Faktor Exacta

Vol. 16, No. 3, Sept 2023, pp. 182~189
eISSN: 2502-339X, pISSN: 1979-276X, DOI: 10.30998/faktorexacta.v16i3.17314

Terapan Metode *Least Significant Bit* untuk Deteksi Keaslian *e-certificate*

Aulia Paramita[1], Akbar Muchbarak[2], Aprilia Sulistyohati[3], Alusyanti Primawati[4]
[1,2,3,4]Department of Informatic, Universitas Indraprasta PGRI, Indonesia

Article Info

Article history:

Received Apr 26, 2023
Revised Jul 03, 2023
Accepted Aug 05, 2023

Keywords:

Data Security
Steganography
Least Significant Bit
Website Applications

ABSTRACT

The ease of accessing digital information allows someone to change or manipulate the data contained in the information. Therefore, the security of information or important data from intruders or unauthorized access is important. One case that is currently rife is the spread and falsification of information through fake e-certificates. The purpose of this research is to increase security, as well as check the authenticity and validity of data on e-certificates using steganography techniques with the Least Significant Bit (LSB) method. LSB facilitates the insertion of secret messages with fewer pixels of distortion than other steganography methods. However, LSB is more easily detected by steganalysis. The weakness of LSB becomes an advantage to solve the problem of e-certificate authenticity. The image on the certificate in the form of a pdf file will be inserted with information to validate whether the certificate is genuine or not, by reading the pixels in the image in the file. In this research, the data to be inserted is a checksum value form of 32 hexadecimal characters and also an e-certificate information form a JSON string. The result of this research is a web-based application that is able to check the authenticity of e-certificates.

Corresponding Author:

Alusyanti Primawati,
Department of Informatic,
Universitas Indraprasta PGRI,
Jl. Nangka No. 58 C, Tanjung Barat, Jagakarsa, Jakarta Selatan.
Email: alus.unindra23@gmail.com

1. PENDAHULUAN

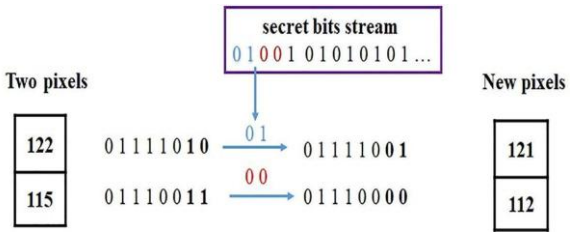
Kondisi global beberapa tahun belakangan ini memaksa para pelaku di bidang bisnis, ekonomi, sosial bahkan pendidikan harus memanfaatkan media digital untuk mengolah data ataupun mendapatkan sebuah informasi. Mudahnya mengakses informasi digital membuat seseorang dapat mengubah atau memanipulasi data yang terdapat pada sebuah informasi. Oleh karena itu, keamanan informasi atau data vital dari penyusup atau akses tidak sah menjadi penting [1]. Salah satu kasus yang saat ini sering terjadi yaitu peredaran dan pemalsuan informasi melalui *e-certificate* palsu. Akibatnya sertifikat tidak dapat lagi digunakan untuk membuktikan keahlian/ketrampilan yang telah dicapai/dimiliki oleh seseorang [2]. Maraknya pemalsuan informasi pada *e-certificate* menandakan bahwa keamanan *e-certificate* perlu dikembangkan, supaya pengiriman informasi dapat dilakukan dengan pengamanan terhadap isi informasi yang dikirim. Pengamanan informasi bisa dilakukan dengan berbagai teknik, salah satunya dengan menggunakan steganografi.

Steganografi merupakan sebuah teknik menyembunyikan informasi/data rahasia dalam media digital sehingga keberadaan informasi/data rahasia tersebut tidak diketahui/sulit dideteksi oleh pihak yang tidak ditujukan. Steganografi berfungsi untuk menyamarkan keberadaan informasi/data rahasia sehingga sulit dideteksi dan juga dapat digunakan untuk melindungi hak cipta dari suatu produk [3]. Steganografi menyembunyikan pesan dalam data lain tanpa mengubah data yang ditumpanginya tersebut sehingga data yang ditumpanginya sebelum dan setelah proses penyembunyian hampir terlihat sama [4].

Penelitian ini membahas pengembangan dari permasalahan keamanan *e-certificate* dengan menggunakan steganografi. Teknologi pengamanan *e-certificate* dapat dilakukan dengan menggunakan teknik steganografi. Secara umum proses steganografi menggunakan kunci sebagai sarana kepemilikan dan keamanan. Kunci ini diperlukan untuk dapat membuka pesan yang disisipkan melalui *encode*r yang berisi algoritma penyisipan steganografi ke dalam media [5]. Steganografi memanfaatkan media digital untuk menyisipkan informasi/data rahasia melalui kode biner pada media digital teks, audio, *image* maupun video. Penelitian ini diharapkan mampu meningkatkan keamanan, keaslian dan keabsahan data pada *e-certificate*.

Ada beberapa penelitian sebelumnya yang membahas mengenai steganografi, salah satunya penelitian yang dilakukan yaitu membandingkan antara 2 metode steganografi yaitu *Least Significant Bit* (LSB) dan *Discrete Cosine Transform* (DCT) [4]. Penelitian dilakukan dengan menghitung rasio *Peak Signal to Signal Ratio* (PSNR). Rasio ini digunakan sebagai ukuran kualitas antara 2 gambar, jika rasio PSNR tinggi maka kualitas gambar lebih baik. Hasil dari penelitian menunjukkan bahwa steganografi berbasis DCT, rasio PSNR nya lebih tinggi dibandingkan berbasis LSB untuk semua jenis gambar baik skala abu dan warna. Penelitian sejenis dilakukan perihal keamanan penyembunyian data pada *image* menggunakan steganografi LSB dengan cara memotong *image* [5]. Hasil dari penelitian ini yaitu metode LSB terbukti memiliki tingkat keamanan yang tinggi dibanding metode yang lain. Metode steganografi yang lain seperti menyisipkan *text* ke dalam *image* dengan menggunakan metode LSB *watermarking* [6]. Hasil dari penelitian ini yaitu dengan menggunakan algoritma *watermarking* mampu mempertahankan kualitas citra dari *gaussian, poisson, salt* dan *pepper*, serta *speckle*. Hasil terbaik didapatkan saat penyisipan *watermark* pada bit ke-8 dalam *image*, hanya saja pada penelitian ini ukuran *watermark* sangat kecil, sehingga untuk penelitian selanjutnya bisa menggunakan *watermark* ukuran besar.

LSB merupakan salah satu alat steganografi dengan cara yang sederhana yaitu menyisipkan data ke dalam gambar dengan teknik subtitusi dan pencocokan [7]. Teknik subtitusi LSB bekerja dengan menyematkan informasi rahasia pada bit paling kanan sehingga distorsi piksel terjadi lebih sedikit ditunjukkan Gambar 1. Keunggulan dari teknik subtitusi LSB lebih mudah diidentifikasikan daripada pencocokan LSB.



Gambar 1. Contoh Teknik Subtitusi LSB [8]

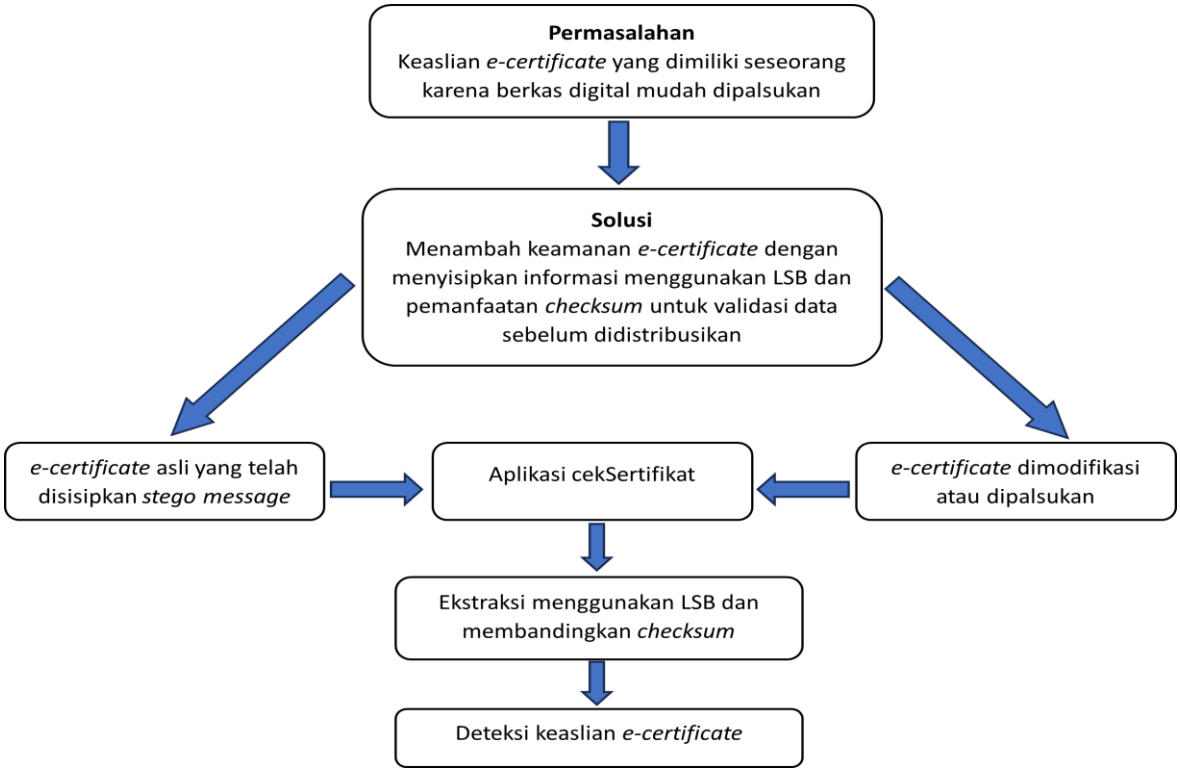
LSB memiliki kelemahan dari isu keamanan pesan yaitu mudah dideteksi oleh steganoanalisis. Pesan yang disembunyikan kedalam citra dengan teknik LSB akan mudah diketahui pihak ketiga. Namun, para peneliti memberikan konsentrasi lebih untuk menghadirkan skema LSB yang kuat berdasarkan kriptografi dan steganografi yang dapat menghindari serangan steganalisis tersebut [9].

Kelemahan LSB dapat berubah menjadi keunggulan pada isu banyak pemalsuan berkas digital. Salah satu berkas digital yang saat ini banyak dihasilkan dari berbagai pelatihan atau seminar *online* adalah *e-certificate*. *E-certificate* bermanfaat untuk dilampirkan seseorang pada biodata pribadi untuk menambahkan nilai kemampuan yang dimiliki oleh si pemilik sertifikat. Permasalahan yang muncul adalah keaslian *e-certificate* yang dimiliki seseorang karena berkas digital mudah dipalsukan atau diedit menggunakan *software* pengolah *image*. Dengan demikian penyelenggara yang mengeluarkan *e-certificate* perlu menyematkan pesan tersembunyi sebagai tanda keaslian dari *e-certificate* yang dikeluarkan. Solusi dari permasalahan pemalsuan *e-certificate* yaitu penerapan analisis citra stegano dengan teknik LSB. Pihak penyelenggara akan menyediakan satu buah wadah untuk pemeriksaan keaslian *e-certificate* tersebut. Salah satu mekanisme deteksi berkas asli dan palsu adalah melihat jumlah *checksum* pesan dari berkas asli dan berkas yang diuji keasliannya [10]. Pemeriksaan ini harus bisa memvalidasi apakah informasi yang tersembunyi sudah diubah oleh orang lain. Selain itu juga harus dapat memeriksa citra gambar yang ada pada *e-certificate* tersebut apakah sudah terjadi modifikasi ataukah belum.

2. METODE

Metode yang kami gunakan adalah pendekatan ekperimental terhadap teknik LSB untuk menyelesaikan permasalahan pemalsuan *e-certificate*. Tahapan penelitian dijelaskan melalui kerangka berpikir (lihat Gambar 2). Tahap pertama, kami menyiapkan modul proses *encode* dan *decode* dengan menggunakan

teknik LSB. Kedua, merancang aplikasi berbasis *website* dan memasukan modul tahap pertama. Tahap ketiga, kami mempersiapkan gambar pada sertifikat dalam bentuk file png akan disisipkan informasi berupa nama kegiatan, nomor sertifikat, nama pemilik sertifikat dan tanggal pengesahan. Tujuannya adalah untuk memvalidasi sertifikat tersebut asli atau tidak, dengan cara membaca *pixel* yang ada dalam gambar pada file tersebut. Jika hasilnya *valid*, maka akan muncul informasi yang terdapat di dalam sertifikat tersebut. Tidak hanya *e-certificate*, kami juga menyiapkan sertifikat yang belum dimasukan *stego message* (kami mengklasifikasikan bukan *e-certificate* yang valid).



Gambar 2. Kerangka Berpikir

LSB akan menyisipkan bit data ke dalam masing-masing warna di bit paling akhir yaitu 1 bit pada warna merah, 1 bit pada warna hijau, dan 1 bit pada warna biru. Karena disisipkan hanya pada bit terakhir di setiap warna, maka secara kasat mata tidak akan terlihat perbedaan warnanya, bahkan hampir tidak terlihat perubahan warnanya. Pada penelitian ini, data yang akan disisipkan yaitu berupa nilai *checksum* dalam bentuk 32 karakter *hexadecimal* dan juga informasi *e-certificate* dalam bentuk string JSON (Java Script Object Notation). Pada gambar 3 disajikan struktur data di dalam JSON-nya. Setelah dihitung panjang informasi *e-certificate*, kemudian akan dilakukan perhitungan *checksum* mengunakan MD5 dari informasi tersebut dan juga dari keseluruhan *binary* gambarnya. Tujuan dihitung *checksum* untuk digunakan nantinya pada proses pemeriksaan keaslian dari sertifikat tersebut. Jadi, jika terjadi proses *edit* pada gambarnya atau dikompresi tentu saja akan merubah *binary* gambar tersebut. Dengan perubahan *binary* pada gambar maka hasil perhitungan *checksum*-nya akan berbeda hasilnya dengan nilai *checksum* yang tersimpan pada *stego message*.

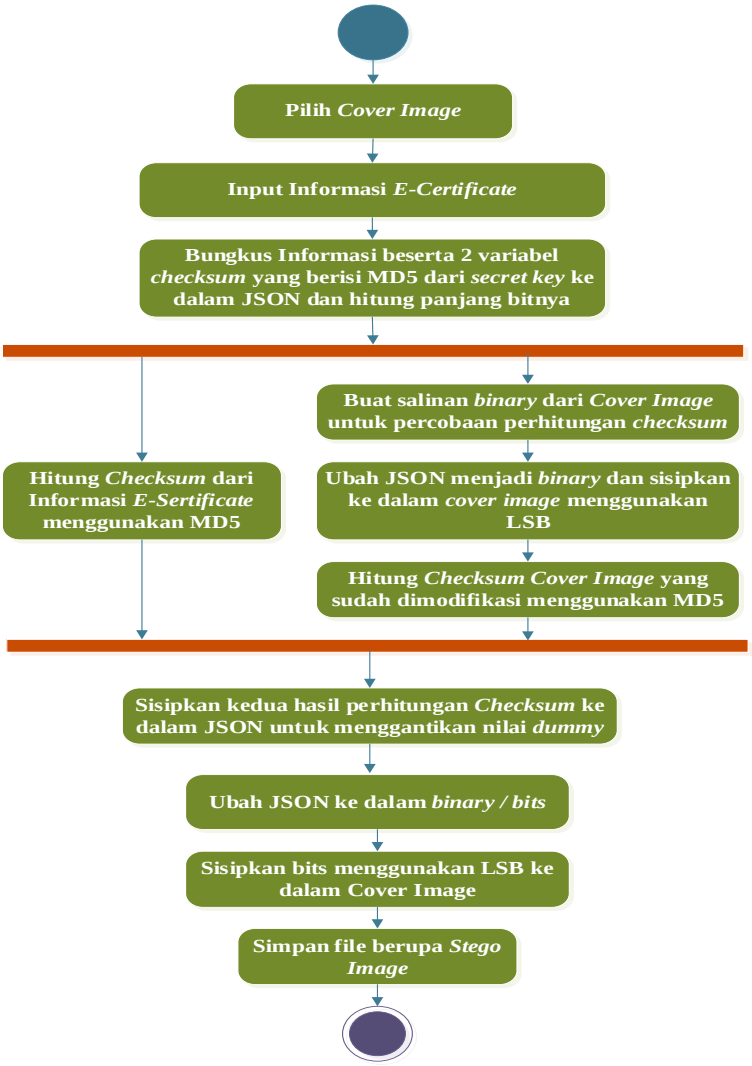
```
{
  "md5_data" : "(32 bytes string)",
  "md5_image" : "(32 bytes string)",
  "no" : "(text)",
  "publish" : "(dd/mm/yyyy)",
  "name" : "(text)",
  "email" : "(text)",
  "more_info" : "(text)"
}
```

Gambar 3. Struktur data JSON

3. HASIL DAN PEMBAHASAN

3.1. Proses *Encode*

Metode untuk pengamanan data pada penelitian ini yaitu menggunakan teknik steganografi yang kemudian dilakukan proses pendeteksian dan pengoreksian *error* data dengan menggunakan *checksum. Checksum* digunakan untuk melindungi keabsahan/keaslian data yaitu dengan cara membandingkan antara hasil perhitungan *checksum* data asli dengan hasil perhitungan *checksum* pada *stego image*. Pada proses *encode* maupun *decode* akan digunakan sebuah nilai yang diberi nama *secret key* pada saat perhitungan dan membandingkan nilai *checksum*. Gambaran alur dari proses pengamanan dan pengecekan data di dalam sistem pada penelitian ini ditunjukkan pada Gambar 4 dan Gambar 5.



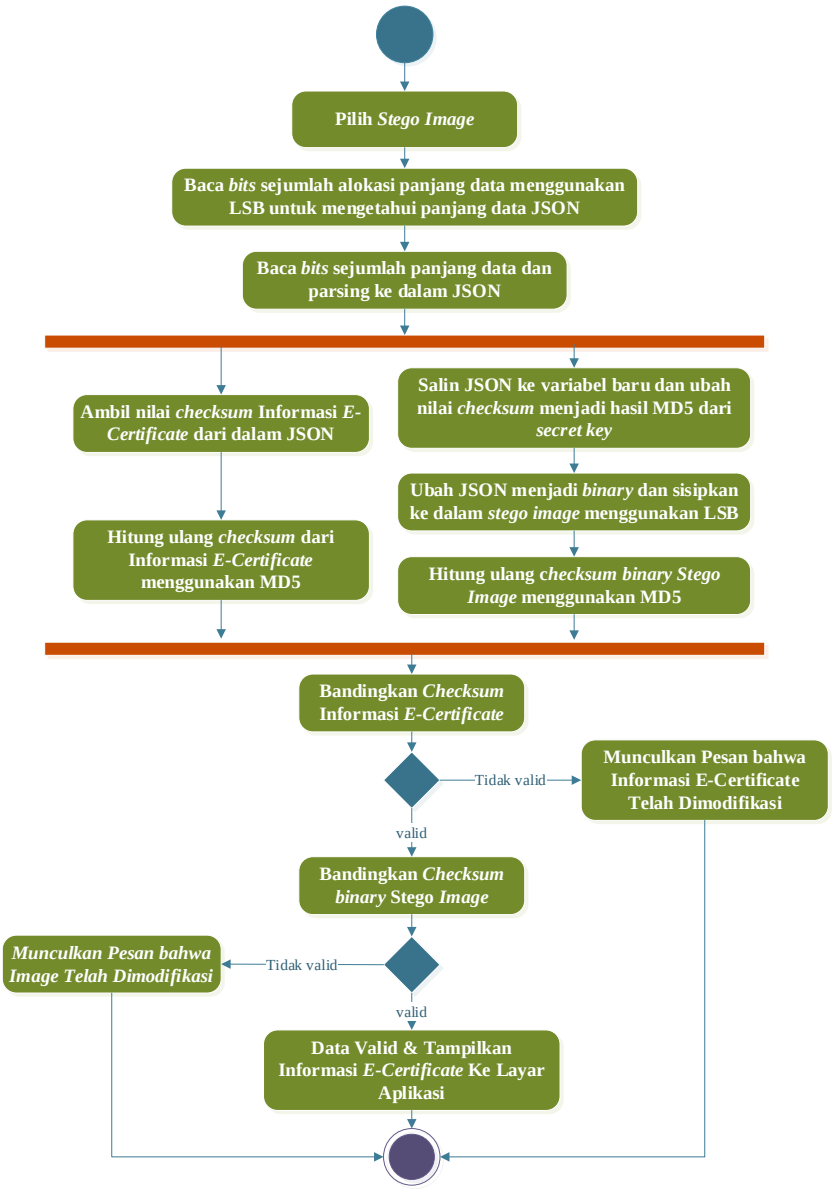
Gambar 4. Proses *Encode*

Proses *encode* ditunjukkan pada Gambar 4. Proses *encode* diawali dengan memilih *cover image*/*e-certificate* yang akan dilakukan pengecekan. Kemudian pengguna sistem harus mengisi informasi tambahan terkait *E-Sertficate* seperti halnya nomor sertifikat, nama pemilik, tanggal sertifikat dan lain sebagainya yang berkaitan dengan data di dalam sertifikat tersebut. Informasi inilah yang nantinya akan disisipkan ke dalam *cover image*. Namun untuk nantinya kita dapat melakukan pengecekan keaslian dari informasi tersebut, maka harus dilakukan perhitungan *checksum* terlebih dahulu menggunakan MD5 terhadap informasi yang akan disimpan, dan terhadap *binary image* yang sudah berisi informasi. Proses *encode* ini menghasilkan stego *image* yang kemudian hasil *file stego image*-nya dapat diperiksa datanya pada aplikasi melalui proses *decode* terlebih dahulu.

 DOI: 10.30998/faktorexacta.v16i3.17314

3.2. Proses *Decode*

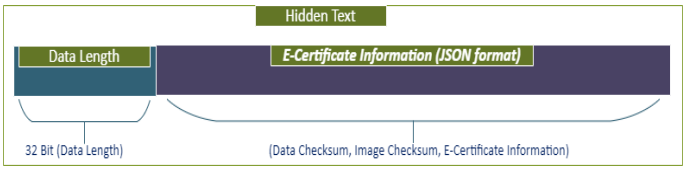
Proses *decode* ditunjukkan pada Gambar 5. Diawali dengan memilih stego *image* yang merupakan hasil dari proses *encode* sebelumnya. Kemudian hasil decode ini akan ditampilkan di layar aplikasi.



Gambar 5. Proses *Decode*

Pada proses *decode* ini akan membandingkan antara hasil *checksum* stego *image* yang sudah dimodifikasi dan hasil *checksum* informasi dari *e-certificate*. Pengecekan pertama dengan membandingkan hasil *checksum* informasi dari *e-certificate*, jika tidak valid maka akan muncul pesan bahwa *e-certificate* tersebut telah dimodifikasi, jika hasil pembandingan valid maka akan dilanjutkan dengan membandingkan hasil *checksum* stego *image*. Jika hasilnya tidak sama maka akan muncul pesan *image* telah dimodifikasi, sebaliknya jika hasilnya sama maka data valid dan akan menampilkan informasi *e-certificate* ke *user*.

Pada proses *encode* maupun *decode*, bit yang berisi informasi data *e-certificate* dimasukan ke dalam struktur data khusus. Adapun struktur data yang dirancang dalam penelitian ini ditunjukkan pada Gambar 6.

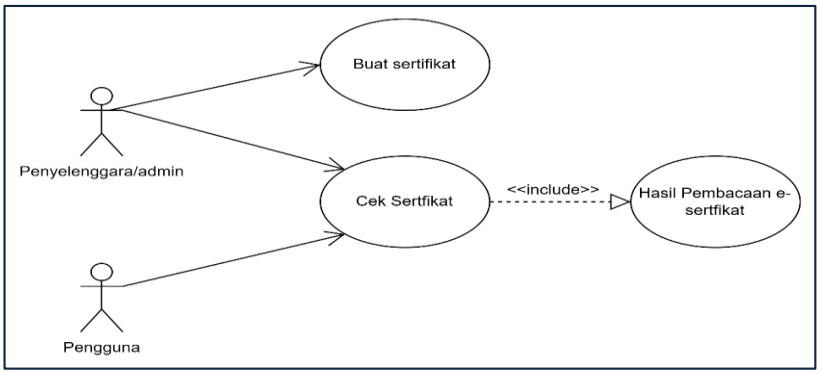


Gambar 6. Struktur Data

Pada *stego message/hidden message*, 32 bit *binary* pertama digunakan untuk menampung panjang informasi/pesan yang akan disisipkan. Kemudian Bit setelah urutan ke-32 akan terisi oleh informasi *E-certificate* sampai dengan urutan bit sesuai panjang *binary* informasi tersebut. Informasi *e-certificate* akan disimpan ke dalam format JSON agar lebih mudah untuk dibaca dan diolah ketika proses *encode maupun decode*. Format JSON yang dirancang terdiri dari 1 buat objek *checksum* yang berisi hasil perhitungan data/informasi *e-certificate*, 1 buah objek *checksum* hasil perhitungan binary *image*, dan beberapa objek yang berisi informasi *e-certificate* itu sendiri.

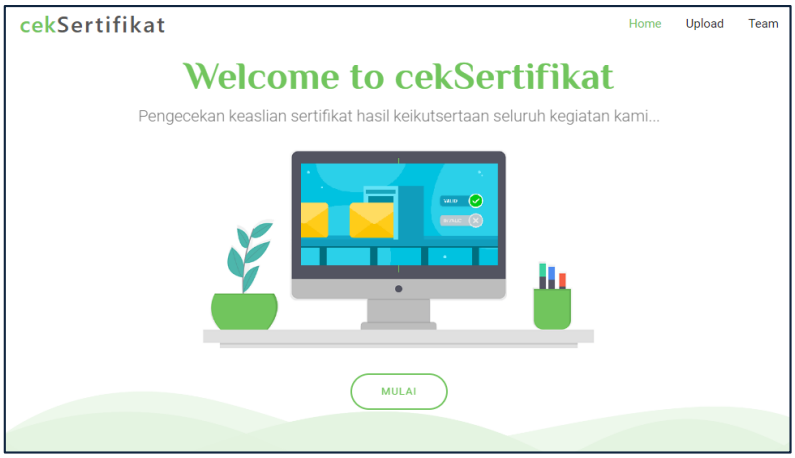
3.3. Implementasi

Tahapan akhir, kami merancang *user interface* dengan memasukkan modul proses *encode* dan *decode* berbasis aplikasi *website* dengan nama cekSertifikat. Gambaran fungsional dari cekSertifikat kami jelaskan melalui *Use Case* Diagram (lihat Gambar 7). Aktor yang terlibat ada 2 yaitu penyelenggara sertifikat sebagai admin dan pengguna yang memiliki kepentingan untuk cek keaslian *e-certificate*. Fungsional sistem cekSertifikat diantaranya adalah buat *e-certificate* dan cek *e-certificate*. Fungsional buat *e-certificate* hanya bisa diakses oleh penyelenggara.



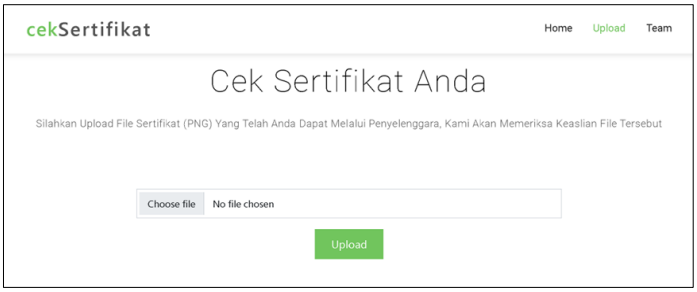
Gambar 7. *Usecase diagram* cek Sertifikat

Implementasi rancangan layar *user interface* kami sajikan ke dalam Gambar 8,9, 10, dan 11. Gambar 8 menampilkan beranda *website* cekSertifikat yang merupakan halaman awal dari aplikasi pengecekan *e-certificate*.



Gambar 8. Tampilan beranda

Gambar 9 menampilkan tampilan unggah dokumen *e-certificate* yang didapat dari penyelenggara. Pemilik dapat unggah dokumen sertifikatnya yang diperoleh dari pihak penyelenggara seminar atau pelatihan sebagai bentuk keikutsertaannya. Pemilik dapat menggunggah dokumen sertifikatnya yang berekstension .png pada tampilan ini.

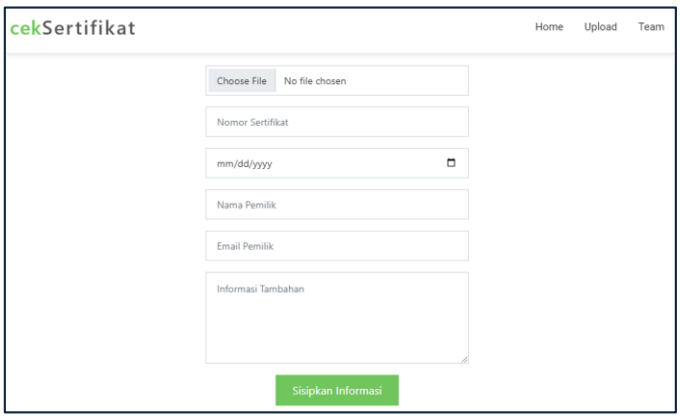


Gambar 9. Tampilan cek *e-certificate*

Setelah dokumen diunggah, selanjutnya sistem akan melakukan pengecekan sertifikat secara langsung dan hasilnya ditampilkan pada laman yang sama di bagian bawah (lihat Gambar 10). Pada saat melakukan *upload* inilah terjadi proses *encode* di dalam aplikasi.

Gambar 10. Hasil pembacaan *e-certificate*

Penyisipan *stego message* hanya bisa dilakukan oleh admin. Halaman khusus Admin untuk menyisipkan *stego message*, kami sajikan ke dalam Gambar 11. *Stego message* yang akan digunakan pada saat pengecekan sertifikat.



Gambar 11. Tampilan *form* buat sertifikat untuk menyisipkan *stego message*

Pada tahap ini admin atau penyelenggara harus sudah memiliki file *e-certificate* dalam format JPG terlebih dahulu. Karena file inilah yang akan diupload ke dalam aplikasi dan nantinya akan disisipkan pesan berupa informasi *e-certificate* tersebut beserta nilai *checksum* untuk pengecekan keasliannya. Selain file asli (*cover image*) dari *e-certificate* tersebut, admin juga harus mengisi beberapa informasi yang sesuai dengan data yang ada di dalam *e-certificate* tersebut. Setelah memilih file dan melengkapi informasi, maka admin hanya perlu menekan tombol ”Sisipkan Informasi” dan server akan membuatkan *stego image* yang secara otomatis akan terdownload pada *browser* yang digunakan oleh admin. Selanjutnya file *stego image* berupa *e-certificate* yang sudah disisipi *stego message* inilah yang nantinya akan didistribusikan ke peserta kegiatan.

4. PENUTUP

Aplikasi pengecekan *e-certificate* telah berhasil kami buat dengan rancangan berbasis *website*. Tidak hanya menawarkan fitur pengecekan keaslian *e-certificate*, kami juga menyediakan halaman untuk menyisipkan *stego message* ke dalam *e-certificate* yang sudah dibuat oleh penyelenggara. Halaman penyisipan *stego message* ini dirancang hanya untuk diakses oleh pihak penyelenggara seminar atau pelatihan sebagai admin sebelum mendistribusikan *e-certificate* ke para peserta kegiatan. Yang perlu digaris bawahi adalah hanya file *e-certificate* yang sudah melewati proses *encode* atau penyisipan informasi menggunakan aplikasi yang kami rancanglah yang dapat dilakukan pengecekan keasliannya. Berdasarkan pengujian langsung terhadap fungsi pengecekan *e-certificate*, aplikasi berhasil menyajikan informasi dari *e-certificate* yang tersimpan di dalam *pixel* gambar *e-certificate* menggunakan metode LSB.

Teknik *encode* dan *decode* menggunakan metode LSB dengan penambahan perhitungan *checksum* menggunakan MD5 telah mampu mendeteksi adanya perubahan pada *binary e-certificate*. Hal ini tentu saja sangat berguna untuk mendeteksi apabila *e-certificate* telah diedit visualisasinya menggunakan *software* pengolah gambar ataupun ketika ada orang lain melakukan modifikasi *binary* data pada *e-certificate* tersebut. Dengan teknik dan metode yang telah digunakan pada aplikasi yang kami buat mampu memenuhi tujuan dalam penelitian yaitu untuk untuk membuat suatu aplikasi yang dapat digunakan untuk memvalidasi suatu sertifikat asli atau tidak, sehingga mampu meningkatkan keamanan, keaslian dan keabsahan data pada *e-certificate*. Namun, penelitian ini dapat dikembangkan lebih lanjut, diantaranya pengembangan modul *encode* dan *decode* dengan memasukan teknik *deep learning* seperti *convolutional neural network* (CNN) serta fungsional cek sertifikat tidak hanya untuk file png tetapi juga semua jenis file gambar dan pdf.

DAFTAR PUSTAKA

[1] M. M. Yahaya and A. Ajibola, “Cryptosystem for Secure Data Transmission using Advance Encryption Standard (AES) and Steganography,” *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, no. May 2021, pp. 317–322, 2019, doi: 10.32628/cseit195659.

[2] A. A. Aripin, “Potensi pemanfaatan teknologi Blockchain terhadap ketepatan waktu, efisiensi dan keamanan proses operasi pada subsektor perbankan,” 2018.

[3] F. Kurniawan, M. L., & Erna Dewi, “Peran Kepolisian Dalam Penyidikan Tindak Pidana Penyalahgunaan Ijazah Palsu.,” *J. Poenale*, 2018.

[4] E. Walia and P. Jain, “An Analysis of LSB & DCT based Steganography,” *Glob. J. Comput. Sci. Technol. GJCST Comput. Classif. F*, vol. 10, no. 1, p. 1, 2010.

[5] K. A. Al-Afandy, O. S. Faragallah, A. Elmhalawy, E. S. M. El-Rabaie, and G. M. El-Banby, “High security data hiding using *image* cropping and LSB least significant bit steganography,” *Colloq. Inf. Sci. Technol. Cist*, vol. 0, pp. 400–404, 2016, doi: 10.1109/CIST.2016.7805079.

[6] N. Bansal, V. K. Deolia, A. Bansal, and P. Pathak, “Digital *image watermarking* using least significant bit technique in different bit positions,” *Proc. - 2014 6th Int. Conf. Comput. Intell. Commun. Networks, CICN 2014*, pp. 813–818, 2014, doi: 10.1109/CICN.2014.174.

[7] P. C. Mandal, I. Mukherjee, G. Paul, and B. N. Chatterji, “Digital *image* steganography: A literature survey,” *Inf. Sci. (Ny).*, vol. 609, pp. 1451–1488, Sep. 2022, doi: 10.1016/j.ins.2022.07.120.

[8] K.-H. Jung and K.-Y. Yoo, “Steganographic method based on interpolation and LSB substitution of digital *image*s,” *Multimed. Tools Appl.*, vol. 74, no. 6, pp. 2143–2155, Mar. 2015, doi: 10.1007/s11042-013-1832-y.

[9] X. Zhou, W. Gong, W. Fu, and L. Jin, “An improved method for LSB based color *image* steganography combined with cryptography,” in *2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, Jun. 2016, pp. 1–4. doi: 10.1109/ICIS.2016.7550955.

[10] A. Muchbarak, N. AZ, and M. Hardjianto, “Prototipe Validasi Dan Proteksi Data Pada File *Image* Dengan Menggunakan Advanced Encryption Standard Dan Least Significant Bit Berbasis Android,” *Jurnal TIKOM*, vol. 59, no. 9–10, 2014, doi: 10.13140/RG.2.1.2483.3449.