PEMANFAATAN *DIGITAL SIGNATURE* PADA PENGELOLAAN SISTEM ADMINISTRASI RW

Aprianti Nanda Sari [1], Bambang Wisnuadhi [2], Aldira Ahmad Wibawa [3], Hadi Alfian [4], Rakarizal Muhammad Zidan [5]
[1,2,3,4,5] Jurusan Teknik Komputer dan Informatika, Politeknik Negeri Bandung
aprianti.nanda@polban.ac.id

ABSTRAK

Salah satu tugas Rukun Warga (RW) dan Rukun Tetangga (RT) melaksanakan pelayanan administrasi warga yang ada dalam wilayahnya. Pelayanan administrasi ini meliputi membuat surat pengantar, domisili, dan dokumen administratif lainnya. Mengurus dokumen di tingkat RT secara manual sering kali memiliki beberapa celah keamanan yang dapat disalahgunakan. Beberapa diantaranya adalah pemalsuan dan penyalahgunaan tanda tangan ketua RT dan RW. Untuk mengatasi permasalahan tersebut, penelitian ini mengajukan sebuah metode tanda tangan digital yang merupakan terapan dari teknik kriptografi. Algoritma yang digunakan adalah *Elliptic-Curve Digital Signature Algorithm* (ECDSA). ECDSA dipilih karena komputasi yang cepat namun masih dengan tingkat keamanan yang baik. Selain itu, penelitian ini menggunakan *Secure Hash Algorithm 3* (SHA-3) untuk mengambil nilai hash dari file surat yang berbentuk pdf. Pada penelitian ini terdiri dari dua proses utama yaitu penandatanganan surat dan verifikasi surat. Dari hasil percobaan, metode yang diusulkan dapat menyelesaikan semua skenario pengujian dengan baik sehingga manipulasi data dan penyalahgunaan tanda tangan pengurus RW dapat dihindari.

Keyword : *digital signature, Elliptic-Curve Digital Signature Algorithm, ECDSA, SHA-3, sistem administrasi RW*

1. PENDAHULUAN

Struktur pemerintahan terkecil yang bersentuhan langsung dengan masyarakat adalah Rukun Tetangga (RT) dan Rukun Warga (RW). Pengurus RT dan RW memiliki tugas dalam melayani masyarakat terutama dalam hal administrasi permohonan surat. Permohonan surat ini digunakan masyarakat sebagai rujukan untuk mengurus administrasi ke organisasi pelayanan publik yang lebih tinggi seperti kelurahan atau desa.

Mengurus dokumen di tingkat RT secara manual sering kali memiliki beberapa celah keamanan yang dapat disalahgunakan terutama dalam hal integritas data. Dua kasus yang paling banyak terjadi adalah pemalsuan surat pengantar dan penyalahgunaan tanda-tangan ketua RT dan ketua RW.

Integritas data adalah salah satu aspek pada kemananan informasi yang bertujuan untuk memastikan data tidak dimodifikasi oleh pihak-pihak yang tidak berwenang [1], [2]. Salah satu cara untuk menjaga integritas data adalah dengan menggunakan skema *digital signature* yang merupakan implementasi dari kriptografi asimetri.

Beberapa algoritma kriptografi asimetri yang dapat digunakan untuk skema *digital signature* adalah RSA (Rivest-Shamir-Adleman), ElGamal, dan ECC (*Elliptic-Curve Cryptography*). Pada penelitian ini dipilih skema *digital signature* berbasis ECC atau yang biasa disebut sebagai *Elliptic-Curve Digital Signature Algorithm* (ECDSA). ECDSA dipilih karena memiliki komputasi yang ringan namun masih memiliki tingkat keamanan yang baik [3], [4]. Selain itu, pada penelitian ini juga menggunakan SHA-3 (*Secure Hash Algorithm 3*) sebagai fungsi yang memetakan file surat menjadi nilai hash. SHA-3 dipilih karena memiliki performa yang lebih baik dibanding pendahulunya [5].

2. TINJAUAN PUSTAKA

Pada bagian ini, teori dasar mengenai Sistem Administrasi RW, *digital signature,* ECDSA, dan SHA-3 akan dibahas secara singkat.

2.1. Sistem Administrasi RW

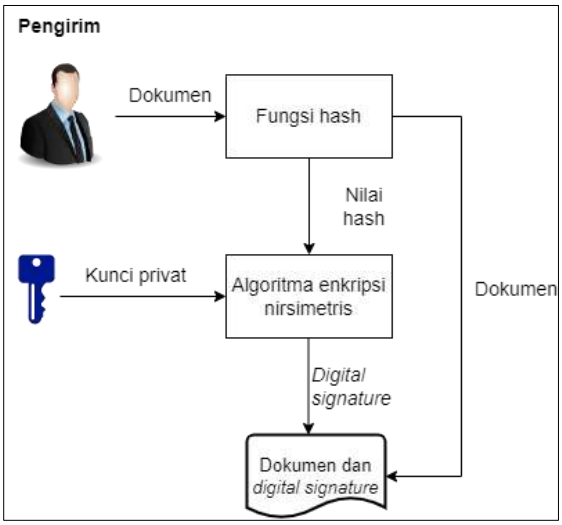
Sistem Administrasi RW merupakan sistem administrasi berbasis web untuk mempermudah pengelolaan data di tingkat RT dan RW. Data yang dikelola meliputi data kependudukan, pembukuan keuangan, dan pengelolaan surat. Pengguna Sistem Administrasi RW terdiri dari admin, Sekertaris RT, Ketua RT, Sekertaris RW, dan Ketua RW.

2.2. *Digital Signature*

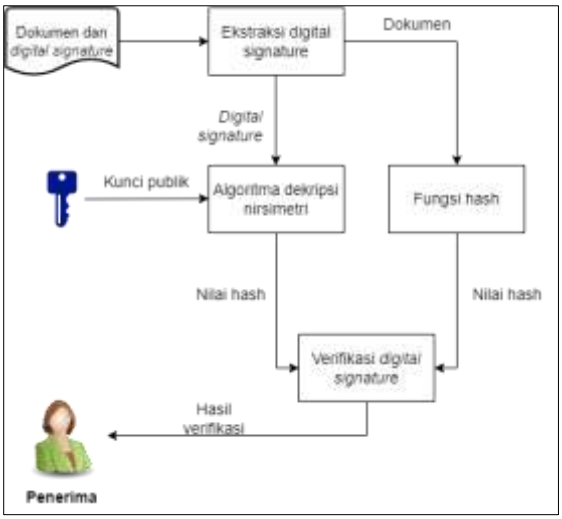
Cara kerja dari *digital signature* mirip dengan tanda tangan pada dunia nyata yaitu memberikan kepastian, autentikasi, dan verifikasi terhadap pengirim/pemilik dokumen serta keabsahan isi dokumen [6].

Secara garis besar, skema *digital signature* memiliki dua proses utama yaitu penandatanganan dokumen dan verifikasi dokumen [7]. Pada proses penandatanganan, dokumen atau pesan yang akan ditandatangani dipetakan oleh fungsi hash menjadi nilai hash. Nilai hash dokumen akan dienkripsi berdasarkan kunci privat milik pengirim sehingga menghasilkan sebuah blok pendek yang disebut

sebagai *digital signature.* Setelah itu, Pengirim kemudian mengirimkan dokumen asli berikut *digital signature.* Ilustrasi dari proses penandatanganan dokumen ini dapat dilihat pada Gambar 1.



Gambar 1. Proses penandatanganan [7]



Gambar 2. Proses verifikasi *digital signature* [7]

Untuk memastikan bahwa dokumen yang diterima belum mengalami perubahan dan dikirim oleh pengirim yang sah, maka proses verifikasi dilakukan. Pertama-tama, *digital signature* dipisahkan dari dokumen. Setelah itu, *digital signature* didekripsi dengan menggunakan kunci publik untuk menghasilkan nilai hash. Pada saat yang bersamaan, dokumen yang diterima juga dipetakan oleh fungsi hash menjadi nilai hash. Jika kedua nilai hash tersebut identik, maka dapat dipastikan bahwa dokumen tidak mengalami perubahan dan dikirim oleh pengirim yang autentik. Proses verifikasi *digital signature* dapat dilihat pada Gambar 2.

2.3. ECDSA

ECDSA adalah sebuah teknik *digital signature* yang memanfaatkan algoritma kriptografi asimetri bernama *Elliptic-Curve Cryptography* (ECC) [8][9]. ECC menggunakan sifat-sifat pada kurva elips dan bekerja pada bilangan bulat. Secara umum, persamaan kurva elips dibuat berdasarkan persamaan

(1).

$$y^2 = x^3 + ax + b \pmod{n} \quad (1)$$

Di mana $a, b \in \mathbb{F}_n$ dengan syarat $4a^3 + 27b^2 \not\equiv 0 \pmod{n}$. Kemudian, buatlah sebuah himpunan titik-titik yang memenuhi persamaan

(1) selain $\mathcal{O}$ (disebut juga *point at infinity*).

Dari himpunan titik tersebut, ambil sebuah titik lalu lakukan operasi yaitu penjumlahan atau perkalian. Misal, dua buah titik $P = (x_1, y_1)$ dan $Q = (x_2, y_2)$ merupakan titik yang ada pada kurva $y^2 = x^3 + ax + b$ maka berlaku aturan penjumlahan sebagai berikut.

Jika $P \neq Q$ dan keduanya bukanlah $\mathcal{O}$ dan hanya jika $x_1 \neq x_2$ maka titik hasil $Q = (x_3, y_3)$ didapat dari persamaan (2) dan (3).

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2 \pmod{n} \quad (2)$$

$$y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) - y_1 \pmod{n} \quad (3)$$

Tetapi jika $x_1 = x_2$ dan $y_1 \neq y_2$, maka titik hasil $Q = (x_3, y_3) = \mathcal{O}$.

Jika $P = Q$ dan bukanlah $\mathcal{O}$ dan $y_1 = y_2 \neq 0$ maka berlaku persamaan $P + Q = 2P$ dengan koordinat hasil (x_3, y_3) didapat dari persamaan (4) dan (5). Namun apabila $y_1 = y_2 = 0$ maka $P + Q = 2P = \mathcal{O}$

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1 \pmod{n} \quad (4)$$

$$y_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)(x_1 - x_3) - y_1 \pmod{n} \quad (5)$$

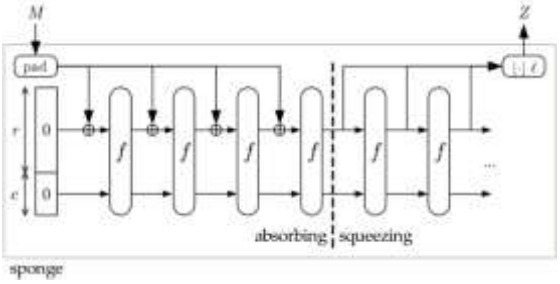
Terakhir, jika titik P dijumlahkan dengan $\mathcal{O}$ maka berlaku persamaan (6).

$$P + \mathcal{O} = P \quad (6)$$

2.4. SHA-3

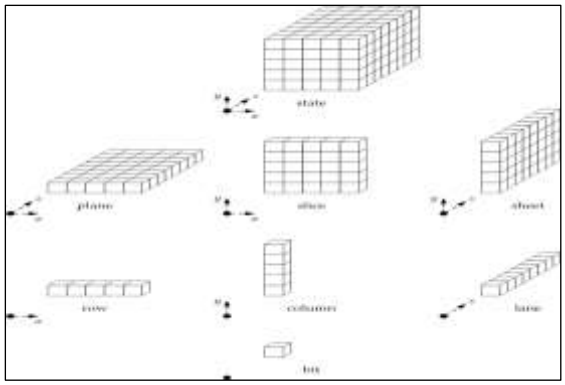
Fungsi SHA-3 atau yang juga dikenal dengan Keccak merupakan pemenang dari kompetisi yang diselenggarakan oleh NIST (*National Institute of*

Standards and Technology). Keccak didesain oleh Guido Breton, Joan Daemen, Michael Peeters, dan Gilles van Assche [10] yang berbasis konstruksi spons (*sponge construction*). Artinya, Keccak menggunakan fungsi non-kompresi untuk menyerap (*absorbing*) dan memeras (*squeezing*) seperti pada Gambar 3.



Gambar 3. *Sponge construction* [11]

Sponge construction memiliki dua parameter yaitu *bitrate* r *dan* kapasitas c [12]. Pada tahap *absorbing*, pesan dibagi menjadi blok berukuran $r + c$ bit lalu dioperasikan dengan $r + c$ bit *state* (biasanya direpresentasikan dengan *array* tiga dimensi seperti pada Gambar 4) dan fungsi f. Fungsi f merupakan fungsi permutasi yang memetakan pesan menjadi string dengan panjang $r + c$ bit yang sama. Tahap ini berlangsung hingga semua blok pesan selesai dioperasikan.



Gambar 4. Ilustrasi *state* [13]

Pada tahap *squeezing*, r bit pertama dari *state* dikembalikan sebagai blok output lalu disisipkan fungsi f. Banyaknya blok output yang digunakan dapat ditentukan oleh pengguna.

3. METODE PENELITIAN

Penelitian ini terdiri dari studi literatur, analisis, perancangan, implementasi, serta pengujian. Pada bagian ini akan dibahas analisa dan perancangan. Sedangkan tahap implementasi dan pengujian dibahas pada bab 4 dan 5.

3.1. Analisis

Analisis yang dilakukan terdiri dari dua jenis yaitu analisis sistem yang berjalan dan analisis sistem yang akan dibangun.

3.2. Analisis Sistem yang Berjalan

Analisis sistem yang berjalan bertujuan untuk mengetahui prosedur dan *stakeholder* sebagai dasar untuk menganalisa sistem yang akan dibangun agar sesuai dengan kebutuhan.

3.3. Analisis prosedur

Dari hasil observasi, disimpulkan bahwa proses pengelolaan surat dibagi menjadi dua yaitu pengelolaan surat di tingkat RT dan pengelolaan surat di tingkat RW.

Proses pengelolaan surat di tingkat RT dimulai saat warga mengajukan permohonan surat kepada Sekertaris RT. Jika data diri dan syarat sudah dipenuhi warga, maka Sekertaris RT akan membuatkan surat dan meminta Ketua RT untuk menandatanganinya. Setelah ditandatangani, surat akan diberikan kepada warga dan nomor surat diarsipkan.

Surat yang sudah ditandatangani oleh Ketua RT kemudian dibawa oleh warga untuk diproses di tingkat RW. Surat tersebut akan diterima terlebih dahulu oleh Sekertaris RW sebelum diberikan kepada Ketua RW untuk ditandatangani. Setelah ditandatangani oleh Ketua RW, surat akan diberikan kepada warga dan nomor surat diarsipkan

3.4. Analisis *stakeholder*

Dari analisis prosedur yang dilakukan, terdapat lima *stakeholder* yang terlibat. Masing-masing *stakeholder* memiliki perannya yang dapat dilihat pada

Tabel 1.

Tabel 1 Peran *stakeholder* pada sistem yang berjalan

Stakeholder	**Peran**
Warga	1. Mengisi data diri perihal pengajuan surat 2. Melampirkan dokumen persyaratan 3. Menerima surat yang telah diberi tanda tangan oleh Ketua RT dan Ketua RW
Sekertaris RT	1. Melakukan pendataan permohonan surat untuk warga 2. Memastikan bahwa warga yang dilayani adalah penduduk wilayah RT/RW setempat 3. Membuatkan surat 4. Memberikan surat kepada Ketua RT untuk disahkan dan diberi tanda tangan 5. Melakukan arsip nomor surat pada buku agenda surat keluar RT
Ketua RT	1. Memberikan persetujuan terhadap surat

Stakeholder	**Peran**
	2. Memberikan tanda tangan
Sekertaris RW	1. Memberikan surat yang dikeluarkan oleh Ketua RT kepada ketua RW untuk diberi tanda tangan 2. Melakukan arsip nomor surat pada buku agenda surat keluar RW
Ketua RW	Memberikan tanda tangan pada surat sebagai tanda mengetahui atau mengerti arti dari surat tersebut

3.5. Analisa Sistem yang Dibangun

Pada bagian in dijelaskan analisis sistem yang dibangun meliputi analisis *stakeholder* dan kebutuhan fitur. Hasil dari analisis menghasilkan kebutuhan aplikasi untuk menyelesaikan masalah yang ada.

Tabel 2. Sekertaris RW tidak menjadi *stakeholder* pada sistem yang akan dibangun karena peran Sekertaris RW untuk mengarsipkan surat dapat digantikan oleh sistem. Sedangkan warga yang mengajukan permintaan dan pihak lain yang ingin mengecek keaslian surat digabungkan menjadi *stakeholder* Penerima Surat.

3.6. Analisis *stakeholder*

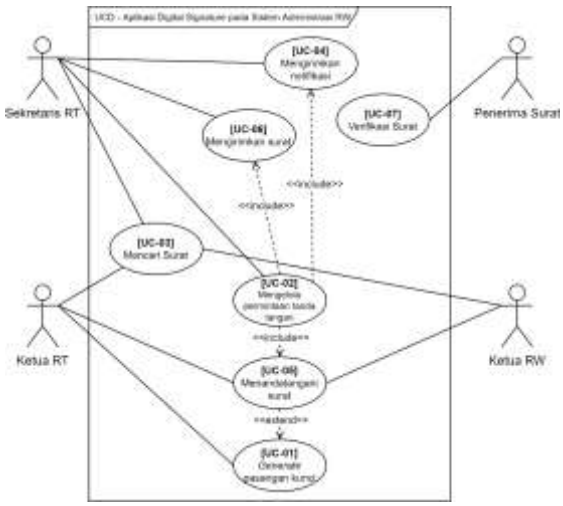
Pada sistem yang akan dibangun, terdapat empat *stakeholder* yang terlibat yaitu Sekertaris RT, Ketua RT, Ketua RW, dan Penerima surat. Penjelasan mengenai *stakeholder* pada sistem yang akan dibangun dijelaskan pada

Tabel 2. Peran *stakeholder* pada sistem yang dibangun

Stakeholder	**Peran**	***Interest***
Sekertaris RT	Sekertaris RT memiliki tanggung jawab atas perekapan surat dan pembuatan surat.	Sekertaris RT menginginkan data surat yang telah ditandatangani dapat direkap pada agenda surat keluar RT
Ketua RT	Ketua RT memiliki peran memberikan tanda tangan pada surat untuk menunjukkan bahwa Ketua RT menyetujuan surat tersebut	Ketua RT menginginkan memberi tanda tangan surat secara aman sehingga penyalahgunaan tanda tangan dapat dihindari
Ketua RW	Ketua RW memiliki peran memberikan tanda tangan pada surat yang telah disetujui ketua RT untuk menunjukkan bahwa Ketua RW mengetahui surat tersebut	Ketua RW menginginkan memberi tanda tangan secara aman sehingga penyalahgunaan tanda tangan dapat dihindari
Penerima surat	Penerima surat memiliki peran untuk mendapatkan surat yang telah ditandatangani Ketua RT dan Ketua RW	Penerima surat menginginkan surat tidak dimodifikasi dan dapat memastikan siapa yang telah menyetujui (memberi tanda tangan) pada surat

3.7. Analisis kebutuhan fitur

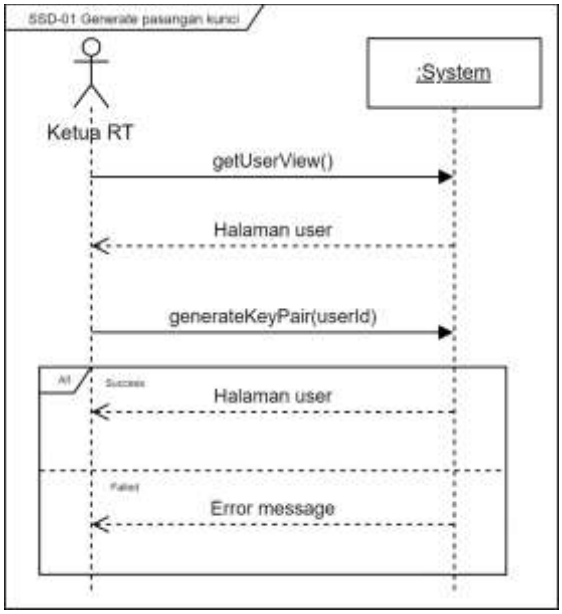
Kebutuhan fitur pada sistem yang akan dibangun digambarkan pada *use case diagram* seperti pada Gambar 5.



Gambar 5. *Use case diagram* sistem yang akan dibangun

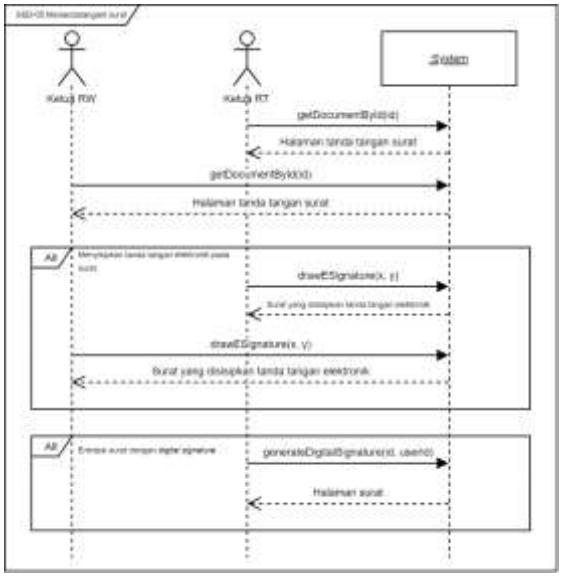
Pada sistem yang akan dibangun, akan ada tujuh fitur utama yaitu *Generate* pasangan kunci (UC-01), Mengelola permintaan tanda tangan (UC-2), Mencari surat (UC-03), Mengirimkan notifikasi (UC-04), Menandatangani surat (UC-05), Mengirimkan surat (UC-06), dan Verifikasi surat (UC-07). Namun pada artikel ini hanya akan dijelaskan lebih detail tiga fitur utama yaitu UC-01, UC-05, dan UC-07.

Pada UC-01 yaitu *Generate* pasangan kunci, Ketua RT dapat menekan tombol "*Generate* kunci*"* untuk membangkitkan sepasang kunci privat dan publik dengan metode ECDSA dan menyimpan kunci tersebut dalam *database.* Detail proses *Generate* pasangan kunci diilustrasikan dengan *System sequence diagram* yang dapat dilihat pada Gambar 6.



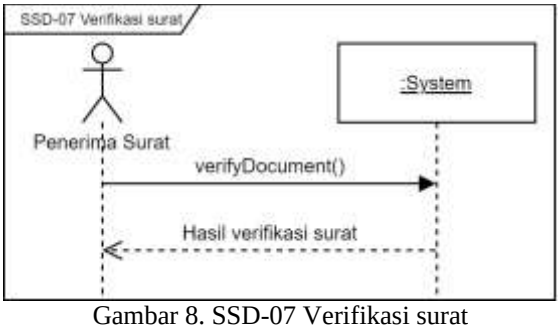
Gambar 6. SSD-01 *Generate* pasangan kunci

Untuk menandatangani surat, Ketua RT atau ketua RW memilih surat yang akan ditandatangani dan menggambar tanda tangan pada bagian yang telah disediakan. Sistem kemudian akan menempelkan gambar tanda tangan pada surat dan membangkitkan *digital signature* dengan mengenkripsi nilai hash surat dengan menggunakan kunci privat penandatangan. Hasil enkripsi tersebut kemudian disisipkan sebagai metadata pada surat dan mengarsipkan surat yang sudah ditandatangani. *System sequence diagram* dari UC-05 dapat dilihat pada Gambar 7.



Gambar 7. SSD-05 Menandatangani surat

Penerima surat (warga atau pihak lainnya) dapat mengecek keabsahan surat melalui proses verifikasi. Penerima surat cukup mengunggah surat yang akan diverifikasi dan hasil verifikasi akan dikembalikan kepada penerima surat seperti pada Gambar 8.



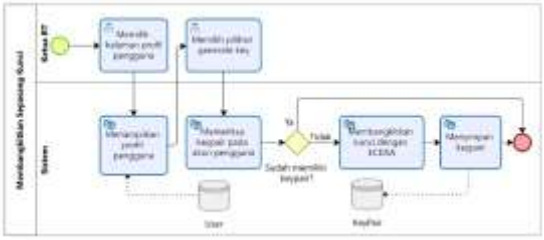
Gambar 8. SSD-07 Verifikasi surat

3.8. Perancangan Sistem

Setelah dianalisis, sistem yang dibangun kemudian dirancang. Perancangan yang dimaksud meliputi perancangan proses bisnis dan perancangan *database*.

3.9. Perancangan Proses Bisnis

Proses bisnis pada sistem yang dibangung meliputi tujuh fitur. Namun pada artikel ini hanya akan dibahas tiga fitur utama yaitu *generate* pasangan kunci, menandatangani surat, dan verifikasi surat seperti pada Gambar 9, Gambar 10, dan Gambar 11.



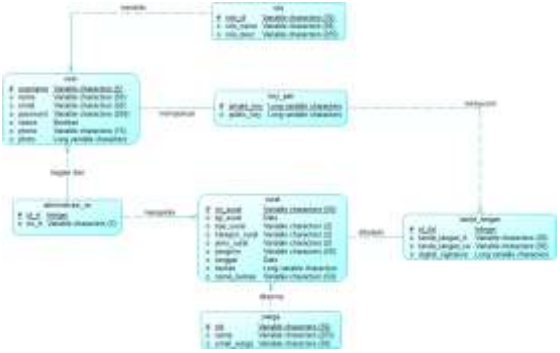
Gambar 9. Proses bisnis generate pasangan kunci

Gambar 10. Proses bisnis menandatangani surat

Gambar 11. Proses bisnis verifikasi surat

3.10. Perancangan Database

Berdasarkan hasil analisis sistem yang akan dibangun, terdapat kebutuhan untuk menyimpan data. Oleh karena itu, dilakukan perancangan *database* yang direpresentasikan dalam *physical data model* seperti pada Gambar 12.



Gambar 12. *Physical data model* sistem yang dibangun

4. HASIL DAN PEMBAHASAN

Pada bagian ini menjelaskan tampilan sistem hasil implementasi *digital signature* pada Sistem Administrasi RW. Ketua RT dan Ketua RW sebagai pihak yang memberikan tanda tangan dapat mengakses tombol "*generate key*" untuk membangkitkan pasangan kunci privat dan publik seperti pada Gambar 13.

Gambar 13. Halaman profil untuk mengakses tombol "*generate key*"

Jika ada permintaan surat dari warga, Sekertaris RT akan *login* dan mengunggah surat baru, mengisi detail, dan mengisi pihak yang akan menandatangani seperti pada Gambar 14.

Gambar 14. Halaman tambah permintaan tanda tangan

Sebagai contoh, pihak yang diminta tanda tangan adalah Ketua RT. Maka, Ketua RT akan login dan menampilkan pratinjau surat yang akan ditandatangani seperti pada Gambar 15. Selain mengimplementasikan skema *digital signature*, sistem yang dibangun juga dilengkapi fitur penyisipan gambar tanda tangan dengan *input mouse*, *stylus*, atau gambar pindai tanda tangan seperti pada Gambar 16.

Gambar 15. Halaman pratinjau surat yang akan ditandatangani

Gambar 16. Halaman menggambar tanda tangan

Setelah dibubuhkan gambar tanda tangan, surat akan diproses dengan skema *digital signature* dengan metode ECDSA. Identitas surat dan *digital signature* kemudian akan disimpan dalam *database* untuk diarsipkan. Contoh surat yang sudah ditandatangani dapat dilihat pada Gambar 17.

Gambar 17. Contoh surat yang sudah ditandatangani

File surat yang sudah ditandatangani kemudian akan dikirim ke alamat *email* warga atau penerima surat lainnya. Untuk memastikan keaslian surat, penerima surat dapat melakukan verifikasi dengan cara mengunggah dokumen surat. Hasil verifikasi

surat kemudian akan muncul seperti pada tampilan di Gambar 18.

Gambar 18. Halaman verifikasi surat

Setelah diimplementasikan, sistem yang dibangun kemudian diuji agar mengetahui ketercapaian tujuan penelitian. Pengujian dilakukan dengan metode *black-box testing* terhadap fungsionalitas sistem. Hasil pengujian dapat dilihat pada

Tabel 3.

Tabel 3. Hasil pengujian

No	**Pengujian**	**Hasil yang diharapkan**	**Keterangan**
1	Menyisipkan gambar tanda tangan pada surat	Tanda tangan berada pada tempat yang diinginkan	Berhasil
2	Menggambar tanda tangan pada kolom yang disediakan	Gambar tanda tangan tersimpan	Berhasil
3	*Generate* pasangan kunci	Muncul *pop up* berhasil	Berhasil
4	Mengekstraksi nilai hash surat	Muncul *pop up* berhasil	Berhasil
5	Melakukan verifikasi surat dengan mengunggah dokumen yang valid	Muncul tulisan "Dokumen valid"	Berhasil
6	Melakukan verifikasi keaslian surat dengan mengunggah surat yang tidak valid	Muncul peringatan "Dokumen tidak valid"	Berhasil
7	Mengirim notifikasi kepada pihak yang menandatangani surat melalui *email*	Muncuk *pop up* berhasil	Berhasil
8	Mengirim surat yang telah ditandatangani kepada penerima melalui *email* yang valid	Muncul *pop up* berhasil	Berhasil
9	Mengirim surat yang telah ditandatangani kepada penerima melalui *email* yang tidak valid	Muncul notifikasi "*Your email address in the format: email@example.com*"	Berhasil
10	Mengunduh surat yang belum ditandatangani	Dokumen terunduh	Berhasil
11	Mengunduh surat yang sudah ditandatangani	Dokumen terunduh	Berhasil
12	Menampilkan daftar surat yang belum ditandatangani	Muncul daftar surat dengan status "Menunggu Tanda Tangan"	Berhasil
13	Menampilkan daftar surat berdasarkan kategori tertentu	Muncul daftar surat sesuai dengan kategori yang dipilih	Berhasil
14	Menampilkan daftar surat berdasarkan masukan *user*	Muncul daftar surat sesuai dengan masukan *user*. Daftar surat kosong jika tidak ada kata kunci yang cocok	Berhasil
15	Menyimpan data surat yang membutuhkan tanda tangan dengan data yang valid	Muncul *pop up* berhasil	Berhasil
16	Menyimpan data surat yang membutuhkan tanda tangan dengan data yang tidak valid	Muncul peringatan pada *field* yang diisi tidak sesuai format	Berhasil
17	Memperbaharui surat yang membutuhkan tanda tangan	Muncul *pop up* berhasil	Berhasil
19	Menghapus surat yang membutuhkan tanda tangan	Muncul *pop up* berhasil	Berhasil

5. KESIMPULAN DAN SARAN

Dari hasil implementasi, sistem yang dibuat dapat membantu Ketua RT dan Ketua RW untuk mengurus surat yang memerlukan tanda tangan serta dapat membantu Sekertaris RT melakukan rekap surat keluar. Selain itu, sistem yang dibangun dapat memberikan kemananan dan menjamin keaslian surat dengan penerapan *digital signature* dengan kombinasi algoritma ECDSA dan SHA-3.

DAFTAR PUSTAKA

[1] S. Duggineni, "Data Integrity and Risk," *Open Journal of Optimization*, vol. 12, no. 02, pp. 25–33, 2023, doi: 10.4236/ojop.2023.122003.

[2] A. N. Sari, "Securing graph steganography by using Shamir's secret sharing," in *IOP Conference Series: Materials Science and Engineering*, Institute of Physics Publishing, May 2020. doi: 10.1088/1757-899X/830/3/032022.

[3] J. Groth and V. S. Dfinity, "On the security of ECDSA with additive key derivation and presignatures," 2022.

[4] Y. Genc and E. Afacan, "Design and implementation of an efficient elliptic curve digital signature algorithm (ECDSA)," in *2021 IEEE International IOT, Electronics and Mechatronics Conference, IEMTRONICS 2021 - Proceedings*, Institute of Electrical and

Electronics Engineers Inc., Apr. 2021. doi: 10.1109/IEMTRONICS52119.2021.9422589.
[5] A. Dolmeta, M. Martina, and G. Masera, "Comparative Study of Keccak SHA-3 Implementations," Dec. 01, 2023, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/cryptography7040060.
[6] L. B. Sihombing, "Keabsahan Tanda Tangan Elektronik Dalam Akta Notaris," *Jurnal Education and development*, vol. 8, no. No. 1, p. Hal. 134, 2020.
[7] A. N. Sari, D. S. Pribadi, T. Gelar, A. Rahmawati, N. Azzahra, and H. Oktoharitsa, "Implementasi Digital Signature pada Laporan Tugas Akhir," *Jurnal Informatika Polinema*, vol. 10, no. 1, 2023.
[8] D. Johnson, A. Menezes, and S. Vanstone, "The Elliptic Curve Digital Signature Algorithm (ECDSA)," *Int J Inf Secur*, vol. 1, no. 1, pp. 36–63, 2001, doi: 10.1007/s102070100002.
[9] B. N. Koblitz, "Elliptic Curve Cryptosystems," vol. 4, no. 177, pp. 203–209, 1987.
[10] G. Bertoni, M. Peeters, and G. Van Assche, "The keccak reference," 2011.
[11] R. V. K. Guido Bertoni, Joan Daemen, Seth Hoffert, Michaël Peeters, Gilles Van Assche, "The sponge and duplex constructions."
[12] M. J. Kannwischer, P. Pessl, and R. Primas, "Single-trace attacks on keccak," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2020, no. 3, pp. 243–268, 2020, doi: 10.13154/tches.v2020.i3.243-268.
[13] G. Bertoni, J. Daemen, and G. Van Assche, "Keccak specifications," pp. 1–7, 2009