from manual STRIDE identification and does not yet include empirical data from real attacks or operational logs. This limits the ability of AI models to generalize and predict threats with high accuracy.

In conclusion, these limitations do not diminish the main contribution of the study in demonstrating the effectiveness of STRIDE as a proactive framework for threat analysis in educational e-payment systems. Rather, acknowledging these limitations opens opportunities for future research to expand the scope, integrate empirical data, compare different threat modeling approaches, and develop AI-based systems that are more adaptive and grounded in real-world data to produce more comprehensive mitigation recommendations.

4. Conclusion

This study successfully applied the STRIDE threat modeling method to analyze the security of a web-based school canteen e-payment system. The analysis effectively identified six critical categories of threats, ranging from identity spoofing to potential privilege escalation, distributed across multiple system components. These findings strongly emphasize that systems developed without adequate and integrated security considerations from the outset remain vulnerable to sophisticated cyberattacks, even if their functionality appears to operate effectively.

The STRIDE method has proven to be a systematic and comprehensive framework for proactively identifying vulnerabilities. The proposed mitigation strategies, such as strong multi-factor authentication, strict server-side input validation, and immutable logging mechanisms, provide crucial practical guidance for developers to significantly strengthen system defenses. Ultimately, security is not merely an additional feature to be added later but a fundamental foundation that must be integrated from the very beginning of the software

development lifecycle. This is particularly critical for applications managing sensitive financial data and user information within vulnerable educational environments.

In addition, this study contributes a STRIDE-based threat dataset design derived from the identified threat scenarios and mitigation mappings. This work does not implement or evaluate any AI-based mechanism, instead the dataset is proposed as a reusable foundation for future research on AI-assisted threat classification and prioritization in educational e-payment systems. Therefore, the primary contribution of this paper lies in the STRIDE-driven threat analysis and practical mitigation recommendations, while the AI direction remains a planned extension enabled by the proposed dataset structure. The dataset design includes fields for STRIDE category, affected DFD component, threat description, likelihood–impact scores, and mitigation controls.

Overall, this study highlights the importance of proactive, structured, and forward-looking security approaches. The integration of STRIDE with AI-based dataset design provides a foundation for future research to expand scope, incorporate empirical data, and develop resilient, adaptive, and sustainable security systems capable of addressing the evolving landscape of cyber threats.

Acknowledgment

This research is part of an internal grant from the Faculty of Engineering, Universitas Pasundan. We sincerely thank the faculty for their moral and financial support in facilitating this study. We also extend our gratitude to our research partner, PT Kunci Transformasi Digital, for supporting and providing the School Quality Management Application, which served as the object of this research.

Declarations

Author contribution. All authors actively contributed to every stage of this research, including methodology design, data collection and analysis, and manuscript preparation. We collectively take responsibility for the content of this article and approve the submitted version.
Funding statement. This research received funding support from LPPM, Faculty of Engineering, Universitas Pasundan.
Conflict of interest. The authors declare that there is no conflict of interest in this study.
Additional information. No additional information is available for this article.

Data and Software Availability Statements

The school e-payment application analyzed in this study is a module of the School Quality Management System developed by PT Kunci Transformasi Digital, our research partner. The system is used internally by schools and is not publicly available. However, a demo version of the application can be accessed through the official link: https://demosmk.sekolah.kunci.co.id/login.

References

[1] C.-Y. Chen, C. Fu, Y.-C. Hsu, and C.-Y. Lu, "A study on enterprises based on information security education and training to improve continuous information security governance," *Education and Awareness of Sustainability*. p. 59, Nov. 16, 2020. [Online]. Available: https://www.semanticscholar.org/paper/9c541d10932e7c9f152e84c29a316027f9c53955

[2] L. Lissa'idah, M. A. Rosid, and A. S. Fitrani, "Web-based canteen payment system with RFID technology," *Journal of Physics Conference Series*, vol. 1232, no. 1, p. 12028, Sep. 2019, doi: 10.1088/1742-6596/1232/1/012028.

[3] C. Ritthitraiphop and W. Hamra, "Organization and Parental Perceptions of Electronic Payments by Selected Seventh-day Adventist (SDA) International Schools in Thailand," in *Abstract Proceedings International Scholars Conference*, Dec. 2019, p. 1143. doi: 10.35974/isc.v7i1.1017.

[4] A. T. Oyewole, C. C. Okoye, O. C. Ofodile, and C. E. Ugochukwu, "Cybersecurity risks in online banking: A detailed review and preventive strategies applicatio," *World Journal of Advanced Research and Reviews*, vol. 21, no. 3. GSC Online Press, p. 625, Mar. 11, 2024. doi: 10.30574/wjarr.2024.21.3.0707.

[5] K. Sathupadi, S. Achar, S. Bhaskaran, N. Faruqui, and J. Uddin, "BankNet: Real-Time Big Data Analytics for Secure Internet Banking," *Big Data and Cognitive Computing*, vol. 9, no. 2, p. 24, Jan. 2025, doi: 10.3390/bdcc9020024.

[6] I. Goran, "Cyber Security Risks in Public High Schools," Jan. 2017, Accessed: Aug. 2025. [Online]. Available: https://academicworks.cuny.edu/cgi/viewcontent.cgi?article=1002&context=jj_etds

[7] R. Baskerville, P. Spagnoletti, and J. Kim, "Incident-centered information security: Managing a strategic balance between prevention and response," *Information & Management*, vol. 51, no. 1, p. 138, Nov. 2013, doi: 10.1016/j.im.2013.11.004.

[8] S. Al-Azzani, A. Al-Natour, and R. Bahsoon, "Architecture-Centric Testing for Security," in *Elsevier eBooks*, Elsevier BV, 2013, p. 245. doi: 10.1016/b978-0-12-407772-0.00009-5.

[9] A. Jawad, J. Jaskolka, A. Matrawy, and M. Ibnkahla, "strideSEA: A STRIDE-centric Security Evaluation Approach," 2025, doi: 10.48550/ARXIV.2503.19030.

[10] A. Aljaradat, G. Sarkar, and S. K. Shukla, "Modelling cybersecurity impacts on digital payment adoption: A game theoretic approach," *Journal of Economic Criminology*, vol. 5, p. 100089, Aug. 2024, doi: 10.1016/j.jeconc.2024.100089.

[11] W. Hasselbring, M. Wojcieszak, and S. Dustdar, "Control Flow Versus Data Flow in Distributed Systems Integration: Revival of Flow-Based Programming for the Industrial Internet of Things," *IEEE Internet Computing*, vol. 25, no. 4, p. 5, Jul. 2021, doi: 10.1109/mic.2021.3053712.

[12] M. N. Johnstone, "Threat Modelling with Stride and UML," Jan. 2010, doi: 10.4225/75/57b670493477c.

[13] M. Girdhar, "Advanced Cybersecurity Strategies for Cyber-Physical Systems: Case Studies in EV Charging Stations, Connected & Automated Vehicles, and Digital Substations," *Deep Blue (University of Michigan)*, Jan. 2025, Accessed: Jul. 2025. [Online]. Available: https://hdl.handle.net/2027.42/196335

[14] H. Guan, W. R. Chen, L. Han, and J. Wang, "STRIDE – Based Risk Assessment for Web Application," *Applied Mechanics and Materials*, p. 1323, Jun. 2011, doi: 10.4028/www.scientific.net/amm.58-60.1323.

[15] P. K. Yeng, D. Stephen, and B. Yang, "Comparative Analysis of Threat Modeling Methods for Cloud Computing towards Healthcare Security Practice," *International Journal of Advanced Computer Science and Applications*, vol. 11, no. 11, Jan. 2020, doi: 10.14569/ijacsa.2020.0111194.

[16] Z. Bokolo and O. Daramola, "Elicitation of security threats and vulnerabilities in Insurance chatbots using STRIDE," *Scientific Reports*, vol. 14, no. 1, Aug. 2024, doi: 10.1038/s41598-024-68791-z.

[17] F. T. Chimuco, J. B. F. Sequeiros, T. M. C. Simões, M. M. Freire, and P. R. M. Inácio, "Expediting the design and development of secure cloud-based mobile apps," *International Journal of Information Security*, vol. 23, no. 4, p. 3043, Jul. 2024, doi: 10.1007/s10207-024-00880-6.

[18] M. Ramachandran, "Software security requirements management as an emerging cloud computing service," *International Journal of Information Management*, vol. 36, no. 4, p. 580, Apr. 2016, doi: 10.1016/j.ijinfomgt.2016.03.008.

[19] M. Ouaissa and M. Ouaissa, "Analyzing and Mitigating Attacks in IoT Smart Home Using a Threat Modeling Approach-Based STRIDE," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 19, no. 2, p. 126, Jan. 2025, doi: 10.3991/ijim.v19i02.52377.

[20] I. T. Moon, M. Shamsuzzaman, M. M. R. Mridha, and A. S. Md. M. Rahaman, "Towards the Advancement of Cashless Transaction: A Security Analysis of Electronic Payment Systems," *Journal of Computer and Communications*, vol. 10, no. 7, p. 103, Jan. 2022, doi: 10.4236/jcc.2022.107007.

[21] D. Angermeier, H. Wester, K. Beilke, G. Hansch, and J. Eichler, "Security Risk Assessments: Modeling and Risk Level Propagation," *ACM Transactions on Cyber-Physical Systems*, vol. 7, no. 1, p. 1, Nov. 2022, doi: 10.1145/3569458.

[22] S. B. Tete, "Threat Modelling and Risk Analysis for Large Language Model (LLM)-Powered Applications," *arXiv (Cornell University)*, Jun. 2024, doi: 10.48550/arxiv.2406.11007.

[23] S. Hart, A. Margheri, F. Paci, and V. Sassone, "Riskio: A Serious Game for Cyber Security Awareness and Education," *Computers & Security*, vol. 95, p. 101827, Apr. 2020, doi: 10.1016/j.cose.2020.101827.

[24] W. Ni, A. Asheralieva, J. Kang, Z. Xiong, C. Maple, and X. Wei, "An Enhanced Block Validation Framework With Efficient Consensus for Secure Consortium Blockchains," *IEEE Transactions on Services Computing*, vol. 17, no. 2, p. 420, Dec. 2023, doi: 10.1109/tsc.2023.3343839.

[25] J. Wetzels, D. D. Santos, and M. Ghafari, "Insecure by Design in the Backbone of Critical Infrastructure," p. 7, May 2023, doi: 10.1145/3576914.3587485.

[26] G. M. Makrakis, C. Kolias, G. Kambourakis, C. Rieger, and J. Benjamin, "Vulnerabilities and Attacks Against Industrial Control Systems and Critical Infrastructures," *arXiv (Cornell University)*, Jan. 2021, doi: 10.48550/arxiv.2109.03945.

[27] G. Sargsyan, N. Castellon, R. Binnendijk, and P. Cozijnsen, "Blockchain Security by Design Framework for Trust and Adoption in IoT Environment," p. 15, Jul. 2019, doi: 10.1109/services.2019.00018.

[28] M. Tahmasebi, "Beyond Defense: Proactive Approaches to Disaster Recovery and Threat Intelligence in Modern Enterprises," *Journal of Information Security*, vol. 15, no. 2, p. 106, Jan. 2024, doi: 10.4236/jis.2024.152008.

[29] S. Ghasemshirazi, G. Shirvani, and M. A. Alipour, "Zero Trust: Applications, Challenges, and Opportunities," *arXiv (Cornell University)*, Jan. 2023, doi: 10.48550/arxiv.2309.03582.

[30] C. C. Nwoye, "Next-Generation Protection Protocols and Procedures for Securing Critical Infrastructure," *International Journal of Research Publication and Reviews*, vol. 5, no. 11, p. 4830, Nov. 2024, doi: 10.55248/gengpi.5.1124.3328.

[31] M. Zaqy, G. Galih, M. I. Hermanto, "SPP Payment Recording Information System and SMS Gateway Using the Waterfall Method," Media Jurnal Informatika, Vol 17, No 1a, 2025, doi: 10.35194/mji.v17i1a.5771.