APLIKASI CHAT MENGGUNAKAN ALGORITMA VIGENERE CIPHER BERBASIS ANDROID

Oleh :

Muhamad Ismail Saleh 3311311002

Disusun untuk memenuhi syarat kelulusan Program Diploma III

PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

BATAM

2017

HALAMAN PENGESAHAN

APLIKASI CHAT MENGGUNAKAN ALGORITMA VIGENERE CIPHER

BERBASIS ANDROID

Disusun oleh :

Muhamad Ismail Saleh

(3311311002)

Telah diuji dan dipertahankan di depan Tim Penguji
dalam Sidang Tugas Akhir
pada tanggal 30 Mei 2017
dan dinyatakan **LULUS**.

Batam, 05 Juni 2017

Diperiksa dan disetujui oleh;

Pembimbing,

Dwi Ely Kurniawan, M.Kom
NIK. 112094

Penguji I,

Ari Wibowo, M.T
NIK. 197604282012121003

Penguji II,

Sandi Prasetyaningsih, S.ST
NIP. 113106

HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311311002

Nama :Muhamad Ismail Saleh

adalah mahasiswa Teknik Informatika Politeknik Negeri Batam yang menyatakan bahwa tugas akhir dengan judul:

Aplikasi Chat Menggunakan Algoritma Vigenere Cipher Berbasis Android

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil tugas akhir ini.

Batam, 02 Juni 2017

Muhamad Ismail Saleh
3311311002

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadirat Allah Swt. Yang telah memberikan rahmat, hidayah dan karuniaNya sehingga penulis dapat menyelesaikan tugas akhir yang menjadi salah satu syarat untuk menyelesaikan program studi Teknik Informatika jenjang Diploma-3 di Politeknik Negeri Batam. Shalawat serta salam semoga tetap tercurah kepada Nabi besar Muhammad Shalallahu,,alaihi wasallam, keluarga, sahabat dan para pengikutnya hingga hari kiamat. Dalam penyusunan laporan ini penulis menyadari banyak sekali kekurangan, namun berhubung banyaknya pihak luar yang mendukung dan turut membantu, sehingga laporan ini dapat diselesaikan tepat waktu. Oleh karena itu dalam kesempatan ini penulis ingin mengucapkan terimakasih yang sebesar-besarnya kepada:

1. Bapak Ari Wibowo, MT., selaku Wali kelas.
2. Bapak Afdol Dzikri, S.ST., M.T., selaku Ketua Program Studi Teknik Informatika Politeknik Negeri Batam.
3. Bapak Dwi Ely Kurniawan, S.Pd., M.Kom., selaku dosen Tugas Akhir dan selaku dosen pembimbing dalam penyusunan Tugas Akhir.
4. Seluruh dosen Teknik Informatika Politeknik Negeri Batam yang telah mengajar dan memberikan arahan kepada penulis selama ini.
5. Ibu, Istri, Adik dan orang tersayang yang telah banyak memberikan doa, motivasi dan dorongan dalam menempuh perkuliahan dan dalam menyelesaikan tugas akhir ini.
6. Teman-teman di Politeknik Negeri Batam dan semua rekan kerja yang telah memberikan semangat dan motivasi kepada penulis hingga saat ini.

Penulis menyadari bahwa dalam penyusunan tugas akhir ini masih banyak kekurangannya. Oleh karena itu penulis mengharapkan kritik dan saran yang dapat menyempurnakan penyusunan tugas akhir ini sehingga dapat bermanfaat dan berguna untuk pengembangan ilmu pengentehuan dan teknologi. Amin.

Batam, 02 Juni 2017

Penulis

ABSTRAK

APLIKASI CHAT MENGGUNAKAN ALGORITMA VIGENERE CHIPPER BERBASIS ANDROID

Perkembangan teknologi saat ini, memungkinkan manusia dapat berkomunikasi dan dapat bertukar informasi secara jarak jauh. Seiring dengan itu tuntutan akan kerahasiaan informasi saling dipertukarkan tersebut semakin meningkat. Oleh karena itu, dikembangkanlah cabang ilmu yang mempelajari tentang cara-cara pengamanan data atau lebih dikenal dengan kriptografi. Algoritma *vigenere cipher* termasuk dalam kriptografi klasik yang memiliki kunci simetris (hanya ada satu kunci) yang digunakan untuk menenkripsi ataupun mendeskripsi. Aplikasi ini melakukan kriptografi pada teks berupa huruf, hasil dari penelitian ini adalah berupa aplikasi berbasis android yang dapat melakukan pengiriman pesan yang telah terenkripsi menggunakan algoritma *vigenere cipher,* sehingga kerahasiaan dari pesan tersebut dapat terjaga keamanannya.

Kata Kunci : Kriptografi, Chat, *Android, Vigenere Cipher*.

ABSTRACT

APPLICATIONS USING CHAT ALGORITHM VIGENERE CHIPPER ANDROID BASED

Current technological developments enable people to communicate and can exchange information remotely. Along with that will the confidentiality of interchangeable information that is increasing. Therefore, developed a branch of science that learn about the ways of securing data or better known as cryptography. The vigenere cipher algorithm belongs to classical cryptography that has a symmetric key (there is only one key) that is used to encrypt or. This application is done cryptography on the text of the capital letters, the results of this study is an android-based applications that can send a message that has been encrypted using vigenere cipher algorithm, so the confidentiality of this message can be gated.

Keywords: Cryptography, Chat, Andorid, Vigenere cipher.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang

Seiring berkembangnya jaringan komunikasi, menjadikan informasi sangat penting dan bermanfaat untuk pertukaran berbagi informasi, baik dalam bentuk teks, gambar, audio maupun video. Namun saat ini semakin banyak kejahatan yang membuat khawatir dengan keamanan data yang akan dikirim, sehingga perlu keamanan dari pesan yang akan dikirim agar tidak bisa dilihat oleh orang yang tidak bertanggung jawab.

Kerahasiaan pesan atau data yang dimiliki oleh seseorang merupakan hal terpenting dalam pengiriman pesan agar pesan tersebut hanya dapat diberikan oleh orang tertentu saja yang dapat mengakses informasi tersebut. Untuk menjaga kerahasiaan pesan diperlukan pengamanan data atau yang lebih dikenal sebagai kriptografi, dimana kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan berita.

Pada pengamanan dalam kriptografi ini banyak metode atau algoritma yang digunakan, salah satu dari algoritma kriptografi adalah *vigenere cipher*. Algoritma *vigenere cipher* termasuk dalam kriptografi klasik yang memiliki kunci simetris (hanya ada satu kunci) yang digunakan untuk menenkripsi ataupun mendeskripsi. Karena *vigenere* merupakan kriptografi klasik maka proses enkripsi dan deskripsinya pun cukup mudah dengan cara subtitusi atau dengan tabel bujur sangkar *vigenere*. Dengan adanya sistem keamanan ini isi pesan yang bersifat personal atau rahasia dapat tersampaikan secara aman.

Berdasarkan uraian diatas, maka penulis akan membuat suatu aplikasi *chat* yang akan mengamankan pesan atau data sehingga tingkat keamanan dan kerahasiaan pesan atau data yang dikirim menjadi lebih baik.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang dibahas di atas, masalah-masalah yang akan dibahas adalah

1. Bagaimana membuat aplikasi chat menggunakan algoritma vigenere cipher?
2. Bagaimana mengimplementasikan algoritma vigenere cipher?

1.3 Batasan Masalah

Adapun batasan masalah dari laporan ini adalah

1. Aplikasi ini tidak membahas proses pengiriman dan penerimaan pesan yang terjadi diluar aplikasi.
2. Aplikasi ini hanya bisa mengirimkan pesan berupa teks.
3. Aplikasi hanya dapat digunakan pada *smartphone* yang berbasis *android.*
4. Proses enkripsi pesan hanya digunakan untuk data teks.
5. Penanganan enkripsi akan dilakukan setelah pesan dikirimkan.
6. Aplikasi tidak menangani lupa *password*
7. Aplikasi hanya bisa dijalankan oleh dua pengguna saja.
8. Aplikasi ini tidak menangani proses enkripsi dan deskripsi berbentuk file, suara, gambar dan video.

1.4 Tujuan Penelitian

Tujuan dari pembuatan aplikasi ini adalah

1. Mengimplementasikan kriptografi pada aplikasi chatting berbasis android.
2. Menguji keamanan aplikasi chat menggunakan algoritma vignere chipher.

1.5 Sistematika Penulisan

Laporan ini terdiri dari bab Pendahuluan, Tinjauan Pustaka, Analisis dan Perancangan, Implementasi dan Pengujian, Kesimpulan dan Saran.

Bab I : Pendahuluan

Bab ini membahas berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan dan sistematika penulisan untuk menjelaskan pokok-pokok pembahasan.

Bab II : Landasan Teori

Bab ini menjelaskan tentang landasan teori yang digunakan dalam merancang aplikasi.

Bab III : Analisis dan Perancangan Aplikasi

Bab ini berisi tentang perancangan aplikasi yang dibuat tentang struktur navigasi, rancangan tampilan aplikasi, langkah-langkah pembuatan, spesifikasi perangkat lunak dan perangkat keras yang mendukung implementasi.

Bab IV : Implementasi dan Pengujian

Bab ini memuat uraian langkah implementasi dan pengujian atau validasi

Bab V : Kesimpulan dan Saran

Bab ini menjelaskan tentang simpulan yang diambil dari hasil perancangan aplikasi serta memuat tentang saran yang diberikan untuk pengembangan aplikasi selanjutnya.

BAB II
LANDASAN TEORI

2.1.Tinjauan Pustaka

Beberapa jurnal dan skripsi yang berkaitan dengan kriptografi, sebagai berikut :

Fitri Apriani (2014) dalam jurnal yang berjudul Aplikasi chatting dengan sistem enkripsi menggunakan algoritma blowfish berbasis android,yang menjelaskan tentang bagaimana menerapkan algoritma blowfish pada aplikasi chatting berbasis android. Aplikasi chatting ini dapat digunakan untuk mengirim pesan yang sifatnya rahasia dengan memasukkan kunci terlebih dahulu sebelum dienkripsi, kunci yang dikirim akan disimpan dalam database untuk melakukan deskripsi.

Angga Kusumah, Maman Abdurrohman, Dodi Wicaksono Sudiharto (2012) dalam journal yang berjudul *secure chatting* menggunakan metode enkripsi blowfish, twofish dan AES, yang menjelaskan tentang perbandingan waktu proses pembangkit kunci, proses enkripsi serta tingkat keamanan dari ketiga algoritma tersebut.

Edy Timanta Karo Karo (2013) dalam journal yang berjudul Analisis dan perancangan keamanan pesan chatting menerapkan algoritma caesar, yang menjelaskan tentang bagaimana proses enkripsi dan metode penyandian dari algoritma caesar cipher.

Septian Fajar Nugraha (2010) dalam skripsi yang berjudul Rancang bangun aplikasi chat conference pada mobile phone dengan menggunakan enkripsi, yang menjelaskan tentang bagaimana merancang aplikasi dan membuat aplikasi chat yang dapa melakukan pengamanan berupa enkripsi pada setiap kata yang ditulis.

Ryan Maulana (2012) dalam skripsi yang berjudul Penerapan algoritma kriptografi WAKE pada aplikasi chatting & internet monitor berbasis LAN, yang menjelaskan tentang bagaimana menerapkan kriptografi WAKE pada aplikasi chatting dan aplikasi chatting mampu melakukan proses enkripsi dan deskripsi pesan untuuk meningkatkan keamanan pesan.

Berdasarkan penelitian diatas maka penulis akan membuat suatu aplikasi *chat* yang akan mengamankan pesan atau data sehingga tingkat keamanan dan kerahasiaan pesan atau data yang dikirim menjadi lebih baik.

2.2.Dasar Teori

2.2.1 Pengertian Chatting

Chatting merupakan aplikasi internet yang menggunakan teknologi *Internet Relay Chat*, yang artinya sebuah percakapan yang dilakukan oleh pengguna internet. IRC memungkinkan dua orang berkomunikasi melalui jaringan internet dalam waktu seketika (*real time*). Maksudnya pesan yang dikirim akan sampai dengan seketika kepada penerima, dan akan dibalas saat itu juga. Berbeda dengan email, komunikasi dengan IRC bisa terjadi apabila kedua belah pihak sama-sama sedang mengakses internet (*online*) pada saat yang sama.

2.2.2 Android

Android adalah sistem operasi untuk *handphone* yang berbasis *linux*. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. *Android* merupakan subset perangkat lunak untuk perangkat *mobile* yang meliputi sistem operasi, *middleware* dan aplikasi inti yang di *release* oleh Google. Sedangkan *android* SDK *(Software Development Kit)* menyediakan *tools* dan API yang diperlukan untuk mengembangkan aplikasi pada *platform android* dengan menggunakan bahasa pemrograman Java. (Mulyadi 2010) *Android* adalah sistem operasi untuk telepon seluler yang berbasis *Linux.*

Android menyediakan *platform* terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google *Inc*. membeli *Android Inc*, pendatang baru yang

membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, *HTC, Intel, Motorola,Qualcomm, T-Mobile,* dan *Nvidia.* Pada saat perilisan perdana *Android*, 5 November 2007, *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode - kode *Android* di bawah *lisensi Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Di dunia ini terdapat dua jenis distributor sistem operasi *Android*. Pertama yang mendapat dukungan penuh dari *Google* atau *GoogleMail Services* (GMS) dan kedua adalah yang benar - benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution (OHD).*

Pada perkembangannya sistem operasi *Android* mengalami beberapa perubahan dan perbaikan, dan yang paling menarik adalah versi keluaran *Android* yang diberi nama seperti nama makanan, berikut versi *Android*yang digunakan dalam aplikasi chat menggunakan algoritma vigenere cipher : Versi 4.2.x bernama *Jelly Bean* yang dirilis pada 13 November 2012.

2.2.3 Java

Bahasa *Java* dikembangkanoleh *Sun Microsystem*tahun 1991 sebagai bagian dari suatu proyek penelitian untuk mengembangkan *software* bagi konsumen barang-barang elektronik seperti televisi, VCR, *toaster* dan mesin – mesin lainnya yang dapat dibeli di swalayan. Tujuan penciptaan *Java* pada waktu itu adalah menjadi suatu program yang berukuran kecil, efisien, dan *portable* di segala jenis *hardware* Tujuan yang sama ini membuat *Java* menjadi satu bahasa yang ideal untuk mendistribusikan program-program yang dapat dijalankan melalui *www* dan juga suatu bahasa pemrograman untuk segala tujuan untuk mengembangkan program- program yang dapat digunakan dengan mudah dan *portable* di berbagai *platform* yang berbeda. Sekarang, *Sun* telah mengeluarkan berbagai program *Java* yang dapat digunakan seperti *Java API,* atau *JDK* atau *Java Developer Kit*. Selain itu, banyak juga program-program lain yang dapat digunakan untuk membuat

program *Java*, seperti *Eclipse*, *NetBeans, JBuilder, JCreator, J++*, dan sebagainya.

2.2.4 Android Studio

Android Studio merupakan lingkungan pengembangan android baru berdasarkan *IntelliJ IDEA*. Mirip dengan *eclipse* dengan *ADT Plugin*, *android studio* menyediakan alat pengembang terintegrasi untuk pengembangan dan *debugging.*

Android studio menawarkan :

1. Berbasis *Gradle* membangun dukungan.
2. *Android-spesifik refactoring* dan perbaikan yang cepat.
3. Alat Lint untuk menangkap kinerja, kegunaan, versi kompatibilitas dan masalah lainnya.
4. ProGuard dan app-signature.
5. *Wizard* untuk design dan membuat komponen-komponen umum
6. Sebuah *layout editor* yang memungkinkan untuk *drag-and-drop* komponen UI, pratinjau layout pada beberapa konfigurasi layar, dan banyak lagi.

Built-in dukungan untuk *Google Cloud platform*, sehingga mudah untuk mengintegrasikan *Google Cloud Messaging* dan *App Engine* sebagai komponen *server-side*.

2.2.5 Keamanan Data

Secara umum data dikategorikan menjadi dua, yaitu data yang bersifat rahasia dan data yang tidak bersifat rahasia.Data yang tidak bersifat rahasia biasanya tidak akan terlalu diperhatikan.Yang sangat perlu diperhatikan adalah data yang bersifat rahasia, dimana setiap informasi yang ada didalamnya akan sangat berharga bagi pihak yang membutuhkan karena data tersebut dapat dengan mudah digandakan.Untuk mendapatkan informasi didalamnya, biasanya dilakukan berbagai cara yang tidak sah.

Keamanan data biasanya terkait hal-hal berikut:

a. Fisik**,** dalam hal ini pihak yang tidak berwenang terhadap data berusaha mendapatkan data dengan melakukan kegiatan sabotase atau penghancuran tempat penyimpanan data.
b. Organisasi, dalam hal ini pihak yang tidak berwenang untuk mendapatkan data melalui kelalaian atau kebocoran anggota yang menangani data tersebut.
c. Ancaman dari luar, dalam hal ini pihak yang tidak mendapatkan data melalui media komunikasi dan juga melakukan pencurian data yang tersimpan di dalam komputer.

2.2.6 Kriptografi

Kriptografi merupakan ilmu sekaligus seni untuk menjaga keamanan pesan (*Cryptography is the art and science of keeping messages secure*) selain itu ada pengertian tentang kriptografi yaitu kriptografi merupakan ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan, integritas data, serta otentikasi. Kata "seni" di dalam definisi di atas maksudnya adalah mempunyai cara yang unikuntuk merahasiakan pesan. Kata "*graphy*" di dalam "*cryptography*" itu sendiri sudah menyiratkan sebuah seni.Di dalam sistem kriptografi terdapat 5 bagian yaitu

1. *Plaintext* adalah pesan atau data dalam bentuk aslinya teks yang dapat terbaca. Plaintext adalah masukan bagi algoritma enkripsi.
2. *Secret Key* adalah masukan bagi algoritma enkripsi merupakan nilai yang bebas terhadap teks asli dan menentukan hasil keluaran algoritma enkripsi.
3. *Chipertext* adalah keluaran algoritma enkripsi. *Ciphertext* dapat dianggap sebagai pesan tersembunyi yang akan terlihat acak.
4. Algoritma Enkripsi memiliki 2 masukan teks asli dan kunci rahasia. Algoritma enkripsi melakukan transpormasi terhadap teks asli sehingga menghasilkan teks sandi.

5. Algoritma Dekripsi memiliki 2 masukan yaituteks sandi dan kunci rahasia. Algoritma dekripsi memulihkan kembali teks sandi menjadi teks asli bila kunci rahasia algoritma enkripsi sama dengan algoritma dekripsi.

2.2.7 Tujuan Kriptografi

Ada empat tujuan mendasar dari ilmu kriptografi ini yang juga merupakan aspek keamanan informasi, yaitu:

1. *Convidentiality* (kerahasiaan), yaitu memberikan kerahasiaan pesan dan menyimpan data dengan menyembunyikan data dengan menyembunyikan informasi lewat teknik-teknik enkripsi.
2. *Message intergrity* (integritas data), yaitu memberikan jaminan bahwa dari setiap bagian tidak mengalami perubahan dari saat data dibuat/ dikirim sampai dengan saat data tersebut di buka.
3. *Non-repudiation* (nirpenyangkalan), yang memberikan cara untuk membuktikan bahwa suatu dokumen dating dari setiap seseorang apabila ia mencoba menyangkal memiliki dokumen tersebut.
4. *Auntentication* (autentikasi), yang memberikan dua layanan. Yang pertama mengidentifikasi keaslian dari suatu pesan dan memberikan jaminan keotentikannya. Kedua, untuk menguji identitas seseorang apabila ia akan memasuki sebuah sistem.

2.2.8 Vigenere Cipher

Vigenere Cipher adalah salah satu jenis kriptografi klasik yang pada dasarnya adalah melakukan substitusi chpher abjad majemuk (*polyalphabetic substitution*). Metode ini pertama kali dipublikasikan oleh seorang diplomat (sekaligus seorang kriptologis) Perancis, Blaise de Vigenere pada abad ke 16, tepatnya pada tahun 1586, tetapi sebenarnya Giovan Batista Belaso telah menggambarkannya pertama kali pada tahun 1553 seperti ditulis di dalam bukunya La Cifra del Sig. Metode Vigenere Cipher ini berhasil dipecahkan oleh matematikawan Inggris Charles Babbage dan Kasiski pada pertengahan abad 19. Vigenere cipher ini digunakan oleh tentara konfiderasi pada perang sipil Amerika. Perang sipil akhirnya berhasil dihentikan setelah vigenere cipher

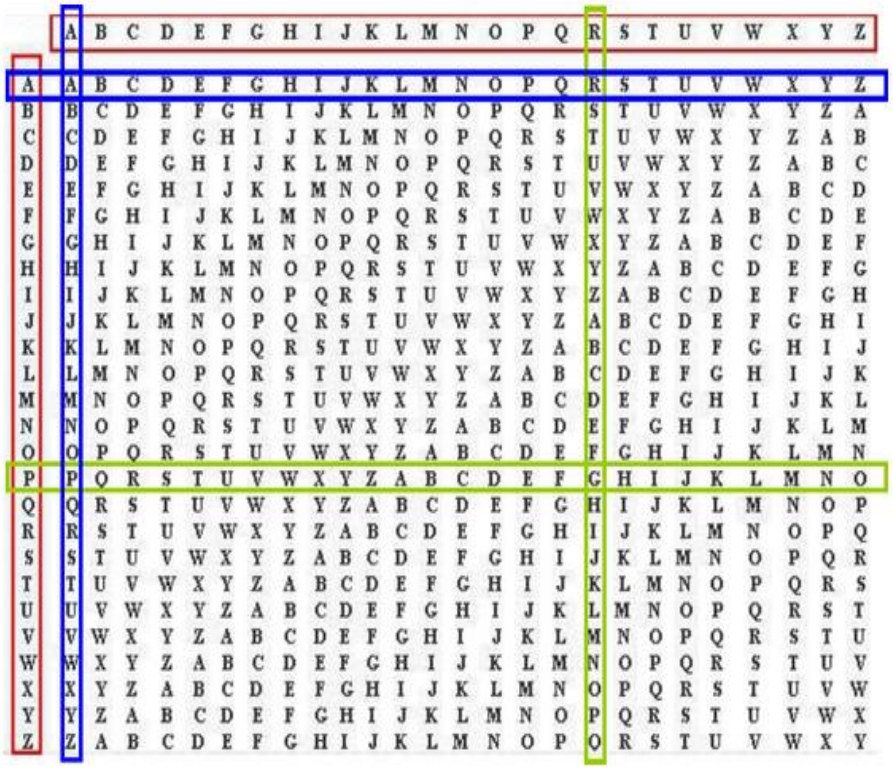
berhasil dipecahkan. Di metode kriptografi klasik Caeasr cipher, setiap huruf alphabet akan disubstitusi sepanjang 3 huruf sesudah huruf tersebut.

Contoh, huruf A akan diganti dengan huruf D, B akan diganti dengan huruf E, Y akan diganti dengan huruf B dengan metode Caesar cipher. Vigenere Cipher ini menerapkan prinsip Caesar Cipher dalam pengenkripsiannya.

Cara Enkripsi dan Deskripsi Vigenere Cipher:

Kita bisa melakukan proses enkripsi dan deskripsi vigenere cipher secara manual menggunakan 2 cara, yaitu

1. Cara Enkripsi dan deskripsi vigenere cipher menggunakan tabula recta



Gambar 2.1 Bujur Sangkar Vigenere

Misalnya teks terang yang hendak disandikan adalah perintah"PANGERAN", sedangkan kata kunci antara pengirim dan tujuan adalah"RASA". Huruf pertama pada plainteks P, disandikan dengan menggunakanbaris berjudul R, huruf pertama pada kata kunci. Pada baris R dan kolom P ditabel Vigenère, terdapat huruf G. Demikian pula untuk huruf kedua,digunakan huruf yang terletak pada baris A (huruf kedua kata kunci)

dankolom A (huruf kedua plainteks), yaitu huruf A. Proses ini dijalankan terussehingga akan didapatkan : Plainteks : PANGERAN

Kunci : RASARASA Cipherteks : GAFGVRSN.

Proses sebaliknya (disebut dekripsi), dilakukan :

1. Tempatkan alphabet kunci pada sisi kiri tabel berdasarkan baris.
2. Telusuri sepanjang baris tersebut hingga ditemukan alphabet ciphertext.
3. Index kolom lokasi alphabet ciphertext berada merupakan alphabet plaintext

Berdasarkan contoh diatas, diketahui :

Ciphertext : GAFGVRSN

Key : RASARASA

2. Cara enkripsi dan deskripsi vigenere cipher menggunakan penjumlahan index

Misalkan kita memiliki teks yang ingin kita enkripsi “PANGERAN”, metode yang akan kita pakai adalah dengan menggunakan penjumlahan indeks plain text dan index key.

Sebagai contoh key yang kita pakai adalah “RASARASA”.

Plaintext	P	A	N	G	E	R	A	N
Key	R	A	S	A	R	A	S	A

Proses berikutnya, kita ubah key nya jadi index hurufnya, yang nantinya akan kita tambahkan ke index plaintexnya (A=0 sampai dengan Z=25)

Plaintext	P	A	N	G	E	R	A	N
Index Plaintext	15	0	13	6	4	17	0	13
Key	R	A	S	A	R	A	S	A
Index Key	17	0	18	0	17	0	18	0
(Plaintext+Key)mod 26	6	0	5	6	21	17	18	13
Cipher text	G	A	F	G	V	R	S	N

Sedangkan untuk deskripsinya, kita tinggal membalikkan saja :

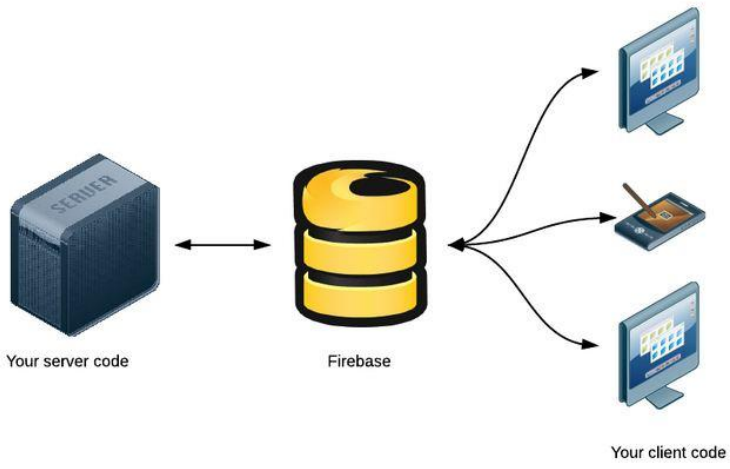
Cipher text	G	A	F	G	V	R	S	N
Index Plaintext	6	0	5	6	21	17	18	13
Key	R	A	S	A	R	A	S	A
Index Key	17	0	18	0	17	0	18	0
(Plaintext+Key)mod 26	15	0	13	6	4	17	0	13
Cipher text	P	A	N	G	E	R	A	N

2.2.9 Firebase

Firebase adalah BaaS (*Backend as a Service*) yang saat ini dimiliki oleh Google. *Firebase* ini merupakan solusi yang ditawarkan oleh Google untuk mempermudah pekerjaan *Mobile Apps Developer*. Dengan adanya *Firebase*, apps developer bisa fokus mengembangkan aplikasi tanpa harus memberikan effort yang besar untuk urusan *backend*.

Dua fitur yang menarik adalah *Firebase Remote Config* dan *Firebase Real Time Database*. Secara sederhananya, *Remote Config* adalah fitur yang memungkinkan developer mengganti / mengubah beberapa konfigurasi aplikasi Android / iOS tanpa harus memberikan update aplikasi via Play Store / App Store. Salah satu konfigurasi yang bisa dimanipulasi adalah seperti warna / tema aplikasi.

Sedangkan *Firebase Real Time Database* adalah fitur yang memberikan sebuah NoSQL database yang bisa diakses secara *real time* oleh pengguna aplikasi. Dan hebatnya adalah aplikasi bisa menyimpan data secara lokal ketika tidak ada akses internet, kemudian melakukan sync data segera setelah mendapatkan akses internet.



Gambar 2.2 Sistem Kerja Fire Base

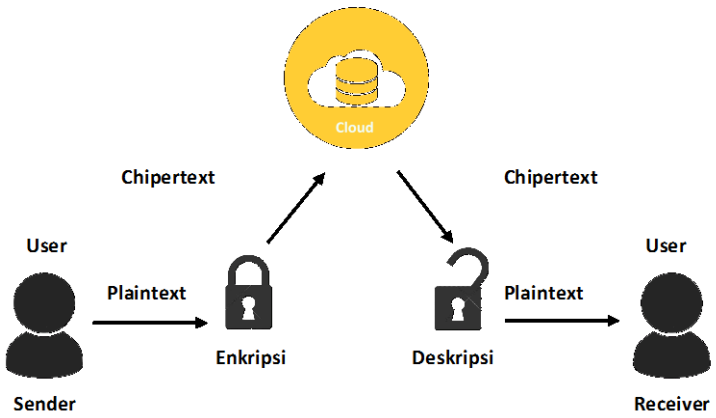
2.2.10 Fiddler

Fiddler merupakan web debugging proxy yang dapat memeriksa semua lalu lintas HTTP dalam format *userfriendly*, serta tes debug dan merekam web dengan menggunakan *fiddler*. *Fiddler* juga memasukkan fasilitas *event-based scripting subsystem* yang powerfull dan dapat diperkaya menggunakan bahasa pemograman .NET. *Fiddler* adalah perangkat lunak gratis dan dapat digunakan untuk men-*debug* lalu lintas dari hampir semua aplikasi yng mendukung proxy, termasuk internet explorer, google chrome, apple safari, mozilla firefox, opera dan lainnya. Fiddler juga bisa digunakan untuk men-*debug* lalu lintas dari perangkat populer seperti wndows phone, ipod/ipad, dan lainnya.

BAB III
ANALISIS DAN PERANCANGAN

3.1 Deskripsi Umum Aplikasi

Pada aplikasi chat menggunakan algoritma vigenere cipher, pengirim (*sender)* melakukan kirim pesan, pesan yang berupa *plaintext* (pesan asli) akan dienkripsi oleh aplikasi, kemudian pesan yangsudah terenkripsi akan dikirim ke database (*firebase).* Selanjutnya apabila penerima akan membaca pesan yang dikirim oleh pengirim, penerima harus menekan dan tahan pesan tersebut, aplikasi akan menampilkan *plaintext* (pesan asli), namun apabila pesan tidak ditahan, pesan akan kembali terenkripsi.

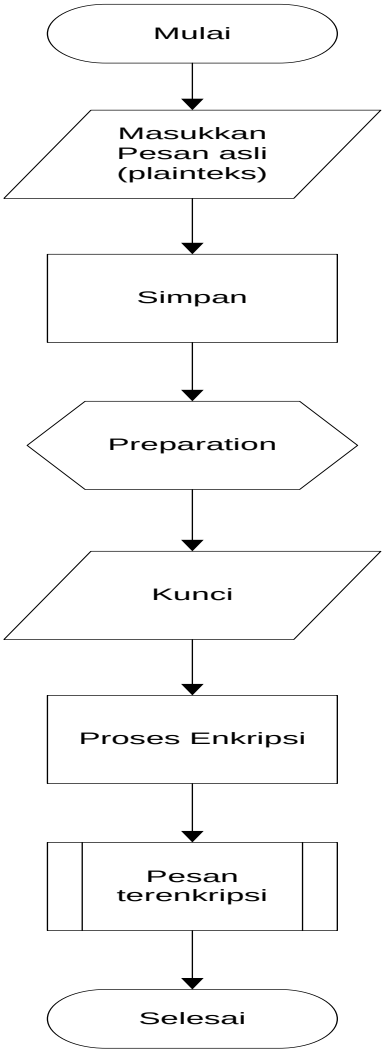


Gambar 3. 1 Deskripsi Umum Aplikasi Chat

3.2 Flowchart

Flowchart adalah suatu metode untuk menggambarkan tahap-tahap pemecahan masalah dengan mempresentasikan simbol-simbol tertentu yang mudah dimengerti, mudah digunakan dan standar.

3.2.1 Flowchart Proses Enkripsi

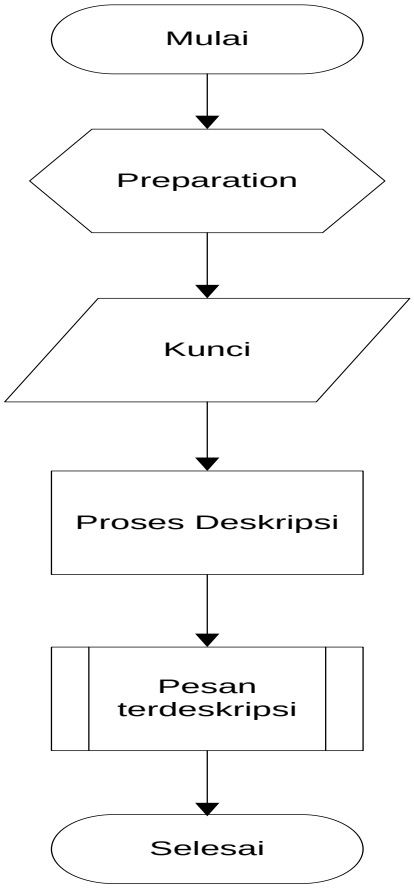


Gambar 3. 2 Flowchart Proses Enkripsi

Penjelasan algoritma dari flowchart proses enkripsi sebagai berikut:

1. Masukkan pesan teks.
2. Simpan kedalam database.
3. Proses mempersiapkan penyimpanan sebagai tempat pengolahan data.
4. Masukkan kunci untuk mengenkripsi.
5. Proses enkripsi dalam database.
6. Data dalam database sudah terenkripsi.
7. Data tersimpan ke database.

3.2.2 Flowchart Proses Deskripsi



Gambar 3. 3 Flowchart Proses Deskripsi

Penjelasan algoritma dari flowchart proses enkripsi sebagai berikut:

1. Proses mempersiapkan penyimpanan sebagai tempat pengolahan pesan.
2. Masukkan kunci untuk mengembalikan pesan asli.
3. Proses pengubahan kedalam pesan asli (*plaintext).*
4. Pesan sudah terdeskripsi

3.3 Analisis Kebutuhan

Pada kebutuhan sistem dapat kita bagi menjadi dua bagian, yaitu kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan fungsional adalah dekripsi tentang aktifitas atau layanan yang harus disedikan oleh sistem.

Sedangkan kebutuhan non fungsional adalah dekripsi tentang fitur, karakteristik, dan batasan lainnya yang menentukan apakah sistem itu memuaskan atau tidak.

3.3.1 Kebutuhan Fungsional

Dari permasalahan umum dan permasalahan khusus yang telah dirumuskan sebelumnya, maka dapat dijabarkan kebutuhan fungsional dan non fungsional pada aplikasi ini.

Tabel 3.1 Kebutuhan Fungsional.

ID Fungsi	Keterangan
F-001	*User* dapat melakukan percakapandengan *user* lain.
F-002	Aplikasi menampilkan pesan percakapan berupa teks.
F-003	Aplikasi dapat mengirim pesan dan menerima pesan dalam bentuk teks.

3.3.2 Kebutuhan Non Fungsional

Tabel 3.2 Kebutuhan Non Fungsional

ID Fungsi	Keterangan
NF-001	Judul "Aplikasi Chatting Menggunakan Algoritma Vigenere Cipher Berbasis Android".
NF-002	*Background* warna orange.
NF-003	Keamanan data pesan teks menggunakan algoritma vigenere Chiper.

3.4 Spesifikasi Kebutuhan Pengguna

Spesifikasi perangkat keras (hardware) dan perangkat lunak (software) yang dibutuhkan dalam pembuatan aplikasi chat menggunakan algoritma vigenere chipper adalah :

Tabel 3.3 Spesifikasi Kebutuhan Pengguna

Hardware	Software
Intel Core i3	Android Studio 1.5.1 build
RAM 2 GB	Android SDK (Software Development Kit)
HDD 500 GB	

3.5 Spesifikasi Kebutuhan Android

Android yang digunakan penulis untuk uji coba aplikasi chat menggunakan algoritma vigenere chiper, pada *android Jelly Bean* ini adalah Samsung GT-I8262 yang mempunyai spesifikasi sebagai berikut :

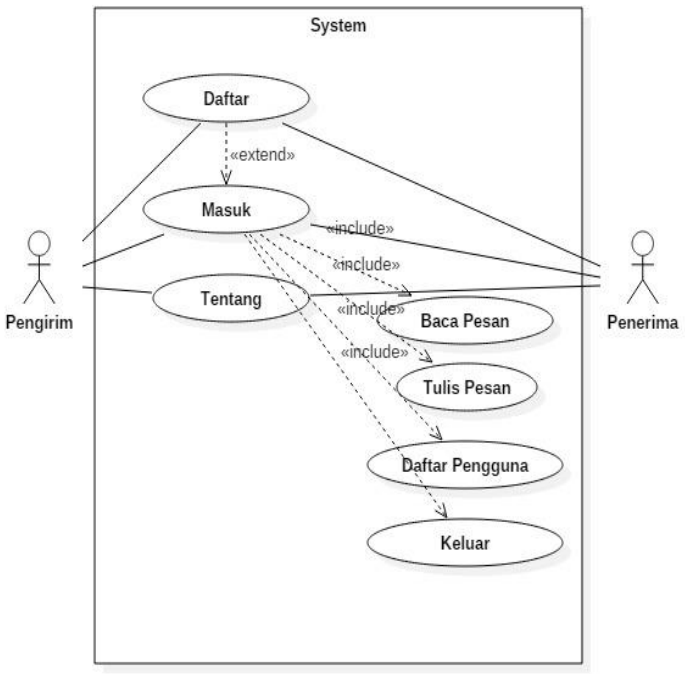
Tabel 3.4 Spesifikasi Kebutuhan Andorid

Jenis Spesifikasi	Keterangan
Processor	Snapgragon MSM8225 S4 Play CPU Dual Core 1.2 Ghz
Memory	1 GB
Platform	Android OS, v4.1.2 (*Jelly Bean*)

3.6 Use case diagram

Use case diagram menjelaskan apa yang akan dilakukan oleh sistem yang akan dibangun dan siapa yang berinteraksi dengan sistem

Dibawah ini adalah gambar use case diagram :



Gambar 3.4 Use case diagram.

Pada aplikasi ini yang pertama kali dilakukan oleh pengirim dan penerima adalah memilih menu daftar untuk melakukan chatting, jika *user* sudah terdaftar maka user langsung memilih menu masuk dengan mengisi nama pengguna dan kata sandi untuk masuk ke dalam sistem. Kemudian *user* dapat melihat daftar pengguna dan menu keluar. Daftar pengguna digunakan untuk melihat pengguna dan melakukan chatting.

3.7 Skenario Use Case

Setelah menjelaskan use case pada bahasan sebelumnya, maka berikut ini akan dijelaskan spesikasi use case yang telah ditemukan. Adapun tahapan-tahapan skenario use case Aplikasi Chatting Menggunakan Algoritma Vigenere Cipher Berbasis Android yang sedang berjalan adalah sebagai berikut.

1. Tabel Skenario Daftar

Tabel 3.5 Skenario Daftar.

Nama Use Case	Daftar
Deskripsi	Use case ini digunakan*user*untuk mendaftarkan user baru ke dalam database.
Kondisi Awal	*User* belum terdaftar sebagai user didatabase.
Kondisi Akhir	*User* telah melakukan pendaftaran dan terdaftar di database.
Skenario	1. *User* memilih menu daftar. 2. *User* memasukkan nama pengguna & kata sandi. 3. Sistem melakukan validasi data. 4. Aplikasi kembali ketampilan masuk.

2. Tabel Skenario Masuk

Tabel 3.6 Skenario Masuk

Nama Use Case	Masuk
Deskripsi	Use case ini digunakan *user* untuk masuk ke halaman utama aplikasi.
Kondisi Awal	*User* berada diluar aplikasi.
Kondisi Akhir	*User* masuk ke dalam halaman utama aplikasi.
Skenario	1. *User* memasukkan nama pengguna dan kata sandi dan menekan tombol masuk. 2. Sistem melakukan validasi nama pengguna dan kata sandi. 3. Sistem memasukkan *user* ke sistem dan aplikasi menampilkan halaman utama.
Skenario Alternatif	1. a. Jika terjadi kesalahan dalam memasukkan nama pengguna atau kata sandi maka kembali ke langkah awal.

3.8 Activitiy diagram

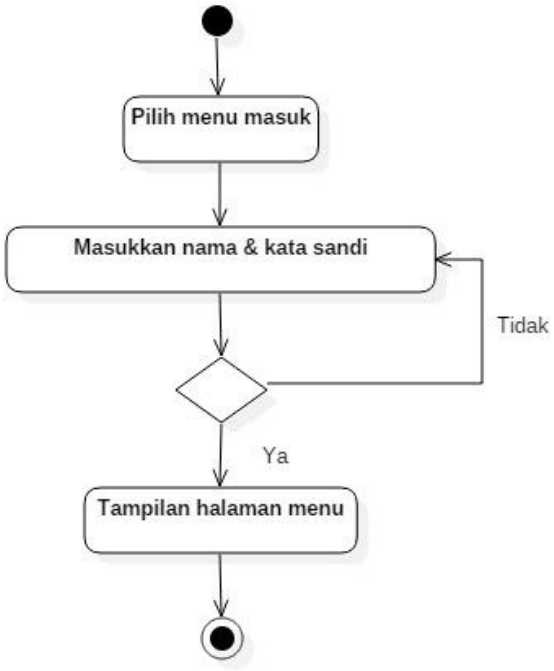
Activity diagram memodelkan alur kerja (*work flow*) sebuah urutan aktivitas pada suatu proses. Diagram ini sangat mirip dengan flow chart karena kita dapat memodelkan proses logika, proses bisnis dan alur kerja. Perbedaan utamanya adalah flow chart dibuat untuk menggambarkanalur kerja dari sebuah sistem, sedangkan activity diagram dibuat untuk menggambarkan aktivitas aktor. Berikut akan digambarkan satu persatu activity diagram untuk masing-masing use case :

1. Activity Diagram Daftar

Gambar 3.5 Activity Diagram Daftar.

Activity diagram ini menjelaskan proses *user* melakukan daftar pada aplikasi. *User* meng-klik daftar ditampilan utama aplikasi, lalu *user* mengisi data *user* sepertinama pengguna dan kata sandi, kemudian setelah data diinput, aplikas akan melakukan proses validasi. Jika salah satu dari daftar tidak diisi, maka aplikasi akan menampilkan data tidak boleh kosong. Jika daftar sudah lengkap, maka daftar berhasil.

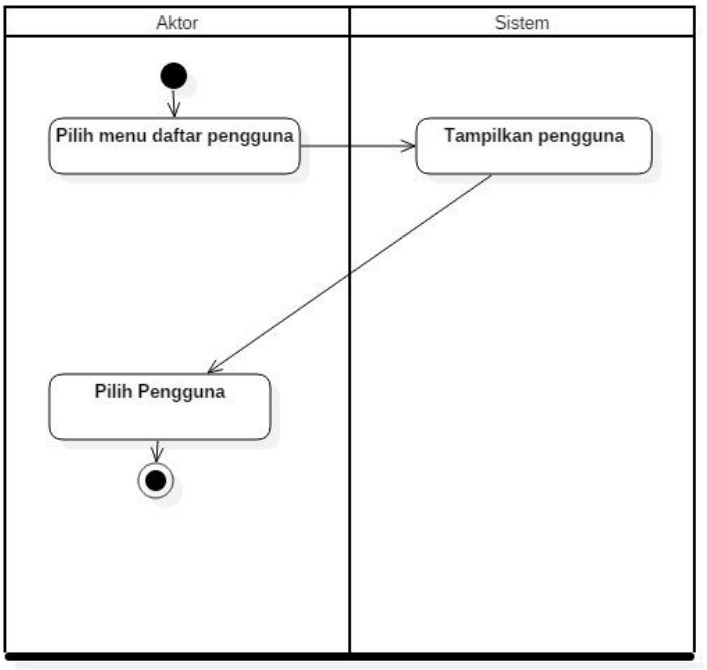
2. Activity Diagram Masuk



Gambar 3.6 Activity Diagram Masuk.

Pada activity diagram ini *user* memilih menu masuk, kemudian sistem akan menampilkan form untuk masuk kedalam aplikasi, user memasukkan nam pengguna dan kata sandi. Sistem akan melakukan cek validasi username dan password di database, jika nama dan kata sandi benar maka akan masuk ke halaman utama aplikasi, namun jika nama dan kata sandi salah maka *user* harus memasukkan lagi nama dan kata sandi dengan benar.

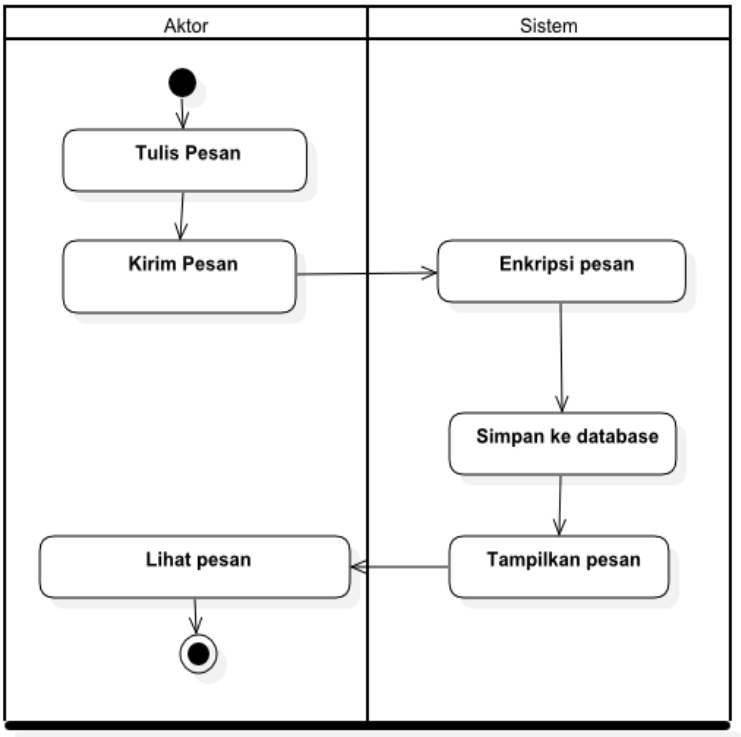
3. Activity Diagram Daftar Pengguna



Gambar 3.7 Activity Diagram Daftar Pengguna.

Pada activity diagram ini *user* memilih menu daftar pengguna, kemudian sistem akan menampilkan nama-nama pengguna yang terdaftar, kemudian *user* akan memilih nama pengguna untuk melakukan *chatting*.

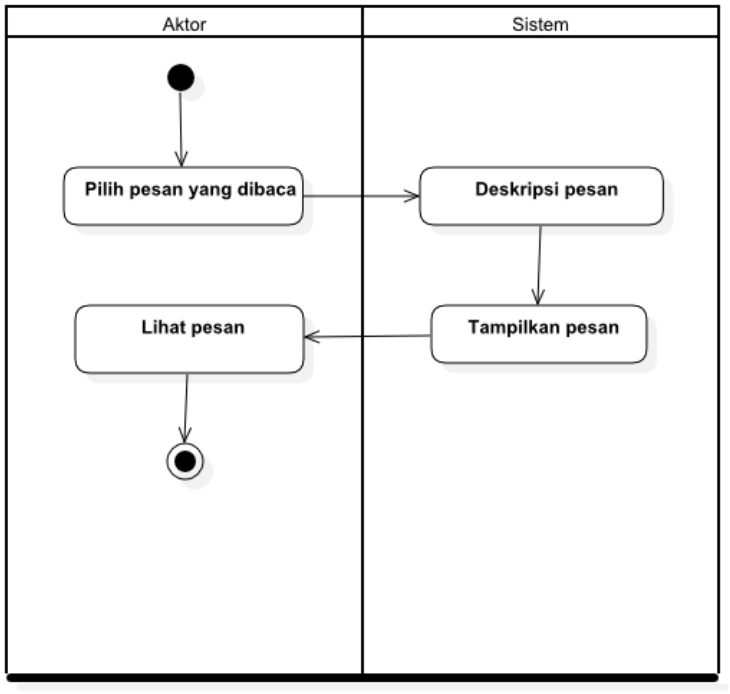
4. Activity Diagram Tulis Pesan



Gambar 3.8 Activity Diagram Tulis Pesan.

Pada activity diagram ini *user* pertama melakukan tulis pesan dan kirim pesan, kemudian sistem melakukan enkripsi pesan dan disimpan ke *database* selanjutnya menampilkan pesan, selanjutnya *user* melihat pesan.

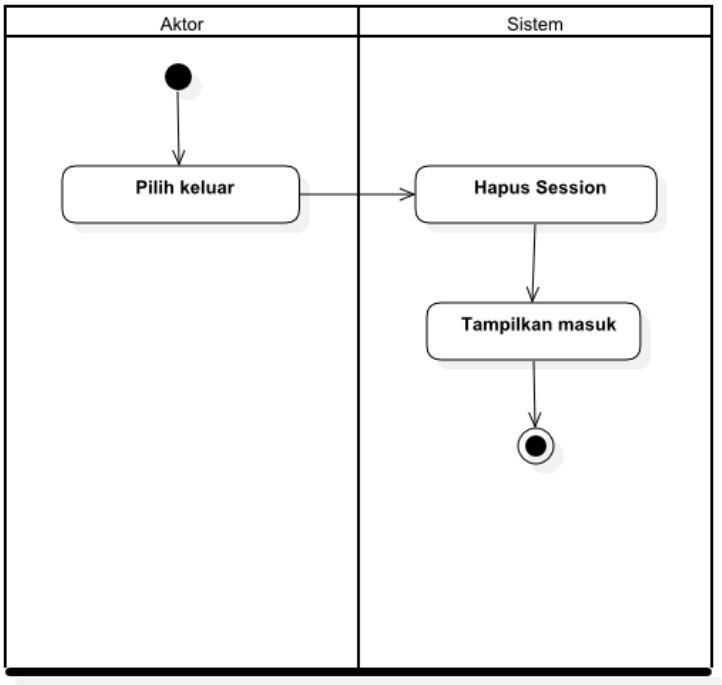
5. Activity Diagram Baca Pesan



Gambar 3.9 Activity Diagram Baca Pesan.

Pada activity diagram ini *user* melakukan pilih pesan yang ingin dibaca, kemudian sistem akan mendeskripsikan pesan dan tampilkan pesan, selanjutnya *user* akan melihat pesan dalam bentuk *plaintext* (pesan asli).

6. Activity Diagram Keluar



Gambar 3.10 Activity Diagram Keluar.

Pada activity diagram ini menjelaskan proses *user* melakukan pilih menu keluar, kemudian sistem akan mengapus *session*, selanjutnya sistem akan menampilkan kembali menu masuk aplikasi.

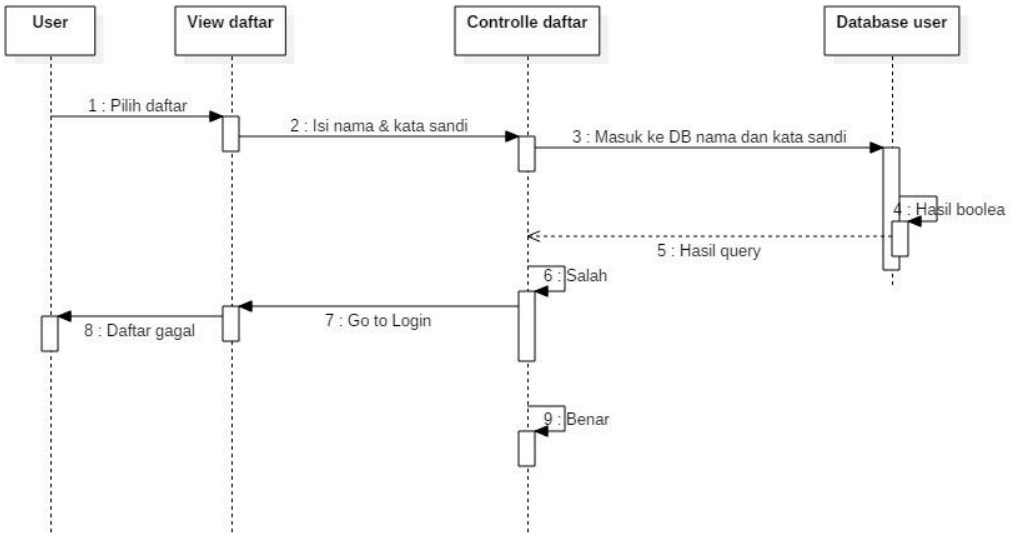
3.9 Sequence diagram

Sequence diagram menggambarkan interaksi antar objek didalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa message yang digambarkan terhadap waktu.

Berikut ini adalah gambaran sequence diagram dalam aplikasi kriptografi chatting menggunakan algoritma vigenere cipher:

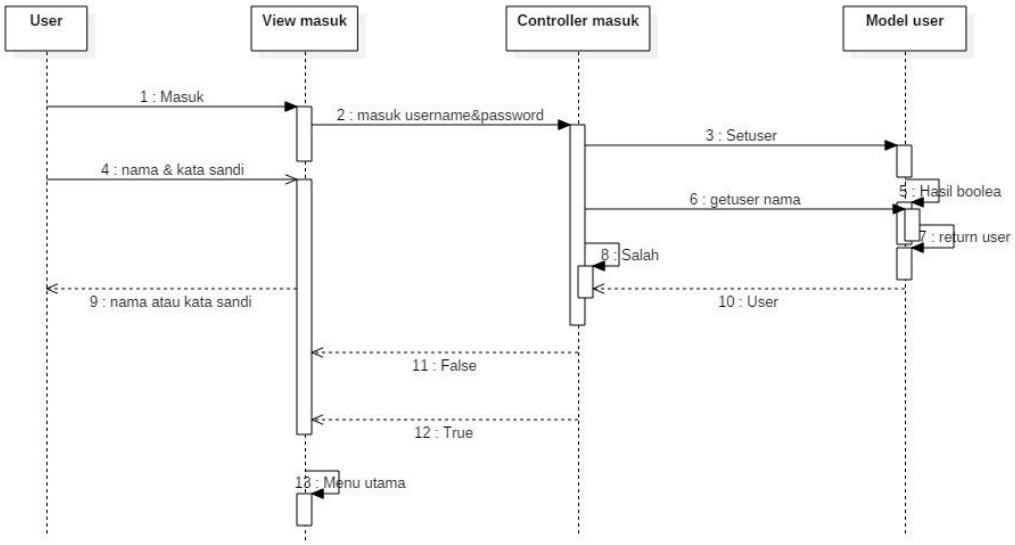
1. Sequence diagram daftar

Pada sequence diagram ini *user* memilih menu daftar, kemudian sistem akanmenampilkan form untuk daftar, pada form tersebut *user* menginput data *user* seperti namadan kata sandi, kemudian setelah menginput data, *user* memilih button daftar dan data tersebut akan tersimpan di database.



Gambar 3.11 Sequence Diagram Daftar.

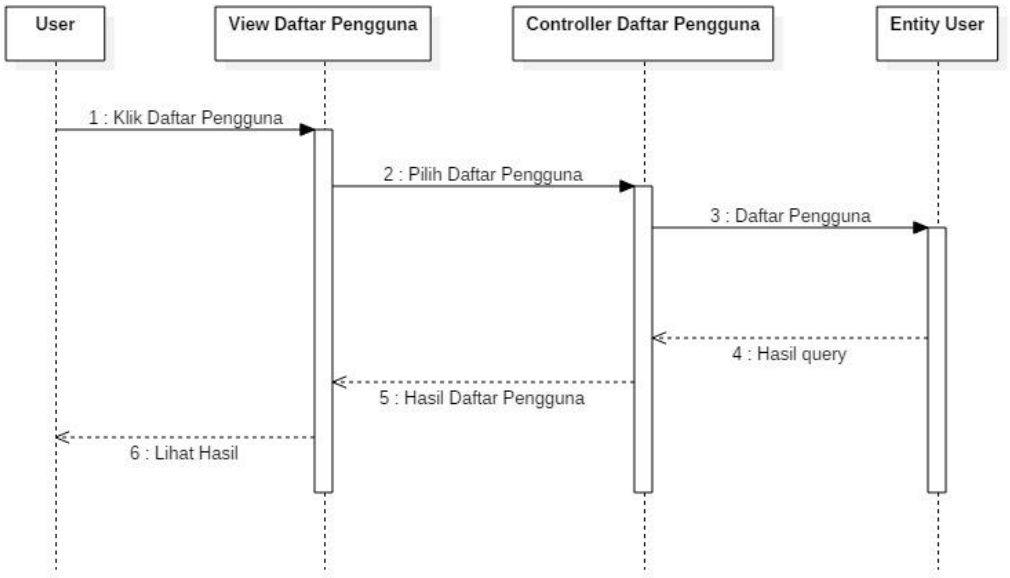
2. Sequence diagram masuk



Gambar 3.12 Sequence Diagram Masuk.

Pada sequence diagram ini dijelaskan bagaimana proses masuk *user* untuk masuk ke halaman utama aplikasi. Pertama aplikasi akan menampilkan form masuk. Lalu, *user* memasukkan nama dan kata sandi, kemudian aplikasi akan melakukan proses validasi di *database.* Jika namadan kata sandi benar, maka tampilan menu utama akan muncul. Jika tidak, maka aplikasi menampilkan pesan gagal.

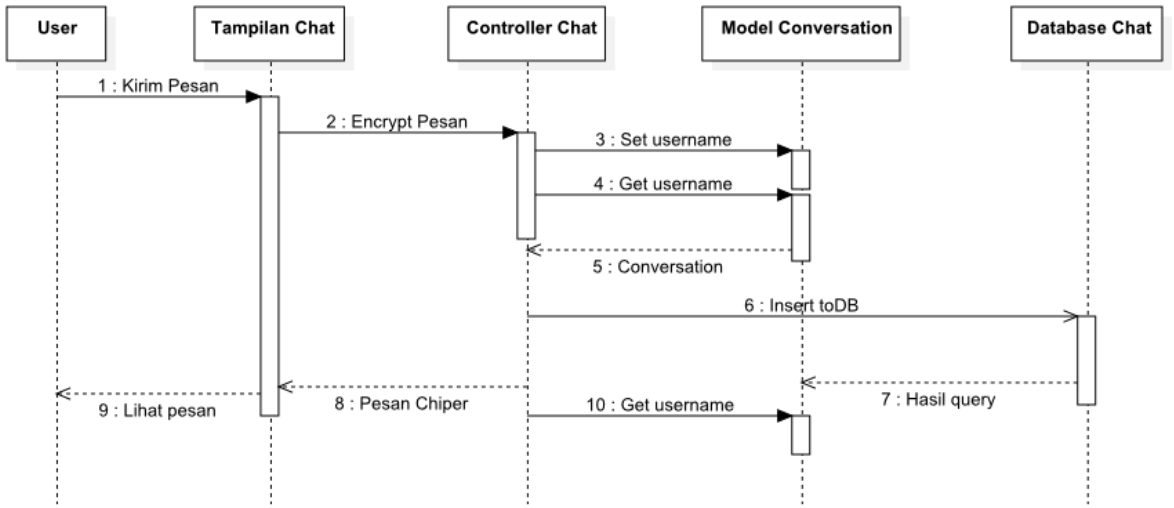
3. Sequence Diagram Daftar Pengguna



Gambar 3.13 Sequence Diagram Daftar Pengguna

Pada sequence diagram ini menjelaskan proses *user* memilih daftar pengguna untuk melakukan *chatting. User* masuk ke dalam menu daftar pengguna, kemudan *user* memilih teman yang sudah terdaftar sebelumnya di aplikasi chat ini.

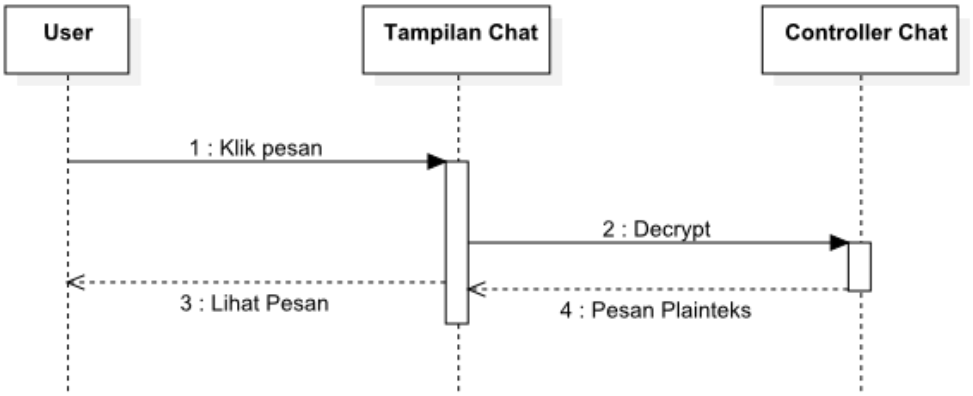
4. Sequence diagram Kirim Pesan.



Gambar 3.14 Sequence Diagram Kirim Pesan .

Pada sequence diagram ini menjelaskan proses *user* melakukan proses kirim pesan, setelah pesan dikirim oleh *user*, aplikasi akan mengenkripsi pesan yang dikirim oleh *user*, kemudian pesan tersebut akan dimasukkan kedalam database, selanjutnya database akan mengirim hasil *query* ke aplikasi dan dari aplikasi pesan akan ditampilkan dalam bentuk *plaintext* (pesan asli).

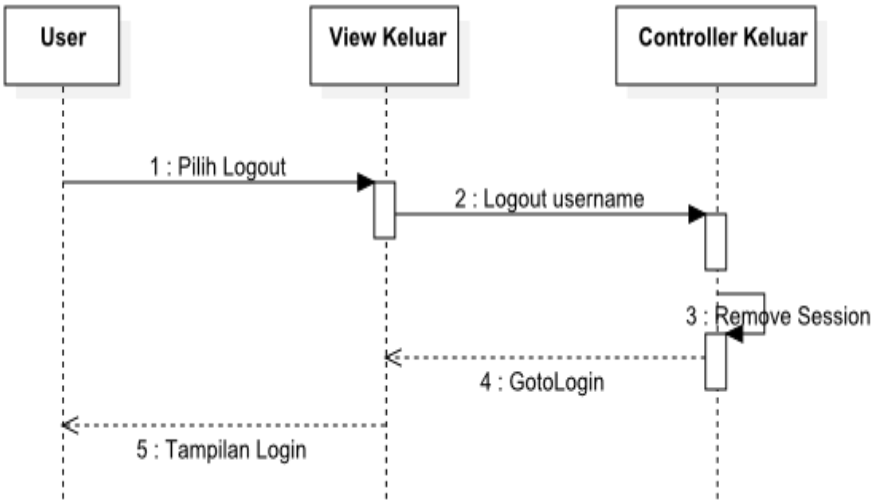
5. Sequence diagram Baca Pesan.



Gambar 3.15 Sequence Diagram Baca Pesan .

Pada sequence diagram ini menjelaskan proses *user* melakukan proses baca pesan, *user* mengklik pesan, kemudian aplikasi akan mendeskripsikan pesan. Aplikasi akan menampilkan pesan dalam bentuk *plaintext*, *user* melihat pesan dalam bentuk *plaintext* (pesan asli).

6. Sequence diagram Keluar

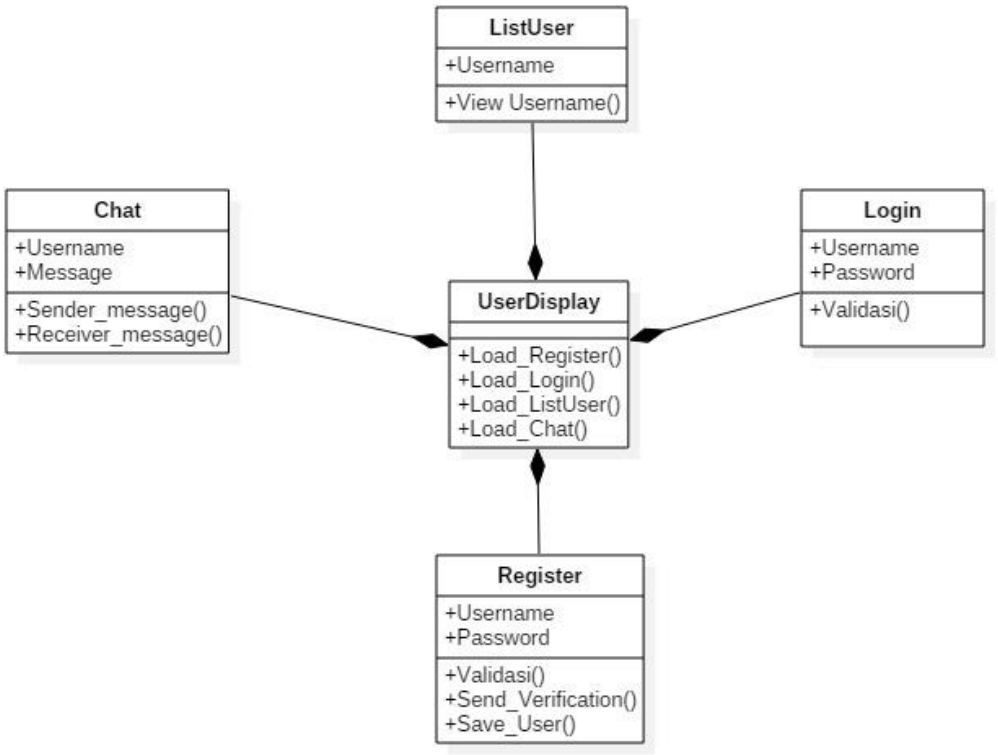


Gambar 3.16 Sequence Diagram Keluar.

Pada sequence diagram ini *user* memilih menu keluar untuk keluar dari aplikasi chatting dan menampilkan tampilan masuk kepada *user.*

4.0 Class Diagram

Class Diagram digunakan untuk menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).



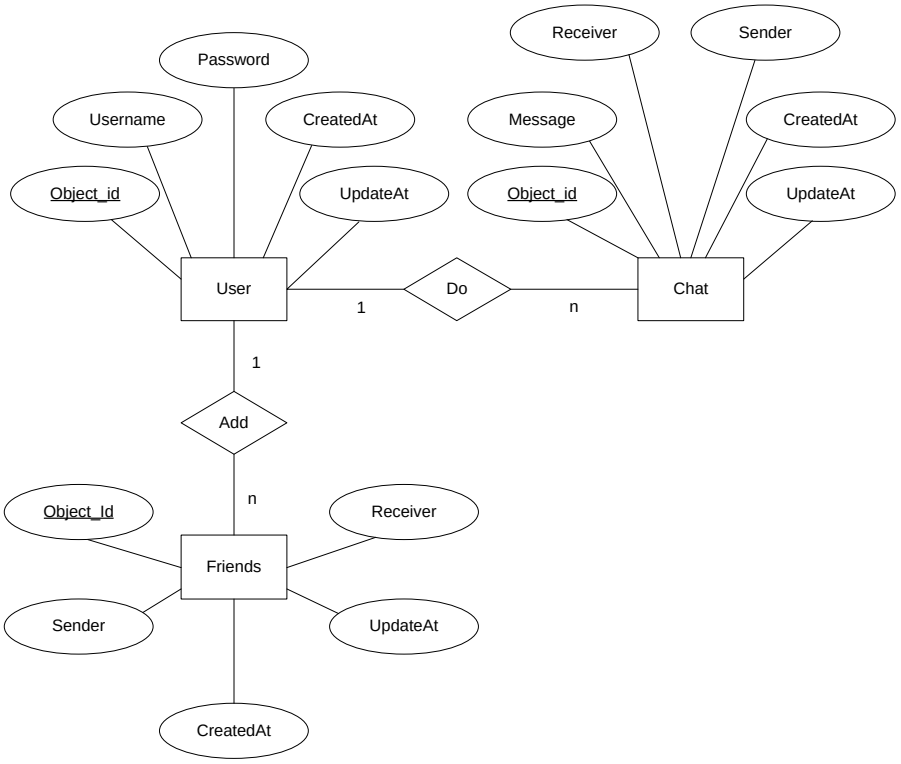
Gambar 3.17 Class Diagram

Pada aplikasi ini, terdapat 5 kelas yang mempunyai kegunaan masing-masing. Deskripsi kelas-kelas tersebut antara lain :

1. Kelas UserDisplay merupakan kelas untuk menampilkan antarmuka login,register, list friend, add friend, chat
2. Kelas Login merupakan kelas untuk login ke aplikasi chat.
3. Kelas Register merupakan kelas untuk mendaftar ke akun aplikasi chat Vigenere Cipher.
4. Kelas ListUser merupakan kelas untuk melihat daftar teman.
5. Kelas Chat merupakan kelas untuk mengirim pesan

4.1 ER Diagram

ER Diagram digunakan untuk menjelaskan hubungan-hubungan antar data-data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan yang dihubungkan oleh suatu relasi.



Gambar 3.18 ER Diagram

Pada aplikasi ini, terdapat 3 tabel yang mempunyai kegunaan masing-masing. Deskripsi tabel-tabel tersebut antara lain:

1. Tabel user mempunyai 5 atribut, yaitu object_id, yaitu object_id, username, password, createdAt, dan updatedAt. Object_id berarti id dari user. Username berarti username dari pengirim dan penerima.
2. Tabel friend mempunyai 5 atribut, yaitu object_id, sender, createddAt, updatedAt, sendername, reciever, recievername. Object_id yang berarti id dari friend. Sender berarti nama dari pengirim, receiver nama dari penerima.
3. Tabel chat mempunyai 5 atribut, yaitu object_id, username, password, fullname, dan online. Object_id berarti id baik dari pengirim dan penerima

pesan. *Username* berarti nama pengguna dari pengirim dan penerima pesan. *Password* berarti kata sandi dari pengirim dan penerima pesan. Name berarti nama dari pengirim dan penerima pesan.

4.2 Perancangan Antar Muka

1. Rancangan Antarmuka Halaman Aplikasi

Gambar 3.19 Rancangan Antarmuka Halaman Aplikasi

Tabel 3.7 Deskripsi Antarmuka Halaman Aplikasi

Jenis	Ikon / Nama	Keterangan
Tombol	Masuk	Tombol digunakan untuk login ke halaman utama aplikasi
Tombol	Daftar	Tombol digunakan untuk registrasi user baru
Teks field	Nama Pengguna	Teks field untuk memasukkan nama pengguna
Teks field	Kata Sandi	Teks field untuk memasukkan kata sandi

2. Rancangan Antarmuka Daftar

3.20 Rancangan Antar Muka Daftar

Tabel 3.8 Deskripsi Antarmuka Daftar

Jenis	Nama	Keterangan
Teks field	Nama Pengguna	Teks field untuk memasukkan nama pengguna
Teks field	Kata Sandi	Teks field untuk memasukkan kata sandi
Tombol	Daftar	Tombol daftar digunakan untuk menyimpan data user ke dalam database

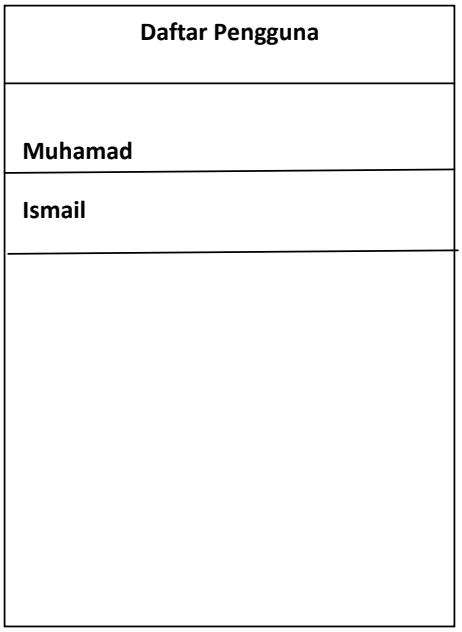
3. Rancangan Antarmuka Masuk

3.21 Rancangan Antar Muka Masuk

Tabel 3.9 Deskripsi Antarmuka Masuk

Jenis	Nama	Keterangan
Teks field	Nama Pengguna	Teks field untuk memasukkan nama pengguna
Teks field	Kata Sandi	Teks field untuk memasukkan kata sandi
Tombol	Masuk	Tombol login digunakan untuk meneruskan nama pengguna dan kata sandi ke dalam proses validasi

4. Rancangan Antarmuka Daftar Pengguna



3.22 Rancangan Antarmuka Daftar Pengguna

Tabel 3.10 Deskripsi Antarmuka Daftar Pengguna

Jenis	Nama	Keterangan
Teks field	online	Teks field menandakan seseorang sedang online.
Teks field	offline	Teks field menandakan seseorang sedang offline.

5. Rancangan Antarmuka Chat

3.23 Rancangan Antarmuka Chat

Tabel 3.11 Deskripsi Antarmuka Chat

Jenis	Nama	Keterangan
Teks field	Isi chat	Teks field untuk melihat percakapan
Teks field	Isi pesan	Teks field untuk memasukkan pesan
Tombol	Send	Tombol update digunakan untuk mengirim pesan

BAB IV
IMPLEMENTASI & PENGUJIAN

4.1 Implementasi Class Diagram

Berdasarkan dokumen perancangan yang telah dilakukan maka hasil implementasi dari antarmuka yang dibuat secara detail dapat dilihat pada tabel berikut.

Tabel 4.1 Implementasi Class Diagram

No	Antarmuka	Nama File Fisik	Nama File Executable
1	Masuk	Masuk.xml	Masuk.xml
2	Daftar	Daftar.xml	Daftar.xml
3	Daftar Pengguna	Daftar_pengguna.xml	Daftar_pengguna.xml
4	Tentang	Tentang.xml	Tentang.xml
5	Keluar	Keluar.xml	Keluar.xml

Pada tahap desain dan tahap implementasi tetap terdapat lima antarmuka yaitu daftar, masuk, daftar pengguna, tentang, dan keluar.

4.2 Implementasi Basis Data

1. Tabel Basis Data User

Nama Tabel : User

Primary Key : Object_id

Foreign key : -

Fungsi : Tabel untuk memasukkan data user

Tabel 4.2 Basis Data User

No	Kolom	Tipe	Panjang	Null	Index
1	Object_id	Int	10	No	Primary
2	Username	Varchar	20	No	
3	Password	Varchar	50	No	
4	Created_at	Timestamp		No	
5	Updated_at	Timestamp		No	

2. Tabel Basis Data User Friend

Nama Tabel : Friend

Primary Key : Object_id

Foreign key : -

Fungsi : Tabel untuk data friend

Tabel 4.3 Basis Data Friend

No	Kolom	Tipe	Panjang	Null	Index
1	Object_id	Int	20	No	Primary
2	Sender	String	50	No	
3	Reciever	String	50	No	
4	Created_at	Timestamp		No	
5	Updated_at	Timestamp		No	

3. Tabel Basis Data Chat

Nama Tabel : Chat

Primary Key : Object_id

Foreign key : -

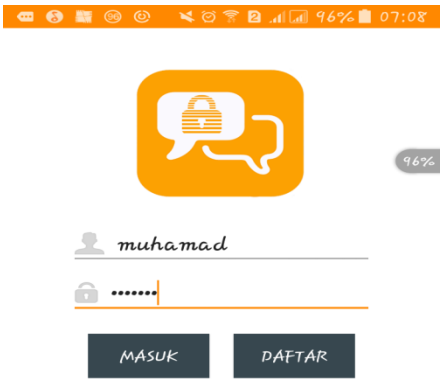
Fungsi : Tabel untuk melihat isi chat

Tabel 4.4 Basis Data Chat

No	Kolom	Tipe	Panjang	Null	Index
1	Object_id	Int	20	No	Primary
2	Message	String	50	No	
3	Sendername	String	50	No	
4	Created_at	Timestamp		No	
5	Updated_at	Timestamp		No	

4.3 Implementasi Antarmuka

1. Form masuk

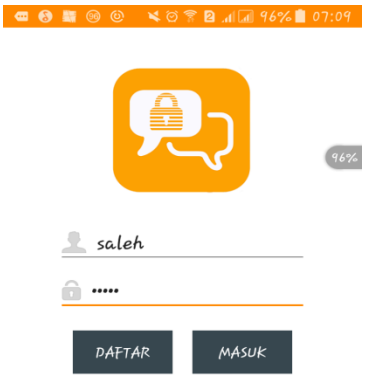


Gambar 4.1 Antarmuka Masuk

Tabel 4.5 Deskripsi Antarmuka Proses Masuk

Jenis	Nama	Gambar	Keterangan
Teks	Nama pengguna		Teks field untuk memasukkan nama pengguna.
Teks	Kata sandi		Teks field untuk memasukkan kata sandi.
Button	Login	MASUK	Tombol Masuk digunakan untuk meneruskan nama pengguna dan kata sandi kedalam proses validasi.

2. Form daftar

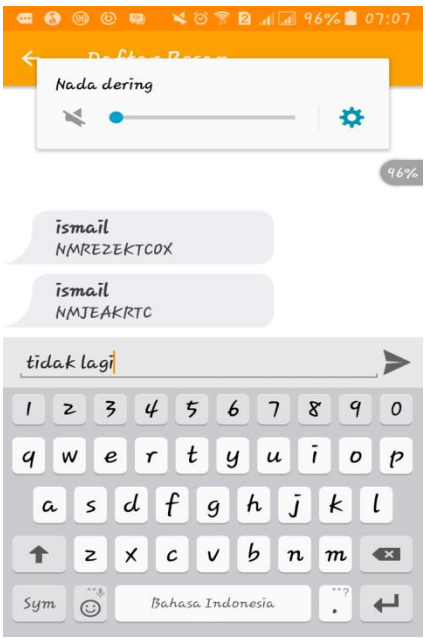


Gambar 4.2 Antarmuka Daftar

Tabel 4.6 Antarmuka Daftar

Jenis	Nama	Gambar	Keterangan
Teks	Nama Pengguna		Teks field untuk memasukkan nama pengguna.
Teks	Kata sandi		Teks field untuk memasukkan kata sandi.
Button	Daftar		Tombol Daftar digunakan untuk meneruskan kedalam proses validasi.

3. Isi pesan

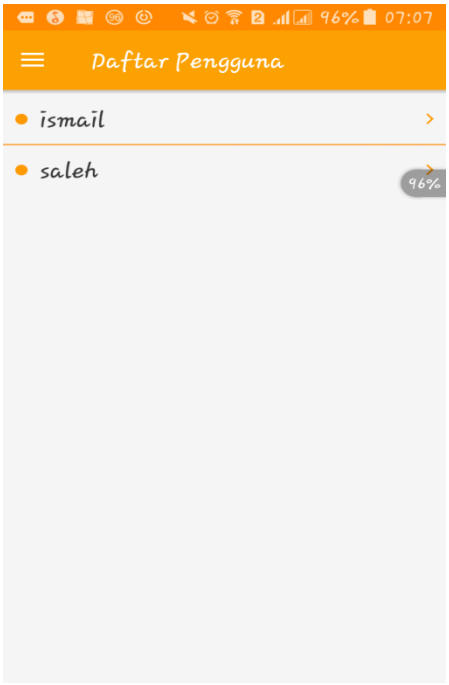


Gambar 4.3 Isi pesan

Tabel 4.7 Isi Pesan

Jenis	Nama	Gambar	Keterangan
Teks	Isi teks pengirim		Teks field isi teks yang akan dikirim.
Teks	Isi teks penerima		Teks field teks yang diterima.
Button	Kirim		Tombol kirim.

4. Daftar Pengguna



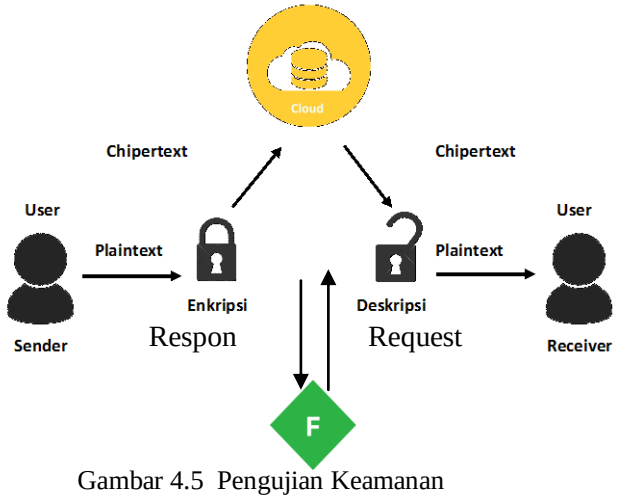
Gambar 4.4 Daftar Pengguna

Tabel 4.8 Daftar Pengguna

Jenis	Nama	Gambar	Keterangan
Teks	Daftar Pengguna		Teks field isi teks yang akan ditambah.

4.4 Pengujian Keamanan

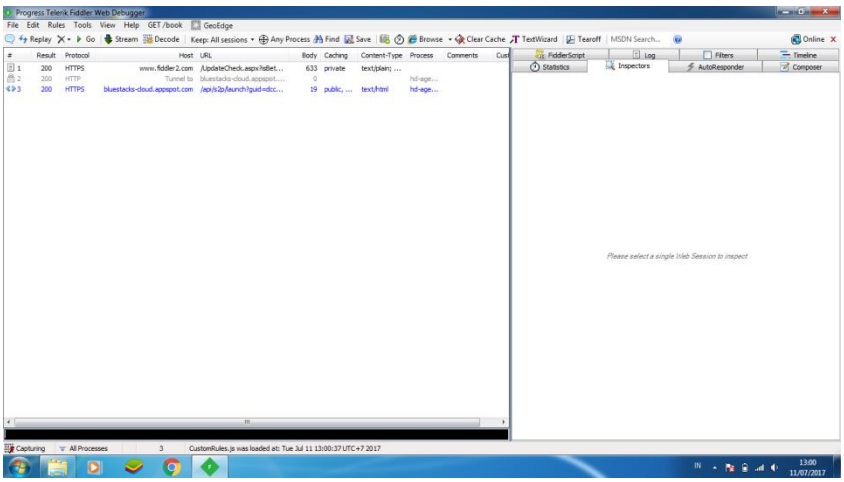
Pengujian ini dilakukan untuk mengetahui apakah pesan yang dikirimkan sudah dalam bentuk terenkripsi atau bentuk pesan yang mudah dibaca. Pengujian ini dilakukan dengan menggunakan bantuan perangkat lunak *wireshark* untuk menangkap paket data yang melewati sebuah jaringan. Pengujian dengan mengirimkan sebuah pesan melalui aplikasi chat *vigenere cipher* dan *wireshark* akan menangkap paket data yang dikirimkan selama proses pesan tersebut dikirim.



Gambar 4.5 Pengujian Keamanan

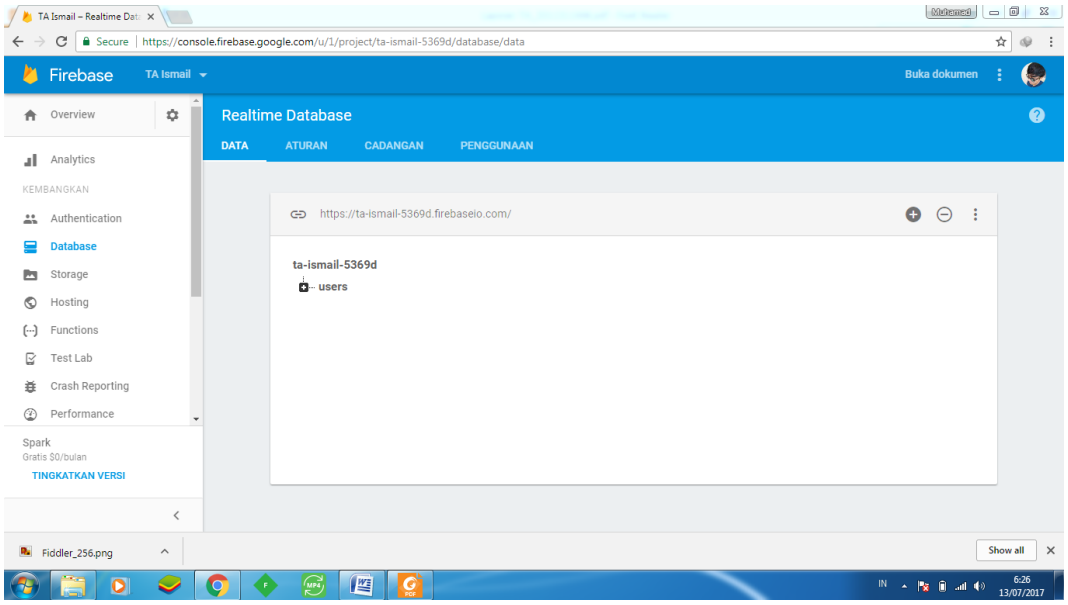
Langkah-Langkah pengujian menggunakan *Fiddler 4.*

1. Halaman awal *fiddler.*



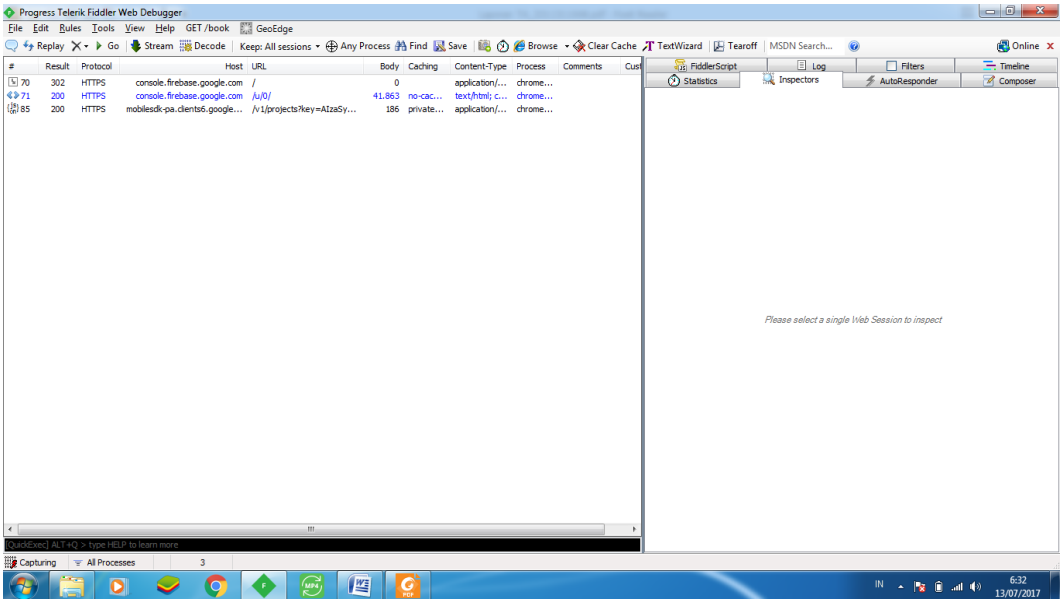
Gambar 4.6 Halaman Awal *Fiddler.*

2. Buka google chrome dan ketik url https://console.firebase.google.com/.



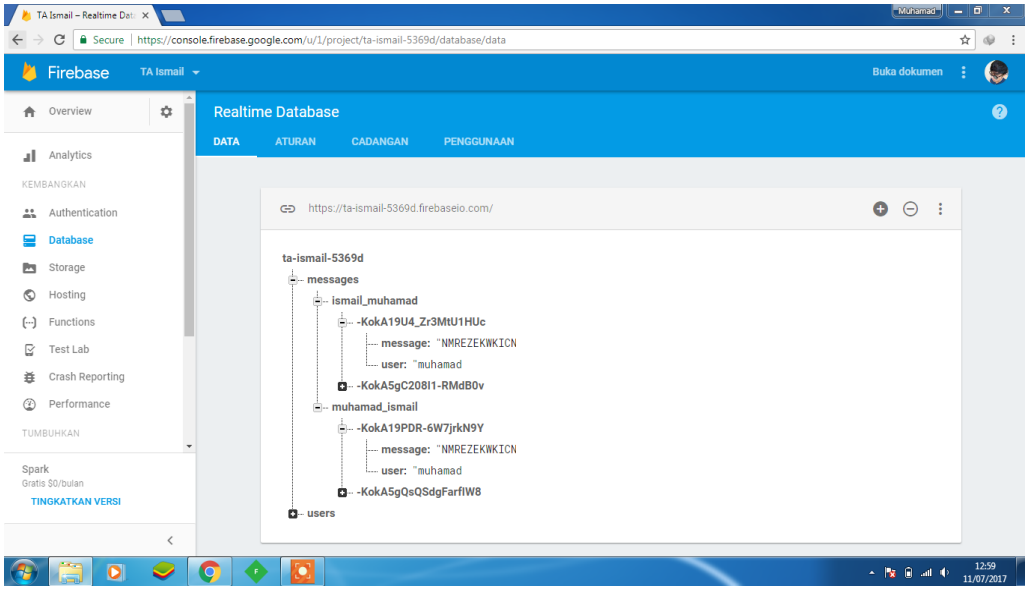
Gambar 4.7 Halaman Awal *Firebase.*

3. Buka kembali aplikasi *fiddler*, pada halaman ini *fiddler* akan menangkap aktivitas traffic langsung ke server.



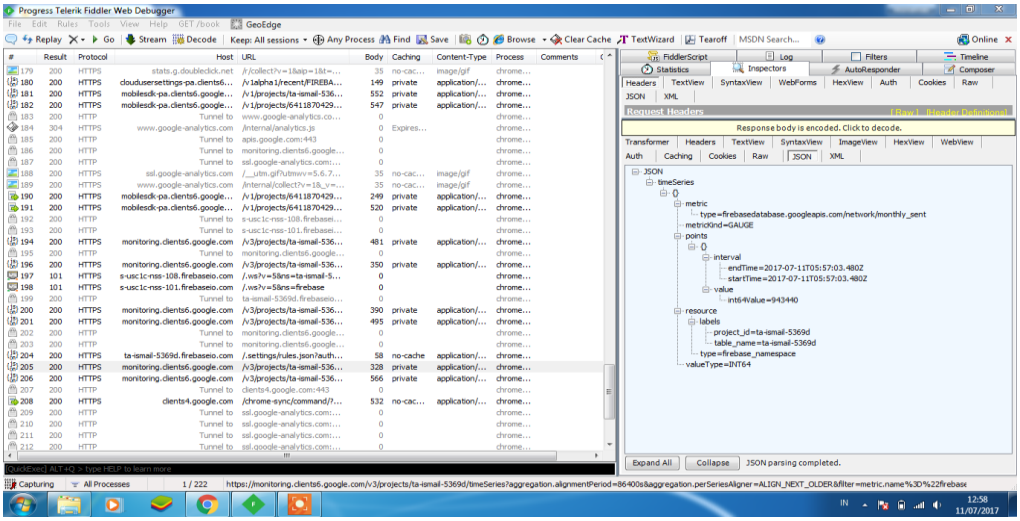
Gambar 4.8 Halaman Aktivitas *Fiddler.*

4. Buka kembali google chrome, kemudin login pada *firebase*, pilih menu database untuk melihat tabel dari pesan tersebut.



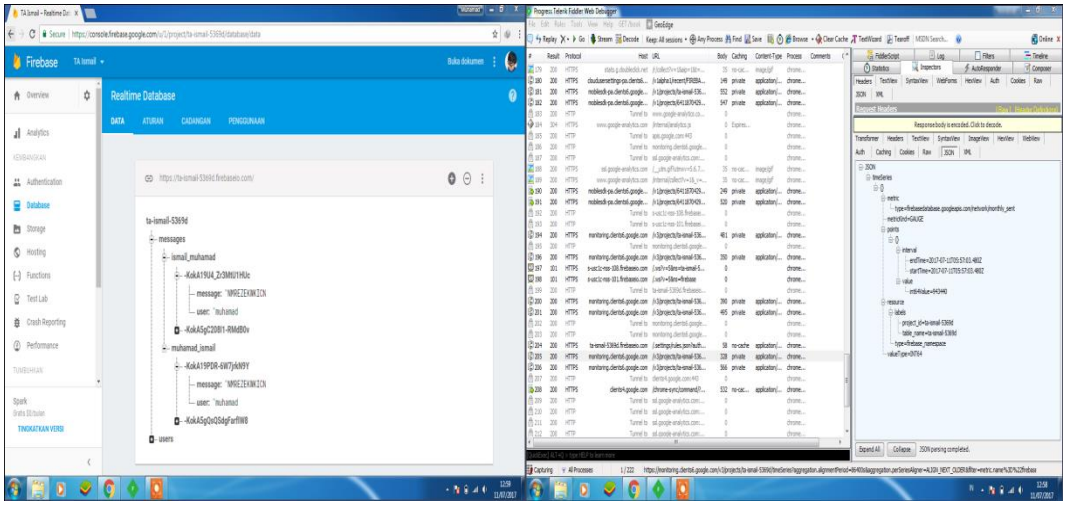
Gambar 4.9 Halaman Tabel Chat Database.

5. Kembali ke aplikasi *fiddler* dan lihat di list web session, list tersebut adalah aktivitas dari request halaman website. Kemudian pilih menu *inspectors* selanjutnya pilih *JSON*, menampilkan respon dari isi table chat database *firebase*.



Gambar 4.10 Halaman *Fiddler* Setelah Mendapat Respon.

6. Dari kedua tampilan pada gambar dibawah ini menjelaskan, data atau pesan yang dikirim oleh pengguna aplikasi chat menggunakan algoritma metode *vigenere cipher* berbasis *android*, sudah terenkripsi menggunakan algoritma *vigenere cipher*. Dibuktikan dengan pengujian menggunakan aplikasi fiddler.



Gambar 4.11 Halaman *Firebase* Dan *Fiddler*.

6.1 Hasil Pengujian

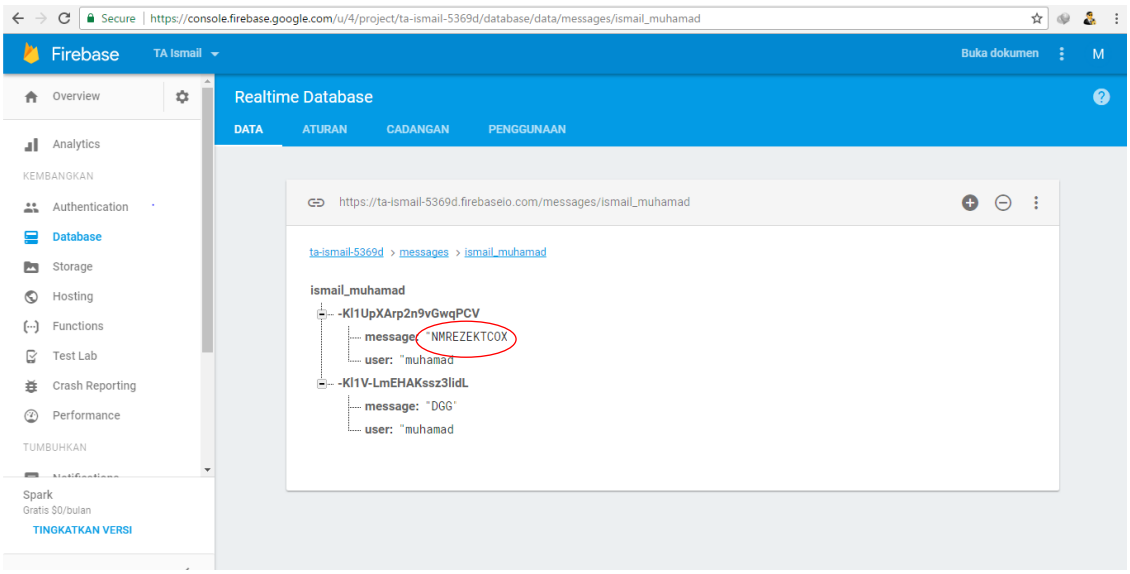
Hasil pengujian disajikan dalam bentuk tabel seperti pada tabel berikut ;

Tabel 4.9 Hasil Pengujian Aplikasi Chat Vigenere Cipher

No	Use case	Fungsi	Skenario	Data Uji	Target	Pengujian	
						Benar	Tidak
1	Daftar	Pendaftaran akun	1. Mengisi Nama dan Kata sandi	**Data Benar** Nama Pengguna : ismail Kata sandi: ismail **Data Salah** Nama Lengkap : ismail Kata sandi :	Berhasil menyimpan ke database. -Memberikan pesan kesalahan invalid daftar atau masih ada kolom yang kosong.	√	
2	Masuk	Autentikasi	1. Nama dan Kata sandi yang telah ada dalam database. 2. Menekan tombol Masuk.	**Data Benar** Nama Pengguna : ismail Kata sandi: ismail **Data Salah** Nama Pengguna: ismail Kata sandi :	Berhasil menyimpan ke database. -Memberikan pesan kesalahan invalid daftar atau data tidak boleh kosong.	√	
3	Daftar Pengguna	Memulai obrolan	User memilih nama teman yang ingin dikirimi pesan.	User mengetikan isi pesan dan klik kirim pada tombol kirim.	Pesan berhasil dikirim	√	
4	Tentang	Melihat Tentang Aplikasi.	User memilih menu tentang .	User menekan tombol tentang.	Tentang berhasil ditampilkan.	√	
5	Keluar	Keluar.	User memilih menu keluar.	User menekan tombol keluar.	Kembali pada tampilan awal, atau isikan kembali nama pengguna dan kata sandi jika ingin masuk kembali	√	

6.2 Pengujian Keamanan di database

Pengujian ini dilakukan untuk mengetahui apakah pesan yang dikirimkan sudah dalam bentuk terenkripsi atau bentuk pesan yang mudah dibaca atau plainteks, jika aplikasi mengimplementasikan enkripsi. Pengujian ini dapat dilihat didalam database.



Gambar 4.12 Pengujian Keamanan Di Database

BAB V
KESIMPULAN & SARAN

5.1 Kesimpulan

Berdasarkan tahapan analisis, perancangan serta implementasi pada aplikasi chatting ini, maka dapat diambil kesimpulan:

1. Aplikasi yang dibuat sudah mampu memenuhi kebutuhan sebagai aplikasi chat yang menerapkan metode algoritma *vigenere cipher*, dimana isi pesan dapat terenkripsi dan dapat dideskripsikan.
2. Hasil pengujian terhadap metode *vigenere cipher* untuk proses enkripsi pesan teks menunjukan bahwa perangkat lunak tersebut secara fungsional mengeluarkan hasil yang sesuai dengan tujuan aplikasi.
3. Dengan melakukan pengujian keamanan menggunakan *software wireshark*, pesan yang terkirim aman dan benar-benar sudah terenkipsi menggunakan algoritma *vigenere cipher*.

5.2 Saran

Saran-saran yang dianggap dapat menyempurnakan pengembangan aplikasi ini adalah:

1. Aplikasi dapat ditambahkan fitur-fitur yang memudahkan pengguna saat mengoperasikan aplikasi, merancang aplikasi agar lebih *user friendly*, dan lebih mengoptimalkan penggunaan aplikasi.
2. Untuk tahap pengembangan selanjutnya disarankan agar algoritma pengenkripsian yang digunakan kedepannya pada fitur chat adalah metode pengenkripsian modern seperti DES atau RSA yang lebih powerful.

DAFTAR PUSTAKA

1. Safaat H, Nazruddin. (2012).*Pemograman Aplikasi Mobile Smartphone dan Table PC Berbasis Android*. Bandung: Informatika Bandung.
2. Sadikin, R. (2012). *Kriptografi untuk Keamanan Jaringan*. Yogyakarta: Andi.
3. Fitri Apriyani. 2014. ”Aplikasi Catting dengan sistem enkripsi menggunakan algoritma blowfish berbasis android”. Skripsi. Yogyakarta: Sekolah Tinggi Manajemen Informatika dan Komputer Amikom.
4. Widodo Joko, T.S. 2014. “Implementasi Algoritma Kriptografi AES 128 Bit Sebagai Pengaman SMS Pada Smartphone Berbasis Android” Skripsi. Yogyakarta: Sekolah Tinggi Manajemen Informatika dan Komputer Amikom.
5. Primartha, Rifkie. (2011). Penerapan Enkripsi dan Deskripsi File Menggunakan Algoritma Data Encryption Standard (DES), 17 halaman. Tersedia:http://ejournal.unsri.ac.id/index.php/jsi/index. (Diakses 28 September 2015)
6. Pranarelza, Randy. 2013. Implementasi Algoritma Rijndael Untuk Enkripsi Dan Deskripsi Pesan SMS Pada Smartphone Berbasis Android. Journal Pendidikan. STMIK EL RAHMA : Yogyakarta.
7. Massandy, Danang.T. 2011. Studi dan Implementasi Cryptography Package Pada Sistem Operasi Android. Journal Pendidikan. Sekolah Teknik Elektro dan Informatika : Bandung