SYNTIA – Systems and Technologies in Artificial Intelligence Applications.
Volume 1 Issue 1 (2025). Page: 33-41.

Design and Development of a Web-Based Archival Information System Using the Rijndael Algorithm

Muh Rizki Rizaldi[1*], Mohammad Yazdi Pusadan[2], Mohamed Fal Mohamed Fadel[3]

[1] *Informatics Engineering Study Program, Faculty of Engineering, Universitas Tadulako, Palu, Indonesia.*
[2] *Department of Information Technology, Faculty of Engineering, Universitas Tadulako, Palu, Indonesia.*
[3] *Department of International Relations, School of Humanities and Social Sciences, Al Akhawayn University, Ifrane, Morocco.*

**Corresponding author: ti.rizaldi2023@gmail.com*

Abstract: Local election supervisory offices still keep most of their records on paper, which slows retrieval and exposes documents to loss and unauthorized reading. This study designed and implemented a web-based archival information system for the election supervisory office of North Palu sub-district (Panwascam), using CodeIgniter 3 and MySQL, with the Rijndael cipher (AES) for data protection. The system was evaluated with a black-box functional test and a DeLone and McLean questionnaire completed by six staff members, and the encryption example of the source study was checked against an independent AES-128 implementation. All 13 tested functions gave the expected result. Recomputed from item-level data, mean system quality was 3.94 and mean information quality 4.10 on a five-point scale, with overlapping 95% confidence intervals, so the source study's claim that information quality is below average is not supported. The independent implementation reproduced the published ciphertext under PKCS#7 padding, but the printed hand calculation did not match the standard in rounds 1–3 and repeated one state in rounds 4–9. Design issues include a 32-bit integer field for a 13-digit document number, an unspecified cipher mode, and reversible storage if the cipher protects passwords. With six users, single test runs, and no tests of encryption, search, or file handling, the findings are indicative only, and evidence of improved retrieval or security is still needed.

Keywords : archival information system; Rijndael; AES; DeLone and McLean; web application.

Received: November 05, 2025 | Revised: November 28, 2025 | Accepted: December 18, 2025 | Published: December 31, 2025

1. Introduction

Records are the memory of an institution: they support daily administration, serve as evidence in disputes, and make management accountable and transparent. Indonesian Regulation of the General Election Supervisory Agency (Bawaslu) No. 11 of 2020 defines archives as records of activities or events in any form and medium, created and received by state bodies and individuals, and requires that the medium follow developments in information and communication technology [1]. Many local supervisory offices, among them the Panwascam of North Palu, still store incoming and outgoing letters, supervision forms, and other documents on paper, in an order that is not systematic. Retrieval is slow, copies are redundant, and paper deteriorates.

A web-based archival system answers part of this problem: documents are registered once with their metadata, searched from any browser, and reached only after login. Web-based systems in government also need to be judged by their users; the DeLone and McLean model of information systems success has been validated for e-government systems [2] and is the evaluation framework used here [3]. The second part of the problem is protection. The cipher Rijndael, standardized as the Advanced Encryption Standard (AES) [4], [5], is the usual choice for encrypting stored data, but its use in an application raises design questions that go beyond the algorithm: what is encrypted, in which mode, and with which key [6].

This article describes the design and implementation of the archival system and examines three questions. (RQ1) What was built, and do its functions work as specified? (RQ2) How do the intended users

SYNTIA – Systems and Technologies in Artificial Intelligence Applications is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

rate the system quality and information quality, and are the source's statistics and conclusion consistent with its data? (RQ3) Does the Rijndael example in the source agree with a standard implementation, and what does the design imply for security? The contributions are an account of the system and its database, a reanalysis of the questionnaire from item level with uncertainty, and an independent check of the worked encryption example. Section 2 describes the method, Section 3 reports and discusses the results, and Section 4 concludes.

2. Method

2.1. Setting and data

The study followed a qualitative, descriptive approach at the Panwascam office of North Palu from September 2023 to January 2024. The objects were the archived documents (incoming mail, outgoing mail, supervision forms, and other documents) of the five urban villages (*kelurahan*) of the sub-district, and the main sources were interviews with the head of the office, who is also coordinator of the human-resources division, and with an HR staff member, together with a literature study and the text of Regulation No. 11 of 2020. The interviews took place in September 2023 and covered how letters and forms are received, filed, and retrieved. The interviewees agreed to take part and are identified by role only, and the office permitted the use of its documents for this study.

2.2. System development

The system was developed with the Waterfall model [7]: requirements analysis, system design, implementation with unit testing, integration with system testing, and operation with maintenance. The design used data flow diagrams (DFD), entity–relationship diagrams (ERD), and the UML. The tools were Visual Studio Code, the CodeIgniter 3 framework, Bootstrap, Laragon, and MySQL; development used a laptop with a Core i3-1115G4 processor and 4 GB of memory, and the stated minimum for running the system is an Intel N3050 dual-core processor with 4 GB of memory and a 1280 × 720 screen, with any of Windows 7–11 and the Chrome, Firefox, or Edge browser. Fig. 1 shows the resulting structure.

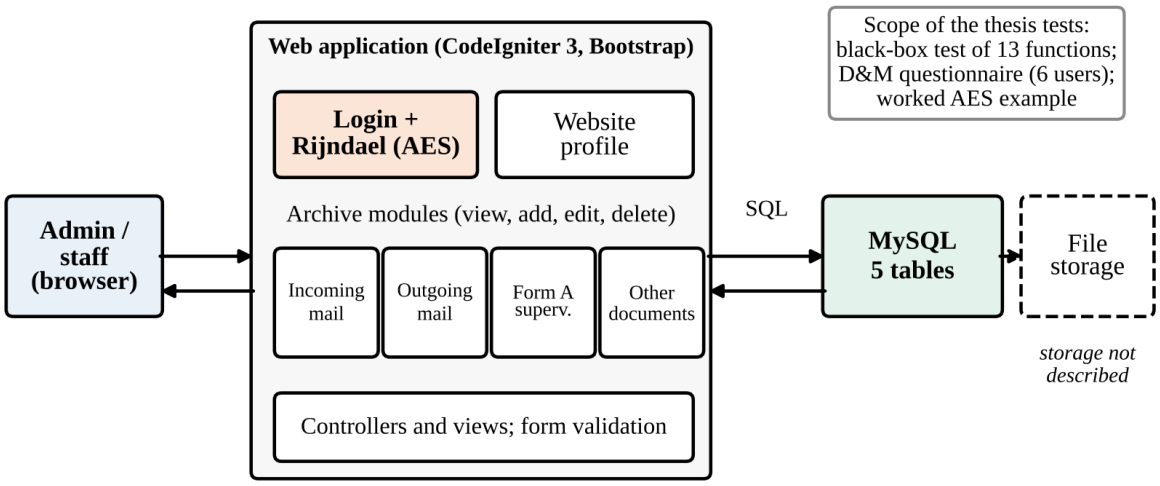


Figure 1. Structure of the archival system and the scope of the tests reported in the source.

2.3. Evaluation design

Functional test. Each function was exercised against its expected result in a black-box test [8], which needs no knowledge of the code. *Success evaluation.* The DeLone and McLean questionnaire covers two dimensions: system quality (six items on ease of use, display, learnability, reliability, relevance of functions, and ease of becoming expert) and information quality (five items on accuracy, conciseness, format, readability, and relevance), resembling end-user computing satisfaction items [9]; the source does

not name the origin of the items, which follow the dimensions of the model [3]. Each item is answered on a five-point scale from 1 (strongly disagree) to 5 (strongly agree) by the six staff members who will use the system. With so few users the sample suits the detection of usability problems [10] but not the estimation of population means, so the results are described, not generalized. From the item-level data we recompute the total and the mean item score of each respondent, the mean, median, and standard deviation across respondents, a 95% t-interval, and Cronbach's α [11], [12] for the items of each dimension:

$$\alpha = \frac{k}{k-1}\left(1 - \frac{\sum_{j=1}^{k} \sigma_j^2}{\sigma_T^2}\right) \tag{1}$$

where k is the number of items, σ_j^2 the variance of item j, and σ_T^2 the variance of the total score. The ordinal ratings are treated as interval, as is common practice [13]. *Cipher check.* AES with a 128-bit key encrypts a 16-byte block in ten rounds [4], [5]. The state s is first combined with the round key K_0; rounds 1 to 9 apply four transformations, and round 10 omits the column mixing:

$$s_r = \mathrm{AddRoundKey}\left(\mathrm{MixColumns}\left(\mathrm{ShiftRows}\left(\mathrm{SubBytes}(s_{r-1})\right)\right), K_r\right), \quad r = 1, \ldots, 9 \tag{2}$$

We reimplemented these steps from the standard and checked the implementation against a reference cryptographic library, then re-ran the source example (plaintext "abcd", key "Informatika@2023") with the two padding rules that could apply: ASCII "0" characters (as the printed hand calculation uses) and PKCS#7. The length of a stored ciphertext follows from the padding: a plaintext of n bytes gives $c = 16(\lfloor n/16 \rfloor + 1)$ bytes under PKCS#7, and Base64 text of $4\lceil c/3 \rceil$ characters, which is needed to size the database fields.

3. Results and Discussion

3.1. System design and database (RQ1)

The context diagram has one external entity, the administrator, and one process that stores archives and returns archive information. The level-1 DFD splits the process into login, incoming mail, outgoing mail, supervision Form A, other documents, and website profile, and each archive process allows viewing, adding, changing, and deleting records. The data are kept in five MySQL tables (Table 1). The use-case diagram of the source, however, also lists criteria, sub-criteria, year, and alternative, and a second role (staff) next to the administrator; none of them appears in the DFD, the tables, or the tests, and they look like leftovers of a decision-support template. On the evidence of the DFD, the tables, and the tests, the system has one role, the administrator, and the staff role and these use cases do not exist in it.

Table 1. Database tables of the system (field name, type, and length as printed in the source).

Table	Fields
user	id (int, primary), name (50), email (50), password (50)
incoming mail	id, letter number (int, 25), date (20), type (25), name (100), address (25), phone (50), file (100)
outgoing mail	same fields as incoming mail
Form A supervision	id, number (int, 25), date (50), stage (25), position (25), file (100)
other documents	id, document name (50), subject (50), year (int, 25), file (50)

Two observations follow from the table. First, the letter number is declared as an integer, but the sample number used in the source, 8000194302015, has 13 digits and exceeds both the signed (2,147,483,647) and the unsigned (4,294,967,295) 32-bit range, and an integer would also drop leading zeros; a text field is the safe type for identifiers of this kind. Second, the "file" fields store a path or name of 50–100 characters, but the source does not describe where the files are kept or whether they are encrypted.

3.2. Interface

The system has a login page, a dashboard with shortcuts to the four archive types and the website profile, and for each archive type a list page with edit and delete buttons and an input form (Fig. 2). The dashboard shows the Bawaslu logo and the name of the office, which are set in the website profile.

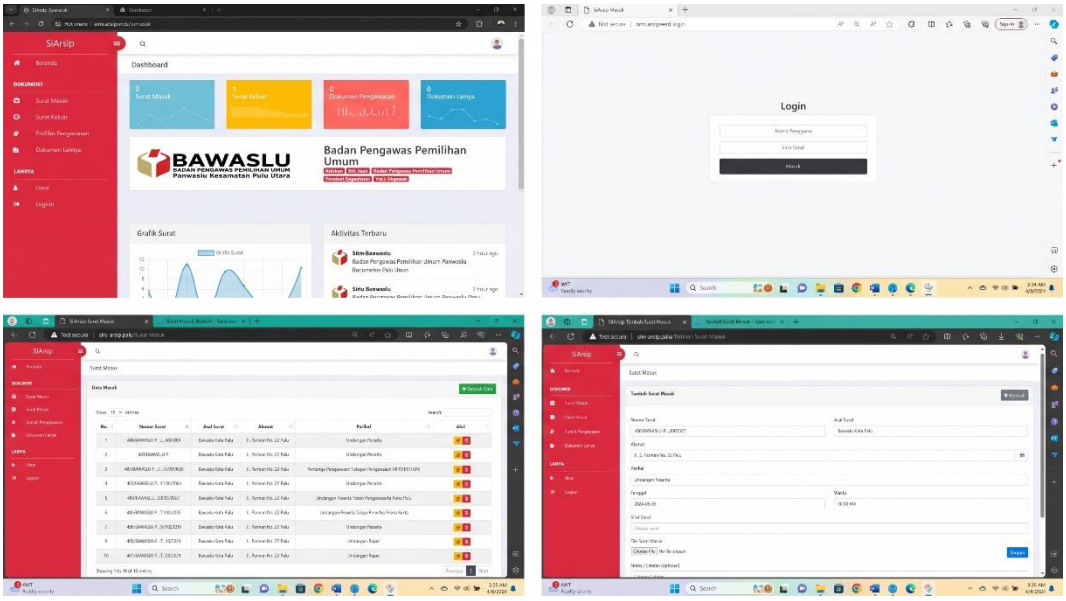


Figure 2. System screens: (a) login page; (b) dashboard; (c) incoming mail list; (d) incoming mail input form.

3.3. Functional test (RQ1)

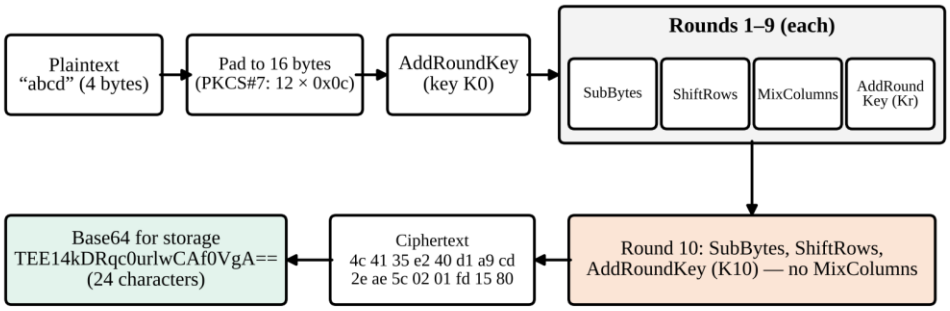
All 13 tested functions gave the expected result (Table 2): login, and saving, changing, and deleting a record in each of the four archive types. The source adds that the system shows an error message when a function is used wrongly, but no such case is in the table. Each function was run once with valid input, so the test shows that the interface responds as designed, not that it is robust. Logout, viewing a record, searching, classification, uploading and downloading files, and report export are not in the test; the use-case figure of the earlier version of this article showed several of them (and a separate "encrypt/decrypt archive data" use case), but the source neither describes nor tests them.

Table 2. Black-box test of the system: 13 functions, one run each.

Module	Functions tested	Number	Result
Login	Login with a registered account	1	As expected
Incoming mail	Save, change, delete	3	As expected
Outgoing mail	Save, change, delete	3	As expected
Form A supervision	Save, change, delete	3	As expected
Other documents	Save, change, delete	3	As expected

3.4. Rijndael implementation and verification (RQ3)

The source states that Rijndael is embedded in the login to keep unauthorized users out, and the aim is to protect the archives; it demonstrates the cipher with a worked example (Fig. 3): the plaintext "abcd", padded to 16 bytes, is encrypted with the 16-character key "Informatika@2023" and gives the Base64 string TEE14kDRqc0urlwCAf0VgA==. The source does not say which data the cipher protects (passwords, document numbers, the uploaded files, or all three), the encryption mode, how the key is stored, or whether there is an initialization vector.



Key: "Informatika@2023" (16 bytes = AES-128). The ciphertext equals the output of a standard AES-128 implementation with PKCS#7 padding (Section 3.4).

Figure 3. AES-128 encryption of the source example, from plaintext to the stored Base64 text.

Our independent computation gives two results. First, the final ciphertext, 4c 41 35 e2 40 d1 a9 cd 2e ae 5c 02 01 fd 15 80, which is TEE14kDRqc0urlwCAf0VgA== in Base64, is exactly what a standard AES-128 implementation produces for "abcd" with PKCS#7 padding. The running system therefore uses the cipher correctly, at least for this input. Second, the printed hand calculation does not reproduce that result (Table 3). It pads with ASCII "0" (0x30), which gives a different ciphertext (3f 0c 5e 58 …); its printed round states agree with the standard values in none of the 16 byte positions for rounds 1, 2, and 3; the printed round key 1 also differs from the key schedule of the standard in all 16 positions; and the printed states of rounds 4–9 are identical to the state of round 1, so only three different states appear in nine rounds, whereas a correct computation gives nine. Even single operations carry slips: the third byte of the initial XOR is printed as 32 where 0x30 ⊕ 0x32 is 02, and in round 1 the last bytes of rows 3 and 4 are printed as f9 and 08 where the XOR of the printed operands gives b7 and ed. The calculation also arranges the state by rows, whereas AES fills it by columns, so its ShiftRows step moves the wrong bytes. The worked example thus illustrates the structure of the algorithm but cannot be used as a test vector.

Table 3. Printed hand calculation of the source compared with a standard AES-128 computation (key "Informatika@2023", plaintext "abcd"). Bytes are in hexadecimal.

Item	Printed in source	Standard AES-128
Padding	ASCII "0" (0x30) × 12	PKCS#7 (0x0c × 12) gives the printed ciphertext
Initial XOR, row 4	02 00 32 03	02 00 02 03
Round key 1	da 87 e1 8e / 7b 16 77 03 / …	4c 4d a5 4c 3e 20 c4 38 57 4b a5 78 65 7b 97 4b
State after round 1	fc 23 4a 79 / d2 ce d5 eb / …	5a f2 e9 7b 71 cc 42 54 0f e0 f6 00 12 d2 54 5f
Rounds 4–9	same state as round 1 in each round	nine different states
Final ciphertext	4c 41 35 e2 … (not obtained from the printed operands)	4c 41 35 e2 … (PKCS#7, matches the system)

The example also raises four design points. *Mode.* A single block gives the same ciphertext in ECB mode and in CBC mode with an all-zero initialization vector, so the example does not show the mode. If ECB is used, equal plaintexts give equal ciphertexts, which reveals repeated values such as identical document numbers [6]. *Key.* A 16-character text string is used directly as the 128-bit key, which gives a key space far smaller than 2^{128}, and the key is printed in the source; a randomly generated key kept outside the source code and the database is preferable. *Passwords.* If the cipher protects the login passwords, as the source states, the storage is reversible: whoever has the key can read them all. Guidance for authentication recommends storing a salted hash made with a key-derivation function [15], and the properties of password schemes are compared in [14]. *Field length.* A Base64 ciphertext is much longer than the plaintext (Table 4); with the 50-character password field of Table 1, passwords of up to 31 characters fit and a 32-character password does not (64 characters).

Table 4. Stored size of a PKCS#7 ciphertext in Base64 for a plaintext of n bytes (AES block 16 bytes).

Plaintext n (bytes)	Ciphertext (bytes)	Base64 (characters)
4 ("abcd")	16	24
13 (document number)	16	24
15	16	24
16	32	44
31	32	44
32	48	64

These points do not mean that the system is insecure; they show that the source does not give the information needed to judge it. The security claim therefore stays unsupported until the data, mode, key handling, and password storage are specified and tested, for example with known-answer tests against the standard vectors.

3.5. DeLone and McLean evaluation (RQ2)

Fig. 4 shows the 66 item responses of the six respondents (R1–R6). Table 5 gives the recomputed statistics next to the values printed in the source. The totals of system quality agree with the items (18, 30, 28, 24, 22, 20), but the total of respondent 5 for information quality is printed as 22 while the five items (4, 3, 3, 4, 4) sum to 18. The summary table of the source also gives the number of cases as 9 instead of 6, and its statistics do not follow from its own data: the mean of the system-quality totals is 23.67 and the median 23 (printed: 24 and 24); the mean of the information-quality totals is 20.5 and the median 21.5 (printed: 19.5 and 20; the printed total 22 would give 21.17 and 22.5). Fig. 5 shows the differences.

Table 5. Recomputed statistics of the questionnaire (6 respondents): total score (mean and SD), median, range, mean item score on the 1–5 scale with its 95% confidence interval, Cronbach's α, and the mean/median printed in the source.

Dimension	Items	Total mean (SD)	Median	Range	Item mean	95% CI	α	Printed
System quality	6	23.67 (4.63)	23	18–30	3.94	3.13–4.75	0.80	24 / 24
Information quality	5	20.50 (3.51)	21.5	15–24	4.10	3.36–4.84	0.88	19.5 / 20

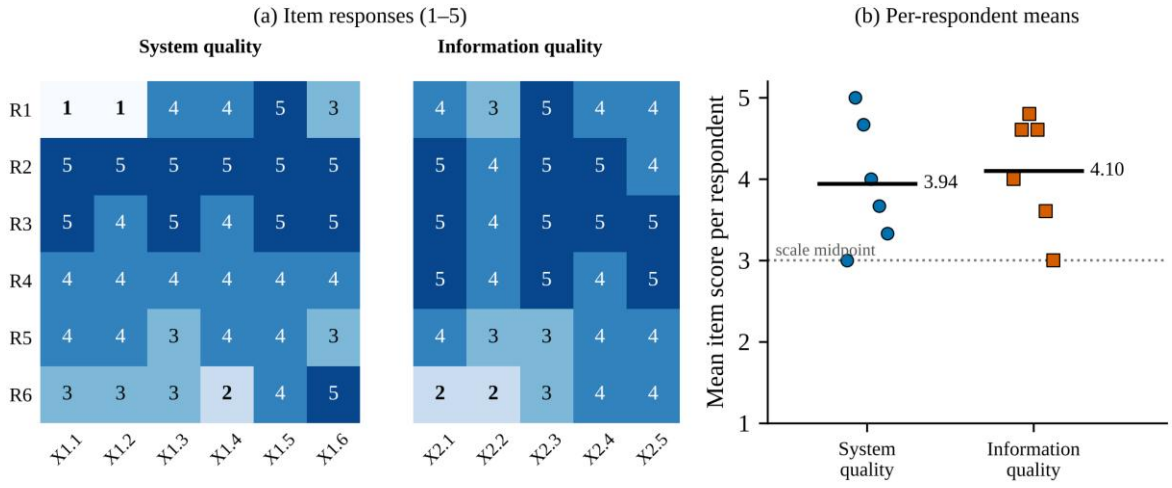


Figure 4. (a) Responses of the six respondents to the 11 items; (b) mean item score of each respondent, with the group mean.

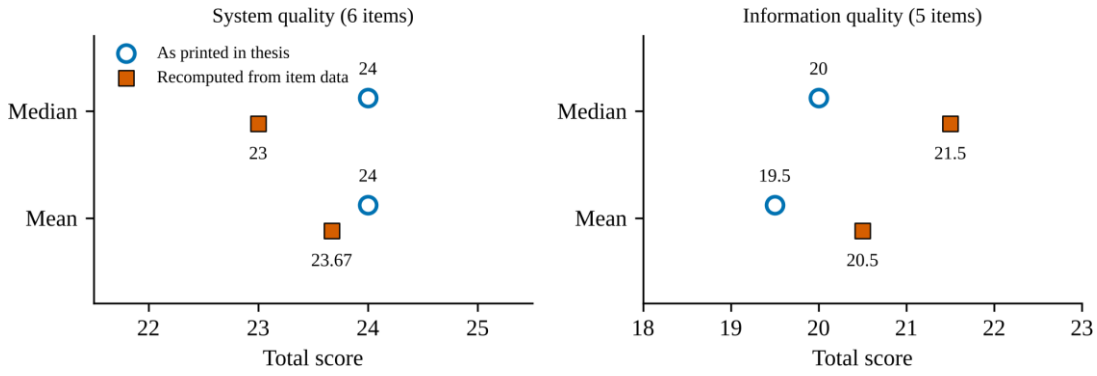


Figure 5. Mean and median of the total scores as printed in the source and as recomputed from its item data.

The conclusion of the source, that information quality is below average while system quality is above it, does not hold. The totals of the two dimensions are not comparable, because the scales have six and five items; per item, information quality (4.10) is slightly higher than system quality (3.94), and both are above the scale midpoint of 3. The two confidence intervals overlap widely (3.13–4.75 and 3.36–4.84), so with six respondents no difference can be claimed, and the reading that the "information output does not satisfy the users" is not supported. The picture is rather that most respondents agree (R2, R3, and R4 give 4 or 5 to almost every item) and two respondents drive the low values. Respondent 1 answers 1 to both X1.1 (ease of use) and X1.2 (display) but 3 to 5 to the other system items, and respondent 6 answers 2 to X1.4 (access without problems), X2.1 (convincing information), and X2.2 (conciseness). The lowest item means are X2.2 (3.33) and X1.2 (3.50). The two totals are positively related (Spearman $\rho = 0.72$, $p = 0.10$, $n = 6$), which is only suggestive. Cronbach's α is 0.80 and 0.88, but with six respondents the estimates are unstable [12].

The result supports a cautious statement: the six staff members rate both the system and its information as good, with ease of use and conciseness of information as the points to examine. It does not show that the system is successful in the sense of the model, which also includes use, user satisfaction, and net benefits [3], [16], [17], none of which was measured. The six respondents are all the staff members who will use the system, and they completed the questionnaire on paper at the office after a demonstration of the system. The source records no respondent comments, so the reasons for the low ratings of X1.1, X1.2, and X2.2 remain unknown.

3.6. Limitations

The evaluation covers one office and six users, with single test runs and no performance data. The system was not tested for encryption and decryption, searching, file upload and download, access control, or invalid input, and the source states no response time, storage size, or encryption time. The comparison with the earlier paper-based practice, for example the time to retrieve a document, was not measured, although retrieval time is the problem named in the introduction. The hand calculation of the cipher cannot serve as a test vector, and the cipher mode and key handling are unknown.

4. Conclusion

The archival system for the Panwascam of North Palu manages four document types with login and create, edit, and delete functions, and all 13 tested functions behave as designed. The six staff members rate system quality (3.94 of 5) and information quality (4.10) as good; the printed conclusion that information quality is below average comes from comparing totals of scales of different length and from arithmetic that does not match the data. The Rijndael output of the source example is the output of a standard AES-128 with PKCS#7 padding, but the printed hand calculation does not reproduce it, and the design leaves open the cipher mode, the key handling, and, if passwords are encrypted, their reversible storage. Evidence of improved retrieval or of security is not yet available.

The next work should specify what is encrypted, use a random key, an authenticated mode with a fresh initialization vector, and salted password hashes; correct the field types; test encryption and decryption with known-answer vectors and the system with invalid inputs and larger files; and measure retrieval time against the paper practice with more users.

AI Disclosure Statement

The authors used Generative AI and AI-Assisted Technologies throughout the preparation of this manuscript: to draft and revise portions of the text, to write and run the scripts behind the reanalysis of the questionnaire data and the independent AES computation, to generate the figures, and to format the document from the authors' own research data. The authors verified the scripts and their outputs, reviewed the resulting text, cross-checked every reference against its original source, and accept full responsibility for the content of this publication.

Author Contributions Statement

Muh Rizki Rizaldi: Conceptualization, Methodology, Software, Investigation, Data Curation, Writing.
Mohammad Yazdi Pusadan: Supervision, Validation, Formal Analysis, Writing – Review and Editing.
Mohamed Fal Mohamed Fadel: Resources, Writing – Review and Editing.

References

[1] Badan Pengawas Pemilihan Umum Republik Indonesia, *Peraturan Badan Pengawas Pemilihan Umum Nomor 11 Tahun 2020 tentang Klasifikasi Arsip* (in Indonesian). Jakarta, Indonesia, 2020.

[2] C.-L. Wang and Y.-C. Liao, "Assessing eGovernment systems success: A validation of the DeLone and McLean model of information systems success," *Gov. Inf. Q.*, vol. 25, no. 4, pp. 717–733, 2008, doi: 10.1016/j.giq.2007.06.002.

[3] W. H. DeLone and E. R. McLean, "The DeLone and McLean model of information systems success: A ten-year update," *J. Manage. Inf. Syst.*, vol. 19, no. 4, pp. 9–30, 2003, doi: 10.1080/07421222.2003.11045748.

[4] J. Daemen and V. Rijmen, *The Design of Rijndael: AES – The Advanced Encryption Standard*. Berlin, Germany: Springer, 2002, doi: 10.1007/978-3-662-04722-4.

[5] National Institute of Standards and Technology, "Advanced Encryption Standard (AES)," Federal Information Processing Standards Publication 197, Gaithersburg, MD, USA, 2001, doi: 10.6028/NIST.FIPS.197.

[6] M. J. Dworkin, "Recommendation for block cipher modes of operation: Methods and techniques," NIST Special Publication 800-38A, Gaithersburg, MD, USA, 2001, doi: 10.6028/NIST.SP.800-38a.

[7] N. B. Ruparelia, "Software development lifecycle models," *ACM SIGSOFT Softw. Eng. Notes*, vol. 35, no. 3, pp. 8–13, 2010, doi: 10.1145/1764810.1764814.

[8] S. Nidhra, "Black box and white box testing techniques – A literature review," *Int. J. Embedded Syst. Appl.*, vol. 2, no. 2, pp. 29–50, 2012, doi: 10.5121/ijesa.2012.2204.

[9] W. J. Doll and G. Torkzadeh, "The measurement of end-user computing satisfaction," *MIS Q.*, vol. 12, no. 2, pp. 259–274, 1988, doi: 10.2307/248851.

[10] J. Nielsen and T. K. Landauer, "A mathematical model of the finding of usability problems," in *Proc. INTERACT '93 and CHI '93 Conf. Human Factors Comput. Syst.*, Amsterdam, The Netherlands, 1993, pp. 206–213, doi: 10.1145/169059.169166.

[11] L. J. Cronbach, "Coefficient alpha and the internal structure of tests," *Psychometrika*, vol. 16, no. 3, pp. 297–334, 1951, doi: 10.1007/BF02310555.

[12] M. Tavakol and R. Dennick, "Making sense of Cronbach's alpha," *Int. J. Med. Educ.*, vol. 2, pp. 53–55, 2011, doi: 10.5116/ijme.4dfb.8dfd.

[13] G. M. Sullivan and A. R. Artino, "Analyzing and interpreting data from Likert-type scales," *J. Grad. Med. Educ.*, vol. 5, no. 4, pp. 541–542, 2013, doi: 10.4300/JGME-5-4-18.

[14] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "The quest to replace passwords: A framework for comparative evaluation of Web authentication schemes," in *Proc. IEEE Symp. Security Privacy*, 2012, pp. 553–567, doi: 10.1109/SP.2012.44.

[15] P. A. Grassi *et al.*, "Digital identity guidelines: Authentication and lifecycle management," NIST Special Publication 800-63B, Gaithersburg, MD, USA, 2017, doi: 10.6028/NIST.SP.800-63b.

[16] S. Petter, W. DeLone, and E. McLean, "Measuring information systems success: Models, dimensions, measures, and interrelationships," *Eur. J. Inf. Syst.*, vol. 17, no. 3, pp. 236–263, 2008, doi: 10.1057/ejis.2008.15.

[17] P. B. Seddon, "A respecification and extension of the DeLone and McLean model of IS success," *Inf. Syst. Res.*, vol. 8, no. 3, pp. 240–253, 1997, doi: 10.1287/isre.8.3.240.