IMPLEMENTASI ALGORITMA AES DALAM OPTIMALISASI KEAMANAN CITRA DIGITAL PADA LAYANAN CLOUD

LAPORAN TUGAS AKHIR

Laporan ini Disusun untuk Memenuhi Salah Satu Syarat Memperoleh
Gelar Sarjana Strata 1 (S1) pada Program Studi Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Sultan Agung Semarang

DISUSUN OLEH :

HILMI FATHULLAH
NIM 32602100054

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG
SEMARANG
2025

FINAL PROJECT

IMPLEMENTATION OF AES ALGORITHM IN OPTIMIZING DIGITAL IMAGE SECURITY IN CLOUD SERVICES

This Final Assignment Report was prepared as one of the requirements for obtaining a Bachelor's Degree (S1) in the Informatics Engineering Study Program, Sultan Agung Islamic University, Semarang.

Arranged By :

HILMI FATHULLAH

NIM 32602100054

MAJORING OF INFORMATICS ENGINEERING

INDUSTRIAL TECHNOLOGY FACULTY

SULTAN AGUNG ISLAMIC UNIVERSITY

SEMARANG

2025

LEMBAR PENGESAHAN

TUGAS AKHIR

IMPLEMENTASI ALGORITMA AES DALAM OPTIMALISASI KEAMANAN CITRA DIGITAL PADA LAYANAN CLOUD

HILMI FATHULLAH

NIM 32602100054

Telah dipertahankan di depan tim penguji ujian sarjana tugas akhir

Program Studi Teknik Informatika

Universitas Islam Sultan Agung

Pada tanggal : 24 Februari 2025

TIM PENGUJI UJIAN SARJANA :

Badie'ah, ST, M.Kom — 13 - 03 - 2025

NIDN. 0619018701

(Ketua Penguji)

Ghufron, ST, M.Kom — 12 - 03 - 2025

NIDN. 0602079005

(Anggota Penguji)

Bagus SWP, S.Kom, M.Cs — 18-03-2025

NIDN. 210616051

(Pembimbing)

Semarang, ...

Mengetahui,

Ka. Prodi Teknik Informatika

Universitas Islam Sultan Agung

Moch. Taufik, ST., MIT

NIDN. 0622037502

iii

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Hilmi Fathullah

NIM : 32602100054

Judul Tugas Akhir : Implementasi Algoritma AES Dalam Optimalisasi ``Keamanan Citra Digital Pada layanan Cloud

Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apbila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 13 Maret 2025

Yang Menyatakan,

Hilmi Fathullah

iv

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama : Hilmi Fathullah

NIM : 32602100054

Program Studi : Teknik Informatika

Fakultas : Teknologi industri

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : Implementasi Algoritma AES Dalam Optimalisasi Keamanan Citra Digital Pada Layanan Cloud

Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung.

Semarang, 13 Maret 2025

Yang menyatakan,

Hilmi Fathullah

v

KATA PENGANTAR

Dengan mengucap rasa syukur alhamdulillah atas kehadirat Allah SWT yang telah memberikan rahmat dan karunianya kepada penulis, sehingga dapat menyelesaikan Tugas Akhir dengan judul "Implementasi Algoritma AES Dalam Optimalisasi Keamanan Citra digital Dalam Layanan Cloud" ini untuk memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar sarjana (S-1) pada Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Sultan Agung Semarang.

Tugas Akhir ini disusun dan dibuat dengan adanya bantuan dari berbagai pihak, materi maupun teknis, oleh karena itu saya selaku penulis mengucapkan terima kasih kepada :

1. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto, S.H., M.H yang telah mengizinkan penulis menimba ilmu di kampus ini.
2. Dekan Fakultas TeknologI Industri Ibu Dr. Novi Marlyana, S.T., M.T.
3. Dosen pembimbing penulis Bagus Satrio Waluyo Poetro, S.Kom.,M.Cs yang telah meluangkan waktu, tenaga, dan pikiran untuk penulis.
4. Kepada Almarhum Bapak Cismo dan Ibu Sulastri, yang telah memberikan dukungan, kasih saying, dan do'a yang tiada henti. Dukungan mereka adalah sumber semangat yang tiada terhingga dalam perjalanan akademik dan hidup penulis.
5. Dan kepada semua pihak yang tidak dapat saya sebutkan satu persatu.

Dengan segala kerendahan hati, penulis menyadari bahwa laporan tugas akhir ini masih memiliki banyak kekurangan dalam hal kualitas, kuantitas, isi maupun penyajian. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun untuk membantu laporan ini menjadi lebih baik di masa mendatang.

Semarang, 27 Februari 2025

Hilmi Fathullah

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Kemajuan teknologi digital yang pesat meningkatkan kebutuhan akan perlindungan data, terutama dalam penyimpanan dan pengelolaan citra digital di layanan cloud. Penelitian ini menerapkan Algoritma Rijndael atau Advanced Encryption Standard (AES) dengan mode Cipher Block Chaining (CBC) untuk enkripsi dan dekripsi citra digital serta mengintegrasikannya dengan layanan cloud guna meningkatkan keamanan penyimpanan data. Sistem dikembangkan menggunakan Python dengan framework Flask serta MySQL untuk penyimpanan, sementara integrasi dengan Google Drive API memungkinkan unggah otomatis setelah enkripsi. Evaluasi dilakukan dengan mengukur kinerja algoritma berdasarkan kecepatan enkripsi dan dekripsi. Hasil penelitian menunjukkan bahwa penerapan AES-CBC mampu menjaga keamanan citra digital secara efektif tanpa menyebabkan penurunan kualitas gambar yang signifikan, serta memungkinkan penyimpanan terenkripsi yang lebih aman dan praktis di layanan cloud.

Kata kunci: Kriptografi, AES, keamanan data, Penyimpanan Cloud.

ABSTRACT

The rapid advancement of digital technology has increased the need for data protection, especially in the storage and management of digital images in cloud services. This research implements the Rijndael Algorithm or Advanced Encryption Standard (AES) with Cipher Block Chaining (CBC) mode for encrypting and decrypting digital images and integrates it with cloud services to enhance data security. The system is developed using Python with the Flask framework and MySQL for storage, while integration with the Google Drive API enables automatic uploads after encryption. The evaluation measures the algorithm's performance based on encryption and decryption. The results show that AES-CBC implementation effectively ensures digital image security without significantly degrading image quality while enabling safer and more practical encrypted storage in cloud services.

Keywords: Cryptography, AES, data security, cloud storage

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi informasi dan komunikasi yang pesat dalam beberapa dekade terakhir telah menciptakan dampak besar dalam semua aspek kehidupan, termasuk dalam hal penyebaran dan pengelolaan data. Data digital, terutama citra digital, kini menjadi salah satu jenis informasi yang paling banyak dipertukarkan di dunia maya. Citra digital ini, yang dapat berisi informasi pribadi, medis, atau bahkan dokumen penting, sering kali menjadi target utama dalam ancaman terhadap keamanannya yang dapat menimbulkan kerugian besar.

Salah satu dokumen penting yang sering digunakan dalam berbagai layanan administrasi adalah *Kartu Tanda Penduduk* (KTP). Sebagai dokumen identitas resmi, KTP sering kali digunakan dalam transaksi daring maupun luring yang memerlukan verifikasi identitas. Namun, penyalahgunaan dan pemalsuan KTP dalam bentuk digital menjadi ancaman yang serius, terutama dalam kasus pencurian identitas, manipulasi data, dan penyalahgunaan untuk kejahatan siber. Oleh karena itu, diperlukan metode yang dapat menjaga keaslian serta keamanan citra digital KTP agar tetap terlindungi dalam penyimpanan dan distribusi daring.

Di Indonesia, banyak kasus penyalahgunaan data pribadi, termasuk pemalsuan KTP untuk mendaftarkan akun di aplikasi *pinjol* (pinjaman online) ilegal. Pelaku menggunakan data KTP orang lain yang diperoleh dari kebocoran data atau media sosial, kemudian melakukan edit pada citra KTP agar dapat digunakan untuk meminjam uang tanpa sepengetahuan pemilik asli. Akibatnya, korban sering kali mendapatkan tagihan dari pinjaman yang tidak pernah mereka ajukan.

Banyak oknum yang memanfaatkan teknologi pengeditan gambar untuk memalsukan KTP dalam pembuatan dokumen resmi seperti SIM, paspor, atau BPJS. Salah satu kasus yang sempat viral adalah pemalsuan KTP untuk mendapatkan bantuan sosial (bansos) yang seharusnya ditujukan untuk masyarakat kurang mampu. Dengan enkripsi yang baik, penyalahgunaan seperti ini dapat

dicegah karena dokumen digital akan memiliki perlindungan terhadap modifikasi yang tidak sah.

Salah satu pendekatan yang paling efektif untuk mengatasi permasalahan ini adalah melalui penerapan metode enkripsi. Ada berbagai metode untuk mengatasi keamanan pada pengiriman data. Enkripsi salah satu metode yang handal untuk digunakan untuk mengamanankan data. Enkripsi merupakan suatu pengamanan data dengan melakukan pengkodean (Chandra et al., 2019). Dengan menerapkan enkripsi, data tetap aman dan terlindungi dari akses tidak sah. Terdapat berbagai algoritma enkripsi yang dapat digunakan untuk mengamankan data digital, salah satunya adalah algoritma Rijndael, yang lebih dikenal sebagai *Advanced Encryption Standard* (AES). AES merupakan algoritma enkripsi simetris yang telah terbukti memiliki tingkat keamanan yang tinggi dan efisiensi pemrosesan yang baik, sehingga menjadi pilihan utama dalam berbagai aplikasi pengamanan data.

Namun, penerapan enkripsi pada *citra digital* menimbulkan tantangan baru, yaitu kemungkinan adanya perubahan kualitas citra setelah proses enkripsi dan dekripsi. Perubahan ini dapat berdampak pada keakuratan analisis dan diagnosis medis yang bergantung pada ketajaman detail gambar. Oleh karena itu, penting untuk melakukan pengujian menggunakan metrik *Structural Similarity Index Measure* (*SSIM*) dan *Peak Signal-to-Noise Ratio* (*PSNR*) guna memastikan bahwa informasi yang terkandung dalam *citra digital* tetap dapat digunakan tanpa mengalami degradasi yang signifikan.

Selain aspek keamanan dan kualitas *citra digital*, efisiensi sistem juga menjadi faktor penting dalam penerapan algoritma enkripsi. Proses enkripsi dan dekripsi harus dilakukan dengan kecepatan yang optimal agar tidak menghambat alur kerja tenaga medis. Oleh sebab itu, dalam penelitian ini dilakukan pengujian terhadap kecepatan enkripsi dan dekripsi untuk memastikan bahwa algoritma yang digunakan tetap dapat berjalan dengan efisien dalam lingkungan sistem penyimpanan *cloud*. Selain itu, penggunaan memori juga menjadi faktor yang perlu diperhitungkan agar sistem tetap berjalan dengan stabil tanpa menghabiskan sumber daya komputasi yang berlebihan.

Dalam konteks layanan cloud, penggunaan algoritma Rijndael dapat meningkatkan keamanan citra digital yang disimpan dan dikirim melalui jaringan. Salah satu mode operasi yang dapat digunakan untuk memperkuat enkripsi adalah *Cipher Block Chaining* (CBC). Mode ini meningkatkan keamanan dengan memperkenalkan hubungan antar blok data terenkripsi, sehingga pola dalam *ciphertext* menjadi lebih sulit dikenali oleh pihak yang tidak berwenang.

Dalam penelitian ini, sistem dikembangkan menggunakan bahasa pemrograman *Python* dan framework *Flask* untuk membangun aplikasi berbasis web yang mampu melakukan enkripsi dan dekripsi *citra digital* secara langsung. Selain itu, integrasi dengan *Google Drive API* juga dilakukan agar hasil enkripsi dapat disimpan secara otomatis di *cloud*, memungkinkan pengguna untuk mengelola file terenkripsi dengan lebih mudah. Dengan adanya sistem ini, proses perlindungan data medis dapat dilakukan secara otomatis dan aman tanpa membebani pengguna dengan prosedur yang rumit.

Penelitian ini ditunjukkan untuk mengeksplorasi dan mengimplementasikan algoritma AES dalam pengamanan citra digital pada layanan cloud. Diharapkan dengan penerapan algoritma ini, keamanan citra digital dapat lebih terjamin. Dengan ini, penelitian diharapkan dapat memberikan hasil dalam pengembangan metode pengamanan data yang lebih efektif dan andal dalam lingkungan komputasi awan.

1.2 Perumusan Masalah

Berdasarkan permasalahan, maka dapat diindetifikasikan permasalahan sebagai berikut :

1. Bagaimana cara mengimplementasikan algoritma AES Mode CBC dalam Enkripsi dan Dekripsi Citra Digital?
2. Seberapa efektif algoritma AES dan Mode CBC dalam meningkatkan keamanan citra digital pada layanan cloud?

1.3 Pembatasan Masalah

Untuk pembatasan masalah dari penulisan proposal yaitu sebagai berikut:

1. Jenis citra digital yang digunakan hanya berformat .png, .jpg, sebagai objek penelitian. Jenis citra lain seperti video atau citra tiga dimensi tidak akan dibahas.
2. Algoritma yang akan digunakan hanya AES mode CBC dalam enkripsi citra digital.

1.4 Tujuan Tugas Akhir

Adapun tujuan tugas akhir dari penulisan proposal ini yaitu sebagai berikut :

1. Mengimplementasikan algoritma AES untuk enkripsi dan dekripsi citra digital.
2. Menganalisis efektivitas algoritma AES dalam meningkatkan keamanan citra digital pada layanan cloud

1.5 Manfaat

Penelitian ini diharapkan dapat memberikan manfaat dalam meningkatkan keamanan citra digital Kartu Tanda Penduduk (KTP) yang disimpan dan dibagikan secara daring. Dengan menerapkan algoritma Advanced Encryption Standard (AES) dalam mode Cipher Block Chaining (CBC), citra digital KTP dapat dilindungi dari akses tidak sah dan risiko pemalsuan. Pengujian yang dilakukan, seperti perhitungan SSIM dan PSNR, memberikan evaluasi objektif terhadap kualitas citra setelah proses enkripsi dan dekripsi, sehingga dapat memastikan bahwa metode yang diterapkan tetap mempertahankan keakuratan data medis. Integrasi dengan layanan cloud juga memberikan kemudahan dalam aksesibilitas data yang terenkripsi tanpa mengorbankan keamanan perkembangan teknologi keamanan data, terutama dalam penerapan enkripsi berbasis AES-CBC untuk dokumen digital yang bersifat sensitif.

1.6 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam pembuatan laporan tugas akhir ini meliputi :

BAB 1 : **PENDAHULUAN**

Pada bab I berisi penjelasan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, serta sistematika penulisan.

BAB II : **TINJAUAN PUSTAKA DAN DASAR TEORI**

Pada bab II berisi tentang penjelasan penelitian penelitian terdahulu yang relevan dengan penelitian ini dan dasar teori mengenai Algoritma Rijndael (AES) sebagai referensi peneliti untuk menulis penelitian ini.

BAB III : **METODE PENELITIAN**

Pada bab ini berisi proses serta alur dari tahapan penelitian mulai dari pembuatan flowchart sistem sampai implementasinya.

BAB IV : **HASIL DAN ANALISIS PENELITIAN**

Pada bab ini menjelaskan hasil dari penelitian yaitu, sistem enkripsi dan dekripsi beserta pengujianya.

BAB V : **KESIMPULAN DAN SARAN**

Pada bab ini berisi kesimpulan dan saran proses penelitian dari awal proses penelitian hingga akhir penelitian.

BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Pada bagian ini, akan dibahas mengenai literatur dan penelitian terdahulu yang relevan dengan Implementasi Algoritma Rijndael Dalam Optimalisasi Keamanan Citra Digital Pada Layanan Cloud. Penelusuran literatur sangat penting untuk memberikan landasan teori yang kuat serta menghindari duplikasi penelitian. Algoritma Rijndael (AES) : Penelitian yang dilakukan oleh Azanuddin,dkk menunjukkan bahwa hasil enkripsi citra digital menggunakan algoritma AES 128 bit memberikan *output chiperimage* yang memiliki tingkat keamanan yang baik. Hasil enkripsi citra digital menggunakan algoritma AES 128 bit berupa *chiperimage* yang memiliki perubahan ukuran size dari citra *plainimage*(Azanuddin dkk., 2022).

Keuntungan AES dalam keamanan citra: Penerapan AES (Rijndael) pada citra digital terbukti memberikan tingkat keamanan yang tinggi tanpa banyak memperngaruhi kualitas citra. Penelitian (Joseph & Krishna,) AES dipilih karena sifat keamanan yang kuat dan implementasi sederhana baik dalam perangkat lunak dan perangkat keras. Penggunaan ukuran kunci yang lebih besar meningkatkan kekuatan kriptografi tetapi mensyaratkan bahwa jumlah yang lebih besar dari putaran berulang dilakukan. Algoritma AES juga digunakan karena karena keuntungan untuk mengamankan dokumen dan terbukti aman berdasarkan NIST Standard.

Kualitas Citra Setelah Enkripsi dan Dekripsi : Salah satu tantangan utama dalam mengamankan citra digital adalah menjaga kualitas citra setelah dilakukan proses enkripsi dan dekripsi. Penelitian oleh (Fajriati Romli dkk., 2023) menunjukkan bahwa setelah pengujian didaptkan nilai PSNR di atas 30, itu menunjukkan bahwa tingkat kesalahan yang dihasilkan sangat rendah, dan citra tersebut tidak terlalu mencurigakan setelah mengalami proses penyisipan pesan

Pengujian Enkripsi menggunakan AES mode CBC : Dari hasil pengujian enkripsi menggunakan algoritma AES mode CBC terhadap sebuah dokumen pdf yang berisi plaintext atau pesan asli menghasilkan *chipertext* atau hasil enkripsi yang isinya menampilkan karakter atau simbol-simbol yang unik sehingga informasi atau pesan akan sulit dibaca dan dipahami (Manullang dkk., 2023).

Keamanan data dalam cloud computing : Berdasarkan penelitian oleh(Dwi Setyo Wiratomo dkk., 2022) dihasilkan kombinasi dari algoritma AES dan LZW dapat digunakan dalam pengamanan file dokumen. Dalam penelitian ini algoritma LZW memberikan pengaruh terhadap ukuran file setelah melakukan proses pengamanan file menggunakan algoritma AES yang berakibat ukuran file setelah dilakukan kompresi mengalami kenaikan dari pada ukuran file asli yang disebabkan oleh enkripsi file yang mengakibatkan isi file tersebut memiliki nilai acak.

Pengamanan file menggunakan AES: Penelitian oleh (Bibiola dkk., 2023) menghasilkan peningkatan keamanan file dari pihak yang tidak berwenang, dalam pengujianya dapat dinyatakan bahwa fungsional dari aplikasi pengamanan file memakai algoritma AES berbasis web telah berjalan dengan baik dan aplikasi dapat mengamankan file dokumen asli melalui algoritma AES.

Pengamanan Database sistem menggunakan AES: Penelitian oleh (Tarisa Auliya Ramadhani dkk., 2024) menghasilkan penerapan algoritma AES 128 pada aplikasi registrasi pasien dilakukan perumusan kesimpulan bahwa bahasa pemrograman php dan paket Open SSL digunakan untuk berhasil mengimplementasikan algoritma AES-128. proses enkripsi dekripsi berjalan sesuai dengan yang diharapkan, dengan data yang terenkripsi dapat diubah kembali menjadi bentuk aslinya tanpa kehilangan informasi. Hasil dekripsi yang sesuai dengan data asli mengaskan bahwa implementasi algoritma ini telah dilakukan dengan benar dan efektif.

Pengamanan data penjualan sepatu menggunakan AES: Dalam penelitian (Putra Ramadani Tarigan dkk., 2023) tersebut dapat ditarik kesimpulann dalam menganalisa data penjuala sepatu, langkah yang dilakukan adalah memperoleh data

penjualan dari DU. Enos Ginting melalui observasi dan wawancara. Kemudian data dianalisa sesuai dengan perhitungan algoritma AES.

Penelitian oleh (Poetro dkk., 2010) menunjukkan bahwa kriptografi citra digital menggunakan Rijndael dan transformasi wavelet diskrit Haar dapat menghasilkan suatu citra digital yang tidak jelas objeknya pada saat proses enkripsi dan dekripsi, semakin besar level transformasi citra semakin cepat juga proses kriptografi yang dilakuka karena ukuran citra semakin kecil.

Penelitian oleh (Sidiq dkk., 2023) menunjukkan Untuk meningkatkan keamanan email melalui enkripsi AES dan penyisipan pesan dalam gambar menggunakan metode LSB di website. Untuk melindungi pesan, mengintegrasikan teknik ini dalam antarmuka website, menguji keamanannya, dan berkontribusi pada keamanan informasi secara keseluruhan. Dengan demikian, penelitian ini berfokus untuk mengamankan komunikasi email dengan menggabungkan kriptografi dan steganografi dalam konteks online.

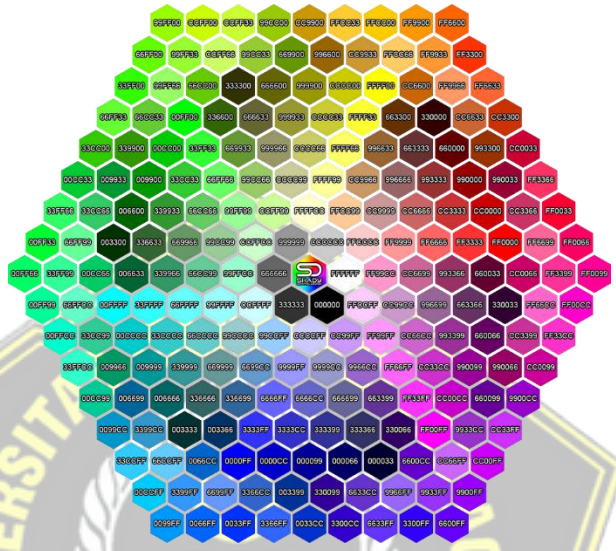
2.2 Dasar Teori

2.2.1 Citra Digital

Citra adalah gambaran atau tiruan dari suatu objek yang dapat berupa gambar nyata maupun digital. Secara umum, citra terbagi menjadi dua jenis utama, yaitu citra tampak seperti foto dan lukisan, serta citra tak tampak seperti citra digital. Dari kedua jenis tersebut, hanya citra digital yang dapat diolah menggunakan komputer. Setiap citra memiliki karakteristik tertentu, seperti ukuran, resolusi, dan formatnya. Biasanya, citra berbentuk persegi dengan dimensi lebar dan tinggi yang spesifik.

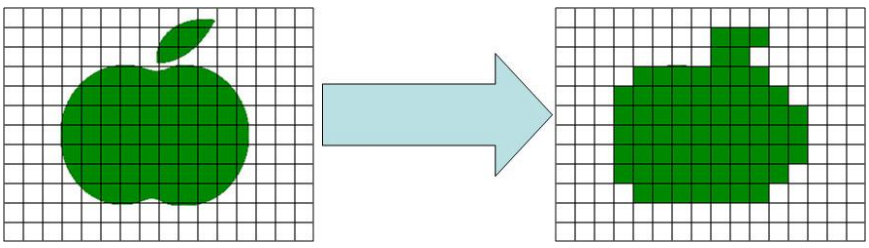
Teknologi yang digunakan dalam pembentukan serta tampilan warna pada citra digital didasarkan pada penelitian yang menunjukkan bahwa warna dapat diciptakan melalui kombinasi tiga warna dasar, yaitu merah, hijau, dan biru (Red, Green, Blue – RGB). Model RGB adalah sistem yang menggabungkan ketiga warna ini untuk menghasilkan berbagai spektrum warna. Setiap warna dasar, seperti merah, memiliki nilai dalam rentang tertentu. Pada monitor komputer, nilai warna berkisar antara 0 hingga 255, yang sesuai dengan sistem representasi 8-bit bilangan biner yang digunakan dalam komputer (Henry dkk., 2016).

Dengan sistem ini, kombinasi warna yang dapat dihasilkan mencapai 16.777.216 variasi (256 x 256 x 256). Dalam komputer, nilai setiap komponen warna biasanya disimpan dalam bentuk bilangan integer dengan rentang 0 hingga 255, yang sesuai dengan kapasitas penyimpanan 1 byte (8-bit). Nilai tersebut dapat direpresentasikan dalam format desimal maupun heksadesimal.



Gambar 2. 1 Hex Triplet Color Chart

Citra digital adalah representasi visual dari suatu objek dalam format numerik yang dapat diolah oleh perangkat komputer. Citra ini terdiri dari elemen-elemen kecil yang disebut piksel, di mana setiap piksel memiliki nilai numerik yang merepresentasikan warna atau tingkat keabuan. Berdasarkan jumlah bit yang digunakan untuk menyimpan informasi warna, citra digital dapat dikategorikan menjadi citra biner, citra keabuan (*grayscale*), dan citra berwarna (*RGB* atau *CMYK*).



Gambar 2. 2 Ilustrasi struktur piksel dalam citra digital

Dalam berbagai aplikasi, citra digital digunakan untuk keperluan identifikasi, analisis, serta komunikasi visual. Penerapan citra digital dapat ditemukan di berbagai bidang, seperti pengolahan gambar medis, sistem

pengenalan wajah, keamanan data digital, dan autentikasi dokumen. Oleh karena itu, pengelolaan citra digital yang baik diperlukan untuk memastikan keandalan dan keamanannya saat digunakan dalam berbagai sistem.

Gambar 2. 3 Ilustrasi perbedaan antara citra biner, keabuan, dan berwarna

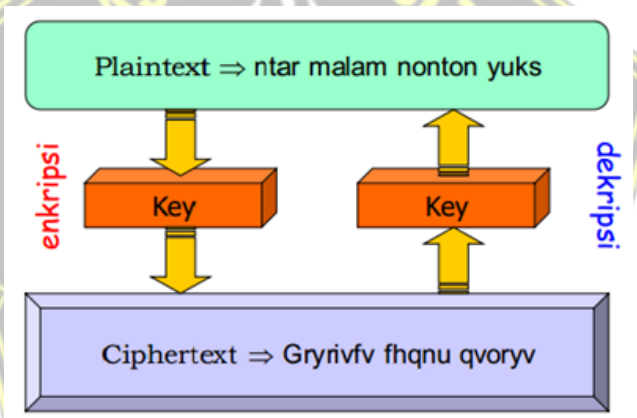
Penyimpanan dan pengiriman citra digital sering kali menghadapi tantangan terkait ukuran data dan keamanan. Format citra digital seperti *JPEG*, *PNG*, dan *BMP* memiliki kelebihan dan kekurangannya masing-masing dalam aspek kompresi dan kualitas. Selain itu, citra digital yang dikirim melalui jaringan dapat berisiko mengalami manipulasi, pencurian, atau akses yang tidak sah, sehingga diperlukan metode perlindungan yang tepat untuk menjaga integritas dan kerahasiaannya.

Salah satu metode perlindungan yang dapat diterapkan dalam pengolahan citra digital adalah teknik enkripsi. Dengan enkripsi, citra digital dapat dikonversi menjadi bentuk yang tidak dapat dipahami oleh pihak yang tidak memiliki izin akses. Hal ini penting terutama dalam aplikasi yang membutuhkan tingkat keamanan tinggi, seperti sistem verifikasi identitas dan penyimpanan data di layanan *cloud*.

Dalam penelitian ini, citra digital akan dienkripsi sebelum disimpan atau dikirim untuk memastikan keamanan dan keasliannya. Proses enkripsi ini harus dilakukan tanpa menyebabkan degradasi kualitas citra yang signifikan, sehingga citra tetap dapat digunakan dengan baik setelah didekripsi. Oleh karena itu, evaluasi kualitas citra setelah enkripsi dan dekripsi juga menjadi aspek penting dalam penelitian ini.

2.2.2 *Kriptografi*

Menurut (Cristy & Riandari, 2021) Kriptografi berasal dari bahasa Yunani, *crypto* dan *graphia*. *Crypto* berarti secret (rahasia) dan *graphia* berarti writing (tulisan). Menurut terminologinya, Kriptografi adalah ilmu dan seni untuk menjaga keamanan pesan ketika pesan atau data dikirim dari suatu tempat ke tempat yang lain. Selain itu menurut (Studi dkk., 2022) yaitu cabang ilmu yang mempelajari teknik komputasi yang berkaitan dengan masalah keamanan informasi seperti kerahasiaan, *integritas* data, dan *otentikasi*. Berdasarkan beberapa definisi tersebut, maka dapat disimpulkan bahwa kriptografi adalah cabang ilmu yang mempelajari beberapa teknik komputasi yang berkaitan dengan aspek keamanan informasi guna menjaga kerahasiaan pesan. Adapun contoh ilustrasi dari kriptografi pada gambar 1 dibawah ini.



Gambar 2. 4 Ilustrasi Kriptografi

Algoritma kriptografi terbagi menjadi dua jenis utama, yaitu *kriptografi simetris* dan *kriptografi asimetris*. Dalam *kriptografi simetris*, enkripsi dan dekripsi menggunakan kunci yang sama, seperti pada algoritma *Advanced Encryption Standard (AES)*. Sementara itu, *kriptografi asimetris* menggunakan pasangan kunci publik dan kunci privat, seperti pada algoritma *Rivest-Shamir-Adleman (RSA)*. AES sering digunakan dalam penyimpanan data sensitif, termasuk citra digital, karena memiliki efisiensi yang tinggi dalam proses enkripsi dan dekripsi.

2.2.3 Algoritma Advanced Encryption Standard (AES)

Algoritma Advanced Encryption Standard(AES) adalah suatu algoritma *block chipper* dan mempunyai sifat simetri yang menggunakan kunci simetri pada waktu proses enkripsi dan dekripsi. Pada tahun 2001, AES digunakan sebagai standar algoritma kriptografi terbaru yang dipublikasikan oleh NIST (National Institute of Standard and Technology) sebagai pengganti algoritma DES (Data Encryption Standard) yang sudah berakhir masa penggunaannya (Prameshwari & Sastra, 2018).

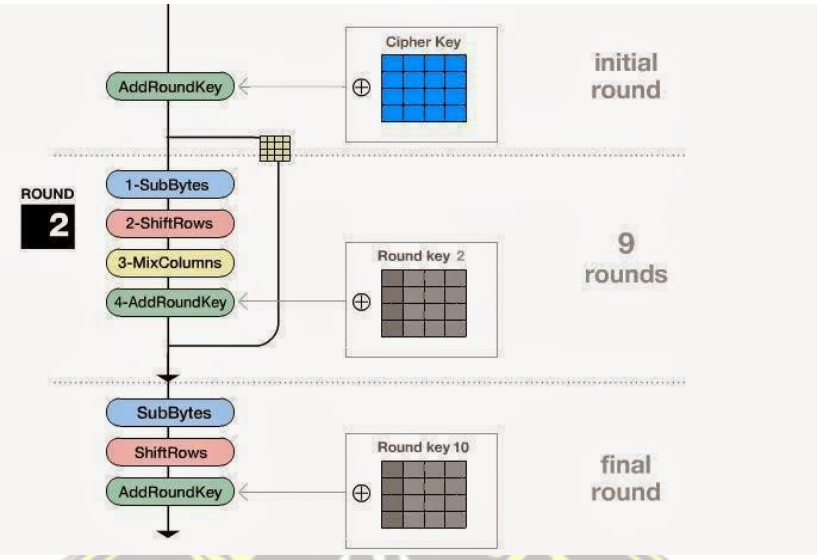
Algoritma AES mendukung berbagai ukuran kunci yang tetap, yaitu 128, 192, dan 256 bit. Ukuran blok dalam AES adalah 128 bit (16 byte). Jumlah iterasi dalam proses enkripsi dan dekripsi bergantung pada panjang kunci yang digunakan.

Tabel 2. 1 Tiga Versi AES

Versi AES	Panjang Kunci (NK words)	Ukuran Blok (NB words)	Jumlah Putaran (Nr)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

Dalam proses enkripsi dan dekripsi, AES menggunakan teknik substitusi, permutasi, serta sejumlah iterasi yang diterapkan pada tiap blok. Setiap iterasi menggunakan kunci yang berbeda, yang dikenal sebagai round key. Tidak seperti

DES yang berbasis bit, AES beroperasi dalam orientasi byte sehingga lebih efisien untuk implementasi dalam perangkat lunak dan perangkat keras.



Gambar 2. 5 Tahapan SubBytes, ShiftRows, MixColumns, dan AddRoundKey.

Dalam implementasi AES, terdapat bebrapa mode operasi salah satunya adalah Chiper Block Chaining (CBC) adalah mode operasi enkripsi yang menggunakan umpan balik antar blok. Setiap blok plaintext akan di-XOR dengan hasil ciphertext dari blok sebelumnya sebelum dienkripsi. Akibatnya, setiap blok ciphertext dalam mode CBC dipengaruhi oleh semua blok sebelumnya, meningkatkan keamanan dengan membuat pola lebih sulit dikenali. Proses enkripsi diawali dengan penggunaan Initialization Vector (IV), yang berfungsi untuk memastikan setiap plaintext memiliki hasil enkripsi yang unik.

Secara matematis, enkripsi dalam mode CBC dirumuskan sebagai:

$$C_i = E_k\,(P_i \oplus C_{i-1}) \qquad 2.1$$

Sedangkan rumus untuk dekripsi adalah:

$$P_i = D_k\,(C_i \oplus C_{i-1}) \qquad 2.2$$

Blok pertama dienkripsi dengan IV sebagai nilai awal, yang dapat diberikan oleh pengguna atau dihasilkan secara acak oleh sistem. Dalam proses dekripsi,

plaintext pertama diperoleh dengan meng-XOR-kan hasil dekripsi blok ciphertext pertama dengan IV.

Rumus matematis untuk dekripsi dalam mode CBC adalah kebalikan dari enkripsi:

Keterangan :

C_i = Blok *chipertext* ke -*i*

E_k= Enkripsi

P_i = Blok *plaintext* ke -*i*

C_0 = Blok plaintext pertama

IV = Initialization Vector

D_k = Dekripsi

$\oplus$ = Operasi *XOR*.

2.2.4 Keamanan Digital

Keamanan digital merupakan aspek penting dalam perlindungan data dan informasi dalam sistem komputer maupun jaringan. Seiring dengan meningkatnya penggunaan teknologi digital, ancaman seperti pencurian data, peretasan, dan manipulasi informasi semakin berkembang. Keamanan digital mencakup berbagai metode dan teknik untuk melindungi data dari akses tidak sah, baik melalui enkripsi, autentikasi, maupun penerapan sistem keamanan berbasis perangkat keras dan perangkat lunak. Salah satu ancaman terbesar dalam dunia digital adalah serangan siber yang dapat mengakibatkan kebocoran informasi pribadi, termasuk dokumen-dokumen penting seperti identitas digital, sertifikat, atau dokumen resmi lainnya.

Pengamanan file dalam lingkungan digital melibatkan perlindungan informasi sensitif dan rahasia dari akses yang tidak sah. Langkah-langkah yang dapat diambil termasuk enkripsi data, penggunaan otentikasi yang kuat, pengawasan akses, dan kebijakan keamanan yang ketat. Penting untuk melakukan penelitian tentang metode dan teknologi pengamanan file yang efektif, serta kebijakan dan praktik terbaik dalam pengelolaan dan perlindungan file digital. Dalam konteks pengamanan file, analisis ancaman yang mungkin timbul, seperti serangan ransomware atau pencurian data, dapat memberikan wawasan tentang langkah-langkah untuk melindungi file-file tersebut(Soesanto dkk., 2023).

Salah satu teknik utama dalam keamanan digital adalah penggunaan algoritma enkripsi untuk memastikan bahwa data hanya dapat diakses oleh pihak yang berwenang. Enkripsi bekerja dengan mengubah data asli menjadi format yang tidak dapat dibaca tanpa kunci dekripsi yang valid. Algoritma seperti *Advanced Encryption Standard* (AES) dan *Rivest-Shamir-Adleman* (RSA) banyak digunakan untuk melindungi informasi dalam komunikasi, penyimpanan data, maupun transaksi digital. Selain enkripsi, metode lain seperti autentikasi dua faktor (2FA) dan sistem keamanan berbasis *blockchain* juga semakin populer dalam meningkatkan perlindungan data.

Selain itu, keamanan digital juga mencakup strategi perlindungan terhadap integritas data agar tidak mengalami modifikasi yang tidak sah. Misalnya, dalam penyimpanan dokumen digital, integritas data dapat dijaga dengan teknik hashing seperti *SHA-256*, yang memastikan bahwa suatu data belum diubah sejak pertama kali disimpan. Dengan menerapkan metode keamanan digital yang efektif, risiko penyalahgunaan dan pencurian data dapat diminimalisir, sehingga sistem informasi dapat beroperasi dengan lebih aman dan terpercaya.

2.2.5 Python

Gambar 2. 6 Python

Python adalah bahasa pemrograman yang menggunakan interpreter untuk menjalankan kode programnya. Interpreter tersebut dapat menerjemahkan kode secara langsung, dan Python dapat dijalankan di berbagai platform seperti Windows, Linux, dan lain-lain. Pythonmengadopsi paradigma pemrograman dari beberapa bahasa lain, termasuk paradigma pemrograman prosedural seperti bahasa C, pemrograman berorientasi objek seperti Java, dan bahasa fungsional seperti Lisp. Kombinasi paradigma ini memudahkan para

programmer dalam mengembangkan berbagai proyek menggunakan Python (Rahman dkk., 2023). *Python* adalah bahasa pemrograman tingkat tinggi yang banyak digunakan dalam berbagai bidang, termasuk pengolahan data, keamanan siber, dan kecerdasan buatan. Bahasa ini memiliki sintaks yang sederhana namun powerful, sehingga memudahkan pengembang dalam membangun aplikasi berbasis keamanan data.

Salah satu keunggulan *Python* adalah dukungannya terhadap algoritma kriptografi, termasuk *Advanced Encryption Standard (AES)*. Dengan adanya pustaka seperti *PyCryptodome* dan *cryptography*, proses enkripsi dan dekripsi dapat dilakukan dengan aman dan efisien. Implementasi *AES-CBC* dalam *Python* memastikan bahwa data yang dienkripsi memiliki tingkat keamanan yang tinggi, terutama dalam melindungi citra digital di layanan cloud.

Selain itu, *Python* juga menyediakan dukungan untuk integrasi dengan berbagai layanan cloud, seperti *Google Drive API*. Dengan bantuan pustaka seperti *requests* dan *google-auth*, proses autentikasi dan pengelolaan file di layanan cloud dapat dilakukan dengan lebih mudah. Hal ini menjadi penting dalam konteks keamanan citra digital, di mana data harus dienkripsi sebelum diunggah dan hanya dapat diakses oleh pihak yang berwenang.

2.2.5 Flask

Flask adalah framework berbasis *Python* yang ringan dan fleksibel untuk membangun aplikasi web. Dibandingkan dengan framework lain seperti *Django*, *Flask* lebih sederhana dan modular, sehingga cocok untuk proyek yang membutuhkan performa tinggi dengan ukuran kode yang minimal. Dalam implementasi sistem enkripsi citra digital, *Flask* digunakan untuk membangun backend yang mengelola proses enkripsi, dekripsi, serta integrasi dengan penyimpanan cloud.

Struktur dasar *Flask* terdiri dari *routing*, *request handling*, dan penggunaan template dengan *Jinja2*. Setiap permintaan yang dikirim oleh pengguna akan diproses oleh *Flask*, baik untuk mengunggah gambar, melakukan enkripsi, maupun menampilkan hasil dekripsi. Dengan dukungan pustaka tambahan seperti *Flask-Login* dan *Flask-WTF*, keamanan aplikasi dapat ditingkatkan melalui autentikasi

pengguna dan perlindungan terhadap serangan seperti *Cross-Site Request Forgery (CSRF)*.

Selain itu, *Flask* juga mendukung integrasi dengan layanan pihak ketiga seperti *Google Drive API* menggunakan protokol *OAuth 2.0*. Dalam sistem ini, pengguna dapat mengunggah hasil enkripsi secara otomatis ke akun *Google Drive* mereka setelah proses enkripsi selesai. Hal ini tidak hanya meningkatkan keamanan penyimpanan citra digital, tetapi juga memudahkan akses terhadap data tanpa mengorbankan privasi pengguna.

2.2.6 Google Drive

Gambar 2. 7 Google Drive

Google Drive adalah layanan penyimpanan berbasis *cloud* yang dikembangkan oleh Google untuk memungkinkan pengguna menyimpan, mengelola, dan berbagi berbagai jenis file secara daring. Pada dasarnya layanan google drive sama seperti cloud storage lain seperti Dropbox atau OneDrive(Di dkk.). Google Drive merupakan media penyimpanan daring yang membuat data tersimpan secara komputasi awan sehingga bisa digunakan setiap setiap saat tanpa batasan jarak dan waktu. Dengan menggunakan Google Drive, pengguna dapat mengakses file mereka dari berbagai perangkat yang terhubung dengan akun Google. Keunggulan utama layanan ini adalah kemampuannya untuk menyinkronkan data secara otomatis, sehingga perubahan yang dilakukan pada satu perangkat dapat langsung tercermin di perangkat lain yang memiliki akses ke akun tersebut.

Keamanan dalam Google Drive menjadi aspek penting dalam penggunaannya, terutama untuk menyimpan file yang bersifat pribadi atau sensitif. Google Drive menerapkan berbagai mekanisme perlindungan seperti autentikasi dua faktor (*two-factor authentication*), enkripsi data saat transit dan saat tersimpan (*encryption in transit and at rest*), serta kontrol akses berbasis izin yang

memungkinkan pengguna untuk mengatur siapa saja yang dapat melihat atau mengedit file yang disimpan. Selain itu, Google Drive juga menyediakan fitur pemulihan data untuk mengembalikan file yang dihapus dalam jangka waktu tertentu.

Integrasi Google Drive dengan berbagai aplikasi dan layanan lain membuatnya menjadi pilihan utama dalam penyimpanan digital. Melalui API Google Drive, pengembang dapat mengakses dan mengelola file yang tersimpan dalam akun pengguna secara programatik. API ini memungkinkan pengunggahan otomatis, pengambilan file, serta pengaturan izin akses secara terprogram. Dengan fitur ini, Google Drive sering digunakan sebagai bagian dari sistem yang membutuhkan penyimpanan berbasis *cloud*, seperti dalam sistem enkripsi dan pengarsipan dokumen digital.

2.2.7 XAMPP

Gambar 2. 8 Xampp

XAMPP adalah perangkat lunak yang menyediakan paket lengkap untuk menjalankan server lokal di komputer. Paket ini mencakup Apache sebagai server *web*, MySQL sebagai basis data, serta PHP dan Perl sebagai bahasa pemrograman yang mendukung pengembangan aplikasi berbasis *web*. XAMPP dirancang agar mudah digunakan oleh pengembang, terutama dalam proses pengujian dan pengembangan aplikasi sebelum diterapkan ke server produksi.

Dalam konteks penelitian ini, XAMPP digunakan sebagai lingkungan pengujian basis data MySQL sebelum diintegrasikan dengan sistem utama. Dengan menggunakan XAMPP, pengembang dapat membuat, mengelola, dan menguji basis data secara lokal tanpa perlu koneksi ke server eksternal. XAMPP juga

menyediakan *phpMyAdmin*, sebuah antarmuka berbasis *web* yang mempermudah pengelolaan database MySQL, termasuk dalam membuat tabel, memasukkan data, dan menjalankan kueri SQL.

Keunggulan utama XAMPP adalah kemudahannya dalam pengaturan dan kompatibilitasnya dengan berbagai sistem operasi, seperti Windows, macOS, dan Linux. Dengan menggunakan XAMPP, proses pengembangan dan pengujian sistem enkripsi citra digital dalam penelitian ini dapat dilakukan dengan lebih fleksibel dan efisien sebelum diterapkan pada lingkungan *cloud*.

2.2.8 Anaconda

Gambar 2. 9 Anaconda

Anaconda adalah distribusi Python yang dirancang khusus untuk pengolahan data, pembelajaran mesin (*machine learning*), dan pengembangan aplikasi berbasis *data science*. Anaconda menyediakan berbagai pustaka dan alat yang memudahkan pengguna dalam mengelola proyek berbasis data, termasuk pustaka seperti NumPy, Pandas, Matplotlib, dan TensorFlow. Selain itu, Anaconda memiliki *package manager* bernama Conda yang memungkinkan pengguna menginstal dan mengelola pustaka serta lingkungan pengembangan dengan lebih efisien.

Dalam penelitian ini, Anaconda digunakan untuk mengelola lingkungan pengembangan Python yang digunakan dalam proses enkripsi dan dekripsi citra digital. Dengan menggunakan Anaconda, berbagai pustaka yang diperlukan dapat diinstal dalam lingkungan virtual terisolasi, sehingga meminimalkan konflik dependensi antar pustaka. Selain itu, Anaconda juga menyediakan *Jupyter Notebook*, sebuah alat yang mempermudah pengujian dan dokumentasi kode selama proses pengembangan sistem keamanan citra digital.

Keunggulan utama Anaconda adalah kemampuannya untuk menangani pemrosesan data dalam skala besar dengan performa optimal. Dengan fitur-fitur yang dimilikinya, Anaconda mendukung pengolahan citra digital dalam penelitian ini,

baik dalam proses enkripsi menggunakan algoritma AES-CBC maupun dalam analisis kualitas citra setelah dekripsi. Anaconda juga mendukung integrasi dengan berbagai teknologi lain, seperti Flask untuk pengembangan aplikasi berbasis *web* dan MySQL untuk penyimpanan data.

BAB III
METODE PENELITIAN

3.1 Dekripsi Sistem

Untuk mengamankan citra digital dari kebocoran data, diperlukan penggunaan algoritma Rijndael dalam mode CBC. Mode CBC memberikan tingkat keamanan yang lebih tinggi karena mengacak setiap blok ciphertext berdasarkan IV dan blok sebelumnya, menghindari pola yang dapat dikenali. Kebutuhan sistem meliputi pemrosesan citra dalam berbagai format, pembagian citra menjadi blok-blok yang dapat dienkripsi, serta penggunaan kunci dan IV yang aman untuk menjaga kerahasiaan citra. Tahap ini juga mengidentifikasi pentingnya pengujian kinerja dan kualitas citra yang dipulihkan untuk memastikan bahwa sistem enkripsi yang dibangun dapat berfungsi dengan baik.

Proses utama dalam sistem ini mencakup beberapa tahapan, yaitu unggah citra digital, enkripsi menggunakan AES-CBC, penyimpanan hasil enkripsi, serta pengujian kualitas citra setelah dekripsi menggunakan metrik *Structural Similarity Index (SSIM)*. Implementasi sistem dilakukan menggunakan *Flask* sebagai kerangka kerja backend, *MySQL* sebagai basis data untuk menyimpan log aktivitas pengguna, serta integrasi dengan *Google Drive API* untuk mengelola penyimpanan terenkripsi. Dengan pendekatan ini, sistem mampu menyediakan mekanisme perlindungan data medis yang lebih aman dan memastikan integritas citra tetap terjaga setelah proses enkripsi dan dekripsi.

3.2 Studi Literatur

Penelitian ini mengacu pada berbagai studi terdahulu yang membahas implementasi algoritma *Advanced Encryption Standard (AES)* dalam keamanan data digital, khususnya dalam bidang citra digital. Beberapa penelitian menunjukkan bahwa metode enkripsi AES dengan mode operasi *Cipher Block Chaining (CBC)* mampu memberikan perlindungan yang kuat terhadap ancaman keamanan, seperti akses tidak sah dan modifikasi data

Selain itu, studi terkait penggunaan enkripsi dalam penyimpanan citra digital di *cloud* juga menjadi referensi utama. Beberapa penelitian telah mengkaji dampak enkripsi terhadap kualitas citra digital menggunakan metrik evaluasi seperti *Structural Similarity Index (SSIM)*. Hasil penelitian ini menunjukkan bahwa meskipun terdapat sedikit perubahan setelah proses enkripsi dan dekripsi, citra digital tetap dapat digunakan untuk keperluan diagnosis. Dengan mengacu pada studi-studi tersebut, penelitian ini bertujuan untuk mengembangkan sistem enkripsi citra digital berbasis AES-CBC yang dapat meningkatkan keamanan data medis saat disimpan di *cloud*.

3.3 Perancangan Sistem

3.3.1 Analisis Kebutuhan

Penelitian ini menganalisis perangkat lunak yang diperukan agar pengembangan agar sesuai yang diharapkan mulai dari proses input hingga hasil akhir. Berikut daftar perangkat yang digunakan dalam pengembangan sistem:

1. Komputer

   Komputer yang digunakan pada penelitian ini adalah laptop Lenovo Ideapad 5 AMD Ryzen 5 4500U *with Radeon Graphics* 2.38 *GHz*, RAM 8,00 GB dengan operasi sistem 64-bit

2. Python 3.13.2

   Python merupakan bahasa pemrograman tingkat tinggi yang fleksibel yang berfokus pada keterbacaan dan kesederhanaan. Diproduksi oleh Guido van Rossum dan pertama kali tersedia pada tahun 1991, Python menjadi salah satu bahasa pemrograman paling terkenal dan banyak digunakan di seluruh dunia. Penelitian ini menggunakan bahasa pemrograman Python versi 3.13.2 karena memiliki perpustakaan yang lengkap dan bersifat open source.

3. Anaconda

   Anaconda merupakan distribusi open source dari bahasa pemrograman Python yang dimanfaatkan guna mengembangkan ilmu data, pembelajaran mesin, komputasi ilmiah, dan aplikasi terkait lainnya. Dalam penelitian ini,

menggunakan Anaconda untuk membuat lingkungan virtual. Anaconda memungkinkan membuat lingkungan virtual terisolasi untuk berbagai proyek.

4. Visual Studio Code (VSCode)

   Visual Studio Code (VSCode) merupakan editor teks untk kode sumber yang dibuat oleh Microsoft. Dalam penelitian ini, VSCode digunakan untuk membuat dan menjalankan program. Karena VSCode memiliki integrasi Git bawaan yang memungkinkan pengguna mengelola repositori, melakukan, mendorong dan menarik, dan melihat riwayat perubahan kode langsung dari editor. Dalam penelitian ini VSCode digunakan untuk proses deteksi realtime.

5. MySQL

   MySQL adalah sistem manajemen basis data relasional (*Relational Database Management System* atau RDBMS) yang digunakan untuk menyimpan informasi terkait proses enkripsi dan dekripsi, seperti data pengguna, log aktivitas, menyimpan riwayar user login, serta metadata file yang diunggah.

6. Flask

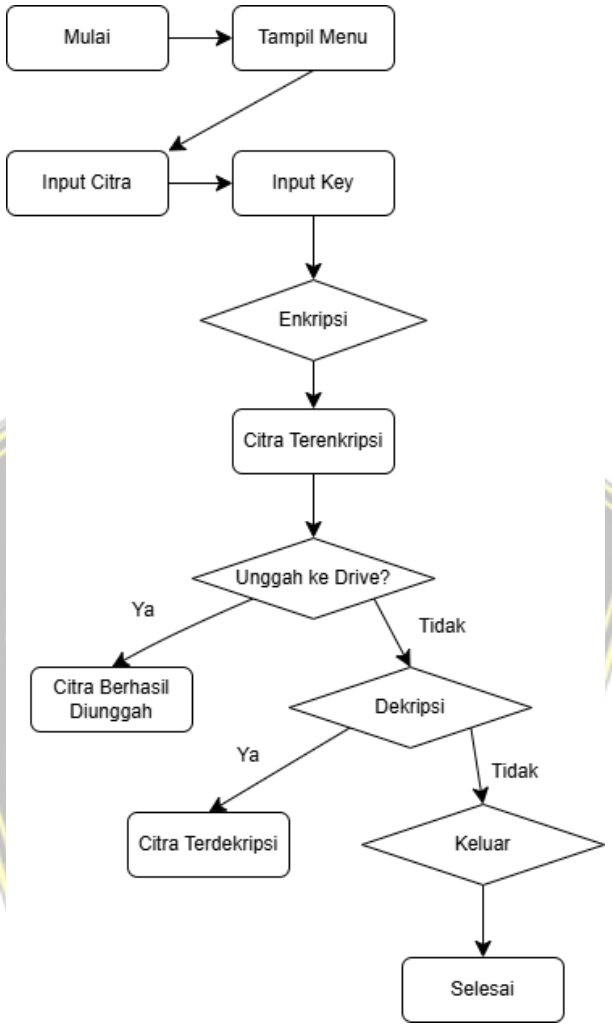
   Flask merupakan kerangka kerja (*framework*) berbasis Python yang ringan dan fleksibel untuk membangun aplikasi web, termasuk dalam proyek ini untuk menangani antarmuka pengguna dan komunikasi dengan backend.

7. PyCrypyodome

   PyCryptodome adalah pustaka Python yang menyediakan algoritma kriptografi, termasuk AES-CBC, yang digunakan dalam sistem untuk mengenkripsi dan mendekripsi citra digital guna meningkatkan keamanan saat disimpan di cloud.

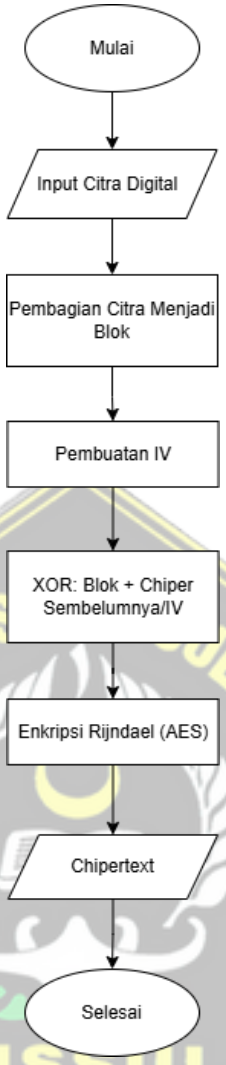
3.3.2 Perancangan Arsitektur Sistem

Pada tahap ini peneliti akan merancang alur kerja dari sistem yang akan dibuat. Untuk mempresentasikan alur kerja sistem peneliti akan membuatnya dalam bentuk *flowchart* yang akan ditampilkan oleh gambar 3.1



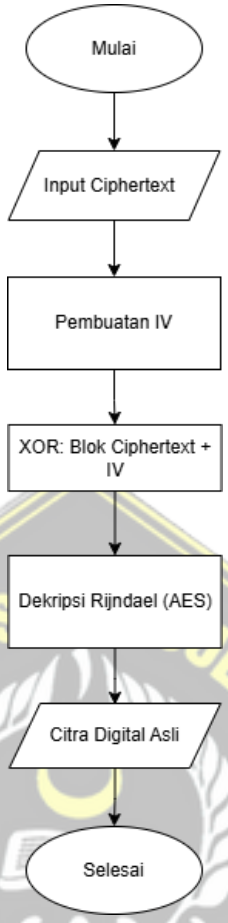
Gambar 3. 1 activity diagram

Gambar 3.1 merupakan flowhart alur untuk enkripsi dan dekripsi menggunakan algoritma rijndael pada layanan cloud. *Activity diagram* berupa *flowchart* yag digunakan untuk memperlihatkan aliran kerja dari aplkasi. Pada gambar 3. memperlihatkan aliran dari interaksi *user* terhadap *system* kriptografi melalui rangkaian *state* awal (*mulai*) hingga rangkaian *state* terakhir (*menutup aplikasi*) dengan aliran Keputusan "*ya*" atau "*tidak*" untuk melakukan "*enkripsi*" dan "*dekripsi*" sehingga menghasilkan "*data terenkripsi*" dan "*data terdekripsi*".



Gambar 3. 2 Alur Enkripsi

Pada gambar 3.2 meruapakan alur kerja dari enkripsi algoritma rijndael (AES) mode CBC dimulai dengan input citra digital, citra yang diinput dibagi menjadi blok-blok kecil, pembuatan IV digunakan untuk meningkatkan keamanan enkripsi, operasi XOR dimana blok citra dienkripsi dengan blok citra dan IV / blok sebelumnya, blok yang telah di XOR kemudian dienkripsi menggunakan algoritma Rijndael (AES), hasil dari enkripsi adalah *chipertext*.



Gambar 3. 3 Alur Dekripsi

Gambar 3.3 menunjukkan flowchart alur dekripsi dimulai dari input *chipertext*, pembuatan IV (Initialization vector) dimana IV yang digunakan enkripsi harus tersedia untuk dekripsi, operasi XOR antara *chipertext* dengan IV, data yang sudah di XOR kemudian didekripsi menggunakan algoritma Rijndael (AES), hasil dari dekripsi adalah citra digital asli.

3.4 Tahapan Perancangan Model

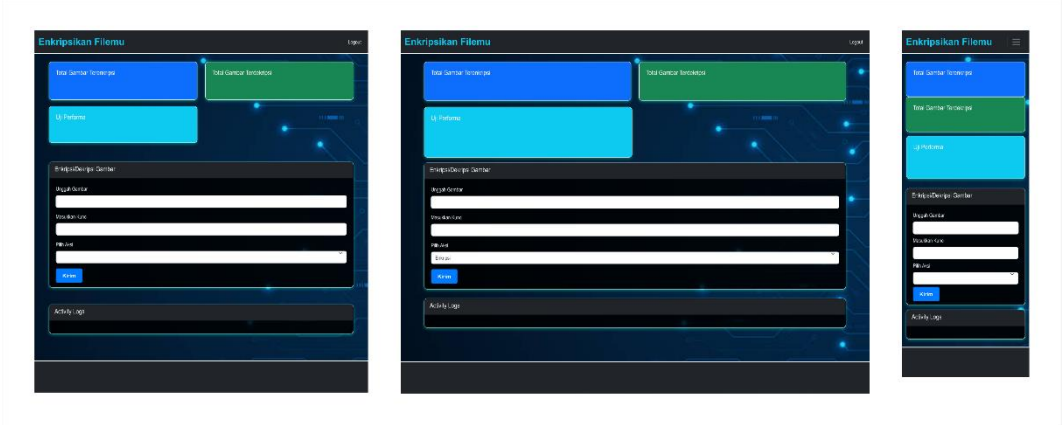
3.4.1 Perancangan AntarMuka (UI/UX)

Perancangan antarmuka pengguna (User Interface atau UI) dan pengalaman pengguna (User Experience atau UX) merupakan bagian penting dalam pengembangan sistem enkripsi citra digital berbasis AES-CBC. Tujuan utama dari perancangan antarmuka adalah untuk memastikan sistem mudah digunakan, memiliki tata letak yang intuitif, serta mendukung fungsionalitas utama seperti unggah gambar, enkripsi, dekripsi, dan penyimpanan hasil di Google Drive.

Sebelum melakukan perancangan antarmuka, dilakukan analisis kebutuhan untuk menentukan elemen-elemen yang harus ada dalam sistem. Beberapa kebutuhan utama yang harus dipenuhi adalah:

1. Kemudahan Penggunaan: Antarmuka harus dirancang agar pengguna, terutama User atau peneliti, dapat dengan mudah mengunggah gambar dan melakukan proses enkripsi atau dekripsi.
2. Responsif dan Fleksibel: Sistem harus mendukung berbagai ukuran layar agar dapat digunakan di berbagai perangkat seperti PC, laptop, dan tablet.

Setelah analisis kebutuhan dilakukan, tahap selanjutnya adalah pembuatan *wireframe* sebagai rancangan awal struktur tampilan sistem. *Wireframe* membantu dalam menentukan tata letak elemen UI yang akan digunakan. Untuk desain yang lebih realistis, dibuat *mockup* menggunakan alat seperti Figma, yang memberikan gambaran tampilan sebelum implementasi kode dilakukan.

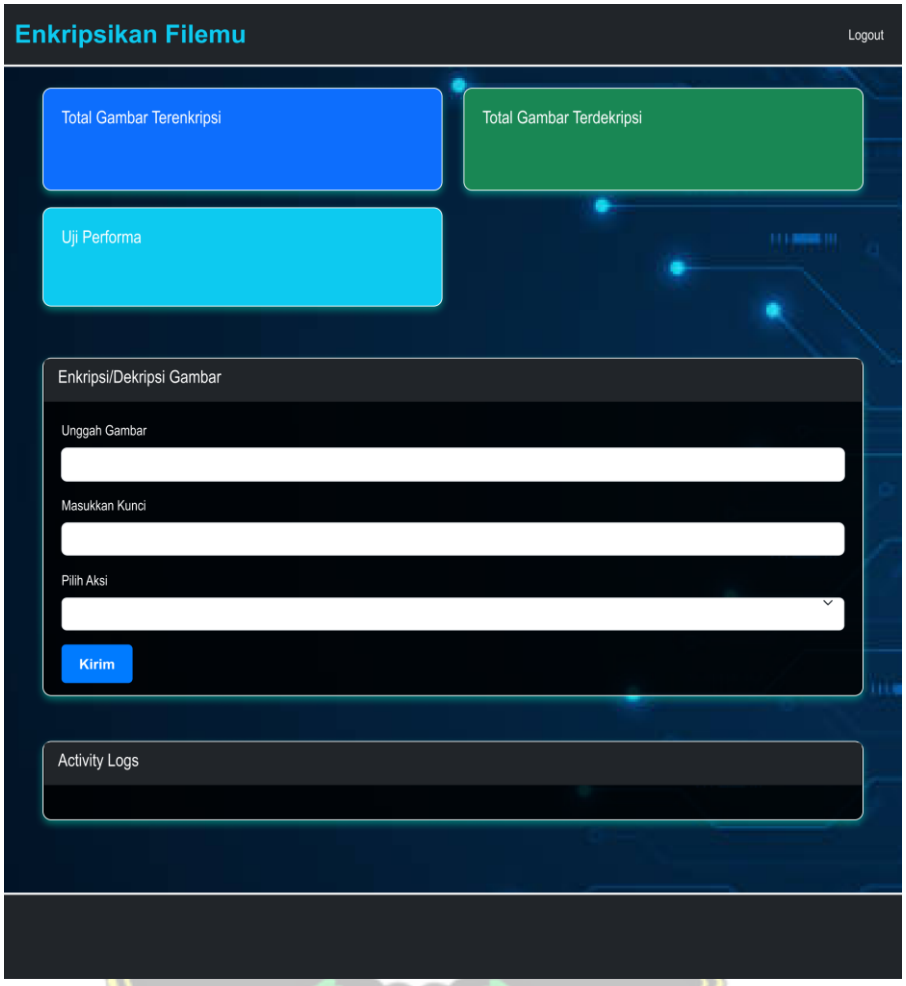


Gambar 3. 4 Tampilan Responsif dari design mockup di Figma

Dari gambar 3.4 diatas menunjukkan tampilan desain figma tampilan responsif unruk masing-masing perangkat mulai dari laptop, tablet, dan hp.

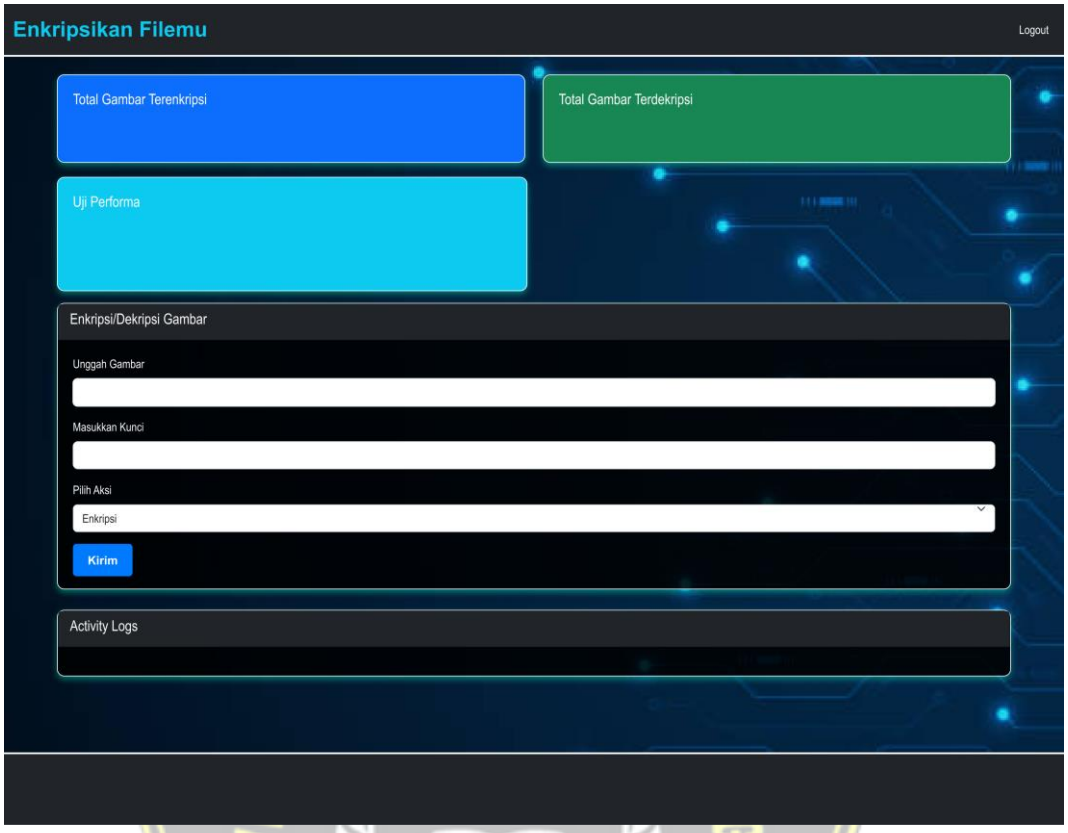
Gambar 3. 5 Tampilan respomsive untuk Smartphone

Dari Gambar 3.5 diatas adalah tampilan antarmuka untuk Smartphone desain antarmuka yang sederhanadengan navigasi yang mudah dijangkau oleh jari pengguna dengan dimensi ukuran 390px.



Gambar 3. 6 Tampilan Responsive untuk tablet

Dari gambar 3. 6 diatas adalah tampilan untuk tablet dengan dimensi 1024px yang memungkinkan tampilan lebih luas dan lebih banyak konten yang bisa ditampilkan tanpa terlalu banyak *scrolling*.



Gambar 3. 7 Tampilan Responsive pada Laptop dan Dekstop

Gambar 3.7 menunjukkan tampilan dari perangjat laptop dan dekstop yang memiliki dimensi 1440px, tata letak dapat dibuat lebih kompleks dengan lebih banyak elemen yang ditampilkan dalam satu layer.

Dengan adanya perancangan antarmuka ini, sistem diharapkan dapat digunakan dengan lebih mudah, aman, dan efisien oleh pengguna dalam proses enkripsi dan dekripsi citra.

3.4.2 Enkripsi Dekripsi Citra dengan AES

Proses enkripsi pada algoritma AES terdiri dari 4 jenis transformasi bytes, yaitu SubBytes, ShiftRows, MixCloumn, dan AddRoundkey. Sama halnya dengan proses enkripsi, proses dekripsi juga terdiri dari 4 jenis transformasi bytes yaitu Inv SubBytes, InvShiftRows, InvMixcolumns, dan AddRoundKey. Proses SubBytes merupakan transformasi dengan setiap elemen pada data masukan (*state*) akan dipetakan dengan menggunakan sebuah tabel substitusi (S-Box). Sedangkan untuk InvSubBytes hanya berbeda tabel substitusinya (Inv S-Box).

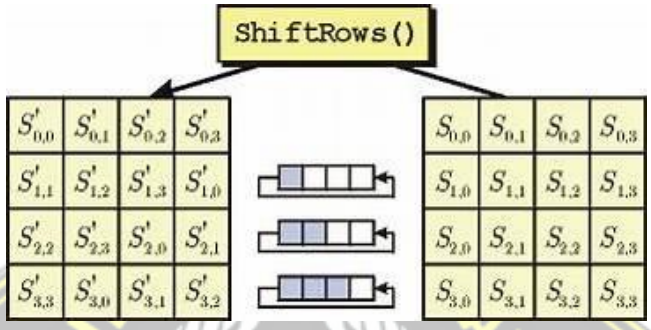
L(xy)	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	12	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d5	b3	29	e3	2f	84
5	53	dl	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	00c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1v	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Gambar 3. 8 Tabel Substitusi S-Box

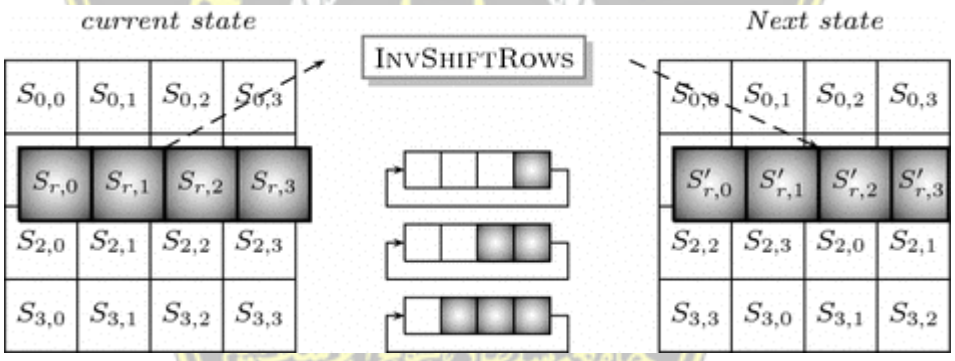
L(xy)	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	79
9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
a	47	f1	1a	71	1f	29	c5	89	6f	b7	62	0e	aa	18	be	1b
b	fc	56	3e	4b	v6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
d	60	51	7f	a9	19	b5	4a	0d	2d	e4	7a	9f	93	c9	9c	ef
e	a0	e0	3b	4d	sr	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
f	17	2b	04	7e	bs	77	d6	26	e1	69	14	63	55	21	0c	7d

Gambar 3. 9 Tabel Substitusi Inv S-Box

Transformasi *ShiftRows* pada dasarnya adalah proses penggeseran *byte* dengan *byte* paling kiri akan dipindahkan menjadi bye palingkanan (rotasi). Transformasi ini diterapkan pada baris 2, baris 3, dan baris 4. Baris 2 akan mengalami pergeseran bit sebanyak satu kali, sedangkan baris ke 3 dan baris 4 masing-masing mengalami pergeseran bit sebanyak dua kali dan tiga kali. Sedangkan transformasi InvShiftrows merupakan proses pergeseran byte dengan arah yang berlawanan dengan transformasi Shiftrows yaitu ke arah kanan.



Gambar 3. 10 Proses *ShiftRows*



Gambar 3. 11 Proses InvShiftRows

Proses *MixColumns* mengoperasikan setiap elemen yang berada dalam satu kolom pada state. Elemen pada kolom dikalikan dengan suato polinomial tetap. Proses *InvMixcolumns* sama seperti transformasi Mixcolumns hanya berbeda polinomial pengalinya.

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Gambar 3. 12 Polonomial tetap pada *MixColumns*

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix}$$

Gambar 3. 13 Polinomial tetap pada *InvMixColumns*

Proses transformasi terakhir yaitu *AddRoundKey* yang merupakan penjumlahan bitwise XOR antara plaintext dengan *key* (dirubah ke dalam notasi biner terlebih dahulu)

BAB IV
HASIL DAN ANALISIS PENELITIAN

4.1 Hasil dan Analisis

Pada tahap ini, dilakukan pengujian terhadap sistem yang telah dikembangkan guna mengevaluasi kinerja dari metode enkripsi *AES-CBC* dalam mengamankan *citra digital* yang disimpan di *cloud*. Beberapa aspek yang diuji meliputi kualitas *citra digital* setelah proses enkripsi dan dekripsi, kecepatan proses enkripsi serta dekripsi, serta penggunaan memori sistem. Hasil pengujian ini dianalisis untuk menilai sejauh mana sistem yang dikembangkan mampu menjaga keamanan dan efisiensi dalam pengelolaan *citra digital*.

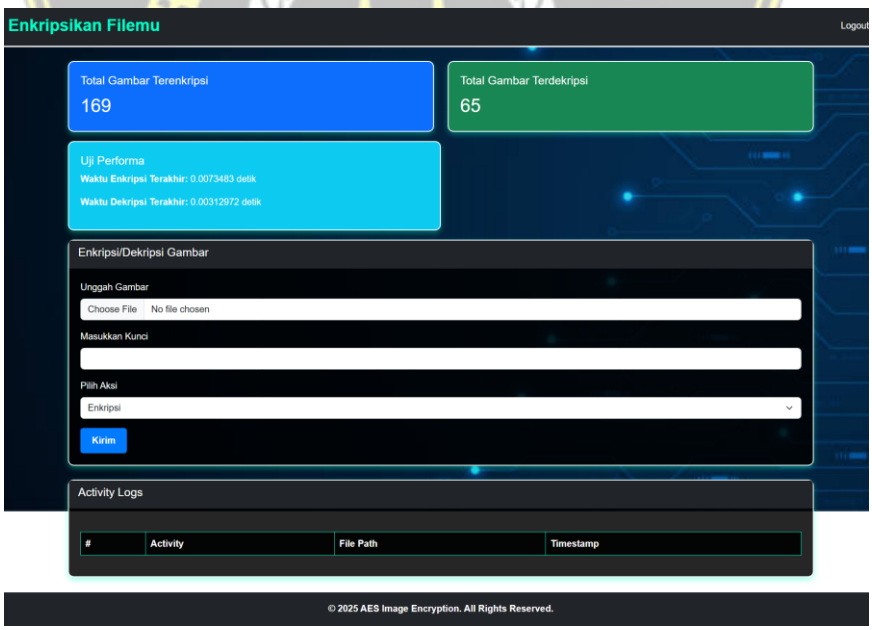
Pengujian kualitas *citra digital* dilakukan menggunakan metrik *Structural Similarity Index Measure* (*SSIM*) guna mengukur sejauh mana perubahan yang terjadi pada citra digital setelah melalui proses enkripsi dan dekripsi. Dari hasil pengujian, diperoleh nilai *SSIM* yang menunjukkan bahwa citra digital hasil dekripsi masih memiliki tingkat kemiripan yang tinggi dengan *citra digital* asli, yang berarti informasi dalam *citra digital* tetap dapat digunakan tanpa mengalami degradasi yang signifikan.

Selain itu, dilakukan pengujian terhadap kecepatan enkripsi dan dekripsi untuk mengetahui efisiensi algoritma dalam menangani citra digital berukuran besar. Hasil pengujian menunjukkan bahwa sistem mampu melakukan enkripsi dan dekripsi dengan waktu yang relatif cepat, sehingga tidak menghambat proses penyimpanan dan akses data dalam *cloud*. Pengujian penggunaan memori juga dilakukan untuk memastikan bahwa implementasi sistem dapat berjalan secara optimal tanpa menghabiskan sumber daya komputasi yang berlebihan. Hasil pengujian menunjukkan bahwa penggunaan memori masih dalam batas wajar, sehingga sistem dapat diterapkan dalam lingkungan medis yang memerlukan efisiensi tinggi.

4.2 Implementasi Sistem

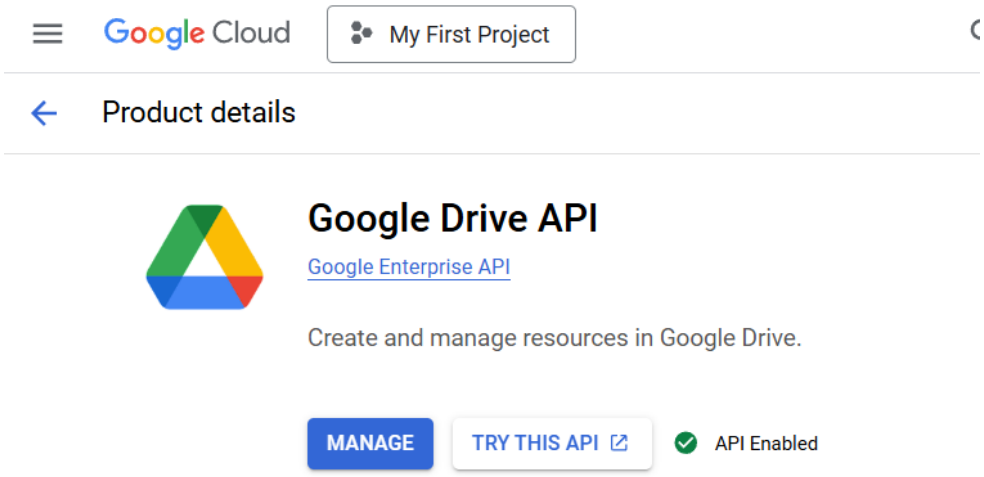
Sistem yang dikembangkan memiliki antarmuka berbasis web yang memungkinkan pengguna untuk melakukan proses enkripsi dan dekripsi *citra digital* dengan mudah. Pada antarmuka utama, pengguna dapat mengunggah gambar medis dalam format tertentu, seperti *DICOM*, *X-ray*, atau *MRI*, kemudian memilih opsi untuk melakukan enkripsi. Setelah proses enkripsi selesai, sistem menyediakan tombol untuk mengunduh gambar yang telah dienkripsi serta opsi untuk mengunggahnya ke layanan *cloud*, dalam hal ini Google Drive.

Setelah pengguna mengunggah gambar terenkripsi, sistem juga menyediakan fitur untuk melakukan dekripsi guna mengembalikan gambar ke bentuk aslinya. Antarmuka ini dirancang agar intuitif dengan tata letak yang sederhana dan mudah digunakan. Selain itu, sistem juga mencatat aktivitas enkripsi dan dekripsi dalam tabel *ActivityLog* untuk keperluan pemantauan dan analisis keamanan.



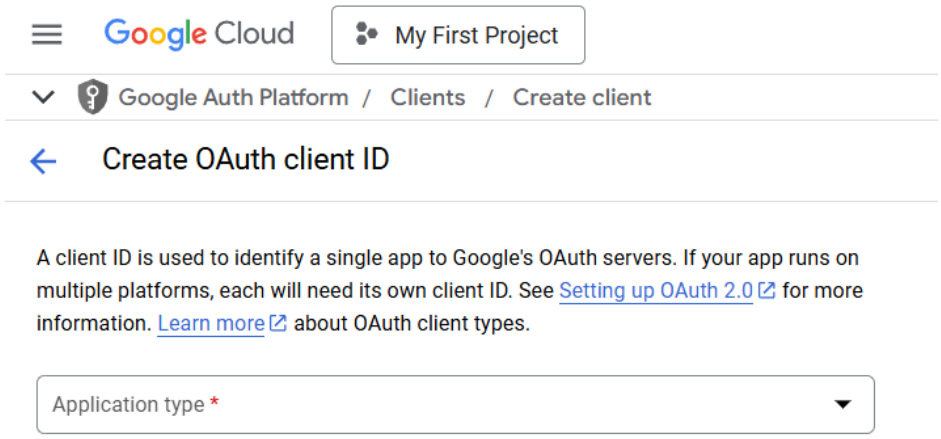
Gambar 4. 1 Halaman Utama Dashboard

Selain halaman utama, sistem juga menyediakan fitur autentikasi menggunakan *Google OAuth 2.0* agar pengguna dapat menyimpan file terenkripsi ke akun Google Drive masing-masing. Dengan fitur ini, setiap pengguna memiliki akses eksklusif ke file yang mereka unggah. Berikut adalah Konfigurasi Google Drive:



Gambar 4. 2 Mengaktifkan Layanan Google Drive API

Langkah pertama dalam konfigurasi API Google Drive adalah membuat proyek baru di *Google Cloud Console*. Proyek ini berfungsi sebagai wadah untuk mengelola sumber daya API yang digunakan. Setelah proyek dibuat, layanan Google Drive API harus diaktifkan melalui *Google Cloud Console* agar aplikasi dapat berkomunikasi dengan Google Drive. Pengguna dapat mengakses halaman API & Services dan mencari Google Drive API, kemudian mengaktifkannya agar dapat digunakan dalam proyek.



Gambar 4. 3 Membuat OAuth Client ID

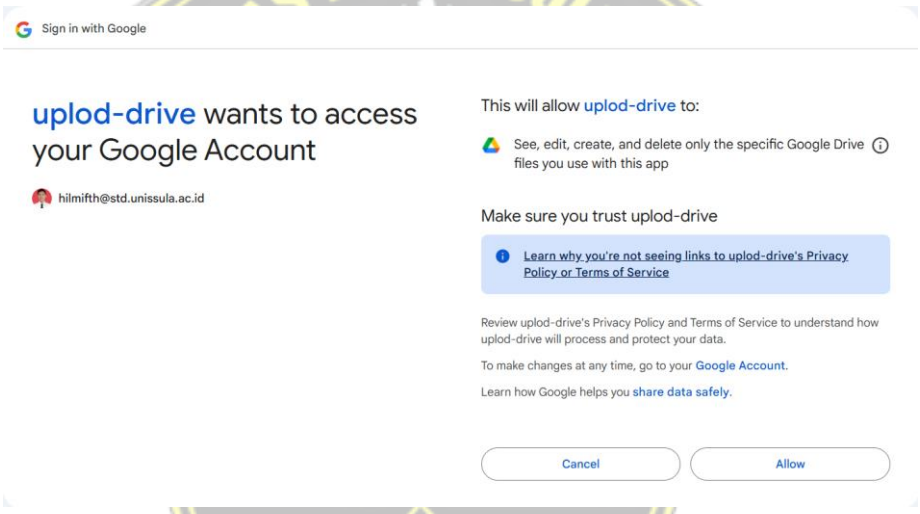
Setelah API diaktifkan, langkah berikutnya adalah membuat kredensial *OAuth 2.0*, yang digunakan untuk mengotorisasi akses aplikasi ke akun Google Drive pengguna. Kredensial ini dibuat dengan memilih opsi Create Credentials

pada *Google Cloud Console*, lalu memilih OAuth Client ID sebagai metode autentikasi.


```
{
  "web": {
    "client_id": "568419877187-6os3bb2au181cjnjjfm4q4u65jghadjq.apps.googleusercontent.com",
    "project_id": "phrasal-spirit-452416-k9",
    "auth_uri": "https://accounts.google.com/o/oauth2/auth",
    "token_uri": "https://oauth2.googleapis.com/token",
    "auth_provider_x509_cert_url": "https://www.googleapis.com/oauth2/v1/certs",
    "client_secret": "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",
    "redirect_uris": ["http://127.0.0.1:5000/callback"]
  }
}
```

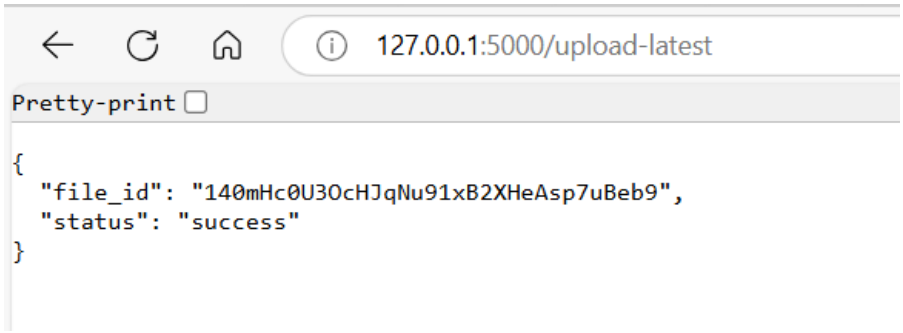

Gambar 4. 4 Client Secret

Gambar 3.13 diatas menunjukan hasil setelah kredensial dibuat, *Client ID* dan *Client Secret* yang dihasilkan harus disimpan untuk digunakan dalam implementasi autentikasi di aplikasi.



Gambar 4. 5 Proses OAuth 2.0

Pada gambar 3.14 menunjukkan proses autentikasi dimulai dengan mengarahkan pengguna ke halaman login Google menggunakan URL yang dibuat berdasarkan *OAuth Flow*. Setelah pengguna berhasil login dan memberikan izin, aplikasi menerima *authorization code*, yang kemudian digunakan untuk mendapatkan token akses (*access token*).



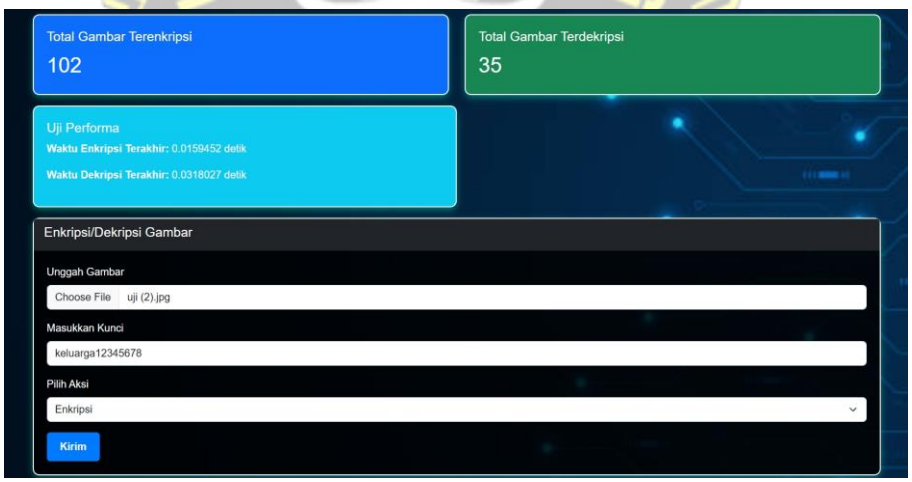
Gambar 4. 6 Akses token

Pada gambar 3.15 menunjukkan proses setelah autentikasi login drive dan mendapatkan akses token, dimana akses token ini memungkinkan aplikasi mengunggah file terenkripsi ke Google Drive atas nama pengguna.

4.3 Pengujian Sistem

4.3.1 Pengujian Fungsi Enkripsi dan Dekripsi

Dari program yang telah dibuat dilakukan pengujian enkripsi terhadap citra digital ‘‘uji (2).jpeg’’ dilakukan proses enkripsi dengan pembentukan IV (initialization Vector) secara acak dan memberikan kunci sepanjang 16 byte ‘‘keluarga12345678’’.

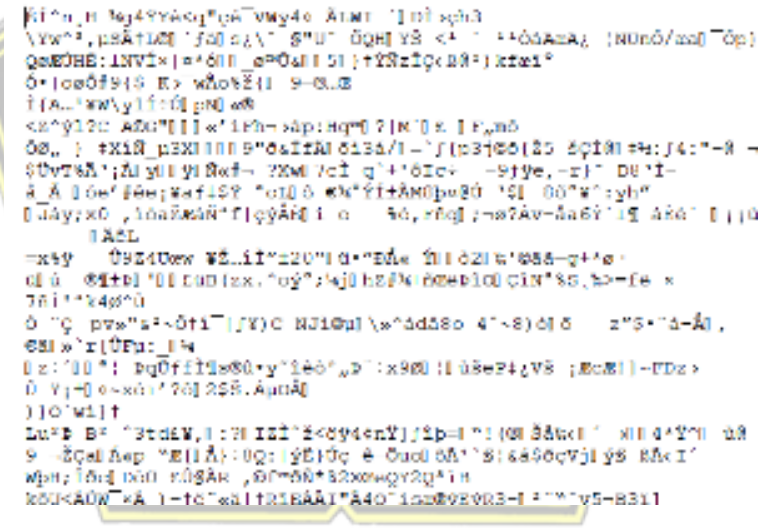


Gambar 4. 7 Tampilan Dashboard

Dari hasil pengujian untuk proses enkripsi berhasil dilakukan, dan output file yang telah dienkripsi tersimpan di folder ENCRYPTED_FOLDER. Untuk dile enkripsi diberikan penambahan ekstensi ‘‘.enc’’ dari hasil file pengujian.

Gambar 4. 8 Citra Asli

Gambar 4.8 diatas adalah citra asli sebelum di enkripsi dimana citra ini memiliki informasi visual yang dapat dibaca oleh manusia dan oerangkat lunak. Dalam konteks keamanan digital, citra asli atau KTP ini mengandung informasi sensitif yang perlu dilindungi dari akses tidak sah.



Gambar 4. 9 Citra Ternekripsi

Gambar 4.9 ini menunjukkan hasil dari proses enkripsi, hasilnya adalah citra yang telah diubah menjadi format yang tidak dapat dibaca oleh manusia tanpa proses dekripsi. Enkripsi dilakukan menggunakan algoritma seperti AES-CBC, di mana setiap piksel citra dikonversi menjadi data terenkripsi berdasarkan kunci rahasia. Dengan demikian, citra terenkripsi tidak bisa dipahami atau dikembalikan ke bentuk aslinya tanpa proses dekripsi yang tepat.

Gambar 4. 10 Citra Terdekripsi

Gambar 4.10 diatas menunjukkdaran hasil Citra terdekripsi adalah hasil dari proses dekripsi yang bertujuan untuk mengembalikan citra terenkripsi ke bentuk aslinya. Proses ini menggunakan kunci enkripsi yang sama dengan yang digunakan saat enkripsi (dalam kasus algoritma simetris seperti AES). Idealnya, citra terdekripsi harus memiliki kualitas yang mendekati citra asli, yang dapat diukur menggunakan metrik seperti SSIM (Structural Similarity Index Measure)

Dari tabel diatas fungsi enkripsi dan dekripsi berjalan dengan benar Dimana

citra digital yang terenkripsi menjadi tidak bisa dibaca dikarenakan isi dari file tersebut mengandung kode acak yang tidak bisa dibaca. Untuk fungsi dekripsi berjalan dengan benar dimana file yang terenkripsi bisa di balikkan lagi menjadi file asli.

4.3.2 Pengujian Kualitas Citra Menggunakan SSIM

Pengujian kualitas citra setelah proses enkripsi dan dekripsi dilakukan untuk mengetahui apakah terjadi perubahan signifikan terhadap citra asli. Salah satu metode yang digunakan adalah *Structural Similarity Index Measure* (SSIM), yang mengukur tingkat kesamaan struktural antara dua citra. SSIM lebih unggul dibandingkan metode tradisional seperti *Mean Squared Error* (MSE) karena mempertimbangkan luminansi, kontras, dan struktur dari citra.

Rumus SSIM didefinisikan sebagai berikut:

$$SSIM\ (\frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad 4.1$$

Dengan:

- μ_x dan μ_y adalah rata-rata intensitas piksel dari citra asli dan citra hasil dekripsi.
- $\sigma_x^2 \; dan \; \sigma_y^2$ adalah varians dari masing masing citra.
- σ_{xy} adalah kovarians antara citra asli dan citra hasil dekripsi.
- C_1 dan C_2 adalah konstanta untuk menghindari pembagian dengan nol.

Dalam penelitian ini, SSIM dihitung dengan pustaka *sckit-image* pada python. Perbandingan dilakukan antara citra asli sebelum di enkripsi dan citra hasil dekripsi untuk mengetahui tingkat kesamaan.

Pengujian dilakukan terhadap beberapa citra yang dienkripsi dan didekripsi menggunakan algoritma AES. Berikut adalah hasil perhitungan SSIM dari beberapa sampel citra.

Tabel 4. 1 Pengujian SSIM

No	Nama File	SSIM
1.	Dataset1.jpg	1.000000
2.	Dataset2.jpeg	1.000000
3.	Dataset3.jpeg	1.000000

Dari hasil pengujian diatas diperoleh nilai SSIM = 1.000000, yang menunjukkan bahwa ciyra asli dan citra hasil dekripsi identik secara struktural. Nilai SSIM yang mencapai 1.0 menandakan tidak adanya perubahan atau degradasi kualitas setelah proses enkripsi dan dekripsi. Dengan demikian, algoritma AEs mode CBC yang digunakan dalam siste mini terbukti mampu menjaga integritas citra tanpa kehilangan informasi visual.

Hasil ini mengindikasikan bahwa proses enkripsi hanya mengubah representasi biner citra tanpa merusak data asli, dan proses dekripsi berhasil mengembalikan citra ke bentuk semula. Oleh karena itu, metode yang digunakan untuk menjaga keamanan citra dalam layanan cloud tanpa mengurangi kualitas gambar setelah dilakukan enkripsi dan dekripsi.

4.3.2 Pengujian kecepatan Enkripsi dan Dekripsi

Pengujian ini dilakukan untuk mengukur waktu eksekusi enkripsi dan dekripsi pada berbagai ukuran file gambar. Hasil pengujian ditunjukkan pada tabel dibawah ini

Tabel 4. 2 Tabel uji kecepatan

Ukuran File (kb)	Waktu Enkripsi (s)	Waktu Dekripsi (s)
718 KB	0.031755 detik	0.016070 detik
1.381 KB	0.016015detik	0.015994 detik
3.421 KB	0.032031 detik	0.015926 detik
5.328 KB	0.032046 detik	0.025060 detik
12.355 KB	0.031669 detik	0.031858 detik
17.534 KB	0.123651 detik	0.143434 detik

Dari tabel pengujian diatas dapat disimpulkan bahwa:

1. Ukuran File mempengaruhi waktu enkripsi dan dekripsi

   Semakin besar ukuran file, semakin lama waktu yang dibutuhkan untuk proses enkripsi dan dekripsi. Misalnya, file berukuran 718 KB dienkripsi dalam 0.031755 detik, sedangkan file 17.534 KB membutuhkan 0.123651 detik.

2. Perbedaan waktu enkripsi dan dekripsi

   Waktu enkripsi dan dekripsi cenderung tidak jauh berbeda, tetapi dalam beberapa kasus, dekripsi bisa lebih lambat dari enkripsi. Contohnya, file 17.534 KB memiliki waktu enkripsi 0.123651 detik, sedangkan dekripsi memakan waktu 0.143434 detik.

3. Stabilitas waktu proses pada ukuran tertentu

   Pada ukuran file sedang (3.421Kb hingga 12.355KB), waktu enkripsi relatif stabil dikisaran 0.03detik, sedangkan dekripsi berada direntan 0.015 hingga 0.03 detik.

Secara keseluruhan, hasil ini menunjukkan bahwa algoritma AES-CBC yang digunakan cukup efisien dalam proses enkripsi dan dekripsi, meskipun terdapat sedikit peningkatan waktu seiring bertambahnya ukuran file.

4.3.3 Pengujian penggunaan memori

Untuk mengukur efesiensi sistem pengujian dilakukan dengan mengamati konsumsi memori selama proses enkripsi dan dekripsi.

Tabel 4. 3 Tabel Uji Penggunaan Memori

Ukuran File (kb)	Penggunaan Memori Enkripsi	Penggunaan memori dekripsi
718 KB	2.421875 MB	1.355469 MB
1.381 KB	4.054688 MB	4.054688 MB
3.421 KB	10.031250 MB	10.035156 MB
5.328 KB	15.605469 MB	15.609375 MB
12.355 KB	15.609375 MB	15.609375 MB
17.534 KB	51.375000 MB	51.375000 MB

Dari tabel hasil pengujian konsumsi memori didapati bahwa:

1. Penggunaan memori meningkat seiring bertambahnya ukuran file.

   Semakin besar ukuran file yang dienkripsi atau didekripsi, semakin besar pula penggunaan memori. Misalnya, file berukuran 718 KB membutuhkan 2.421875 MB untuk enkripsi, sedangkan file 17.534 KB membutuhkan 51.375000 MB.

2. Konsumsi memori enkripsi dan dekripsi cenderung sama

   Untuk file dengan ukuran lebih besar (di atas 3.421 KB), konsumsi memori untuk enkripsi dan dekripsi hampir identik. Contohnya, file 12.355 KB dan 17.534 KB memiliki penggunaan memori yang sama antara proses enkripsi dan dekripsi.

3. Perbedaan penggunaan memori pada file kecil

   Pada file kecil (718 KB), terlihat bahwa proses enkripsi memerlukan lebih banyak memori (2.421875 MB) dibandingkan dekripsi (1.355469 MB). Hal ini menunjukkan bahwa pada ukuran file kecil, enkripsi cenderung lebih membebani memori dibandingkan dekripsi.

Secara keseluruhan, hasil uji ini menunjukkan bahwa algoritma enkripsi dan dekripsi AES-CBC yang digunakan memiliki konsumsi memori yang cukup besar,

terutama untuk file dengan ukuran lebih besar. Namun, penggunaan memori untuk enkripsi dan dekripsi relatif stabil pada file yang lebih besar.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai implementasi algoritma AES dalam optimalisasi keamanan citra digital pada layanan *cloud*, dapat disimpulkan bahwa sistem enkripsi dan dekripsi menggunakan AES-CBC bekerja secara optimal dalam mengamankan citra digital tanpa mengurangi kualitas gambar setelah proses enkripsi dan dekripsi. Pengujian SSIM (Structural Similarity Index) menunjukkan bahwa nilai SSIM antara citra asli dan hasil dekripsi adalah 1.000000, yang menandakan bahwa citra yang didekripsi identik dengan citra aslinya tanpa adanya perubahan kualitas. Dari segi performa, proses enkripsi dapat dilakukan dalam waktu sekitar 0.02 - 0.03 detik, sedangkan dekripsi lebih cepat, yaitu sekitar 0.002 - 0.005 detik, menunjukkan bahwa *AES-CBC* bekerja secara efisien tanpa menyebabkan latensi yang signifikan. Selain itu, pengujian terhadap penggunaan memori menunjukkan bahwa proses enkripsi membutuhkan sekitar 1.03 MB, sementara dekripsi hanya menggunakan 0.14 MB, yang berarti enkripsi memerlukan lebih banyak sumber daya dibandingkan dekripsi, tetapi masih dalam batas wajar untuk implementasi pada sistem berbasis *cloud*. Dengan integrasi sistem ke dalam layanan *cloud* menggunakan Google Drive API, hasil enkripsi dapat langsung diunggah ke akun masing-masing pengguna, memastikan bahwa file tetap aman dan hanya dapat diakses dengan autentikasi yang sesuai

5.2 Saran

1. **Evaluasi Keamanan Pada Layanan Cloud**

   Penggunaan teknik tambahan seperti enkripsi end-to-end atau mekanisme otentikasi berbasis token dapat dieksplorasi untuk meningkatkan perlindungan data yang diunggah ke cloud.

2. **Pengujian Lebih Lanjut**

   Perlu dilakukan pengujian terhadap berbagai format gambar (JPEG, PNG, BMP, TIFF) untuk melihat pengaruhnya terhadap hasil enkripsi dan efisiensi algoritma. Pengujian dengan dataset gambar yang lebih besar dan kompleks perlu dilakukan untuk mengukur skalabilitas sistem.

DAFTAR PUSTAKA

Azanuddin, A., Yakub, S., & Prayudha, J. (2022). Implementasi Keamanan Citra Menggunakan Algoritma AES-128 Dengan Aplikasi Client-Server. *Jurasik (Jurnal Riset Sistem Informasi Dan Teknik Informatika)*, *7*(1), 51. https://doi.org/10.30645/jurasik.v7i1.415

Bibiola, F., Kalsum, T. U., & Alamsyah, H. (2023). Penerapan Algoritma Advance Encryption Standard (AES) Untuk Pengamanan File Pada Aplikasi Berbasis WEB. *Jurnal Surya Energy*, *8*(1), 35. https://doi.org/10.32502/jse.v8i1.6461

Chandra, R. V. H., Kusyanti, A., & Data, M. (2019). Analisis Performa Proses Enkripsi dan Dekripsi Menggunakan Algoritme AES-128 Pada Berbagai Format File. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, *3*(1), 481–486. http://j-ptiik.ub.ac.id

Cristy, N., & Riandari, F. (2021). Implementasi Metode Advanced Encryption Standard (AES 128 Bit) Untuk Mengamankan Data Keuangan. *Jurnal Ilmu Komputer Dan Sistem Informasi (JIKOMSI)*, *4*(2), 75–85. https://ejournal.sisfokomtek.org/index.php/jikom/article/view/181

Di, P., Fashion, P., & Xyz, C. V. (2014). *Jurnal Ilmiah Niagara Vol. XVI No. 2, Desember 2014*. *XVI*(2), 137–153.

Dwi Setyo Wiratomo, Bayu Hananto, & I Wayan Widi Pradnyana. (2022). Jurnal 4 - IMPLEMENTASI KEAMANAN FILE PADA APLIKASI. *Senamika*, 521–530.

Fajriati Romli, S., Id Hadiana, A., & Rakhmat Umbara, F. (2023). Penerapan Kriptografi Advanced Encryption Standard (AES) dan Steganografi Spread Spectrum Untuk Mengamankan Pesan Dalam Gambar. *DES 2023 Journal of Informatics and Communications Technology*, *5*(2), 196–209.

Joseph, D. P., & Krishna, M. (2015). *Cognitive Analytics and Comparison of Symmetric and Asymmetric Cryptography Algorithms*. *6*(3), 51–56.

Manullang, S., Allwine, & Jakaria Sembiring. (2023). Pengamanan Data File Dokumen Menggunakan Algoritma Advanced Encryption Standard Mode Chiper Block Chaining. *Antivirus : Jurnal Ilmiah Teknik Informatika*, *17*(1),

52–65. https://doi.org/10.35457/antivirus.v17i1.2811

Poetro, B. S. W., Studi, P., Informatika, T., & Diponegoro, U. (2010). Kriptografi Citra Digital dengan Algoritma Rijndael dan Transformasi Wavelet Diskrit Haar. *Prosiding Seminar Nasional Ilmu Komputer Universitas Diponegoro*, 175–178. http://eprints.undip.ac.id/33564/1/KRIPTOGRAFI_CITRA_DIGITAL_DENGAN_ALGORITMA_RIJNDAEL.pdf

Prameshwari, A., & Sastra, N. P. (2018). Implementasi Algoritma Advanced Encryption Standard (AES) 128 Untuk Enkripsi dan Dekripsi File Dokumen. *Eksplora Informatika*, *8*(1), 52. https://doi.org/10.30864/eksplora.v8i1.139

Putra Ramadani Tarigan, A., Ramadhan, P. S., & Ibnutama, K. (2023). Penerapan Kriptografi Untuk Pengamanan Data Penjualan Sepatu Dengan Metode AES (Advanced Encryption Standard). *Jurnal Cyber Tech*, *5*(1), 26. https://doi.org/10.53513/jct.v5i1.7851

Rahman, S., Sembiring, A., Siregar, D., Khair, H., Gusti Prahmana, I., Puspadini, R., & Zen, M. (2023). Python : Dasar Dan Pemrograman Berorientasi Objek. In *Penerbit Tahta Media*.

Sidiq, R. F., Rahayu, R. E. G., & Supriatna, A. D. (2023). Implementasi Kriptografi Advanced Encryption Standard dan Least Significant Bit untuk Keamanan Pesan Email dalam Gambar. *Jurnal Algoritma*, *20*(2), 305–315. https://doi.org/10.33364/algoritma/v.20-2.1407

Soesanto, E., Romadhon, A., Dwi Mardika, B., & Fahmi Setiawan, M. (2023). Analisis dan Peningkatan Keamanan Cyber: Studi Kasus Ancaman dan Solusi dalam Lingkungan Digital Untuk Mengamankan Objek Vital dan File. *SAMMAJIVA : Jurnal Penelitian Bisnisdan Manajemen*, *1*(2), 186.

Studi, P., Informatika, T., Teknik, F., & Batam, U. P. (2022). *Implementasi Caesar Cipher Pada Algoritma*.

Tarisa Auliya Ramadhani, Fajaryanto Cobantoro, A., & Sugianti, S. (2024). Implementasi Algoritma Advanced Encryption Standard 128 untuk Pengamanan Database Sistem Registrasi Pasien. *Jurnal Informatika Polinema*, *10*(4), 521–526. https://doi.org/10.33795/jip.v10i4.5619


UNIVERSITAS ISLAM SULTAN AGUNG
UNISSULA
جامعة سلطان أجونج الإسلامية