APLIKASI KRIPTOGRAFI BERBASIS ANDROID MENGGUNAKAN ALGORITMA RAIL FENCE CIPHER

SKRIPSI

Oleh :

RADEN DIKY ZAILANI
NPM. 16010175

PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DEHASEN
BENGKULU
2023

APLIKASI KRIPTOGRAFI BERBASIS ANDROID MENGGUNAKAN ALGORITMA RAIL FENCE CIPHER

SKRIPSI

RADEN DIKY ZAILANI
NPM. 16010175

Diajukan Untuk Memperoleh Gelar Sarjana Komputer
Pada Program Studi Informatika

PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DEHASEN
BENGKULU
2023

APLIKASI KRIPTOGRAFI BERBASIS ANDROID MENGGUNAKAN ALGORITMA RAIL FENCE CIPHER

SKRIPSI

Oleh :

RADEN DIKY ZAILANI
NPM. 16010175

DISETUJUI OLEH :

Pembimbing Utama	Pembimbing Pendamping
Khairil, S.Kom., M.Kom	Abdussalam Al Akbar, S.Kom., M.Kom
NIDN. 02.130475.01	NIDN. 02.051092.01

Mengetahui,
Ketua Program Studi Informatika

Liza Yulianti, S.Kom., M.Kom
NIDN. 02.160772.01

iii

APLIKASI KRIPTOGRAFI BERBASIS ANDROID MENGGUNAKAN ALGORITMA RAIL FENCE CIPHER

SKRIPSI

Disusun Oleh :

RADEN DIKY ZAILANI
NPM. 16010175

Telah Dipertahankan di depan TIM Penguji
Universitas Dehasen Bengkulu

Hari : Sabtu
Tanggal : 17 Juni 2023
Ujian Gedung Universitas Dehasen Bengkulu

Skripsi ini telah diperiksa dan disetujui oleh TIM Penguji.

Penguji	Nama	NIDN	Tanda Tangan
Ketua	Khairil, S.Kom., M.Kom	02.130475.01	
Anggota	Abdussalam Al Akbar, S.Kom., M.Kom	02.051092.01	
Anggota	Juju Jumadi, S.Kom., M.Kom	02.111282.01	
Anggota	Deri Lianda, S.Kom., M.Kom	02.250489.04	

Mengetahui,
Dekan Fakultas Ilmu Komputer

Siswanto, SE, S.Kom, M.Kom
NIDN. 02.240363.01

iv

RIWAYAT HIDUP

Raden Diky Zailani adalah nama penulis ini. lahir pada 09 Desember 1997 di Kota Bengkulu Provinsi Bengkulu, penulis merupakan anak ke dua dari dua bersaudara, dari pasangan Bahder Johan dan Kurniah, penulis pertama kali masuk pendidikan di SD Negri 1 Kota Bengkulu pada tahun 2003 dan tamat 2009 pada tahun yang sama melanjutkan pendidikan SMP Negri 3 Kota Bengkulu dan tamat pada tahun 2012. Setelah tamat di SMP penulis melanjut kan ke SMA Muhammadiyah 4 Kota Bengkulu dan tamat tahun 2015. Kemudian penulis melanjutkan pendidikan ke Perguruan Tinggi yaitu pada Universitas Dehasen (UNIVED) Bengkulu dengan mengambil Program Studi Informatika pada Fakultas Ilmu Komputer, untuk jenjang Strata Satu (S1)

MOTTO

“ “Hidup adalah perjuangan dan harus diperjuangkan. Sempurnakan usaha dengan doa, kemudian bersabar menunggu hasil yang sempurna.” (Anonim)

“Berharaplah yang terbaik, dan usahakanlah yang terbaik. Harapan tanpa usaha, biasanya adalah perjalanan yang lama dan tak kunjung sampai.”(Anonim)

PERSEMBAHAN

Dengan mengucapkan syukur Alhamdulillah kepada Allah SWT, yang telah memberikan kesehatan, rahmat dan hidayah, sehingga penulis masih diberikan kesempatan untuk menyelesaikan skripsi ini, sebagai syarat untuk memperoleh gelar kesarjanaan pada program studi informatika, di Universitas Dehasen Bengkulu. Walaupun jauh dari kata sempurna, namun penulis bangga telah mencapai pada titik ini, dengan penuh rasa bangga dan bahagia skripsi ini ku persembahkan kepada :

1. Prof. Dr, Husaini, SE., M.Si,.AK,CA,CRP selaku rektor universitas dehasen Bengkulu yang telah memberikan penulis kesempatan untuk menempuh pendidikan di peguruan tinggi.
2. Kedua orang tua yang aku sayangi, Bapakku Bahder Johan dan Ibuku tercinta Kurniah yang tiada henti memberikan motivasi serta merawatku, membersarkanku, mendidikku sejak kecil hingga dewasa serta mendukung dan mendoakan setiap langkah usaha dan perjuangan ku hingga hari ini, esok dan seterusnya nya
3. Kakakku Yudha Kresna Sanjaya S.E yang selalu memberikan dorongan, uang jajan, nasehat, serta semangat kepadaku agar dapat menyelesaikan skripsi ini
4. Buat keluarga besar di Bengkulu Terimakasih telah mendukung saya
5. Terimakasih untuk Bapak Khairil S.kom,M.kom pembimbing 1 dan Bapak Abdussalam Al akbar S.kom,M.kom pembimbing 2 yang telah memberikan kontribusi burupa bimbingan, motivasi,saran atas terselesaikannya skripsi ini
6. Terimakasih untuk Bapak Juju Jumadi S.Kom., M.Kom sebagai penguji 1 dan Bapak Deri Lianda S.Kom., M.Kom sebagai penguji 2 yang telah banyak memberikan kritik dan saran yang sangat berharga demi terselesaikannya skripsi ini
7. Bapak Siswanto SE., S.Kom., M.Kom selaku dekan fakultas ilmu komputer
8. Ibu liza Yulianti S.Kom., M.Kom selaku ketua program studi informatika
9. Teman-teman seperjuangan ku, (Nopri, Michel,) yang selalu memberikan saran, semangat, dan motivasi untuk menyelesaikan skripsi ini

10. Sahabat dan sepupuku (Apri, Mas rudi, Mas Jeri) yang selalu memberikan kopi, dorongan nasehat dan semangat untuk mencapai cita-cita ku
11. Penyemagat kedua orang tua
12. Almamater kuning yang aku bangga kan

ABSTRAK

APLIKASI KRIPTOGRAFI BERBASIS ANDROID MENGGUNAKAN ALGORITMA RAIL FENCE CIPHER

Oleh :

Raden Diky Zailani [1)]
Khairil, S.Kom., M.Kom[2)]
Abdussalam Al-Akbar, S.Kom., M.Kom[2)]

Aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher* dibuat menggunakan bahasa pemrograman *Java* (*Android Studio*), bahasa pemrograman *PHP* (*File web Service JSON*) dan *database MySQL*. Aplikasi ini terdapat *link url* untuk menyimpan file PHP dan data pesan di *database*, adapun *link url* tersebut http://railfencecipher.my.id/. Aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher* telah di upload ke google playstore dengan memasukkan kata kunci rail fence cipher

Aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher* ini dapat digunakan untuk membantu komunikasi antara kedua belah pihak baik pengirim dan penerima pesan. Selain itu pada aplikasi ini telah diterapkan algoritma *rail fence cipher* yang digunakan untuk menjaga kerahasiaan pesan tersebut yang disimpan di dalam database dari aplikasi. Record pada kolom isi pesan di database dalam bentuk teracak, sehingga tidak dapat dibaca. Hal ini tentunya membuat aplikasi tersebut terjaga kerahasiaan pesan yang dikirim, karena terjadi proses pengacakan (enkripsi) pada pesan tersebut.

Berdasarkan hasil pengujian yang telah dilakukan, fungsional dari aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher* berjalan dengan baik sesuai yang diharapkan dan pesan yang dikirim antara penerima dan pengirim terjaga kerahasiaan keamanan pesan tersebut karena sudah teracak

Kata Kunci : *Kriptografi, Android, Algoritma Rail Fence Cipher*

1) Calon Sarjana
2) Dosen Pembimbing

ABSTRACT

ANDROID-BASED CRYPTOGRAPHY APPLICATIONS USING THE RAIL FENCE CIPHER ALGORITHM

By:

Raden Diky Zailani [1)]
Khairil [2)]
Abdussalam Al-Akbar [2)]

Android-based cryptographic applications using the rail fence cipher algorithm are made using the Java programming language (AndroidStudio), the PHP programming language (JSON File Web Service) and the MySQL database. This application has a url link for storing PHP files and message data in the database, while the url link is http://railfencecipher.my.id/. An Android-based cryptographic application using the rail fence cipher algorithm has been uploaded to Google Playstore by entering the rail keyword fence cipher. An android-based cryptographic application using the rail fence cipher algorithm can be used to assist communication between the two parties, both the sender and the recipient of the message. In addition, this application has implemented a rail fence cipher algorithm that is used to maintain the confidentiality of the message stored in the application's database. The records in the message body column in the database are in random form, therefore they cannot be read. This of course makes the application maintain the confidentiality of the messages sent, because there is a process of scrambling (encryption) of the message. Based on the results of the tests that have been carried out, the functionality of the android-based cryptographic application using the rail fence cipher algorithm runs well as expected and the message sent between the recipient and the sender is kept confidential because the message is encrypted.

Keywords: Cryptography, Android, Rail Fence Cipher Algorithm

1) Student
2) Supervisors

KATA PENGANTAR

Puji syukur kepada Allah SWT berkat Rahmat, Hidayah, dan Karunia-Nya kepada kita semua sehingga penulis dapat menyelesaikan skripsi ini dengan judul **"Aplikasi Kriptografi Berbasis Android Menggunakan Algoritma *Rail Fence Cipher*"**. Shalawat serta salam juga penulis panjatkan kepada junjungan Nabi Besar Muhammad SAW.

Adapun maksud dan tujuan diajukannya skripsi ini adalah untuk memperoleh Gelar Sarjana Komputer Pada Program Studi Informatika Fakultas Ilmu Komputer Universitas Dehasen Bengkulu.

Skripsi ini tidak akan selesai tanpa bantuan dari berbagai pihak. Untuk itu, dalam kesempatan ini penulis mengucapkan terima kasih banyak kepada berbagai pihak yang telah membantu penulis, diantaranya :

1. Bapak Siswanto, SE, S.Kom., M.Kom selaku Dekan Fakultas Ilmu Komputer Universitas Dehasen Bengkulu.
2. Ibu Liza Yulianti, S.Kom., M.Kom selaku Ketua Program Studi Informatika Universitas Fakultas Ilmu Komputer Dehasen Bengkulu.
3. Bapak Khairil, S.Kom., M.Kom selaku Dosen Pembimbing I yang telah memberikan masukan serta arahan yang membangun dalam pembuatan skripsi ini
4. Bapak Abdussalam Al-Akbar, S.Kom., M.Kom selaku Dosen Pembimbing II yang telah memberikan masukan serta arahan yang membangun dalam pembuatan skripsi ini.

5. Segenap Civitas Akademik Pada Program Studi Informatika Fakultas Ilmu Komputer Universitas Dehasen Bengkulu
6. Berbagai pihak yang telah banyak membantu dalam penyusunan skripsi ini.

Diharapkan, skripsi ini bisa bermanfaat untuk semua pihak. Selain itu, kritik dan saran yang membangun sangat penulis harapkan dari pembaca sekalian agar skripsi ini bisa lebih baik lagi

Bengkulu, Juni 2023

Penulis

DAFTAR ISI

Halaman

DAFTAR TABEL

Tabel Halaman

DAFTAR GAMBAR

Gambar Halaman

DAFTAR LAMPIRAN

1. Time Schedule
2. Kartu Bimbingan Proposal Skripsi

BAB I

PENDAHULUAN

1.1. Latar Belakang

Komunikasi merupakan salah satu kegiatan dasar dalam kehidupan manusia yang memungkinkan manusia saling dapat bertukar informasi. Pertukaran informasi jarak jauh ini menuntut keamanan terhadap kerahasiaan informasi yang dipertukarkan. Suatu informasi akan memiliki nilai tinggi apabila menyangkut tentang aspek keamanan, kepentingan umum dan kepentingan pribadi, sehingga informasi tersebut akan banyak diminati oleh pihak lain yang tidak berwenang untuk mendapatkan isi dari informasi tersebut. Salah satu cara dalam mengamankan informasi tersebut yaitu dengan melakukan keamanan terhadap informasi melalui kriptografi.

Kriptografi adalah bidang ilmu yang mempelajari tentang cara untuk menjaga keamanan pesan dalam proses pengiriman dengan menggunakan metode penyandian tertentu, dengan tujuan agar informasi dalam pesan tersebut tidak disalahgunakan oleh orang yang bukan penerima aslinya. Kriptografi memiliki beragam metode untuk menyandikan pesan atau informasi yang ingin kita sembunyikan, seperti *Caesar Cipher*, *Affine*, *Monoalphabetic*, *Polyalphabetic*, *Vigenere*, *Rail Fence Cipher*, Transposisi, dan banyak lagi metode-metode dalam kriptografi ini.

Dalam penelitian ini dilakukan penerapan terhadap algoritma kriptografi ke bentuk aplikasi *chatting*. Aplikasi *chatting* yang berfungsi sebagai media berkomunikasi juga digunakan dalam kegiatan komunikasi

untuk saling bertukar informasi. Pengiriman pesan melalui internet menggunakan aplikasi *chatting* bersifat *realtime* yang dilakukan dengan transaksi paket antara *client* dengan *server*. Salah satu algoritma kriptografi yang diterapkan ke dalam aplikasi *chating* untuk menjaga kerahasiaan informasi *chat* antara pengirim dan penerima yaitu algoritma *rail fence cipher*. Algoritma *Rail Fence Cipher* merupakan salah satu algoritma *cipher* transposisi yang mengacak urutan huruf-huruf pesan. Algoritma ini melibatkan penulisan plainteks ke bawah secara berturut turut yang memiliki baris atas dan baris bawah. Sedangkan *ciphertext* nya diperoleh dengan membaca huruf berdasarkan baris.

Dari uraian tersebut di atas, maka penulis tertarik untuk mengangkat judul "Aplikasi Kriptografi Berbasis Android Menggunakan Algoritma *Rail Fence Cipher*".

1.2. Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini yaitu bagaimana membuat aplikasi chatting berbasis android menggunakan Algoritma *Rail Fence Cipher* ?

1.3. Batasan Masalah

Agar tidak melebar dari pembahasan, maka dibatasi masalah dalam penelitian ini antara lain :

a. Aplikasi *chatting* dibuat menggunakan platform android.
b. Aplikasi *chatting* dibuat menggunakan Android Studio
c. Aplikasi *chatting* menggunakan akses koneksi internet

d. Aplikasi *chatting* dapat di install melalui *google play store*

1.4. Tujuan Penelitian

Adapun tujuan dari penelitian ini dilakukan, yaitu

a. Untuk memenuhi persyaratan dalam menyusun Skripsi pada Program Studi Informatika Fakultas Ilmu Komputer Universitas Dehasen Bengkulu
b. Untuk menjaga kerahasiaan dan keamanan informasi menggunakan algoritma *rail fence cipher*.
c. Untuk membuat aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher*.

1.5. Manfaat Penelitian

Adapun manfaat dari penelitian ini dilakukan, antara lain :

a. Dapat menjaga kerahasiaan dan keamanan informasi sehingga informasi tersebut tidak dapat dibaca oleh orang yang tidak berwenang.
b. Dapat dijadikan sebagai bahan referensi dalam membuat aplikasi kriptografi berbasis android.

BAB II

LANDASAN TEORI

2.1. Aplikasi *Chatting*

Aplikasi *chatting* adalah aplikasi yang digunakan untuk berkomunikasi dalam satu jaringan lokal atau internet yang saling terhubung satu sama lain untuk mempermudah percakapan. Aplikasi *chatting* yang saat ini banyak berkembang pesat antara lain *WhatsApp*, *Line*, dan *Telegram* aplikasi tersebut memiliki keunggulan dan fitur masing-masing, karena pada dasarnya aplikasi tersebut digunakan oleh umum dan diakses dari seluruh penjuru dunia (Prabowo & Pramusinto, 2018).

Chatting adalah menghubungkan dua orang atau lebih tapi terhubung melalui internet. memungkinkan untuk berkomunikasi secara langsung di tempat yang berbeda secara *realtime* yang berupa pesan teks. Aplikasi *Chatting* adalah salah satu media komunikasi yang sering digunakan untuk menyampaikan pesan. Pada kepentingan atau tujuan tertentu seseorang ingin mengirim pesan yang isi pesannya tidak ingin diketahui oleh orang lain selain si penerima pesan yang dituju karena isi pesan tersebut bersifat sangat rahasia atau pribadi (Alfajar & Akbar, 2021).

2.2. Kriptografi

Kriptografi (*cryptography*) berasal dari bahasa Yunani : "*cryptos*" artinya "*secret*"(rahasia), sedangkan "*graphein*" aritnya "*writing*" (tulisan).

Jadi, kriptografi berarti "*secret writing*" (tulisan rahasia) (Rosna, et al., 2019).

Kriptografi adalah metode ataupun suatu teknik untuk mengubah teks awal (asli) menjadi suatu teks tersandi atau tidak terbaca dan menjadi aman dengan cara mengubah teks tersebut menggunakan kunci tertentu. Teks asli yang masih bisa dibaca semua orang itu disebut sebagai plaintext, dan teks yang sudah dijadikan kode dan tidak dapat dibaca semua orang dan tidak mempunyai makna ini disebut sebagai *ciphertext*. Proses pengubahan *plaintext* menjadi *ciphertext* disebut sebagai enkripsi, sedangkan proses pengembaliannya disebut sebagai proses dekripsi. Proses penyandian ini digunakan dengan tujuan untuk mengamankan informasi yang ada di teks dari pihak-pihak yang tidak bertanggung jawab (Purba & Puspasari, 2020).

Kriptografi merupakan ilmu mengenai metode untuk mengirimkan pesan secara rahasia sehingga hanya penerima yang dimaksud yang dapat menghapus dan membaca pesan tersebut atau memahaminya. Kriptografi dapat dibagi menjadi dua kategori yaitu kriptografi klasik dan kriptografi modern. Kriptografi klasik merupakan kriptografi yang sudah dikembangkan bahkan sejak belum ada komputer. Beberapa metode kriptografi klasik adalah *substitution cipher* dan *transposition cipher* (Utomo, et al., 2020).

Kriptografi berasal dari bahasa Yunani, *crypto* dan *graphia*. *Crypto* berarti *secret* (rahasia) dan *graphia* berarti *writing* (tulisan). Kriptografi menurut terminologinya adalah sebuah ilmu dan seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ke tempat lain.

Secara istilah kriptografi didefinisikan sebagai ilmu dan seni untuk menjaga kerahasiaan pesan baik berupa data maupun informasi yang mempunyai arti atau nilai dengan cara menyamarkan (mengacak) menjadi bentuk yang tidak dapat dimengerti dan hanya penerima yang dapat mengubah kode-kode tersebut menjadi pesan asli yang dapat dimengerti (Jamaludin & Romindo, 2020).

Setiap algoritma kriptografi terdiri algoritma enkripsi (E) dan algoritma dekripsi (D). Dasar matematis yang mendasari proses enkripsi dan dekripsi adalah relasi antara dua himpunan yaitu himpunan yang berisi elemen *plaintext* dan himpunan yang berisi elemen *ciphertext*. Enkripsi dan dekripsi merupakan fungsi transformasi antara dua himpunan tersebut. Secara umum dapat digambarkan secara matematis sebagai berikut (Ardi, 2020) :

$$E_k(P) = C(Proses\ Enkripsi)$$

$$D_k(C) = P(Proses\ Dekripsi)$$

$$D_k(E(P)) = P(Proses\ Dekripsi)$$

Definisi sederhana dari kriptografi adalah teknik untuk menjaga kerahasiaan pesan dengan cara menyandikannya sehingga tidak dapat dimengerti lagi maknanya. Kriptografi memiliki algoritma dalam melakukan proses penyandian suatu agar dapat terjaga keasliannya. Algoritma kriptografi terdiri dari tiga fungsi dasar (Yanti, et al., 2018) :

a. Enkripsi

Enkripsi merupakan istilah lain dari proses menyandikan data penting ke dalam bentuk simbol-simbol yang tidak dapat dimengerti lagi oleh pihak lain sehinggan keaslian dan keamanan data dapat terjaga.

b. Dekripsi

Dekripsi adalah proses untuk merubah atau mengembalikan data tersandi kebentuk aslinya agar arti data dapat dimengerti oleh penerima.

c. Kunci

Kunci merupakan elemen yang paling penting dalam mengimplementasikan proses enkripsi dan dekripsi. Keamanan kunci di dalam kriptografi menjadi prioritas karena serumit apapun algoritma yang digunakan akan dapat dipecahkan bila kunci yang digunakan berhasil ditemukan. Kunci terbagi menjadi dua bagian, kunci rahasia (private key) dan kunci umum (public key).

Dari paparan awal dapat dirangkumkan bahwa kriptografi bertujuan untuk member layanan keamanan. Yang dinamakan aspek-aspek keamanan :

1. Kerahasiaan (*confidentiality*) adalah layanan yang ditujukan untuk menjaga agar pesan tidak dapat dibaca oleh pihak-pihak yang tidak berhak.
2. Integritas data (*data integrity*) adalah layanan yang menjamin bahwa pesan masih asli atau belum pernah dimanipulasi selama pengiriman.
3. Otentikasi (*authentication*) adalah layanan yang berhubungan dengan identifikasi, baik mengidentifikasi kebenaran pihak-pihak yang berkomunikasi (*user autehentication*).

4. *Non-repudiation* adalah layanan untuk menjaga entitas yang berkomunikasi melakukan penyangkalan.

2.3. Algoritma *Rail Fence Cipher*

Algoritma *Rail Fence Cipher* merupakan salah satu algoritma *cipher* transposisi yang mengacak urutan huruf-huruf pesan. Algoritma ini melibatkan penulisan plainteks ke bawah secara berturut turut yang memiliki baris atas dan baris bawah. Sedangkan *ciphertext* nya diperoleh dengan membaca huruf berdasarkan baris. Algoritma *Rail Fence Cipher* menyusun plainteks secara *ziq-zag* dengan turun ke bawah dan naik ke atas sesuai ukuran kolom dan baris yang ditentukan oleh *key* (Purnamasari, 2021).

Algoritma *Railfence Chiper* merupakan salah satu variasi implementasi cipher transposisi. Pada *Railfence Chiper, plainteks* dituliskan secara vertikal ke bawah sepanjang *n-rails,* dan menulis lagi ke kolom baru ketika telah mencapai karakter ke-*n*. *Cipherteks* yang dihasilkan adalah urutan karakter yang dibaca secara horizontal. Sebagai contoh, kita mempunyai n=3 dan sebuah pesan (Ratna, 2018).

Rail Fence Cipher merupakan salah satu algoritma cipher transposisi yang mengacak urutan huruf-huruf pesan. Algoritma ini melibatkan penulisan plainteks ke bawah secara berturut-turut yang memiliki baris atas dan baris bawah, sedangkan Cipherteksnya diperoleh dengan membaca huruf berdasarkan baris. Algoritma *Rail Fence Cipher* menyusun plainteks secara ‘zig-zag’, dengan turun ke bawah dan naik keatas sesuai ukuran kolom dan baris yang ditentukan oleh key. Cipherteks

diperoleh dengan membaca susunan huruf secara horizontal. *Rail Fence Cipher* pernah digunakan selama Perang Saudara Amerika, ketika digunakan untuk penyembunyian pesan militer Union maupun mata-mata Konfederasi (Girsang, et al., 2019).

Metode enkripsi *Rail Fence* adalah salah satu bentuk cipher transposisi yang sederhana yang diinspirasi dari model *Polybius square*. *Polybius square* adalah menyusun huruf sebagai matriks 5x5 dan mengkodekan huruf A sebagai 1-1, huruf B sebagai 1-2 dan seterusnya. Setiap karakter pada *Polybius square* diganti dengan indeks *cell* matriks tanpa menggunakan kunci khusus dan hanya mengubah posisi sehingga teks tidak terbaca. Berbeda dengan *Polybius square*, metode *Rail Fence* menyusun teks secara zig-zag yang model matriksnya diketahui oleh pengirim dan penerima pesan (Latifah, et al., 2017).

Cipher transposisi adalah metode penyusunan kembali karakter dengan menyesuaikan beberapa skema yang sering kali digunakan pada penggambaran beberapa geometri. *Ciphertext* diperoleh dengan pengubahan posisi. Dengan kata lain, algoritma ini mentransposisi rangkaian karakter di dalam *text*. Nama lain dari metode ini adalah permutasi, karena transposisi semua karakter di dalam teks sama dengan mempermutasikan karakter-karakter tersebut dengan menggunakan kunci penggambaran tambahan dan metode untuk menuliskan serta meletakkannya ke dalam urutan tertentu dengan cara zig-zag per baris

Enkripsi pada metode zig-zag dilakukan dengan langkah-langkah berikut ini :

a. Letakkan setiap urutan karakter ke *plaintext* ke dalam matriks per kolom secara zig-zag sebanyak baris yang disesuaikan dengan besarnya nilai kunci.
b. *Cipher* didapatkan dengan membaca karakter per baris dari matriks yang terbentuk atau dengan formula berikut :

   Baris = 1 $E_x(X)$ diambil urutan *plaintext* pada posisi ke-1 yang dilanjutkan dengan mengambil posisi *plaintext* dengan jeda c dan jeda *(K*2-d)* secara bergantian selama nilai jeda ≤ ukuran *plaintext* dan diketahui *K* = banyak baris sebagai kunci, nilai *d* = 2 dan nilai *c* = 0.

   Baris = 2 $E_x(X)$ diambil urutan *plaintext* pada posisi ke-1 yang dilanjutkan dengan mengambil posisi *plaintext* dengan jeda *c* dan jeda *(K*2-d)* secara bergantian selama nilai jeda ≤ ukuran *plaintext* dengan nilai *d* dan *c* ditambah 2.

   Baris = K analog dengan proses baris ke-2

c. *Cipher* didapat dengan menampilkan $E_x(X)$.

Metode *rail fence* ini memiliki tingkat keamanan yang rendah sehingga mudah di bobol oleh seorang ahli dengan mencoba beberapa nilai kedalaman untuk menentukan banyaknya baris yang digunakan. Meskipun begitu metode *rail fence* ini dapat digabungkan dengan metode keamanan yang lain untuk meningkatkan keamanan data yang telah di enkripsi sehingga lebih susah untuk dipecahkan ataupun dibobol oleh orang yang tidak berwenang (Purba & Puspasari, 2020).

Contoh Penerapan Algoritma *Railfence Chiper*, sebagai berikut :

1. Proses Enkripsi

   Plainteks :

   WE ARE DISCOVERED FLEE AT ONCE

   Kunci : 3 Baris

   Proses Enkripsi :

W			R			I			O			R			F			E			O			E		
	E			E			S			V			E			L			A			N			X	
		A			D			C			E			D			E			T			C			X

   Ciphertext :

   WRIORFEOE EESVELANX ADCEDETCX

2. Proses Dekripsi

   Ciphertext :

   WRIORFEOE EESVELANX ADCEDETCX

   Kunci : 3 Baris

   Jumlah Karakter : 27 Karakter

   Kunci Dekripsi = 27/3 = 9

   Proses Dekripsi :

Baris 1	W	E	A
Baris 2	R	E	D
Baris 3	I	S	C
Baris 4	O	V	E
Baris 5	R	E	D
Baris 6	F	L	E
Baris 7	E	A	T
Baris 8	O	N	C

Baris 9	E	X	X

Plaintext :

WE ARE DISCOVERED FLEE AT ONCE

2.4. *Android*

Android adalah sebuah sistem operasi perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi. *Android* adalah "sistem operasi berbasis *linux* yang di gunakan untuk telepon seluler (*mobile*) seperti telepon pintar (*smartphone*) dan komputer tablet (Safitri & Basuki, 2020).

2.5. *Android Studio*

Android studio ini adalah lingkungan pengembangan baru dan terintegrasi dengan penuh, yang telah di rilis oleh *google* untuk sistem operasi *Android* dan di rancang untuk menjadi peralatan baru dalam pengembangan aplikasi dan memberi *alternaif* selain *Eclips* yang saat ini menjadi IDE yang banyak dipakai (Safitri & Basuki, 2020).

Android Studio adalah *Integrated Development Environment* (IDE) resmi untuk pengembangan aplikasi Android, yang didasarkan pada *IntelliJ IDEA*. Selain sebagai editor kode dan fitur developer *IntelliJ* yang andal, Android Studio memiliki fitur-fitur yang digunakan dalam pembuatan aplikasi, seperti (https://developer.android.com/studio/intro?hl=id, 2022) :

a) Sistem *build* berbasis *Gradle* yang fleksibel
b) Emulator yang cepat dan kaya fitur

c) Lingkungan terpadu tempat Anda bisa mengembangkan aplikasi untuk semua perangkat Android
d) Terapkan Perubahan untuk melakukan *push* pada perubahan kode dan *resource* ke aplikasi yang sedang berjalan tanpa memulai ulang aplikasi
e) *Template* kode dan integrasi *GitHub* untuk membantu Anda membuat fitur aplikasi umum dan mengimpor kode sampel
f) *Framework* dan alat pengujian yang lengkap
g) Alat *lint* untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya
h) Dukungan C++ dan NDK
i) Dukungan bawaan untuk *Google Cloud Platform*, yang memudahkan integrasi *Google Cloud Messaging* dan *App Engine*

Setiap *project* di Android Studio berisi satu atau beberapa modul dengan *file* kode sumber dan *file resource*. Jenis modul meliputi :

a) Modul aplikasi Android
b) Modul *library*
c) Modul Google *App Engine*

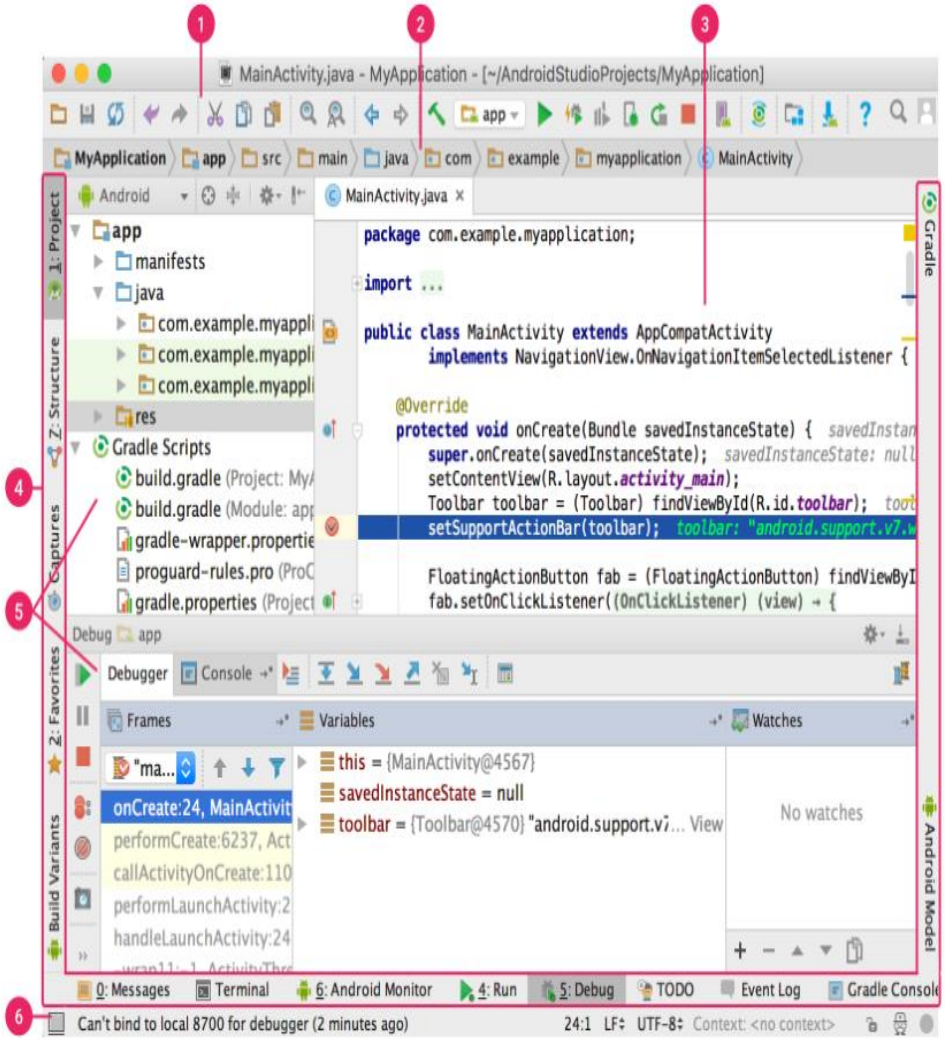
Secara *default*, *Android Studio* menampilkan *file project* Anda dalam tampilan *project Android*, seperti yang ditunjukkan pada gambar 1. Tampilan ini disusun menurut modul untuk memberikan akses cepat ke *file* sumber utama *project* Anda.

Semua *file build* terlihat di tingkat teratas di bagian *Gradle Script* dan setiap modul aplikasi berisi folder berikut:

a) manifes: Berisi *file AndroidManifest.xml*.

b) *java*: Berisi *file* kode sumber *Java*, termasuk kode pengujian *JUnit*.

c) *res*: Berisi semua *resource* non-kode, seperti tata letak *XML*, *string UI*, dan gambar *bitmap*.



Gambar 2.1. Jendela Utama Android Studio

Keterangan Gambar 2.1 :

1) *Toolbar* memungkinkan Anda melakukan berbagai tindakan, termasuk menjalankan aplikasi dan meluncurkan alat Android.

2) Menu navigasi membantu Anda menjelajah *project* dan membuka *file* untuk diedit. Menu ini memberikan tampilan struktur yang lebih ringkas yang terlihat di jendela Project.

3) Jendela editor adalah tempat Anda membuat dan memodifikasi kode. Bergantung pada jenis *file* yang ada, editor ini dapat berubah. Misalnya, saat menampilkan *file* tata letak, editor akan menampilkan *Layout Editor*.
4) Panel jendela fitur berada di sisi luar jendela IDE dan berisi tombol-tombol yang memungkinkan Anda memperluas atau menciutkan setiap jendela fitur.
5) Jendela fitur memberi Anda akses ke tugas tertentu seperti pengelolaan *project*, penelusuran, kontrol versi, dan lainnya. Anda dapat memperluas dan menciutkan jendela ini.
6) Status bar menampilkan status *project* Anda dan IDE itu sendiri, serta semua peringatan atau pesan.

2.6. Bahasa Pemrograman Java

Bahasa pemrograman java bersifat open source dan merupakan produk dari Sun Microsystem dan sekarang dipegang oleh Oracle. Bahasa java adalah bahasa modern yang telah diterima masyarakat komputasi dunia. Sifat dan jenis bahasa pemrograman java yaitu bahasa pemrograman multiplatform (dapat berjalan di berbagai macam sistem operasi) karena pada dasarnya java mempunyai JRE (Java Runtime Environment) atau sebagai mesin tersendiri untuk meneksekusi binary code hasil dari compilasi program yang telah dibuat, berbeda dengan bahasa pemrograman visual basic, C++ yang memanfaatkan komponen sistem dalam windows untuk mengeksekusi binary code hasil kompilasi program (Efitra, 2021).

Java merupakan bahasa pemrograman dan sekaligus juga suatu platform dimana tidak seperti bahasa pemrograman yang lainnya yang menyediakan compiler, namun java memiliki mesin virtual sendiri yang disebut sebagai Java Virtual Machine (JVM) dan juga runtime environment atau yang dikenal sebagai Java Runtime Environment (JRE) (Dewanta & Nuha, 2021).

2.7. Bahasa Pemrograman *PHP*

Hypertext Preprocessor atau lebih akrab dengan *PHP* merupakan bahasa pemrograman *script server-side* yang di desain untuk pengembangan *web*. *PHP* disebut bahasa pemrograman *server-side* karena diproses pada komputer *server*. Hal ini berbeda dengan bahasa pemrograman *client-side* seperti *javascript* yang diproses di dalam *web browser*. *PHP* dapat digunakan secara gratis dan bersifat *open source*. *PHP* dirilis dalam lisensi *PHP License*, sedikit berbeda dengan lisensi *GNU* (*General Public License*) yang biasa digunakan untuk proyek *open source* (Jannah, et al., 2019).

Saat ini sudah banyak *web server* yang dapat di instal di dalam komputer, salah satunya aplikasi *Xampp*. Di dalam aplikasi ini terdapat beberapa fitur yang digunakan untuk menjalankan kode *PHP*, termasuk *web server Apache*. *Web Server Apache* berguna untuk memilah cara menjalankan kode *script* yang telah ditulis, sehingga *apache* akan memberitahu *web server* bahwa kode yang sedang dijalankan adalah kode *PHP*.

2.8. *Database MySQL*

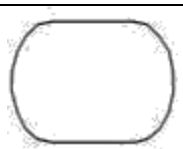
MySQL adalah sebuah basis data yang mengandung satu atau jumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel (Hans, 2016).

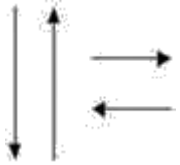
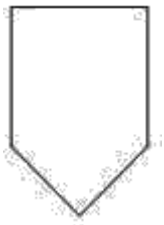
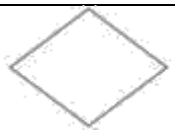
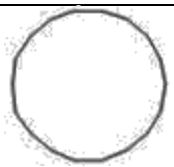
MySQL adalah *database server open source* yang cukup popular keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat *software database* ini banyak digunakan oleh praktisi untuk membangun suatu *project*. Adanya fasilitas *API* (*Application Programming Interface*) yang dimiliki oleh *MySQL*, memungkinkan bermacam – macam aplikasi komputer yang ditulis dengan berbagai bahasa pemrograman dapat mengakses basis data *MySQL*.

2.9. *Flowchart*

Flowchart adalah dalam bahasa Indonesia diagram alir, merupakan diagram yang memuat simbol-simbol grafis yang menyatakan aliran algoritma atau proses dari langkat-langkat instruksi dalam bentuk-bentuk kotak persegi dan bulat dan pernyataan instruksi, dimana hubungan dan urutan proses tiap instruksi ditujukan dengan simbol tanda panah (Anggrawan, 2018).

Tabel 2.1. Simbol *Flowchart*

Simbol	Keterangan	Penjelasan
	Simbol *Terminator* (simbol *start* dan *end*)	Simbol untuk tanda mulai (*start*) dan tanda selesai (*stop*/*end*) dari kegiatan proses

	Simbol Arah Aliran	Simbol yang menghubungkan antara simbol yang satu dengan simbol lainnya (atau antara kegiatan proses) dan sekaligus menyatakan arah proses
	Simbol keluaran/masukan (*Input*/*output*)	Simbol yang menyatakan proses *input* dan *output* (berlaku untuk semua media *input* dan *output*)
	Simbol Proses	Simbol yang melambangkan kegiatan pemrosesan/pengolahan *input*
	Simbol Konektor	Simbol untuk tanda penyambungan proses pada lembar atau halaman yang berbeda.
	Simbol Percabangan atau Pilihan Keputusan	Simbol proses pemilihan keputusan tergantung kondisi, jika pemeriksaan kondisi terpenuhi benar maka jalur pilihan yang diproses adalah jalur ya atau *yes*, dan sebaliknya jika pemeriksaan tidak terpenuhi tidak benar, maka jalur tidak atau No.
	Simbol Konektor	simbol untuk tanda penyambungan proses pada lembar atau halaman yang sama

BAB III

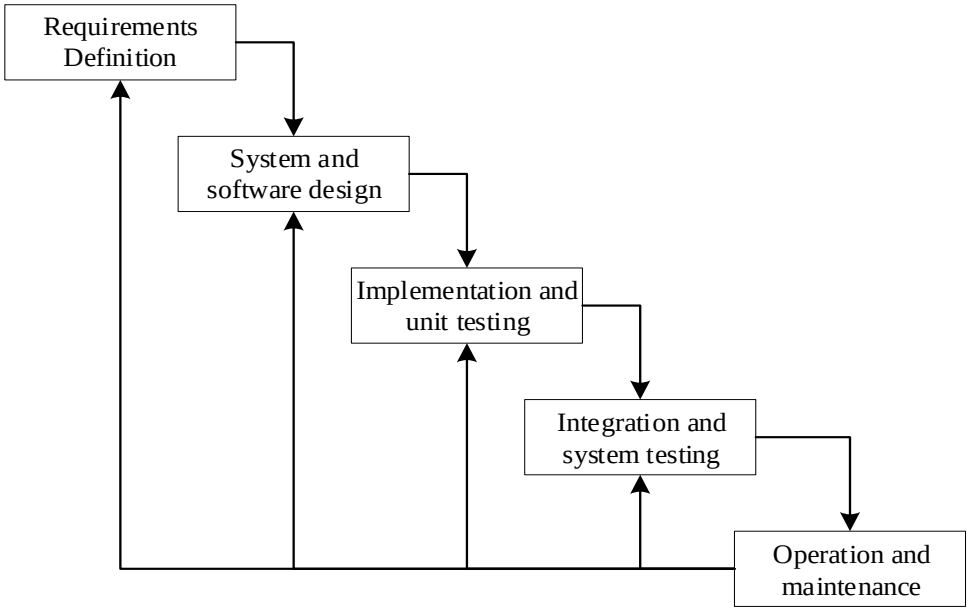
METODOLOGI PENELITIAN

3.1. Tempat dan Waktu Penelitian

Waktu penelitian dimulai dari 1 Oktober 2022 sampai dengan 1 Maret 2023. Penelitian ini dilakukan secara mandiri di kediaman pribadi yang beralamat di Jalan Iskandar 6b RT.15 RW.04 Kelurahan Tengah padang Kota Bengkulu Provinsi Bengkulu.

3.2. Metode Penelitian

Metode penelitian yang digunakan yaitu Metode *waterfall* dimana tahapan-tahapan dari metode ini, seperti Gambar 3.1.



Gambar 3.1. Metode Waterfall

Keterangan :

1) *Requirements analysis and definition*

Layanan sistem, kendala, dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna yang kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

2) *System and software design*

Tahapan perancangan sistem mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat lunak dan hubungannya.

3) *Implementation and unit testing*

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

4) *Integration and system testing*

Unit-unit individu program atau program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak.

5) *Operation and maintenance*

Tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. *Maintenance* melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru.

3.3. Perangkat Keras dan Perangkat Lunak

1. Perangkat Keras (*Hardware*)

   Perangkat keras (*Hardware*) yang digunakan dalam penelitian ini, antara lain :

   a. Laptop Lenovo
   b. HDD 1TB
   c. RAM 8GB

2. Perangkat Lunak (*Software*)

   Perangkat lunak (*Software*) yang digunakan dalam penelitian ini, antara lain :

   a. Sistem Operasi Windows 10
   b. Android Studio Chipmunk 2021.2.1 Patch 1
   c. XAMPP

3.4. Metode Pengumpulan Data

Adapun metode pengumpulan data yang digunakan, antara lain :

a. Studi Praktikum

Tahap ini, penulis melakukan penerapan algoritma *rail fence cipher* dalam aplikasi berbasis android. Studi praktikum dilakukan dengan menguji coba aplikasi dengan memberikan masukan input untuk mengetahui apakah aplikasi sudah berjalan dengan semestinya.

b. Studi Pustaka

Studi pustaka adalah suatu metode pengumpulan data yang diambil dari perpustakaan atau instansi yang berupa karya ilmiah, jurnal, buku-buku yang berhubungan dengan penulisan ini.

3.5. Metode Perancangan Sistem

Pada subbab ini, dilakukan analisis terhadap sistem yang akan dibangun dimana dibagi menjadi 2 bagian yaitu analisis sistem aktual dan analisis sistem baru.

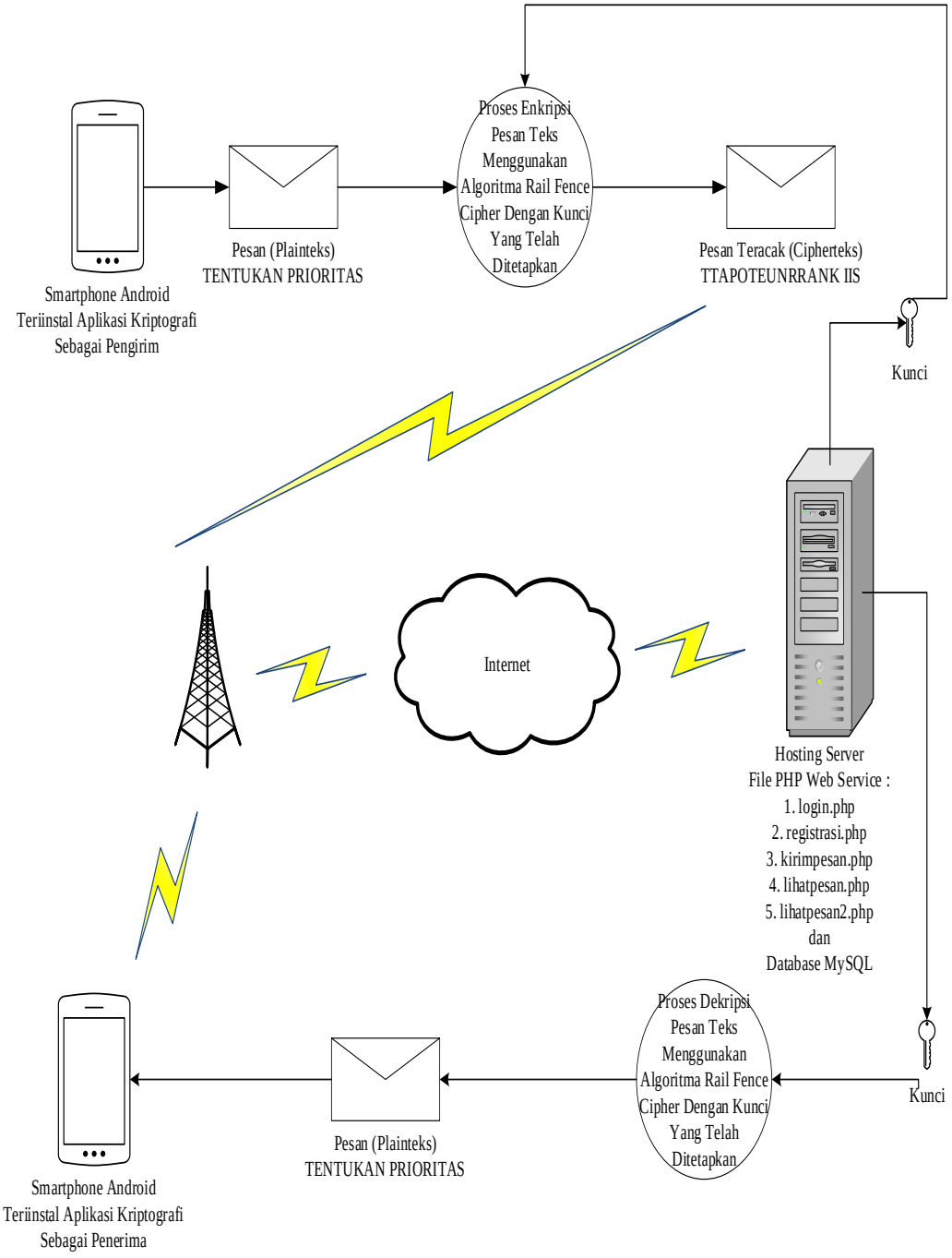
3.5.1. Analisis Sistem Aktual

Pertukaran informasi jarak jauh ini menuntut keamanan terhadap kerahasiaan informasi yang dipertukarkan. Suatu informasi akan memiliki nilai tinggi apabila menyangkut tentang aspek keamanan, kepentingan umum dan kepentingan pribadi, sehingga informasi tersebut akan banyak diminati oleh pihak lain yang tidak berwenang untuk mendapatkan isi dari informasi tersebut. Salah satu cara dalam mengamankan informasi tersebut yaitu dengan melakukan keamanan terhadap informasi melalui kriptografi.

3.5.2. Analisis Sistem Baru

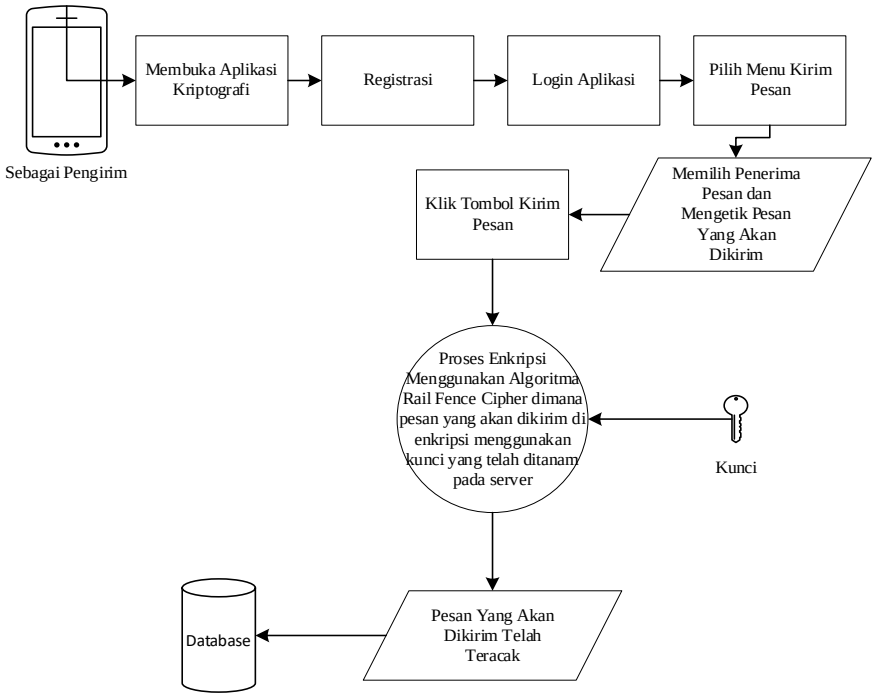
Analisis sistem baru merupakan pengembangan dari hasil analisis sistem aktual. Oleh karena itu, diperlukan suatu keamanan terhadap informasi tersebut sebelum informasi dikirim ke penerima. Kriptografi adalah bidang ilmu yang mempelajari tentang cara untuk menjaga keamanan pesan dalam proses pengiriman dengan menggunakan metode penyandian tertentu, dengan tujuan agar informasi dalam pesan tersebut tidak disalahgunakan oleh orang yang bukan penerima aslinya.

Aplikasi ini dibangun berbasis android dengan menggunakan bahasa pemrograman *java* sebagai *interface* android, bahasa pemrograman *PHP* sebagai *JSON* (penghubung/*web service*) dan *database MySQL* sebagai tempat penyimpanan pesan teks. Blog diagram aplikasi dimana terlihat *smarphone* android telah terinstall aplikasi android dan terkoneksi internet, kemudian melakukan pengiriman pesan. Dan pesan tersebut akan tersimpan di *database MySQL* pada *hosting server* melalui perintah PHP *Web Service*. Adapun blog diagram aplikasi seperti Gambar 3.2.



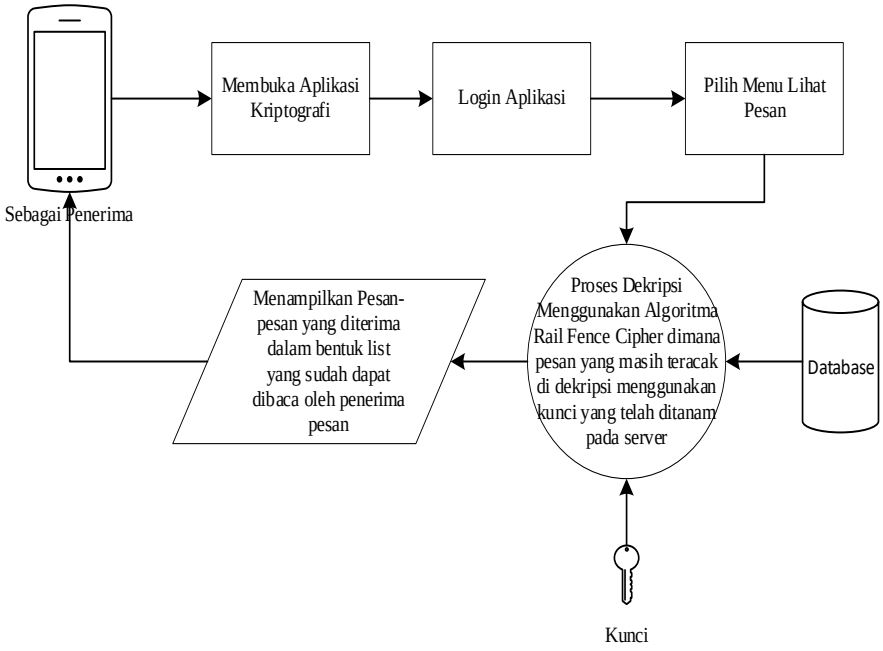
Gambar 3.2. Blog Diagram Aplikasi

Adapun ilustrasi pengiriman pesan teks melalui aplikasi smarphone android dimana user sebagai pengirim, seperti Gambar 3.3.



Gambar 3.3. Ilustrasi User Sebagai Pengirim Pesan Teks

Adapun ilustrasi pengiriman pesan teks melalui aplikasi smarphone android dimana user sebagai pengirim, seperti Gambar 3.4.



Gambar 3.4. Ilustrasi User Sebagai Penerima Pesan Teks

A. Penerapan Algoritma Rail Fence Cipher

Proses enkripsi dilakukan dengan cara mempersiapkan *plaintext* serta kunci yang digunakan untuk mendapatkan hasil *ciphertext*. Sebaliknya dalam proses dekripsi dilakukan dengan cara mempersiapkan *ciphertext* serta kunci yang digunakan untuk mendapatkan hasil *plaintext*.

1. Proses Enkripsi

   Plainteks :

   TENTUKAN PRIORITAS

   Kunci : 2

   Dalam proses enkripsi, kunci dibuat ke dalam bentuk 3 baris sepanjang karakter plainteks, sebagai berikut :

T		N		U		A		(spasi)		R		O		I		A	
	E		T		K		N		P		I		R		T		S

   Ciphertext :

TNUA(spasi)ROIAETKNPIRTS

2. Proses Dekripsi

   Ciphertext :

   TNUA(spasi)ROIAETKNPIRTS

   Kunci : 32

   Dalam proses dekripsi, kunci 2 tersebut diolah kembali dengan cara jumlah karakter dibagi dengan kunci, sehingga diperoleh kunci dekripsi sebagai berikut :

   Jumlah karakter = 18 karakter

   Kunci dekripsi = 18/2 = 9

   Proses Dekripsi :

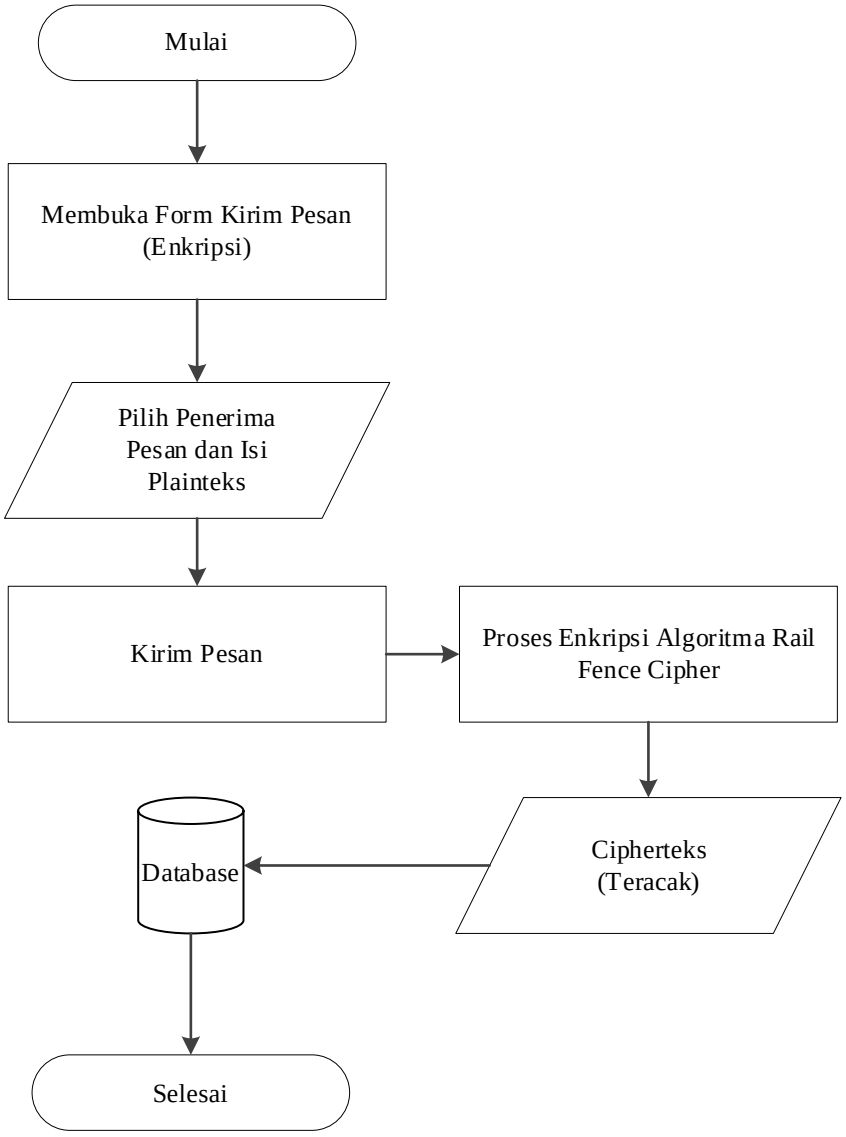
Baris 1	T	E
Baris 2	N	T
Baris 3	U	K
Baris 4	A	N
Baris 5	(spasi)	P
Baris 6	R	I
Baris 7	O	R
Baris 8	I	T
Baris 9	A	S

   Plaintext :

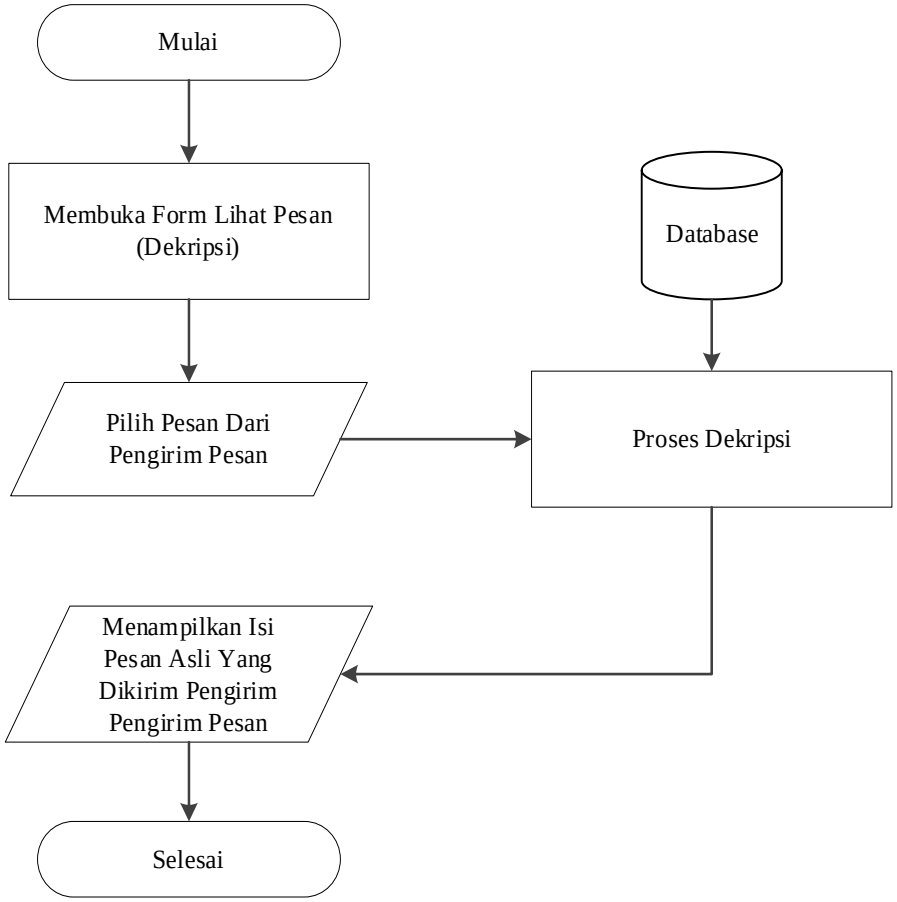
   TENTUKAN PRIORITAS

B. *Flowchart*

Flowchart digunakan untuk menggambarkan suatu sistem dalam hal ini yaitu menggambarkan bagan alir proses enkripsi dan dekripsi pada algoritma *rail fence cipher*. Adapun *flowchart* tersebut, terlihat pada gambar 3.5. dan gambar 3.6.



Gambar 3.5. *Flowchart* Enkripsi



Gambar 3.6. *Flowchart* Dekripsi

C. Rancangan File

Adapun rancangan file pada aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher*, antara lain :

1. File User

   Nama Tabel : tbluser

   Kunci Utama : kduser

   Kunci Tamu : -

   Tabel 3.1. File User

Field	Tipe Data	Size

kduser	int	5
nama	Varchar	50
hp	Varchar	12
email	Varchar	50
username	Varchar	50
password	Varchar	50

2. File Pesan

   Nama Tabel : tblpesan

   Kunci Utama : kdpesan

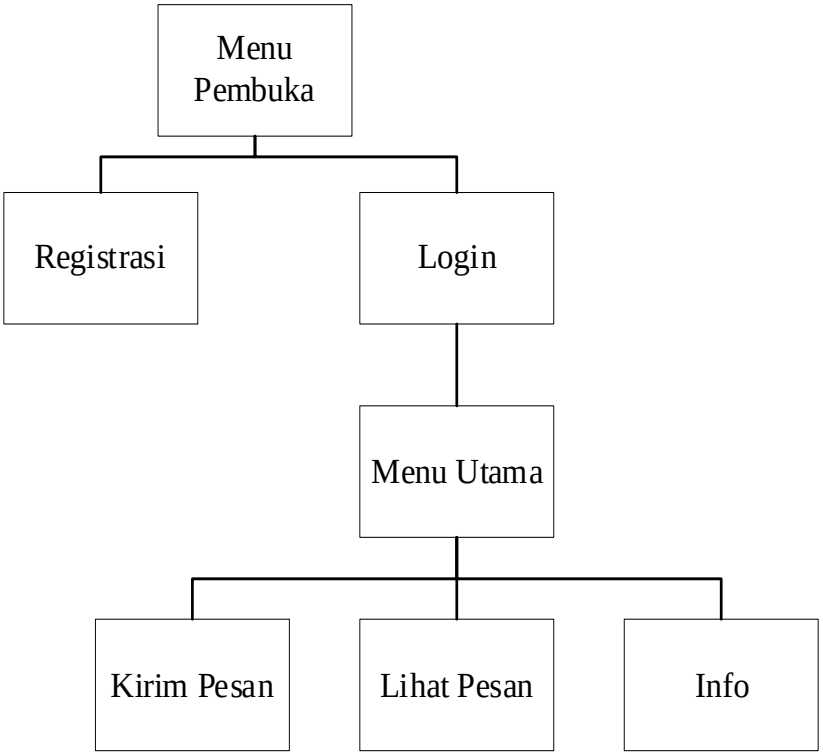
   Kunci Tamu : -

   Tabel 3.2. File Pesan

Field	Tipe Data	Size
kdpesan	int	5
tglpesan	Datetime	-
userkirim	Varchar	50
userterima	Varchar	50
isipesan	Text	-

D. Rancangan Struktur Menu

Rancangan struktur menu pada aplikasi terdiri menjadi beberapa bagian dimana terdapat menu-menu yang dapat diakses. Adapun rancangan struktur menu pada aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher* seperti Gambar 3.7.



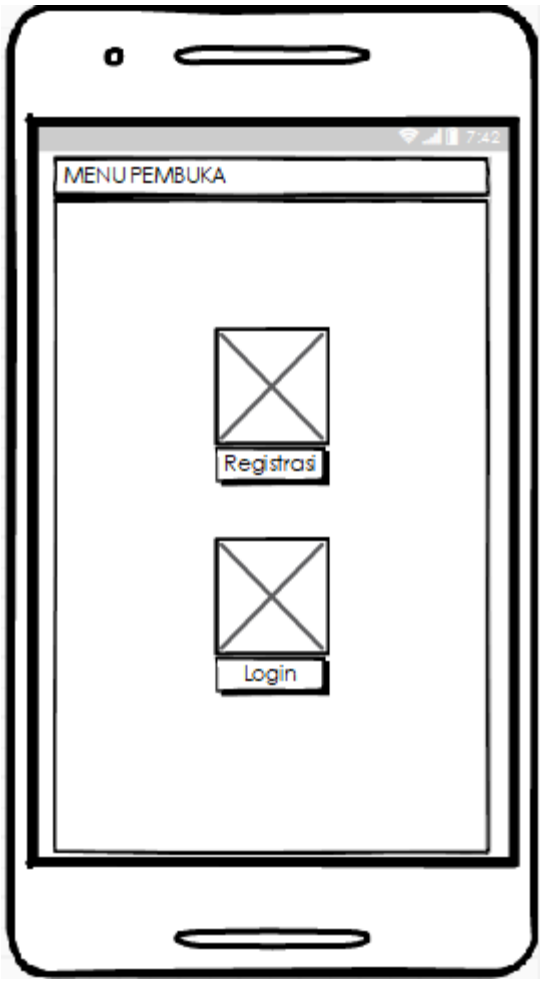
Gambar 3.7. Struktur Menu

E. Perancangan Aplikasi

Adapun rancangan aplikasi kriptografi berbasis android menggunakan algoritma *rail fence cipher* antara lain :

1. Menu Pembuka

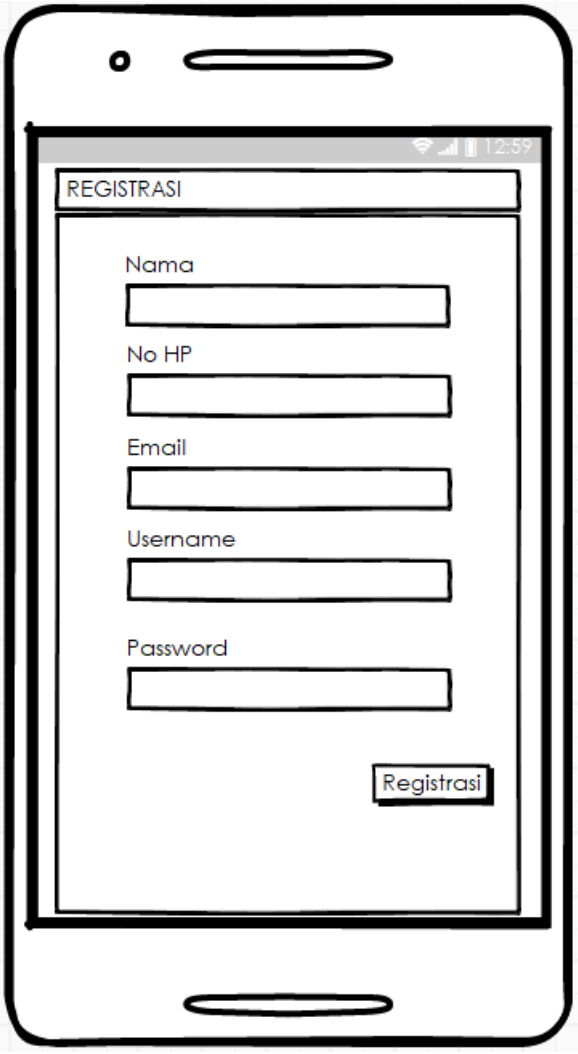
   Merupakan rancangan yang akan tampil pertama kali ketika membuka aplikasi, dimana terdapat 2 button menu yaitu registrasi dan login. Adapun rancangan menu pembuka seperti Gambar 3.8.



Gambar 3.8. Menu pembuka

2. Registrasi

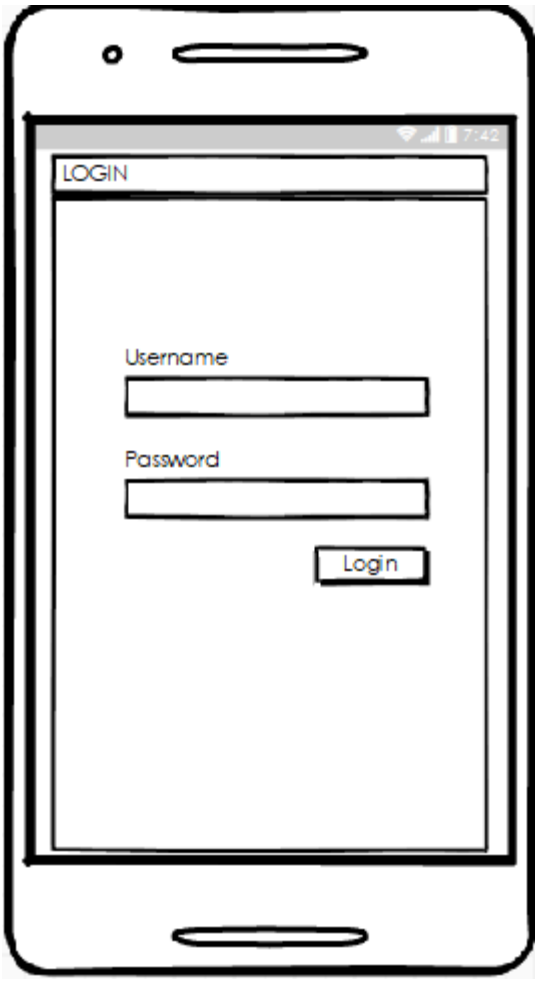
Merupakan rancangan yang digunakan oleh pengguna aplikasi untuk melakukan registrasi agar dapat melakukan login pada aplikasi. Adapun rancangan form registrasi seperti Gambar 3.9.



Gambar 3.9. Registrasi

3. Login

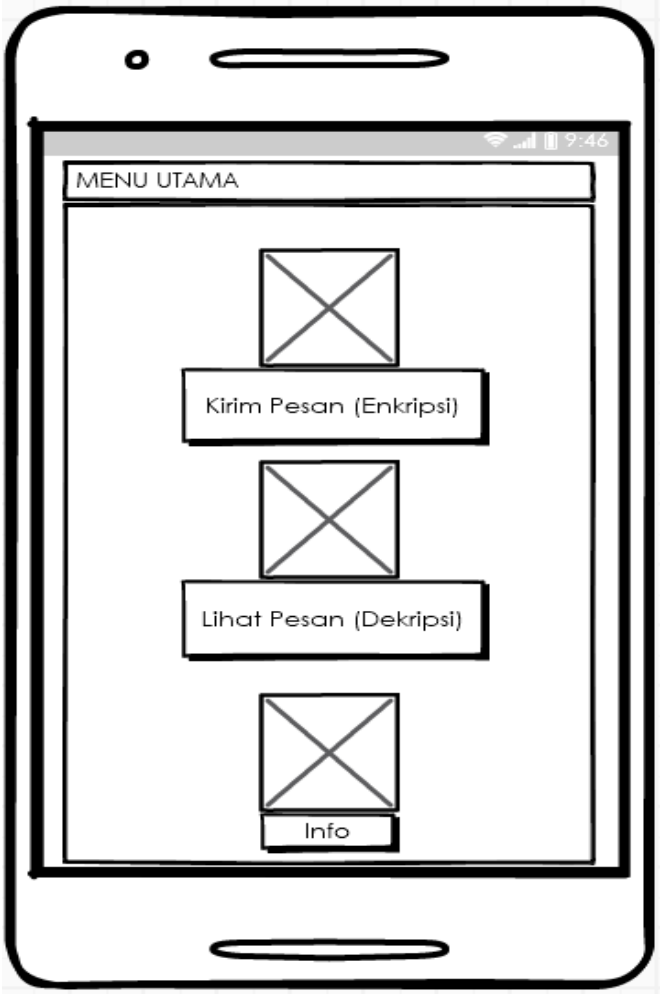
Merupakan rancangan yang digunakan oleh pengguna aplikasi untuk login pada aplikasi agar dapat bertukar informasi kepada pengguna lainnya. Adapun rancangan form login seperti Gambar 3.10.



Gambar 3.10. Login

4. Menu Utama

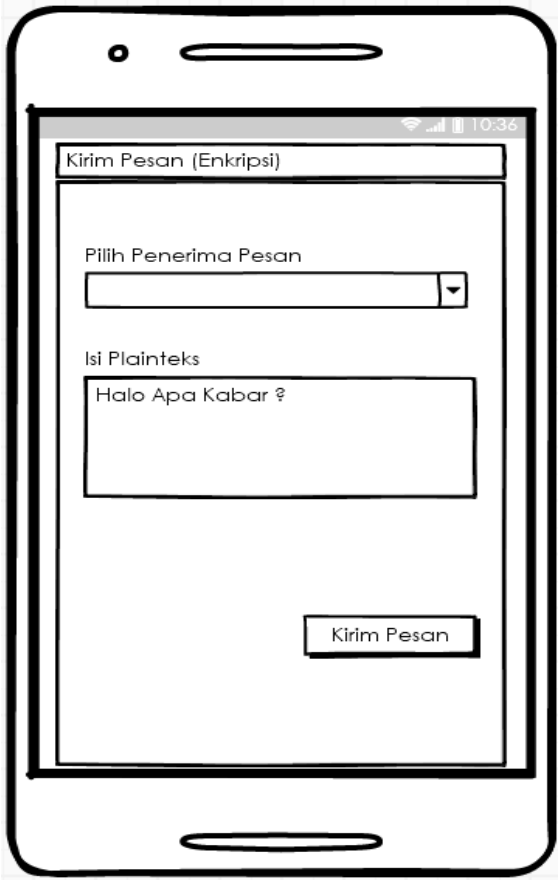
Merupakan rancangan yang akan tampil pertama kali ketika membuka aplikasi, dimana terdapat 3 button menu yaitu kirim pesan (enkripsi), lihat pesan (dekripsi), dan info. Adapun rancangan menu utama seperti Gambar 3.11.



Gambar 3.11. Menu Utama

5. Kirim Pesan (Enkripsi)

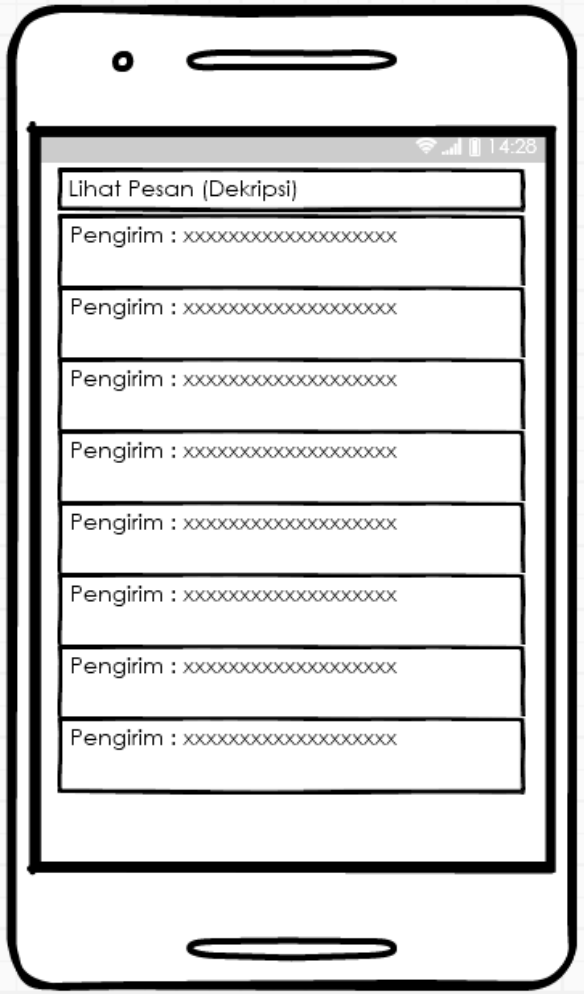
Merupakan rancangan form yang digunakan untuk melakukan pengiriman pesan ke penerima pesan. Pada form ini telah diterapkan algoritma rail fence cipher yang digunakan untuk melakukan proses enkripsi tersebut. Proses enkripsi dilakukan dengan cara pengirim mengetik plainteks (teks yang asli) terlebih dahulu, kemudian klik tombol enkripsi dan secara otomatis akan pesan akan tersimpan ke dalam database dalam bentuk ciphertext (teks yang telah teracak) hasil enkripsi. Adapun rancangan form kirim pesan (enkripsi) seperti Gambar 3.12.



Gambar 3.12. Kirim Pesan (Enkripsi)

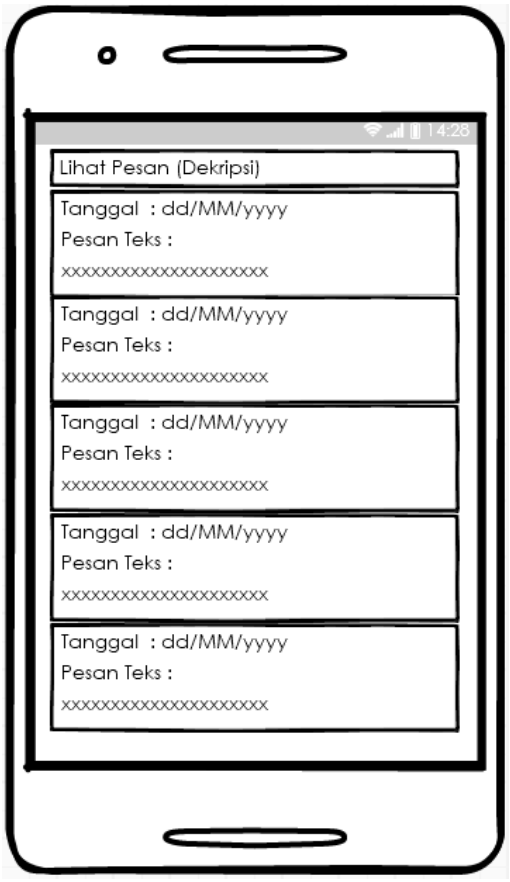
6. Lihat Pesan (Dekripsi)

Merupakan rancangan form yang digunakan untuk melakukan melihat pesan-pesan yang diterima telah dikirim oleh pengirim pesan. Pada form ini terjadi proses dekripsi secara otomatis sehingga penerima pesan langsung dapat membaca pesan yang dikirim oleh pengirim pesan tersebut dalam bentuk list pesan, seperti Gambar 3.13



Gambar 3.13. Lihat Pesan (Dekripsi) (1)

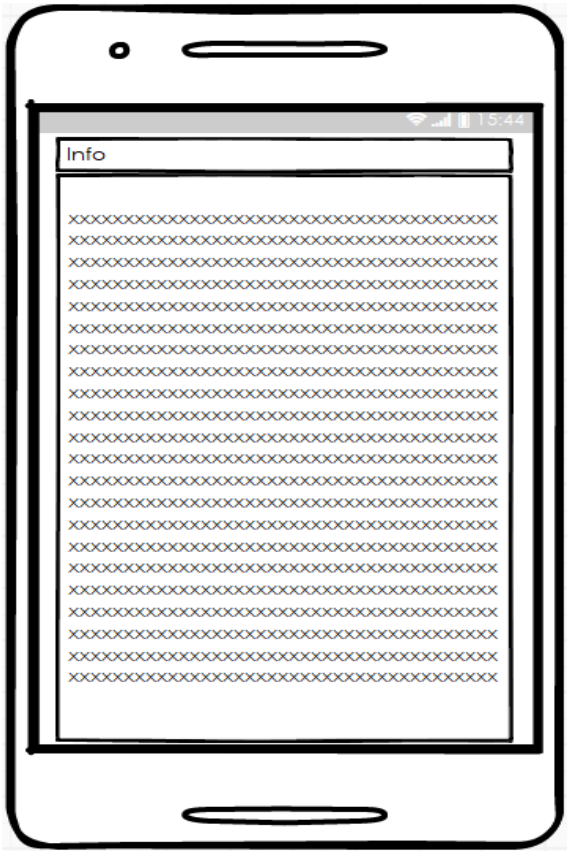
Pada Gambar 3.13. terdapat list nama pengirim pesan. Jika ingin melihat pesan yang dikirim oleh pengirim pesan, maka klik salah satu pada list tersebut, dan akan muncul pesan seperti Gambar 3.14.



Gambar 3.14. Lihat Pesan (Dekripsi) (2)

7. Info

   Merupakan rancangan form yang berisi informasi atau cara penggunaan aplikasi untuk mempermudah pengguna/user dalam menjalankan aplikasi ini. Adapun rancangan form info seperti Gambar 3.15.



Gambar 3.15. Info

3.6. Pengujian Sistem

Pengujian sistem merupakan proses eksekusi sistem dengan tujuan mencari kesalahan atau kelemahan dari program tersebut. Proses tersebut dilakukan dengan mengevaluasi kemampuan program. Suatu program yang diuji akan dievaluasi apakah keluaran atau *output* yang dihasilkan telah sesuai dengan yang diinginkan atau tidak. Metode pengujian yang dipakai dalam sistem ini adalah metode *black box*. Adapun komponen pengujian yang dilakukan, seperti Tabel 3.3.

Tabel 3.3. Pengujian Sistem

No.	Form Yang Diuji	Rencana Pengujian	Hasil Pengujian
1	Form Registrasi	Mengisi field pada data registrasi dan menyimpan data registrasi tersebut.	
2	Form Login	Memasukkan username dan password yang benar pada form login	
		Memasukkan username atau password yang salah pada form login	
3	Kirim Pesan	Melakukan pengiriman pesan teks dengan memilih penerima pesan dan mengisi pesan yang akan dikirim ke penerima	
		Menjalankan web service JSON kirimpesan.php ketika pengirim pesan klik tombol “kirim pesan”	
4	Lihat Pesan	Melihat pesan-pesan yang masuk yang telah dikirim oleh setiap pengirim pesan	
		Menjalankan web service JSON lihatpesan.php ketika penerima pesan membuka form lihat pesan	
5	database pesan teks	Melihat database yang menyimpan pesan teks antara pengirim dan penerima pesan apakah berbentuk acak (cipherteks) atau berbentuk teks asli (plainteks)	