Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th October 2025. Vol. 5. No. 1; e-ISSN: 2808-4519

Design of a Digital Signature Application for Digital Forms Using the Ong-Schnorr-Shamir Algorithm

Raymond Tandi[1]*, Octara Pribadi[2], Jackri Hendrik[3]

[1,2,3]*Informatics Engineering, STMIK Time, Medan, Indonesia*
raymondtandi88@gmail.com[1*], octarapribadi@gmail.com[2], jackri.hendrik@gmail.com[3]

Abstract

Electronic documents (e-documents) are part of public services used to replace paper documents because they have more flexible characteristics, easier searching, save space, digital archiving, easier document transfer, and better security and easy data restoration. In order to maintain and improve data security in a digital document, a mechanism is needed to protect it. The digital signature system is applied to improve the quality of document publishing services. A digital signature on a document means that the signing party has known and approved the document. In today's era when incidents of data destruction and forgery are increasingly common, it has become a necessity to protect any data sent online. For this reason, digital signatures are increasingly popular among professionals thanks to their ability to validate the authenticity of a document, file or software. The method used to create a digital signature in this study is the Ong-Schnorr-Shamir method. The result of this study is a website that is able to produce an formulir digital for a text document. The website also provides facilities to carry out the verification process for the formulir digital to prove the authenticity of the text document in question.

Keywords: *website, e-document, digital form, digital signature, Ong-Schnorr-Shamir method*

1. Introduction

Data plays a vital role in many areas of life and has become even more significant with advancements in computer technology. As digital capabilities expand, more people are able to access and alter data, even when it's securely stored [1]. Electronic documents are now commonly used in public services, replacing traditional paper documents due to their benefits, including flexibility, easy access, space-saving storage, digital archiving, fast transmission, enhanced security, and simple data recovery. To protect these digital documents, robust security measures are essential [2].

Digital signatures are introduced to enhance the quality and trust in the document issuance process. They serve as proof that the signer has reviewed and accepted the document's content [3]. These signatures function by encrypting a hashed version of the document using the sender's private key, thus ensuring that the document remains unchanged. Even the smallest unauthorized edit will be flagged [4]. As a mathematical authentication tool, a digital signature not only confirms the sender's identity but also verifies the origin of the document or message. This makes it a credible means of confirming the authenticity of digital communication [5]. In today's world—where digital tampering and forgery are on the rise—it is crucial to safeguard information shared over networks. This is why digital signatures have become increasingly favored by professionals, as they help validate the integrity and origin of digital assets like documents and software [6].

Various algorithms are used in digital signature development, such as RSA, DSA, and ECDSA. However, this research focuses on the Ong-Schnorr-Shamir method, known for its strong resistance to mathematical and cryptanalytic attacks. It is also resource-efficient, producing smaller signature sizes than some other algorithms like RSA, and supports Zero-Knowledge Proofs, which enable identity verification without exposing sensitive information. This method, developed by H. Ong, C.P. Schnorr, and A. Shamir [7], relies on linear sequential equations and belongs to the field of cryptography. It applies polynomial functions with modulo-n operations, and its security depends on the difficulty of solving these polynomial equations. This study specifically explores a version that uses quadratic polynomials [8].

2. Theoretical Framework

Cryptography

Cryptography is a field that focuses on applying mathematical methods to secure information. It ensures that data is protected by converting it into an unreadable format for unauthorized users, supporting key security goals such as confidentiality, data integrity, and authentication of both the data and its sender or recipient.

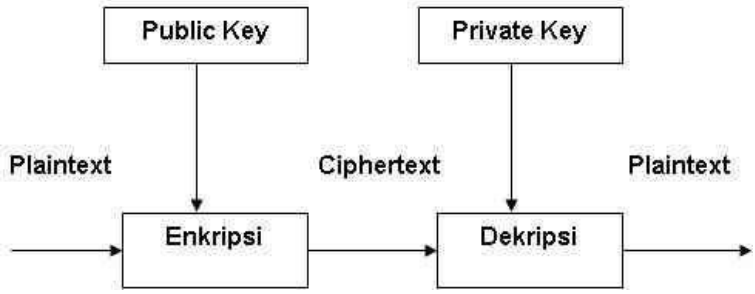

Fig. 1: Cryptographic Process

Digital Signature

A digital signature is a method within cryptography designed to verify that an electronic document or message is both genuine and unchanged. It works through public-key encryption, where the sender signs the document using their private key. The recipient then uses the sender's public key to validate the signature, ensuring the content's authenticity and that it remained intact during transfer [9].

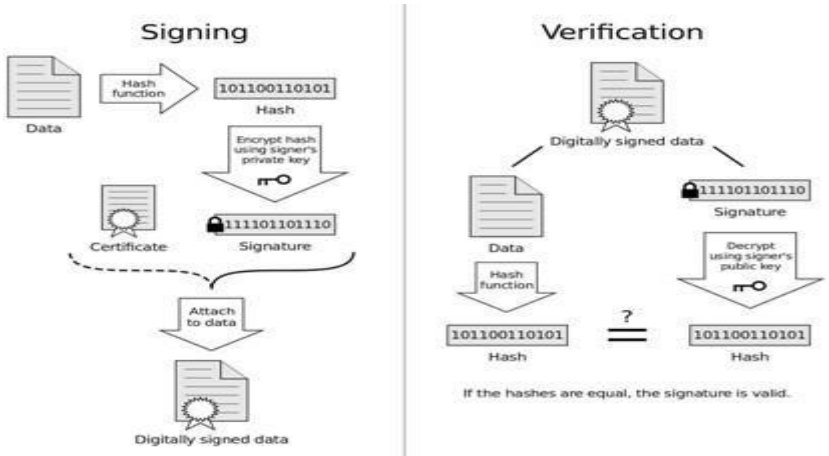

Fig. 2: Digital Signature Mechanism

Ong-Schnorr-Shamir Method

The sender transmits a plaintext message, which is successfully received by the intended recipient without interference.

1. Issue: Is the recipient able to confirm that the message truly came from the sender and that it has not been altered?
2. Resolution: Both the sender and receiver may implement a cryptographic digital signature method, such as the Ong-Schnorr-Shamir scheme, to ensure message authenticity and integrity.

Here is the operational process of the Ong-Schnorr-Shamir digital signature scheme:

1. Identify a large integer (n) and an integer (k) to be used.
   a. The integers n and k must be coprime, which means their greatest common divisor (GCD) equals 1.
   b. The value n serves as the public key, which is openly shared and accessible to anyone. It allows others to verify the authenticity of the digital signature generated by the sender.
   c. The value k functions as the private key, known exclusively to the sender or message originator. This key remains confidential and should never be disclosed, as it is essential for creating the digital signature on the transmitted message.

2. Evaluate the value of h using the equation as follows.

$$h = -(1/k)^2 \bmod n \tag{1}$$

   a. Both h and n serve as public keys, which implies that these values can be accessible to others. Specifically, h is utilized solely during verification, whereas n is involved in both signing and verifying operations.
   b. The key k functions as a private key, which remains exclusively known to the message originator (the sender).

3. Determine a random integer (r) to be used.
   a. The integers n and r need to be coprime, which means their greatest common divisor (GCD) equals 1.

b. r serves as a public key, indicating that this value may be accessible to others.

4. Evaluate S1 and S2 as the sender's signature on message (M) according to the formula provided.

$$S_1 = {}^{1}/_{2} * (M/r + r) \bmod n$$
$$S_2 = {}^{k}/_{2} * (M/r - r) \bmod n \quad (2)$$

a. Both n and r function as public keys, indicating that these values may be accessible to others.
b. M represents the ASCII codes corresponding to the message characters.
c. S1 and S2 constitute the digital signatures.

5. The recipient validates the message from the sender by applying the formula below.

$$S_1^2 + h \cdot S_2^2 \square M \pmod n \quad (3)$$

a. Both n and h function as public keys, which means these values may be accessible to others.
b. M represents the ASCII codes corresponding to the characters in the message.
c. S1 and S2 constitute the digital signatures.

3. Reasearch Method

3.1. Digital Signature Creation

1. Generate Signature : The digital signature is created by applying the user's private key with the OSS algorithm to the document or data. This ensures that only the legitimate private key owner can produce a valid signature, confirming the document's authenticity.
2. Private Key Selection : Users have the option to select which private key they want to use for signing documents. Typically, this key is securely stored in encrypted form either on a server or on the user's device.
3. Data or Document Input : This feature allows users to submit or upload the data they wish to sign, including various formats such as PDFs, text files, or other types of documents.

Below are the typical procedures for creating private and public keys within a cryptographic system that employs the Schnorr algorithm, widely utilized for digital signature applications.

1. Parameter Determination: Before key creation, a set of basic parameters must be identified:
   a. P represents a large prime number utilized to form a cyclic group.
   b. q denotes the order of the group, commonly a prime factor of P minus 1.
   c. g is the generator element within the cyclic group Z/qZ that is used for modular arithmetic operations.

Example: Select a large prime p to establish the group Zp or Identify a generator g in the group that has an order equal to q.

2. Private Key Generation**:** The private key consists of a randomly chosen number within a defined interval. For the Schnorr algorithm, the private key x is selected at random from the interval [1, q – 1].

Example: Randomly select x such that x lies between 1 and q – 1.

3. Public Key Generation: The public key y is derived from the private key x and the chosen parameters. The public key can be computed using the formula: $y = g^x \bmod p$

This means that y is the outcome of raising g to the power of x modulo p, and it belongs to the group generated by g.

4. (Optional) Implementation of Shamir's method for key distribution: The private key x can be segmented into multiple shares using the "Shamir's Secret Sharing" scheme. This approach enables the private key to be distributed among several participants, ensuring the complete key is never stored in a single location.

Steps of Shamir's Secret Sharing:

a. Select the threshold t and the total number of participants n.
b. Employ a random polynomial to split the private key x into multiple shares.
c. Distribute each share to individual participants, ensuring that the complete private key can be reconstructed when at least t participants collaborate.

5. Optional Signature Step**:** If proving control over a public key is the goal, the user may digitally sign a message using their key pair and transmit the signed message for verification.

3.2. Verifying a Digital Signature

1. Digital Signature Verification: This function verifies the authenticity of a digital signature using the matching public key. It ensures the signature is legitimate and that the document remains unchanged during transmission.
2. Verification Status Display: Recipients or users can view the outcome of the verification process—whether the signature is valid or invalid—offering clear feedback and enhancing confidence in the document's authenticity.

4. Results and Discussion

By entering localhost:8000 in your browser, you can run the digital signature system designed for digital forms based on the Ong-Schnorr-Shamir algorithm. The system will then display the main landing page, as depicted in the figure below.

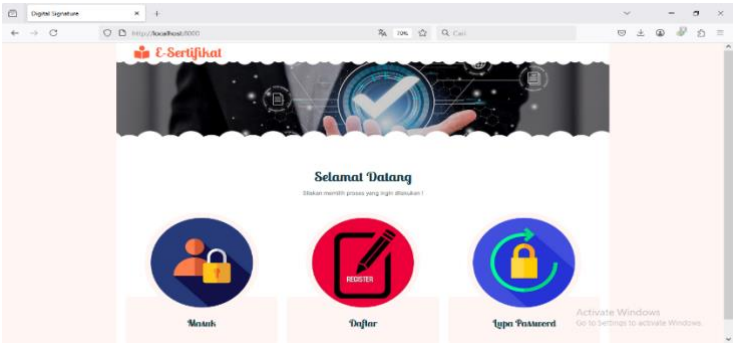

Fig. 3: Main Page Display

The Main Page contains several clickable buttons, including:

1. 'Login' button – used to navigate to the login screen where users can enter their credentials.
2. 'Register' button – opens the registration page, allowing new users to create an account.
3. 'Forgot Password' button – directs users to the password recovery page in case they have lost access.

Users are required to register before they can log into the system. Clicking the 'Register' button takes them to the account creation page.

Users must provide valid data during the registration process. Once the form is completed, clicking the Register button will submit and store the data in the system's database. After successful registration, the user may proceed to log in. To do this, they simply need to click the Login button on the homepage, which will navigate them to the login interface, as illustrated in the image below.

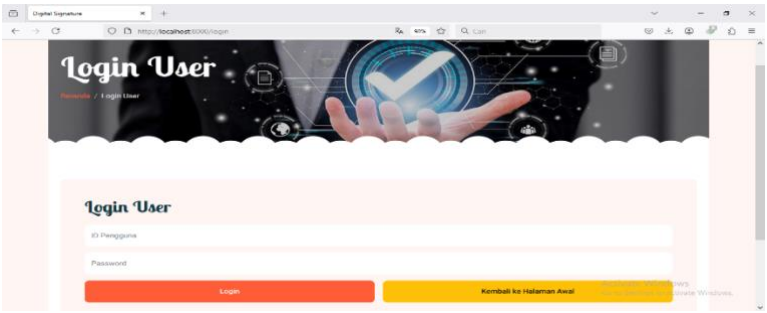

Fig. 4: Home Page Display

Users are required to log in using registered credentials. If the authentication is successful, the system will display the main dashboard page, as shown in the following figure.

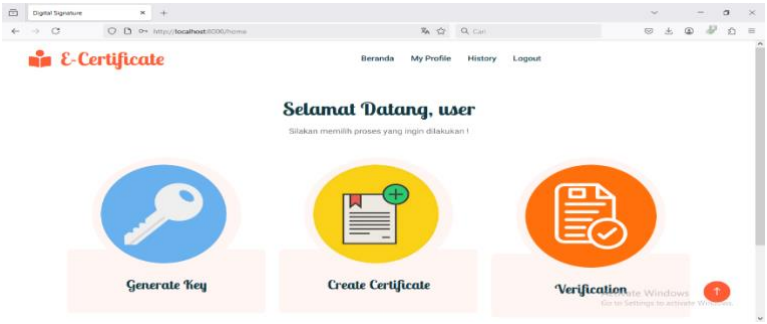

Fig. 5 Main Dashboard Page Display

Users can interact with multiple buttons on the main interface, including:

1. 'Generate Key' button – allows users to navigate to the page where they can generate cryptographic keys.
2. 'Create Certificate' button – directs users to the section for creating digital certificates.
3. 'Verification' button – opens the page for performing signature or document verification.

This page includes several navigation menus in addition to the buttons, which are:

1. 'Home' menu – navigates users back to the dashboard after they log in.
2. 'My Profile' menu – shows detailed information about the user's account.
3. 'History' menu – provides access to the user's past activities within the system.
4. 'Logout' menu – allows users to sign out and return to the initial landing page.

To generate a new set of private and public keys, users can click the ‘Generate Key’ option on the main page, prompting the system to open the Generate Key interface shown in the following figure.

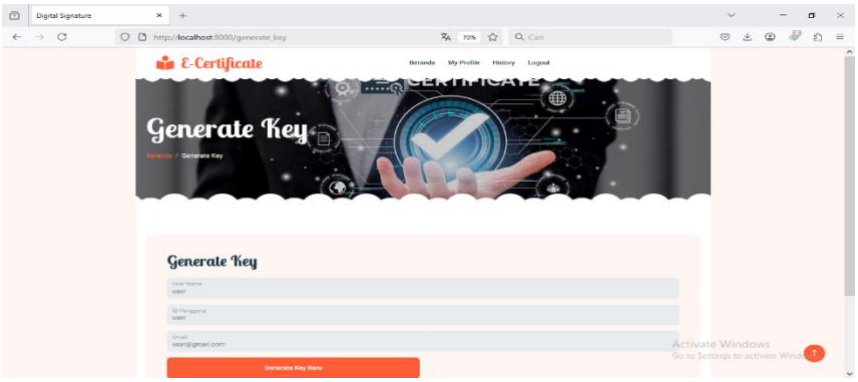

Fig. 6: Generate Key Display

Users can initiate the creation of a new key pair by pressing the Generate New Key button, after which the system generates both private and public keys for the user and then navigates to the History page.

To create a new document certificate, the user can click the Create Certificate button found on the main screen, which will open the Create Certificate interface as shown in the image below.

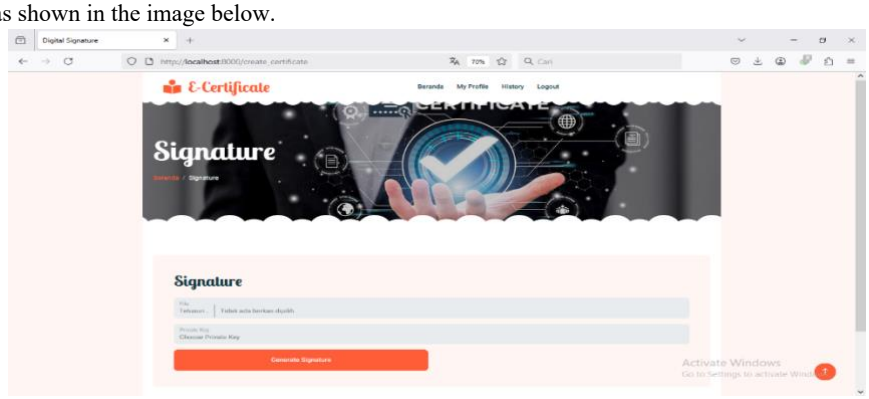

Fig. 7: Create Signature Display

Users can click the Browse button on this page to select a file, prompting the system to show the Open File dialog window.

After choosing a file and opening it, the system reads the file content. The user then selects the appropriate private-public key pair for certificate generation. By clicking Generate Signature, the certificate is produced, and the system presents the Signature Result page with a downloadable certificate file, as displayed in the image below.

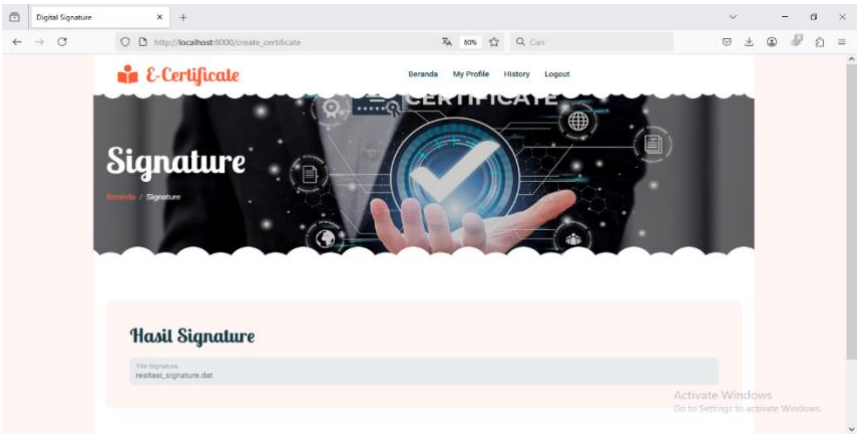

Fig. 8: Result Display

To verify a certificate for a particular document, the user simply clicks the ‘Verification’ option on the main page, leading the system to open the Verification interface shown in the following figure.

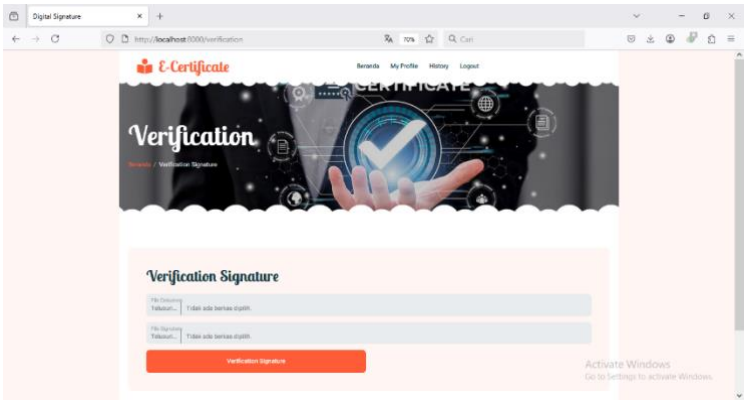

Fig. 9: Verification Display

After selecting the relevant document and certificate files, users can initiate the verification by pressing the Verification Signature button. If the verification succeeds, the system will show the Verification Result page, as illustrated in the image below.

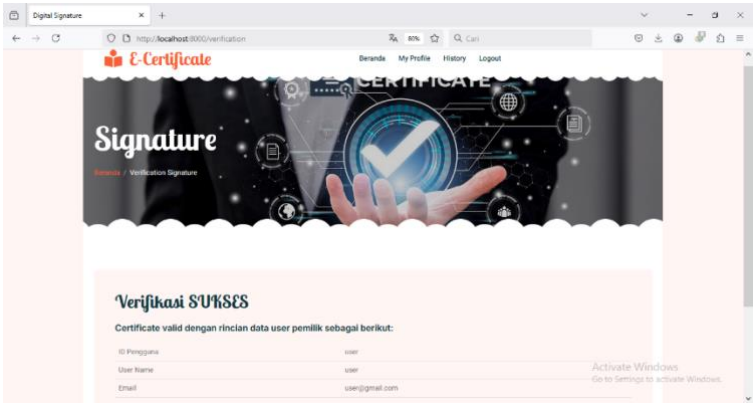

Fig. 11: Success Verification Display

When the document file submitted for verification is different from the one associated with the certificate, the verification will not succeed, and the failure message will appear as shown in the following figure.

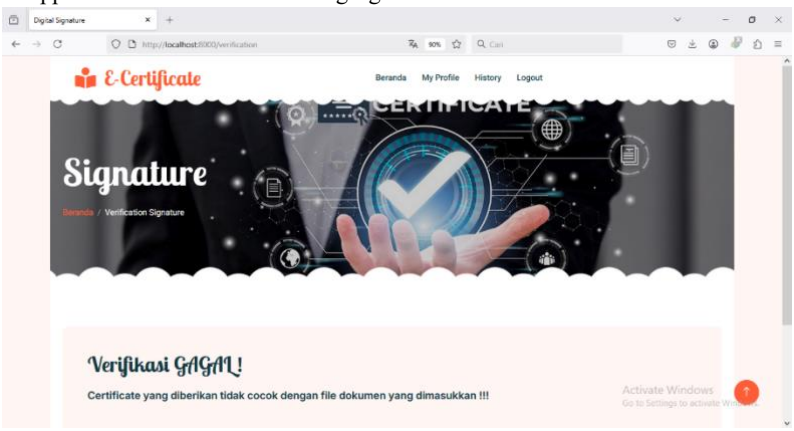

Fig.12: Fail Verification Display

5. Conclusion

Upon finishing the development of the software based on the Ong-Schnorr-Shamir scheme, the author draws these conclusions:

1. The Ong-Schnorr-Shamir digital signature scheme serves as an effective cryptographic technique for producing certificates on document files and facilitates the verification of those digital signatures.
2. The implemented software successfully enables users to append digital signatures to messages or documents.

References

[1] Y. Suharya and H. Widia, "Implementasi Digital Signature Menggunakan Algoritma Kriptografi RSA untuk Pengamanan Data di SMK Wirakarya 1 Ciparay," Jurnal Informatika – COMPUTING, vol. 07, no. 01, pp. 20-29, 2020.

[2] Afrianto, A. Heryandi, A. Finadhita and S. Atin, "Design of E-Document System with Digital Signature using User Centered Design Method," Prosiding Seminar Nasional Teknologi Informasi dan Kedirgantaraan : Peran Teknologi untukRevitalisasi Bandara dan Transportasi Udara, vol. V, pp. 345-356, 2019.

[3] Henderi, D. Rositawati and P. Romansyah, "Model Digital Signature Pada Dokumen Formal Akademik," Program Studi Magister Teknik Informatika Fakultas Sains dan Teknologi Universitas Raharja, vol. 6, no. 1, pp. 22-32, 2020.
[4] N. Sari, D. S. Pribadi, T. Gelar, A. Rahmawati, N. Azzahra and H. Oktoharitsa, "Implementasi Digital Signature pada Laporan Tugas Akhir," JIP (Jurnal Informatika Polinema), vol. 10, no. 1, pp. 99-106, 2023.
[5] M. Taufiqurrahman, Irawan and I. Syamsuddin, "Perancangan Sistem Tanda Tangan Digital (Digital Signature)," Prosiding Seminar Nasional Teknik Elektro dan Informatika (SNTEI) 2020, pp. 60-65, 2020.
[6] E. Wahyuni, S. Rahman and A. Risma, "Keabsahan Digital Signature/Tanda tangan Elektronik Dinjau Dalam Perspektif Hukum Perdata dan UU ITE," Journal of Lex Generalis (JLS), vol. 3, no. 5, pp. 1082-1098, 2022.
[7] M. A. Virgiawan and G. P. Utama, "Penggunaan Metode Ong-Schnorr- Shamir Pada Pembuatan Tanda Tangan Digital," Jurnal Teknik Informatika Unika St. Thomas (JTIUST), vol. 05, no. 01, pp. 51-59, 2020.
[8] D. Adhar, Risman, L. Wahyuni and A. Sabir, "Digital Signature Security Learning Animation Design Using the Ong-Schnorr-Shamir Method," KAPALAMADA: Jurnal Multidisipliner, vol. 3, no. 01, pp. 32-43, 2024.
[9] T. W. Antika Lorien1, "Implementasi Sistem Otentikasi Dokumen Berbasis Quick Response (QR) Code dan Digital Signature," vol. Vol 5 No 4 (2021): Agustus 2021, p. 665, 2021.
[10] D. Ariyu, Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi, Yogyakarta: Penerbit : Andi, 2008, pp. 298-311.
[11] S. M. Samsudin, "Perancangan Sistem Informasi Pembelajaran Algoritma dan Pemrograman," Jurnal Informatika Universitas Pamulang, Vols. Vol. 5, No. 4, pp. 522-523, 2020.
[12] M. W. Khalilurrahman, R. M. Awangga and W. C. Adiwiguno, Pengenalan Golang dan Membuat Package, M. Y. H. Setyawan, Ed., Bandung: PT. Penerbit Buku Pedia, 2023, p. 4.
[13] M. S. Iksanudin, Belajar Santai OOP PHP, Guru Programmer, 2017.
[14] R. Habibi and R. Aprilian, Tutorial dan penjelasan aplikasi e-office berbasis web menggunakan metode RAD, Kreatif, 2020, 2020, pp. 27-28.
[15] B. Apriyanto and A. Agustin, 24 Jam Menguasai Laravel, Jawa Tengah: Eureka Media Aksara, Anggota IKAPI Jawa Tengah No.225/JTE/2021, 2024, p. 1.
[16] Amiruddin, "Tinjauan Awal Penerapan Kriptografi untuk Keamanan," vol. Jilid 4 Nomor 2, p. 562, Tahun 2015.