BAB III
METODE PENELITIAN

3.1 Objek Penelitian

Fokus utama dari penelitian ini adalah penerapan algoritma enkripsi AES-128 dan Caesar Cipher dalam sistem pengamanan data penduduk berbasis aplikasi website. Website ini dirancang khusus untuk digunakan oleh admin, yang memiliki akses penuh untuk memasukkan data, melakukan enkripsi, serta mendekripsi informasi penduduk guna memastikan keamanan dan kerahasiaan data yang tersimpan dalam sistem.

AES-128 dipilih karena keamanannya yang tinggi serta efisiensi dalam proses enkripsi dan dekripsi, sementara Caesar Cipher digunakan sebagai lapisan tambahan untuk meningkatkan perlindungan data. Dengan kombinasi kedua algoritma ini, sistem dapat menerapkan *double encryption*, di mana data yang di *input* oleh admin akan terlebih dahulu dienkripsi menggunakan Caesar Cipher sebelum akhirnya diamankan lebih lanjut dengan AES-128.

Penelitian ini bertujuan untuk mengembangkan sistem pengamanan data berbasis web yang memungkinkan admin untuk mengelola informasi penduduk secara aman, serta mengevaluasi efektivitas kombinasi algoritma enkripsi dalam menjaga integritas data. Dengan demikian, penelitian ini diharapkan dapat menghasilkan sistem yang mampu mencegah kebocoran data, meningkatkan keamanan penyimpanan informasi dalam database, serta memastikan bahwa hanya pihak yang berwenang yang dapat mengakses dan mendekripsi data yang telah terenkripsi.

3.2 Metode Penelitian

Penelitian ini menggunakan metode Design and Development (D&D), yang menurut Ellis dan Levy (2010) merupakan suatu bentuk penelitian yang menekankan pada kontribusi terhadap pengembangan pengetahuan ilmiah, bukan sekadar pengembangan produk. *"The key differentiator between routine design and design research is the clear identification of a contribution to the archival*

Azka Ermaya Diwangsa, 2025
IMPLEMENTASI KRIPTOGRAFI AES-128 DAN CAESAR CIPHER UNTUK PROTEKSI DATA PENDUDUK BERBASIS APLIKASI WEBSITE
Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

knowledge base of foundations and methodologies". Dengan demikian, metode D&D relevan digunakan dalam penelitian karena tidak hanya menghasilkan artefak berupa aplikasi sebagai solusi atas permasalahan, tetapi juga memberikan kontribusi pada bidang penelitian. Metode ini memiliki kerangka enam tahap, mulai dari identifikasi masalah, perumusan tujuan, perancangan dan pengembangan artefak, pengujian, evaluasi, hingga komunikasi hasil penelitian (Ellis & Levy, 2010). Adapun alur penelitian metode Design and Development yang digunakan dalam penelitian ini dapat dilihat pada Gambar 3.1.

Gambar 3. 1 Alur Penelitian Metode *Design and Development*

3.2.1 Identifikasi masalah (*Identify the problem*)

Tahap pertama dalam metode *Design and Development* (D&D) adalah analisis, yang diawali dengan studi literatur mengenai topik penelitian yang akan dikembangkan. Studi literatur ini bertujuan untuk memahami perkembangan teknologi serta hasil penelitian terdahulu yang relevan. Dengan mengkaji berbagai referensi, penelitian dapat memperoleh wawasan mengenai solusi yang telah diuji sebelumnya serta menentukan pendekatan yang paling sesuai.

Dalam penelitian ini, analisis dilakukan untuk mengidentifikasi permasalahan keamanan data penduduk dalam sistem berbasis web, merumuskan permasalahan, serta menetapkan tujuan penelitian. Salah satu solusi yang dipilih adalah penerapan enkripsi menggunakan algoritma AES-128 dan Caesar Cipher, yang memungkinkan data dienkripsi sebelum disimpan dalam database. Studi literatur juga dilakukan untuk memahami cara kerja algoritma enkripsi tersebut serta implementasi keamanan data dalam sistem berbasis web. Selain itu, dilakukan analisis kebutuhan sistem yang mencakup pemilihan lingkungan pengembangan, bahasa pemrograman, serta struktur database guna memastikan sistem dapat berjalan dengan optimal. Hasil dari tahap ini menjadi dasar dalam proses perancangan dan pengembangan sistem.

3.2.2 Mendeskripsikan Tujuan (*Describe the objectives*)

Tujuan utama dari penelitian ini adalah untuk mengetahui cara mengimplementasikan algoritma enkripsi AES-128 dan Caesar Cipher dalam sistem berbasis web guna meningkatkan keamanan data penduduk. Penelitian ini bertujuan untuk mengeksplorasi bagaimana kedua algoritma tersebut dapat diterapkan pada aplikasi web untuk melindungi data sensitif penduduk, baik saat pengiriman maupun penyimpanan data. Selain itu, penelitian ini juga bertujuan untuk mengintegrasikan kedua algoritma enkripsi tersebut guna meningkatkan perlindungan data penduduk dari potensi ancaman keamanan. Dengan menggabungkan metode enkripsi AES-128 yang kuat dengan Caesar Cipher yang lebih sederhana, diharapkan dapat tercipta lapisan perlindungan yang lebih efektif terhadap data pribadi, sekaligus mengurangi risiko kebocoran atau serangan yang dapat mengancam keamanan informasi yang disimpan dalam aplikasi.

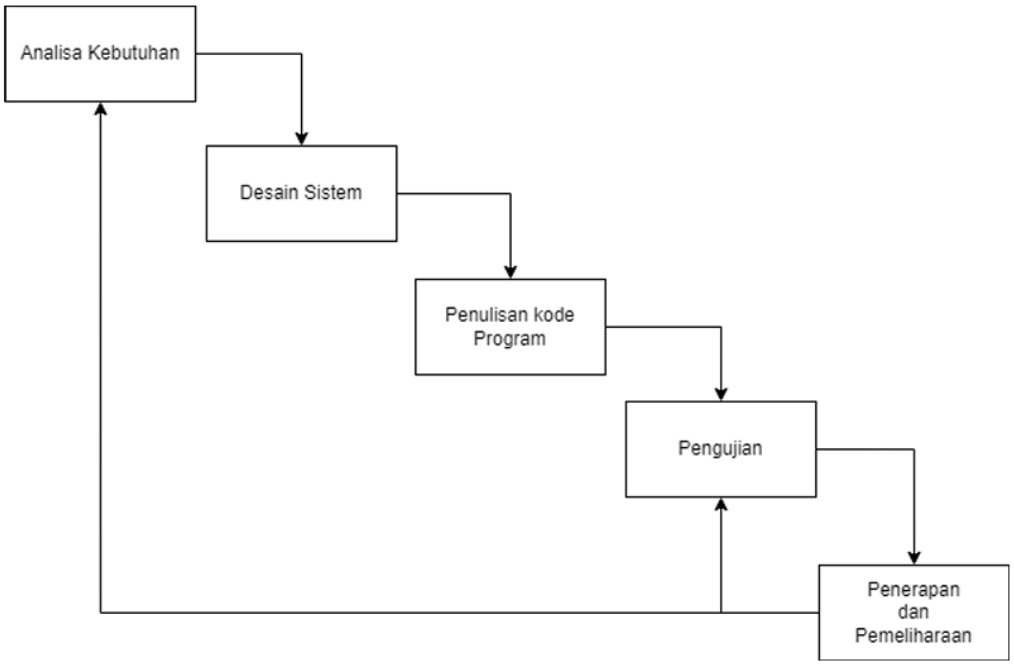
3.2.3 Desain Pengembangan Produk (*Design & develop the artifact*)

Tahap ketiga dalam metode *Design and Development* (D&D) adalah perancangan dan pengembangan sistem, di mana dilakukan pengembangan kerangka konseptual untuk mendesain aplikasi berbasis web yang mengimplementasikan algoritma enkripsi AES-128 dan Caesar Cipher. Model yang akan digunakan dalam penelitian ini adalah Model Waterfall yang merupakan salah satu model pengembangan sistem yang paling banyak digunakan dalam berbagai disiplin ilmu, termasuk dalam penelitian pengembangan.

Tahap pertama dalam model Waterfall adalah analisis kebutuhan, di mana peneliti atau pengembang mengidentifikasi dan mendokumentasikan secara rinci semua kebutuhan dan persyaratan yang diperlukan untuk sistem atau produk yang akan dikembangkan. Setelah itu, tahap kedua adalah desain, yang mencakup perancangan sistem atau solusi yang akan dikembangkan, baik dari sisi teknis maupun fungsional. Tahap berikutnya adalah pengembangan atau implementasi, di mana desain yang telah dibuat diterjemahkan menjadi kode atau produk nyata. Setelah pengembangan selesai, tahap pengujian dilakukan untuk memastikan bahwa sistem yang dibangun memenuhi kebutuhan yang telah ditetapkan dan berfungsi dengan baik. Terakhir, tahap pemeliharaan adalah proses berkelanjutan di

mana sistem yang sudah diterapkan tetap diperbarui dan diperbaiki sesuai dengan umpan balik atau kebutuhan baru yang muncul.

Meskipun model Waterfall menawarkan struktur yang jelas dan terorganisir, penting untuk diingat bahwa sifatnya yang linear terkadang dapat membatasi fleksibilitas, terutama jika terjadi perubahan kebutuhan di tengah jalan. Setiap fase dalam model Waterfall harus diselesaikan sebelum tim pengembang dapat bergerak ke fase berikutnya, menciptakan struktur yang teratur dan dapat diprediksi dalam proses pengembangan perangkat lunak (Hidayatulloh dkk., 2023). Pendekatan ini diilustrasikan dengan 5 langkah yang harus diselesaikan secara bertahap, untuk lebih jelasnya alur Waterfall dapat dilihat pada Gambar 3.2



Gambar 3. 2 Alur Kerja Model Waterfall

3.2.3.1 Analisa Kebutuhan

Pada tahap ini penulis melakukan analisa serta menetapkan kebutuhan perangkat keras dan perangkat lunak yang digunakan. Adapun perangkat keras yang digunakan selama pengembangan aplikasi adalah sebagai berikut:

1. Prosesor AMD Ryzen 7 4700U
2. RAM 16GB DDR 4

3. Penyimpanan 500 GB SSD

Selain perangkat keras, terdapat beberapa perangkat lunak yang digunakan selama pengembangan aplikasi ini, antara lain sebagai berikut:

Tabel 3. 1 Rincian Kebutuhan Teknologi Perangkat Lunak

No.	Nama Teknologi	Fungsi
1.	Visual Studio Code	VSCode adalah editor kode sumber yang digunakan untuk menulis, mengedit, dan mengembangkan aplikasi web. VSCode digunakan untuk menulis kode program (HTML, CSS, JavaScript, dan bahasa pemrograman lainnya) yang mendukung implementasi enkripsi AES-128 dan Caesar Cipher. Selain itu, VSCode memiliki dukungan untuk ekstensi yang mempermudah debugging dan integrasi dengan alat lainnya seperti Git.
2.	XAMPP	Digunakan sebagai server lokal untuk menjalankan dan menguji aplikasi website yang terhubung dengan database MySQL dan melakukan implementasi enkripsi untuk melindungi data penduduk.
3.	Java Script	Digunakan untuk mengimplementasikan enkripsi AES-128 dan Caesar Cipher, serta untuk berinteraksi dengan elemen-elemen HTML di halaman web. JavaScript juga dapat digunakan untuk mengelola alur data pengguna yang masuk ke sistem dan mengenkripsi data sebelum disimpan di server.
4.	Node Js	Node.js adalah platform berbasis JavaScript yang digunakan untuk menjalankan kode JavaScript di sisi server. Dalam penelitian ini,

No.	Nama Teknologi	Fungsi
		Node.js memungkinkan pengembangan aplikasi website berbasis JavaScript dengan kemampuan server-side.
5.	Express Js	Express.js adalah framework untuk Node.js yang mempermudah pembuatan API dan routing dalam aplikasi web. Dalam konteks penelitian ini, Express.js digunakan untuk membangun RESTful API yang mengatur komunikasi antara aplikasi web dan database, serta untuk menangani proses enkripsi dan dekripsi data penduduk menggunakan AES-128 dan Caesar Cipher.
6.	Crypto Js	CryptoJS berfungsi untuk mengimplementasikan enkripsi dan dekripsi data dalam aplikasi web, seperti yang digunakan dalam sistem perlindungan data penduduk berbasis AES-128 dan Caesar Cipher. Dengan CryptoJS, data sensitif seperti nama dan alamat dapat dienkripsi untuk menjaga kerahasiaannya dan hanya dapat dibaca oleh pihak yang memiliki akses sah. Pustaka ini juga memungkinkan dekripsi data terenkripsi, memastikan data tetap aman selama penyimpanan dan transmisi dalam aplikasi web.
7.	MySQL	MySQL adalah sistem manajemen database relasional yang digunakan untuk menyimpan data pengguna, seperti data penduduk dalam penelitian ini. Database MySQL akan menyimpan data yang telah dienkripsi dan

No.	Nama Teknologi	Fungsi
		melakukan proses dekripsi saat data diakses. MySQL mendukung pengelolaan data yang terstruktur dan dapat diintegrasikan dengan aplikasi backend menggunakan Node.js dan Express.js.
8.	RESTful API	RESTful API adalah arsitektur yang digunakan untuk menghubungkan *frontend* dengan *backend* aplikasi web. Dalam penelitian ini, RESTful API digunakan untuk melakukan pengolahan data penduduk dengan menambah, mengambil, dan menghapus data yang aman menggunakan enkripsi AES-128 dan Caesar Cipher.
9.	Github	GitHub adalah platform untuk *version control* dan kolaborasi kode menggunakan Git. Dalam penelitian ini, GitHub digunakan untuk menyimpan dan melacak perubahan kode yang digunakan dalam proyek pengembangan aplikasi website.
10.	Draw Io	Digunakan untuk merancang dan menggambarkan alur kerja aplikasi website, termasuk alur data yang mengenkripsi dan mendekripsi informasi penduduk.

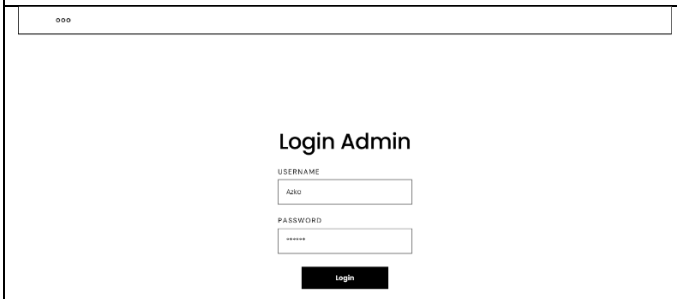
Dengan kombinasi perangkat keras dan perangkat lunak yang memadai, proses pengembangan sistem diharapkan dapat berjalan secara optimal dan mendukung implementasi enkripsi serta dekripsi data secara efisien.

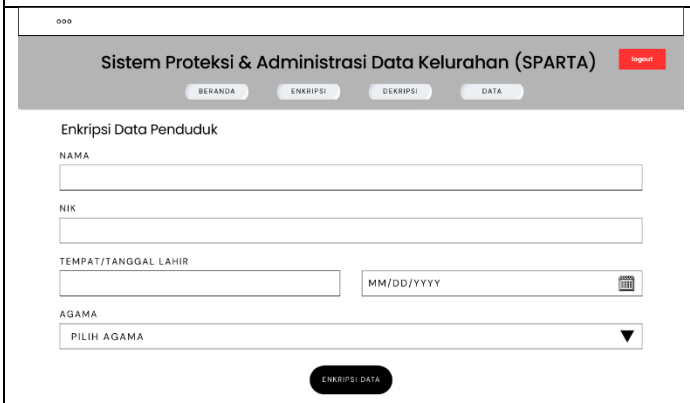
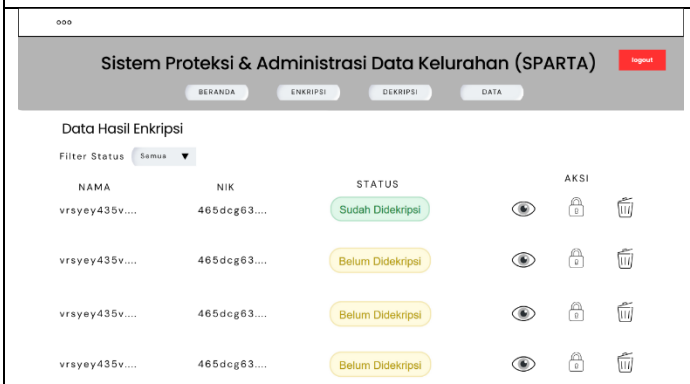
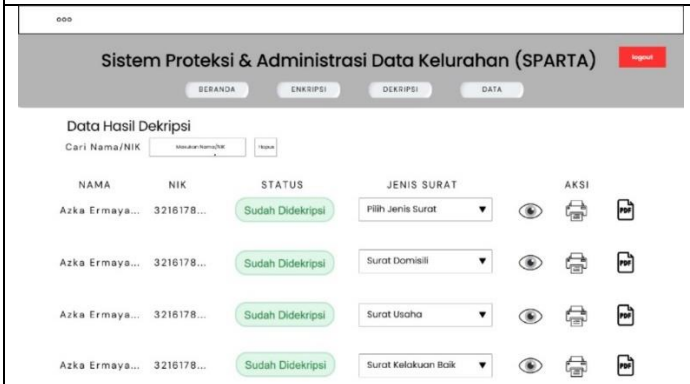
3.2.3.2 Desain Sistem

Pada tahap ini dilakukan perancangan desain sistem dalam membantu mendefinisikan arsitektur sistem pada tahap perancangan sistem perangkat lunak SPARTA (Sistem Pengelolaan Data Penduduk dengan Enkripsi), Proses ini mencakup pembuatan Mockup tampilan antarmuka sistem untuk memberikan gambaran visual awal mengenai bentuk dan susunan elemen pada halaman web, serta pembuatan desain sistem menggunakan diagram *Unified Modeling Language* (UML), seperti *use case* diagram, *flowchart system*, *activity* diagram dan *Entity Relationship* Diagram (ERD).

Mockup ini bersifat statis dan belum memiliki fungsi interaktif, namun berperan penting dalam membantu pengembang dan pemangku kepentingan memahami rancangan sebelum proses implementasi dilakukan. Pada tabel 3.2 tersaji *mockup* untuk website SPARTA (Sistem Pengelolaan Data Penduduk dengan Enkripsi).

Tabel 3. 2 Tampilan *Mockup* SPARTA

Tampilan	Deskripsi
	Gambar ini menunjukkan halaman *login* sistem. Tampilan terdiri dari *form username* dan *password* dengan tombol *login* yang ditempatkan secara sederhana di tengah layar. Desain ini memudahkan pengguna untuk langsung melakukan autentikasi sebelum mengakses sistem.

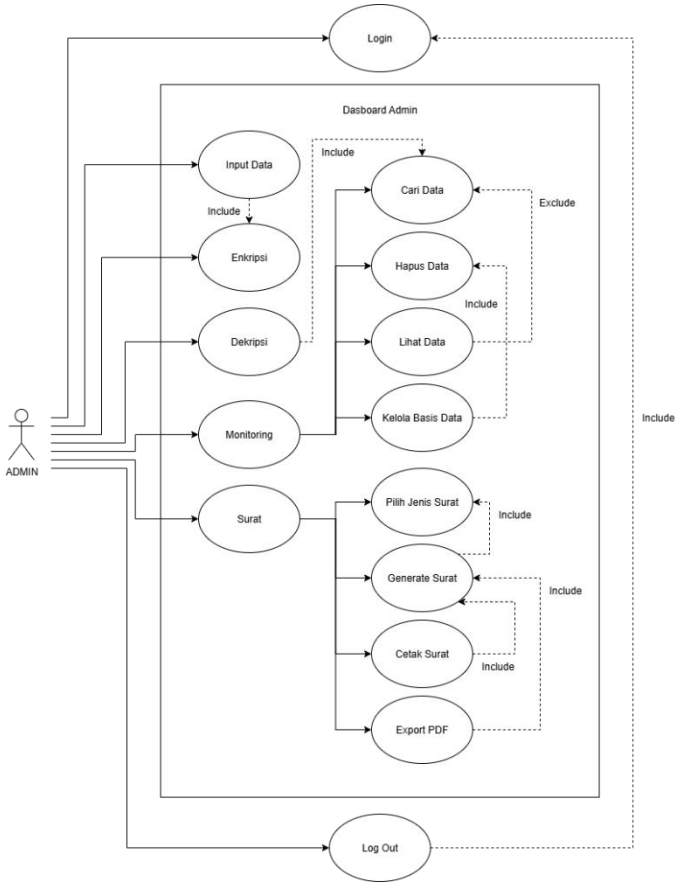
Tampilan	Deskripsi
	Pada gambar ini ditampilkan menu navigasi untuk mengakses fitur-fitur seperti beranda, enkripsi, dekripsi, dan manajemen data. Tampilan memperlihatkan halaman untuk *input* data penduduk dan melakukan proses enkripsi data.
	Gambar ini menampilkan halaman yang memperlihatkan hasil enkripsi dalam format hex. Pada kolom aksi, tersedia ikon untuk melihat detail data, proses dekripsi data, dan menghapus data.
	Gambar ini menampilkan hasil dekripsi data pada sistem. Halaman dilengkapi dengan menu navigasi utama di bagian atas serta kolom pencarian untuk memudahkan pengguna menemukan data

Tampilan	Deskripsi
	berdasarkan nama atau NIK. Bagian utama halaman berisi tabel yang menampilkan data hasil dekripsi, meliputi nama, NIK, status dekripsi, dan pilihan jenis surat yang akan dibuat. Pada kolom aksi, tersedia ikon untuk melihat detail data, mencetak surat, dan mengunduhnya dalam format PDF.

Penulis menggunakan Visual Studio Code sebagai IDE utama untuk pengembangan aplikasi web yang mengimplementasikan teknologi *frontend* standar (HTML5, CSS3, JavaScript ES6+) dengan integrasi library CryptoJS untuk enkripsi data, jsPDF dan html2canvas untuk pembuatan dokumen PDF, serta teknologi *backend* Node.js dengan Express.js sebagai web server untuk menangani HTTP *requests* dan mengelola *routing* API. Sistem menggunakan MySQL sebagai *database management system* dengan mysql2 *driver*, mengimplementasikan autentikasi berbasis bcrypt untuk keamanan password, serta komunikasi *client-server* melalui protokol HTTP/HTTPS dengan format JSON. Proses perancangan ini mendukung pembangunan aplikasi pengelolaan data penduduk dengan keamanan kriptografi ganda (AES-128 dan Caesar Cipher) dan fitur pembuatan surat otomatis, yang disajikan dalam bentuk diagram UML (*Use Case, Flowchart, Activity* dan ERD) untuk memberikan gambaran menyeluruh mengenai arsitektur, alur kerja, dan struktur data dalam sistem SPARTA.

1. Use Case Diagram

Use case diagram merupakan salah satu alat bantu penting dalam pemodelan perangkat lunak yang termasuk ke dalam *Unified Modeling Language* (UML). Diagram ini berfungsi untuk menggambarkan interaksi antara aktor dengan sistem, sekaligus memvisualisasikan fungsionalitas utama sistem dari perspektif pengguna (Yunita dkk., 2023). Diagram ini menjadi bagian esensial dalam analisis kebutuhan dan desain sistem, membantu pengembang untuk merumuskan dan mendokumentasikan kebutuhan fungsi yang diharapkan oleh pengguna akhir. Dengan menampilkan hubungan antara aktor (pengguna atau sistem luar) dan *use case* (fungsi/fitur yang dapat dimanfaatkan oleh aktor), diagram membantu dalam memahami sistem secara menyeluruh (Monalisa dkk., 2024).

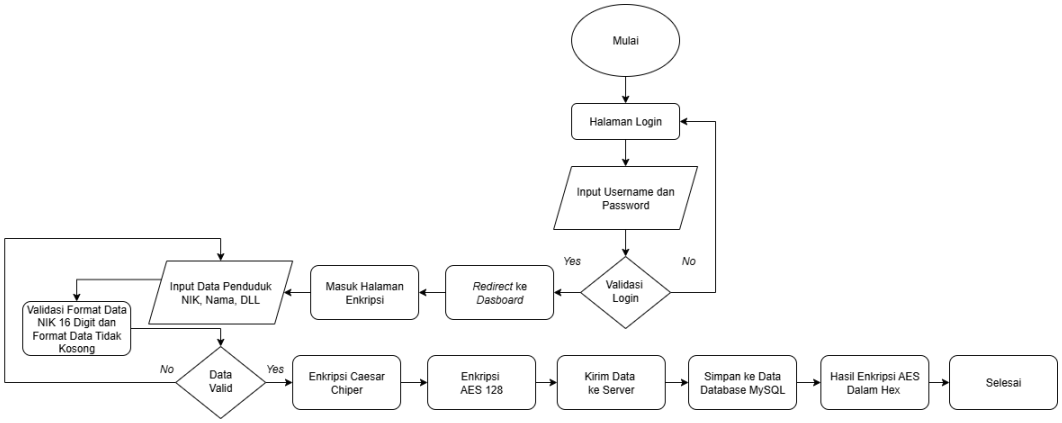


Gambar 3. 3 *Use Case* Diagram

Diagram *use case* yang terdapat pada Gambar 3.3 ini menggambarkan alur interaksi dalam sistem perangkat lunak untuk admin *dashboard*, admin sebagai satu-satunya pengguna yang memiliki akses penuh terhadap sistem. Dimulai dengan *Login* sebagai pintu masuk utama. Setelah *login* berhasil, admin dapat mengakses *Dashboard* Admin, yang memungkinkan beberapa fungsi, seperti *Input* Data yang mencakup proses Enkripsi dan Dekripsi untuk memastikan keamanan data. Selain itu, admin dapat melakukan Monitoring terhadap data yang ada, serta mencari dan menghapus data melalui Cari Data dan Hapus Data. Admin juga dapat Lihat Data untuk memverifikasi atau mengevaluasi informasi yang tersimpan, serta Kelola Basis Data untuk mengatur dan memelihara sistem. Di bagian Surat, admin dapat memilih Jenis Surat, kemudian melakukan *Generate* Surat, yang bisa langsung dicetak atau diekspor dalam bentuk PDF. Setelah semua aktivitas selesai, admin dapat melakukan Log Out untuk keluar dan mengamankan sistem.

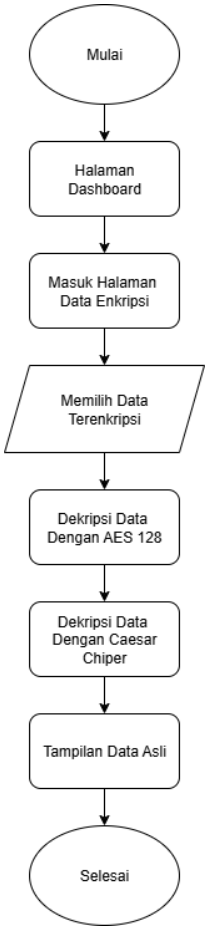
2. Flowchart system

Pada tahap ini, algoritma AES-128 dan Caesar Cipher dipilih untuk meningkatkan keamanan data penduduk yang akan diimplementasikan dalam sistem. Proses enkripsi dimulai dengan *input* data penduduk yang mencakup informasi sensitif seperti nama, alamat, dan NIK, yang kemudian dienkripsi menggunakan Caesar Cipher sebagai lapisan pertama. Setelah itu, data yang telah dienkripsi oleh Caesar Cipher akan kembali dienkripsi menggunakan AES-128 untuk memastikan tingkat keamanan yang lebih tinggi.



Gambar 3. 4 *Flowchart* Enkripsi

Gambar 3.4 menggambarkan alur proses enkripsi data penduduk. Dimulai dengan plaintext yang dimasukkan, Dalam sistem ini, proses mengubah plaintext menjadi ciphertext dilakukan melalui dua tahap enkripsi, yaitu Caesar Cipher sebagai lapisan pertama dan AES-128 sebagai lapisan kedua, untuk menghasilkan ciphertext yang aman, model enkripsi ini memerlukan beberapa komponen utama, yaitu data asli dan kunci enkripsi yang digunakan dalam setiap tahap proses. Lalu hasil enkripsi ini kemudian disimpan dalam database.

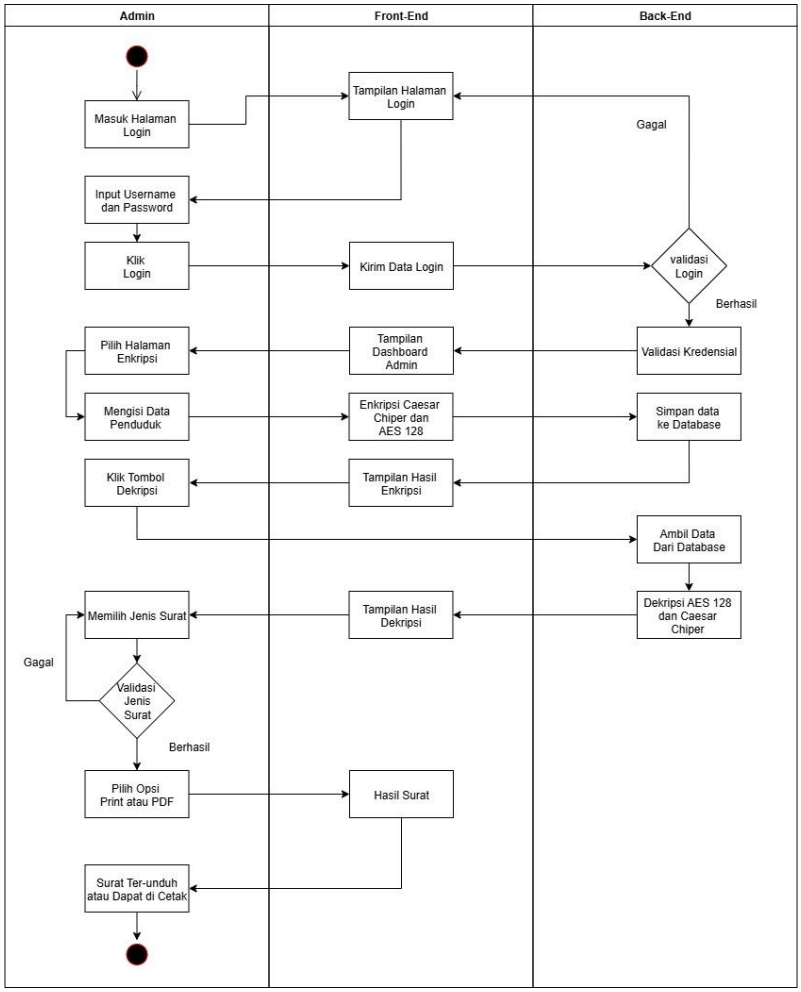


Gambar 3. 5 *Flowchart* Dekripsi

Flowchart ini menggambarkan alur proses dekripsi data dalam sistem perangkat lunak. Dimulai dengan admin akan diarahkan ke Halaman *Dashboard*, di mana mereka dapat mengakses fitur terkait. Dari halaman *Dashboard*, pengguna memilih opsi Masuk Halaman Data Enkripsi, yang membawa mereka ke halaman khusus untuk memilih data yang telah terenkripsi. Setelah memilih data terenkripsi, selanjutnya data akan di dekripsi dengan AES 128, yang merupakan metode enkripsi simetris yang umum digunakan, dan kedua, dengan menggunakan Dekripsi data dengan Caesar Chiper, yang merupakan metode enkripsi klasik berbasis pergeseran huruf. Setelah data berhasil di dekripsi, sistem kemudian akan menampilkan Tampilan Data Asli yang telah dipulihkan. Proses ini diakhiri dengan Selesai, menandakan bahwa alur dekripsi telah berhasil dilakukan.

3. Activity Diagram

Activity diagram atau diagram aktivitas adalah satu alat dalam pemodelan perilaku sistem, digunakan untuk menggambarkan alur kerja dan interaksi dari berbagai aktivitas yang terlibat dalam proses bisnis atau sistem perangkat lunak. Merupakan bagian dari *Unified Modeling Language* (UML), diagram aktivitas menawarkan representasi visual yang membantu dalam memahami dan menyampaikan proses yang kompleks dengan cara yang jelas dan intuitif. Pada dasarnya, diagram ini menggambarkan alur kontrol dari aktivitas-aktivitas yang saling berkaitan, termasuk *input, output,* dan kondisi pengendali yang mungkin terjadi di dalam sistem (Hnatkowska & Cebinka, 2021).

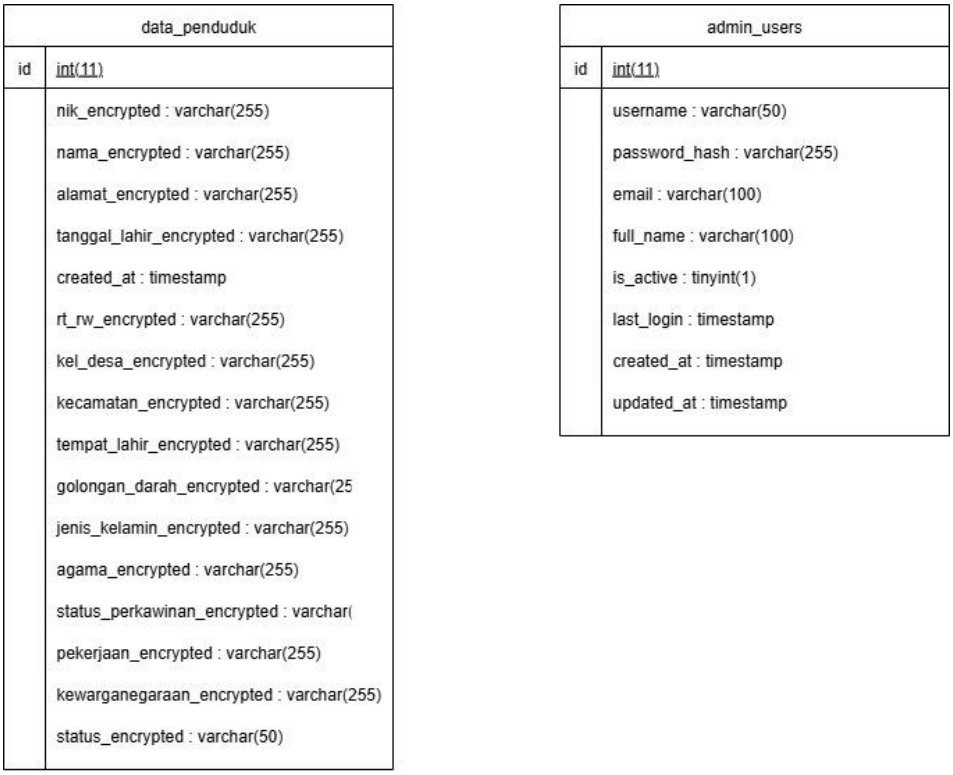


Gambar 3. 6 *Activity* Diagram

Pada Gambar 3.6 ini menggambarkan alur interaksi antara Admin, *Front-End*, dan *Back-End* dalam sistem perangkat lunak. Proses dimulai dengan Admin yang mengakses Halaman *Login*, memasukkan *Username* dan Password, kemudian mengklik *Login*. Setelah itu, data *login* dikirimkan ke *Front-End*, yang akan menampilkan Halaman *Login* dan mengirim data *login* ke *Back-End* untuk diverifikasi. *Back-End* kemudian memvalidasi kredensial pengguna. Jika kredensial valid, maka Admin akan diarahkan ke *Dashboard* Admin, di mana mereka dapat memilih Halaman Enkripsi untuk memasukkan Data Penduduk dan langsung eksekusi enkripsi Caesar chiper dan AES-128 setelah itu hasil data enkripsi di simpan di database lalu di *Front-End* menampilkan hasil enkripsinya. Setelah itu data yang telah ter enkripsi dapat di dekripsi dengan mengklik tombol dekripsi lalu mencocokkan data yang berada di database lalu di dekripsi Caesar chiper dan AES-128. Lalu *Front-End* menampilkan hasil dekripsi. Setelah itu admin dapat memilih jenis surat kemudian memvalidasi jenis surat, jika sudah tervalidasi selanjutnya dapat menentukan apakah surat ingin dicetak atau diekspor ke PDF.

4. Entity Relationship Diagram

Entity Relationship Diagram (ERD) merupakan salah satu alat penting dalam perancangan basis data yang digunakan untuk memodelkan hubungan antar entitas dalam suatu sistem informasi. Konsep utama dari ERD adalah menyajikan representasi yang jelas mengenai struktur data serta interaksi antar entitas yang terlibat, baik dalam konteks aplikasi sederhana maupun sistem yang lebih kompleks. ERD terdiri dari entitas, atribut, dan relasi, yang secara grafis menjelaskan bagaimana data akan terhubung dan berinteraksi (Novita dkk., 2024). Pentingnya ERD dalam pengembangan sistem informasi tidak hanya terletak pada kemampuannya untuk menggambarkan data, tetapi juga dalam efisiensi pengelolaan informasi.



Gambar 3. 7 *Entity Relationship* Diagram

Pada Gambar 3.7, Diagram ERD ini menggambarkan dua entitas utama dalam sistem perangkat lunak, yaitu data penduduk dan admin *user*. Data penduduk menyimpan informasi pribadi penduduk yang dienkripsi, seperti NIK, nama, alamat, tanggal lahir, dan status perkawinan, dengan setiap entri memiliki id dan *created at* untuk mencatat waktu pembuatan data. Sedangkan admin *user* berfungsi untuk mengelola akses sistem, menyimpan data admin *seperti username, password hash, email,* dan *full name*, serta mencatat *last login, created at*, dan *updated at.* Kedua entitas ini tidak terhubung.

3.2.3.3 Penulisan Kode Program

Setelah melakukan rancangan desain sistem, tahapan selanjutnya adalah implementasi melalui penulisan kode program. Dalam penelitian ini, hasil rancangan desain diimplementasikan menjadi produk berupa perangkat lunak berbasis aplikasi web. Untuk pengembangan perangkat lunak, penulis menggunakan Visual Studio Code sebagai ide utama untuk pengembangan aplikasi web yang mengimplementasikan teknologi *frontend* standar (HTML5, CSS3,

JavaScript ES6+) dengan integrasi *library* CryptoJS untuk enkripsi data, jsPDF dan html2canvas untuk pembuatan dokumen PDF, serta teknologi *backend* Node.js dengan Express.js sebagai web server untuk menangani HTTP *requests* dan mengelola *routing* API. Sistem menggunakan MySQL sebagai database *management system* dengan mysql2 *driver*, mengimplementasikan autentikasi berbasis *bcrypt* untuk keamanan *password*, serta komunikasi *client*-server melalui protokol HTTP/HTTPS dengan format JSON. Proses perancangan ini mendukung pembangunan aplikasi pengelolaan data penduduk dengan keamanan kriptografi ganda (AES-128 dan Caesar Cipher) dan fitur pembuatan surat otomatis, yang disajikan dalam bentuk diagram UML (*Use Case, Flowchart, Activity* dan ERD) untuk memberikan gambaran menyeluruh mengenai arsitektur, alur kerja, dan struktur data dalam sistem SPARTA.

3.2.3.4 Pengujian

Tahap keempat pada model waterfall adalah proses pengujian. Pada penelitian ini, pengujian dilakukan dengan menggunakan metode *Black Box Testing, Wireshark, dan Korelasi Pearson*. Setelah tahap pengkodean selesai, *Black Box Testing* digunakan untuk memastikan bahwa sistem mampu menerima input, memproses data, serta menghasilkan output sesuai dengan yang diharapkan. Sementara itu, pengujian menggunakan *Wireshark* bertujuan untuk memverifikasi bahwa data penduduk selama proses transmisi tidak terekspos dalam bentuk asli (*plaintext*). Pengujian korelasi Pearson pada penelitian ini diterapkan untuk mengevaluasi efektivitas gabungan algoritma Caesar Cipher dan AES-128 dalam melindungi data penduduk. Pengujian dilakukan dengan mengukur tingkat hubungan *linear* antara *plaintext* dan *ciphertext* yang dihasilkan dari proses enkripsi. Data uji berupa NIK 16 digit dikonversi terlebih dahulu ke nilai numerik berdasarkan kode ASCII setiap digit, kemudian dihitung koefisien korelasinya menggunakan rumus Pearson. Nilai korelasi yang mendekati nol menunjukkan bahwa hasil enkripsi bersifat acak secara statistik, sedangkan nilai yang mendekati +1 atau −1 menandakan adanya pola *linear* yang masih terbawa dari *plaintext* ke *ciphertext.*

3.2.3.5 Penerapan dan Pemeliharaan

Pada tahap terakhir metode waterfall yaitu penerapan dan pemeliharaan, sistem yang telah dikembangkan diuji secara menyeluruh untuk memastikan seluruh fitur bekerja sesuai dengan rancangan dan memenuhi kebutuhan pengguna. Proses ini dilakukan dalam lingkungan simulasi atau pengujian akademis. Setelah penerapan selesai, pemeliharaan dilakukan melalui evaluasi berkelanjutan, termasuk perbaikan bug dan pengoptimalan fitur berdasarkan hasil pengujian, guna memastikan sistem tetap relevan dan stabil sebagai bagian dari proyek akademik.

3.2.4 Uji Coba Produk (*Test the artifact*)

Tahap keempat dalam penelitian ini adalah tahap pengujian. Proses pengujian dilakukan dengan menggunakan metode *Black Box Testing, Wireshark,* dan *Korelasi Pearson. Black Box Testing* merupakan metode pengujian perangkat lunak yang berfokus pada pengujian fungsi aplikasi tanpa memperhatikan detail internal maupun struktur kode. Pada pendekatan ini, pengujian dilakukan melalui interaksi dengan antarmuka sistem, tanpa adanya akses langsung terhadap kode sumber ataupun mekanisme internal yang digunakan. Sedangkan untuk pengujian *Wireshark* dalam konteks pengujian jaringan, *Wireshark* berfungsi sebagai alat diagnostik yang memungkinkan administrator jaringan dan peneliti untuk memperoleh wawasan mendalam tentang lalu lintas jaringan, deteksi masalah, dan analisis keamanan. Korelasi Pearson diterapkan untuk mengevaluasi tingkat hubungan antara data plaintext dan ciphertext yang dihasilkan oleh algoritma enkripsi (seperti AES-128 dan Caesar Cipher). Dengan menggunakan Korelasi Pearson, peneliti dapat mengukur sejauh mana pola asli data berhubungan dengan data terenkripsi. Nilai korelasi yang rendah atau mendekati nol menunjukkan bahwa data terenkripsi berhasil menyamarkan informasi asli, sehingga algoritma enkripsi memiliki tingkat keamanan yang baik.

Selanjutnya, pengujian menggunakan metode Black Box Testing, mencakup fitur-fitur utama sistem yang mendasari semua fungsi aplikasi web. Tabel 3.3 menggambarkan setiap rangkaian kasus uji yang dirancang untuk mengevaluasi respons sistem terhadap *input* tertentu dan membandingkannya dengan *output* yang

diharapkan. Perancangan pengujian ini menjadi landasan untuk validasi fungsi yang lebih mendalam pada Bab IV.

Tabel 3. 3 Rancangan Pengujian Komponen Utama Sistem

No	Komponen	Fitur Pengujian	Skenario Pengujian	Hasil Yang Di Harapkan
1	Sistem Autentikasi Dan Sesi	Login Admin (Benar)	Admin Mengisi Username Dan Password Valid Lalu Klik Login	Sistem Mengautentikasi, Menampilkan Halaman Utama, Dan Menyimpan Sesi Login
2		Login Admin (Password Salah)	Admin Mengisi Username Benar, Password Salah	Sistem Menolak Login Dan Menampilkan Pesan "Username Atau Password Salah"
3		Login Admin (Form Kosong)	Admin Menekan Login Dengan Username Atau Password Kosong	Sistem Menolak Login Dan Menampilkan Pesan "Username Dan Password Harus Diisi!"
4		Logout Admin	Admin Menekan Tombol Logout	Sistem Menghapus Sesi, Kembali Ke Halaman Login
5	Navigasi UI	Navigasi Beranda	Admin Klik Menu Beranda	Sistem Menampilkan Konten Beranda

No	Komponen	Fitur Pengujian	Skenario Pengujian	Hasil Yang Di Harapkan
6		Navigasi Enkripsi	Admin Klik Menu Enkripsi	Sistem Menampilkan Form Input Data Penduduk
7		Navigasi Dekripsi	Admin Klik Menu Dekripsi	Sistem Menampilkan Tabel Data Hasil Dekripsi
8		Navigasi Data Enkripsi	Admin Klik Menu Data Enkripsi	Sistem Menampilkan Tabel Data Terenkripsi Beserta Status
9	Proses Enkripsi Data	Simpan Data Valid	Admin Mengisi SeluruhField Form Dengan Data Valid Lalu Submit	Sistem Mengenkripsi Data (Caesar + Aes-128), Menyimpan Ke Database, Dan Menampilkan Hasil
10		Gagal Simpan (Data Tidak Lengkap)	Admin Mengosongkan Minimal Satu Field Wajib Lalu Submit	Sistem Menolak Penyimpanan Dan Menampilkan Pesan “Data Tidak Lengkap”
11		Validasi Nik	Admin Mengisi Nik Kurang Dari 16 Digit Karakter Tidak Valid	Sistem Menolak Dan Meminta Perbaikan Format Nik
12	Manajemen Data	Ambil Semua Data	Admin Membuka Halaman Data Enkripsi	Sistem Mengambil Dan Menampilkan Data Dari Database Urut Terbaru

No	Komponen	Fitur Pengujian	Skenario Pengujian	Hasil Yang Di Harapkan
13		Filter Status	Admin Memilih Filter "Belum Didekripsi" Atau "Sudah Didekripsi"	Sistem Menampilkan Baris Sesuai Status Terpilih
14		Pencarian Nama	Admin Memasukkan Kata Kunci Nama Pada Pencarian	Sistem Menampilkan Baris Dengan Nama Yang Mengandung Kata Kunci
15		Pencarian Nik	Admin Memasukkan Nik Pada Pencarian	Sistem Menampilkan Baris Dengan Nik Yang Cocok
16		Preview Data	Admin Menekan Tombol Detail	Sistem Menampilkan Seluruh Data Penduduk Secara Lengkap
17	Proses Dekripsi Data	Dekripsi Data	Admin Menekan Tombol Dekripsi Pada Data Yang Siap	Sistem Mendekripsi (Aes → Caesar Chiper) Dan Menampilkan Nilai Asli
18	Aksi Data	Hapus Data	Admin Menekan Tombol Hapus Pada Salah Satu Baris	Sistem Meminta Konfirmasi, Menghapus Data Dari Database, Dan Menyegarkan Tabel

No	Komponen	Fitur Pengujian	Skenario Pengujian	Hasil Yang Di Harapkan
19	Sistem Surat	Generate Surat Domisili	Admin Memilih Jenis Surat (Domisili, Tidak Mampu/ Usaha/Kelakuan Baik/Belum Menikah)	Sistem Menampilkan Surat Sesuai Template Dengan Data Terisi Benar
20		Download Pdf Surat	Admin Menekan Tombol Download Pdf	Sistem Menghasilkan File Pdf Yang Berisi Tampilan Surat
21		Print Surat	Admin Menekan Tombol Print	Sistem Menampilkan Dialog Cetak Dengan Layout Surat

3.2.5 Evaluasi Hasil Tes (*Evaluate testing result*)

Setelah proses pengujian dan, langkah berikutnya adalah evaluasi hasil tes. Pada tahap ini hasil dari uji coba akan dianalisis untuk menilai apakah kedua metode enkripsi tersebut efektif dalam melindungi data penduduk. Evaluasi juga meliputi pengujian performa aplikasi menggunakan *Black Box Testing*, Pengujian *Wireshark* dan Pengujian korelasi *Pearson*

Black Box Testing akan dilakukan untuk memastikan bahwa aplikasi berfungsi sesuai dengan kebutuhan dan spesifikasi yang telah ditentukan. Semua pengujian akan dilakukan berdasarkan fungsionalitas aplikasi dan bagaimana aplikasi merespons input dari pengguna tanpa memperhatikan struktur internal kode sumbernya.

Wireshark akan digunakan untuk memonitor dan menganalisis data yang dikirimkan antara *client* dan server, guna memastikan bahwa data yang dienkripsi

benar-benar terlindungi selama proses transmisi. Pengujian ini bertujuan untuk mengidentifikasi potensi kebocoran data yang bisa terjadi selama komunikasi antara klien dan server.

Pengujian korelasi Pearson dilakukan untuk mengukur tingkat hubungan linear antara data asli (*plaintext*) dan data hasil enkripsi (*ciphertext*) menggunakan algoritma AES-128. Tujuan pengujian ini adalah untuk memastikan bahwa *ciphertext* yang dihasilkan tidak memiliki keterkaitan statistik yang signifikan dengan *plaintext*, sehingga data hasil enkripsi bersifat acak dan tidak dapat diprediksi. Nilai koefisien korelasi r yang mendekati nol menunjukkan bahwa proses enkripsi telah berhasil menghilangkan pola linear dari data asli, sesuai dengan prinsip *confusion* dalam teori kriptografi.

3.2.6 Mengkomunikasikan Hasil Uji Coba (Communicate the testing result)

Pada tahap ini hasil penelitian dan uji coba disajikan dalam bentuk laporan. Hasil uji coba, baik dari segi keamanan maupun performa aplikasi, disajikan secara terstruktur dan dianalisis untuk memberikan rekomendasi terkait penerapan kedua metode enkripsi dalam aplikasi yang lebih luas. Dengan mengikuti langkah-langkah ini, penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan aplikasi website yang lebih aman dan efisien dalam melindungi data pribadi pengguna, serta memberikan wawasan tentang kelebihan dan kekurangan dari kedua metode enkripsi yang diterapkan.