PEMBUATAN APLIKASI ENCODE DAN DECODE BERBASIS WEB MENGGUNAKAN ALGORITMA BASE64 UNTUK KONFIRMASI PENGIRIMAN PIN

Anderias Eko Wijaya.[*1], Deni Rahmat G.[#2]

Program Studi Teknik Informatika, STMIK Subang
Jl. Marsinu No. 5 - Subang, Tlp. 0206-417853 Fax. 0206-411873
*E-mail: ekowjy09@yahoo.com[*1], deni_rahmat_g @yahoo.co.id[#2]*

ABSTRAKSI

Keamanan data merupakan hal yang sangat penting dalam menjaga kerahasiaan informasi, terutama yang berisi informasi sensitif yang hanya boleh diketahui isinya oleh pihak tertentu, sehingga perlu dilakukan penyandian data supaya beberapa pihak yang tidak memiliki kewenangan tidak akan dapat membuka informasi yang dikirim. Salah satu cara yang digunakan untuk pengamanan data adalah menggunakan sistem kriptografi yaitu dengan menyediakan isi informasi (plaintext) menjadi isi yang tidak dipahami melalui proses encode (encipher), dan untuk memperoleh kembali informasi yang asli, dilakukan proses decode (decipher), dengan menggunakan kunci yang benar. Cukup banyak algoritma pada kriptografi, salah satu-nya adalah algoritma Base64. Transformasi base64 digunakan untuk Encoding dan Decoding suatu data ke dalam format ASCII, yang didasarkan pada bilangan dasar 64 atau bisa dikatakan sebagai salah satu metoda yang digunakan untuk melakukan encoding terhadap data biner. Aplikasi ini akan menyajikan implementasi dari proses enkripsi dan dekripsi suatu data bersifat text.

Kata kunci: ***Algoritma, kriptografi, base64, encoding, decoding***

1. Pendahuluan

1.1. *Latar Belakang*

Dalam dunia internet yang telah berlangsung dalam skala global, penyampaian pesan adalah suatu hal yang lumrah. Namun adakalanya muncul kekhawatiran jika pesan yang ingin kita sampaikan tersebut jatuh pada pihak-pihak yang tidak kita inginkan. Untuk mencegah hal itu, perlu dilakukan proses penyandian terhadap pesan yang akan kita sampaikan. Proses penyandian ini bisa dilakukan dengan *encode* dan *decode*.

Dalam dunia komputer, encode adalah proses penempatan urutan karakter (huruf, angka, tanda baca, dan symbol tertentu) ke dalam format khusus sehingga menjadi sebuah sandi. Sedangkan decode adalah proses sebaliknya yaitu konversi dari format yang disandikan kembali pada karakter asli.

Banyak cara yang bisa dilakukan dalam proses encode dan decode ini. Diantaranya dengan menggunakan algoritma Base64. Algoritma base64 merupakan algoritma yang menggunakan Block Cipher dan beroperasi pada mode bit. Namun algoritma Base64 ini lebih mudah dalam pengimplementasiannya dari algoritma-algoritma yang lainnya.

Base64 melakukan encoding (penyandian) terhadap data binary menjadi format 6-bit character. Pada algoritma ini, rangkaian bit-bit plainteks dibagi menjadi blok-blok bit dengan panjang yang sama, biasanya 64 bit yang direpresentasikan dengan karakter ASCII. Base64 menggunakan karakter A–Z, a–z dan 0–9 untuk 62 nilai pertama, sedangkan 2 nilai terakhir digunakan symbol (+ dan /).

1.2. Identifikasi Masalah

- Perlu adanya sistem yang dapat menerapkan algoritma Base64 terhadap teknik encode dan decode.
- Perlu adanya aplikasi encode dan decode untuk menguji kemampuan algoritma Base64 dalam proses enkripsi.

1.3. Tujuan

Tujuan yang diperoleh dari penelitian ini adalah merancang dan membangun sebuah perangkat

lunak yang dapat melakukan proses encode dan decode dengan algoritma Base64.

1.4. Manfaat

- Membantu pemahaman mengenai proses encode dan decode.
- Mengetahui cara penerapan algoritma Base64 ke dalam aplikasi berbasis web.
- Melatih kemampuan Penulis untuk lebih menguasai pemrograman berbasis web.
- Sebagai perangkat lunak untuk melakukan teknik encode dan decode.
- Sebagai fasilitas pendukung pembelajaran

1.5. Metodologi Penelitian

Metode penelitian yang akan digunakan dalam pembuatan sistem penentu keputusan ini adalah metode prancangan perangkat lunak *Waterfall*. Pengembangan metode *Waterfall* sendiri melalui beberapa tahapan yaitu:

- Penelitian Lapangan (*Field Research*), Penelitian dilakukan dengan mengumpulkan data dan informasi di lapangan.
- Penelitian Kepustakaan (*Library Research*), Penelitian ini bertujuan untuk mendapatkan data yang bersifat teori seperti mengumpulkan buku-buku atau bahan lainnya.
- Observasi, Observasi yang dilakukan penulis adalah mengamati secara langsung data yang diperoleh.
- Analisis Perangkat Lunak, Kegiatan analisis perangkat lunak meliputi analisis spesifikasi perangkat lunak yang akan digunakan sebagai alat bantu penelitian.
- Perancangan Perangkat Lunak, Perancangan perangkat lunak meliputi perancangan keras dan perancangann antarmuka dari hasil analisis.
- Implementasi Perangkat Lunak, Implementasi dari hasil analisis dan perancangan perangkat lunak.
- Pengujian Perangkat Lunak, Pengujian terhadap perangkat lunak yang telah diimplementasikan.

2. Tinjauan Pustaka

2.1 Pengertian Encode dan Decode

Di bidang kriptografi, encode ialah proses mengamankan suatu informasi dengan membuat informasi tersebut tidak dapat dibaca tanpa bantuan pengetahuan khusus. Dikarenakan encode telah digunakan untuk mengamankan komunikasi di berbagai negara, hanya organisasi-organisasi tertentu dan individu yang memiliki kepentingan yang sangat mendesak akan kerahasiaan yang menggunakan proses encode. Di pertengahan tahun 1970-an, dimanfaatkan untuk pengamanan oleh sekretariat agen pemerintah Amerika Serikat pada domain publik, dan saat ini proses encode telah digunakan pada sistem secara luas, seperti Internet e-commerce, jaringan Telepon bergerak dan ATM pada bank.

> *The term encrypt implies a methode, or system, of secret writing which, generally speaking, is unlimited in scope; it should be possible, using any one given crypter, to transform any plain text whatever, regardless of its length and the language in which it is written, into a cryptogram, or single encrypted message. The process, that of transforming the cryptogram into a plaintext, is called decryptment..*(Helen Fouché Gaines, 1939)

Sedangkan menurut Rinaldi Munir (2004): *"Proses menyandikan plainteks menjadi cipherteks disebut enkripsi (encryption) atau enciphering (standard nama menurut ISO 7498-2*)."“Proses mengembalikan cipherteks menjadi plainteksnya disebut dekripsi (decryption) atau deciphering (standard nama menurut ISO 7498-2).

Sehingga dari beberapa pengertian di atas dapat ditarik kesimpulan bahwa enkripsi (encryption) adalah sebuah proses menjadikan pesan yang dapat dibaca (plaintext) menjadi pesan acak yang tidak dapat dibaca (ciphertext). Berikut adalah contoh enkripsi yang digunakan oleh Julius Caesar, yaitu dengan mengganti masing-masing huruf dengan 3 huruf selanjutnya (disebut juga Additive/Substitution Cipher).

Dalam melakukan proses untuk meng-enkripsi diperlukan sebuah algoritma dan key.

"A key-based algorithm uses an encryption key to encrypt the message. This means that the encrypted message is generated using not only the message, but also using a key". (Helen Fouché Gaines, 1939)

2.2 Tujuan Proses Encode dan Decode

Ada empat tujuan mendasar dari proses encode dan decode yang juga merupakan aspek keamanan informasi yaitu (Bruce Schneier, John Wiley & Sons, 1996) :

a. Kerahasiaan, adalah layanan yang digunakan untuk menjaga isi dari informasi dari siapapun kecuali yang memiliki otoritas atau kunci rahasia untuk membuka/mengupas informasi yang telah disandi.
b. Integritas data, adalah berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain kedalam data yang sebenarnya.
c. Autentikasi, adalah berhubungan dengan identifikasi/pengenalan, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan melalui kanal harus diautentikasi keaslian, isi datanya, waktu pengiriman, dan lain-lain.
d. Non-repudiasi, atau nirpenyangkalan adalah usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman/terciptanya suatu informasi oleh yang mengirimkan/membuat.

2.3 Fungsi Dasar

Proses Encode dan Decode, mempunyai tiga fungsi dasar yaitu (Rinaldi Munir, 2004) :

a. Enkripsi, merupakan hal yang sangat penting dalam kriptografi yang merupakan pengamanan data yang dikirim terjaga kerahasiaannya. Pesan asli disebut (plaintext) yang diubah menjadi kode-kode yang tidak dimengerti.
b. Dekripsi, merupakan kebalikan dari enkripsi, pesan yang telah dienkripsi dikembalikan kebentuk asalnya (plainteks) tersebut dengan dekripsi pesan.

Kunci, yang dimaksud di sini adalah kunci yang dipakai untuk melakukan enkripsi dan dekripsi, kunci terbagi 2 (dua) bagian yaitu kunci pribadi (privacy key) dan kunci umum (public key).

2.4 Istilah-istilah dalam proses Encode dan Decode

Sebelum membahas lebih jauh, berikut ini diberikan beberapa istilah yang umum digunakan dalam proses Encode dan Decode (Rinaldi Munir, 2004) :

a. Pesan, Plaintext dan Ciphertext
   Pesan (message) adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Nama lain untuk pesan adalah plainteks (plaintext/cleartext). Pesan dapat berupa data atau informasi yang dikirim (melalui kurir, saluran telekomunikasi, dan sebagainya) atau disimpan dalam media perekaman (kertas, storage, dan sebagainya). Pesan yang tersimpan tidak hanya berupa teks, tetapi juga dapat berbentuk citra (image), suara/bunyi (audio), dan video, atau berupa berkas biner lainnya. Bentuk pesan yang tersandi disebut cipherteks (ciphertext) atau kriptogram (cryptogram). Cipherteks harus dapat ditransformasikan kembali menjadi bentuk plainteks semula agar pesan yang diterima bisa dibaca.
b. Pengirim (Sender) dan Penerima (Receiver)
   Komunikasi data melibatkan pertukaran pesan antara dua entitas yaitu pengirim dan penerima. Pengirim (sender) adalah entitas yang mengirim pesan kepada entitas lainnya. Penerima (receiver) adalah entitas yang menerima pesan. Entitas disini dapat berupa orang, mesin (komputer), kartu kredit, dan sebagainya. Pengirim tertentu menginginkan pesan dapat dikirim secara aman, yaitu ia yakin bahwa pihak lain tidak dapat membaca isi pesan yang ia kirim. Solusinya adalah dengan cara menyandikan pesan menjadi cipherteks.
c. Enkripsi dan Dekripsi

Proses menyandikan plainteks menjadi cipherteks disebut enkripsi (encryption) atau enciphering (standar nama menurut ISO 7498-2). Sedangkan proses mengembalikan cipherteks menjadi plainteks semula dinamakan dekripsi (decryption) atau deciphering (standar nama menurut ISO 7498-2).

d. Cipher dan Kunci

Algoritma kriptografi disebut juga cipher yaitu aturan untuk enciphering dan deciphering, atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Beberapa cipher memerlukan algoritma yang berbeda untuk enciphering dan deciphering.

Konsep matematis yang mendasari algoritma kriptografi adalah relasi antara 2 (dua) buah himpunan yaitu himpunan yang berisi elemen-elemen plainteks dan himpunan yang berisi elemen-elemen cipherteks. Enkripsi dan dekripsi merupakan fungsi yang memetakan elemen-elemen antara kedua himpunan tersebut. Jika P menyatakan plainteks dan C menyatakan cipherteks, maka fungsi enkripsi E memetakan P ke C,

$$E(P)=C$$

dan fungsi deskripsi D, memetakan C ke P,

$$D(C)=P$$

Karena proses enkripsi kemudian deskripsi mengembalikan pesan ke pesan asal, maka kesamaan berikut harus sama,

$$D(E(P))=P$$

Pada skema enkripsi konvensional atau *symmetric-key*, digunakan sebuah kunci untuk melakukan proses enkripsi dan deskripsinya. Kunci terssebut dinotasikan dengan K,. Sehingga proses kriptografinya adalah sebagai berikut :

$$\text{Enkripsi} : E_k(P) = C$$
$$\text{Deskripsi} : D_k(C) = P \text{ atau } D_k(E(P)) = P$$

Sedangkan pada system *asymmetric-key* digunakan kunci umum (public key) untuk enkripsi dan kunci pribadi (*privacy key*) untuk proses deskripsinya. Sehingga kedua proses tersebut dinyatakan dengan :

$$\text{Enkripsi} : Ep_k(P) = C$$
$$\text{Deskripsi} : D_{sk}(C) = P \text{ atau } D_{sk}(E_{pk}(P)) = P$$

e. Sistem Kriptografi

Kriptografi membentuk sebuah system yang dinamakan dengan system kriptografi. Sistem kriptografi (*cryptosystem*) adalah kumpulan yang terdiri dari algoritma kriptografi, semua plainteks dan cipherteks yang mungkin, dan kunci. Di dalam system kriptografi, cipher hanyalah salah satu komponen saja.

f. Penyadap

Penyadapan (eavesdropper) adalah orang yang mencoba menangkap pesan selama ditransmisikan. Tujuan penyadap adalah untuk mendapatkan informasi sebanyak-banyaknya mengenai sistem kriptografi yang digunakan untuk berkomunikasi dengan maksud untuk memecahkan cipherteks. Nama lain penyadap: enemy, adversary, intruder, interceptor, bad guy. Ron Rivest, seorang pakar kriptografi, menyatakan bahwa,

"cryptography is about communication in the presence of adversaries." (Ron Rivest, 1991)

g. Kriptanalisis atau kriptologi

Kriptologi berkembang sedemikian rupa sehingga melahirkan bidang yang berlawanan yaitu kriptanalisis. Kriptanalisis (*cryptanalisis*) adalah ilmu dan seni untuk memecahkan cipherteks menjadi plainteks tanpa mengetahui kunci yang digunakan. Pelakunya disebut kriptanalis. Jika seseorang kriptografer (*cryptographer*) mentranformasikan plainteks menjadi cipherteks dengan suatu algoritma dan kunci, maka sebaliknya seorang kriptanalis berusaha untuk memecahkan cipherteks tersebut untuk menemukan plainteks atau kunci. Kriptologi (*cryptology*) adalah studi mengenai kriptografi dan kriptanalisis.

2.5 Algoritma Base 64

Base64 sejatinya bukan enkripsi, namun hanyalah sebuah standar penyandian (encoding).

"Base64 is a group of similar encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation. The Base64 term originates from a specific MIME content transfer encoding."

"Base64 encoding schemes are commonly used when there is a need to encode binary data that needs to be stored and transferred over media that are designed to deal with textual data. This is to ensure that the data remains intact without modification during transport. Base64 is commonly used in a number of applications including email via MIME, and storing complex data in XML." (IETF, 2006)

Sejarah Base64 berawal dari surat elektronik (email). Pada waktu itu, email dikirim dengan protokol SMTP (simple mail transfer protocol) ke mail server kita, lalu dikirim ke mailbox orang yang dituju di mail server tujuan. "Protokol" adalah tata cara mesin (komputer) saling berkomunikasi via jaringan. Supaya e-mail bisa sampai ke orang yang dituju, ia harus mengunduhnya terlebih dahulu. Proses download email menggunakan protokol POP (post office protocol). Saat ini POP sudah mencapai versi 3 sehingga disebut POP3. Alternatif yang lebih baik dari POP adalah IMAP (internet mail access protocol). IMAP sudah mencapai versi 4.

Baik POP maupun SMTP adalah protokol berbasis teks. Encoding yang digunakan adalah ASCII. Tidak masalah bila kita hanya ingin mengirim email teks saja. Masalah muncul ketika email berkembang, menjadi punya kemampuan untuk mengirim lampiran (attachment). Apa yang dilampirkan adalah file, dan file ini bisa file apa saja termasuk file biner. Kebetulan POP dan SMTP sama dalam hal terminasi pesan. Mereka menggunakan deretan karakter CR-LF. CR-LF (carriage return – line feed – tanda titik – carriage return – line feed) sebagai akhir dari pesan. Apa yang terjadi bila file biner kita di tengah-tengah terdapat byte-byte berikut : 0D 0A 2E 0D 0A? Nilai rangkaian byte tadi adalah kode ASCII dari CR-LF. CR-LF sehingga server akan menganggap pesan yang dikirim berhenti sampai di sana. File yang kita lampirkan kita akan putus di tengah.

Untuk mengatasi masalah ini, dibuatlah penyandian Base64. Cara kerja Base64 adalah sebagai berikut :

- Kelompokkan pesan setiap 3 karakter (3 byte = 24 bit). Bila terdapat sisa di akhir, tambahkan bit 0 sehingga panjangnya genap 24 bit.
- Pecah 24 bit tadi menjadi 4 kelompok yang masing-masing beranggotakan 6 bit.
- Setiap kelompok sekarang punya 2^6 kemungkinan susunan bit, berarti ada $2^6 = 64$ karakter tersedia untuk merepresentasikan 6 bit ini. Petakan setiap kelompok dengan karakter yang terdapat dalam tabel.

Karakter yang dipakai adalah huruf latin A-Z, huruf kecil a-z, dan angka 0-9. Semua berjumlah 62. Dua sisanya memakai simbol + dan / sehingga totalnya 64. Ditambah satu karakter khusus untuk padding byte yaitu simbol =. Bila dalam kelompok 3 byte itu satu byte terakhir hanya berisi padding bit, maka satu karakter = ditambahkan. Bila dua, maka dua karakter = (menjadi ==).

3. Analisa

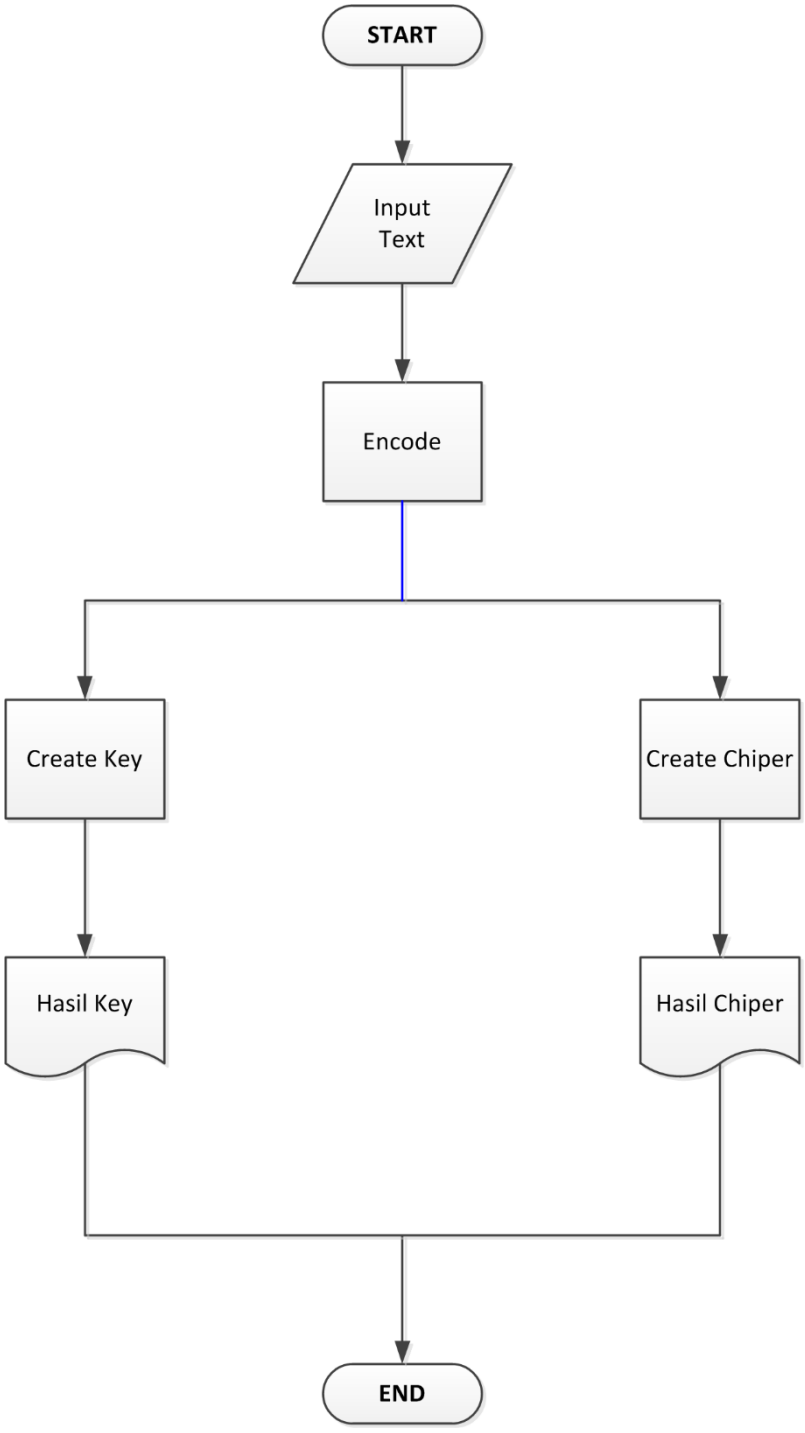
3.1 Deskripsi Sistem

Aplikasi yang dikembangkan ini adalah aplikasi berbasis web dengan menggunakan PHP sebagai pemrograman script dan web server apache sebagai penyedia layanan melalui protocol HTTP.

Aplikasi terutama sekali ditargetkan untuk melakukan proses encoding terhadap plaintext ataupun sebaliknya yaitu untuk melakukan proses decoding dari hasil encode menjadi plainrext, sebagai suatu metode untuk menjaga kerahasiaan pesan.

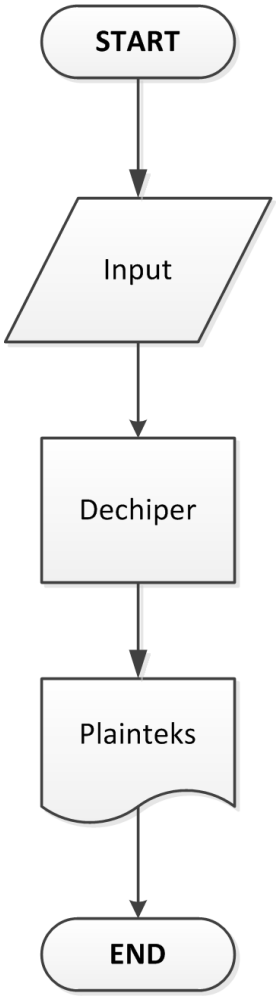
Algoritma kriptografi Base64 ini sebenarnya menggunakan algoritma kunci simetris atau disebut juga algoritma kriptografi konvensional, yaitu algoritma yang menggunakan kunci untuk proses *encoding* sama dengan kunci untuk proses *decoding*. Algoritma Base64 terdiri dari dua tahap besar, yaitu tahap encode dan tahap decode. Tahap pertama adalah pemilihan teks atau informasi (plainteks),

yang akan diubah menjadi isi yang tidak dipahami melalui proses encode (encipher), proses tersebut menghasilkan dua file yaitu file enkripsi dan file kunci (yang dinamakan enkripsi konvensional), file kunci digunakan pada saat memperoleh kembali informasi yang asli (decipher).



Gambar 1 Metode encoding

Untuk proses decoding, maka yang berlaku adalah kebalikan dari proses encoding, yaitu memasukkan hasil chipper dari proses encoding untuk kemudian di-decode dengan membalikkan proses yang terjadi pada kegiatan encoding sehingga dihasilkan plainteks kembali.



Gambar 2 Metode decoding

Sebagaimana tujuan dibuatnya aplikasi ini yaitu untuk melakukan proses encode dan decode, maka pengguna (user) dari aplikasi ini adalah mereka yang membutuhkan proses encode dan decode secara online agar dapat menyampaikan pesan mereka secara rahasia. Cara kerja untuk melakukan aplikasi ini adalah sebagai berikut :

1. Pengguna terlebih dahulu mengetikkan alamat dari aplikasi ini (pada pengujian masih dilakukan secara lokal).
2. Pada aplikasi akan terdapat dua pilihan yaitu melakukan proses encode atau proses decode.

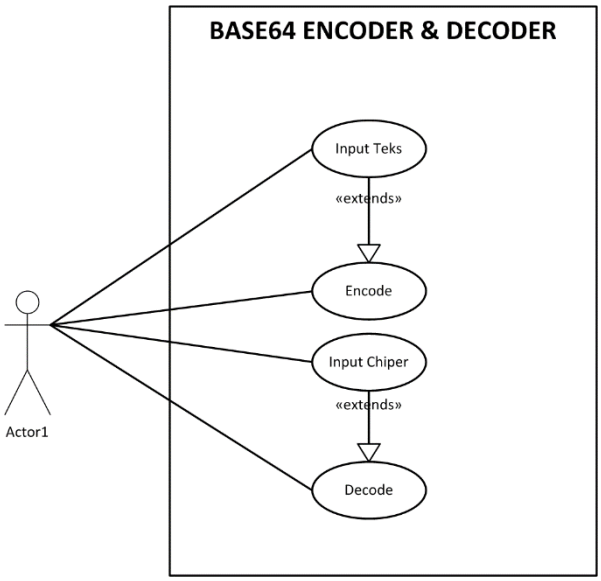
3. Untuk proses encode tinggal memilih tombol encode, maka kemudian akan tampil dua text area yaitu untuk memasukan plainteks dan area untuk hasil encode.
4. Selanjutnya masukkan teks yang akan di encode pada bagian plainteks.
5. Kemudian klik tombol encode, maka pada area teks di bawahnya akan tampak sederet huruf yang merupakan chipper sebagai hasil dari proses encode.
6. Untuk proses decode pada halaman awal pilih tombol decode.
7. Kemudian masukkan kode chipper yang akan di-decode
8. Klik tombol decode, maka kode chipper yang dimasukkan akan berubah menjadi plainteks.

3.1 Model Proses

UML adalah sebuah bahasa pemodelan yang telah menjadi standar dalam industri perangkat lunak untuk visualisasi, perancangan, dan pendokumentasian sistem perangkat lunak . Bahasa Pemodelan UML lebih cocok untuk pembuatan perangkat lunak dalam bahasa pemrograman berorientasi objek (C, Java, VB.NET, PHP 5), namun demikian tetap dapat digunakan pada bahasa pemrograman prosedural.

3.1.1. Use Case Diagram

Use Case Diagram dibuat untuk menggambarkan hubungan antara Actor (dalam hal ini user) dan Use Case (dalam hal ini service). Pada aplikasi ini ada beberapa use case yaitu input teks, encode, input chipper, dan decode.

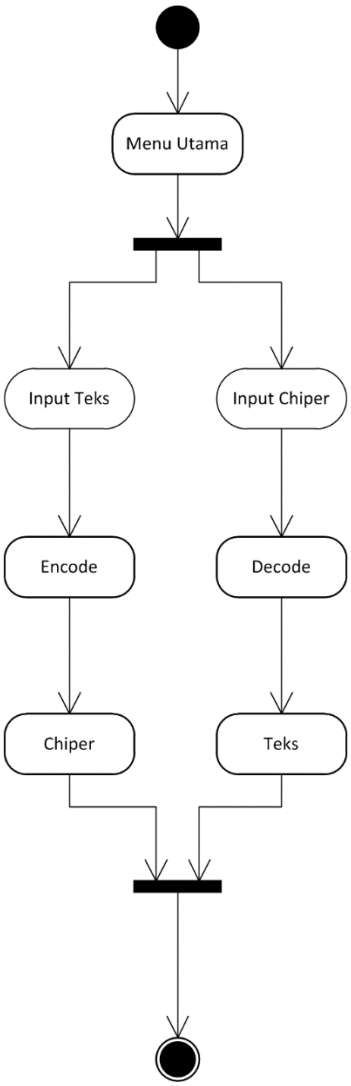


Gambar 3 Diagram use-case aplikasi

3.1.2. Activity Diagram

Activity Diagram atau diagram aktivitas adalah representasi grafis dari alur kerja kegiatan bertahap dan tindakan dengan dukungan untuk pilihan, iterasi dan konkurensi. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Pada Unified Modeling Language, diagram aktivitas dapat digunakan untuk menjelaskan bisnis dan operasional langkah-demi-langkah alur kerja komponen di dalam sistem.

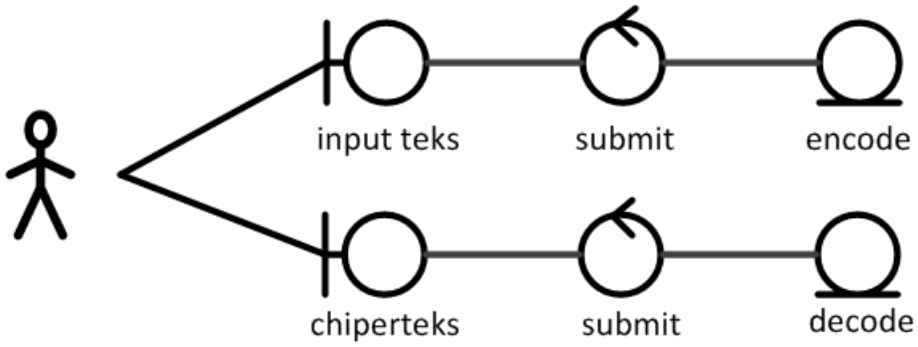
Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.



Gambar 4 Diagram aktifitas aplikasi

3.1.3. Communication Diagram

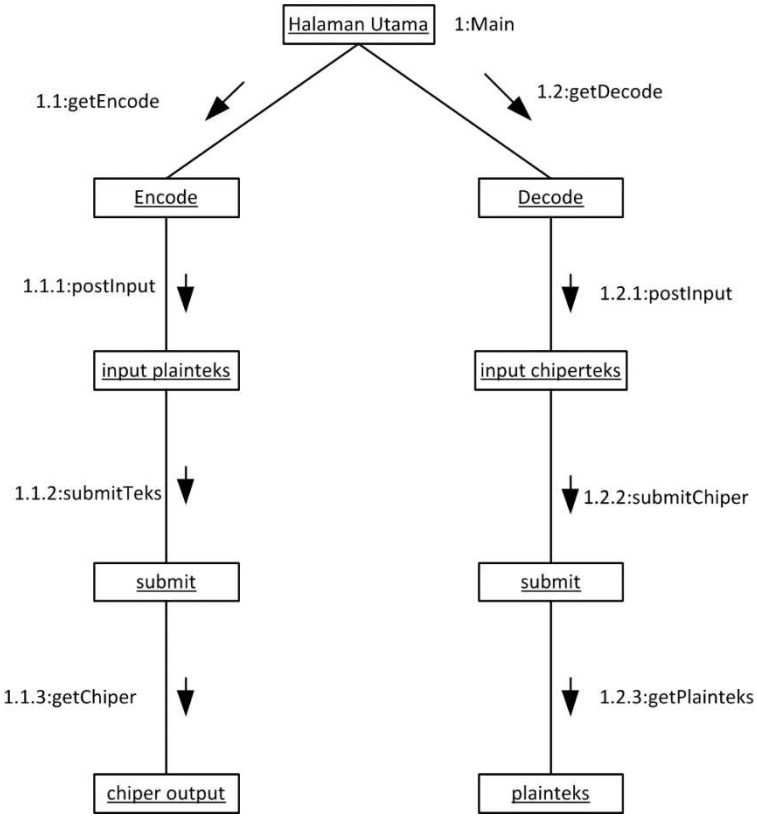
Communication Diagram dipakai untuk memodelkan interaksi antar object di dalam sistem. Berbeda dengan sequence diagram yang lebih menonjolkan kronologis dari operasi-operasi yang dilakukan, *Communication Diagram* lebih fokus pada pemahaman atas keseluruhan operasi yang dilakukan oleh object.



Gambar 5 Diagram komunikasi aplikasi

3.1.4. Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antar objek, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, di mana *message* dari level tertinggi memiliki nomor 1. *Messages* dari level yang sama memiliki prefiks yang sama.

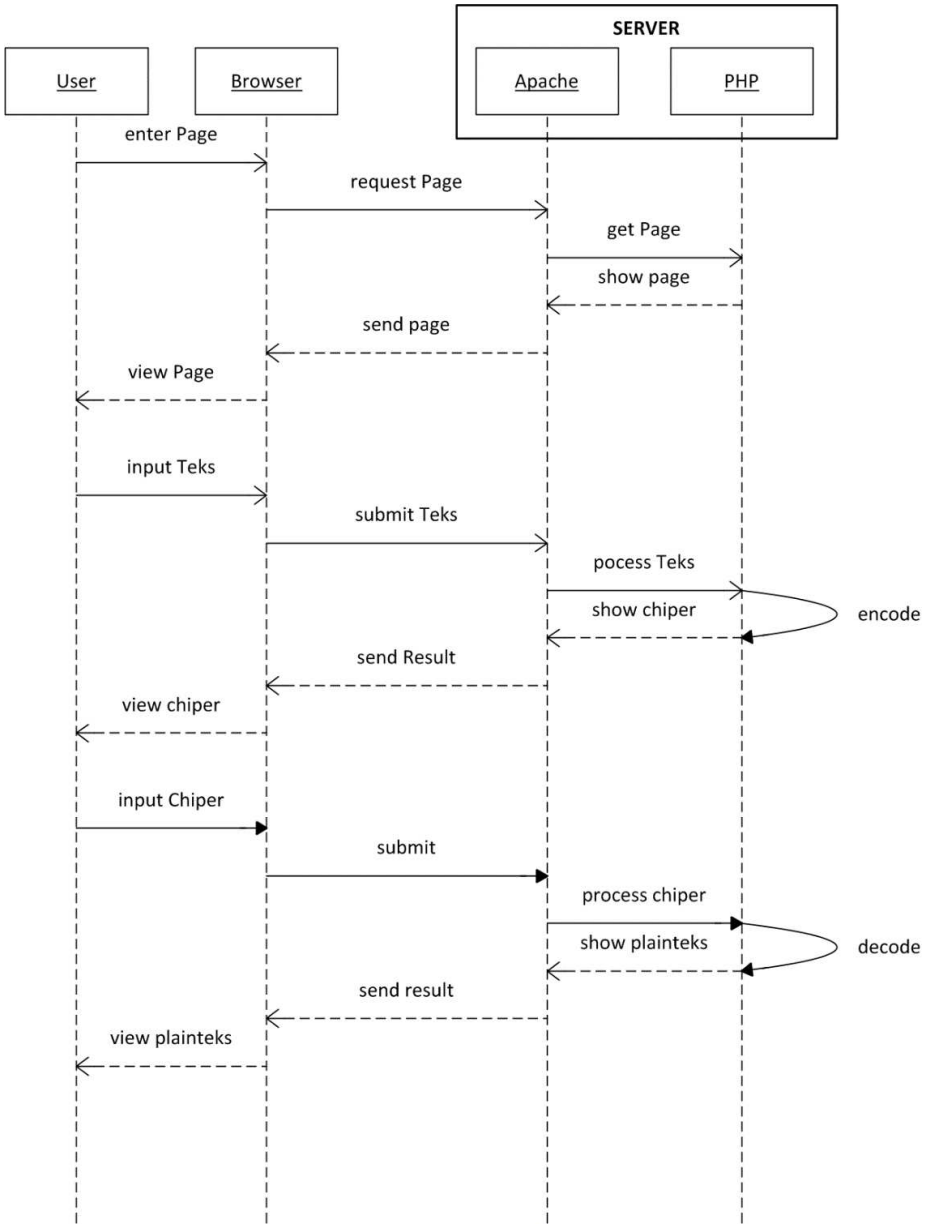


Gambar 6 Collaboration diagram aplikasi

3.1.5. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence* diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah *event* untuk menghasilkan output tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan.

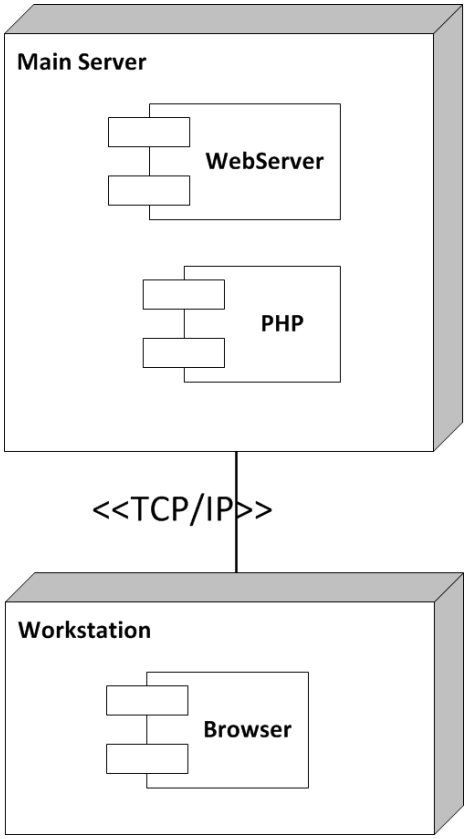


Gambar 7 Sequence diagram aplikasi

3.1.6. Deployment Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisikal.

Sebuah node adalah server, workstation, atau piranti keras lain yang digunakan untuk men-deploy komponen dalam lingkungan sebenarnya. Hubungan antar node (misalnya TCP/IP) dan requirement dapat juga didefinisikan dalam diagram ini.



Gambar 8 Deployment diagram aplikasi encode/decode

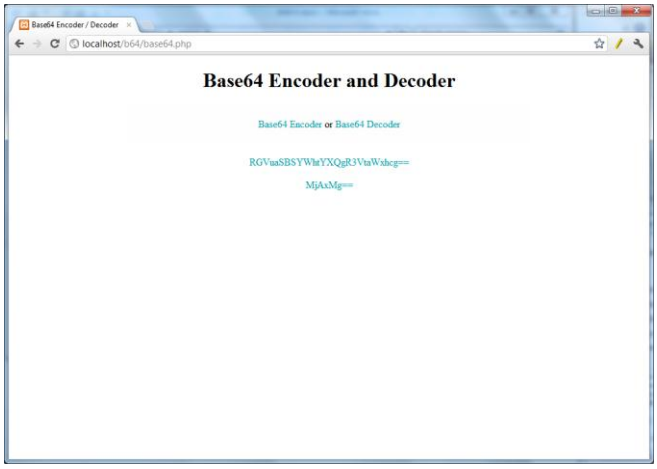
4. Hasil dan Pembahasan

4.1 Implementasi

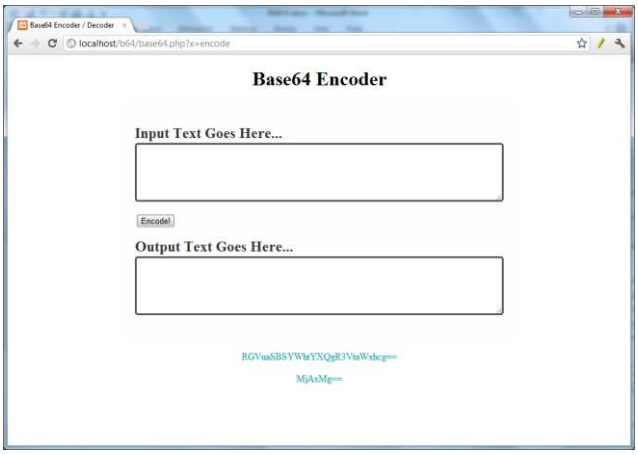
Dari hasil rancangan seperti yang telah digambarkan pada BAB III, maka tahap selanjutnya adalah mengimplementasikan rancangan tersebut dengan perangkat lunak yang telah disebutkan sebelumnya. Adapun tahapan-tahapan tersebut adalah sebagai berikut :

1. Melakukan instalasi perangkat lunak pendukung utama yaitu Apache dan PHP.
2. Membuat kerangka antarmuka aplikasi dalam bentuk HTML dengan menggunakan edit plus.
3. Membuat scrypt PHP dan menerapkannya pada HTML yang telah didesain sebelumnya.
4. Mendesain antarmuka aplikasi dengan menggunakan edit plus.

Berikut adalah hasil desain dan implementasi antarmuka perangkat lunak encode dan decode menggunakan algoritma Base64.



Gambar 9 Implementasi desain antarmuka aplikasi encode/decode



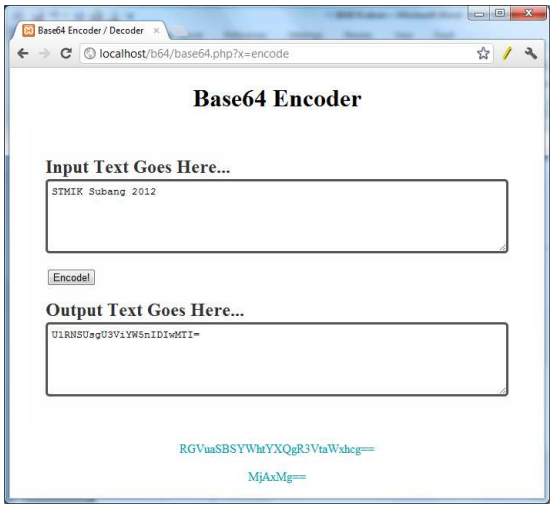
Gambar 10 Implementasi antarmuka untuk melakukan proses encode

Gambar 11 Implementasi antarmuka untuk melakukan proses decode

4.2 Pengujian Perangkat Lunak

Untuk mengetahui keberhasilan aplikasi ini, maka Penulis melakukan instalasi aplikasi dengan mentransfer file-file yang dibutuhkan untuk menjalankan aplikasi ini pada web server. Pengujian ini dilakukan dalam localhost dan menggunakan paket WAMP sebagai paket web server Apache, MySQL, dan PHP.

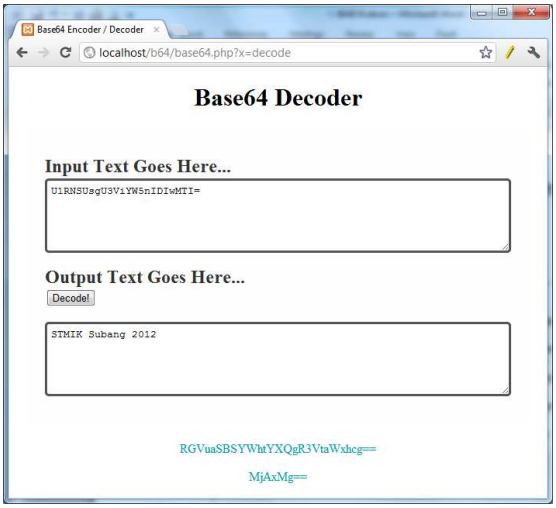
Untuk menguji keberhasilan proses encode, maka dilakukan pengujian dengan cara memasukkan teks pada area teks yang terdapat pada halaman proses encode. Berikut adalah hasil uji coba yang dilakukan.



Gambar 12 Hasil uji coba proses encode

Dari hasil yang diperlihatkan pada gambar 12, penulis mencoba melakukan proses encode dengan memasukkan plainteks berupa kalimat “STMIK Subang 2012”, kemudian melakukan submit plainteks tersebut agar diproses (di-encode). Dari proses encode atas kalimat di atas diperoleh hasil encode yaitu “U1RNSUsgU3ViYW5nIDIwMTI=”.

Untuk mencoba proses decode, penulis memasukkan hasil proses encode sebelumnya yaitu chipper berupa “U1RNSUsgU3ViYW5nIDIwMTI=”. Hasil chiperteks tersebut kemudian dicoba di-decode dengan mengakses halaman encode dan melakukan input chiperteks di atas. Berikut adalah hasil ujicoba proses decode.



Gambar 13 Hasil ujicoba proses decode

Dari hasil coba pada gambar 13, diperoleh hasil plainteks berupa "STMIK Subang 2012" yang merupakan input pada saat melakukan proses encode.

5. Simpulan

Hasil dari penelitiannya ini dapat disimpulkan sebagai berikut:

1. Bahwa proses encode dan decode dengan menggunakan algoritma Base64 dapat diterapkan menggunakan PHP.
2. Proses encode dan decode dengan menggunakan algoritma Base64 dapat diaplikasikan sebagai perangkat lunak berbasis web.
3. Aplikasi encode dan decode yang dibuat dengan menggunakan algoritma Base64 dapat digunakan sebagai metode penyembunyian pesan.
4. Aplikasi encode dan decode yang dibuat dapat dijadikan sebagai sarana pembelajaran untuk memahami algoritma base64

Pustaka

Dharwiyanti, Sri, *Pengantar Unified Modeling Language (UML)*, http://setia.staff.gunadarma.ac.id/Downloads/files/6077/Modul_UML.pdf.

IETF Community, *The Base16, Base32, and Base64 Data Encodings*, https://tools.ietf.org/html/rfc4648.

Munir, Rinaldi, *Pengantar Kriptografi*, Departemen Teknik Informatika Institut Teknologi Bandung, 2004.

Nugroho, Adi, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*, Bandung Informatika, 2005.

Rivest, Ronald, *The MD4 Message Digest Algorithm*, Advances in Cryptology-CRYPT0' 90 (Springer Berlin / Heidelberg), 1991.

Rumbaugh, J., Jacobson, I., & Booch Grady, *The Unified Modeling Language Reference Manual*, Addison-Wesley, 2005.

Scheiner, Bruce, *Aplied Cryptography*, John Wiley & Sons, 1996.

Scheiner, Bruce, *Practical Cryptography*, John Wiley & Sons, 2003.