Journals.Telkomuniversity.ac.id

Journal of Software Engineering and Multimedia **(JASMED)**

Vol. 3 No. 1 (2025) | eISSN:3025-6372 | OPEN ACCESS
https//doi.org/10.20895/jasmed.v3i1.10096

Full Paper

A Web Application For Playfair Cipher Cryptography: Method and Implementation

Argia Putri Ramadhani [1*],
Universitas Tunas Pembangunan, Surakarta, Indonesia
f0223005_argiaputriramadhani@student.utp.ac.id
Apriliana Vera Ardiyanti[2],
Universitas Tunas Pembangunan, Surakarta, Indonesia
f0223003_aprilianaveraardiyanti@student.utp.ac.id
Erni Widarti[3],
Universitas Tunas Pembangunan, Surakarta, Indonesia
erni.widarti@lecture.utp.ac.id

*Corresponding Author

ABSTRACT

Amidst the increasing digital threats, understanding the basic principles of cryptography is crucial for data security education. While the Playfair Cipher remains a valuable classical algorithm, its manual use is prone to human error and is not compatible with modern computing systems. This research aims to address these issues by upgrading the Playfair Cipher through a software engineering approach, creating a web application that runs using HTML, CSS, and JavaScript. Unlike previous research that focused on descriptive analysis, this research emphasizes the application of new technologies by designing a structured client-side architecture, resulting in seamless processing and server-side execution. The application features a responsive user interface (UI) designed for easy user interaction and immediate visual feedback while constructing key matrices and solving bigrams. The results show that the application achieves 100% accuracy in validating manual logic, especially when verifying the ciphertext "CVSMOBLIMAPUOYHDBY," and offers highly efficient computational performance. This research provides a powerful and interactive software tool that upgrades classical methods and offers an extensible solution for algorithm validation and data security education.

KEYWORDS

Cryptography; Playfair Cipher; Encryption; Message Security; Web Application

Ramadhani, A. P., Ardiyanti, A. V., Widarti, E.. (2025). A Web Application For Playfair Cipher Cryptography: Method and Implementation. jasmed, 3(1), pp. 20-29. https//doi.org/10.20895/jasmed.v3i1.10096

Article Submitted 06/11/2025. Revision uploaded 25/12/2025. Accepted 25/11/2025.

1. INTRODUCTION

In today's increasingly digital world, maintaining security when sharing information across various communication platforms is crucial [1]. Sensitive data remains a prime target for cybercriminals using methods such as hacking and phishing, requiring robust solutions to protect message confidentiality [2]. Cryptography is a crucial tool for maintaining security, as it transforms data into an unreadable form, allowing only those with a special key to decipher it [3]. One fundamental symmetric method is the Playfair cipher, a digraph substitution system invented by Charles Wheatstone and popularized by Baron Lyon Playfair [4]. Unlike simple encryption systems like Caesar or Vigenère, Playfair offers a more complex and secure approach to protecting information on a network.

Historically known for its military tactical strategy, the Playfair Cipher was used to increase obfuscation in communications during World War I [5]. Today, in educational contexts, it has moved into the academic realm as a fundamental learning tool. It is used to introduce the logical concepts of 6x6 matrix expansion or pairwise letter substitution in digital text embedding [6]. Although its role has been supplanted by more sophisticated modern encryption systems, the science and art of encrypting messages through Playfair Ciphers remains crucial to understanding the historical development of cryptography.

Research on classical cryptography continues to advance, incorporating various algorithms to enhance message security. For example, this algorithm has been used to protect corporate asset data, although research is still limited to certain types of tables [7]. Other implementations utilize random keypads to enhance encryption security, but results indicate that non-letter characters are not yet supported in these systems [8]. Furthermore, blending this cipher with the Least Significant Bit (LSB) method for digital images has been attempted to enhance data confidentiality [9]. Another specialized method uses ASCII symbols to write mathematical equations, although more complex symbols are not yet available [10]. Another studies there have been attempts to improve the system by gradually modifying Morse code to enhance security, but this makes the execution process more complex and time-consuming [11].

Recent studies have shown that expanding the standard key matrix to 12x8 significantly increases the complexity of the algorithm. This allows for the inclusion of numeric characters and symbols while making the system more resistant to brute-force attacks by computers [12]. In addition, there are plans to incorporate a 10x10 Polybius Square into the Playfair framework, thus expanding the character space and allowing the acceptance of printable ASCII symbols. These symbols are usually absent in the commonly used 5x5 matrix [13]. The development of hybrid cryptographic systems that combine the Double Playfair Cipher with asymmetric algorithms such as Elgamal has proven effective in forming multi-layer security protocols, specifically for protecting digital multimedia files [14]. Comparative analysis between Playfair Cipher and Shift Cipher shows that digraph-based encryption still excels in character frequency pattern capabilities, although it requires modern approaches to counter automated cryptanalysis tools [15]. Other studies transforming encryption tasks into digital formats through cloud platforms such as Google Colaboratory facilitate the execution of more flexible and user-friendly algorithms, for testing the security of text on a large scale in modern environments [16].

Despite advances in modifying the mathematical rules of algorithms, significant gaps in the existing literature remain regarding the accessibility and visualization of the implementation process. Most previous studies focused on complex multi-step modifications or rigid and less interactive database integrations, making it difficult for learners to validate the logic in real time. This research addresses these limitations by shifting the focus from pure algorithm modification to a software engineering approach. The novelty of this study lies in the development of a functional web-based application that utilizes a modular client-side architecture. Unlike previous research, which was generally console-based or server-dependent, this application is capable of processing data without delay and provides a user-friendly and responsive interface. This application acts as a modern teaching tool specifically designed to connect classical theory with practical application, allowing users to directly see the process of moving from manual bigram splitting to automatic ciphertext generation. This contribution provides a scalable software solution for cryptography education and algorithm testing in a modern web environment.

2. METHODS

This chapter aims to understand, implement, and test the Playfair Cipher algorithm. This research includes data collection, algorithm use, and verification and validation of message encryption and decryption results, both manually and through a web-based application. This research follows the research flow outlined in Error! Reference s ource not found..

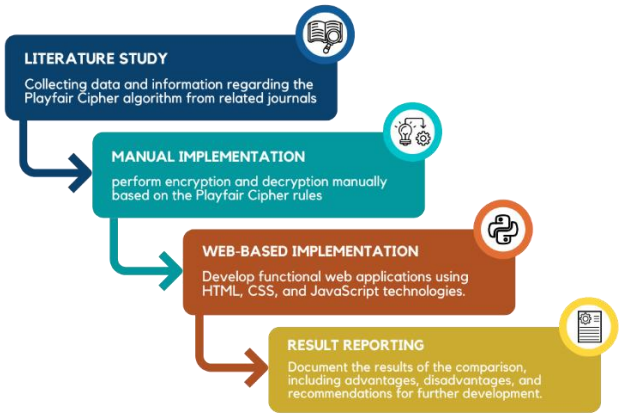

Figure 1. Research Flow

Based on **Figure 1**, this research began with a literature review, collecting data and information from relevant journals. These references were used to understand the basic concepts of the Playfair cipher algorithm, its encryption and decryption rules, how to construct the cipher matrix, and its implementation in case studies.

The second stage of this research is manual implementation, where the Playfair cipher algorithm is manually applied to understand the encryption and decryption processes. The first step is to create a 5x5 cipher matrix based on the keyword by removing the letter "J" from the matrix. An example is shown in

Figure 2.

A	B	C	D	E
F	G	H	I	K
L	M	N	O	P
Q	R	S	T	U
V	W	X	Y	Z

Figure 2. 5x5 Cipher Matrix

Next, the plaintext is broken down into letter pairs (bigrams), inserting the letter "Z" if there are identical letter pairs or if the number of letters is odd. Afterward, the encryption process is carried out based on the Playfair Cipher algorithm rules to produce the ciphertext. The final step in this manual implementation is the decryption process, which returns the ciphertext to the original plaintext. The results of this manual implementation are documented and used as a comparison for the web application implementation.

After manual implementation, the third stage is the implementation of a web application. The Playfair Cipher algorithm is automated in the form of an interactive web application built using HTML, CSS, and JavaScript. This begins with creating a JavaScript function to dynamically generate a cipher matrix based on user input, followed by creating an algorithm that automatically processes the text into bigrams. Next, the algorithm's encryption and

decryption rules are implemented in JavaScript code to process the text on the client side. The results of this web application implementation will be compared with the previous manual implementation.
The final stage is reporting the results. This includes documentation comparing the manual implementation and the web application, their advantages and disadvantages, and recommendations for further algorithm development. This documentation serves as a guide for evaluating the research results and as a reference for future follow-up studies.

3. RESULT AND DISCUSSION

This chapter will explain the encryption and decryption processes, the results of implementing the Playfair Cipher algorithm manually and through a web application, and how to check and validate the implementation results. The explanation includes the formation of the key matrix, the processing of plaintext into ciphertext, and the verification of the decryption results to ensure the original message can be correctly recovered.

3.1 MANUAL IMPLEMENTATION

Manual implementation is done to understand the Playfair Cipher process in depth by following the algorithm rules directly without the help of automated tools.

3.1.1 Encryption Process

The manual implementation begins with the formation of a cipher matrix based on a keyword. To demonstrate this process, **"ALGORITHM"** is chosen as an example keyword. The letters are entered into a 5x5 matrix, eliminating duplicate letters. Letters not included in the keyword are filled in alphabetically (excluding the letter "J"). The resulting key matrix can be seen in **Figure 3**.

A	L	G	O	R
I	T	H	M	B
C	D	E	F	K
N	P	Q	S	U
V	W	X	Y	Z

Figure 3. "ALGORITHM" Key Matrix

Once the key matrix is formed, the plaintext is split into letter pairs (bigrams). For this manual implementation, the plaintext **"INFORMATION SYSTEM"** is used as an example. The splitting rule is : if there are repeated or identical letters in a pair, the letter 'Z' is inserted between them. If the number of letters is odd, the letter 'Z' is added at the end to fulfill the bigram format. The bigram splitting result for this example is shown in Error! Reference source n ot found..

PLAINTEXT : INFORMATION SYSTEM
BIGRAMS : IN FO RM AT IO NS YS TE MZ

Figure 4. Results of breaking down plaintext into bigrams

The encryption process using the Playfair Cipher algorithm follows the following rules:

1. **Same row :** If both letters in a bigram are on the same row, each letter is replaced by the letter to its right in that row. If the letter is in the last column, it is replaced by the letter in the first column of the same row.
2. **Same column :** If both letters in a bigram are in the same column, then each letter is replaced by the letter below it in that column. If the letter is in the last row, then it is replaced by the letter in the first row in the same column.
3. **Square / rectangular corners :** If the two letters form a square or rectangular corner (not in the same row or column), swap each letter with a letter in the same row but in the column of the other letter pair.

Next, the manual encryption process is applied to the **"ALGORITHM"** keyword matrix shown in **Figure *3*** and the **"INFORMATION SYSTEM"** plaintext shown in Figure 4 by following the rules above. As a visual example of the application of these three rules:

- **Rule 1 (Same row)** is applied to the NS bigram (encrypted into PU), as shown in **Figure 5**.
- **Rule 2 (Same columns)** is applied to the YS bigram (encrypted into OY), as shown in **Figure 6**.
- **Rule 3 (Square / rectangular corners)** is applied to the TE bigram (encrypted as HD), as shown in **Figure 7**.

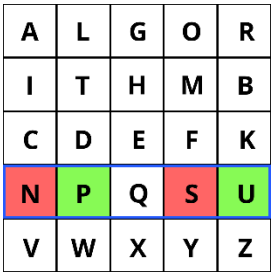

Figure 5. Example of applying Rule 1 (Same rows) to NS bigrams

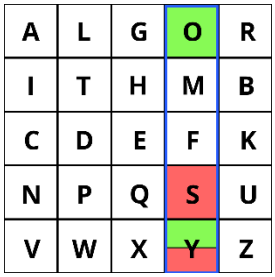

Figure 6. Example of applying Rule 2 (Same columns) to YS bigrams

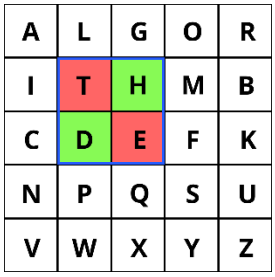

Figure 7. Example of applying Rule 3 (Square corners) to TE bigrams

After all bigrams were manually processed, the entire **"INFORMATION SYSTEM"** plaintext was successfully encrypted into the ciphertext **"CVSMOBLIMAPUOYHDBY"**. This process provides a clear picture of the manual implementation of the Playfair Cipher algorithm and serves as a basis for comparison for application implementation.

3.1.2 Decryption Process

The decryption process converts the ciphertext back to the original plaintext. This process uses the same key matrix, the **"ALGORITHM"**. The key matrix used can be seen again in **Figure 3**. But applies the reverse rules of encryption.

The ciphertext used in the decryption process is **"CVSMOBLIMAPUOYHDBY"**, which is the result of the **"INFORMATION SYSTEM"** encryption. Before decryption, the ciphertext is broken down into letter pairs (bigrams), as shown in **Figure 8**.

Figure 8. The result of breaking the ciphertext into bigrams

The decryption rules in the Playfair Cipher algorithm are the reverse of the encryption rules, as follows:

1. **Same row :** If both letters in a bigram are in the same row, each letter is replaced by the letter to its left in that row. If the letters are in the first column, they are replaced by the letter in the last column of the same row.
2. **Same column :** If both letters in a bigram are in the same column, each letter is replaced by the letter above it in that column. If the letters are in the first row, they are replaced by the letter in the last row of the same column.
3. **Square/rectangular corner :** If both letters form a square/rectangular corner, each letter is replaced by the letter in the same row but in the column of the other letter pair.

Using the rules above, the decryption process is applied to each bigram in **Figure 8**. As a visual example of the application of the three rules :

- The PU bigram is decrypted into NS using **Rule 1 (Same row)**, as shown in **Figure 9**.
- The CV bigram is decrypted into IN using **Rule 2 (Same column)**, as shown in **Figure 10**.
- The bigram MA is decrypted into IO using **Rule 3 (Square corners)**, as shown in **Figure 11**.

A	L	G	O	R
I	T	H	M	B
C	D	E	F	K
N	P	Q	S	U
V	W	X	Y	Z

Figure 9. Example of applying Rule 1 (Same row) to decrypt PU bigram into NS

A	L	G	O	R
I	T	H	M	B
C	D	E	F	K
N	P	Q	S	U
V	W	X	Y	Z

Figure 10. Example of applying Rule 2 (Same Column) to decrypt CV bigram into IN

A	L	G	O	R
I	T	H	M	B
C	D	E	F	K
N	P	Q	S	U
V	W	X	Y	Z

Figure 11. Example of applying Rule 3 (Square corners) to decrypt MA bigrams into IO.

After all the ciphertexts were processed, the original plaintext that was successfully recovered was **"INFORMATION SYSTEMZ"**. This decryption process demonstrated that the Playfair Cipher algorithm can accurately restore the ciphertext to its plaintext form.

3.2 WEB APPLICATION IMPLEMENTATION

To demonstrate practical implementation and provide a visual validation tool, a functional web-based application was developed. This implementation was built using standard web technologies. HTML was used to construct the interface structure, CSS for modern visual styling, and JavaScript to handle all the client-side logic of the Playfair Cipher algorithm. This application is fully functional and can accept random keywords and plaintext or ciphertext from users. However, to demonstrate validity and consistency with manual calculations, testing of this web application was conducted using the same parameters as the manual case study.

This test, shown in Figures 12, 13, and 14, uses the keyword **"ALGORITHM"** and the plaintext **"INFORMATION SYSTEM"**. **Figure *12*** shows the application interface for key matrix generation. **Figure *13*** shows that the web application successfully encrypts the plaintext into the ciphertext **"CVSMOBLIMAPUOYHDBY"**, which is identical to the manual calculation result. Furthermore, **Figure *14*** shows that the ciphertext is successfully decrypted back to the original plaintext, fully validating the web application implementation.

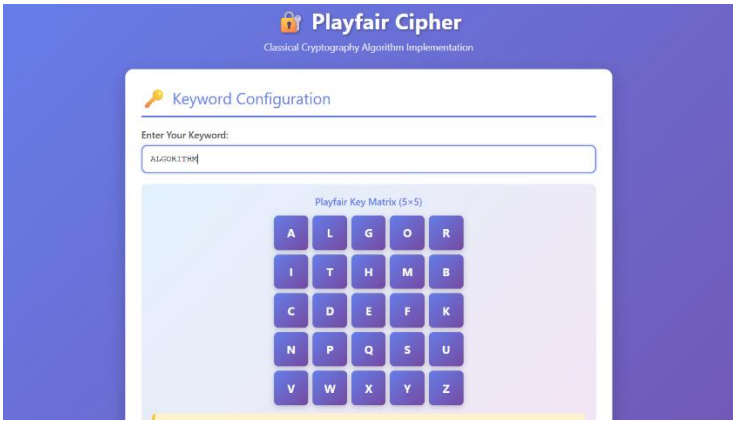

Figure 12. Web application view for keyword configuration (ALGORITHM) and automatic key matrix generation.

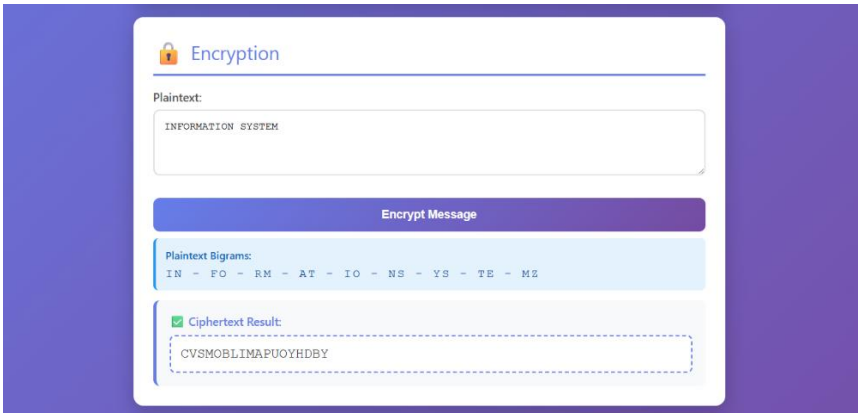

Figure 13. The results of the INFORMATION SYSTEM encryption test on the web application, show the plaintext bigrams and ciphertext result (CVSMOBLIMAPUOYHDBY).

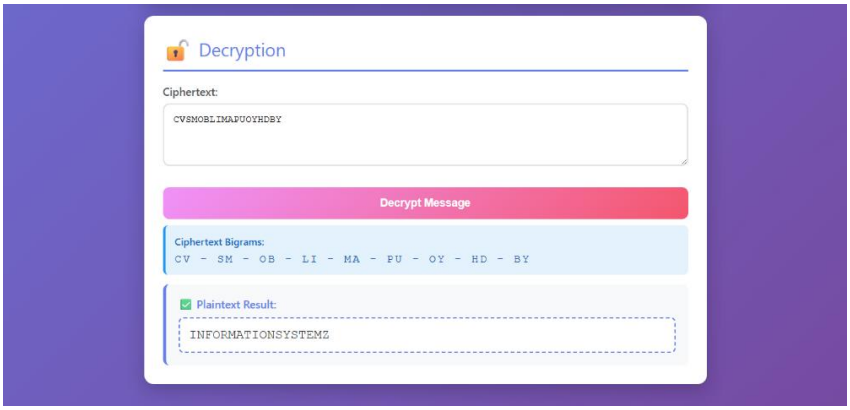

Figure 14. The results of the decryption test, validate the recovery of the ciphertext CVSMOBLIMAPUOYHDBY back to the plaintext INFORMATIONSYSTEM

3.3 REPORTING RESULTS

Successful recovery of the original plaintext in both methods demonstrated that the algorithm logic was implemented correctly. The comparison between the two approaches highlighted significant differences in efficiency. The manual implementation took significantly longer because each step, such as key matrix formation and bigram cracking, had to be performed individually. In contrast, the web application implementation could complete the entire encryption and decryption process automatically and instantly, while reducing the potential for human error in ciphertext and plaintext calculations. To further test the application's robustness, a more extensive evaluation was conducted with various test cases beyond the core case. These tests included longer texts and keywords with complex repeating characters to assess the system's stability in processing data. The results consistently demonstrated that the client-side JavaScript logic was able to separate bigrams with 100% accuracy and correctly insert the 'Z' filler character across all test cases. This more in-depth testing demonstrated that the web application not only matched manual logic in simple cases but was also more reliable and error-free than manual methods, especially when handling extreme cases and large data volumes without compromising performance.

From this implementation, several advantages of the Playfair Cipher algorithm can be identified. One of its main advantages is its simplicity, which makes it easy to understand and implement both manually and programmatically. Furthermore, its keyword-based flexibility allows encryption results to vary depending on the key used. However, the Playfair Cipher algorithm also has several drawbacks. Its security level is relatively low compared to modern algorithms, making it vulnerable to frequency attacks because it only uses bigram-based substitution. Furthermore, this algorithm does not support characters other than the alphabet (A-Z), which limits the flexibility of its use in various languages and contexts involving numbers or symbols.

4. CONCLUSION

This research successfully updated the Playfair Cipher by creating a functional web-based application, providing a bridge between classical cryptographic theory and modern software applications. While this application effectively examines the algorithm's logic for educational purposes, it has several noteworthy shortcomings. The current system only supports letters of the alphabet (A-Z) and does not support numbers or symbols, limiting its use in more modern situations. Furthermore, because this method belongs to classical cryptography, its security level is relatively low and it is still vulnerable to attacks by frequency analysis methods, meaning it is not suitable for protecting highly sensitive information in real-world environments.

For future development, it is recommended to combine this tool with modern encryption standards, such as the Advanced Encryption Standard (AES), to create a more comprehensive security model. Improvements should also focus on better input validation standards to protect against non-letter characters and develop a dynamic bigram visualization mode to further enhance the tool's usefulness in learning. By addressing these issues, future iterations could create a more comprehensive and secure platform for testing and learning cryptography.

5. REFERENCE

[1] D. Susanti, "Analisis Modifikasi Metode Playfair Cipher Dalam Pengamanan Data Teks," *Indonesian Journal of Data and Science*, vol. 1, no. 1, pp. 11–18, 2020, doi: 10.33096/ijodas.v1i1.4.

[2] E. Susanto, Lady Antira, K. Kevin, E. Stanzah, and A. A. Majid, "Manajemen Keamanan Cyber Di Era Digital," *Journal of Business And Entrepreneurship*, vol. 11, no. 1, p. 23, 2023, doi: 10.46273/jobe.v11i1.365.

[3] Dola Ramalinda, Jayadi, and Agung Rachmat Raharja, "Strategi Perlindungan Data Menggunakan Sistem Kriptografi Dalam Keamanan Informasi," *Journal of International Multidisciplinary Research*, vol. 2, no. 6, pp. 665–671, 2024, doi: 10.62504/jimr679.

[4] C. A. D. Hasibuan, "Implementasi Kriptografi dalam Penyisipan Pesan pada Citra Digital menggunakan Metode Playfair Cipher dan Least Significant Bit (LSB)," 2021.

[5] R. Kristianto Hondro, "Modifikasi Platform Kunci Algoritma Playfair Untuk Meningkatkan Nilai Confusion Pada Ciphertext," *Journal of Computer System and Informatics (JoSYC)*, vol. 1, no. 2, pp. 76–82, 2020.

[6] A. S. Ismaya, G. E. Yuliastuti, and A. Rachman, "Implementasi Metode Modifikasi Playfair Cipher Pada Data Pribadi Stakeholder di SMK Islam Al Futuhiyyah," pp. 1–8, 2023.

[7] G. A. Perdana, Carudin, and Rini Mayasari, "Implementasi Algoritma Kriptografi Playfair Cipher untuk Mengamankan Data Aset," *Jurnal Informatika Polinema*, vol. 7, no. 2, pp. 109–114, 2021, doi: 10.33795/jip.v7i2.394.

[8] P. Pristiwanto, Heri Sunandar, and Berto Nadeak, "Analysis and Implementation of PlayFair Chipper Algorithm in Text Data Encoding Process," *Jurnal Info Sains : Informatika dan Sains*, vol. 10, no. 2, pp. 19–23, 2020, doi: 10.54209/infosains.v10i2.33.

[9] Hermansa, Rusydi Umar, and Anton Yudhana, "Implementation of Playfair Cipher and Least Significant Bit Algorithms in Digital Imagery," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 3, pp. 454–461, 2020, doi: 10.29207/resti.v4i3.1877.

[10] R. Suriadi, R. Satra, and F. Fattah, "Peningkatan Keamanan Data dengan Menggunakan Equation pada Metode Playfair Cipher," *Buletin Sistem Informasi dan Teknologi Islam*, vol. 1, no. 4, pp. 266–269, 2020, doi: 10.33096/busiti.v1i4.685.

[11] E. Prasetyo and Y. F. A. Lubis, "Optimasi Keamanan Hasil Enkripsi Algoritma Playfair Cipher ke dalam Kode Morse," *JiTEKH*, vol. 11, no. 1, pp. 41–50, 2023, doi: 10.35447/jitekh.v11i1.703.

[12] L. Yanti Sipayung, R. Fanry Siahaan, R. Hanum Lubis, D. Novia Amalia, and S. Pelita Nusantara Medan, "Analysis of Data Security Improvement with the 12x8 Matrix Cipher Playfair Algorithm", doi: 10.54209/jurnalinstall.v17i06.407.

[13] G. M. Miguel Manliclic, K. R. Andrei Lamac, R. C. Regala, M. R. Christopher Blanco, and R. M. Dioses, "Improving the Extended 10x10 Polybius Square Key Matrix for Playfair, Bifid, and Polybius Cipher".

[14] I. Putra Sinaga, "Implementasi Kriptografi Hybrid Algoritma Elgamal Dan Double Playfair Cipher Dalam Pengamanan File Jpeg Berbasis Dekstop," 2021. [Online]. Available: https://djournals.com/jieeeJIEEE,

[15] A. A. Siagian and Z. Indra, "Analisis Teknik Playfair dan Shift Cipher sebagai Metode Kriptografi Klasik untuk Keamanan Data," *Jurnal Komputer dan Teknologi*, vol. 4, no. 1, pp. 13–19, Jan. 2025, doi: 10.58290/jukomtek.v4i1.315.

[16] M. Tahir, H. Basri, A. Z. Agustina, N. Nofiyanti, S. P. Kinanti, and A. E. Putra, "Transformasi Digital Enkripsi Teks: Implementasi Playfair Cipher pada Platform Google Colaboratory," *Jutisi : Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 14, no. 2, p. 1080, Aug. 2025, doi: 10.35889/jutisi.v14i2.2794.

6. AUTHORS

Argia Putri Ramadhani is an undergraduate student in the fifth semester of the Smart City Information Systems Program, Faculty of Engineering, Universitas Tunas Pembangunan Surakarta. Email : f0223005_argiaputriramadhani@student.utp.ac.id

Apriliana Vera Ardiyanti is an undergraduate student in the fifth semester of the Smart City Information Systems Program, Faculty of Engineering, Universitas Tunas Pembangunan Surakarta. Email : f0223003_aprilianaveraardiyanti@student.utp.ac.id

Erni Widarti is a Lecturer at the Smart City Information Systems Program, Faculty of Engineering, Universitas Tunas Pembangunan Surakarta. She teaches courses in Cryptography and Programming. Her research interests include data security and software development. Email: erni.widarti@lecture.utp.ac.id