

Pengamanan Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES) Pada Kantor Badan Kepegawaian Dan Pengembangan Sumber Daya Manusia Kab.Aceh Singkil

Document Security Using the Advanced Encryption Standard (AES) Cryptographic Algorithm at the Office of the Civil Service and Human Resources Development Agency, Aceh Singkil Regency

Mhd. Try Zuchrin Andriano Z. Lubis^{a,1}, Fhery Agustin^{b,2}

^{a,b}Program Studi Informatika, Universitas Potensi Utama, KL. Yos Sudarso Km. 6,5 No. 3A, Medan, 20241, Indonesia

tioandrianotio@gmail.com¹, fheryagustin@gmail.com²

ABSTRAK

Perkembangan teknologi informasi yang semakin pesat menuntut adanya sistem pengamanan data dan dokumen yang lebih efektif, khususnya pada instansi pemerintahan. Kantor Badan Kepegawaian dan Pengembangan Sumber Daya Manusia Kabupaten Aceh Singkil memiliki berbagai dokumen penting terkait data kepegawaian yang bersifat rahasia dan rentan terhadap penyalahgunaan apabila tidak dilindungi dengan baik. Untuk itu, diperlukan penerapan teknologi kriptografi dalam menjaga kerahasiaan, integritas, dan keamanan dokumen. Penelitian ini menerapkan algoritma Advanced Encryption Standard (AES) dalam proses enkripsi dan dekripsi dokumen. Metode AES dipilih karena memiliki tingkat keamanan yang tinggi, efisiensi dalam proses pengolahan data, serta telah menjadi standar internasional dalam pengamanan informasi. Hasil penelitian menunjukkan bahwa penerapan algoritma AES mampu mengamankan dokumen secara efektif, di mana dokumen asli yang dienkripsi tidak dapat dibaca tanpa kunci dekripsi yang sesuai, serta proses enkripsi dan dekripsi berjalan cepat dengan tingkat keberhasilan 100%. Dengan demikian, sistem ini dapat membantu meningkatkan keamanan dokumen di lingkungan Kantor Badan Kepegawaian dan Pengembangan Sumber Daya Manusia Kabupaten Aceh Singkil, sekaligus meminimalisir risiko kebocoran data penting.

Kata Kunci : Kriptografi, Advanced Encryption Standard (AES), Pengamanan Dokumen, Keamanan Data, BKPSDM Aceh Singkil.

ABSTRACT

The rapid development of information technology demands an effective system for securing data and documents, especially in government institutions. The Human Resources and Personnel Development Agency (BKPSDM) of Aceh Singkil Regency manages various confidential personnel documents that are vulnerable to misuse if not properly protected. Therefore, the implementation of cryptographic technology is required to ensure the confidentiality, integrity, and security of documents. This study applies the Advanced Encryption Standard (AES) algorithm in the encryption and decryption process of documents. AES was chosen because it provides a high level of security, efficiency in data processing, and is recognized as an international standard for information security. The results of this research show that the implementation of AES can effectively secure documents, where the encrypted files cannot be read without the appropriate decryption key, while the encryption and decryption processes run quickly with a 100% success rate. Thus, this system can enhance document security within the Human Resources and Personnel Development Agency of Aceh Singkil Regency and minimize the risk of sensitive data leakage.

Keywords : Cryptography, Advanced Encryption Standard (AES), Document Security, Data Protection, BKPSDM Aceh Singkil.

Info Artikel

Disubmit: 08 September 2025

Direview: 07 Oktober 2025

Diterima : 10 November 2025

Copyright © 2025 – senadimu. All rights reserved.

1. PENDAHULUAN

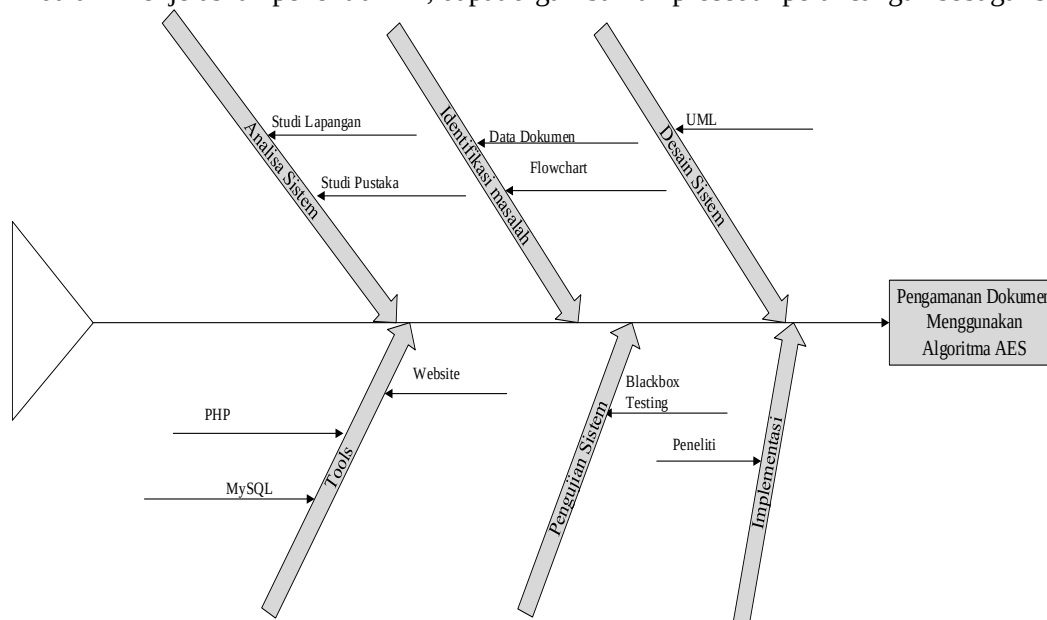
Ilmu teknologi yang semakin berkembang khususnya dalam bidang komputer berkaitan dengan semakin banyaknya penggunaan sistem komputer. Perkembangan ini pun berpengaruh terhadap keamanan komputer khususnya pengamanan file dokumen atau data. Keamanan dari data diperlukan untuk menjaga agar informasi yang tersimpan tidak dapat dibaca atau dibuka oleh pihak yang tidak berhak.

Salah satu misi Badan Kepegawaian Dan Pengembangan Sumber Daya Manusia Kabupaten Aceh Singkil adalah mewujudkan sistem administrasi kepegawaian yang sistematis, menyeluruh, terintegrasi dan berbasis teknologi digital. Dalam mewujudkan misi tersebut BKPSDM Aceh Singkil sering mengalami masalah keamanan. Permasalahan yang sering terjadi di BKPSDM Aceh Singkil adalah kebocoran dokumen yang berisi data rahasia dalam format *file* PDF sehingga banyak yang menyalah gunakan file tersebut untuk kepentingan pribadi. Lemahnya sistem keamanan data membuat penyadapan sangat mudah dilakukan sehingga penyebaran berita hoax sering ditemukan di internet. Kerusakan file yang sering terjadi membuat file memproses ulang dan membutuhkan waktu yang lama. Untuk mengatasi masalah tersebut dapat dilakukan dengan mengamankan data dengan menggunakan konsep kriptografi.

Kriptografi adalah bidang ilmu yang mempelajari bagaimana cara mengamankan suatu pesan atau informasi. Dalam bidang kriptografi terdapat dua konsep yang sangat penting yaitu enkripsi dan dekripsi. Proses pengiriman pesan akan melalui proses enkripsi untuk mengubah teks asli (*plaintext*) menjadi teks sandi (*ciphertext*). Dengan proses enkripsi suatu informasi akan menjadi lebih sulit untuk diketahui oleh orang yang tidak berhak. Keamanan tersebut diperlukan untuk menghindari adanya penyadapan terhadap file yang mengandung informasi penting bagi penggunaanya dan menjaga integritas tersebut agar tetap aman, maka dibutuhkan sebuah algoritma yang dapat memproteksi file adalah Algoritma Advanced Standard Encryption (AES).

2. METODE

Di dalam menjelaskan penelitian ini, dapat digambarkan prosedur perancangan sebagai berikut:



Gambar 1. Kerangka Fishbone

Deskripsi prosedur perancangan diatas adalah sebagai berikut :

1. Analisis Sistem
 - a. Studi Lapangan
Merupakan metode yang dilakukan dengan mengadakan studi langsung ke Kantor Badan Kepegawaian Dan Pengembangan Sumber Daya Manusia Kab.Aceh Singkil untuk mengumpulkan informasi yang berhubungan dengan keamanan data..
 - b. Studi Kepustakaan (*Library Research*)
Penulis melakukan studi pustaka untuk memperoleh refrensi yang berhubungan dengan penulisan skripsi dari berbagai sumber bacaan seperti buku panduan pembuatan aplikasi pengolah *database* MySQL dan PHP, UML, metode AES dan buku atau jurnal yang membahas tentang konsep pengamanan data..
2. Identifikasi Masalah
Kebocoran dokumen yang berisi data rahasia dalam format *file* PDF sehingga banyak yang menyalah gunakan file tersebut untuk kepentingan pribadi.
3. Desain Sistem
Adapun usulan pemecahan masalah untuk mengatasi permasalahan terhadap analisis sistem yang ada adalah sebagai berikut:
 - a. Merancang sebuah aplikasi yang dapat mempermudah pengamanan dokumen untuk Kantor Badan Kepegawaian Dan Pengembangan Sumber Daya Manusia Kab.Aceh Singkil.
 - b. Merancang sistem yang kuat dengan menerapkan algoritma kriptografi Advanced Encryption Standard (AES) pada aplikasi pengamanan dokumen untuk Kantor Badan Kepegawaian Dan Pengembangan Sumber Daya Manusia Kab.Aceh Singkil.
 - c. Merancang aplikas berbasis web pada Kantor Badan Kepegawaian Dan Pengembangan Sumber Daya Manusia Kab.Aceh Singkil sehingga proses ulang data tidak membutuhkan waktu yang lama.
 - d. Perancangan sistem menggunakan model *Unified Modelling Language* dengan *usecase diagram*, *class diagram*, *sequence diagram* dan *activity diagram*.
4. Tools
Menggunakan bahasa pemrograman PHP yang ditulis menggunakan aplikasi *Sublime Text*. Sedangkan pengolahan data menggunakan mesin *database* MySQL. Adapun Spesifikasi komputer yang digunakan minimal *dualcore*, RAM 4 GB serta Hard Drive 500 Gb.
5. Pengujian Sistem
Berisi langkah-langkah yang dilakukan saat pengujian peralatan secara keseluruhan, besaran-besaran yang akan diuji, dan ukuran untuk menilai apakah alat sudah bekerja dengan baik sesuai spesifikasi.
6. Implementasi
Menerapkan sistem ke dalam lingkungan nyata dan memastikan semua terhubung secara fisik dan digital.

3. HASIL DAN PEMBAHASAN

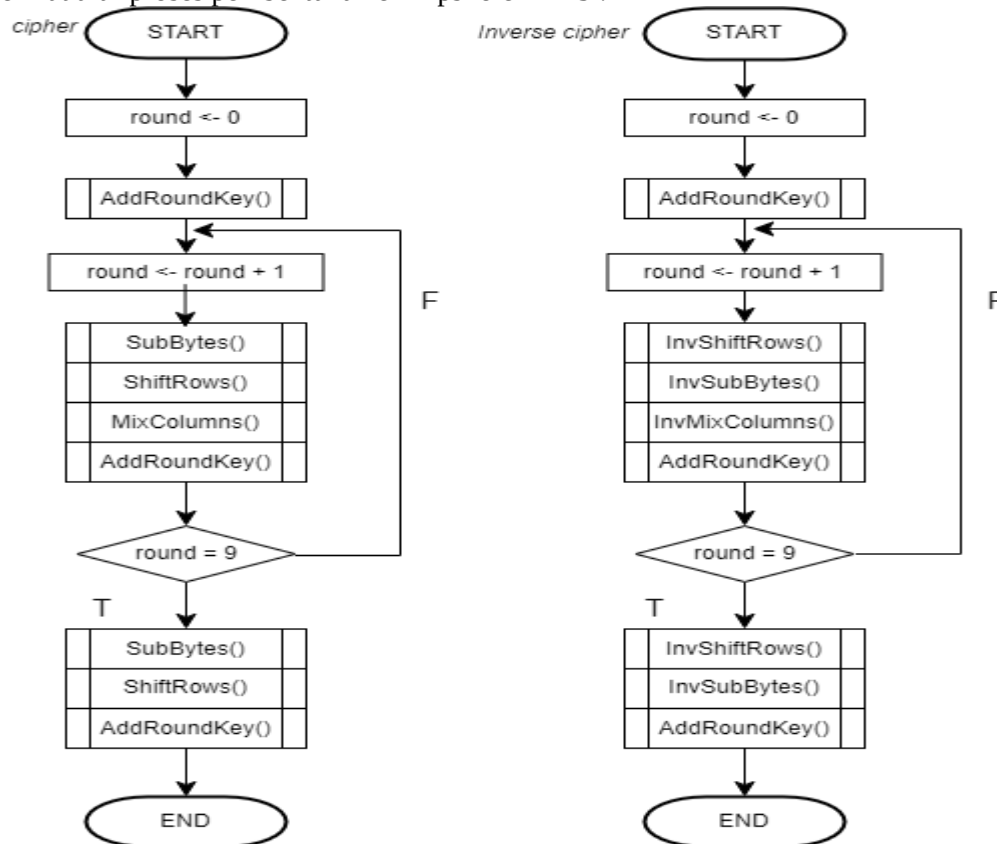
A. Pembahasan

Analisis ini bertujuan mengidentifikasi masalah utama dan merumuskan solusi teknis yang dapat diimplementasikan. Berikut adalah isu-isu spesifik yang diidentifikasi:

1. Dokumen sensitif disimpan tanpa *enkripsi*, meningkatkan risiko kebocoran data.
2. Proses manual menyebabkan inefisiensi dalam pencarian dan pengelolaan dokumen.
3. Tidak adanya sistem *log* untuk melacak aktivitas pengelolaan dokumen.
4. Ketiadaan mekanisme otentikasi dan otorisasi yang kuat untuk akses dokumen.
5. Data dokumen tidak terorganisir berdasarkan kategori atau departemen.

Untuk mengatasi masalah tersebut, diusulkan pengembangan sistem berbasis *web* yang mengintegrasikan algoritma *kriptografi AES-256* untuk mengamankan dokumen. Sistem ini akan menyediakan fitur otentikasi pengguna dengan *role* (*Admin* dan *Pegawai*), pengelolaan dokumen berdasarkan kategori dan departemen, serta pencatatan *log* untuk setiap aktivitas. *AES-256* dipilih karena kekuatannya dalam mengenkripsi data dengan kunci 256-bit, memastikan kerahasiaan dokumen sensitif. Sistem akan menggunakan *PHP* sebagai bahasa pemrograman utama, *SQLite* sebagai basis data ringan, dan antarmuka pengguna yang responsif untuk memudahkan akses. Fitur utama mencakup *upload* dokumen dengan *enkripsi* otomatis, pengelolaan data departemen dan kategori, serta laporan *log* untuk memantau aktivitas. Pendekatan ini memungkinkan pengelolaan dokumen yang efisien, aman, dan terlacak, sekaligus mendukung kebutuhan BKPSDM Aceh Singkil dalam lingkungan yang sederhana dan mudah diimplementasikan oleh pengembang pemula. Berikut adalah strategi teknis yang diusulkan untuk implementasi sistem.

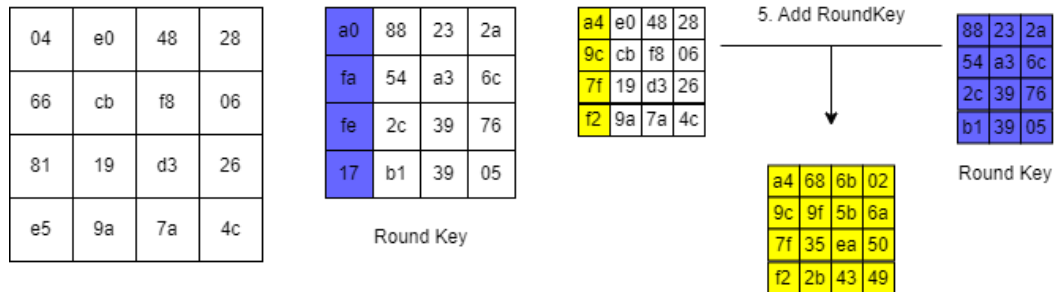
Berikut alur proses pembentukan enkripsi oleh AES :



Gambar 2 Alur Proses Pembentukan Enkripsi oleh AES

I. AddRoundKey

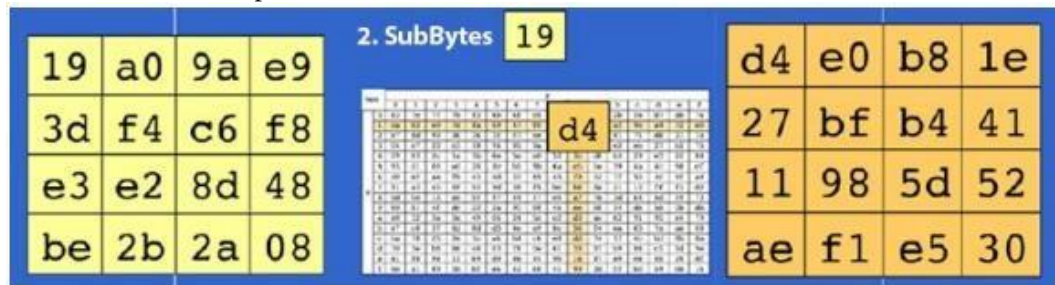
Melakukan XOR antara *state* awal (plaintexts) dengan *chipper key*. Tahap ini juga disebut initial round



Gambar 3 Proses AddRoundKey

II. Putaran sebanyak $Nr - 1$ kali . Putaran yang dilakukan pada setiap putaran adalah :

- a. *SubBytes* : Pada tahap ini, setiap byte dalam blok data digantikan oleh byte lain berdasarkan pada tabel *S-Box* .



Gambar 4 Proses Subbytes

- b. *ShiftRows*: Langkah ini mengubah posisi byte dalam setiap baris blok data sesuai dengan aturan tertentu. Hal ini menyebabkan data tersebar dengan lebih baik dalam blok, meningkatkan keamanan enkripsi.
- c. *MixColumns*: Setiap kolom dalam blok data dikalikan dengan matriks tetap. Proses ini menciptakan efek difusi, yang berarti setiap bit pada keluaran enkripsi dipengaruhi oleh banyak bit masukan.
- d. *AddRoundKey*: Tahap terakhir melibatkan proses XOR (eksklusif OR) antara setiap byte dalam blok data dengan kunci putaran (*round key*) yang telah dijadwal sebelumnya. Kunci putaran berubah pada setiap langkah, memberikan lapisan keamanan tambahan.

III. Final Round

Pada tahapan terakhir ini tidak ada operasi *MixColumn*. Hanya ada tiga langkah yang diterapkan yaitu *SubBytes*, *ShiftRows* dan *AddRoundKey*.

a. Simulasi Proses Perhitungan AES

Plainteks : 32 43 f6 a8 | 88 5a 30 8d | 31 31 98 a2 | - - - -

Karena kurang dari 16 karakter maka ditambah dengan 04 04 04 04 sehingga menjadi:

Plainteks : 32 43 f6 a8 | 88 5a 30 8d | 31 31 98 a2 | 04 04 04 04

Chipperkey : 2b 7e 15 16 | 28 ae d2 a6 | - - - - | - - - -

Karena kurang dari 16 karakter maka ditambah dengan 08 sebanyak 8 kali sehingga menjadi :

Chipperkey : 2b 7e 15 16 | 28 ae d2 a6 | 08 08 08 08 | 08 08 08 08

Masukkan ke kolom 4 x 4

Plainteks :

32	88	31	04
43	5a	31	04
f6	30	98	04
a8	8d	a2	04

Chiperkey :

2b	28	08	08
7e	ae	08	08
15	d2	08	08
16	a6	08	04

Konversikan ke biner
Plainteks , Chiperkey

Kita hitung dulu key schedule, hasil dari perhitungan akan digunakan pada proses selanjutnya untuk proses enkripsi. Langkahlangkah ekspansi kunci (key schedule) sebagai berikut:

1. Kolom *cipherkey* paling kanan dirotasi dan dikonversi menggunakan tabel SBox.
2. Kemudian hasilnya akan di xor dengan kolom *cipherkey* paling kiri dan di xor kan lagi dengan tabel rcon pada kolom 1.
3. Lakukan proses ini sampai semua kolom dihitung, sehingga akan diperoleh *roundkey* ke-n.

Contoh:

2b	28	08	08
7e	ae	08	08
15	d2	08	08
16	a6	08	04

2b	28	08	08		2b	30	01		1a	32	3a	32
7e	ae	08	08		7e	30	00	=	4e	e0	e8	e0
15	d2	08	08		15	30	00		25	f7	ff	f7
16	a6	08	08		16	30	00		26	80	88	80
							rcon					

Enkripsi

AddRoundKey atau juga bisa disebut sebagai *initial round*

32	88	31	04
43	5a	31	04
f6	30	98	04
a8	8d	a2	04

Plainteks :

2b	28	08	08
7e	ae	08	08
15	d2	08	08
16	a6	08	04

Chiperkey :

Studi kasus :

Dalam contoh ini, kita akan fokus pada bagaimana kunci AES dan data pesan di enkripsi AES. Untuk memberikan contoh perhitungan AES, kita perlu menjelaskan bagaimana kedua algoritma ini bekerja secara teoritis. Karena perhitungan sebenarnya akan sangat rumit jika dilakukan secara manual, saya akan memberikan gambaran langkah-langkah yang diambil dan beberapa contoh angka untuk menunjukkan proses ini.

Generasi Kunci AES

- Klien memilih kunci privat acak, misalnya $x = 6$
- Kunci public dihitung sebagai : $y = g^x \bmod p$
 $= 5^6 \bmod 23 = 8$
- Jadi, kunci publik klien adalah $y = 8$, dan kunci privatnya $x = 6$

Pembuatan Kunci AES

- Server menghasilkan kunci AES secara acak, misalnya $K_{AES} = 19$

Enkripsi Kunci AES

- Server memilih bilangan acak $k = 7$
- Hitung nilai a dan b untuk mengenkripsi kunci AES:
 - o a dihitung dengan rumus:

$$a = g^k \bmod p$$

$$= 5^7 \bmod 23 = 17$$
 - o b dihitung dengan rumus $b = K_{AES} * y^k \bmod p$

$$= 19 * 8^7 \bmod 23$$

Perhitungan intermediate o $8^7 \bmod 23 =$
 $2097152 \bmod 23$
 $= 16$

- o Maka :

$$b = 19 * 16 \bmod 23$$

$$= 304 \bmod 23 = 5$$

Jadi, kunci AES yang terenkripsi adalah $(a.b) = (17,5)$

Enkripsi Pesan dengan AES

I. Konversi Pesan ke Bentuk Biner atau Byte

- Kunci AES yang dihasilkan adalah $K_{AES} = 19$
- Misalkan pesan yang akan dikirimkan adalah "HELLO"
- Setiap karakter dalam "HELLO" diubah menjadi nilai ASCII: o H = 72 o E = 69 o L = 76 o L = 76 o O = 79

II. *Padding* Untuk Menyesuaikan Ukuran Blok

- AES menggunakan blok 128-bit atau 16 byte
- Data yang akan dienkripsi saat ini berukuran 5 byte, sehingga perlu ditambahkan *padding*, yaitu 11 (0*0B dalam hexadecimal)
- Maka, data yang akan dienkripsi setelah *padding* menjadi:
 $[72, 69, 76, 76, 79, 11, 11, 11, 11, 11, 11, 11, 11, 11, 11]$
- Dalam notasi *hexadecimal*, data ini akan direpresentasikan sebagai:
 $0x48454C4C4F0B0B0B0B0B0B0B0B0B$

III. Pemilihan Mode Operasi dan *Inisialisasi Vector*

- Mode operasi yang digunakan adalah CBC, yang memerlukan **initialization vector (IV)**
- IV adalah blok acak 128-bit yang biasanya dipilih secara acak untuk setiap pesan yang dienkripsi agar lebih aman. Misalnya, kita anggap IV ini adalah :
 $0x1A2B3C4D5E6F778899AABBCCDDEEFF00$

IV. Langkah Enkripsi Blok Pertama (128-bit)

- Setelah padding dan penambahan IV, kita siap untuk mengenkripsi blok data
- Enkripsi CBC dilakukan dengan Langkah-langkah berikut untuk setiap blok data (dalam hal ini, hanya satu blok karena pesan kecil) :
 - a. Melakukan XOR byte per byte
 - Byte Pertama : $0x48 \oplus 0x1A = 0x52$
 - Byte kedua : $0x45 \oplus 0x2B = 0x6E$
 - Byte ketiga : $0x4C \oplus 0x3C = 0x70$
 - Byte keempat : $0x4C \oplus 0x4D = 0x01$
 - Byte kelima : $0x4F \oplus 0x5E = 0x11$
 - ... lanjutkan hingga semua byte di-XOR
 - Hasilnya adalah blok data setelah XOR dengan IV:
 $0x526E700111041C0A91A6AFC0D4E4F0B$

V. Enkripsi Blok Data dengan AES

- Blok hasil XOR di atas kemudian dienkripsi menggunakan kunci AES ($K_{AES} = 19$)
- Dalam AES, kunci akan digunakan dalam beberapa round (putaran) enkripsi
- Setiap putaran enkripsi terdiri dari empat Langkah utama, yaitu SubBytes, ShiftRows, MixColumns, dan AddRoundKey
- Hasil enkripsi dari blok ini setelah semua putaran menghasilkan cipherteks berikut:
 $C = 0x89ABCD1234EF5678AABBCCDDEEFF1122$

VI. Menggabungkan IV dan Cipherteks

$IV \parallel C = 0x1A2B3C4D5E6F778899AABBCCDDEEFF00 \parallel$
 $0x89ABCD1234EF5678AABBCCDDEEFF1122$

Dekripsi oleh Klien

- Klien menerima pasangan (a,b) dan pesan C yang di enkripsi
- Klien akan mendekripsi kunci AES = 6
- Klien menghitung nilai s sebagai :
 $S = a^x \bmod p$
 $= 17^6 \bmod 23 = 13$
- Kemudian, klien menghitung K_{AES} sebagai :
 $K_{AES} = b * s^{-1} \bmod p$
 $= 5 * 14 \bmod 23 = 19$

Dekripsi pesan dengan AES :

I. Ekstraksi IV dengan Cipherteks

- Klien memisahkan IV dan cipherteks dari data yang diterima
- $IV = 0x1A2B3C4D5E6F778899AABBCCDDEEFF00$
 $C = 0x89ABCD1234EF5678AABBCCDDEEFF1122$

II. Dekripsi Cipherteks dengan kunci AES

- Klien mendekripsikan cipherteks menggunakan K_{AES} dalam mode CBC
- Dekripsi ini dilakukan dalam beberapa round (putaran) AES yang sama dengan jumlah putaran enkripsi
- Proses ini diulang sampai putaran pertama, menghasilkan data plainteks setelah dekripsi
- Plainteks setelah dekripsi = $0x526E700111041C0A91A6AFC0D4E4F0B$

III. XOR hasil dekripsi dengan IV

- Lakukan XOR byte per byte antara hasil dekripsi dan IV o Byte Pertama : $0x52 \oplus 0x1A = 0x48$

o Byte kedua : $0x6E \oplus 0x2B = 0x45$

o Byte ketiga : $0x70 \oplus 0x3C = 0x4C$

o Byte keempat : $0x01 \oplus 0x4D = 0x4C$ o Byte kelima :
 $0x11 \oplus 0x5E = 0x4F$

o ... lanjutkan hingga semua byte di-XOR

- Hasilnya adalah blok data asli berikut :

P (dengan padding) =

Q0x48454C4C4F0B0B0B0B0B0B0B0B0B0B

IV. Menghapus padding

- Hapus padding untuk mendapatkan data asli:

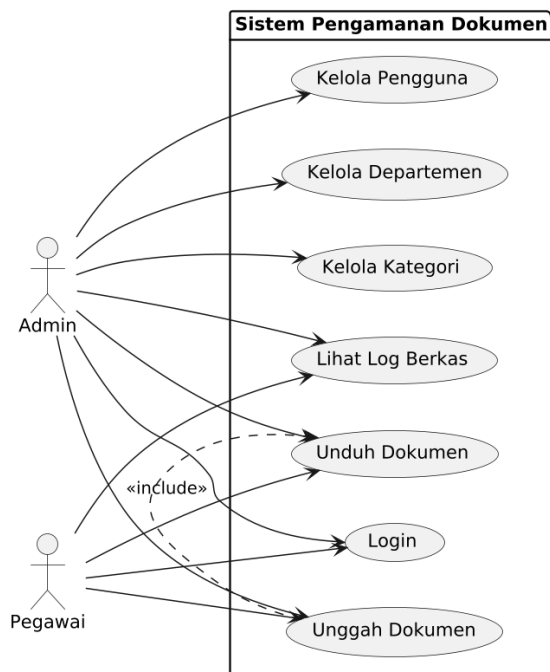
P tanpa padding = 0x48454C4C4F

Dalam bentuk teks ASCII , plainteks ini mewakili pesan “HELLO”

Kesimpulan

Penerpaan metode AES ini memungkinkan pengiriman data yang aman dengan mengenkripsi data menggunakan AES dan memastikan bahwa kunci AES tersebut sangat aman diterapkan pada dokumen kantor BPPSDM Kab.Aceh Singkil.

Sebuah *use case* digambarkan sebagai elips horizontal dalam suatu diagram UML *use case*, dapat dilihat pada gambar 2:



Gambar 5. Use Case Diagram

B. Hasil

1. Form Login

Halaman *Form Login* berfungsi sebagai pintu masuk pengguna ke dalam sistem. Antarmuka ini menampilkan *field* untuk memasukkan *email* dan *password*, dengan tombol *submit* untuk autentikasi. Pesan kesalahan ditampilkan jika autentikasi gagal, seperti *email* atau *password* salah. Halaman ini juga memiliki tautan untuk registrasi bagi pengguna baru. Antarmuka ini dioptimalkan untuk akses cepat dan responsif di berbagai perangkat berbasis *web*, memastikan pegawai kantor dapat masuk dengan mudah. Antarmuka Form Login dapat dilihat pada Gambar 6 berikut :



The screenshot shows a login interface for an application titled "Dokumen Aman". The header includes the application's logo and name. Below the header, there is a subtitle: "Aplikasi pengamanan dokumen menggunakan AES untuk BKDPSDM Aceh Singkil". A welcome message reads: "Selamat datang kembali! Masuk ke akun Anda." The form contains two input fields: "Email" with the value "admin@local.com" and "Password" with masked characters. There is a checkbox labeled "Simpan informasi login" which is checked. At the bottom, there is a green button labeled "Masuk".

Gambar 6. Form Login

2. Form Registrasi

Halaman *Form Registrasi* memungkinkan pengguna baru untuk membuat akun dengan memasukkan data seperti *email*, *password*, nama, dan peran (*role*). Tombol *submit* memicu proses penyimpanan data ke basis data SQLite setelah *enkripsi* menggunakan AES-256 untuk *password*. Antarmuka ini juga menyediakan pesan konfirmasi setelah registrasi berhasil. Desainnya memastikan kemudahan penggunaan bagi *staff* atau *admin* yang baru bergabung. Antarmuka Form Registrasi dapat dilihat pada Gambar 7 berikut :

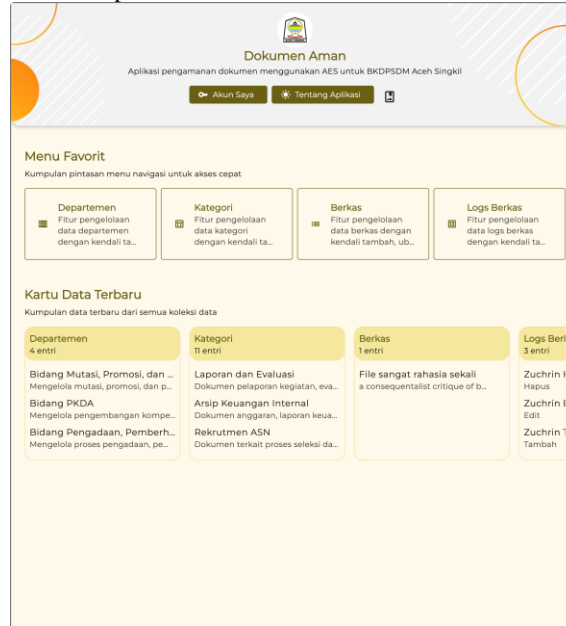


The screenshot shows a registration interface for the same application. The header is identical to the login form. Below the header, there is a subtitle: "Aplikasi pengamanan dokumen menggunakan AES untuk BKDPSDM Aceh Singkil". A welcome message reads: "Selamat datang! Daftar untuk memulai." The form contains two input fields: "Email" with the placeholder "Masukkan email Anda" and "Password" with the placeholder "Masukkan password" and a toggle icon. At the bottom, there is a green button labeled "Daftar". Below the button, there is a link: "Sudah punya akun? Masuk".

Gambar 7. Form Registrasi

3. Dashboard

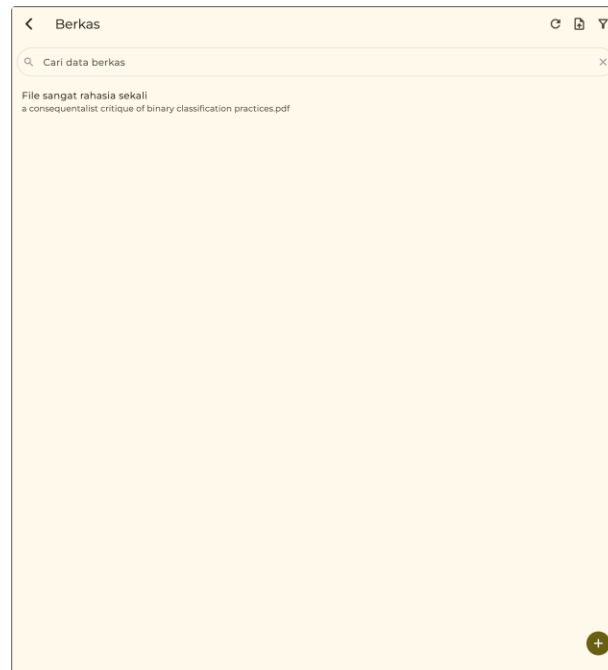
Halaman *Dashboard* berfungsi sebagai pusat navigasi utama setelah pengguna berhasil login. Antarmuka ini menampilkan ringkasan aktivitas, seperti jumlah dokumen, kategori, dan *log* terbaru, dalam bentuk *widget* atau *card*. *Admin* dapat melihat semua data, sementara *Staff* dan *Publik* memiliki akses terbatas sesuai peran. Desainnya responsif, memastikan tampilan optimal di desktop dan tablet. Antarmuka Dashboard dapat dilihat pada Gambar 8 berikut :



Gambar 8 Dashboard

4. Daftar Berkas

Halaman *Daftar Berkas* menampilkan daftar dokumen dalam bentuk tabel dengan kolom seperti nama dokumen, kategori, dan tanggal unggah. Pengguna dapat melakukan aksi seperti melihat detail, mengedit, atau menghapus dokumen (*Admin* only). Antarmuka ini dirancang untuk memudahkan pengelolaan dokumen dalam jumlah besar. Antarmuka Daftar Berkas dapat dilihat pada Gambar 9 berikut :



Gambar 9. Daftar Berkas

5. Form Berkas

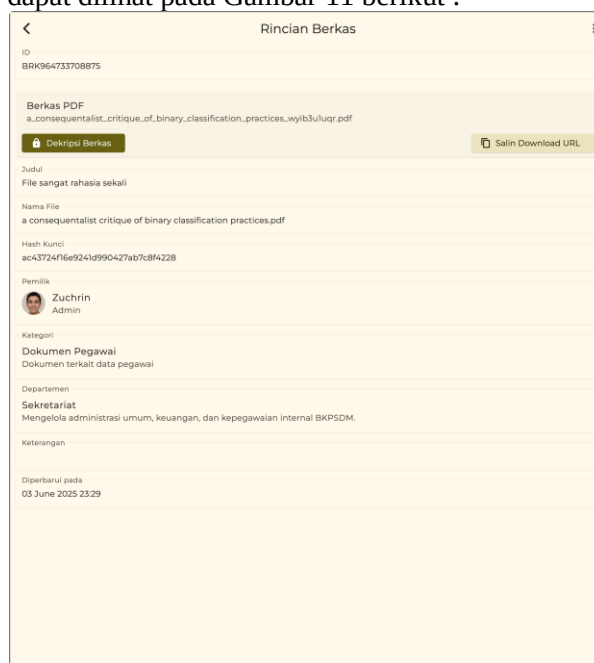
Halaman *Form Berkas* memungkinkan pengguna mengunggah dokumen baru. Antarmuka ini menyediakan *fields* untuk nama dokumen, kategori, dan unggahan file. Dokumen yang diunggah secara otomatis dienkripsi menggunakan AES-256 sebelum disimpan di basis data SQLite. Antarmuka Form Berkas dapat dilihat pada Gambar 10 berikut :

Gambar 10. Form Berkas

6. Detail Berkas

Halaman *Detail Berkas* menampilkan informasi lengkap tentang dokumen tertentu, seperti nama, kategori, tanggal unggah, dan *log* aktivitas terkait. Pengguna berwenang dapat mendekripsi dokumen untuk melihat konten asli. Antarmuka ini juga menyediakan tombol untuk mengedit atau menghapus

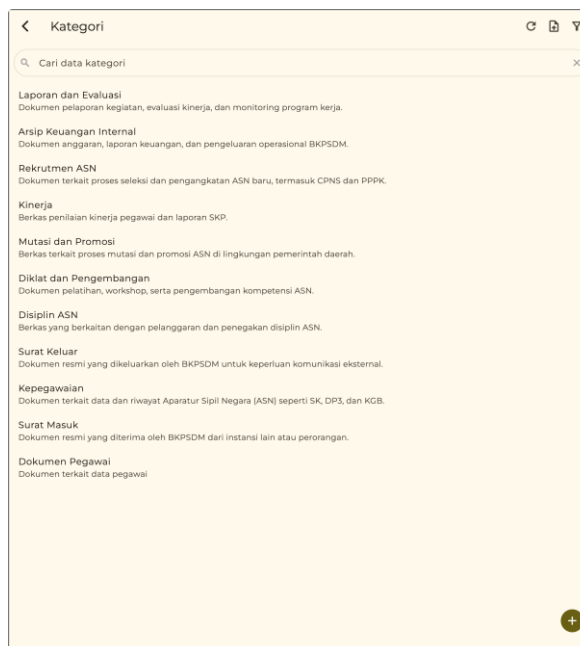
dokumen (*Admin* only). Desainnya fokus pada kejelasan informasi dengan tata letak sederhana. Antarmuka Detail Berkas dapat dilihat pada Gambar 11 berikut :



Gambar 11. Detail Berkas

7. Daftar Kategori

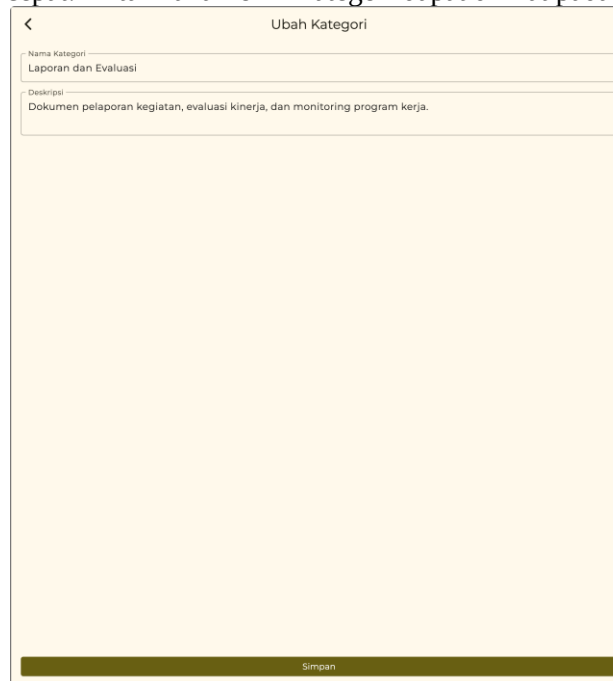
Halaman *Daftar Kategori* menampilkan daftar kategori dokumen dalam tabel dengan kolom seperti nama kategori dan jumlah dokumen terkait. Fitur *search* dan *filter* tersedia untuk memudahkan navigasi. *Admin* dapat menambah, mengedit, atau menghapus kategori. Antarmuka ini dirancang untuk mendukung pengelolaan kategori dokumen secara efisien. Antarmuka Daftar Kategori dapat dilihat pada Gambar 12 berikut :



Gambar 12. Daftar Kategori

8. Form Kategori

Halaman *Form Kategori* memungkinkan *Admin* menambah atau mengedit kategori dokumen. Antarmuka ini menyediakan *field* untuk nama kategori dan deskripsi singkat. Validasi memastikan nama kategori unik dan tidak kosong. Tombol *submit* menyimpan data ke SQLite. Desainnya sederhana untuk mendukung penggunaan cepat. Antarmuka Form Kategori dapat dilihat pada Gambar 13 berikut :

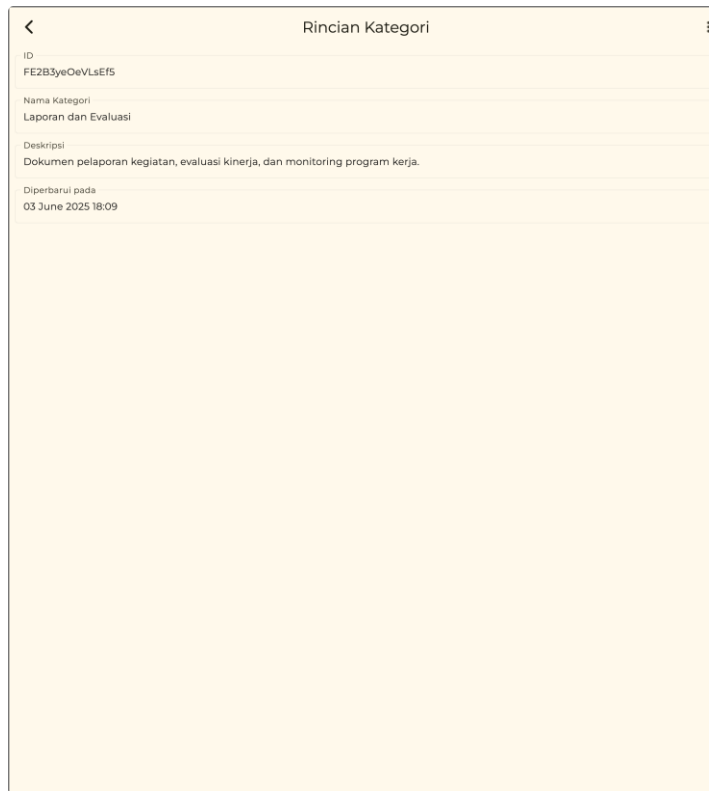


The image shows a mobile application interface for editing a category. At the top, there is a header bar with a back arrow on the left and the title 'Ubah Kategori' in the center. Below the header, there are two text input fields. The first field is labeled 'Nama Kategori' and contains the text 'Laporan dan Evaluasi'. The second field is labeled 'Deskripsi' and contains the text 'Dokumen pelaporan kegiatan, evaluasi kinerja, dan monitoring program kerja.' At the bottom of the form, there is a green button with the text 'Simpan' (Save).

Gambar 13. Form Kategori

9. Detail Kategori

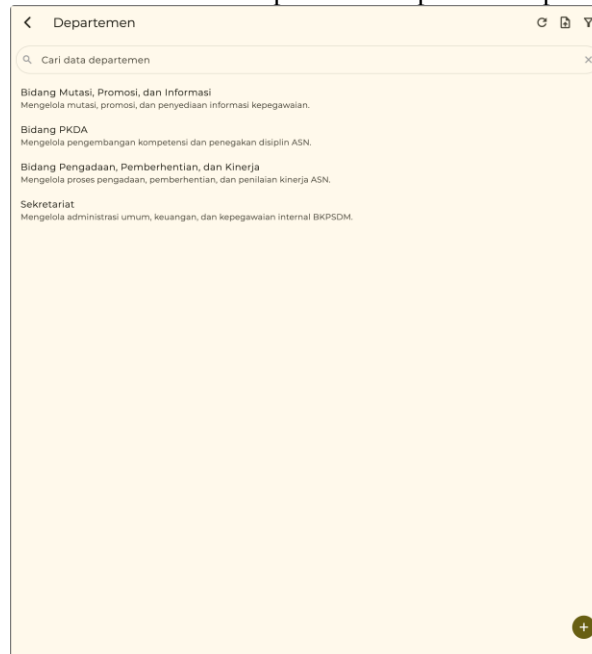
Halaman *Detail Kategori* menampilkan informasi lengkap tentang kategori tertentu, seperti nama, deskripsi, dan daftar dokumen terkait. *Admin* dapat mengedit atau menghapus kategori dari halaman ini. Antarmuka ini dirancang untuk memberikan gambaran jelas tentang kategori tertentu. Antarmuka Detail Kategori dapat dilihat pada Gambar 14 berikut :



Gambar 14. Detail Kategori

10. Daftar Departemen

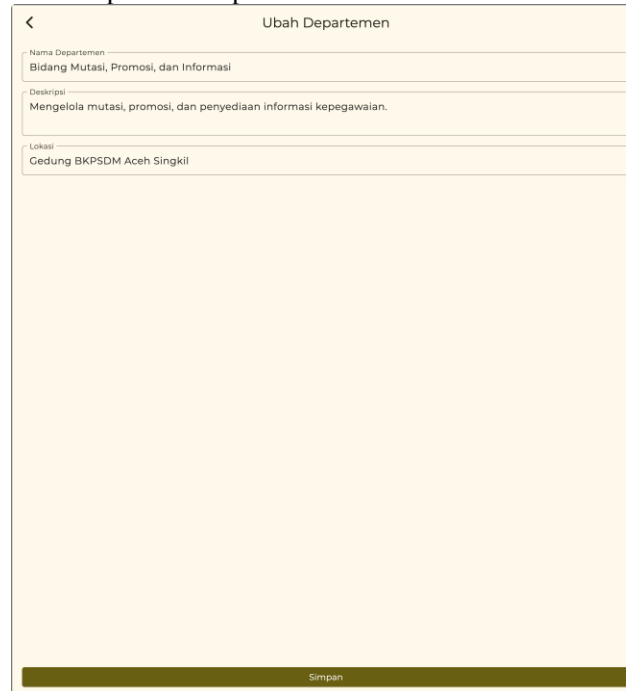
Halaman *Daftar Departemen* menampilkan daftar departemen dalam tabel dengan kolom seperti nama departemen dan jumlah dokumen terkait. Fitur *search* dan *filter* memudahkan navigasi. *Admin* dapat menambah, mengedit, atau menghapus departemen. Antarmuka ini mendukung pengelolaan data departemen secara efisien. Antarmuka Daftar Departemen dapat dilihat pada Gambar 15 berikut :



Gambar 15. Daftar Departemen

11. Form Departemen

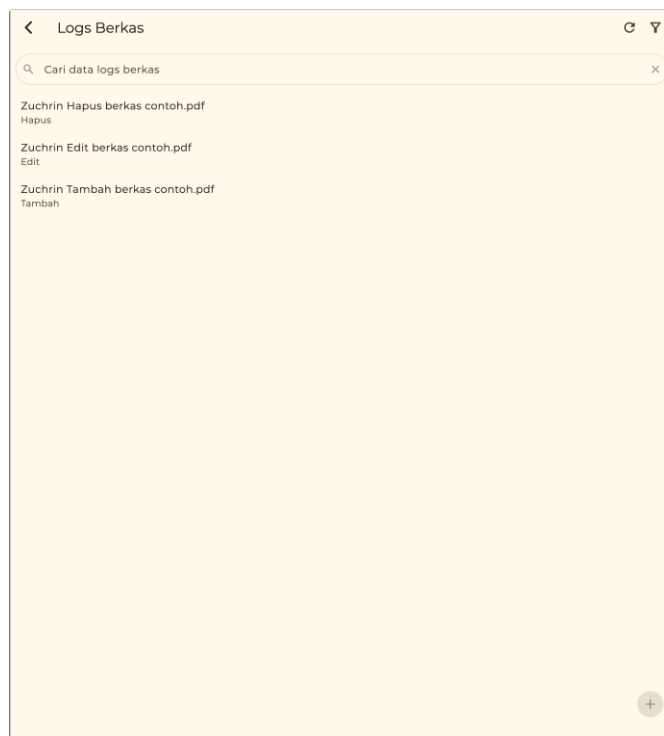
Halaman *Form Departemen* memungkinkan *Admin* menambah atau mengedit data departemen. Antarmuka ini menyediakan *field* untuk nama departemen dan deskripsi. Validasi memastikan data tidak kosong dan unik. Tombol *submit* menyimpan data ke SQLite. Desainnya sederhana dan fungsional. Antarmuka Form Departemen dapat dilihat pada Gambar 16 berikut :



Gambar 16. Form Departemen

12. Daftar Logs Berkas

Halaman *Daftar Logs Berkas* menampilkan riwayat aktivitas dokumen, seperti unggah, edit, atau dekripsi, dalam bentuk tabel. Kolom meliputi waktu, pengguna, dan jenis aktivitas. Fitur *filter* memungkinkan pencarian berdasarkan tanggal atau pengguna. Antarmuka ini membantu *Admin* memantau aktivitas sistem. Antarmuka Daftar Logs Berkas dapat dilihat pada Gambar 17 berikut :



Gambar 17. Daftar Logs Berkas

4. KESIMPULAN

Berikut adalah poin kesimpulan penelitian :

1. Sistem berhasil mengamankan dokumen menggunakan algoritma AES-256 dengan proses *enkripsi* dan *dekripsi* yang andal.
2. Fitur pengelolaan dokumen memungkinkan *Admin* mengatur dokumen, kategori, dan departemen secara efisien.
3. Pembatasan akses berdasarkan peran mencegah akses tidak sah, sesuai dengan kebutuhan kantor.
4. Pencatatan *log* aktivitas mendukung pelacakan dan audit dokumen secara transparan.
5. Antarmuka pengguna yang responsif dan intuitif memudahkan penggunaan oleh pegawai kantor.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Potensi Utama yang telah banyak memberikan masukan dan saran dalam penyelesaian penelitian ini.

REFERENSI

- [1] Azhari, M., Mulyana, D. I., Perwitosari, F. J., & Ali, F. (2022). Implementasi Pengamanan Data pada Dokumen Menggunakan Algoritma Kriptografi Advanced Encryption Standard (AES). *Jurnal Pendidikan Sains Dan Komputer*, 2(01), 163–171. <https://doi.org/10.47709/jpsk.v2i01.1390>
- [2] Cristy, N., & Riandari, F. (2021). Implementasi Metode Advanced Encryption Standard (AES 128 Bit) untuk Mengamankan Data Keuangan. *JIKOMSI [Jurnal Ilmu Komputer Dan Sistem Informasi]*, 4(2), 75–85. <https://ejournal.sisfokomtek.org/index.php/jikom/article/view/181%0A>
- [3] Fitriyani A, Handayani R, & Widanengsih E. (2020). Sistem Pendukung Keputusan Pemilihan Jurusan Pada SMK YMIK Joglo Jakarta Barat Menggunakan Metode Simple Additive Weigting

- (SAW). *Jtksi*, 03(01), 11–19.
- [4] Janis, J. W., Mamahit, D. J., Sugiarso, B. A., Rumagit, A. M., Elektro, T., Sam, U., & Manado, R. (2020). Rancang Bangun Aplikasi Online Sistem Pemesanan Jasa Tukang Bangunan Berbasis Lokasi. *Jurnal Teknik Informatika*, 15(1), 1–12. <https://doi.org/10.35793/jti.15.1.2020.29023>
- [5] Kaban, R., Yunita, W., & Faradillah, Y. (2019). Aplikasi Pemesanan Tiket Bus Berbasis Android (Study Kasus : Pt . Als Terminal Pasar X Tanjung Beringin). *Jurnal Manajemen ...*, 32(1).
- [6] Lubis, R. S., Tulus, & Nababan, E. B. (2022). Pengamanan File Teks Menggunakan Algoritma RSA – LUC dan Algoritma Zig- Zag dalam Hybrid Crypto Sistem. *InfoTekJar : Jurnal Nasional Informatika Dan Teknologi Jaringan*, 6(2), 185–189.
- [7] Marpaung, R., & Informasi, T. (2022). *PENYIMPANAN DATA PRIBADI DENGAN METODE*. 2(11), 1–14.
- [8] Muharromin, M., Informatika, J. T., & Darma, U. B. (n.d.). *Analisis Performance Web Application Firewall ModSecurity dan Shadow Daemon Dalam Keamanan Web Server Apache*. 393–402.
- [9] Nugroho, A. P., & Suseno, H. B. (2020). Keamanan Data Transaksi Nasabah Pada Aplikasi Bank Sampah Berbasis Web Menggunakan Algoritma AES. “*QUERY: Jurnal Sistem Informasi Keamanan Data Transaksi Nasabah Pada Aplikasi Bank Sampah Berbasis Web Menggunakan Algoritma AES.*,” 04(April), 9–17. <http://jurnal.uinsu.ac.id/index.php/query/article/view/8007/3720>
- [10] Perbawa, K. A. (2022). Application of Linear Congruential Generator (LCG) Algorithm in Android Based Mathematics Education Game Penerapan Algoritma Linear Congruential Generator (LCG) dalam Game Edukasi Matematika Berbasis Android. *Jurnal Komputer, Informasi Dan Teknologi*, 2(1), 47–56.
- [11] Prayudha, J., _ S., & _ I. (2019). Implementasi Keamanan Data Gaji Karyawan Pada PT. Capella Medan Menggunakan Metode Advanced Encryption Standard (AES). *Jurnal SAINTIKOM (Jurnal Sains Manajemen Informatika Dan Komputer)*, 18(2), 119. <https://doi.org/10.53513/jis.v18i2.150>
- [12] Ruing, M. O. I., & Ujianto, E. I. H. (2020). *Penerapan Kombinasi Algoritma Kriptografi (Caesar, Vigenere, Zig-Zag) Dan Metode Steganografi Lsb Untuk Mengamankan Pesan Ke Dalam Citra Digital*. 1–8. <http://eprints.uty.ac.id/4888/>
- [13] Setiaji, A. (2020). RANCANG BANGUN APLIKASI PEMESANAN DESAIN JERSEY BERBASIS ANDROID DENGAN MENGGUNAKAN TEKNOLOGI FIREBASE (Studi Kasus : Konfeksi Minister). *Jurnal Sistem Informasi Dan Sains Teknologi*, 2(2). <https://doi.org/10.31326/sistek.v2i2.664>
- [14] Sitepu, D. A., Nurhayati, & Khair, H. (2022). Implementasi Pengamanan Data Koperasi Menggunakan Algoritma Advanced Encryption Standard (AES). *CITISEE 2016 Proceedings*, 6(1), 37–40. [https://citisee.amikompurwokerto.ac.id/assets/proceedings/paper/8__Amikom_Purwokerto_Implementasi_Pengamanan_Data_Koperasi_Menggunakan_Algoritma_Advanced_Encryption_Standar_d_\(Aes\).pdf](https://citisee.amikompurwokerto.ac.id/assets/proceedings/paper/8__Amikom_Purwokerto_Implementasi_Pengamanan_Data_Koperasi_Menggunakan_Algoritma_Advanced_Encryption_Standar_d_(Aes).pdf)
- [15] Suparman, B. (2022). Aplikasi Pengamanan Data Menggunakan Kriptografi Dengan Metode Wake dan Algoritma Des Bebasis Java Desktop. *OKTAL: Jurnal Ilmu Komputer Dan Sains*, 1(07), 808–817. <https://journal.mediapublikasi.id/index.php/oktal/article/view/777%0Ahttps://journal.mediapublikasi.id/index.php/oktal/article/download/777/304>