



Quality Evaluation of an Academic Information System Using Decision Table Testing and State Transition Testing

Fajar Mahardika^{1*}, Kukuh Muhammad², Angga Maulana Purba³, Nur Muniroh⁴

¹⁻²Department of Computer and Business, Informatics Engineering Program, Cilacap State Polytechnic,
Indonesia

³⁻⁴Department of Computer and Business, Multimedia Engineering Technology, Cilacap State Polytechnic,
Indonesia

Correspondence E-mail: fajarmahardika@pnc.ac.id

ABSTRACT

An Academic Information System plays a crucial role in managing course registration, grading, and student data administration, so its quality must be ensured. This study evaluates the quality of an Academic Information System using Decision Table Testing and State Transition Testing as functional testing approaches, focusing on the login and session management modules. Decision Table Testing verifies system behavior based on combinations of input conditions and business rules, while State Transition Testing analyzes system responses across different states and transitions. A total of 8 test cases designed using Decision Table Testing revealed 1 defect (12.5%) related to improper validation of inactive account status. Meanwhile, 5 test cases using State Transition Testing identified 1 defect (20%) in handling logout actions from a logged-out state. Overall, the system achieved a functional success rate of 84.6%, indicating a high level of reliability. These results show that Decision Table Testing is effective for detecting rule-based errors, while State Transition Testing is suitable for identifying weaknesses in state management. Combining both methods provides a comprehensive evaluation approach that improves the reliability and functional quality of the system.

ARTICLE INFO

Article History:

Submitted/Received 07 Apr 2026

First Revised 12 Oct 2026

Accepted 21 Oct 2026

First Available online 22 Oct 2026

Publication Date 22 Oct 2026

Keyword:

Academic Information System,
Software Quality Evaluation,
Decision Table Testing, State
Transition Testing, Functional
Testing

1. INTRODUCTION

The rapid advancement of information technology has significantly transformed the education sector, particularly through the implementation of Academic Information Systems. These systems support essential academic processes such as course registration, grading, and student data management. However, as system complexity increases, ensuring software quality becomes a major challenge. Inadequate testing processes may lead to functional errors, inconsistent data processing, and unreliable system performance, which can ultimately disrupt academic activities and decision-making processes [1]. Despite the widespread use of Academic Information Systems, many institutions still face issues related to improper implementation of business rules and incorrect handling of system states. For example, errors may occur in validating course registration requirements or in managing transitions between academic statuses, such as active[2][3], inactive, or graduated. These issues are often caused by insufficient or inappropriate testing methods. Conventional testing approaches may not effectively capture complex rule combinations or state-dependent behaviors, resulting in undetected defects within the system[4][5].

Previous studies have highlighted the importance of functional testing in improving software quality. Decision Table Testing has been widely used to evaluate systems with complex business rules by systematically analyzing combinations of input conditions and expected outcomes. Meanwhile, State Transition Testing has proven effective in identifying defects related to system state changes and transitions. However, most existing research applies these methods separately[6][7], focusing on either rule-based validation or state-based behavior analysis. Limited studies have explored the combined application of both methods to achieve a more comprehensive evaluation of system quality, particularly in the context of Academic Information Systems, as summarized in **Table 1**. Although Decision Table Testing and State Transition Testing have been individually applied in software testing, there is a lack of integrated studies that combine both methods to evaluate[8][9]. Academic Information Systems comprehensively. Most prior research either emphasizes rule-based testing or state-based testing alone, leaving potential defects undetected when systems involve both complex business rules and multiple state-dependent processes[10][7]. Furthermore, quantitative data on defect detection rates using the combined approach in academic settings is still limited, making it difficult to assess its effectiveness and practical benefits.

Given the critical role of Academic Information Systems in supporting institutional operations, ensuring their quality and reliability is essential. The lack of comprehensive testing approaches may result in undetected errors that affect system functionality and user trust. Therefore, this research is necessary to provide a structured and integrated testing approach by combining Decision Table Testing and State Transition Testing[11][12]. The findings of this study are expected to contribute to improving software testing practices and enhancing the overall quality of Academic Information Systems, particularly in handling complex rules and state-dependent processes[10].

Table 1. State of the Art of Research on Academic System Testing Using Decision Table Testing and State Transition Testing

No	Researcher / Year	Research Focus	Testing Method	Key Findings	Limitation / Research Gap
1	[13]	Student management system testing	Decision Table Testing	Detected errors in business rule combinations	Only focused on rule-based testing; did not test student status transitions.
2	[14]	Online registration system	State Transition Testing	Identified errors in status transitions (active/inactive)	Did not test input condition combinations, so rule-based errors were not detected
3	[15]	Academic system quality evaluation	Decision Table & State Transition (separately)	Validated functionality of specific modules	Methods applied separately; no integrated approach for comprehensive evaluation
4	[16]	Web-based campus information system	Black Box & White Box Testing	Found bugs in login and course registration modules	Less focus on complex business rules and state-dependent processes
5	This Study (2026)	Academic Information System	Combined Decision Table & State Transition Testing	Comprehensive quality evaluation, detecting both rule-based and state-based errors	Fills previous research gaps by integrating both methods and providing quantitative defect data

2. METHODS

2.1. Research Type

This study employs a quantitative descriptive approach aimed at evaluating the quality of an Academic Information System in an objective and measurable manner. The descriptive approach is used to explain the system's condition based on testing results, while the quantitative approach is used to convert testing outcomes into numerical values, particularly in the form of success percentages[17][18].

The testing method applied in this study is black-box testing, which focuses on evaluating system functionality without examining the internal code structure. Two main techniques are utilized: Decision Table Testing, which evaluates logical combinations of input conditions, and State Transition Testing, which analyzes system behavior based on changes in system states triggered by specific events.

2.2. Research Object

The object of this study is an Academic Information System (AIS) used to support academic activities within an educational institution. The system consists of several main modules [19][20] to be tested, including:

- a. Authentication module (login and logout)
- b. Course registration module (Study Plan / KRS)
- c. Academic results module (KHS)
- d. Course scheduling module
- e. Prerequisite validation module

The evaluation focuses on these core functionalities to ensure that the system operates in accordance with user requirements and academic regulations.

2.3. Data Collection Techniques

Data collection in this study is conducted using the following methods:

- a. Observation

Observation involves directly examining how users interact with the system. This aims to understand system workflows, user interactions, and identify potential issues within the system features [21][22].

- b. Documentation

Documentation involves collecting supporting materials such as:

1. Use case diagrams
2. Activity diagrams
3. Standard Operating Procedures (SOPs)
4. System requirement documents

2.4. System Testing

The system testing process is carried out through the following steps:

1. Identifying the system modules to be tested
2. Defining testing scenarios based on system requirements
3. Designing test cases
4. Conducting testing using Decision Table Testing and State Transition Testing
5. Recording test results
6. Analyzing the testing outcomes

2.5. Decision Table Testing

Decision Table Testing is a technique used to evaluate system behavior based on various combinations of input conditions and corresponding system actions [23].

Steps:

1. Identify conditions

Example:

- C1: Valid username
- C2: Valid password
- C3: Active account status

2. Identify actions

- A1: Login successful
- A2: Error message displayed

3. Construct a decision table

4. Determine the number of rules (test cases) using the formula:

$$R = 2^n$$

Where:

- R = number of rules (test cases)
 - n = number of conditions
5. Generate test cases based on the rules
 6. Execute testing
 7. Record the results

2.6. State Transition Testing

State Transition Testing is used to evaluate system behavior based on changes in system states caused by specific events.

Steps:

1. Identify system states

Example:

- S1: Logged Out
 - S2: Logged In
 - S3: Session Expired
2. Identify events
 - E1: Login input
 - E2: Logout
 - E3: Timeout
 3. Create a state transition diagram
 4. Develop a state transition table
 5. Determine the number of possible transitions using the formula:

$$T = S \times E$$

Where:

- T = number of possible transitions
 - S = number of states
 - E = number of events
6. Design test cases for:
 - Valid transitions
 - Invalid transitions
 7. Execute testing and record results

2.7. Data Analysis Techniques

Data analysis is conducted by comparing actual test results with expected outcomes for each test case.

1. Success Rate

$$\text{Success Rate(\%)} = \frac{TC_{\text{success}}}{TC_{\text{total}}} \times 100\%$$

Where:

- TC_{success} = number of successful test cases
- TC_{total} = total number of test cases

2.8. Failure Rate

$$\text{Failure Rate}(\%) = \frac{TC_{\text{failure}}}{TC_{\text{total}}} \times 100\%$$

2.9. System Quality Classification

The success rate is then classified into four quality categories, as presented in **Table 2**.

Table 2. System Quality Classification

Percentage	Category
81% – 100%	Very Good
61% – 80%	Good
41% – 60%	Fair
≤ 40%	Poor

3. RESULTS AND DISCUSSION

3.1. Results

1. Testing Results

This section presents the results of system testing conducted using Decision Table Testing and State Transition Testing on the Academic Information System (AIS). The testing focuses on evaluating key modules such as login, course registration (KRS), and logout processes to ensure that the system operates according to its functional requirements and can properly handle various input conditions and user actions.

a. Decision Table Testing Results

Decision Table Testing was applied to evaluate logical combinations of input conditions in the login module. The identified conditions include username validity, password validity, and account status. Based on these conditions, a decision table was constructed to represent all possible combinations and expected system responses, as shown in **Table 3**.

Table 3. Decision Table for Login Module

Rule	Username Valid	Password Valid	Account Active	Expected Result	Actual Result	Status
R1	Yes	Yes	Yes	Login Successful	Login Successful	Pass
R2	Yes	No	Yes	Error Message	Error Message	Pass
R3	No	Yes	Yes	Error Message	Error Message	Pass
R4	No	No	Yes	Error Message	Error Message	Pass

R5	Yes	Yes	No	Access Denied	Login Successful	Fail
R6	Yes	No	No	Error Message	Error Message	Pass
R7	No	Yes	No	Error Message	Error Message	Pass
R8	No	No	No	Error Message	Error Message	Pass

Based on three conditions, the number of possible test cases was calculated using the formula $R = 2^n = 2^3 = 8$. All test cases were executed, resulting in 7 successful cases and 1 failed case. The failure occurred in Rule 5, indicating that the system incorrectly allowed login for an inactive account, which reflects a weakness in access control validation.

b. State Transition Testing Results

State Transition Testing was conducted to analyze system behavior based on transitions between states triggered by specific events such as login, logout, and session timeout. The identified states include Logged Out, Logged In, and Session Expired, while the events include login, logout, and timeout, with the results summarized in **Table 4**.

Table 4. State Transition Testing Results

Current State	Event	Next State	Expected Result	Actual Result	Status
Logged Out	Login	Logged In	Access Granted	Access Granted	Pass
Logged In	Logout	Logged Out	Session Ended	Session Ended	Pass
Logged In	Timeout	Session Expired	Auto Logout	Auto Logout	Pass
Expired	Login	Logged In	Re-login Required	Re-login Required	Pass
Logged Out	Logout	Logged Out	No Action	Error Occurred	Fail

The total number of possible transitions was calculated using the formula $T = S \times E = 3 \times 3 = 9$. However, only 5 relevant transitions were tested. The results show that 4 transitions were successful, while 1 failed. The failure indicates that the system does not properly handle invalid transitions, particularly when a logout action is performed in a logged-out state.

2. Data Analysis

a. Success Rate Calculation

A summary of the test results for both methods is presented in **Table 5**.

Table 5. Summary of Test Results

Testing Method	Total Test Cases	Successful	Failed
Decision Table Testing	8	7	1
State Transition Testing	5	4	1
Total	13	11	2

The success rate is calculated using the formula:

$$\text{Success Rate} = \frac{11}{13} \times 100\% = 84.6\%$$

This result indicates that most of the system functionalities operate correctly.

b. Failure Rate Calculation

$$\text{Failure Rate} = \frac{2}{13} \times 100\% = 15.4\%$$

This shows that a small portion of the system still experiences functional issues.

c. Defect Density

$$\text{Defect Density} = \frac{2}{13} = 0.15$$

This value indicates a relatively low number of defects per test case.

3. Quality Interpretation

Based on the success rate of 84.6%, the system falls into the “Very Good” category (81%–100%). This indicates that the Academic Information System has a high level of functional reliability. However, the existence of failed test cases suggests that improvements are still needed, particularly in input validation and state transition handling, to further enhance system robustness and reliability.

3.2. Discussion

The results of this study demonstrate that the application of Decision Table Testing is highly effective in identifying logical errors arising from various combinations of input conditions, particularly within the login module of the Academic Information System. The testing process revealed a critical issue related to inactive account validation, where the system incorrectly granted access despite the account status being inactive. This finding indicates that the system does not yet implement adequate access control mechanisms, especially in enforcing account status constraints during authentication processes. Such a weakness may lead to potential security risks and unauthorized access if not properly addressed.

Furthermore, the implementation of State Transition Testing proved to be effective in analyzing system behavior based on changes in user states and system events. The results identified a weakness in handling invalid state transitions, specifically when a logout action

was performed while the system was already in a logged-out state. Ideally, the system should reject or ignore such invalid transitions; however, the observed behavior suggests that the system lacks proper validation of session states prior to executing user commands. This indicates a deficiency in session management logic, which is crucial for maintaining system consistency and preventing unexpected behavior.

Overall, the system demonstrates a relatively high level of functional reliability, as indicated by a success rate exceeding 80%, which falls into the “Very Good” category. This suggests that most system functionalities operate in accordance with their expected behavior. Nevertheless, the presence of failed test cases highlights several areas that require further improvement. These include strengthening input validation mechanisms to ensure all user inputs are properly verified, enhancing session management handling to accurately control user states and transitions, and improving error handling procedures to effectively manage invalid or unexpected user actions.

In light of these findings, several recommendations are proposed to improve the overall quality of the system. First, stricter validation rules should be implemented to ensure that inactive accounts are denied access under all conditions. Second, the system’s session handling logic should be refined to accurately track and validate user states before executing any operation. Third, additional validation should be incorporated to properly handle invalid state transitions, thereby preventing inconsistent system behavior. Finally, periodic system testing should be conducted to continuously monitor system performance, detect potential issues early, and maintain the overall quality and reliability of the Academic Information System over time.

4. CONCLUSION

The Academic Information System (AIS) plays a critical role in managing essential academic activities, including login authentication, course registration, grading, and student data management. Ensuring the system’s quality and reliability is vital for maintaining accurate and efficient academic operations. This study evaluated the quality of the AIS using Decision Table Testing and State Transition Testing, focusing on key modules such as login and session management. A total of 13 test cases were executed, comprising 8 cases using Decision Table Testing and 5 cases using State Transition Testing. The testing identified 2 defects—1 related to improper validation of inactive accounts and 1 associated with invalid state transitions during logout—resulting in an overall functional success rate of 84.6%. The findings demonstrate that Decision Table Testing is effective in detecting errors related to logical and rule-based processing, while State Transition Testing is suitable for uncovering issues in state-dependent processes. The combined use of these methods provides a comprehensive evaluation of system behavior, effectively identifying both rule-based and state-based defects that may otherwise remain undetected. In conclusion, integrating Decision Table Testing and State Transition Testing significantly enhances the functional quality, reliability, and overall performance of the AIS. It ensures compliance with expected system behavior and supports academic institutions in delivering consistent, accurate, and error-free services. Future AIS implementations are recommended to adopt combined functional testing strategies and incorporate systematic defect tracking to further improve system quality and user satisfaction.

5. ACKNOWLEDGMENT

The author would like to express sincere gratitude to all parties who have contributed to this research. The author also appreciates the institutions and laboratories that provided the

facilities and resources supporting the research activities. Thanks are also extended to colleagues, research assistants, and all participants who contributed through their assistance, feedback, and collaboration during data collection, testing, and analysis. This acknowledgment reflects the collective efforts of all parties that made the completion of this research possible and is hoped to provide benefits for the advancement of scientific knowledge.

6. AUTHORS' NOTE

The authors declare that there is no conflict of interest regarding the publication of this article. All data presented are accurate, obtained in accordance with ethical research practices, and all sources have been properly cited. No unethical procedures were used, and all participants, if involved, provided informed consent. This statement ensures transparency, academic integrity, and the credibility of the research findings.

7. REFERENCES

- [1] Z. Ugray, B. D.-J. of I. S. Education, and undefined 2022, "Quality assurance of learning assessments in large information systems and decision analysis courses," *aisel.aisnet.org*, Accessed: Apr. 06, 2026. [Online]. Available: <https://aisel.aisnet.org/jise/vol33/iss4/7/>
- [2] V. Gupta, P. Kaur, H. S.-I. Access, and undefined 2024, "Bi-Level Decision Tree Approach for Web Quality Assessment," *ieeexplore.ieee.org*, Accessed: Apr. 06, 2026. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/10670374/>
- [3] A. Nugroho, J. Purwanto, M. A. Muin, and F. Mahardika, "UI / UX Design of a Web-Based Student Organizations System Using the Design Thinking Method Approach," vol. 7, no. 1, pp. 24–38, 2025.
- [4] S. Valenti, A. Cucchiarelli, M. P.-J. of I. Technology, and undefined 2002, "Computer based assessment systems evaluation via the ISO9126 quality model," *jite.org*, vol. 1, no. 3, 2002, Accessed: Apr. 06, 2026. [Online]. Available: <https://jite.org/documents/Vol1/v1n3p157-175.pdf>
- [5] A. Al Nawaiseh, Y. Helmy, E. K.-I. J. S. T. Res, and undefined 2020, "Software testing with the approach of blackbox testing on the academic information system," *ijistech.org*, Accessed: Apr. 06, 2026. [Online]. Available: <http://ijistech.org/ijistech/index.php/ijistech/article/view/54>
- [6] Z. Khaerunnisa, K. Muhammad, and F. Mahardika, "Indonesian Journal of Digital Business Optimization of Cloud-Based Digital Archiving System Using Golang and the ICONIX Process," vol. 5, no. April, pp. 87–96, 2025.
- [7] S. Pradhan, M. Ray, M. Jena, M. Tripathy, and I. Panda, "Adopting state transition testing techniques in scenario-based test case generation," *Springer*, 2025, doi: 10.1007/S13198-025-03047-1.
- [8] K. Bollaerts, E. Ledent, T. de Smedt, D. W.- Vaccine, and undefined 2020, "ADVANCE system testing: Benefit-risk analysis of a marketed vaccine using multi-criteria decision analysis and individual-level state transition modelling," *Elsevier*, Accessed: Apr. 06, 2026. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0264410X19312435>
- [9] F. Mahardika, A. R. Naufal, and M. AL AMIN, "Desain UI dan UX dalam Sistem Informasi Akademik Menggunakan Metode Extreme Programming," *Progresif J. Ilm. Komput.*, vol. 19, no. 1, pp. 105–116, Feb. 2023, doi: 10.35889/PROGRESIF.V19I1.1023.
- [10] S. Pradhan, M. Ray, S. S.-J. of K. S. University-Computer, and undefined 2022,

- “Transition coverage based test case generation from state chart diagram,” *Elsevier*, Accessed: Apr. 06, 2026. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1319157818311078>
- [11] H. Jayusman and F. Mahardika, “Mobile-Based Event Decoration Ordering System Using UAT Method with PIECES Framework,” *J. Innov. Inf. Technol. Appl.*, vol. 7, pp. 1623–172, 2024.
- [12] F. Mahardika, R. B. B. Sumatri, T. Informatika, J. Komputer, and P. N. Cilacap, “IMPLEMENTASI PENDEKATAN METODE DESIGN THINKING SISTEM PERSURATAN ONLINE BERBASIS,” vol. 13, no. 2, 2025.
- [13] P. A.-F. exacta and undefined 2018, “Perancangan Sistem Informasi Akademik Online Menggunakan Black Box Testing Pada Sekolah Tinggi Ilmu Ekonomi Surya Nusantara,” *jurnal.unai.edu*, vol. 2, no. 1, pp. 41–52, 2023, Accessed: Apr. 06, 2026. [Online]. Available: <https://jurnal.unai.edu/teika/article/view/664>
- [14] S. Nafisah, S. R.-I. J. of D. Studies, and undefined 2021, “Pengembangan Sistem Informasi Kemajuan Akademik Menggunakan Model Incremental Berbasis Evaluasi Usability Dan White Box Testing,” *ejournal.instiki.ac.id*, vol. 2, no. 1, pp. 41–52, 2023, Accessed: Apr. 06, 2026. [Online]. Available: <http://ejournal.instiki.ac.id/index.php/sintechjournal/article/view/661>
- [15] A. Setiawan, S. Samsugi, D. A.-J. I. Dan, and undefined 2023, “Rancang Bangun Sistem Informasi Akademik SMK TAMAN SISWA 1 Tanjung Karang BERBASIS WEB,” *pdfs.semanticscholar.org*, vol. 2, no. 1, pp. 41–52, 2023, Accessed: Apr. 06, 2026. [Online]. Available: <https://pdfs.semanticscholar.org/b64b/5240acace32660bc2de2773206339fac6487.pdf>
- [16] M. Alkhairi, ... S. A.-J. (Jurnal, and undefined 2024, “Implementasi Unit Testing Dan End-To-End Testing Pada Sistem Informasi Akademik Teknik Informatika,” *jurnal.stkipggritlungagung.ac.id*, Accessed: Apr. 06, 2026. [Online]. Available: <https://jurnal.stkipggritlungagung.ac.id/index.php/jipi/article/download/5626/2323>
- [17] R. Sangga Rasefta and S. Esabella, “Sistem Informasi Akademik Smk Negeri 3 Sumbawa Besar Berbasis Web,” *J. Inform. Teknol. dan Sains*, vol. 2, no. 1, pp. 50–58, 2020, doi: 10.51401/jinteks.v2i1.558.
- [18] R. Rasefta, S. E.-J. I. T. dan, and undefined 2020, “Sistem informasi akademik smk negeri 3 sumbawa besar berbasis web,” *scholar.archive.org*, Accessed: Apr. 06, 2026. [Online]. Available: <https://scholar.archive.org/work/acgyhi6gpza4bain4lltzss3am/access/wayback/http://www.jurnal.uts.ac.id/index.php/JINTEKS/article/download/558/400>
- [19] Z. Hakim, P. M.-J. S. Informasi, T. Informatika, and undefined 2022, “Sistem Informasi Akademik Berbasis Webiste,” *jurnal.umj.ac.id*, Accessed: Apr. 06, 2026. [Online]. Available: <https://jurnal.umj.ac.id/index.php/just-it/article/view/13610>
- [20] H. Lukmana, M. Al-Husaini, ... I. H.-T. J., and undefined 2023, “Perancangan Sistem Informasi Deteksi Dini Stunting Berbasis Website Menggunakan Metode User Center Design,” *ojs.uniska-bjm.ac.id*, vol. 14, no. 3, 2023, Accessed: Feb. 11, 2026. [Online]. Available: <https://ojs.uniska-bjm.ac.id/index.php/JIT/article/view/12025>
- [21] O. Musa, “Sistem Informasi Kepegawaian Berbasis Website pada Dinas Pariwisata Dan Kebudayaan,” *J. Teknol. Inf. Indones.*, vol. 5, no. 2, pp. 9–15, 2020, doi: 10.30869/jtii.v5i2.641.
- [22] A. Ariska, W. W.-J. sintaks logika, and undefined 2022, “Penerapan Kriptografi Menggunakan Algoritma Des (Data Encryption Standard),” *jurnal.umpar.ac.id*, vol. 2,

- no. 2, 2022, Accessed: Mar. 03, 2025. [Online]. Available: <http://jurnal.umpar.ac.id/index.php/sylog/article/view/1734>
- [23] G. A. Pradipta, R. R. Huizen, I. M. D. Susila, D. Hermawan, P. D. W. Ayu, and D. P. Hostiadi, "Rancang Bangun Sistem Informasi Administrasi Kantor Pada Badan Usaha Milik Desa Studi Kasus Desa Pemecutan Kaja Mandiri," *J. Tekno Kompak*, vol. 17, no. 2, pp. 100–113, 2023, [Online]. Available: <https://ejurnal.teknokrat.ac.id/index.php/teknokompak/article/view/2332>