BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Pustaka

SMS (*Short Message Service*) merupakan sarana yang paling populer digunakan untuk menyebarkan informasi. Kelemahan yang menjadi permasalahan pada teknologi ini adalah bagaimana melindungi informasi pada SMS agar tidak mudah terbaca orang lain serta masalah terbatasnya jumlah karakter. Berikut ini adalah beberapa jurnal yang membahas tentang keamanan dan pemampatan data.

Rughani dan Pandya (2012) dalam *International Journal of Mobile & Adhoc Network, volume* 2 *issue* 2 halaman 150-152, menerapkan steganografi pada *smartphone* berbasis Android**.** Pemilihan Android untuk steganografi dikarenakan ketersediaan *API* dan *library* yang mengijinkan program untuk menjalankan *low level operation* secara mudah. Rughani & Pandya menggunakan algoritma LSB (*Least Significant Bits*) untuk menyembunyikan pesan teks pada file gambar. Lingkungan percobaan yang digunakan adalah Android OS **:** 2.2 dengan 800 MHz ARM 11 *processor*.

Steganografi juga dapat diterapkan pada teks yang disembunyikan dalam teks lain. Seperti yang dikemukakan Por, Ang dan Delina (2008) dalam tulisannya yang berjudul *WhiteSteg: A New Scheme in Information Hiding Using Text Steganography* yang dimuat dalam jurnal *WSEAS Transactions on Computers, issue* 6 *volume* 7 halaman 735-745. Dalam jurnal ini, dijelaskan tentang metode *whitespace steganography* dimana sebuah pesan teks dapat disembunyikan pada pesan teks lain dengan memanfaatkan spasi antar kata atau antar paragaraf dan spasi di akhir pesan teks.

©UKDW

Pemanfaatan spasi untuk menyembunyikan pesan teks mengakibatkan jumlah karakter teks menjadi lebih banyak, sehingga diperlukan pemampatan data terlebih dahulu. Dalam survei yang dipublikasikan di *International Journal of Advancements in Research & Technology, volume* 2 *issue* 5 halaman 459-464 tentang teknik kompresi dan enkripsi pada SMS, Manoj Patil dan Prof. Vinay Sahu menjabarkan tentang algoritma LZW, Huffman, Deflate, E-Deflate-RLE, dan *Elliptic curve cryptography*. Dari banyaknya teknik kompresi dan enkripsi, algoritma yang dijabarkan dalam survei ini dapat dan telah diterapkan pada *mobile OS* sehingga algoritma tersebut dapat digunakan pada SMS (Patil & Sahu, 2013).

Dalam sebuah studi berjudul *Data Compression Techniques on Text Files: A Comparison Study* yang membandingkan algoritma Huffman, FLC, HFLC, dan LZW dan dimuat di *International Journal of Computer Applications* (0975 – 8887), *volume* 26 *no* 5, halaman 42-54, diambil kesimpulan bahwa algoritma LZW merupakan algoritma yang paling baik untuk kompresi teks, lalu diikuti Huffman, HFLC, dan terakhir FLC (Altarawneh, H. & Altarawneh, M., 2011).

©UKDW

2.2 Kompresi Data

Secara garis besar kompresi data adalah sebuah teknik untuk memampatkan data agar ukuran data tersebut menjadi lebih kecil sehingga tidak membutuhkan ruang penyimpanan yang besar serta dapat lebih efisien dalam pertukaran data.

Ada dua jenis teknik kompresi data yaitu teknik kompresi tanpa hilangnya informasi pada data (*lossless*) dan teknik kompresi dengan hilangnya beberapa informasi pada data (*lossy*). Pada umumnya teknik kompresi *lossless* diterapkan pada data teks sedangkan teknik kompresi *lossy* umumnya diterapkan pada data multimedia.

2.3 Kompresi Text (*Lossless*)

2.3.1 Huffman

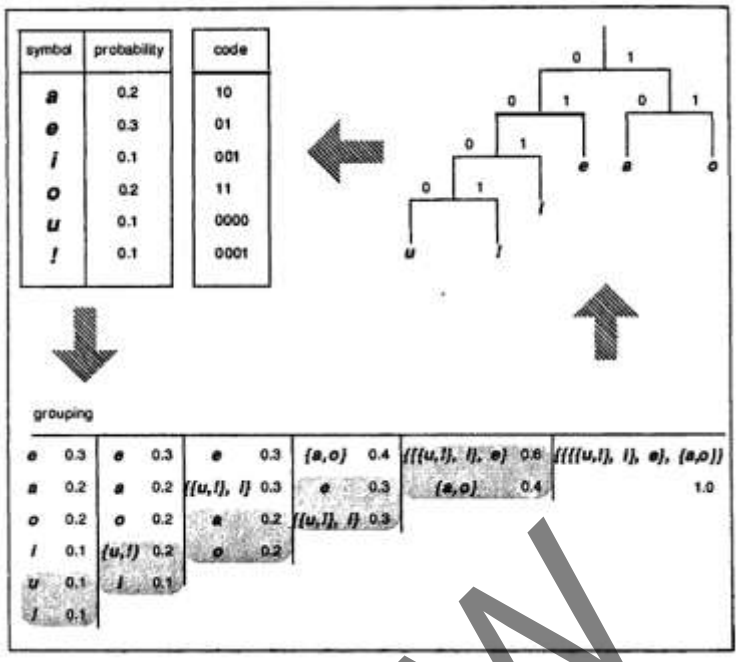
Algoritma Huffman ditemukan oleh seorang mahasiswa MIT pada tahun 1952 bernama David Huffman. Algoritma ini termasuk dalam metode kompresi statistik yang memanfaatkan perhitungan statistika (*Statistical Methods*) untuk melihat probabilitas kemunculan data dari sebuah dokumen. Probabilitas tersebut digunakan guna menentukan cara untuk mengolah data tersebut agar bisa dimampatkan.

Pada dasarnya cara kerja dari algoritma Huffman adalah membuat sebuah binary tree yang disebut dengan Huffman *Tree* untuk menentukan kode baru yang lebih pendek berdasarkan probabilitas kemunculan data. Berikut ini cara kerja kompresi dari algoritma Huffman sesuai yang ditulis dalam buku *Text Compression* (Bell, Cleary, & Witten, 1990, p. 105) :

1. Buat daftar semua simbol yang muncul beserta dengan probabilitas kemunculannya.
2. Urutkan simbol mulai dari probabilitas terkecil ke terbesar.
3. Pindahkan dua simbol dengan probabilitas terkecil menjadi satu set. Lalu gantikan dengan simbol baru yang merepresentasikan kedua simbol tersebut, dimana nilai probabilitasnya adalah jumlah probabilitas dari keduanya.
4. Ulang langkah 1, 2, 3 hingga daftar simbol tinggal satu anggota.

Prosedur ini menghasilkan struktur rekursif dari set dalam set, dimana tiap set memiliki dua anggota. Struktur tersebut dapat direpresentasikan dalam *binary tree* dengan simbol asli dan nilainya terdapat pada daun seperti pada gambar 2.1 berikut :

©UKDW



Gambar 2.1 Contoh dari Huffman coding

Dikutip dari : *Text Compression*.

(Bell, Cleary, & Witten, 1990, p. 106).

Sebagai contoh sebuah *string* 7 huruf "ABACCDA" membutuhkan representasi 7 × 8 bit = 56 bit (7 byte). Pada string "ABACCDA" tersebut, frekuensi kemunculan karakter A = 3, B = 1, C = 2, dan D = 1, sehingga dengan menggunakan algoritma di atas diperoleh kode Huffman seperti pada Tabel 2.1 berikut ini :

Tabel 2.1

Representasi bit string "ABACCDA"

Dikutip dari : INTEGRAL. (Linawati & Panggabean, 2004, vol. 9 p. 9).

Simbol	Frekuensi	Peluang	Kode Huffman
A	3	3/7	0
B	1	1/7	110
C	2	2/7	10
D	1	1/7	111

©UKDW

Dari Tabel 2.1 diatas string “ABACCDA” dapat dikompresi menjadi rangkaian bit 0110010101110 sehingga jumlah bit untuk merepresentasikan *string* tersebut menjadi hanya 13 bit yang sebelumnya membutuhkan 56 bit. Sedangkan untuk proses dekompresi dilakukan dengan cara membaca rangkaian bit 0110010101110 mulai dari awal dan dicocokkan dengan kode Huffman pada tabel diatas. Dari rangkaian bit tersebut, didapat bit pertama yaitu “0” yang merupakan kode untuk “A”, lalu bit selanjutnya yaitu “1” karena tidak ada simbol yang dikodekan dengan bit tersebut maka ditambah dengan bit selanjutnya menjadi “11”. Tidak ada juga simbol yang dikodekan dengan bit “11”, lalu baca lagi kode bit berikutnya, sehingga menjadi “110” yang merupakan kode untuk “B”, dan seterusnya hingga rangkaian bit habis.

Konsep kompresi Huffman adalah menggantikan kode biner setiap karakter dengan kode biner baru sesuai dengan tabel yang dibentuk dari Huffman *binary tree*, sehingga tabel Huffman tersebut harus disertakan dalam data yang telah dikompresi. Hal ini mengakibatkan data hasil kompresi bisa jadi lebih besar dari data asli jika data yang dikompresi sangat kecil.

Untuk file dengan ukuran kecil seperti SMS, kompresi Huffman dapat diterapkan dengan baik apabila menggunakan tabel Huffman yang tetap. Salah satu contoh penerapan kompresi Huffman dengan tabel tetap dijelaskan pada website: www.kejut.com/kompresisms, tabel Huffman yang digunakan adalah tabel Huffman untuk bahasa Inggris. Langkah kompresinya adalah menyusun kode biner karakter yang akan dikompresi berdasarkan tabel Huffman tetap yang sudah dibentuk. Tiap karakter pada tabel terdiri dari prefik untuk mengelompokkan huruf besar, kecil, atau karakter lainnya dan juga body yang berisi bit datanya.

Berikut merupakan tabel Huffman untuk bahasa Inggris yang digunakan pada kompresi SMS yang dijelaskan di www.kejut.com/kompresisms. Pada Tabel 2.2 dibawah ini tidak semua karakter ditampilkan, namun hanya sebagian saja sebagai contoh. Tabel lengkap dapat dilihat pada lampiran A-1 sampai A-2.

©UKDW

Tabel 2.2

Tabel Tetap Huffman (Sebagian) Untuk Bahasa Inggris

Dikutip dari : www.kejut.com/kompresisms.

	Prefik	Body	Hasil
a		0000	00000
b		00101	000101
c		1101	01101
d		01011	001011
f		11001	011001
g		10101	010101
h		1000	01000
i	0	0001	00001
j		110001001	0110001001
k		1100011	01100011
l		1001	01001
m		10100	010100
n		0110	00110
o		0100	00100
p		10110	010110
A		0000	1000000
B		00101	10000101
C		1101	1001101
D		01011	10001011
F	100	11001	10011001
G		10101	10010101
H		1000	1001000
I		0001	1000001
J		110001001	100110001001
<space>	101	1	1011
e		0	1010

©UKDW

2.3.2 LZW (*Lempel–Ziv–Welch*)

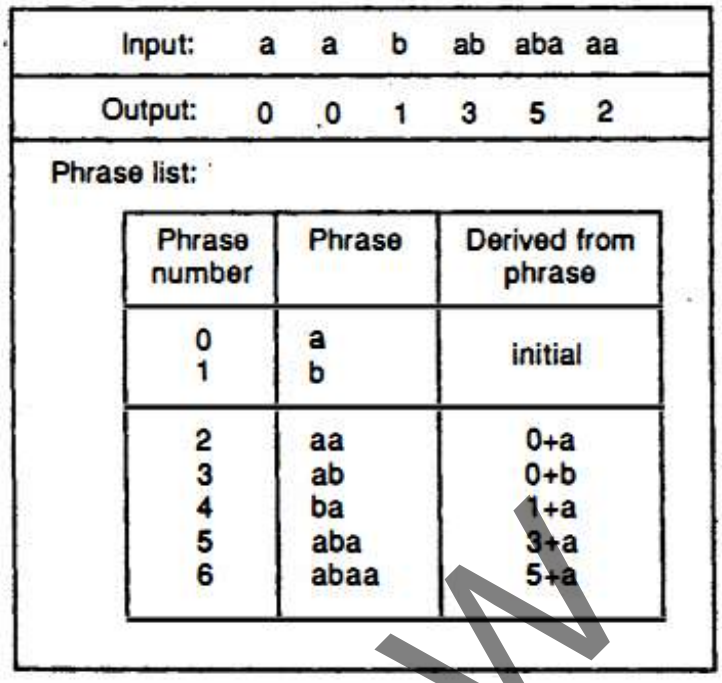
Algoritma LZW atau *Lempel–Ziv–Welch* diperkenalkan oleh Abraham Lempel, Jacob Ziv, dan Terry Welch pada tahun 1984. Algoritma ini termasuk dalam metode kompresi kamus (*Dictionary Methods*) dimana algoritma dengan metode ini menggunakan kamus yang disusun untuk menentukan *codeword* kompresi dan dekompresi. LZW merupakan implementasi dan pengembangan dari algoritma LZ78.

Berikut ini cara kerja kompresi dari algoritma LZW seperti yang ditulis Linawati & Panggabean (2004) dalam jurnal berjudul Perbandingan Kinerja Algoritma Kompresi Huffman, Lzw, Dan Dmc Pada Berbagai Tipe File :

1. Dictionary diinisialisasi dengan semua karakter dasar yang ada : {'A'..'Z','a'..'z','0'..'9'}.
2. P ← karakter pertama dalam stream karakter.
3. C ← karakter berikutnya dalam stream karakter.
4. Apakah string (P + C) terdapat dalam dictionary?
    a. Jika ya, maka P ← P + C (gabungkan P dan C menjadi string baru).
    b. Jika tidak, maka :
        i. Output sebuah kode untuk menggantikan string P.
        ii. Tambahkan string (P + C) ke dalam dictionary dan berikan nomor/kode berikutnya yang belum digunakan dalam dictionary untuk string tersebut.
        iii. P ← C.
5. Apakah masih ada karakter berikutnya dalam stream karakter?
    a. Jika ya, maka kembali ke langkah 2.
    b. Jika tidak, maka output kode yang menggantikan string P, lalu terminasi proses (stop).

Secara garis besar prinsip kerja dari LZW adalah menggantikan *string of characters* dengan kode tunggal. Setiap ditemukan *string of characters* baru, maka langsung dimasukkan ke dalam *table of strings.* Kompresi terjadi ketika diantara *string of character*, ditemukan satu kode tunggal yang merepresentasikan *string of character* tersebut lalu kode tunggal tersebut dijadikan *output.* Gambar 2.2. berikut menunjukkan contoh *encoding* LZW dengan *input string* "aabababaaa" seperti yang dijelaskan dalam buku *Text Compression* (Bell, Cleary, & Witten, 1990, p. 227).

©UKDW



Input:	a	a	b	ab	aba	aa
Output:	0	0	1	3	5	2

Phrase list:

Phrase number	Phrase	Derived from phrase
0	a	initial
1	b	
2	aa	0+a
3	ab	0+b
4	ba	1+a
5	aba	3+a
6	abaa	5+a

Gambar 2.2 LZW *coding* untuk *string* "aababababaaa"

(*phrases 0 and 1 are present before coding begins*)

Dikutip dari : *Text Compression*. (Bell, Cleary, & Witten, 1990, p. 227).

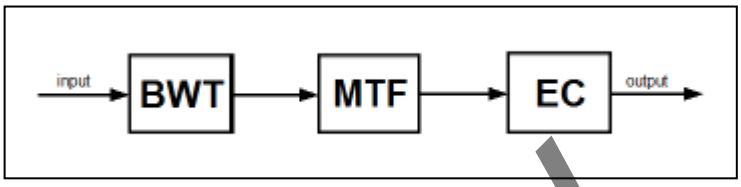
©UKDW

Gambar diatas menunjukkan proses kompresi *string* "aababababaaa" dimana *phrase* dan *phrase number* menunjukkan *string* yang sudah ditambahkan ke dalam *dictionary* dan nomor indeks (kode *output*) dari *string* tersebut. Dari nomor indeks dapat terbentuk *string* kode baru yang terdiri dari 6 karakter yaitu “001352” untuk merepresentasikan *string* "aababababaaa" yang memiliki 10 karakter.

Untuk proses dekompresi pada LZW, pertama *dictionary* diinisialisasi dengan semua karakter dasar yang ada. Lalu pada setiap langkah, kode dibaca satu per satu dari *string* kode, dikeluarkan *string* dari dictionary yang berkorespondensi dengan *string* kode tersebut, dan ditambahkan *string* baru ke dalam *dictionary*.

2.3.3 BWT *Compression*

Kompresi BWT memiliki tiga tahap *encoding* dalam mengkompresi sebuah data, pertama tahap transformasi dengan BWT (*Burrows–Wheeler Transform),* kedua tahap transformasi dengan *Move-To-Front Transform*, dan ketiga tahap kompresi dengan algoritma *Entropy Coding*. Gambar 2.3 berikut ini menunjukkan struktur dari kompresi BWT :



Gambar 2.3 Struktur kompresi BWT

Dikutip dari : Schiller, D. (2012). *The Burrows-Wheeler Algorithm*.

2.3.3.1 BWT (*Burrows–Wheeler Transform*)

Algoritma BWT atau *Burrows–Wheeler transform* ditemukan oleh Michael Burrows dan David Wheleer pada tahun 1994. Algoritma tersebut dibahas dalam makalah mereka yang berjudul "*A block-sorting Lossless data compression Algorithm*". Algoritma kompresi ini berdasarkan pada suatu tranformasi (*Transform-Based Methods*). Sebenarnya tranformasi ini ditemukan oleh David Wheeler pada tahun 1983 namun belum dipublikasikan.

Pada dasarnya cara kerja dari algoritma BWT adalah memproses sebuah blok data teks dan menyusunnya dengan algoritma pengurutan, sehingga menghasilkan blok data yang sama dengan sebelumnya tapi dengan urutan yang berbeda. Berikut ini cara kerja kompresi dari algoritma BWT sesuai yang ditulis dalam buku *Compression Handbook* (Fenwick & Sayood (ED.), 2003, p. 171):

1. Tulis input teks dalam baris pertama sebuah matriks, dengan satu simbol (karakter) per kolom.

©UKDW

2. Buat seluruh rotasi setiap permutasi dari baris tersebut (*input* teks) dan tulis masing-masing dibaris berikutnya.
3. Urutkan baris matriks berdasarkan abjad (*lexicographical order*) dari simbol dalam baris.
4. Ambil simbol (karakter) terakhir dari tiap baris matriks yang sudah diurutkan sebelumnya, ambil juga indeks yang menunjukan urutan baris dari teks asli *(input* teks) dalam matriks yang sudah diurutkan.

Contoh cara kerja dari algoritma transformasi BWT, misalkan *string* "kaukapakapa" dibuat seluruh rotasi permutasinya dan diurutkan berdasarkan abjad seperti pada Tabel 2.3. Baris pertama adalah "akapakaukap" dengan final "p", baris kedua "akaukapakap" dengan final "p", dan seterusnya. Yang dimaksud dengan final adalah huruf pada kolom terakhir tiap baris, yang diambil mulai dari baris pertama sehingga didapat *string* "ppkkkuaaaaa", didapatkan pula nilai 7 yang menunjukan urutan baris dari teks asli (*string* "kaukapakapa").

Tabel 2.3 Transformasi maju BWT
Dikutip dari : *Theoretical Computer Science*.
(Fenwick, P., 2007, vol. 387 p. 3).

Rotated input	Sorted rotations	Final
kaukapakapa	akapakaukap	p
akaukapakap	akaukapakap	p
pakaukapaka	apakapakauk	k
apakaukapak	apakaukapak	k
kapakaukapa	aukapakapak	k
akapakaukap	kapakapakau	u
pakapakauka	kapakaukapa	a
apakapakauk	kaukapakapa	a
kapakapakau	pakapakauka	a
ukapakapaka	pakaukapaka	a
aukapakapak	ukapakapaka	a

©UKDW

Pembalikan transformasi *sorting* BWT memerlukan *string* dari final yaitu “ppkkkuaaaaa” dan urutan baris dari teks asli yaitu 7.

1. Langkah pertama pembalikan transformasi *sorting* BWT dengan mengurutkan sesuai abjad *string* dari final yaitu “ppkkkuaaaaa” sehingga terbentuk string “aaaaakkkppu”.
2. Karena *string* dari final yaitu “ppkkkuaaaaa” merupakan karakter pada kolom terakhir matriks yang sudah diurutkan berdasarkan abjad (*lexicographical order*), maka dapat dipastikan string hasil pengurutan abjad dari *string* dari final yaitu “aaaaakkkppu” merupakan karakter pada kolom pertama matriks. Dari kedua string tersebut, kolom matriks yang lain dapat diketahui.

Pada Tabel 2.4 ditunjukkan proses pembalikkan transformasi *sorting* BWT dengan diketahui *string* karakter pada kolom terakhir matriks adalah “ppkkkuaaaaa”. Berdasarkan hasil dari pengurutan terakhir yaitu kolom 10 dari tabel dan telah diketahui urutan baris dari *string* asli yaitu 7, maka *string* asli hasil dari proses pembalikkan transformasi *sorting* BWT adalah “kaukapakapa”.

Tabel 2.4 Transformasi mundur BWT

Dikutip dari : *Theoretical Computer Science*.

(Fenwick, P., 2007, vol. 387 p. 4).

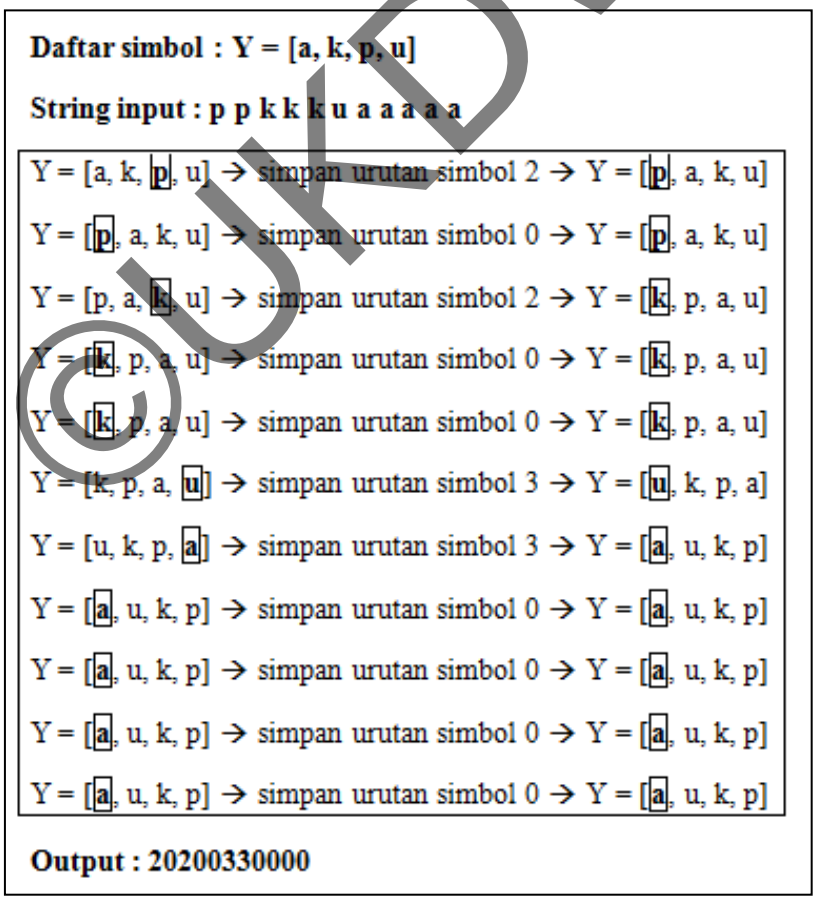
1	2	3	4	5	6	7	8		9	10
Ord 1		Ord 2		Order 3		Order 4			Order 11	
p	a	pa	ak	pak	aka	paka	akap	...	pakapakauka	akapakaukap
p	a	pa	ak	pak	aka	paka	akau	...	pakaukapaka	akaukapakap
k	a	ka	ap	kap	apa	kapa	apak	...	kapakapakau	apakapakauk
k	a	ka	ap	kap	apa	kapa	apak	...	kapakaukapa	apakaukapak
k	a	ka	au	kau	auk	kauk	auka	...	kaukapakapa	aukapakapak
u	k	uk	ka	uka	kap	ukap	kapa	...	ukapakapaka	kapakapakau
a	k	ak	ka	aka	kap	akap	kapa	...	akapakaukap	kapakaukapa
a	k	ak	ka	aka	kau	akau	kauk	...	akaukapakap	kaukapakapa
a	p	ap	pa	apa	pak	apak	paka	...	apakapakauk	pakapakauka
a	p	ap	pa	apa	pak	apak	paka	...	apakaukapak	pakaukapaka
a	u	au	uk	auk	uka	auka	ukap	...	aukapakapak	ukapakapaka

©UKDW

2.3.3.2 MTF *(Move-To-Front) Transform*

Pada umumnya tahap kedua dari kompresi BWT adalah transformasi dengan algoritma MTF (*Move-To-Front*). Prinsip kerja dari MTF adalah membuat daftar dari semua simbol sebuah *string*, lalu mencatat urutan posisi simbol dalam daftar dan kemudian memindahkan posisi simbol tersebut ke urutan pertama dari daftar simbol.

Dari hasil tahap pertama yaitu transformasi BWT, didapatkan sebuah *string* "ppkkkuaaaaa" dan urutan baris dari *string* asli yaitu 8. Dari *string* tersebut dibuat daftar simbol yang diurutkan abjad yaitu Y = [a, k, p, u]. Berikut ini proses transformasi MTF dari *string* "ppkkkuaaaaa" yang menghasilkan *output* 20200330000 seperti yang ditunjukkan pada gambar 2.4 dan urutan baris *string* asli dari transformasi BWT yaitu 7.



Gambar 2.4 Transformasi maju MTF

©UKDW

2.3.3.3 EC (*Entropy Coding*) : *Run-Length Coding*

Pada tahap pertama dan kedua dari kompresi BWT belum terjadi kompresi data, tahap pertama dan kedua merupakan tahap transformasi data, agar hasil kompresi lebih optimal saat data dikompresi dengan algoritma *entropy coding* sederhana pada tahap ketiga ini.

Ada beberapa algoritma *entropy coding* yang dapat digunakan, namun pada tugas akhir ini akan diterapkan algoritma *Run-Length Coding* untuk proses kompresi BWT tahap ketiga. Prinsip dasar dari RLE ini adalah menggantikan karakter berulang pada sebuah teks dengan kode representasi yang lebih pendek.

Dari hasil tahap kedua yaitu transformasi MTF, didapat *output* 20200330000 dan urutan baris dari *string* asli yaitu 7. *Output* dan baris dari *string* asli yaitu 202003300007 inilah yang akan dijadikan *input* dari proses RLE. Pengkodean RLE yang akan digunakan adalah teknik pengkodean dengan membandingkan karakter ke-i dengan karakter ke- (i-1). Jika karakter ke-i sama dengan karakter ke- (i-1) maka karakter ke- (i-1) dikodekan dengan kode biner 1, sedangkan kalau berbeda maka dikodekan dengan kode biner 0 dan juga karakter ke- (i-1) tersebut disimpan dalam *string* baru. Sehingga menghasilkan kode biner 00001010111 dan *string* baru 202030.

Dari hasil tahap pertama sampai ketiga kompresi BWT didapat *output* 202030 yang terdiri dari 6 karakter dan kode biner yang terdiri dari 11 bit, sehingga menghasilkan data sebesar 59 bit. Sedangkan besar *string* awal “kaukapakapa” yang terdiri dari 11 karakter adalah 88 bit.

2.4 Steganografi Teks

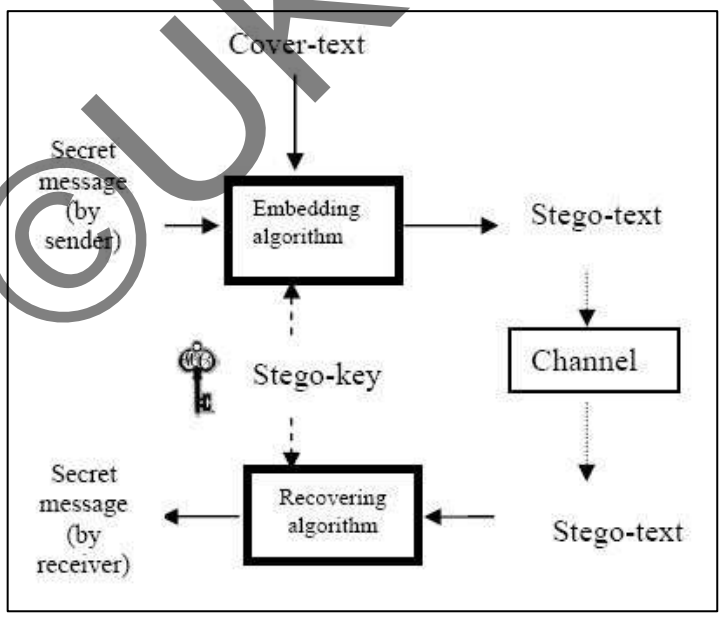
Steganografi adalah teknik menyembunyikan data dalam data lainnya. Dalam steganografi, idealnya data yang disembunyikan (*embedded data*) tidak merusak file pembawa (*cover-object*) sehingga pada umumnya

©UKDW

steganografi lebih efisien jika diterapkan pada file gambar, audio, atau video. Berikut ini merupakan istilah-istilah yang digunakan dalam steganografi:

a. *Embedded data* - pesan rahasia yang disembunyikan.
b. *Cover-object* - media untuk menyembunyikan pesan rahasia.
c. *Stego-object* - media yang sudah terisi *embedded message*.
d. *Stego key* - kunci untuk proses *encoding-decoding* steganografi.
e. *Steganalysis* - proses mendeteksi keberadaan *embedded message*.

Walaupun steganografi lebih efisien diterapkan pada file multimedia, namun steganografi juga dapat diterapkan pada file teks. Gambar 2.5 berikut menunjukkan mekanisme dasar dari steganografi yang dijelaskan Petitcolas, Anderson dan Kuhn. Pertama *embedded data* akan disembunyikan pada *cover-text* dengan menggunakan algoritma *embedding* untuk menghasilkan *stego-text*. Untuk membaca *embedded data* pada *stego-text* diperlukan algoritma *recovering* yang memerlukan *stego-key* untuk membuka pesan rahasia (seperti yang dikutip Por et al, 2008, p. 736).



Gambar 2.5 Mekanisme Steganografi Teks
Dikutip dari : *WSEAS Transactions on Computers*.
(Por et al, 2008, vol. 7 p. 736).

©UKDW

Bender et al (seperti yang dikutip Por et al, 2008, p. 739) menjelaskan bahwa ada 3 metode untuk menerapkan steganografi dalam teks, yaitu metode *open space* yang menyandikan *embedded data* menggunakan manipulasi *whitespace*, metode *syntactic* yang memanfaatkan metode baca dan metode *semantic* yang menyandikan *embedded data* menggunakan manipulasi kata sinonim.

2.5 Whitespace Steganography

Steganografi teks dapat diterapkan dengan menggunakan metode *open space* yang memanfaatkan karakter *whitespace*, sehingga dikenal juga dengan *whitespace steganography*. Karakter *whitespace* adalah karakter yang tidak tercetak pada teks viewer, yang terdiri dari *unprintable control codes* dan *used to control peripherals* seperti yang ada pada lampiran A-3.

Seperti dijelaskan Bender et al. yang disitir oleh Por dan Delina (2008), metode *open space* terbagi menjadi *inter-sentence spacing*, *end-of-line spaces* dan *inter-word spacing*. *Inter-sentence spacing* mengkodekan data biner pesan rahasia ke dalam *cover-text* dengan menambahkan satu spasi untuk mengkodekan "0" dan dua spasi untuk mengkodekan "1" pada akhir karakter terminasi.

End-of-line spaces memanfaatkan spasi di akhir tiap baris teks. Data biner pesan rahasia dikodekan menggunakan jumlah spasi yang sudah ditetapkan dan ditambahkan pada akhir baris *cover-teks*.

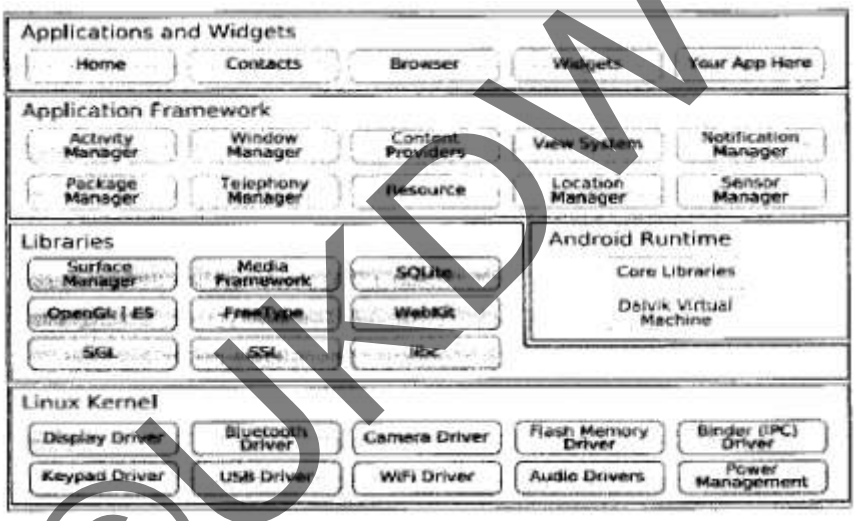
Sedangkan *inter-word spacing* mengkodekan data biner pesan rahasia ke dalam *cover-text* dengan menambahkan satu spasi untuk mengkodekan "0" dan dua spasi untuk mengkodekan "1" pada spasi antar kata (pada kondisi teks *justified*).

Untuk membedakan pengkodean dengan menggunakan spasi antara "0" dan "1" seperti diatas, digunakan karakter tab sebagai pemisah antar kode spasi yang merepresentasikan "0" atau "1".

©UKDW

2.6 Android

Android merupakan sistem operasi *mobile* yang berbasis Linux dan dimiliki oleh Google Inc. sejak tahun 2005. Untuk pengembangan Android dibentuklah *Open Handset Alliance* yang merupakan konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk diantaranya Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. Pada saat perilisan, Android dirilis di bawah lisensi Apache yang merupakan lisensi perangkat lunak yang mendukung *open source* sehingga memungkinkan vendor untuk mengembangkan dan memodifikasi Android.



Gambar 2.6 Arsitektur Android

Dikutip dari : Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. (Safaat H., 2012, p. 9).

Gambar 2.6 diatas merupakan garis besar dari arsitektur Android yang terdiri dari lima lapisan (*layer*) utama seperti yang ditulis Nazruddin Safaat H. dalam buku berjudul Pemrograman Aplikasi *Mobile Smartphone* dan *Tablet PC* Berbasis Android (2012, p. 6-9) yaitu:

©UKDW

1. *Applications* dan *Widgets*

   Layer yang berhubungan dengan aplikasi, *layer* dimana aplikasi diunduh, di-*install*, dan *running*. Di *layer* ini juga terdapat aplikasi inti seperti klien email, kalender, kontak, *browser*, dan lain-lain.

2. *Applications Frameworks*

   Layer dimana aplikasi dapat dirancang dan dibuat. Para pengembang diberi kebebasan dalam mengakses perangkat keras, akses informasi *resource*, menjalankan *service background*, dan sebagainya untuk menghasilkan aplikasi yang inovatif.

3. *Libraries*

   Layer dimana fitur-fitur Android berada. Layer ini berjalan di atas *kernel* dan diakses untuk menjalankan aplikasi. Contohnya *library* C/C++ inti seperti Libc dan SSL.

4. Android *Run Time*

   Layer yang membuat aplikasi Android dapat dijalankan yang dalam prosesnya menggunakan implementasi Linux**.** *Layer* ini menyediakan sebuah *libraries* yang berfungsi menerjemahkan bahasa Java sehingga aplikasi dengan bahasa pemrograman Java dapat dijalankan di DVM (*Dalvik Virtual Machine*) yang merupakan mesin pembentuk dasar kerangka aplikasi Android.

5. Linux *Kernel*

   Layer dimana inti dari sistem operasi Android berada. *Layer* ini berisi file-file sistem yang mengatur sistem *processing*, *memory*, *resource*, *drivers*, dan sistem-sistem operasi Android lainnya.

Aplikasi di Android memiliki konsep yang hampir sama dengan aplikasi *desktop*, namun berbeda *packaged* dan memiliki struktur yang membuat ponsel lebih tahan terhadap *crash*. Berikut ini merupakan empat komponen pada aplikasi Android yang dijelaskan dalam buku *The Busy Coder's Guide to Android Development* (Murphy, 2010, p. 3-4) :

©UKDW

1. *Activities*

   Merupakan komponen yang digunakan pengguna untuk berinteraksi dengan aplikasi (*user interface*). *Activity* dapat dianalogikan dengan *form* atau *window* pada aplikasi *desktop*. Namun *activity* bisa juga tidak memiliki *user interface* dimana *code* program terbungkus dalam *form content providers* atau *services.*

2. *Services*

   Activities memiliki daur hidup yang pendek dan dapat dimatikan kapan saja, sedangkan *services* dirancang untuk tetap dapat berjalan sendiri di *background process* walaupun tidak ada *activity* yang sedang berjalan di *foreground.*

3. *Intents*

   Intents merupakan *system messages* yang berjalan di dalam *device*, berfungsi untuk menyampaikan notifikasi jika terjadi beberapa kejadian pada *device*. Contohnya seperti perubahan pada *hardware* (SD *card* dimasukkan), data masuk (misal SMS diterima), dan sebagainya. Tidak hanya untuk menerima notifikasi, *intents* juga dapat digunakan untuk mengirimkan notifikasi kepada aplikasi lain.

4. *Content Providers*

   Merupakan penyedia kumpulan data secara spesifik sehingga data dapat digunakan oleh lebih dari satu aplikasi. *Content provider* menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu *activity*.

©UKDW

2.7 SMS (*Short Message Service*)

SMS merupakan salah satu sarana untuk bertukar informasi yang disediakan dalam perangkat ponsel. Dalam beberapa tahun terakhir ini, SMS merupakan fitur ponsel yang paling populer dikalangan pengguna ponsel. Kepopuleran SMS tersebut dapat dilihat dari data yang dirilis ATSI mengenai

jumlah SMS yang terkirim di Indonesia pada tahun 2011 mencapai 260 miliar SMS (TeknoJurnal, 2012).

SMS menggunakan mekanisme pengiriman data *store* dan *forward* dengan mekanisme SMS dari pengirim akan di *store* ke SMS-*center*, lalu SMS *center* akan mem-*forward* SMS tersebut ke nomor tujuan penerima. Dengan mekanisme pengiriman data tersebut, pengirim dan penerima tidak harus terus menerus terhubung dengan SMS-*center*.

Menurut penjelasan dari Brown, Shipman, dan Vetter (2007), pesan SMS dapat dikirim melalui perantara SMS-*center* yang disediakan oleh operator seluler. SMS-*center* dapat menyampaikan pesan SMS ke perangkat lain dengan ukuran maksimal 140 oktet, dengan demikian sebuah pesan SMS maksimal dapat memuat 160 karakter 7-bit. Hal ini juga memungkinkan untuk menggunakan skema lain seperti 8-bit atau 16-bit, yang tentunya juga menurunkan panjang pesan maksimal menjadi 140 karakter dan 70 karakter.

Masih menurut Brown et al, pesan SMS juga dapat digunakan untuk mengirim pesan biner, biasanya diperlukan aplikasi khusus pada ponsel untuk membaca pesan biner. Penggunaan pesan biner pada SMS misalnya untuk mengirim dan menerima pesan gambar.

©UKDW

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Kebutuhan Sistem

Aplikasi SMS Rahasia yang akan dibangun berbasis sistem operasi mobile Android dan menggunakan bahasa pemrograman Java, berikut ini adalah kebutuhan sistem dalam pengembangan aplikasi ini.

3.1.1 Kebutuhan Fungsional

Berikut ini merupakan daftar kebutuhan fungsional sistem dari aplikasi SMS Rahasia :

1. SMS Rahasia dapat mengirim dan membaca SMS dalam *inbox* dan *sent item* dengan fungsi standar dari *Android.telephony.*
2. SMS Rahasia dapat menerima *input* karakter *ASCII.*
3. SMS Rahasia dapat menghasilkan *output* karakter *ASCII* yang berisi teks SMS dan pesan rahasia yang sudah disembunyikan.
4. SMS Rahasia dapat mengompresi string biner dari karakter *ASCII.*
5. Pengguna dapat memilih algoritma untuk kompresi pesan rahasia yaitu Huffman, LZW, BWT, atau tanpa kompresi.
6. SMS Rahasia dapat mengetahui algoritma kompresi yang digunakan pengirim saat proses dekompresi.
7. SMS Rahasia dapat mendekompresi string biner dari karakter *ASCII* pada SMS di *inbox* dan *sent item.*
8. SMS Rahasia dapat menyembunyikan pesan rahasia dengan menambahkan kombinasi karakter tab dan spasi di akhir teks SMS.
9. SMS Rahasia membaca pesan rahasia dengan membaca kombinasi karakter tab dan spasi di akhir teks SMS.

©UKDW

3.1.2 Kebutuhan Non Fungsional

3.1.2.1 Kebutuhan *Hardware* dan *Software*

Dalam pengembangan aplikasi SMS Rahasia ini, penulis menggunakan perangkat (*notebook*) dengan spesifikasi :

a. Hardware

1. Prosesor Intel ® Core™ i5-2410M 2,3 GHz
2. Memori 8 GB RAM
3. Harddisk 640 GB
4. VGA Intel ® HD Graphics 3000
5. Monitor 14 inch (resolusi 1366 * 768)

b. Software

1. Sistem Operasi Windows 7 Professional 64-bit
2. Tools : adt-bundle-windows-x86_64

3.1.2.2 *Environment* Implementasi Sistem

Berikut ini merupakan spesifikasi minimum untuk implementasi sistem agar dapat digunakan dengan baik oleh pengguna aplikasi SMS Rahasia :

1. Prosesor dengan kemampuan 850 MHz
2. *Memory* 2 GB ROM dan 512 RAM
3. *Display* 3 inch QVGA
4. Mendukung jaringan CDMA atau GSM
5. Sistem Operasi Android 4.1.2 (*Jelly Bean*)

©UKDW

3.2 Gambaran Kerja Sistem

Gambaran kerja sistem memuat detail spesifikasi sistem SMS Rahasia, yang terdiri dari *flowchart* sistem, *use case* diagram, *sequence* diagram, dan *class* diagram. Diagram-diagram tersebut merupakan model representasi dari sistem yang menjelaskan deskripsi kerja dari sistem SMS Rahasia.

3.2.1 *Flowchart* Sistem

Berikut ini merupakan *flowchart* kirim pesan rahasia, *flowchart* baca pesan rahasia, *flowchart* kompresi dan *encoding* steganografi, serta *flowchart* dekompresi dan *decoding* steganografi dari sistem SMS Rahasia yang ditunjukkan pada gambar 3.1 sampai gambar 3.4.

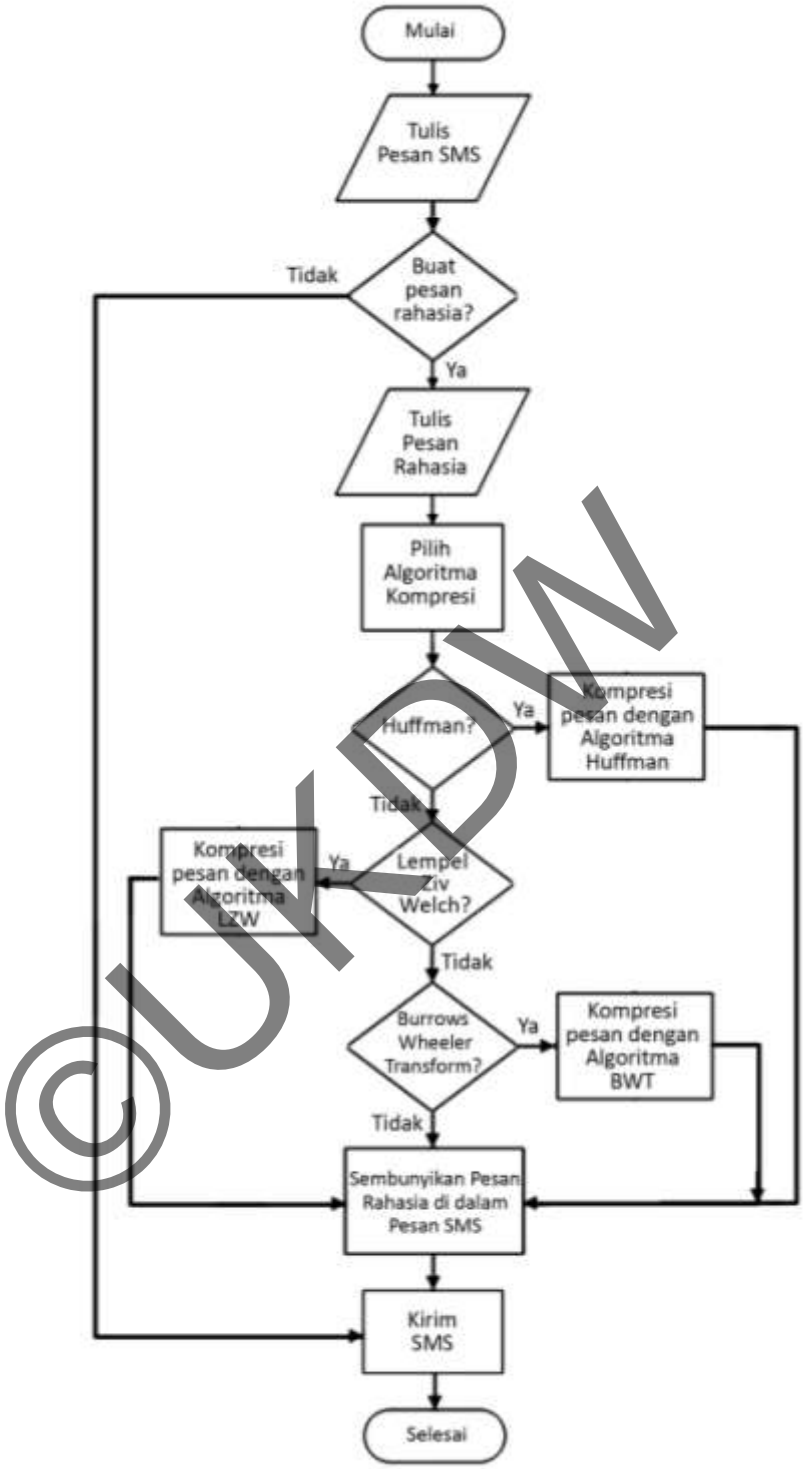
a. Proses Kirim SMS yang Mengandung Pesan Rahasia

Proses yang harus dijalani pengirim pesan untuk mengirim SMS mengandung pesan rahasia rahasia adalah sebagai berikut :

1. Mulai
2. Masukkan input dari aplikasi yaitu nomor tujuan dan isi SMS.
3. Memilih menu tambah pesan rahasia.
4. Masukkan input pesan yang ingin disembunyikan.
5. Memilih pilihan algoritma kompresi.
6. Aplikasi menjalankan proses kompresi dan steganografi sesuai inputan isi sms (*Cover-object*), pesan rahasia (*Embedded data*), dan pilihan algoritma kompresi.
7. Aplikasi mengirimkan *stego-object* pesan melalui sistem SMS pada sistem operasi Android.
8. Selesai.

Rancangan dari proses kirim SMS rahasia dapat dilihat pada gambar 3.1.

©UKDW



Gambar 3.1 *Flowchart* Kirim Pesan Rahasia

©UKDW

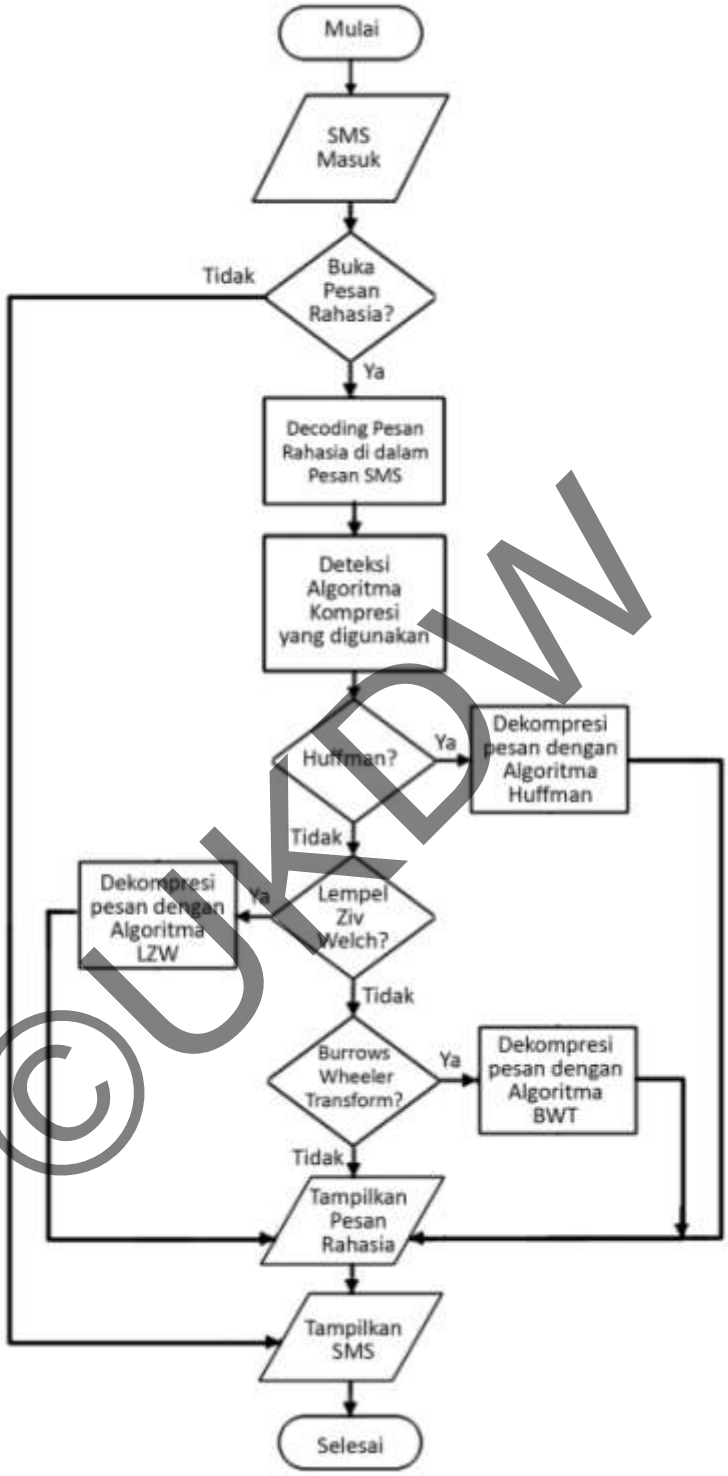
b. Proses Baca SMS yang Mengandung Pesan Rahasia

SMS (*stego-object*) yang diterima dapat dibaca pesan rahasianya dengan menggunakan fasilitas *decoding* steganografi dan dekompresi yang ada pada aplikasi, berikut adalah rangkaian prosesnya:

1. Mulai.
2. Aplikasi mengakses database *inbox* Android.
3. Dua puluh SMS terakhir ditampilkan dalam *list*.
4. Penerima memilih SMS yang ingin dibuka.
5. Penerima memilih buka pesan rahasia.
6. Aplikasi memproses isi SMS untuk *decoding* steganografi pesan rahasia.
7. Aplikasi mendeteksi algoritma kompresi yang digunakan lalu menjalankan dekompresi pesan rahasia.
8. Aplikasi menampilkan pesan rahasia.
9. Selesai.

Rancangan dari proses kirim SMS rahasia dapat direpresentasikan pada *flowchart* yang ditampilkan di gambar 3.2 berikut ini.

©UKDW



Gambar 3.2 *Flowchart* Baca Pesan Rahasia

©UKDW

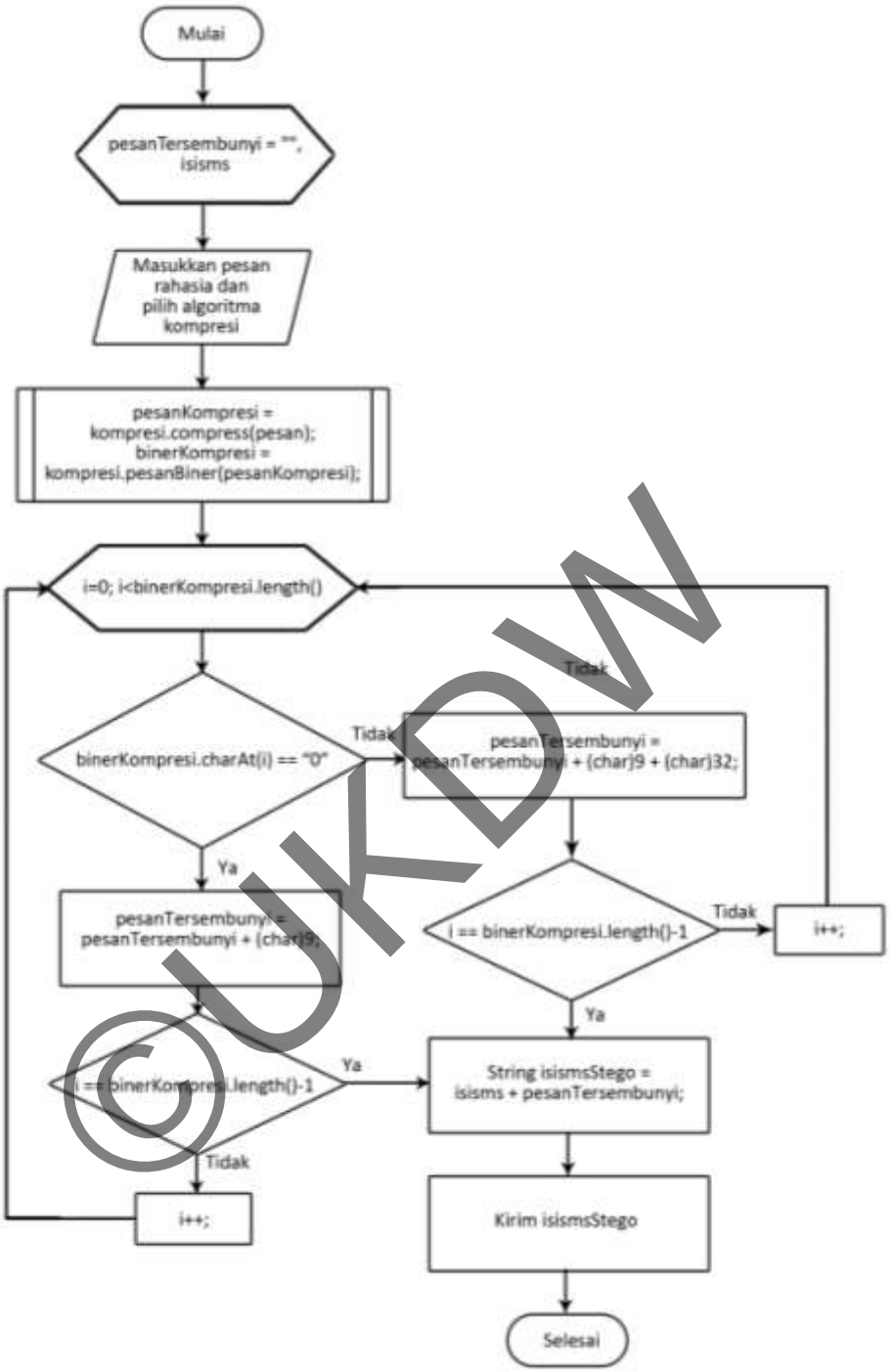
c. Proses Kompresi dan *Encoding* Steganografi

Pada bagian ini, akan dijelaskan proses penyembunyian (steganografi) pesan rahasia pada isi pesan SMS secara lebih detail. Berikut ini merupakan rangkaian prosesnya:

1. Mulai
2. Inisialisasi pesan tersembunyi dengan string kosong dan isi sms dari proses sebelumnya.
3. Masukkan pesan rahasia dan pilih algoritma kompresi yang akan digunakan.
4. Proses kompresi pesan rahasia dan proses pengubahan pesan rahasia menjadi bentuk *binary string*.
5. Cek setiap karakter pada *binary string*, jika karakter sama dengan karakter "0" maka pesan tersembunyi = pesan tersembunyi + tab, jika tidak sama dengan karakter "0" maka pesan tersembunyi = pesan tersembunyi + tab + spasi.
6. Tambahkan pesan tersembunyi di akhir *string* isi sms.
7. Kirim pesan rahasia.
8. Selesai.

Rancangan dari proses *encoding* steganografi pesan rahasia dapat dilihat pada gambar 3.3.

©UKDW



©UKDW

Gambar 3.3 *Flowchart* Kompresi dan *Encoding* Steganografi

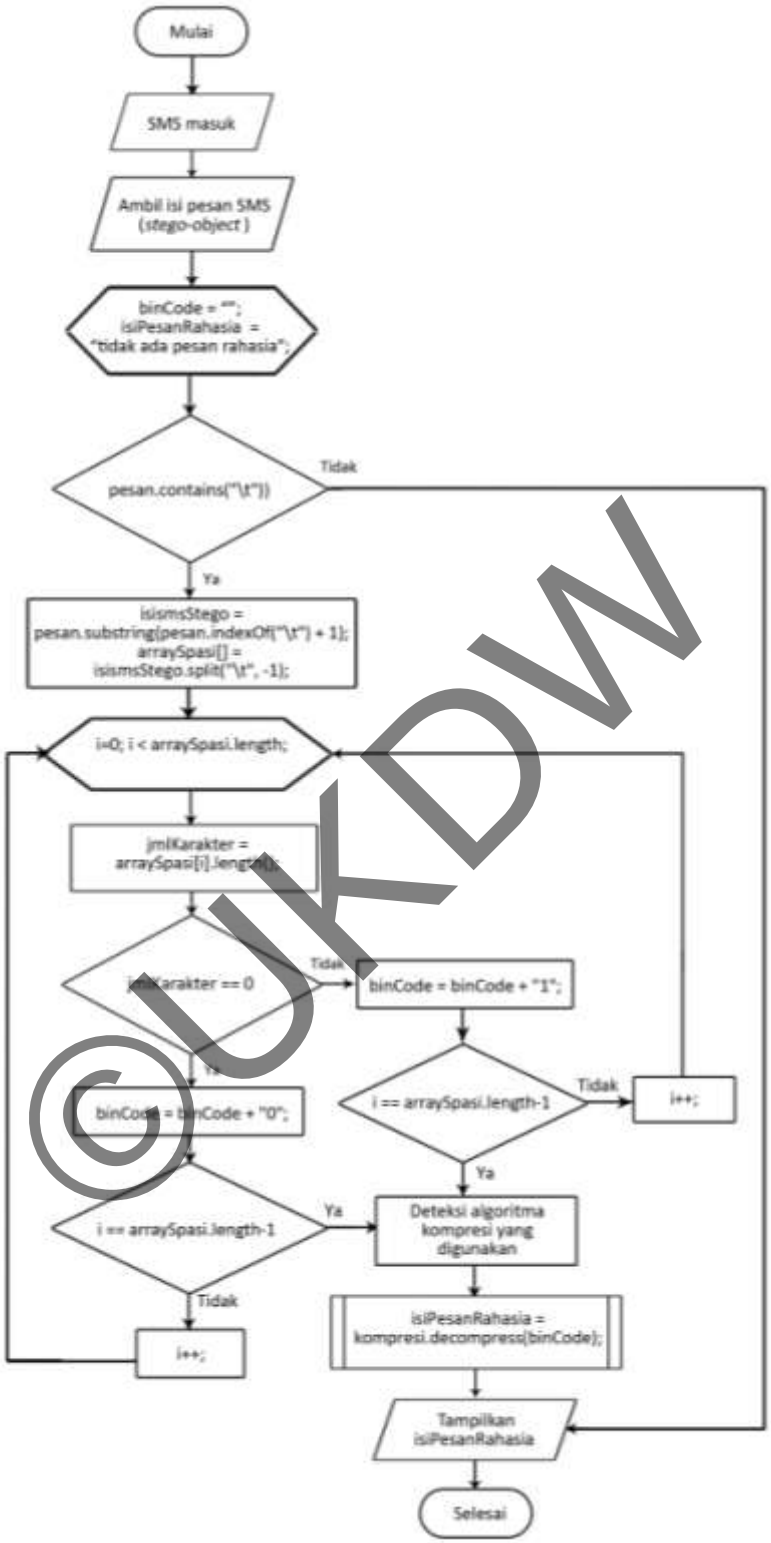
d. Proses Dekompresi dan *Decoding* Steganografi

Pada bagian ini, akan dijelaskan proses *decoding* steganografi pesan rahasia yang ada di dalam isi pesan SMS secara lebih detail. Berikut ini merupakan rangkaian prosesnya:

1. Mulai.
2. Input dari SMS masuk yang akan dibuka.
3. Ambil isi pesan SMS (*stego-object*).
4. Inisialisasi binCode dengan string kosong dan isiPesanRahasia dengan string "Tidak ada pesan rahasia.".
5. Cek apakah isi pesan SMS mengandung karakter "\t" (tab) atau tidak? Jika ya lanjutkan ke langkah berikutnya, jika tidak lompat ke langkah nomor 11.
6. Ambil pesan tersembunyi (*Embedded data*) pada isi pesan sms dengan cara mengambil deretan karakter mulai dari karakter tab pertama sampai karakter terakhir isi pesan sms.
7. Potong *embedded data* dengan dengan fungsi *split* berdasarkan karakter tab, lalu simpan dalam array arraySpasi[].
8. Cek jumlah karakter setiap data array. Jika jumlahnya sama dengan 0 maka binCode = binCode + "0", jika jumlahnya tidak sama dengan 0 maka binCode = binCode + "1".
9. Potong dua karakter terakhir pada binCode untuk mendeteksi algoritma kompresi yang digunakan.
10. Dekompresi binCode menjadi isiPesanRahasia (*embedded data*).
11. Tampilkan isiPesanRahasia.
12. Selesai.

Rancangan dari proses *encoding* steganografi pesan rahasia dapat dilihat pada gambar 3.4.

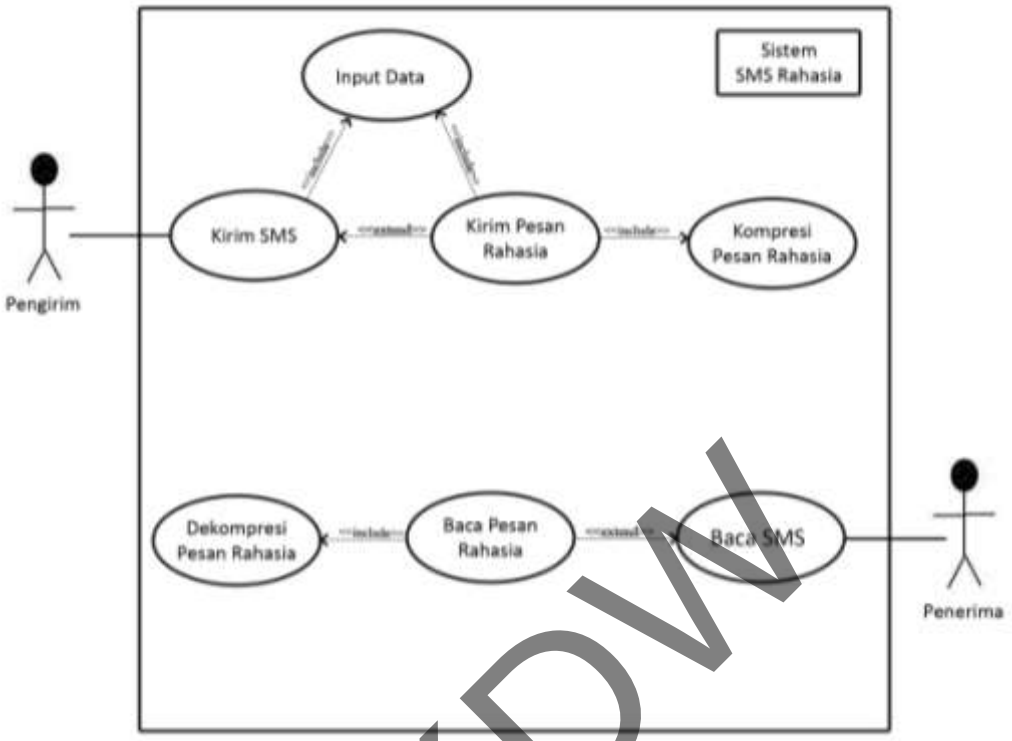
©UKDW



©UKDW

Gambar 3.4 *Flowchart* Dekompresi dan *Decoding* Steganografi

3.2.2 *Use Case* Diagram Sistem



Gambar 3.5 *Use Case* SMS Rahasia

Gambar 3.5 merupakan *Use Case* diagram yang menunjukkan gambaran kerja sistem secara garis besar. Tabel 3.1 dibawah ini merupakan deskripsi dari *Use Case* diagram pada gambar 3.5.

Tabel 3.1 Tabel Deskripsi *Use Case* SMS Rahasia

Deskripsi *Use Case*	**Keterangan *Use Case* SMS Rahasia**
Aktor	Pengguna aplikasi SMS rahasia (Pengirim dan Penerima)
Pra Kondisi	Aplikasi mengakses fungsi *messaging* pada Android untuk membuat SMS baru atau membaca SMS dalam *inbox* dan *sent item* sesuai pilihan pengguna.

©UKDW

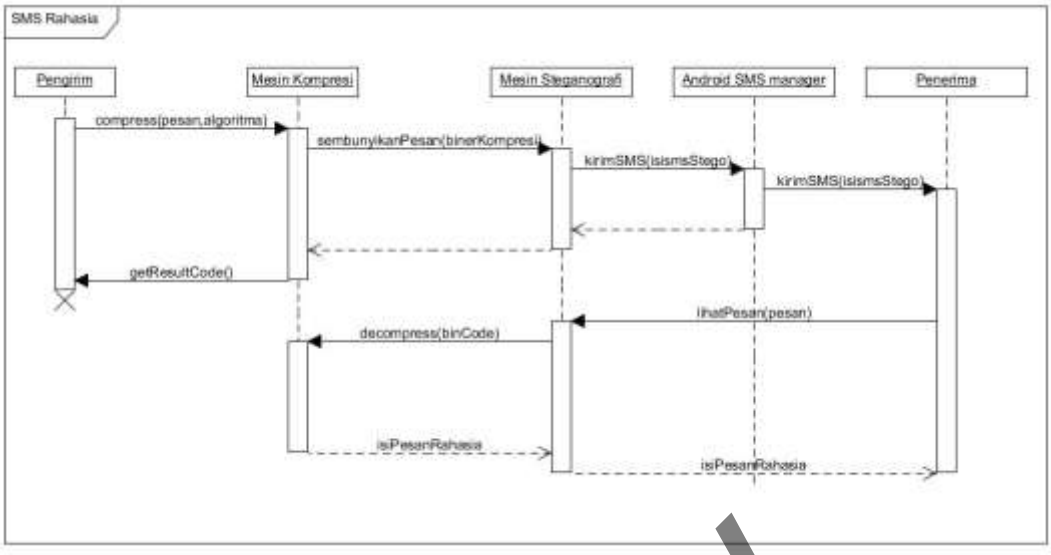
Tabel 3.1 (Sambungan)

Post Kondisi	SMS berisi pesan rahasia berupa karakter *ASCII* yang dapat dikirim melalui fasilitas *messaging* pada Android dan isi pesan rahasia dapat dibaca oleh penerima.
Alur Dasar	1. Saat pengguna memilih buat SMS, aplikasi meminta input data berupa nomor tujuan dan isi sms. 2. Saat buat SMS dipilih, pengguna memilih untuk menambahkan pesan rahasia atau tidak. 3. Jika pengguna memilih menambahkan pesan rahasia, aplikasi meminta inputan data berupa pesan rahasia dan pilihan algoritma kompresi. 4. Saat pengguna memilih baca SMS di *inbox* atau *sent item*, pengguna memilih SMS yang akan dibaca dan menentukan untuk membuka pesan rahasia atau tidak. 5. Jika pengguna memilih membuka pesan rahasia, aplikasi meminta password untuk mengakses fungsi-fungsi *decoding* steganografi dan dekompresi.
Catatan	Password tidak berhubungan dengan isi SMS, algoritma steganografi, maupun algoritma kompresi. Password yang digunakan adalah password aplikasi yang ditentukan pengguna sebelumnya.

©UKDW

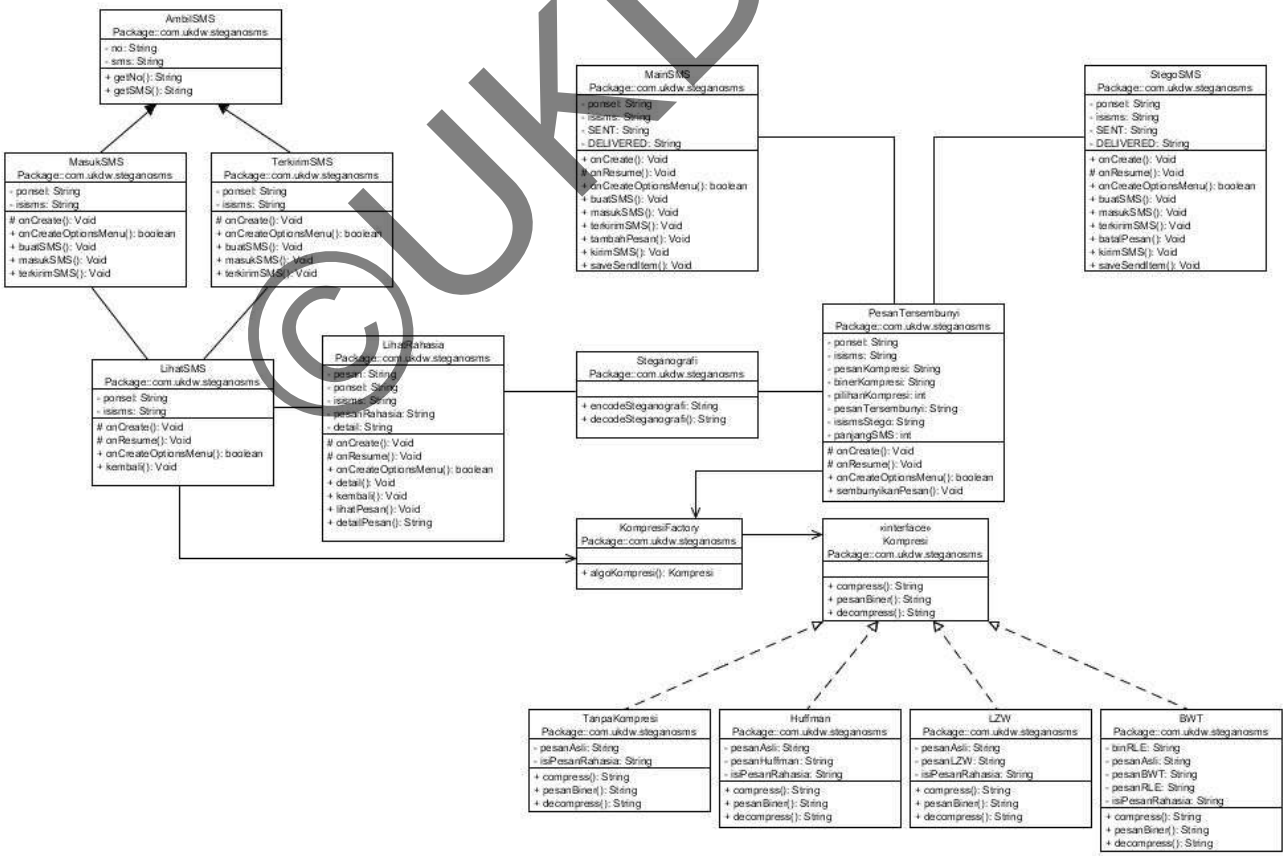
3.2.3 *Sequence* Diagram Sistem

Pada gambar 3.6 berikut ini, ditunjukkan *sequence* diagram dari sistem SMS Rahasia.



Gambar 3.6 *Sequence* Diagram SMS Rahasia

3.2.4 *Class* Diagram Sistem



Gambar 3.7 *Class* Diagram SMS Rahasia

©UKDW

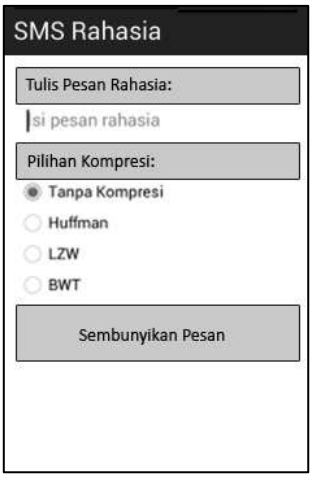
3.3 Perancangan Antar Muka Sistem

Saat aplikasi SMS Rahasia pertama kali dijalankan, pengguna akan langsung masuk ke tampilan menu Buat SMS. Terdapat 3 *button* menu utama pada *layout* bagian atas, untuk berpindah ke menu lain yaitu SMS Masuk dan SMS Terkirim. Pada *layout* bagian tengah, terdapat 2 *edit text* untuk input nomor tujuan dan isi SMS. Pada *layout* bagian bawah terdapat 2 *button* yang berfungsi untuk menambahkan pesan rahasia dan untuk mengirim SMS biasa. Berikut ini tampilan menu Buat SMS yang ditunjukkan pada gambar 3.8.

Gambar 3.8 Rancangan menu Buat SMS

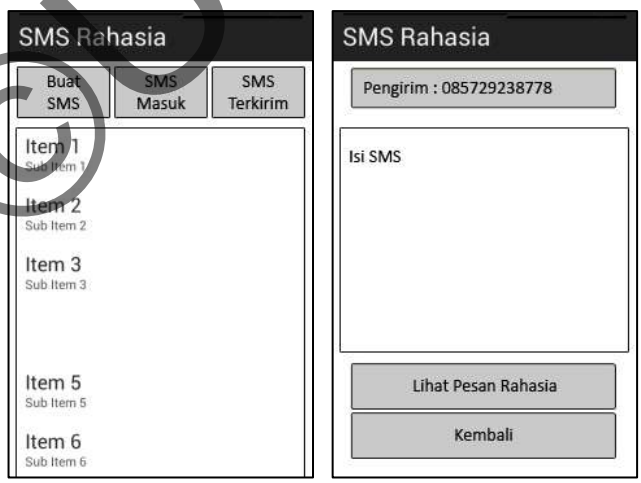
Jika pengguna menekan *button* tambah pesan rahasia, maka akan muncul tampilan menu Pesan Tersembunyi yang digunakan untuk menambah pesan rahasia, menjalankan kompresi pesan rahasia dengan algoritma pilihan pengguna, dan menyembunyikan pesan rahasia. Berikut ini tampilan menu Pesan Tersembunyi yang ditunjukkan pada gambar 3.9.

©UKDW



Gambar 3.9 Rancangan menu Pesan Tersembunyi

Untuk membuka pesan rahasia, pengguna dapat membuka menu SMS Masuk atau SMS Terkirim. Aplikasi akan menampilkan 20 SMS terakhir yang masuk untuk menu SMS Masuk dan 20 SMS terakhir yang telah terkirim untuk menu SMS Terkirim. Pengguna dapat memilih SMS yang ingin dibuka, lalu aplikasi akan menampilkan menu Lihat SMS. Berikut ini tampilan menu SMS Masuk atau SMS Terkirim dan menu Lihat SMS yang ditunjukkan pada gambar 3.10.

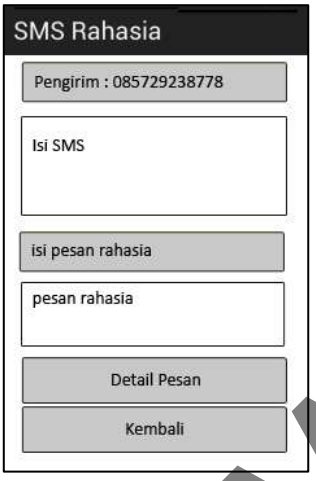


Gambar 3.10

Rancangan menu SMS Masuk / SMS Terkirim & menu Lihat SMS

©UKDW

Pada menu Lihat SMS terdapat *button* lihat pesan rahasia yang digunakan untuk menampilkan pesan rahasia. Gambar 3.11 berikut ini merupakan tampilan menu saat menampilkan pesan rahasia.



Gambar 3.11
Rancangan menu Lihat Rahasia

3.4 Perancangan Pengujian Sistem

Rancangan pengujian yang akan dilakukan menggunakan metode pengujian *black box*. Pengujian *black box* adalah pengujian yang fokus pada kebutuhan fungsional suatu perangkat lunak, tanpa perlu mengetahui apa yang terjadi pada sistem internal perangkat lunak.Dengan metode ini, penguji mencoba beberapa kondisi *input* lalu melihat apakah hasil *output*-nya seperti yang diharapkan. Beberapa kondisi *input* dan hasil *output* tersebut juga akan dianalisis dan dibandingkan.

Pengujian perangkat lunak sistem SMS Rahasia ini menggunakan data uji berupa data input dalam bentuk teks pesan rahasia dari pengguna. Tabel 3.2 berikut ini merupakan deskripsi rencana pengujian yang akan dilakukan.

©UKDW

Tabel 3.2 Tabel Deskripsi Rencana Pengujian

Item Pengujian	Deskripsi
Kirim SMS	1. Analisis pengolahan data pesan rahasia saat proses pengiriman SMS. 2. Analisis jumlah karakter maksimum pesan rahasia yang dapat dikirim melalui SMS jika isi SMS (*cover-object*) dibatasi maksimal 60 karakter.
Baca SMS	3. Analisis perbandingan rasio kompresi yang dihasilkan dari kompresi pesan rahasia dengan algoritma Huffman, LZW, dan BWT, setelah dilakukan proses *encoding* steganografi. 4. Analisis rasio kompresi yang dihasilkan dari kompresi pesan rahasia dengan algoritma BWT, setelah dilakukan proses *encoding* steganografi jika algoritma MTF saat proses kompresi BWT dihilangkan.

©UKDW

BAB IV

IMPLEMENTASI DAN ANALISIS SISTEM

4.1 Implementasi Sistem

Berikut adalah hasil dari implementasi kompresi Huffman (dengan tabel tetap), LZW, dan BWT pesan rahasia yang akan disembunyikan pada SMS dengan menggunakan *whitespace steganography* pada aplikasi SMS berbasis Android. Aplikasi dibuat berdasarkan perancangan yang telah dibuat sebelumnya, dengan sedikit adanya penambahan dan pengurangan dari rencana awal setelah menerima revisi dan perbaikan.

4.1.1 Antar Muka Pengiriman SMS Rahasia

Saat aplikasi mulai dijalankan, menu awal yang ditampilkan adalah menu Buat SMS. Berikut ini tampilan menu Buat SMS yang ditunjukkan pada gambar 4.1.

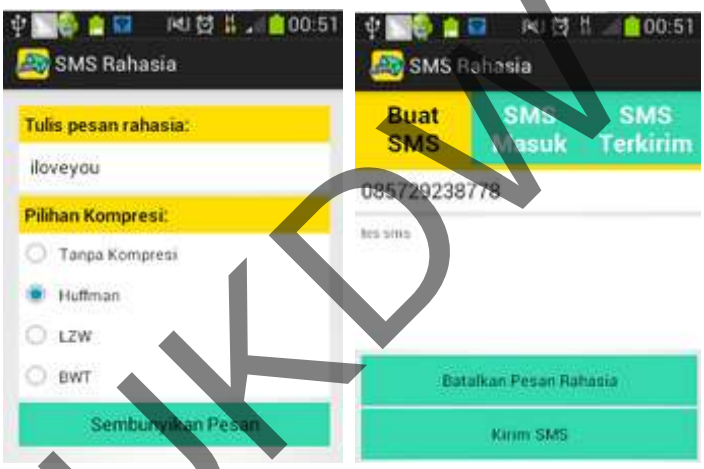
Gambar 4.1 Tampilan menu Buat SMS

Pada menu Buat SMS terdapat 2 *edit text* yaitu untuk menerima *input* nomor tujuan dan teks SMS. Terdapat juga 5 tombol, Buat SMS adalah menu untuk membuat sms baru, SMS masuk dan SMS Terkirim adalah

©UKDW

menu untuk melihat pesan yang masuk dan terkirim, tombol Tambah Pesan Rahasia untuk menambahkan pesan rahasia. Terakhir adalah tombol Kirim SMS untuk mengirim SMS ke nomor tujuan.

Saat Tambah Pesan Rahasia dipilih, akan muncul menu yang terdiri dari *edit text* untuk menambahkan pesan rahasia, *radio group* untuk memilih algoritma kompresi yang digunakan dan tombol Sembunyikan Pesan untuk menjalankan proses kompresi dan steganografi lalu menampilkan menu Stego SMS. Berikut tampilan gambar 4.2 yang menunjukkan tampilan dari menu Tambah Pesan Rahasia dan Stego SMS.



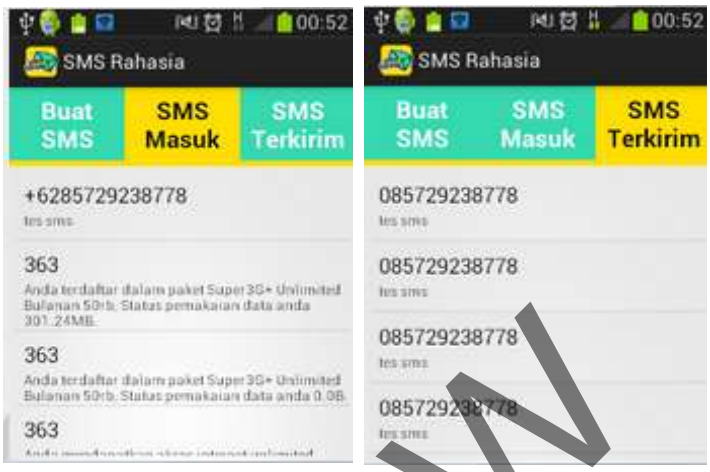
Gambar 4.2 Tampilan menu Tambah Pesan Rahasia dan Stego SMS

Setelah proses kompresi dan steganografi selesai, akan muncul menu Stego SMS seperti pada gambar 4.2 diatas yang berisi *text view* nomor tujuan dan SMS yang sudah berisi pesan rahasia, serta dua tombol untuk mengirim pesan SMS ke nomor tujuan dan untuk membatalkan pesan rahasia. Saat tombol Batalkan Pesan Rahasia dipilih, pengguna akan diarahkan kembali ke menu Buat SMS.

©UKDW

4.1.2 Antar Muka Membuka Pesan Rahasia

Saat pengguna memilih menu SMS Masuk atau SMS Terkirim, akan muncul tampilan seperti pada gambar 4.3.

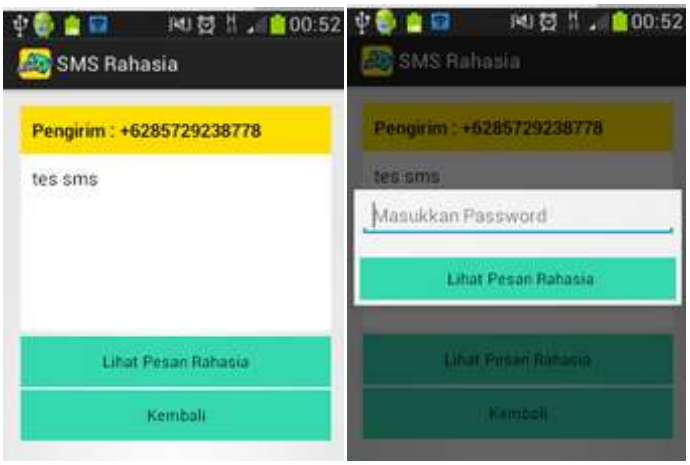


Gambar 4.3 Tampilan menu SMS Masuk dan SMS Terkirim

Sesuai gambar 4.3, pada kedua menu tersebut terdapat *list* berisi 20 SMS terakhir yang ada di *inbox* atau di *sent item.* Pengguna dapat memilih salah satu SMS pada daftar yang ingin dibuka, lalu pengguna akan diarahkan ke menu Lihat SMS.

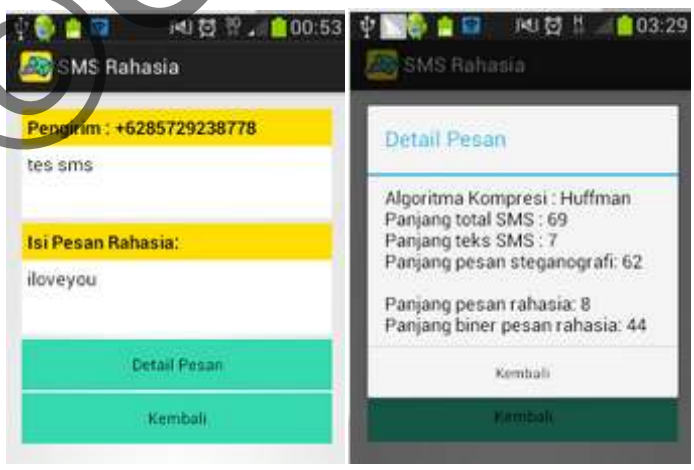
Pada menu Lihat SMS terdapat *text view* berisi SMS yang dipilih sebelumnya, terdapat juga 2 tombol yaitu tombol untuk melihat pesan rahasia dan untuk kembali ke menu SMS Masuk atau SMS Terkirim. Saat pengguna memilih Lihat Pesan Rahasia, maka akan muncul *dialog box* yang meminta pengguna untuk memasukkan *password*. Gambar 4.4 berikut ini merupakan tampilan menu Lihat SMS dan tampilan saat pengguna ingin melihat pesan rahasia.

©UKDW



Gambar 4.4 Tampilan menu Lihat SMS dan
dialog box Masukkan Password

Jika *password* yang dimasukkan pengguna benar, maka pengguna akan diarahkan ke menu Lihat Rahasia yang berisi *text view* pesan SMS dan pesan rahasia. Selain itu juga terdapat 2 tombol untuk melihat detail pesan dan untuk kembali ke menu Lihat SMS. Saat tombol Detail Pesan dipilih, maka akan muncul *dialog box* yang berisi keterangan tentang pasan SMS dan pesan rahasia. Gambar 4.5 berikut ini menunjukkan tampilan menu Lihat Rahasia dan *dialog box* Detail Pesan.



Gambar 4.5 Tampilan menu Lihat Rahasia dan
dialog box Detail Pesan

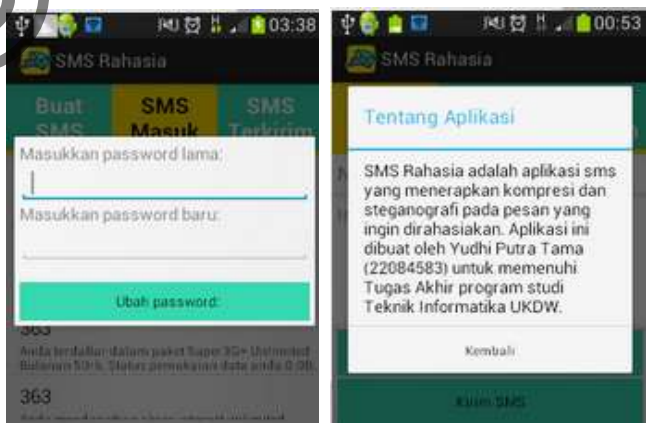
©UKDW

4.1.3 Antar Muka Option Menu

Pada menu Buat SMS, SMS Masuk, SMS Terkirim, dan Stego SMS, pengguna dapat membuka *Option* Menu yang terdiri dari menu Ubah *Password* dan Tentang Aplikasi, seperti ditunjukkan pada gambar 4.6.

Gambar 4.6 Tampilan *Option* Menu

Saat pengguna memilih menu Ubah *Password*, akan muncul *dialog box* yang meminta pengguna untuk memasukkan *password* lama dan *password* baru. Jika pengguna memilih menu Tentang Aplikasi, akan muncul *dialog box* berisi informasi tentang aplikasi. Gambar 4.7 menunjukkan tampilan *dialog box* Ubah Password dan Tentang Aplikasi.



Gambar 4.7 Tampilan *dialog box* Ubah Password dan Tentang Aplikasi

©UKDW

4.2 Analisis Sistem

4.2.1 Analisis Pengiriman SMS

e. Kompresi Pesan Rahasia

Sebelum disembunyikan, pesan rahasia yang diketikkan manual oleh pengguna, terlebih dahulu dikompresi dengan salah satu dari algoritma Huffman (dengan tabel tetap), LZW, atau BWT sesuai pilihan pengguna. Setelah proses kompresi selesai, pesan hasil dari proses kompresi diubah ke dalam bentuk *binary string* lalu ditambah kode algoritma kompresi sesuai algoritma yang dipilih pengguna sebelumnya.

Khusus algoritma BWT, proses mengubah bentuk pesan hasil kompresi ke dalam bentuk *binary string* sudah dilakukan saat proses kompresi, sehingga langkah selanjutnya tinggal menambahkan kode algoritma kompresi yang digunakan.

Setelah proses kompresi selesai dan pesan sudah dalam bentuk *binary string*, proses selanjutnya adalah menyembunyikan *binary string* pesan rahasia dengan menggunakan *whitespace steganography*. *Pseudo code* dari proses kompresi Huffman, LZW, dan BWT dapat dilihat pada lampiran A-7 sampai A-11.

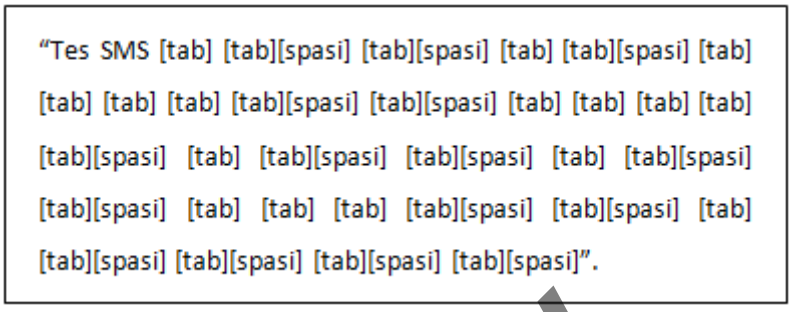
f. Steganografi Pesan Rahasia

Pada aplikasi ini penulis menggunakan *whitespace steganography* untuk menyembunyikan pesan, dengan teknik *end-of-line spaces* dimana kode steganografi diletakkan setelah karakter paling akhir pesan SMS. Pengkodean dilakukan tiap satu bit dimana bit 0 dikodekan menjadi karakter tab sedangkan bit 1 dikodekan dengan karakter tab dan spasi, sehingga proses *decoding* steganografi akan menjadi lebih mudah.

Kelemahan dari proses *whitespace steganography* ini adalah jumlah karakter yang membengkak semakin banyak karena untuk merepresentasikan satu bit saja dibutuhkan 1 sampai 2 karakter.

©UKDW

Misalkan sebuah pesan SMS “Tes SMS” dengan jumlah karakter 7, lalu disembunyikan kata “halo” dengan kode biner 01101000 01100001 01101100 01101111, maka pesan SMS yang telah ditambah pesan rahasia akan berubah menjadi :



Gambar 4.8 Hasil *whitespace steganography* kata “halo” pada pesan SMS “Tes SMS”

Hal ini tidak menjadi masalah jika *whitespace steganography* diterapkan pada file berukuran besar seperti pada *body email* atau *file document* lainnya (txt, doc, pdf, dan lain-lain), namun saat diterapkan pada SMS hal ini mengakibatkan pesan tidak dapat dikirim melalui SMS karena satu *part* SMS hanya dapat menampung 140 bytes, atau sama dengan 140 karakter 8-bit, 160 karakter 7-bit atau 70 karakter 16-bit untuk karakter-karakter dalam bahasa yang memakai aksara Kanji/Hanja.

Maka dari itu pesan SMS yang telah ditambah pesan steganografi harus dikirim dalam beberapa *part* dengan cara memecah pesan bila pesan SMS lebih dari karakter maksimal. Proses pemecahan pesan diawali dengan pengkategorian *User Data Header* yang berisi informasi segmentasi, lalu pesan dikirim dalam beberapa *part* dimana satu *part* dikirim sebagai satu pesan SMS. Sistem ponsel penerima akan menerjemahkan kembali pecahan pesan SMS tersebut menjadi satu pesan utuh kembali .

©UKDW

g. Mode Pengiriman Pesan SMS

Aplikasi ini menggunakan mode pengiriman pesan dengan fungsi *SendMultipartMessage()* dari Android, hal ini dikarenakan proses steganografi yang menghasilkan banyak karakter sehingga pesan SMS tidak akan terkirim jika menggunakan fungsi *SendTextMessage()* yang hanya bisa digunakan untuk mengirim sebuah pesan SMS.

Selain itu digunakan juga fungsi *divideMessage()* yang berfungsi untuk memecah pesan dalam beberapa *part*, sehingga pemecehan pesan tidak perlu dilakukan secara manual. Hal ini sangat membantu, karena pada pesan steganografi terdapat karakter tab yang tidak termasuk dalam *GSM 03.38 character set* yang ada pada lampiran A-6, sehingga perlu pengkodean khusus untuk mengkodekan karakter tab tersebut. Diperlukan 2 karakter untuk mengkodekan 1 karakter tab dalam *GSM 03.38 character set*. Oleh karena itu akan sulit jika pemecahan pesan SMS dilakukan secara manual, sebab banyaknya karakter tab akan mempengaruhi batas maksimal jumlah karakter dalam satu *part* SMS.

Jika pesan SMS sudah dipecah dalam beberapa bagian, diperlukan juga pengecekan berapa jumlah *part* sms yang dihasilkan oleh fungsi *divideMessage()* tersebut. Hal ini diperlukan untuk pembatasan jumlah *part* SMS yang akan dikirim, karena jumlah *part* SMS berpengaruh pada biaya SMS dan kemungkinan SMS gagal terkirim karena ada *part* SMS yang *pending* saat proses pengiriman.

©UKDW

4.2.2 Analisis Panjang Karakter Maksimum Pesan Rahasia

Pada penelitian ini jumlah *part* SMS yang dapat dikirim dengan aplikasi SMS Rahasia dibatasai maksimal 5 *part* SMS. Pembatasan ini juga berpengaruh pada jumlah karakter pesan rahasia yang dapat disembunyikan. Di dalam sub-bab ini akan dijelaskan batas maksimal pesan rahasia yang dapat disembunyikan jika panjang sms maksimal dibatasi 5 *part* SMS dan jumlah karakter pesan SMS pembawa (*Cover-object*) sebanyak 60 karakter.

Tabel 4.1 berikut merupakan contoh data hasil pengujian aplikasi SMS Rahasia dengan panjang *input* pesan rahasia lebih dari 20 karakter. Data ini diambil dari tabel pengujian yang ada pada lampiran B-2.

Tabel 4.1
Contoh Data Hasil Pengujian
Dengan Panjang Pesan Rahasia Diatas 20 Karakter

No. Pengujian	Panjang Pesan SMS	Pesan Tersembunyi	Panjang Pesan Tersembunyi	Tanpa Kompresi		Kompresi					
						Huffman		LZW		BWT	
				Panjang pesan steganografi	Panjang Total SMS	Panjang pesan steganografi	Panjang Total SMS	Panjang pesan steganografi	Panjang Total SMS	Panjang pesan steganografi	Panjang Total SMS
48	60	You will be punished!	21	345	405	163	222	245	305	267	327
49	60	What is the password?	21	347	407	163	215	247	307	271	331
50	60	I'll talk to you soon.	22	263	323	163	223	254	314	255	315
51	60	I'll call you tonight.	22	260	320	161	221	248	308	279	339
52	60	Meet me at parking lot	23	267	327	169	229	257	317	288	348
53	60	I love you very much <3	23	270	330	206	266	257	317	287	347
54	60	I will always love you	23	270	334	170	230	274	334	274	334
55	60	Prodi Teknik Informatika	24	285	345	190	250	271	331	309	369
56	60	I sent the money already	25	291	351	173	233	291	351	306	366
57	60	Don't go to Jakarta tomorrow!	29	346	406	227	287	324	384	343	403
58	60	Make sure they don't notice it	30	352	412	218	278	332	392	376	436
59	60	Universitas Kristen Duta Wacana	31	365	425	237	297	354	414	364	424
60	60	Program Studi Teknik Informatika	32	380	440	254	314	366	426	403	463
61	60	barang udah dikirim, no resi 69557	33	386	446	266	326	386	446	405	465
62	60	titip absen ya bro, bay bicor nih	33	379	439	229	289	347	407	397	457
63	60	Bapak saya tunggu di tempat biasa!	34	394	454	252	312	371	431	410	470
64	60	uang sudah saya kirim ke rekeningmu	35	415	475	264	324	379	439	404	464
65	60	Never let them to open the envelope.	36	421	481	257	317	346	406	372	432
66	60	You shall pay me for what you owe me!	37	431	491	270	330	378	438	428	488
67	60	Protect the state and vital facilities	38	439	499	264	324		error	461	521
68	60	Selasa akan ada rapat khusus pimpinan.	38	438	498	275	335	378	438	428	488
69	60	I wrote the message inside the envelope	39	458	518	273	333		error	423	483
70	60	The email show Stratfor's web of informer	41	485	545	299	359		error	484	544

©UKDW

Blok warna abu-abu pada tabel diatas menunjukkan bahwa pengiriman SMS tidak dapat dilakukan karena panjang karakter total SMS yang berisi pesan rahasia telah melebihi batas maksimal 5 *part* SMS.

Pengujian pada tabel diatas diurutkan mulai dari pesan rahasia tependek yaitu 21 karakter hingga terpanjang yaitu 41 karakter. Sehingga dari tabel diatas dapat dilihat pada pengujian ke berapa proses pengiriman SMS mulai gagal dan berapa panjang pesan rahasia yang membuat

pengiriman SMS gagal. Berikut adalah rincian kegagalan proses pengiriman SMS dan panjang maksimum pesan rahasia yang dapat dikirim untuk tiap metode kompresi:

a. Pada pesan rahasia yang disembunyikan tanpa kompresi, kegagalan pengiriman SMS mulai terjadi pada pengujian ke-54 hingga pengujian ke-70. Panjang maksimum pesan rahasia yang dapat dikirim adalah 23 karakter.
b. Pada pesan rahasia yang disembunyikan dengan kompresi Huffman yang menggunakan tabel tetap, kegagalan pengiriman SMS mulai terjadi pada pengujian ke-68 hingga pengujian ke-70. Panjang maksimum pesan rahasia yang dapat dikirim adalah 38 karakter.
c. Pada pesan rahasia yang disembunyikan dengan kompresi LZW, kegagalan pengiriman SMS mulai terjadi pada pengujian ke-56 hingga pengujian ke-70 dan pada pengujian ke-54. Panjang maksimum pesan rahasia yang dapat dikirim adalah 24 karakter.
d. Pada pesan rahasia yang disembunyikan dengan kompresi BWT, kegagalan pengiriman SMS mulai terjadi pada pengujian ke-51 hingga pengujian ke-70. Panjang maksimum pesan rahasia yang dapat dikirim adalah 22 karakter.

Pada pengujian ke-55 kompresi LZW panjang pesan rahasia 24 karakter namun masih dapat dikirim, sedangkan pada pengujian ke-54 dengan panjang pesan rahasia 23 karakter, SMS tidak dapat terkirim. Hal ini menunjukkan bahwa batas panjang karakter pesan rahasia yang dapat dikirim atau tidak dapat dikirim bukan merupakan nilai pasti melainkan sebuah rentan nilai.

Melihat panjang pesan rahasia dari data pengiriman SMS yang gagal pada tabel diatas dan melihat selisih total SMS yang berkisar antara 1 sampai 39 karakter. Sedangkan untuk penkodean steganografi tiap satu karakter pesan rahasia rata-rata dikodekan menjadi 20 karakter, sehingga dalam rentan selisih total SMS 1 sampai 39, ada kemungkinan

©UKDW

penambahan atau pengurangan 2 karakter pesan rahasia. Dari analisis tersebut, dapat diperkirakan rentan panjang maksimum pesan rahasia yang dapat dikirim. Tabel 4.2 berikut merupakan rentan panjang karakter maksimum pesan rahasia yang dapat dikirim aplikasi SMS Rahasia.

Tabel 4.2

Panjang Karakter Maksimum Pesan Rahasia

Kompresi	Panjang Maximum Pesan Rahasia
Tanpa Kompresi	21-25 karakter
Huffman	36-40 karakter
LZW	22-26 karakter
BWT	20-24 karakter

4.2.3 Analisis Rasio Kompresi Pesan Rahasia

Berikut adalah hasil dari pengujian rasio kompresi dari algoritma Huffman, LZW, dan BWT pada pesan rahasia. Dalam pengujian ini rasio kompresi dibedakan menjadi dua yaitu rasio kompresi berdasarkan jumlah bit pesan rahasia yang dihasilkan dari proses kompresi dan rasio kompresi berdasarkan panjang karakter hasil pengkodean steganografi dari data biner pesan rahasia.

Rasio kompresi dibedakan menjadi dua seperti penjelasan diatas karena pada penerapan steganografi, biner 0 dikodekan menjadi satu karakter yaitu tab dan biner, 1 dikodekan menjadi dua karakter yaitu tab spasi, perbedaan jumlah karakter saat pengkodean ini mengakibatkan jumlah karakter hasil steganografi akan berbeda walaupun jumlah bit nya sama, tergantung perbandingan jumlah biner 0 dengan 1.

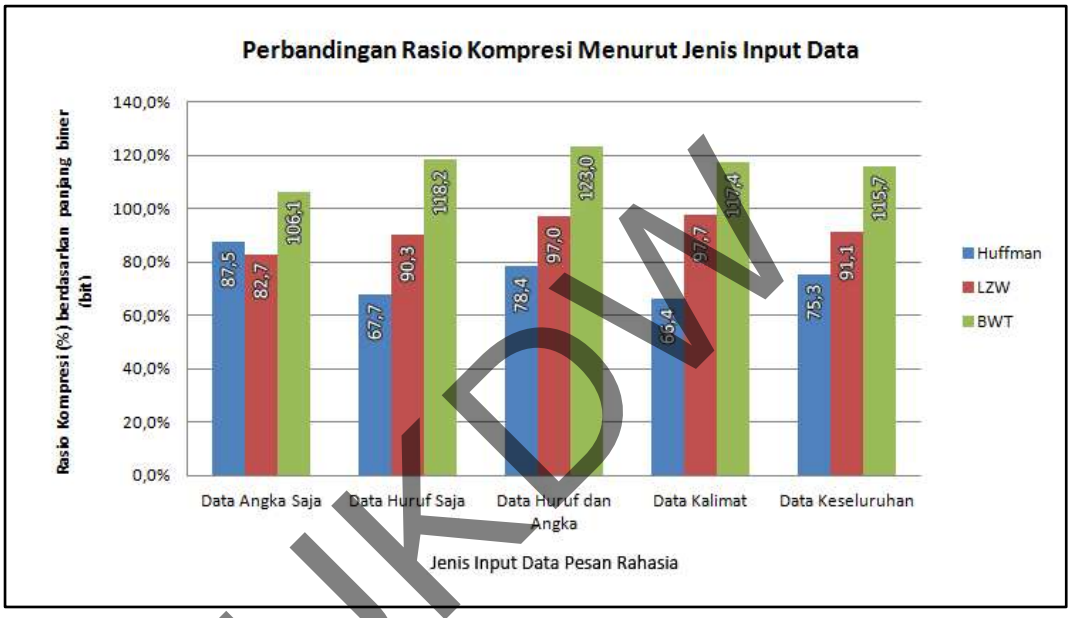
Analisis rasio kompresi ini berdasarkan data pengujian nomor 1 sampai 50 dari tabel pengujian pada lampiran B-1 dan B-2. Berikut ini adalah analisis data hasil pengujian yang dibedakan berdasarkan jenis input data teks pesan rahasia diantaranya angka saja, huruf saja, huruf dan

©UKDW

angka, kalimat normal, serta data pengujian keseluruhan dari nomor pengujian 1 sampai 50:

a. Analisis Rasio Kompresi Berdasarkan Panjang Bit

Gambar 4.9 berikut ini adalah diagram perbandingan rasio kompresi Huffman, LZW, dan BWT dalam persen (%) berdasarkan panjang bit kode biner pesan rahasia:



Gambar 4.9 Diagram perbandingan rasio kompresi (%) berdasarkan panjang bit

©UKDW

Dari gambar 4.9 terlihat bahwa dengan rasio kompresi rata-rata 75,3%, Huffman memiliki rasio kompresi paling baik dibanding rasio kompresi LZW dan BWT kecuali pada jenis input data yang berupa angka saja.

Pada jenis input data angka saja algoritma LZW memiliki rasio kompresi yang paling baik, hal ini dikarenakan karakter jenis angka hanya memiliki 10 jenis karakter sehingga kemungkinan terjadi perulangan karakter lebih besar.

Algoritma BWT memiliki rasio kompresi paling buruk, bahkan panjang bit pesan rahasia hasil kompresi BWT sebagian besar

melebihi panjang bit pesan rahasia tanpa kompresi. Dengan kata lain kompresi menggunakan BWT menjadikan pesan rahasia menjadi semakin besar. Hal ini kemungkinan disebabkan karena rasio kompresi BWT sangat tergantung pada banyaknya karakter yang sama. Sedangkan untuk pesan rahasia yang pada umumnya pendek, kemungkinan munculnya karakter yang sama sangat rendah.

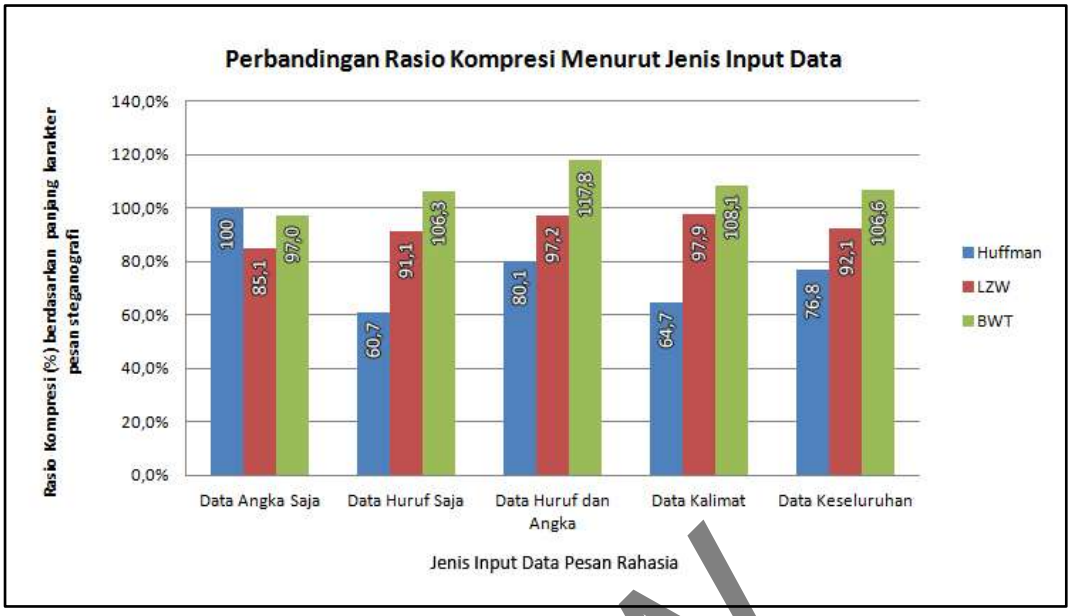
Hasil perbandingan rasio kompresi pada penelitian ini berbeda dengan tinjauan pustaka yang digunakan pada BAB II yaitu sebuah studi berjudul *Data Compression Techniques on Text Files: A Comparison Study* yang dimuat di *International Journal of Computer Applications* (0975 – 8887), *volume* 26 *no* 5, halaman 42-54. Dalam studi tersebut diambil kesimpulan bahwa LZW merupakan algoritma yang paling baik untuk kompresi teks, lebih baik dari Huffman.

Perbedaan hasil penelitian ini dikarenakan pada penelitian ini kompresi Huffman menggunakan tabel Huffman tetap sehingga tabel Huffman tidak perlu dikirimkan bersama data kompresi. Selain itu jumlah karakter teks yang dikompresi pada penelitian ini lebih sedikit, sehingga penggunaan LZW menjadi tidak optimal karena kemungkinan terdapat perulangan kombinasi karakter yang rendah.

b. Analisis Rasio Kompresi Berdasarkan Panjang Pesan Steganografi

Gambar 4.10 berikut ini adalah diagram perbandingan rasio kompresi Huffman, LZW, dan BWT dalam persen (%) berdasarkan panjang karakter hasil pengkodean steganografi dari data biner pesan rahasia:

©UKDW



Gambar 4.10 Diagram perbandingan rasio kompresi (%) berdasarkan panjang karakter pesan steganografi

Perbandingan rasio kompresi berdasarkan panjang karakter pesan steganografi tidak terlalu berbeda dengan rasio kompresi berdasarkan bit. Rasio kompresi Huffman tetap yang paling baik, kecuali pada data angka saja. Rasio kompresi LZW tetap paling baik untuk jenis input data angka saja.

Yang menarik untuk dicermati adalah hubungan antara rasio kompresi berdasarkan bit dengan rasio kompresi berdasarkan panjang karakter pesan steganografi. Pada LZW rasio kompresi berdasarkan bit lebih baik dibanding rasio kompresi berdasarkan panjang karakter pesan steganografi, pada Huffman pun demikian kecuali pada jenis input data huruf saja dan kalimat normal.

Namun pada BWT terjadi sebaliknya, rasio kompresi berdasarkan panjang karakter pesan steganografi lebih baik dibanding rasio kompresi berdasarkan bit. Kemungkinan hal ini disebabkan karena proses kompresi BWT menghasilkan kode biner yang lebih banyak bit 0 sehingga pengkodean steganografi menjadi lebih sedikit.

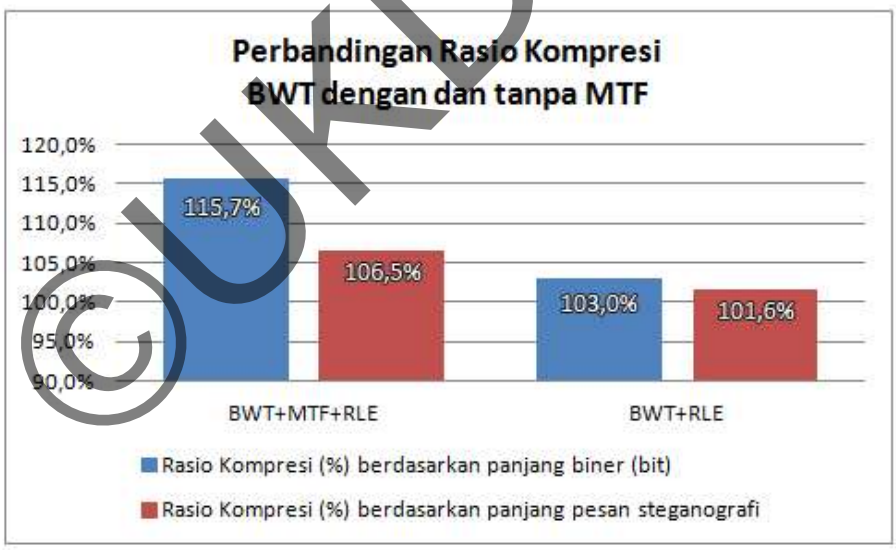
©UKDW

4.2.4 Analisis Algoritma BWT Tanpa MTF

Konsep kompresi BWT adalah mentransformasikan data dengan algoritma BWT sebelum data dikompresi dengan algoritma kompresi *entropy coding*. Dengan transformasi BWT diharapkan data teks dapat lebih terpola, dengan karakter yang sama menjadi berjajar.

Pada penelitian ini proses kompresi BWT menggunakan kombinasi algoritma transformasi BWT dan MTF sebelum dikompresi dengan algoritma RLE. Berdasarkan hasil pengujian kompresi BWT menghasilkan rasio kompresi paling buruk dibandingkan kompresi Huffman dan LZW.

Hal ini menimbulkan pertanyaan seberapa besar pengaruh penambahan algoritma MTF pada kompresi BWT. Maka dari itu dilakukan pengujian untuk mengetahui rasio kompresi BWT jika tanpa menggunakan MTF dibandingkan dengan menggunakan MTF, dan hasilnya dapat dilihat pada gambar 4.11 berikut:



Gambar 4.11 Diagram perbandingan rasio kompresi (%)
BWT dengan dan tanpa MTF

Dari diagram pada gambar 4. 11, terlihat ternyata rasio kompresi BWT tanpa menggunakan MTF lebih baik dibandingkan jika menggunakan MTF. Hal ini mungkin dikarenakan data teks yang dikompresi yang pada umumnya

©UKDW

pendek sehingga penggunaan MTF berakibat merusak pola data teks yang sebelumnya sudah ditransformasi dengan algoritma BWT.

Walaupun kompresi BWT tanpa MTF memiliki rasio kompresi yang lebih baik daripada yang menggunakan MTF, namun perbedaan rasionya tidak terlalu jauh dan tetap lebih buruk dibandingkan kompresi menggunakan Huffman dan LZW. Sehingga hal ini tidak mempengaruhi hasil pengujian sebelumnya yang telah menunjukkan bahwa Huffman memiliki rasio kompresi terbaik dibanding LZW dan BWT.

4.3 Kelebihan dan Kekurangan Sistem

Aplikasi yang dibuat ini masih belum sempurna sehingga memiliki beberapa kelebihan dan juga kekurangan.

4.3.1 Kelebihan Sistem

Beberapa kelebihan yang dimiliki aplikasi ini adalah:

a. Penggunaan *whitespace steganography* yang masih jarang digunakan sehingga lebih tidak mencurigakan dibanding pengamanan data dengan enkripsi yang sangat mencolok.
b. Terdapat *password* aplikasi untuk membuka pesan rahasia, sehingga pesan rahasia menjadi lebih aman.
c. Pengguna dapat memilih algoritma kompresi yang akan digunakan.
d. Saat membaca pesan rahasia, aplikasi dapat mendeteksi secara otomatis algoritma kompresi yang digunakan pengirim.

4.3.2 Kekurangan Sistem

Selain memiliki kelebihan, aplikasi ini juga memiliki beberapa kelemahan yaitu:

a. Nomor tujuan harus diketikkan manual.
b. Biaya pengiriman SMS jadi lebih mahal.
c. Tidak ada fasilitas *delete* dan *forward* pada *inbox* dan *sent item.*
d. Tidak ada fasilitas *reset password* jika pengguna lupa *password*.

©UKDW

e. Tidak ada notifikasi saat ada SMS masuk.
f. Jika dilihat menggunakan beberapa aplikasi SMS yang ada di pasaran, pesan SMS yang memiliki pesan rahasia terlihat memiliki ruang kosong yang panjang karena pengaruh adanya karakter tab.

©UKDW

©UKDW

LAMPIRAN A

- Referensi Program

Tabel Tetap Huffman Untuk Bahasa Inggris

Sumber : http://www.kejut.com/kompresisms

	Awalan	Badan	Ubahan
a		0000	00000
b		00101	000101
c		1101	01101
d		01011	001011
f		11001	011001
g		10101	010101
h		1000	01000
i		0001	00001
j		110001001	0110001001
k		1100011	01100011
l		1001	01001
m		10100	010100
n	0	0110	00110
o		0100	00100
p		10110	010110
q		1100010000	01100010000
r		0111	00111
s		0011	00011
t		111	0111
u		10111	010111
v		0101001	00101001
w		0101000	00101000
x		11000101	011000101
y		00100	000100
z		1100010001	01100010001
A		0000	1000000
B		00101	10000101
C		1101	1001101
D		01011	10001011
F		11001	10011001
G		10101	10010101
H		1000	1001000
I		0001	1000001
J		110001001	100110001001
K		1100011	1001100011
L		1001	1001001
M		10100	10010100
N	100	0110	1000110
O		0100	1000100
P		10110	10010110
Q		1100010000	1001100010000
R		0111	1000111
S		0011	1000011
T		111	100111
U		10111	10010111
V		0101001	1000101001
W		0101000	1000101000
X		11000101	10011000101
Y		00100	10000100
Z		1100010001	1001100010001

©UKDW

Tabel Tetap Huffman (Sambungan)

<space>	101	1	1011
e		0	1010
!	110	0011	1100011
"		0110	1100110
'		0100	1100100
,		0001	1100001
-		0101	1100101
.		0000	1100000
?		0010	1100010
E		0111	1100111
0	111	0000	1110000
1		0001	1110001
3		0011	1110011
4		0100	1110100
5		0101	1110101
6		0110	1110110
7		0111	1110111
8		1000	1111000
9		1001	1111001
(		11010	11111010
)		11011	11111011
/		1011	1111011
2		0010	1110010
:		1100	1111100
@		1010	1111010
SISA HURUF		1111110	1111110

©UKDW

ASCII control characters (character code 0-31)

Sumber : http://www.ascii-code.com/

DEC	OCT	HEX	BIN	Symbol	HTML Number	HTML Name	Description
0	000	00	00000000	NUL	&#000;		Null char
1	001	01	00000001	SOH	&#001;		Start of Heading
2	002	02	00000010	STX	&#002;		Start of Text
3	003	03	00000011	ETX	&#003;		End of Text
4	004	04	00000100	EOT	&#004;		End of Transmission
5	005	05	00000101	ENQ	&#005;		Enquiry
6	006	06	00000110	ACK	&#006;		Acknowledgment
7	007	07	00000111	BEL	&#007;		Bell
8	010	08	00001000	BS	&#008;		Back Space
9	011	09	00001001	HT			Horizontal Tab
10	012	0A	00001010	LF
	Line Feed
11	013	0B	00001011	VT	&#011;		Vertical Tab
12	014	0C	00001100	FF			Form Feed
13	015	0D	00001101	CR
	Carriage Return
14	016	0E	00001110	SO	&#014;		Shift Out / X-On
15	017	0F	00001111	SI	&#015;		Shift In / X-Off
16	020	10	00010000	DLE	&#016;		Data Line Escape
17	021	11	00010001	DC1	&#017;		Device Control 1 (oft. XON)
18	022	12	00010010	DC2	&#018;		Device Control 2
19	023	13	00010011	DC3	&#019;		Device Control 3 (oft. XOFF)
20	024	14	00010100	DC4	&#020;		Device Control 4
21	025	15	00010101	NAK	&#021;		Negative Acknowledgement
22	026	16	00010110	SYN	&#022;		Synchronous Idle
23	027	17	00010111	ETB	&#023;		End of Transmit Block
24	030	18	00011000	CAN	&#024;		Cancel
25	031	19	00011001	EM	&#025;		End of Medium
26	032	1A	00011010	SUB	&#026;		Substitute
27	033	1B	00011011	ESC	&#027;		Escape
28	034	1C	00011100	FS	&#028;		File Separator
29	035	1D	00011101	GS	&#029;		Group Separator
30	036	1E	00011110	RS	&#030;		Record Separator
31	037	1F	00011111	US	&#031;		Unit Separator

©UKDW

ASCII printable characters (character code 32-127)

Sumber : http://www.ascii-code.com/

DEC	OCT	HEX	BIN	Symbol	HTML Number	HTML Name	Description
32	040	20	00100000				Space
33	041	21	00100001	!	!		Exclamation mark
34	042	22	00100010	"	"	"	Double quotes (or speech marks)
35	043	23	00100011	#	#		Number
36	044	24	00100100	$	$		Dollar
37	045	25	00100101	%	%		Procenttecken
38	046	26	00100110	&	&	&	Ampersand
39	047	27	00100111	'	'		Single quote
40	050	28	00101000	(	(		Open parenthesis (or open bracket)
41	051	29	00101001	)	)		Close parenthesis (or close bracket)
42	052	2A	00101010	*	*		Asterisk
43	053	2B	00101011	+	+		Plus
44	054	2C	00101100	,	,		Comma
45	055	2D	00101101	-	-		Hyphen
46	056	2E	00101110	.	.		Period, dot or full stop
47	057	2F	00101111	/	/		Slash or divide
48	060	30	00110000	0	0		Zero
49	061	31	00110001	1	1		One
50	062	32	00110010	2	2		Two
51	063	33	00110011	3	3		Three
52	064	34	00110100	4	4		Four
53	065	35	00110101	5	5		Five
54	066	36	00110110	6	6		Six
55	067	37	00110111	7	7		Seven
56	070	38	00111000	8	8		Eight
57	071	39	00111001	9	9		Nine
58	072	3A	00111010	:	:		Colon
59	073	3B	00111011	;	;		Semicolon
60	074	3C	00111100	<	<	<	Less than (or open angled bracket)
61	075	3D	00111101	=	=		Equals
62	076	3E	00111110	>	>	>	Greater than (or close angled bracket)
63	077	3F	00111111	?	?		Question mark
64	100	40	01000000	@	@		At symbol
65	101	41	01000001	A	A		Uppercase A
66	102	42	01000010	B	B		Uppercase B
67	103	43	01000011	C	C		Uppercase C
68	104	44	01000100	D	D		Uppercase D
69	105	45	01000101	E	E		Uppercase E
70	106	46	01000110	F	F		Uppercase F
71	107	47	01000111	G	G		Uppercase G
72	110	48	01001000	H	H		Uppercase H
73	111	49	01001001	I	I		Uppercase I
74	112	4A	01001010	J	J		Uppercase J
75	113	4B	01001011	K	K		Uppercase K
76	114	4C	01001100	L	L		Uppercase L
77	115	4D	01001101	M	M		Uppercase M
78	116	4E	01001110	N	N		Uppercase N
79	117	4F	01001111	O	O		Uppercase O
80	120	50	01010000	P	P		Uppercase P
81	121	51	01010001	Q	Q		Uppercase Q
82	122	52	01010010	R	R		Uppercase R
83	123	53	01010011	S	S		Uppercase S
84	124	54	01010100	T	T		Uppercase T
85	125	55	01010101	U	U		Uppercase U

©UKDW

ASCII printable characters (Sambungan)

86	126	56	01010110	V	V		Uppercase V
87	127	57	01010111	W	W		Uppercase W
88	130	58	01011000	X	X		Uppercase X
89	131	59	01011001	Y	Y		Uppercase Y
90	132	5A	01011010	Z	Z		Uppercase Z
91	133	5B	01011011	[	[		Opening bracket
92	134	5C	01011100	\	\		Backslash
93	135	5D	01011101	]	]		Closing bracket
94	136	5E	01011110	^	^		Caret - circumflex
95	137	5F	01011111	_	_		Underscore
96	140	60	01100000	`	`		Grave accent
97	141	61	01100001	a	a		Lowercase a
98	142	62	01100010	b	b		Lowercase b
99	143	63	01100011	c	c		Lowercase c
100	144	64	01100100	d	d		Lowercase d
101	145	65	01100101	e	e		Lowercase e
102	146	66	01100110	f	f		Lowercase f
103	147	67	01100111	g	g		Lowercase g
104	150	68	01101000	h	h		Lowercase h
105	151	69	01101001	i	i		Lowercase i
106	152	6A	01101010	j	j		Lowercase j
107	153	6B	01101011	k	k		Lowercase k
108	154	6C	01101100	l	l		Lowercase l
109	155	6D	01101101	m	m		Lowercase m
110	156	6E	01101110	n	n		Lowercase n
111	157	6F	01101111	o	o		Lowercase o
112	160	70	01110000	p	p		Lowercase p
113	161	71	01110001	q	q		Lowercase q
114	162	72	01110010	r	r		Lowercase r
115	163	73	01110011	s	s		Lowercase s
116	164	74	01110100	t	t		Lowercase t
117	165	75	01110101	u	u		Lowercase u
118	166	76	01110110	v	v		Lowercase v
119	167	77	01110111	w	w		Lowercase w
120	170	78	01111000	x	x		Lowercase x
121	171	79	01111001	y	y		Lowercase y
122	172	7A	01111010	z	z		Lowercase z
123	173	7B	01111011	{	{		Opening brace
124	174	7C	01111100	|				Vertical bar
125	175	7D	01111101	}	}		Closing brace
126	176	7E	01111110	~	~		Equivalency sign - tilde
127	177	7F	01111111		&#127;		Delete

©UKDW

The GSM 03.38 7 bit character set

Sumber : https://dynmark.uservoice.com/knowledgebase/articles/83496-gsm-alphabet-character-set

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	@	16	10	?	32	20	SP	48	30	0
1	1	£	17	11	_	33	21	!	49	31	1
2	2	$	18	12	F	34	22	"	50	32	2
3	3	¥	19	13	G	35	23	#	51	33	3
4	4	è	20	14	?	36	24	¤	52	34	4
5	5	é	21	15	O	37	25	%	53	35	5
6	6	ù	22	16	?	38	26	&	54	36	6
7	7	ì	23	17	?	39	27	'	55	37	7
8	8	ò	24	18	S	40	28	(	56	38	8
9	9	Ç	25	19	T	41	29	)	57	39	9
10	A	LF	26	1A	?	42	2A	*	58	3A	:
11	B	Ø	27	1B	<ESC>	43	2B	+	59	3B	;
12	C	ø	28	1C	Æ	44	2C	,	60	3C	<
13	D	CR	29	1D	æ	45	2D	-	61	3D	=
14	E	Å	30	1E	ß	46	2E	.	62	3E	>
15	F	å	31	1F	É	47	2F	/	63	3F	?
Dec	**Hex**	**Char**	**Dec**	**Hex**	**Char**	**Dec**	**Hex**	**Char**	**Dec**	**Hex**	**Char**
64	40	¡	80	50	P	96	60	¿	112	70	P
65	41	A	81	51	Q	97	61	a	113	71	Q
66	42	B	82	52	R	98	62	b	114	72	R
67	43	C	83	53	S	99	63	c	115	73	S
68	44	D	84	54	T	100	64	d	116	74	T
69	45	E	85	55	U	101	65	e	117	75	u
70	46	F	86	56	V	102	66	f	118	76	v
71	47	G	87	57	W	103	67	g	119	77	w
72	48	H	88	58	X	104	68	h	120	78	x
73	49	I	89	59	Y	105	69	i	121	79	y
74	4A	J	90	5A	Z	106	6A	j	122	7A	z
75	4B	K	91	5B	Ä	107	6B	k	123	7B	ä
76	4C	L	92	5C	Ö	108	6C	l	124	7C	ö
77	4D	M	93	5D	Ñ	109	6D	m	125	7D	ñ
78	4E	N	94	5E	Ü	110	6E	n	126	7E	ü
79	4F	O	95	5F	§	111	6F	o	127	7F	à

©UKDW

Pseudo Code Program

a. Pseudo Code Proses Kompresi Huffman

```
Mulai
Inisialisasi String[] tabHuff;
Inisialisasi String diLuarTabel = "1111110";
Inisialisasi String pesanKompresi;
Buat tabel dari tabel huffman tetap dan simpan di tabHuff;
For i kurang dari panjang pesanRahasia
        Inisialisasi char karakterPesan = karakter pesanRahasia ke-i;
        If tabHuff[karakterPesan] tidak sama dengan null
                pesanKompresi = pesanKompresi + tabHuff[karakterPesan];
        Else
                Inisialisasi String karLain = kode biner karakterPesan yang ditambah 0
                        didepannya agar panjangnya menjadi 16;
                pesanKompresi = pesanKompresi + diLuarTabel + karLain;
End For;
Return pesanKompresi;
Selesai
```

b. Pseudo Code Proses Dekompresi Huffman

```
Mulai
Inisialisasi String[] tabHuff;
Inisialisasi String diLuarTabel = "1111110";
Inisialisasi String isiPesanRahasia;
Inisialisasi String tampung;
Inisialisasi String byteKompresi = kode biner output dari proses decoding steganografi;
Buat tabel dari tabel huffman tetap dan simpan di tabHuff;
For i kurang dari panjang byteKompresi
        Inisialisasi char karBiner = karakter byteKompresi ke-i;
        tampung = tampung+karBiner;
        Inisialisasi int c = hasil pencarian string biner tampung pada tabHuff;
        If c lebih dari sama dengan 0
                tampung = "";
                isiPesanRahasia = isiPesanRahasia + karakter representasi dari integer c;
        Else if c sama dengan -2
                tampung = "";
End For;
Return isiPesanRahasia;
Selesai
```

©UKDW

c. Pseudo Code Proses Kompresi LZW

```
Mulai
Inisialisasi  String[] dictLZW;
Inisialisasi  int minDict = 31;
Inisialisasi String in;
Inisialisasi String out;
Inisialisasi String tampung;
Inisialisasi  String pesanKompresi;
        Buat kamus untuk  semua karakter  ASCII dan simpan di dictLZW;
        out = karakter pertama pesanRahasia;
        pesanKompresi = out;
        For i = 1, i kurang  dari panjang  pesanRahasia
                In = karakter ke-i pesanRahasia;
                tampung = out+in;
                If tampung ada dalam kamus  dictLZW
                        pesanKompresi = (pesanKompresi – karakter terakhir pesanKompresi) +
                                        karakter dalam kamus;
                        out = tampung;
                Else
                        Tambah tampung ke dictLZW[]
                        pesanKompresi = pesanKompresi + in;
                        out = in;
        End For
Return pesanKompresi;
Selesai
```

©UKDW

d. Pseudo Code Proses Dekompresi LZW

```
Mulai
Inisialisasi  String[] dictLZW;
Inisialisasi  int minDict = 31;
Inisialisasi String in;
Inisialisasi String out;
Inisialisasi String tampung;
Inisialisasi  String isiPesanRahasia;
Inisialisasi  String byteKompresi = String biner output dari proses decoding steganografi;
Inisialisasi String pesanLZW = String yang dibentuk dari String biner byteKompresi;
        Buat kamus untuk semua karakter ASCII dan simpan di dictLZW;
        out = karakter pertama pesanLZW;
        isiPesanRahasia = out;
        For i = 1, i kurang dari panjang pesanLZW
                If karakter ke-i pesanLZW ada dalam kamus dictLZW
                        In = karakter ke-i pesanLZW;
                        isiPesanRahasia = isiPesanRahasia+in;
                        Inisialisasi String in2= karakter pertama String In;
                        tampung = out+in2;
                        Tambahkan tampung ke dalam kamus dictLZW;
                        out = in;
                        minDict--;
                Else
                        Inisialisasi String out2 = karakter pertama String Out;
                        tampung = out+out2;
                        Tambahkan tampung ke dalam kamus dictLZW;
                        out = tampung;
                        isiPesanRahasia = isiPesanRahasia+tampung;
                        minDict--;
        End For
Return isiPesanRahasia;
Selesai
```

©UKDW

e. Pseudo Code Proses Kompresi BWT (BWT → MTF → RLE)

```
Mulai
Inisialisasi String binRLE = "0";
Inisialisasi String pesanBWT;
Inisialisasi String pesanAsli;
Inisialisasi String[] arrayKarakter;
Inisialisasi Integer indexPesanAsli;
Inisialisasi  String pesanKompresi;
        pesanAsli = pesanRahasia;
        Buat  seluruh rotasi setiap permutasi dari pesanRahasia dan simpan di arrayKarakter;
        Urutkan arrayKarakter sesuai abjad;
        indexPesanAsli = urutan data di arrayKarakter yang datanya sama dengan pesanAsli;
        pesanBWT = karakter terakhir tiap data di arrayKarakter;

Inisialisasi Integer indexKarakter=0;
Inisialisasi Integer kar;
Inisialisasi ArrayList<Character> arrayKarakterMTF;
Inisialisasi Integer[] dict;
Inisialisasi String pesanMTF;
        Buat daftar berisi integer 0 sampai 128 dan simpan di dict;
        Simpan tiap karakter pesanBWT didalam arrayKarakterMTF;
        For char x tiap data pada arrayKarakterMTF
                kar = nilai integer dari x;
                Bandingkan kar dengan data di dict;
                If dict[i] sama dengan kar;
                        indexKarakter = i;
                For idx = indexKarakter, idx != 0, decrement idx
                        dict[idx] = dict[idx-1];
                End For
                dict[0] = kar;
                Tambahkan (char) indexKarakter ke pesanMTF;
        End For
        pesanMTF = pesanMTF + indexPesanAsli;

Inisialisasi String pesanRLE = karakter pertama pesanMTF;
        For int i = 1, i kurang dari panjang pesanMTF, increment i
                If karakter ke i sama dengan karakter ke (i-1)
                        Tambahkan string "1"  pada akhir binRLE;
                Else
                        Tambahkan string "0"  pada akhir binRLE;
                        Tambahkan karakter ke i pesanMTF pada akhir pesanRLE;
        End for
        pesanKompresi = kode biner pesanRLE + string "1" + binRLE ;

Return pesanKompresi;
Selesai
```

©UKDW

f. Pseudo Code Proses Dekompresi BWT (RLE → MTF → BWT)

```
Mulai
Inisialisasi String binCode2;
Inisialisasi String kodeRLE;
Inisialisasi String pesanMTF;
Inisialisasi String pesanBWT;
Inisialisasi String pesanRahasia;
Inisialisasi  String byteKompresi = String biner output dari proses decoding steganografi;
Inisialisasi String pesanRLE = String yang dibentuk dari karakter ke-0 hingga karakter sebelum
                               karakter kelipatan 8 yang bernilai "1" dari String biner byteKompresi;
        Inisialisasi Integer idxKodeRLE = panjang karakter pesanRLE*8 + 1;
        kodeRLE = karakter ke- idxKodeRLE hingga karakter terakhir dari String biner
        byteKompresi;
        Inisialisasi Integer j=0;
        For integer i = 0, i kurang dari panjang kodeRLE, increment i
                If karakter ke-i kodeRLE sama dengan karakter "0"
                        pesanMTF = pesanMTF + karakter ke-j pesanRLE;
                        increment j;
                Else
                        pesanMT F= pesanMTF + karakter ke-(j-1) pesanRLE;
        End for
Inisialisasi Character idxBWT = karakter terakhir pesanMTF;
Inisialisasi String mtfOnly = String pesanMTF tanpa karaker idxBWT;
Inisialisasi ArrayList<Integer> data = array berisi nilai integer tiap karakter pada mtfOnly;
Inisialisasi Integer[] dict;
        For integer dataVal tiap data pada ArrayList data;
                Inisialisasi Character b = karakter ASCII dari nilai integer dataVal;
                For check = dataVal, check!= 0, decrement check
                        dict[check] = dict[check-1];
                End For
                dict[0] = b;
                pesanBWT = pesanBWT+ b;
        End For
        pesanBWT = pesanBWT + idxBWT;
Inisialisasi Character idxAsli= karakter terakhir pesanBWT;
Inisialisasi String bwtOnly = String pesanBWT tanpa karaker idxAsli;
Inisalisasi String[] arrayKarakter = array string berisi tiap karakter pada bwtOnly;
Inisalisasi String[] arrayKarakter2 = array berisi string tiap karakter pada bwtOnly;
        For integer i = 0, i kurang dari panjang bwtOnly - 1, increment i
                Urutkan arrayKarakter2 sesuai abjad;
                For integer j = 0, i kurang dari panjang bwtOnly, increment j
                        Inisialisasi String tamp1 = data String ke-j pada arrayKarakter;
                        Inisialisasi String tamp2 = data String ke-j pada arrayKarakter2;
                        arrayKarakter[j] = tamp1 + karakter terakhir tamp2
                        arrayKarakter2[j] = arrayKarakter[j];
                End For
        End For
        Urutkan arrayKarakter sesuai abjad;
        pesanRahasia = data ke-idxAsli dari arrayKarakter;
Return pesanRahasia;
Selesai
```

©UKDW

©UKDW

LAMPIRAN B

- Tabel dan Grafik Hasil Pengujian

Pengujian dengan panjang pesan SMS 60 karakter

No. Pengujian	Panjang Pesan SMS	Pesan Tersembunyi	Panjang Pesan Tersembunyi	Tanpa Kompresi			Algoritma Kompresi														
							Huffman					LZW					BWT				
				Panjang biner pesan tersembunyi (bit)	Panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS
1	60	123456	6	48	69	129	42	87,5%	69	100,0%	129	48	100,0%	69	100,0%	129	64	133,3%	87	126,1%	147
2	60	123123	6	48	68	128	42	87,5%	68	100,0%	128	40	83,3%	59	86,8%	119	64	133,3%	77	113,2%	157
3	60	12345678	8	64	93	153	56	87,5%	93	100,0%	153	64	100,0%	93	100,0%	153	82	128,1%	112	120,4%	172
4	60	11111111	8	64	88	148	56	87,5%	88	100,0%	148	32	50,0%	49	55,7%	109	34	53,1%	47	53,4%	107
5	60	666666	6	48	72	132	42	87,5%	72	100,0%	132	24	50,0%	37	51,4%	97	32	66,7%	43	59,7%	103
6	60	112233	6	48	68	128	42	87,5%	68	100,0%	128	48	100,0%	68	100,0%	128	64	133,3%	78	114,7%	138
7	60	1313	4	32	46	106	28	87,5%	46	100,0%	106	24	75,0%	36	78,3%	96	46	143,8%	55	119,6%	115
8	60	0007	4	32	43	103	28	87,5%	43	100,0%	103	24	75,0%	36	83,7%	96	30	93,8%	41	95,3%	101
9	60	08011991	8	64	88	148	56	87,5%	88	100,0%	148	64	100,0%	88	100,0%	148	74	115,6%	94	106,8%	154
10	60	007007007	9	72	99	159	63	87,5%	99	100,0%	159	48	66,7%	70	70,7%	130	51	70,8%	66	66,7%	126
11	60	password	8	64	99	159	45	70,3%	61	61,6%	121	64	100,0%	99	100,0%	159	82	128,1%	115	116,2%	175
12	60	qwerty	6	48	75	135	38	79,2%	52	69,3%	112	48	100,0%	75	100,0%	135	56	116,7%	84	112,0%	144
13	60	football	8	64	98	158	41	64,1%	55	56,1%	115	64	100,0%	98	100,0%	158	82	128,1%	105	107,1%	165
14	60	blahblah	8	64	90	150	42	65,6%	52	57,8%	112	48	75,0%	70	77,8%	130	82	128,1%	100	111,1%	160
15	60	pepper	6	48	69	129	31	64,6%	47	68,1%	107	40	83,3%	59	85,5%	119	64	133,3%	79	114,5%	139
16	60	iloveyou	8	64	103	163	44	68,8%	59	57,3%	119	64	100,0%	103	100,0%	163	74	115,6%	107	103,9%	167
17	60	sunshine	8	64	100	160	40	62,5%	56	56,0%	116	64	100,0%	100	100,0%	160	82	128,1%	116	116,0%	176
18	60	cheese	6	48	72	132	27	56,3%	39	54,2%	99	48	100,0%	72	100,0%	132	64	133,3%	84	116,7%	144
19	60	starwars	8	64	98	158	42	65,6%	57	58,2%	115	56	87,5%	87	88,8%	147	82	128,1%	110	112,2%	170
20	60	whatever	8	64	97	157	43	67,2%	59	60,8%	119	64	100,0%	97	100,0%	157	82	128,1%	119	122,7%	179
21	60	abc123	6	48	68	128	37	77,1%	55	80,9%	115	48	100,0%	68	100,0%	128	64	133,3%	88	129,4%	148
22	60	yudhi0801	9	72	102	162	56	77,8%	80	78,4%	140	72	100,0%	102	100,0%	162	91	126,4%	126	123,5%	186
23	60	jamesbond007	12	96	143	203	73	76,0%	103	72,0%	163	96	100,0%	143	100,0%	203	110	114,6%	154	107,7%	214
24	60	seras008	8	64	91	151	46	71,9%	63	69,2%	123	64	100,0%	91	100,0%	151	82	128,1%	109	119,8%	169
25	60	jakarta48	9	72	104	164	56	77,8%	78	75,0%	134	72	100,0%	104	100,0%	164	91	126,4%	125	120,2%	185
26	60	08januari1991	13	104	151	211	83	79,8%	122	80,8%	182	104	100,0%	151	100,0%	211	127	122,1%	172	113,9%	232
27	60	dn4m4t1c	8	64	93	153	47	73,4%	73	78,5%	133	64	100,0%	93	100,0%	153	82	128,1%	115	123,7%	175
28	60	p455w0rd	8	64	93	153	53	82,8%	81	87,1%	141	64	100,0%	93	100,0%	153	82	128,1%	120	129,0%	180
29	60	blink182	8	64	94	154	50	78,1%	73	77,7%	133	64	100,0%	94	100,0%	154	82	128,1%	118	125,5%	178
30	60	ab12ab12ab12	12	96	132	292	75	78,1%	105	79,5%	165	64	66,7%	92	69,7%	152	86	89,6%	105	79,5%	165
31	60	banana	6	48	70	130	31	64,6%	37	52,9%	97	40	83,3%	58	82,9%	118	48	100,0%	62	88,6%	122
32	60	dutawacana	10	80	119	179	54	67,5%	71	59,7%	131	80	100,0%	119	100,0%	179	92	115,0%	130	109,2%	190
33	60	Hahahaha	8	64	84	144	48	75,0%	56	66,7%	116	40	62,5%	57	67,9%	117	50	78,1%	65	75,0%	123
34	60	kaukapakapa	11	88	129	189	67	76,1%	89	69,0%	149	64	72,7%	94	72,9%	154	77	87,5%	100	77,5%	160
35	60	Pendaftaran	10	80	115	175	54	67,5%	73	63,5%	133	72	90,0%	104	90,4%	164	100	125,0%	129	112,2%	189
36	60	13840100010753 2	15	120	165	225	105	87,5%	165	100,0%	225	104	86,7%	146	88,5%	206	129	107,5%	163	98,8%	223
37	60	085729238778	12	96	140	200	84	87,5%	140	100,0%	200	96	100,0%	140	100,0%	200	110	114,6%	142	101,4%	202
38	60	74566982	8	64	92	152	57	89,1%	94	102,2%	154	64	100,0%	92	100,0%	152	82	128,1%	114	123,9%	174
39	60	340406801910003	16	128	173	233	112	87,5%	173	100,0%	233	112	87,5%	154	89,0%	214	138	107,8%	173	100,0%	233
40	60	500600	6	48	64	124	42	87,5%	64	100,0%	124	40	83,3%	56	87,5%	116	40	83,3%	52	81,3%	112

Keterangan :

Blok warna : berdasarkan jumlah kode biner (bit)

Angka Merah : nilai rasio kompresi terbaik tiap data

Blok warna : berdasarkan panjang pesan steganografi

Blok warna : proses kirim SMS gagal

Blok warna : panjang total SMS

© ITDel

Pengujian dengan panjang pesan SMS 60 karakter (Sambungan)

No. Pengujian	Panjang Pesan SMS	Pesan Tersembunyi	Panjang Pesan Tersembunyi	Tanpa Kompresi			Algoritma Kompresi																
							Huffman					LZW					BWT						
				Panjang biner pesan tersembunyi (bit)	Panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS
41	60	Where are you?	14	112	167	227	74	66,1%	105	62,9%	165	104	92,9%	154	92,2%	214	136	121,4%	186	111,4%	246
42	60	Please call me!	15	120	170	230	76	63,3%	109	64,1%	169	120	100,0%	170	100,0%	230	145	120,8%	190	111,8%	250
43	60	When can we meet?	17	136	201	261	88	64,7%	126	62,7%	186	128	94,1%	190	94,5%	250	163	119,9%	215	107,0%	275
44	60	What's the number?	18	144	213	273	96	66,7%	138	64,8%	198	144	100,0%	213	100,0%	273	164	113,9%	225	105,6%	285
45	60	Tomorrow, at 9 PM.	18	144	211	271	103	71,5%	149	70,6%	209	144	100,0%	211	100,0%	271	172	119,4%	232	110,0%	292
46	60	Where's the meeting?	20	160	239	299	102	63,8%	148	61,9%	208	152	95,0%	228	95,4%	288	190	118,8%	255	106,7%	315
47	60	Don't save this SMS!	20	160	231	291	111	69,4%	160	69,3%	220	160	100,0%	231	100,0%	291	190	118,8%	262	113,4%	322
48	60	You will be punished!	21	168	245	305	113	67,3%	162	66,1%	222	168	100,0%	245	100,0%	305	191	113,7%	267	109,0%	327
49	60	What is the password?	21	168	247	307	111	66,1%	155	62,8%	215	168	100,0%	247	100,0%	307	199	118,5%	271	109,7%	331
50	60	I'll talk to you soon.	22	176	263	323	115	65,3%	163	62,0%	223	168	95,5%	254	96,6%	314	192	109,1%	255	97,0%	315
51	60	I'll call you tonight.	22	176	260	320	114	64,8%	161	61,9%	221	168	95,5%	248	95,4%	308		0,0%	279	107,3%	339
52	60	Meet me at parking lot.	23	184	267	327	116	63,0%	169	63,3%	229	176	95,7%	257	96,3%	317		0,0%	288	107,9%	348
53	60	I love you very much <3	23	184	270	330	141	76,6%	206	76,3%	266	176	95,7%	257	95,2%	317		0,0%	287	106,3%	347
54	60	I will always love you.	23	184	274	334	126	68,5%	170	62,0%	230		0,0%	274	100,0%	334		0,0%	274	100,0%	334
55	60	Prodi Teknik Informatika	24	192	285	345	134	69,8%	190	66,7%	250	184	95,8%	271	95,1%	331		0,0%	309	108,4%	369
56	60	I sent the money already.	25	200	291	351	123	61,5%	173	59,5%	233		0,0%	291	100,0%	351		0,0%	306	105,2%	366
57	60	Don't go to Jakarta tomorrow!	29	232	346	406	159	68,5%	227	65,6%	287		0,0%	324	93,6%	384		0,0%	343	99,1%	403
58	60	Make sure they don't notice it	30	240	352	412	148	61,7%	218	61,9%	278		0,0%	332	94,3%	392		0,0%	376	106,8%	436
59	60	Universitas Kristen Duta Wacana	31	248	365	425	167	67,3%	237	64,9%	297		0,0%	354	97,0%	414		0,0%	364	99,7%	424
60	60	Program Studi Teknik Informatika	32	256	380	440	177	69,1%	254	66,8%	314		0,0%	365	96,1%	426		0,0%	403	106,1%	463
61	60	barang udah dikirim, no resi 69557	33	264	386	446	181	68,6%	266	68,9%	326		0,0%	386	100,0%	446		0,0%	405	104,9%	465
62	60	titip absen ya bro, ban bocor nih	33	264	379	439	164	62,1%	229	60,4%	289		0,0%	347	91,6%	407		0,0%	397	104,7%	457
63	60	Besok saya tunggu di tempat biasa!	34	272	394	454	177	65,1%	252	64,0%	312		0,0%	371	94,2%	431		0,0%	410	104,1%	470
64	60	uang sudah saya kirim ke rekeningmu	35	280	415	475	185	66,1%	264	63,6%	324		0,0%	379	91,3%	439		0,0%	404	97,3%	464
65	60	Never let them to open the envelope.	36	288	421	481	174	60,4%	257	61,0%	317		0,0%	346	82,2%	406		0,0%	372	88,4%	432
66	60	You shall pay me for what you owe me!	37	296	431	491	192	64,9%	270	62,6%	330		0,0%	378	87,7%	438		0,0%	428	99,3%	488
67	60	Protect the state and vital facilities	38	304	439	499	182	59,9%	264	60,1%	324		0,0%		0,0%	error		0,0%	461	105,0%	521
68	60	Selasa akan ada rapat khusus pimpinan.	38	304	438	498		0,0%	275	62,8%	335		0,0%	378	86,3%	438		0,0%	428	97,7%	488
69	60	I wrote the message inside the envelope.	39	312	458	518		0,0%	273	59,6%	333		0,0%		0,0%	error		0,0%	423	92,4%	483
70	60	The email show Stratfor's web of informer	41	328	485	545		0,0%	299	61,6%	359		0,0%		0,0%	error		0,0%	484	99,8%	544

Keterangan :

Blok warna : berdasarkan jumlah kode biner (bit)

Angka Merah : nilai rasio kompresi terbaik tiap data

Blok warna : berdasarkan panjang pesan steganografi

Blok warna : proses kirim SMS gagal

Blok warna : panjang total SMS

UNDIP

Tabel dan Grafik Rasio Kompresi (%) Berdasarkan Panjang Biner (Bit) Pesan Rahasia

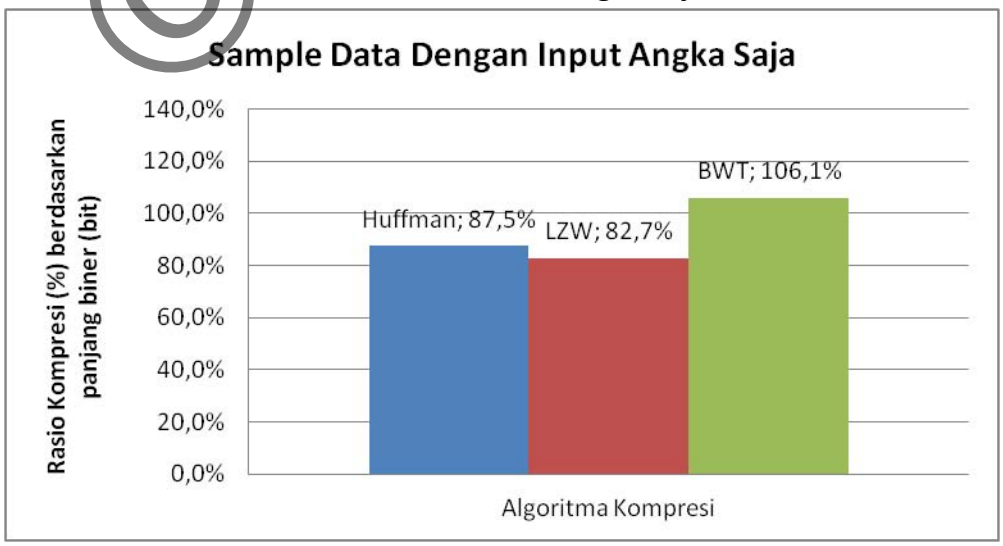
1. Contoh Data Angka Saja

Tabel Contoh Data Angka Saja

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
1	123456	87,5%	100,0%	133,3%
2	123123	87,5%	83,3%	133,3%
3	12345678	87,5%	100,0%	128,1%
4	11111111	87,5%	50,0%	53,1%
5	666666	87,5%	50,0%	66,7%
6	112233	87,5%	100,0%	133,3%
7	1313	87,5%	75,0%	143,8%
8	0007	87,5%	75,0%	93,8%
9	08011991	87,5%	100,0%	115,6%
10	007007007	87,5%	66,7%	70,8%
36	138401000107532	87,5%	86,7%	107,5%
37	085729238778	87,5%	100,0%	114,6%
39	3404060801910003	87,5%	87,5%	107,8%
40	500600	87,5%	83,3%	83,3%
Rata-rata		**87,5%**	**82,7%**	**106,1%**

©UKDW

Grafik Contoh Data Angka Saja

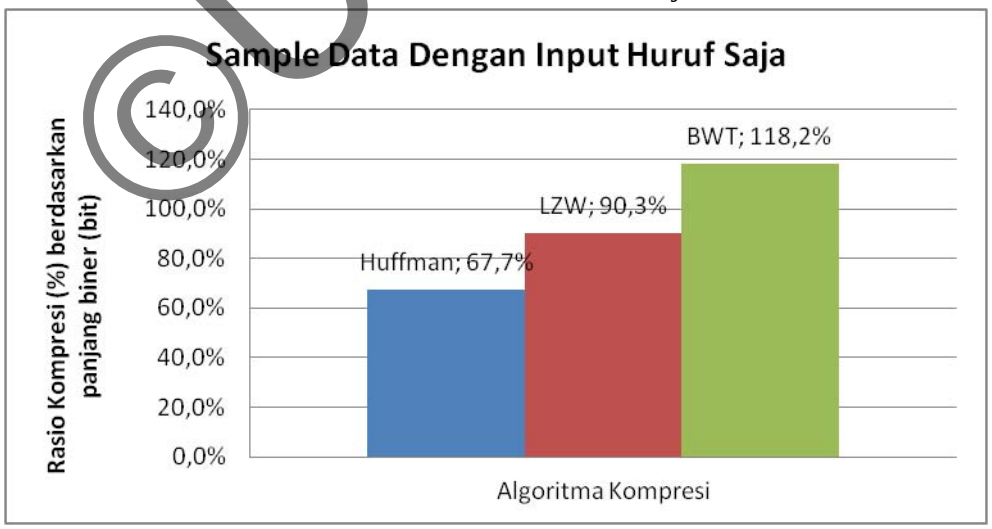


2. Contoh Data Huruf Saja

Tabel Contoh Data Huruf Saja

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
11	password	70,3%	100,0%	128,1%
12	qwerty	79,2%	100,0%	116,7%
13	football	64,1%	100,0%	128,1%
14	blahblah	65,6%	75,0%	128,1%
15	pepper	64,6%	83,3%	133,3%
16	iloveyou	68,8%	100,0%	115,6%
17	sunshine	62,5%	100,0%	128,1%
18	cheese	56,3%	100,0%	133,3%
19	starwars	65,6%	87,5%	128,1%
20	whatever	67,2%	100,0%	128,1%
31	banana	64,6%	83,3%	100,0%
32	dutawacana	67,5%	100,0%	115,0%
33	HaHaHaHa	75,0%	62,5%	78,1%
34	kaukapakapa	76,1%	72,7%	87,5%
35	Pendadaran	67,5%	90,0%	125,0%
Rata-rata		**67,7%**	**90,3%**	**118,2%**

Grafik Contoh Data Huruf Saja



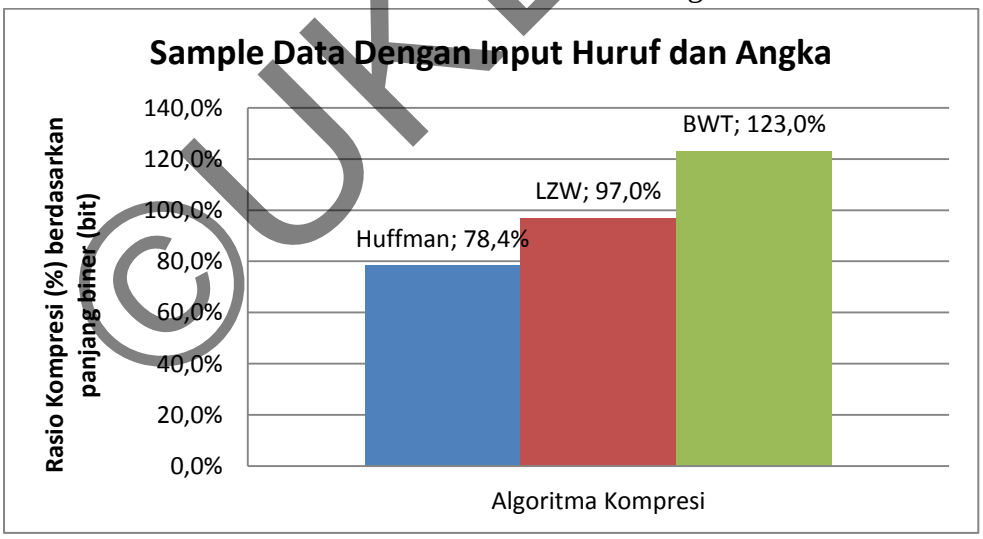
©UKDW

3. Contoh Data Huruf dan Angka

Tabel Contoh Data Huruf dan Angka

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
21	abc123	77,1%	100,0%	133,3%
22	yudhi0801	77,8%	100,0%	126,4%
23	jamesbond007	76,0%	100,0%	114,6%
24	saras008	71,9%	100,0%	128,1%
25	jakarta48	77,8%	100,0%	126,4%
26	08januari1991	79,8%	100,0%	122,1%
27	dr4m4t1c	73,4%	100,0%	128,1%
28	p455w0rd	82,8%	100,0%	128,1%
29	blink182	78,1%	100,0%	128,1%
30	ab12ab12ab12	78,1%	66,7%	89,6%
38	7456E9B2	89,1%	100,0%	128,1%
Rata-rata		**78,4%**	**97,0%**	**123,0%**

Grafik Contoh Data Huruf dan Angka



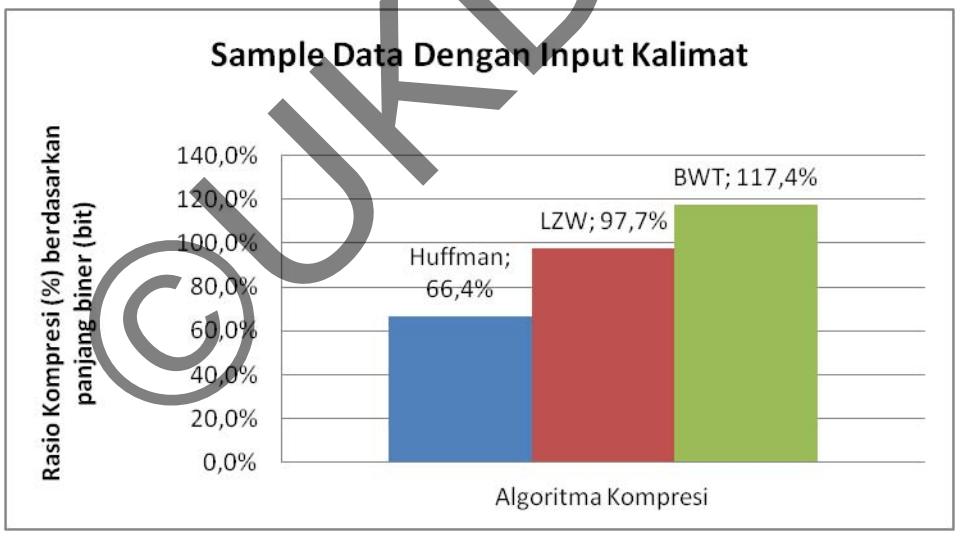
©UKDW

4. Contoh Data Kalimat Normal

Tabel Contoh Data Kalimat Normal

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
41	Where are you?	66,1%	92,9%	121,4%
42	Please call me!	63,3%	100,0%	120,8%
43	When can we meet?	64,7%	94,1%	119,9%
44	What's the number?	66,7%	100,0%	113,9%
45	Tomorrow, at 9 PM.	71,5%	100,0%	119,4%
46	Where's the meeting?	63,8%	95,0%	118,8%
47	Don't save this SMS!	69,4%	100,0%	118,8%
48	You will be punished!	67,3%	100,0%	113,7%
49	What is the password?	66,1%	100,0%	118,5%
50	I'll talk to you soon.	65,3%	95,5%	109,1%
Rata-rata		**66,4%**	**97,7%**	**117,4%**

Grafik Contoh Data Kalimat Normal



©UKDW

5. Contoh Data Keseluruhan No. Pengujian 1-50

Tabel Contoh Data Keseluruhan No. Pengujian 1-50

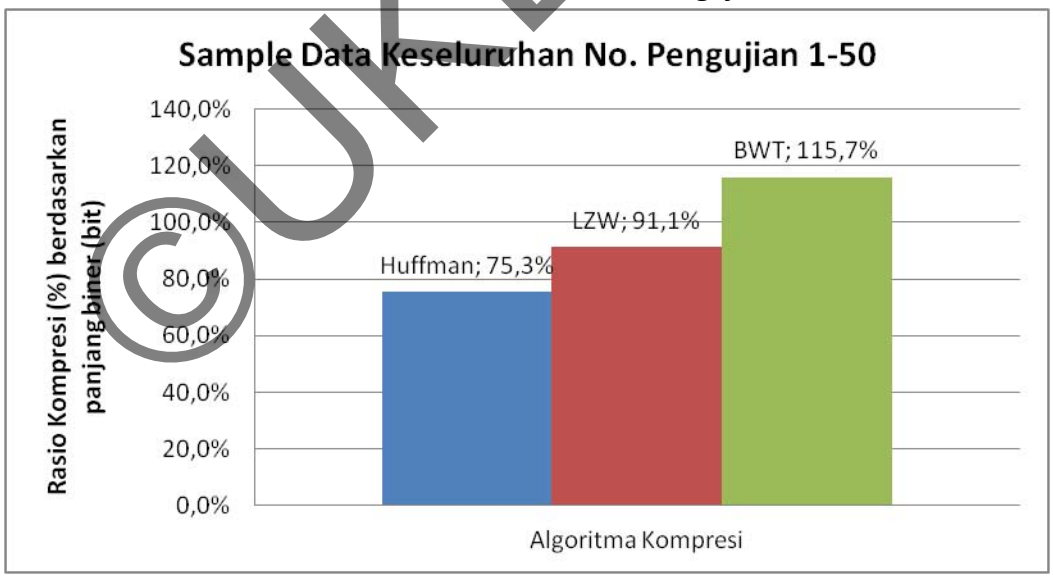
Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
1	123456	87,5%	100,0%	133,3%
2	123123	87,5%	83,3%	133,3%
3	12345678	87,5%	100,0%	128,1%
4	11111111	87,5%	50,0%	53,1%
5	666666	87,5%	50,0%	66,7%
6	112233	87,5%	100,0%	133,3%
7	1313	87,5%	75,0%	143,8%
8	0007	87,5%	75,0%	93,8%
9	08011991	87,5%	100,0%	115,6%
10	007007007	87,5%	66,7%	70,8%
11	password	70,3%	100,0%	128,1%
12	qwerty	79,2%	100,0%	116,7%
13	football	64,1%	100,0%	128,1%
14	blahblah	65,6%	75,0%	128,1%
15	pepper	64,6%	83,3%	133,3%
16	iloveyou	68,8%	100,0%	115,6%
17	sunshine	62,5%	100,0%	128,1%
18	cheese	56,3%	100,0%	133,3%
19	starwars	65,6%	87,5%	128,1%
20	whatever	67,2%	100,0%	128,1%
21	abc123	77,1%	100,0%	133,3%
22	yudhi0801	77,8%	100,0%	126,4%
23	jamesbond007	76,0%	100,0%	114,6%
24	saras008	71,9%	100,0%	128,1%
25	jakarta48	77,8%	100,0%	126,4%
26	08januari1991	79,8%	100,0%	122,1%
27	dr4m4t1c	73,4%	100,0%	128,1%
28	p455w0rd	82,8%	100,0%	128,1%
29	blink182	78,1%	100,0%	128,1%
30	ab12ab12ab12	78,1%	66,7%	89,6%
31	banana	64,6%	83,3%	100,0%
32	dutawacana	67,5%	100,0%	115,0%
33	HaHaHaHa	75,0%	62,5%	78,1%
34	kaukapakapa	76,1%	72,7%	87,5%
35	Pendadaran	67,5%	90,0%	125,0%

©UKDW

Tabel Contoh Data Keseluruhan No. Pengujian 1-50
(Sambungan)

36	138401000107532	87,5%	86,7%	107,5%
37	085729238778	87,5%	100,0%	114,6%
38	7456E9B2	89,1%	100,0%	128,1%
39	3404060801910003	87,5%	87,5%	107,8%
40	500600	87,5%	83,3%	83,3%
41	Where are you?	66,1%	92,9%	121,4%
42	Please call me!	63,3%	100,0%	120,8%
43	When can we meet?	64,7%	94,1%	119,9%
44	What's the number?	66,7%	100,0%	113,9%
45	Tomorrow, at 9 PM.	71,5%	100,0%	119,4%
46	Where's the meeting?	63,8%	95,0%	118,8%
47	Don't save this SMS!	69,4%	100,0%	118,8%
48	You will be punished!	67,3%	100,0%	113,7%
49	What is the password?	66,1%	100,0%	118,5%
50	I'll talk to you soon.	65,3%	95,5%	109,1%
Rata-rata		**75,3%**	**91,1%**	**115,7%**

Grafik Contoh Data Keseluruhan No. Pengujian 1-50



©UKDW

Tabel dan Grafik Rasio Kompresi (%) Berdasarkan panjang karakter pesan steganografi

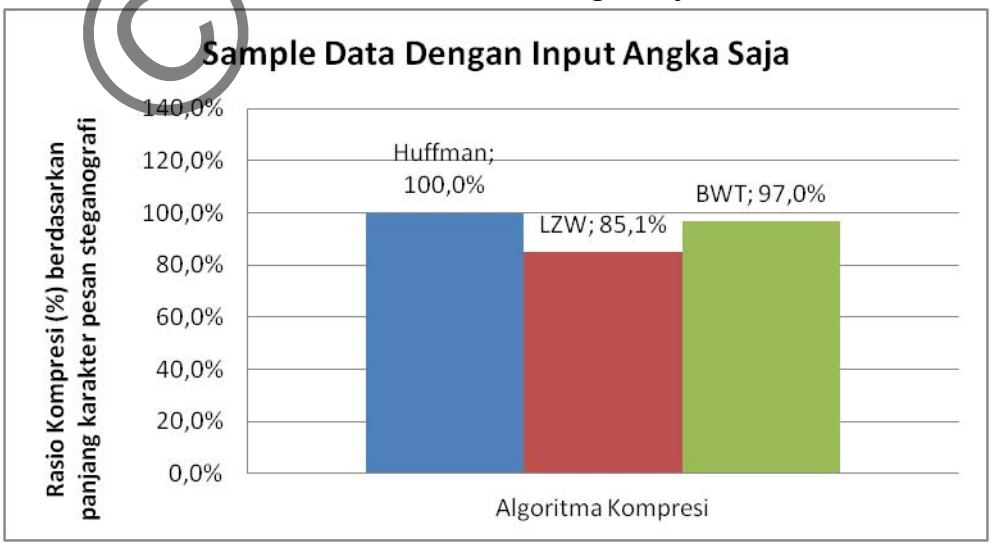
1. Contoh Data Angka Saja

Tabel Contoh Data Angka Saja

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
1	123456	100,0%	100,0%	126,1%
2	123123	100,0%	86,8%	113,2%
3	12345678	100,0%	100,0%	120,4%
4	11111111	100,0%	55,7%	53,4%
5	666666	100,0%	51,4%	59,7%
6	112233	100,0%	100,0%	114,7%
7	1313	100,0%	78,3%	119,6%
8	0007	100,0%	83,7%	95,3%
9	08011991	100,0%	100,0%	106,8%
10	007007007	100,0%	70,7%	66,7%
36	138401000107532	100,0%	88,5%	98,8%
37	085729238778	100,0%	100,0%	101,4%
39	3404060801910003	100,0%	89,0%	100,0%
40	500600	100,0%	87,5%	81,3%
Rata-rata		**100,0%**	**85,1%**	**97,0%**

©UKDW

Grafik Contoh Data Angka Saja



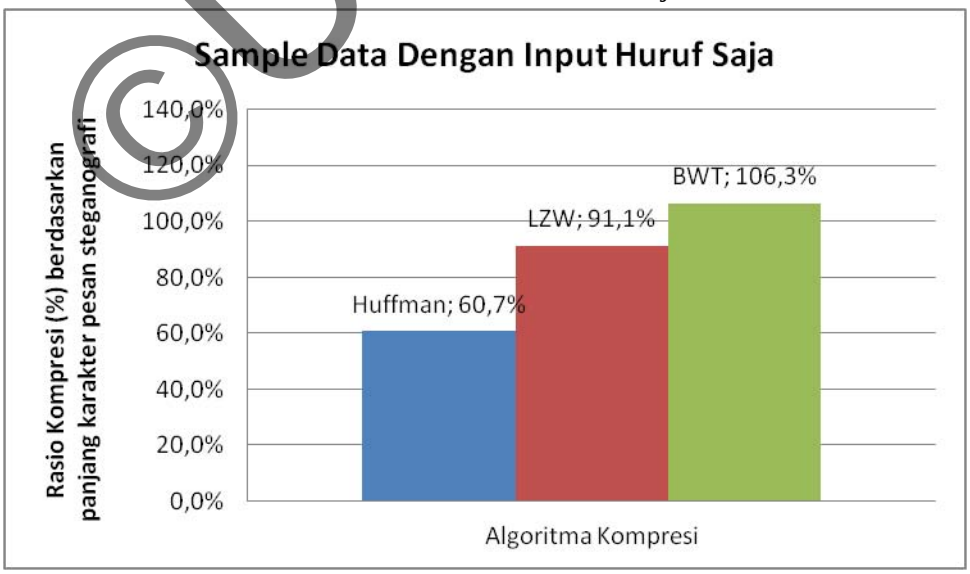
2. Contoh Data Huruf Saja

Tabel Contoh Data Huruf Saja

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
11	password	61,6%	100,0%	116,2%
12	qwerty	69,3%	100,0%	112,0%
13	football	56,1%	100,0%	107,1%
14	blahblah	57,8%	77,8%	111,1%
15	pepper	68,1%	85,5%	114,5%
16	iloveyou	57,3%	100,0%	103,9%
17	sunshine	56,0%	100,0%	116,0%
18	cheese	54,2%	100,0%	116,7%
19	starwars	58,2%	88,8%	112,2%
20	whatever	60,8%	100,0%	122,7%
31	banana	52,9%	82,9%	88,6%
32	dutawacana	59,7%	100,0%	109,2%
33	HaHaHaHa	66,7%	67,9%	75,0%
34	kaukapakapa	69,0%	72,9%	77,5%
35	Pendadaran	63,5%	90,4%	112,2%
Rata-rata		**60,7%**	**91,1%**	**106,3%**

©UKDW

Grafik Contoh Data Huruf Saja

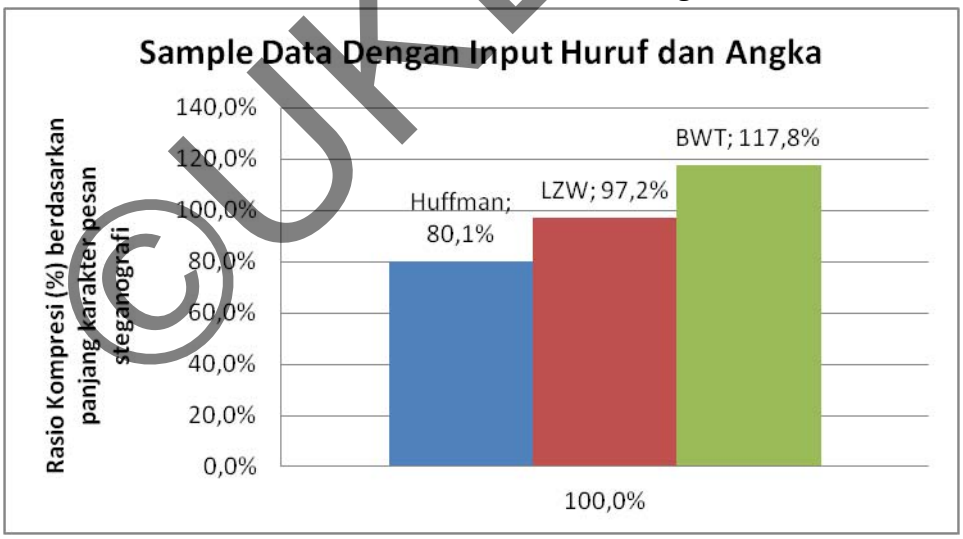


3. Contoh Data Huruf dan Angka

Tabel Contoh Data Huruf dan Angka

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
21	abc123	80,9%	100,0%	129,4%
22	yudhi0801	78,4%	100,0%	123,5%
23	jamesbond007	72,0%	100,0%	107,7%
24	saras008	69,2%	100,0%	119,8%
25	jakarta48	75,0%	100,0%	120,2%
26	08januari1991	80,8%	100,0%	113,9%
27	dr4m4t1c	78,5%	100,0%	123,7%
28	p455w0rd	87,1%	100,0%	129,0%
29	blink182	77,7%	100,0%	125,5%
30	ab12ab12ab12	79,5%	69,7%	79,5%
38	7456E9B2	102,2%	100,0%	123,9%
Rata-rata		**80,1%**	**97,2%**	**117,8%**

Grafik Contoh Data Huruf dan Angka



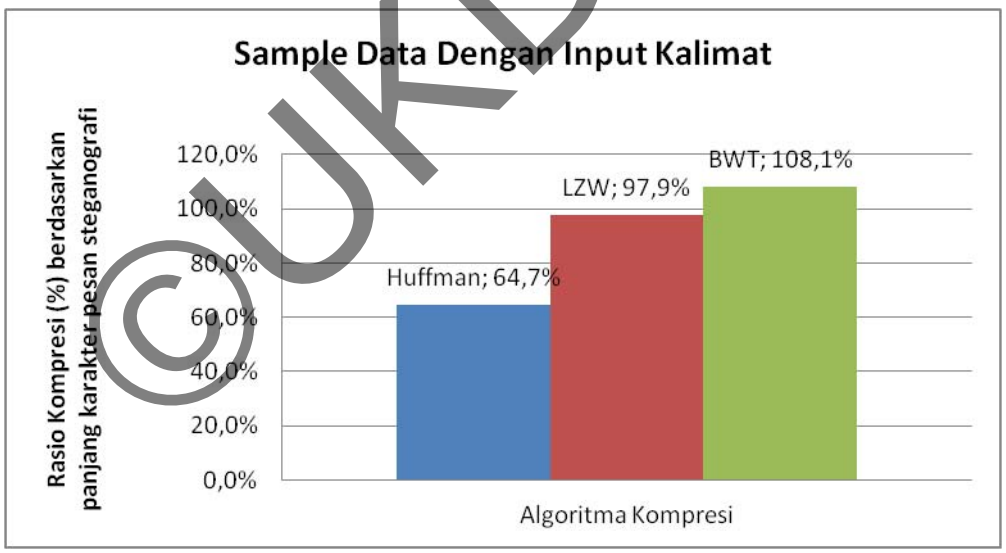
©UKDW

4. Contoh Data Kalimat Normal

Tabel Contoh Data Kalimat Normal

Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
41	Where are you?	62,9%	92,2%	111,4%
42	Please call me!	64,1%	100,0%	111,8%
43	When can we meet?	62,7%	94,5%	107,0%
44	What's the number?	64,8%	100,0%	105,6%
45	Tomorrow, at 9 PM.	70,6%	100,0%	110,0%
46	Where's the meeting?	61,9%	95,4%	106,7%
47	Don't save this SMS!	69,3%	100,0%	113,4%
48	You will be punished!	66,1%	100,0%	109,0%
49	What is the password?	62,8%	100,0%	109,7%
50	I'll talk to you soon.	62,0%	96,6%	97,0%
Rata-rata		**64,7%**	**97,9%**	**108,1%**

Grafik Contoh Data Kalimat Normal



©UKDW

5. Contoh Data Keseluruhan No. Pengujian 1-50

Tabel Contoh Data Keseluruhan No. Pengujian 1-50

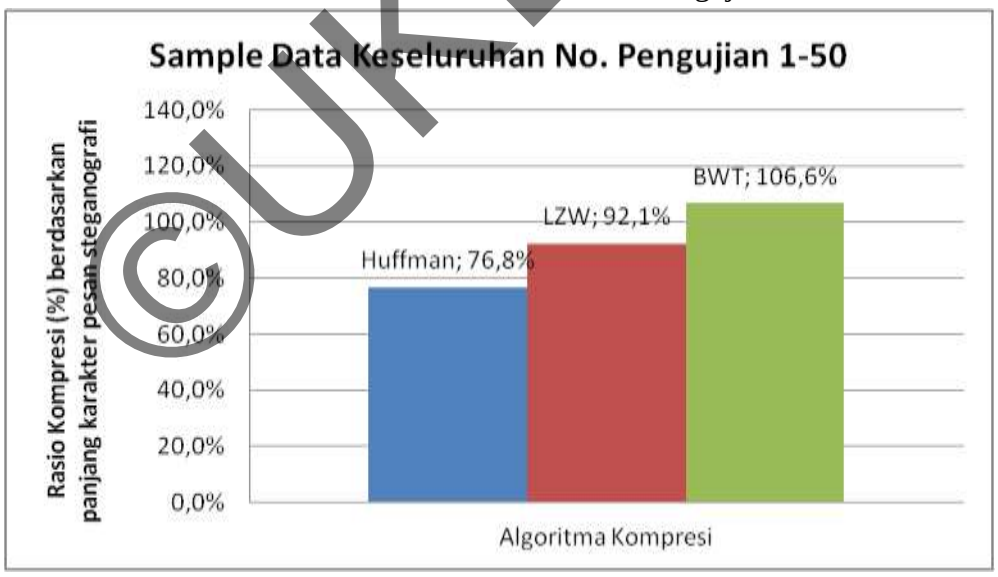
Nomor Pengujian	Pesan Tersembunyi	Huffman	LZW	BWT
1	123456	100,0%	100,0%	126,1%
2	123123	100,0%	86,8%	113,2%
3	12345678	100,0%	100,0%	120,4%
4	11111111	100,0%	55,7%	53,4%
5	666666	100,0%	51,4%	59,7%
6	112233	100,0%	100,0%	114,7%
7	1313	100,0%	78,3%	119,6%
8	0007	100,0%	83,7%	95,3%
9	08011991	100,0%	100,0%	106,8%
10	007007007	100,0%	70,7%	66,7%
11	password	61,6%	100,0%	116,2%
12	qwerty	69,3%	100,0%	112,0%
13	football	56,1%	100,0%	107,1%
14	blahblah	57,8%	77,8%	111,1%
15	pepper	68,1%	85,5%	114,5%
16	iloveyou	57,3%	100,0%	103,9%
17	sunshine	56,0%	100,0%	116,0%
18	cheese	54,2%	100,0%	116,7%
19	starwars	58,2%	88,8%	112,2%
20	whatever	60,8%	100,0%	122,7%
21	abc123	80,9%	100,0%	129,4%
22	yudhi0801	78,4%	100,0%	123,5%
23	jamesbond007	72,0%	100,0%	107,7%
24	saras008	69,2%	100,0%	119,8%
25	jakarta48	75,0%	100,0%	120,2%
26	08januari1991	80,8%	100,0%	113,9%
27	dr4m4t1c	78,5%	100,0%	123,7%
28	p455w0rd	87,1%	100,0%	129,0%
29	blink182	77,7%	100,0%	125,5%
30	ab12ab12ab12	79,5%	69,7%	79,5%
31	banana	52,9%	82,9%	88,6%
32	dutawacana	59,7%	100,0%	109,2%
33	HaHaHaHa	66,7%	67,9%	75,0%
34	kaukapakapa	69,0%	72,9%	77,5%
35	Pendadaran	63,5%	90,4%	112,2%

©UKDW

Tabel Contoh Data Keseluruhan No. Pengujian 1-50
(Sambungan)

36	138401000107532	100,0%	88,5%	98,8%
37	085729238778	100,0%	100,0%	101,4%
38	7456E9B2	102,2%	100,0%	123,9%
39	3404060801910003	100,0%	89,0%	100,0%
40	500600	100,0%	87,5%	81,3%
41	Where are you?	62,9%	92,2%	111,4%
42	Please call me!	64,1%	100,0%	111,8%
43	When can we meet?	62,7%	94,5%	107,0%
44	What's the number?	64,8%	100,0%	105,6%
45	Tomorrow, at 9 PM.	70,6%	100,0%	110,0%
46	Where's the meeting?	61,9%	95,4%	106,7%
47	Don't save this SMS!	69,3%	100,0%	113,4%
48	You will be punished!	66,1%	100,0%	109,0%
49	What is the password?	62,8%	100,0%	109,7%
50	I'll talk to you soon.	62,0%	96,6%	97,0%
Rata-rata		**76,8%**	**92,1%**	**106,6%**

Grafik Contoh Data Keseluruhan No. Pengujian 1-50



©UKDW

Tabel Pengujian BWT dengan MTF dan tanpa MTF

No. Pengujian	Panjang Pesan SMS	Pesan Tersembunyi	Panjang Pesan Tersembunyi	Tanpa Kompresi			Kompresi BWT								
							BWT+MTF+RLE					BWT+RLE			
				Panjang biner pesan tersembunyi (bit)	Panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS	Panjang biner pesan tersembunyi (bit)	Rasio Kompresi (%) berdasarkan panjang biner (bit)	Panjang pesan steganografi	Rasio Kompresi (%) berdasarkan panjang pesan steganografi	Panjang Total SMS
1	60	123456	6	48	69	129	64	133,3%	87	126,1%	147	64	133,3%	88	127,5%	148
2	60	123123	6	48	68	128	64	133,3%	77	113,2%	137	40	83,3%	57	83,8%	117
3	60	12345678	8	64	93	153	82	128,1%	112	120,4%	172	82	128,1%	114	122,6%	174
4	60	11111111	8	64	88	148	34	53,1%	47	53,4%	107	26	40,6%	42	47,7%	102
5	60	666666	6	48	72	132	32	66,7%	43	59,7%	103	24	50,0%	38	52,8%	98
6	60	112233	6	48	68	128	64	133,3%	76	114,7%	138	56	116,7%	77	113,2%	137
7	60	1313	4	32	46	106	46	143,8%	55	119,6%	121	22	68,8%	33	71,7%	93
8	60	0007	4	32	43	103	30	93,8%	41	95,3%	101	22	68,8%	33	76,7%	93
9	60	08011991	8	64	88	148	74	115,6%	94	106,8%	154	74	115,6%	100	113,6%	160
10	60	007007007	9	72	99	159	51	70,8%	66	66,7%	126	35	48,6%	53	53,5%	113
11	60	password	8	64	99	159	82	128,1%	115	116,2%	178	74	115,6%	110	111,1%	170
12	60	qwerty	6	48	75	135	56	116,7%	84	112,0%	144	64	133,3%	95	126,7%	155
13	60	football	8	64	98	158	82	128,1%	105	107,1%	165	66	103,1%	96	98,0%	156
14	60	blahblah	8	64	90	150	82	128,1%	100	111,1%	160	50	78,1%	72	80,0%	132
15	60	pepper	6	48	69	129	64	133,3%	75	114,5%	139	48	100,0%	68	98,6%	128
16	60	iloveyou	8	64	103	163	74	115,6%	107	103,9%	167	74	115,6%	112	108,7%	172
17	60	sunshine	8	64	100	160	82	128,1%	116	116,0%	176	82	128,1%	123	123,0%	183
18	60	cheese	6	48	72	132	64	133,3%	84	116,7%	144	64	133,3%	91	126,4%	151
19	60	starwars	8	64	98	158	82	128,1%	110	112,2%	170	66	103,1%	99	101,0%	159
20	60	whatever	8	64	97	157	82	128,1%	119	122,7%	179	82	128,1%	121	124,7%	181
21	60	abc123	6	48	68	128	64	133,3%	88	129,4%	148	64	133,3%	89	130,9%	149
22	60	yudhi0801	9	72	102	162	91	126,4%	126	123,5%	186	83	115,3%	116	113,7%	176
23	60	jamesbond007	12	96	143	203	110	114,6%	154	107,7%	214	110	114,6%	162	113,3%	222
24	60	saras008	8	64	91	151	82	128,1%	109	119,8%	169	66	103,1%	96	105,5%	156
25	60	jakarta48	9	72	104	164	91	126,4%	125	120,2%	185	83	115,3%	118	113,5%	178
26	60	08januari1991	13	104	154	211	127	122,1%	172	113,9%	232	119	114,4%	167	110,6%	227
27	60	dr4m4t1c	8	64	93	153	82	128,1%	115	123,7%	175	74	115,6%	105	112,9%	165
28	60	p455w0rd	8	64	93	153	82	128,1%	120	129,0%	180	74	115,6%	105	112,9%	165
29	60	blink182	8	64	94	154	82	128,1%	118	125,5%	178	82	128,1%	117	124,5%	177
30	60	ab12ab12ab12	12	96	132	192	86	89,6%	105	79,5%	165	54	56,3%	78	59,1%	138
31	60	banana	6	48	70	130	48	100,0%	62	88,6%	122	40	83,3%	59	84,3%	119
32	60	dutawacana	10	80	119	179	92	115,0%	130	109,2%	190	84	105,0%	124	104,2%	184
33	60	HaHaHaHa	8	64	84	144	50	78,1%	63	75,0%	123	34	53,1%	50	59,5%	110
34	60	kaukapakapa	11	88	129	189	77	87,5%	100	77,5%	160	53	60,2%	82	63,6%	142
35	60	Pendadaran	10	80	115	175	100	125,0%	129	112,2%	189	100	125,0%	138	120,0%	198
36	60	138401000107532	15	120	165	225	129	107,5%	163	98,8%	223	137	114,2%	187	113,3%	247
37	60	085729238778	12	96	140	200	110	114,6%	142	101,4%	202	110	114,6%	153	109,3%	213
38	60	7456E982	8	64	92	152	82	128,1%	114	123,9%	174	82	128,1%	114	123,9%	174
39	60	3404060801910003	16	128	173	233	138	107,8%	173	100,0%	233	115	89,8%	156	90,2%	216
40	60	500600	6	48	64	124	40	83,3%	52	81,3%	112	40	83,3%	57	89,1%	117
41	60	Where are you?	14	112	167	227	136	121,4%	186	111,4%	246	120	107,1%	174	104,2%	234
42	60	Please call me!	15	120	170	230	145	120,8%	190	111,8%	250	137	114,2%	189	111,2%	249
43	60	When can we meet?	17	136	201	261	163	119,9%	215	107,0%	275	163	119,9%	232	115,4%	292
44	60	What's the number?	18	144	213	273	164	113,9%	225	105,6%	285	172	119,4%	245	115,0%	305
45	60	Tomorrow, at 9 PM.	18	144	211	271	172	119,4%	232	110,0%	292	164	113,9%	232	110,0%	292
46	60	Where's the meeting?	20	160	239	299	190	118,8%	255	106,7%	315		0,0%	273	114,2%	333
47	60	Don't save this SMS!	20	160	231	291	190	118,8%	262	113,4%	322	190	118,8%	266	115,2%	326
48	60	You will be punished!	21	168	245	305	191	113,7%	267	109,0%	327		0,0%	280	114,3%	340
49	60	What is the password?	21	168	247	307	199	118,5%	271	109,7%	331		0,0%	282	114,2%	342
50	60	I'll talk to you soon.	22	176	263	323	192	109,1%	255	97,0%	315		0,0%	273	103,8%	333
Rata-rata rasio kompresi data ke 1-45							**115,7%**		**106,5%**			**103,0%**		**101,6%**		

©UKDW

©UKDW

LAMPIRAN C

- Listing Program

MainSMS.java

```
1   package com.ukdw.steganosms;
2
3   import java.util.ArrayList;
4   import android.net.Uri;
5   import android.os.Bundle;
6   import android.app.Activity;
7   import android.app.AlertDialog;
8   import android.app.Dialog;
9   import android.app.PendingIntent;
10  import android.content.BroadcastReceiver;
11  import android.content.ContentValues;
12  import android.content.Context;
13  import android.content.DialogInterface;
14  import android.content.Intent;
15  import android.content.IntentFilter;
16  import android.content.SharedPreferences;
17  import android.telephony.SmsManager;
18  import android.view.KeyEvent;
19  import android.view.Menu;
20  import android.view.MenuItem;
21  import android.view.View;
22  import android.view.Window;
23  import android.widget.Button;
24  import android.widget.EditText;
25  import android.widget.Toast;
26
27  public class MainSMS extends Activity {
28     EditText textPhone, textSMS;
29          Button btnTambahPesanRahasia, btnKirimSMS;
30
31      @Override
32      public void onCreate(Bundle savedInstanceState) {
33          super.onCreate(savedInstanceState);
34          setContentView(R.layout.activity_main_sms);
35
36          btnTambahPesanRahasia = (Button) findViewById(R.id.btnTambahPesanRahasia);
37          btnKirimSMS = (Button) findViewById(R.id.btnKirimSMS);
38          textPhone = (EditText) findViewById(R.id.textPhone);
39          textSMS = (EditText) findViewById(R.id.textSMS);
40
41          btnKirimSMS.setOnClickListener(new View.OnClickListener() {
42
43                      @Override
44                      public void onClick(View v) {
45                             String ponsel = textPhone.getText().toString();
46                             String isisms = textSMS.getText().toString();
47
48                             if (ponsel.length()>0 && isisms.length()>0){
49                                    kirimSMS(ponsel, isisms);
50                                    saveSendItem(ponsel, isisms);
51                             }
52
53                             else
54                 Toast.makeText(getBaseContext(),"Masukkan No ponsel tujuan dan isi
55                 sms!", Toast.LENGTH_SHORT).show();
```

©UKDW

```
56                      }
57                });
58
59          btnTambahPesanRahasia.setOnClickListener(new View.OnClickListener() {
60
61                      @Override
62                      public void onClick(View v) {
63                             String ponsel = textPhone.getText().toString();
64                             String isisms = textSMS.getText().toString();
65
66                             if (ponsel.length()>0 && isisms.length()>0){
67                                    tambahPesan(ponsel, isisms);
68                             }
69                             else
70                Toast.makeText(getBaseContext(),"Masukkan terlebih dulu No ponsel
71                tujuan dan isi sms!", Toast.LENGTH_SHORT).show();
72                             }
73                });
74       }
75
76          @Override
77          protected void onResume() {
78                 super.onResume();
79         Intent batal = getIntent();
80         textPhone.setText(batal.getStringExtra("no_HP"));
81         textSMS.setText(batal.getStringExtra("isi_SMS"));
82          }
83
84      @Override
85      public boolean onCreateOptionsMenu(Menu menu) {
86           // Inflate the menu; this adds items to the action bar if it is present.
87           getMenuInflater().inflate(R.menu.main_sm, menu);
88           return true;
89      }
90
91      public boolean onOptionsItemSelected(MenuItem item) {
92           switch (item.getItemId()) {
93
94            case R.id.ubahpass:
95                 // TODO Auto-generated method stub
96                 final Dialog ubah = new Dialog(MainSMS.this);
97                 ubah.requestWindowFeature(Window.FEATURE_NO_TITLE);
98                 ubah.setContentView(R.layout.ubah_pass);
99
100                Button ubahpass = (Button)ubah.findViewById(R.id.ubahPassword);
101                ubahpass.setOnClickListener(new View.OnClickListener() {
102                       @Override
103                       public void onClick(View v) {
104                       // TODO Auto-generated method stub
105             EditText pass_lama = (EditText)ubah.findViewById(R.id.passwordLama);
106                       EditText pass_baru =
107   (EditText)ubah.findViewById(R.id.passwordBaru);
108                       String kunci_lama = pass_lama.getText().toString();
109                       String kunci_baru = pass_baru.getText().toString();
110             SharedPreferences passPreference = getSharedPreferences("Preference
111             Password",Context.MODE_PRIVATE);
```

©UKDW

```
112                         SharedPreferences.Editor passPrefEdit =
113    passPreference.edit();
114                         String kunci_tersimpan=
115    passPreference.getString("pass_rahasia","ukdw");
116
117                                 if(kunci_tersimpan.equals(kunci_lama)){
118                                         passPrefEdit.putString("pass_rahasia",
119    kunci_baru);
120                                         passPrefEdit.commit();
121                                         Toast.makeText(getBaseContext(),"Password Baru
122    Tersimpan",
123                                                 Toast.LENGTH_SHORT).show();
124                                         ubah.dismiss();
125                                 }else {
126                                         Toast.makeText(getBaseContext(),"Password Lama
127    Salah!",
128                                                 Toast.LENGTH_SHORT).show();
129                                         pass_lama.setText("");
130                                         pass_baru.setText("");
131                                 }
132                         }
133                 });
134                 ubah.show();
135          return true;
136
137          case R.id.about:
138                 AlertDialog.Builder alertMax = new AlertDialog.Builder(MainSMS.this);
139          alertMax.setMessage("SMS Rahasia adalah aplikasi sms yang menerapkan
140                   kompresi" + " dan steganografi pada pesan yang ingin dirahasiakan.
141                   Aplikasi ini dibuat" + " oleh Yudhi Putra Tama (22084583) untuk
142                   memenuhi Tugas Akhir program studi" + " Teknik Informatika
143                   UKDW.");
144                         alertMax.setTitle("Tentang Aplikasi");
145
146          alertMax.setPositiveButton("Kembali",new DialogInterface.OnClickListener(){
147                                 @Override
148                                 public void onClick(DialogInterface dialog, int id) {
149                                         dialog.cancel();
150                                 }
151                         });
152
153                 AlertDialog dialog = alertMax.create();
154                 dialog.show();
155          return true;
156
157                         default:
158              return super.onOptionsItemSelected(item);
159           }
160       }
161
162       public void buatSMS (View view){
163          Intent buat = new Intent(this, MainSMS.class);
164          startActivity(buat);
165       }
166
167       public void masukSMS (View view){
```

©UKDW

```
168          Intent masuk = new Intent(this, MasukSMS.class);
169          startActivity(masuk);
170       }
171
172       public void terkirimSMS (View view){
173          Intent terkirim = new Intent(this, TerkirimSMS.class);
174          startActivity(terkirim);
175       }
176
177       public void tambahPesan (String ponsel, String isisms){
178          Intent tambah = new Intent(this, PesanTersembunyi.class);
179          tambah.putExtra("no_HP", ponsel);
180          tambah.putExtra("isi_SMS", isisms);
181          startActivity(tambah);
182       }
183
184       public void kirimSMS (String ponsel, String isisms){
185          String SENT = "SMS_SENT";
186          String DELIVERED = "SMS_DELIVERED";
187
188          Intent mSentIntent = new Intent(SENT);
189          Intent mDeliveryIntent = new Intent(DELIVERED);
190
191                SmsManager smsMulti = SmsManager.getDefault();
192                ArrayList<String> bagian = smsMulti.divideMessage(isisms);
193                int numParts = bagian.size();
194                if (numParts <= 5){
195                ArrayList<PendingIntent> sentIntents = new
196    ArrayList<PendingIntent>();
197                ArrayList<PendingIntent> deliveryIntents = new
198    ArrayList<PendingIntent>();
199
200          for (int i=0; i < numParts; i++){
201                   sentIntents.add(PendingIntent.getBroadcast(this, 0, mSentIntent,
202    0));
203                   deliveryIntents.add(PendingIntent.getBroadcast(this, 0,
204    mDeliveryIntent, 0));
205                }
206          smsMulti.sendMultipartTextMessage(ponsel, null, bagian, sentIntents,
207          deliveryIntents);
208                }
209                else{
210                      Toast.makeText(getBaseContext(),"SMS lebih dari 5 bagian",
211                      Toast.LENGTH_SHORT).show();
212                }
213
214          registerReceiver(new BroadcastReceiver(){
215          @Override
216                public void onReceive(Context arg0, Intent arg1) {
217                switch (getResultCode())
218                {
219                case Activity.RESULT_OK:
220                      Intent kembali = new Intent(MainSMS.this, MainSMS.class);
221                            kembali.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
222                            startActivity(kembali);
```

©UKDW

```
223                 Toast.makeText(getBaseContext(), "SMS dikirim",
224                 Toast.LENGTH_SHORT).show();
225                 break;
226                 case SmsManager.RESULT_ERROR_GENERIC_FAILURE:
227                        Toast.makeText(getBaseContext(), "Generic failure",
228                 Toast.LENGTH_SHORT).show();
229                        break;
230                 case SmsManager.RESULT_ERROR_NO_SERVICE:
231                 Toast.makeText(getBaseContext(), "No service",
232                 Toast.LENGTH_SHORT).show();
233                        break;
234                 case SmsManager.RESULT_ERROR_NULL_PDU:
235                 Toast.makeText(getBaseContext(), "Null PDU",
236                 Toast.LENGTH_SHORT).show();
237                 break;
238                                case SmsManager.RESULT_ERROR_RADIO_OFF:
239                                Toast.makeText(getBaseContext(), "Radio off",
240                 Toast.LENGTH_SHORT).show();
241                                        break;
242                 }
243                 }
244           }, new IntentFilter(SENT));
245
246           registerReceiver(new BroadcastReceiver(){
247                   @Override
248                   public void onReceive(Context arg0,Intent arg1) {
249                          switch (getResultCode())
250                   {
251                          case Activity.RESULT_OK:
252                          Intent keterkirim = new Intent(MainSMS.this,
253    TerkirimSMS.class);
254                                 keterkirim.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
255                                 startActivity(keterkirim);
256                   Toast.makeText(getBaseContext(), "SMS terkirim",
257                   Toast.LENGTH_SHORT).show();
258                   break;
259                   case Activity.RESULT_CANCELED:
260                   Toast.makeText(getBaseContext(), "SMS gagal",
261                   Toast.LENGTH_SHORT).show();
262                          break;
263               }
264            }
265           }, new IntentFilter(DELIVERED));
266       }
267
268        public void saveSendItem(String ponsel, String isisms) {
269           ContentValues values = new ContentValues();
270           values.put("address", ponsel);
271           values.put("body", isisms);
272           getContentResolver().insert(Uri.parse("content://sms/sent"), values);
273        }
274
275        public boolean onKeyDown(int keyCode, KeyEvent event){
276                   if (keyCode == KeyEvent.KEYCODE_BACK && event.getRepeatCount() == 0){
277                   AlertDialog.Builder alertMax = new AlertDialog.Builder(this);
278                   alertMax.setMessage("Anda yakin ingin keluar dari aplikasi?");
```

©UKDW

```
279                 alertMax.setTitle("Tutup Aplikasi");
280                 alertMax.setPositiveButton("Ya", new
281     DialogInterface.OnClickListener() {
282                 @Override
283                 public void onClick(DialogInterface dialog, int id) {
284                         Intent exit = new Intent(Intent.ACTION_MAIN);
285                         exit.addCategory(Intent.CATEGORY_HOME);
286                         exit.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
287                         startActivity(exit);
288                 }
289           });
290
291                 alertMax.setNegativeButton("Tidak", new
292     DialogInterface.OnClickListener() {
293                         @Override
294                         public void onClick(DialogInterface dialog, int id) {
295                                 dialog.cancel();
296                         }
297                 });
298
299                 AlertDialog dialog = alertMax.create();
300                 dialog.show();
301                 return true;
302         }
303                 return super.onKeyDown(keyCode, event);
304     }
305 }
```

©UKDW

PesanTersembunyi.java

```
1   package com.ukdw.steganosms;
2
3   import java.util.ArrayList;
4
5   import android.os.Bundle;
6   import android.app.Activity;
7   import android.app.AlertDialog;
8   import android.content.DialogInterface;
9   import android.content.Intent;
10  import android.telephony.SmsManager;
11  import android.view.Menu;
12  import android.view.View;
13  import android.widget.Button;
14  import android.widget.EditText;
15  import android.widget.RadioButton;
16  import android.widget.RadioGroup;
17  import android.widget.Toast;
18
19  public class PesanTersembunyi extends Activity {
20
21          Button btnSembunyikanPesan;
22          EditText textRahasia;
23          String ponsel, isisms, pesanKompresi, pesanRahasia, algoritma,
24  binerKompresi;
25          RadioGroup radiokompresi;
26          RadioButton algoritmaKompresi;
27
28
29          @Override
30          protected void onCreate(Bundle savedInstanceState) {
31                  super.onCreate(savedInstanceState);
32                  setContentView(R.layout.activity_pesan_tersembunyi);
33
34                  radiokompresi = (RadioGroup) findViewById(R.id.radio_kompresi);
35                  btnSembunyikanPesan = (Button)
36  findViewById(R.id.btnSembunyikanPesan);
37                  textRahasia = (EditText) findViewById(R.id.textRahasia);
38
39                  btnSembunyikanPesan.setOnClickListener(new View.OnClickListener() {
40
41
42                          @Override
43                          public void onClick(View v) {
44                                  int pilihanKompresi =
45  radiokompresi.getCheckedRadioButtonId();
46                                  algoritmaKompresi = (RadioButton)
47  findViewById(pilihanKompresi);
48                                  pesanRahasia = textRahasia.getText().toString();
49                                  algoritma = algoritmaKompresi.getText().toString();
50                                  if (pesanRahasia.length()>0){
51                                          Kompresi kompresi =
52  KompresiFactory.algoKompresi(algoritma);
53                                          pesanKompresi =
54  kompresi.compress(pesanRahasia);
```

©UKDW

```
55                                      binerKompresi =
56  kompresi.pesanBiner(pesanKompresi);
57                                      sembunyikanPesan(binerKompresi);
58                                  }
59                                  else
60                                      Toast.makeText(getBaseContext(),"Pesan belum
61  ada!",
62                                              Toast.LENGTH_SHORT).show();
63                          }
64                  });
65          }
66
67          @Override
68          protected void onResume() {
69                  super.onResume();
70                  Intent tambah = getIntent();
71                  ponsel = tambah.getStringExtra("no_HP");
72                  isisms = tambah.getStringExtra("isi_SMS");
73          }
74
75          @Override
76          public boolean onCreateOptionsMenu(Menu menu) {
77                  // Inflate the menu; this adds items to the action bar if it is
78  present.
79                  getMenuInflater().inflate(R.menu.pesan_tersembunyi, menu);
80                  return true;
81          }
82
83          public void sembunyikanPesan (String binerKompresi){
84
85                  Steganografi stegano = new Steganografi();
86                  String pesanTersembunyi = stegano.encodeSteganografi(binerKompresi);
87                  String isismsStego = isisms + pesanTersembunyi;
88
89                  int panjangSMS = isismsStego.length();
90                  if (algoritma.equals("Tanpa Kompresi")){
91                          panjangSMS = panjangSMS-2;
92                  }else if (algoritma.equals("Huffman")){
93                          panjangSMS = panjangSMS-3;
94                  }else if (algoritma.equals("LZW")){
95                          panjangSMS = panjangSMS-3;
96                  }else if (algoritma.equals("BWT")){
97                          panjangSMS = panjangSMS-4;
98                  }
99
100
101                 SmsManager cekSMS = SmsManager.getDefault();
102                 ArrayList<String> bagian = cekSMS.divideMessage(isismsStego);
103                 int numParts = bagian.size();
104
105                 if (numParts <= 5 ){
106                         Intent StegoSMS = new Intent(this, StegoSMS.class);
107                         StegoSMS.putExtra("no_HP", ponsel);
108                         StegoSMS.putExtra("isi_SMS", isismsStego);
109                 startActivity(StegoSMS);
110                 }
```

©UKDW

```
111                 else {
112                         AlertDialog.Builder alertMax = new AlertDialog.Builder(this);
113           alertMax.setMessage("Panjang total pesan terlalu panjang. Panjang total
114                   pesan: "+panjangSMS+". Pesan tidak dapat dikirim lewat SMS!");
115                         alertMax.setTitle("Steganografi Gagal");
116           alertMax.setPositiveButton("Ubah Pesan Rahasia", new
117              DialogInterface.OnClickListener() {
118
119                                 @Override
120                                 public void onClick(DialogInterface dialog, int id) {
121
122                     Intent kePesanRahasia = new Intent(PesanTersembunyi.this,
123                        PesanTersembunyi.class);
124                                         kePesanRahasia.putExtra("no_HP", ponsel);
125                                         kePesanRahasia.putExtra("isi_SMS", isisms);
126
127          kePesanRahasia.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
128                                         startActivity(kePesanRahasia);
129
130                                 }
131                         });
132
133           alertMax.setNegativeButton("Ubah SMS", new
134              DialogInterface.OnClickListener(){
135
136                                 @Override
137                                 public void onClick(DialogInterface dialog, int id) {
138
139                                         Intent batal = new
140   Intent(PesanTersembunyi.this, MainSMS.class);
141                                         batal.putExtra("no_HP", ponsel);
142                                         batal.putExtra("isi_SMS", isisms);
143                                         batal.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
144                                         startActivity(batal);
145
146                                 }
147                         });
148
149                     AlertDialog dialog = alertMax.create();
150                     dialog.show();
151                     }
152             }
153   }
```

©UKDW

Kompresi.java

```
1   package com.ukdw.steganosms;
2
3   interface Kompresi {
4           public String compress(String pesan);
5
6           public String pesanBiner(String pesanKompresi);
7
8           public String decompress(String binCode);
9   }
```

©UKDW

KompresiFactory.java

```
1   package com.ukdw.steganosms;
2
3   public class KompresiFactory {
4           public static Kompresi algoKompresi (String algoritma){
5                   if (algoritma.equals("Tanpa Kompresi")){
6                           return new TanpaKompresi();
7                   }else if (algoritma.equals("Huffman")){
8                           return new Huffman();
9                   }else if (algoritma.equals("LZW")){
10                          return new LZW();
11                  }else if (algoritma.equals("BWT")){
12                          return new BWT();
13                  }
14                  return null;
15          }
16  }
```

©UKDW

TanpaKompresi.java

```
1   package com.ukdw.steganosms;
2
3   public class TanpaKompresi implements Kompresi{
4           @Override
5           public String compress(String pesan) {
6                   String pesanAsli;
7                   // TODO Auto-generated method stub
8                   pesanAsli = pesan;
9                   return pesanAsli;
10          }
11
12  /*      ======================== fungsi string to biner
13  ==========================*/
14
15          @Override
16          public String pesanBiner(String pesanKompresi) {
17                  // TODO Auto-generated method stub
18                  String psnBiner;
19                  byte[] pesanBin = pesanKompresi.getBytes();
20                  StringBuilder binary = new StringBuilder();
21
22                  for (byte b : pesanBin){
23                          int val = b;
24                          for (int i=0; i<8; i++){
25                                  binary.append((val & 128) == 0 ? 0 : 1);
26                                  val <<=1;
27                          }
28                  }
29                  psnBiner = binary.toString()+"00";
30                  return psnBiner;
31          }
32
33  /*      ========================== fungsi2 decompresi ========================*/
34
35
36          @Override
37          public String decompress(String binCode) {
38                  String isiPesanRahasia="", byteKompresi;
39                  // TODO Auto-generated method stub
40                  byteKompresi = binCode.substring(0, binCode.length()-2);
41
42
43                  for (int i=0; i<byteKompresi.length(); i+=8){
44                          String binToSring = byteKompresi.substring(i, i+8);
45                          byte isi = Byte.parseByte(binToSring, 2);
46                          String binDecimal = Byte.toString(isi);
47                          int karakter = Integer.parseInt(binDecimal);
48                          isiPesanRahasia = isiPesanRahasia + (char)karakter;
49                  }
50
51                  return isiPesanRahasia;
52          }
53
54  }
```

©UKDW

Huffman.java

```
1   package com.ukdw.steganosms;
2
3   public class Huffman implements Kompresi{
4
5           static String[] tabHuff = new String[128];
6           static String diluarTabel = "1111110";
7
8           @Override
9           public String compress(String pesanRahasia) {
10                  // TODO Auto-generated method stub
11                  String pesanKompresi= "";
12                  fixedTable();
13              for (int i = 0; i < pesanRahasia.length(); i++)
14              {
15                char karakterPesan = pesanRahasia.charAt(i);
16                if (tabHuff[karakterPesan] != null) {
17                    pesanKompresi = pesanKompresi + tabHuff[karakterPesan];
18                } else {
19                    pesanKompresi = pesanKompresi + compressKarLain(karakterPesan);
20                }
21              }
22                  return pesanKompresi;
23          }
24
25          static String compressKarLain(char karakterPesan)
26            {
27              String karLain = Integer.toBinaryString(karakterPesan);
28              karLain = "0000000000000000".substring(0, 16 - karLain.length()) +
29  karLain;
30              return diluarTabel + karLain;
31            }
32
33  /*      ====================== fungsi string to biner ======================*/
34
35          @Override
36          public String pesanBiner(String pesanKompresi) {
37                  // TODO Auto-generated method stub
38                  String psnBiner = pesanKompresi+"01";
39                  return psnBiner;
40          }
41
42  /*      ======================= fungsi2 decompresi =======================*/
43
44          @Override
45          public String decompress(String binCode) {
46                  // TODO Auto-generated method stub
47                  String isiPesanRahasia="", byteKompresi, tampung = "";
48                  byteKompresi = binCode.substring(0, binCode.length()-2);
49
50                  fixedTable();
51
52              for (int i = 0; i < byteKompresi.length(); i++)
53              {
54                char karBiner = byteKompresi.charAt(i);
55                tampung = tampung + karBiner;
```

©UKDW

```
56                int c = cariKarakter(tampung);
57                if (c >= 0)
58                {
59                    tampung = "";
60                  isiPesanRahasia = isiPesanRahasia + (char)c;
61                }
62                else if (c == -2)
63                {
64                    tampung = "";
65                }
66              }
67                 return isiPesanRahasia;
68          }
69
70          static int cariKarakter(String tampung)
71            {
72              if (tampung.length() > 23) {
73                return -2;
74              }
75              if ((tampung.length() == 23) &&
76                (tampung.substring(0, 7).equals(diLuarTabel))) {
77                return Integer.parseInt(tampung.substring(7), 2);
78              }
79              for (int i = 0; i < 128; i++) {
80                if ((tabHuff[i] != null) &&
81                  (tabHuff[i].equals(tampung))) {
82                  return i;
83                }
84              }
85              return -1;
86            }
87
88    /*      ================== fungsi inisialisai tabel fixed ==================*/
89
90          static void fixedTable()
91            {
92                  tabHuff[32] = "1011";
93                  tabHuff[33] = "1100011";
94                  tabHuff[34] = "1100110";
95                  tabHuff[39] = "1100100";
96                  tabHuff[40] = "11111010";
97                  tabHuff[41] = "11111011";
98                  tabHuff[44] = "1100001";
99                  tabHuff[45] = "1100101";
100                 tabHuff[46] = "1100000";
101                 tabHuff[47] = "1111011";
102                 tabHuff[48] = "1110000";
103                 tabHuff[49] = "1110001";
104                 tabHuff[50] = "1110010";
105                 tabHuff[51] = "1110011";
106                 tabHuff[52] = "1110100";
107                 tabHuff[53] = "1110101";
108                 tabHuff[54] = "1110110";
109                 tabHuff[55] = "1110111";
110                 tabHuff[56] = "1111000";
111                 tabHuff[57] = "1111001";
```

©UKDW

```
112                 tabHuff[58] = "1111100";
113                 tabHuff[63] = "1100010";
114                 tabHuff[64] = "1111010";
115                 tabHuff[65] = "1000000";
116                 tabHuff[66] = "10000101";
117                 tabHuff[67] = "1001101";
118                 tabHuff[68] = "10001011";
119                 tabHuff[69] = "1100111";
120                 tabHuff[70] = "10011001";
121                 tabHuff[71] = "10010101";
122                 tabHuff[72] = "1001000";
123                 tabHuff[73] = "1000001";
124                 tabHuff[74] = "100110001001";
125                 tabHuff[75] = "1001100011";
126                 tabHuff[76] = "1001001";
127                 tabHuff[77] = "10010100";
128                 tabHuff[78] = "1000110";
129                 tabHuff[79] = "1000100";
130                 tabHuff[80] = "10010110";
131                 tabHuff[81] = "1001100010000";
132                 tabHuff[82] = "1000111";
133                 tabHuff[83] = "1000011";
134                 tabHuff[84] = "100111";
135                 tabHuff[85] = "10010111";
136                 tabHuff[86] = "1000101001";
137                 tabHuff[87] = "1000101000";
138                 tabHuff[88] = "10011000101";
139                 tabHuff[89] = "10000100";
140                 tabHuff[90] = "1001100010001";
141                 tabHuff[97] = "00000";
142                 tabHuff[98] = "000101";
143                 tabHuff[99] = "01101";
144                 tabHuff[100] = "001011";
145                 tabHuff[101] = "1010";
146                 tabHuff[102] = "011001";
147                 tabHuff[103] = "010101";
148                 tabHuff[104] = "01000";
149                 tabHuff[105] = "00001";
150                 tabHuff[106] = "0110001001";
151                 tabHuff[107] = "01100011";
152                 tabHuff[108] = "01001";
153                 tabHuff[109] = "010100";
154                 tabHuff[110] = "00110";
155                 tabHuff[111] = "00100";
156                 tabHuff[112] = "010110";
157                 tabHuff[113] = "011000010000";
158                 tabHuff[114] = "00111";
159                 tabHuff[115] = "00011";
160                 tabHuff[116] = "0111";
161                 tabHuff[117] = "010111";
162                 tabHuff[118] = "00101001";
163                 tabHuff[119] = "00101000";
164                 tabHuff[120] = "011000101";
165                 tabHuff[121] = "000100";
166                 tabHuff[122] = "011000010001";
167             }
```

©UKDW

```
168     }
```

©UKDW

LZW.java

```
1   package com.ukdw.steganosms;
2
3   public class LZW implements Kompresi{
4
5           static String[] dictLZW = new String[128];
6           int minDict = 31, ketemu = -1;
7           boolean ada= false;
8
9           @Override
10          public String compress(String pesanRahasia) {
11                  // TODO Auto-generated method stub
12                  String pesanKompresi, tampung="", in, out;
13
14                  dictionary();
15
16                  out= pesanRahasia.substring(0, 1);
17                  pesanKompresi = out;
18
19                  for(int i=1; i<pesanRahasia.length();i++){
20                          in = pesanRahasia.substring(i, i+1);
21                          tampung = out+in;
22                          cekDict(tampung);
23                          if (ada==true){
24                                  pesanKompresi= pesanKompresi.substring(0, pesanKompresi.length()-
25  1)+(char)ketemu;
26                                  out = dictLZW[ketemu];
27                          }else{
28                                  dictLZW[minDict]=tampung;
29                                  if(minDict==0){
30                                          dictionary();
31                                          minDict = 32;
32                                  }
33                                  minDict--;
34                                  pesanKompresi = pesanKompresi+in;
35                                  out=in;
36                          }
37                  }
38                  return pesanKompresi;
39          }
40
41  /*      ========================== fungsi string to biner
42  ==========================*/
43
44          @Override
45          public String pesanBiner(String pesanKompresi) {
46                  // TODO Auto-generated method stub
47                  String psnBiner;
48                  byte[] pesanBin = pesanKompresi.getBytes();
49                  StringBuilder binary = new StringBuilder();
50
51                  for (byte b : pesanBin){
52                          int val = b;
53                          for (int i=0; i<8; i++){
54                                  binary.append((val & 128) == 0 ? 0 : 1);
55                                  val <<=1;
```

©UKDW

```
56                          }
57                  }
58                  psnBiner = binary.toString()+"10";
59                  //psnBiner = pesanKompresi;
60                  return psnBiner;
61          }
62
63  /*      ============================ fungsi2 decompresi
64  ============================*/
65
66          @Override
67          public String decompress(String binCode) {
68                  // TODO Auto-generated method stub
69                  String tampung="";
70                  String in, out;
71                  String isiPesanRahasia="", byteKompresi, pesanLZW="";
72                  byteKompresi = binCode.substring(0, binCode.length()-2);
73
74                  for (int i=0; i<byteKompresi.length(); i+=8){
75                          String binToSring = byteKompresi.substring(i, i+8);
76                          byte isi = Byte.parseByte(binToSring, 2);
77                          String binDecimal = Byte.toString(isi);
78                          int karakter = Integer.parseInt(binDecimal);
79                          pesanLZW = pesanLZW + (char)karakter;
80                  }
81
82                  dictionary();
83
84                  out = pesanLZW.substring(0, 1);
85                  isiPesanRahasia = out;
86
87                  for(int i=1;i<pesanLZW.length(); i++){
88                          if (dictLZW[pesanLZW.charAt(i)] != null){
89                                  in = dictLZW[pesanLZW.charAt(i)];
90                                  isiPesanRahasia = isiPesanRahasia+in;
91                                  String in2= in.substring(0, 1);
92                                  tampung = out+in2;
93                                  dictLZW[minDict]=tampung;
94                                  out = in;
95                                  if(minDict==0){
96                                          dictionary();
97                                          minDict = 32;
98                                  }
99                                  minDict--;
100                         }else{
101                                 String out2 = out.substring(0, 1);
102                                 tampung = out+out2;
103                                 dictLZW[minDict]=tampung;
104                                 out = tampung;
105                                 isiPesanRahasia = isiPesanRahasia+tampung;
106                                 if(minDict==0){
107                                         dictionary();
108                                         minDict = 32;
109                                 }
110                                 minDict--;
111                         }
```

©UKDW

```
112                 }
113                 return isiPesanRahasia;
114         }
115
116 /*      ======================== fungsi inisialisai tabel fixed
117 ========================*/
118
119         static void dictionary(){
120                 for (int i=0; i<128; i++ ){
121                         if (i<32){
122                                 dictLZW[i]=null;
123                         }else{
124                                 String a="";
125                                 dictLZW[i]= a + (char)i;
126                         }
127                 }
128         }
129
130         public void cekDict(String tampung){
131                 ada = false;
132                 for(int i=minDict+1;i<128;i++){
133                         if (dictLZW[i].equals(tampung)){
134                                 ketemu=i;
135                                 ada = true;
136                                 break;
137                         }
138                 }
139         }
140
141 }
```

©UKDW

BWT.java

```
1   package com.ukdw.steganosms;
2
3   import java.util.ArrayList;
4   import java.util.Arrays;
5
6   public class BWT implements Kompresi{
7           String binRLE="0";
8           char indexPesanAsli;
9           @Override
10          public String compress(String pesanRahasia) {
11                  String pesanKompresi, pesanBWT,pesanRLE, pesanMTF;
12                  pesanBWT = forwardBWT(pesanRahasia);
13                  pesanMTF = forwardMTF(pesanBWT);
14                  pesanRLE = compRLE(pesanMTF);
15
16                  byte[] pesanBin = pesanRLE.getBytes();
17                  StringBuilder binary = new StringBuilder();
18
19                  for (byte b : pesanBin){
20                          int val = b;
21                          for (int i=0; i<8; i++){
22                                  binary.append((val & 128) == 0 ? 0 : 1);
23                                  val <<=1;
24                          }
25                  }
26                  pesanKompresi = binary.toString()  + "1" + binRLE;
27                  return pesanKompresi;
28          }
29
30          public String forwardBWT(String pesanRahasia){
31                  String pesanAsli, pesanBWT="";
32                  String[] arrayKarakter = new String[pesanRahasia.length()];
33                  pesanAsli = pesanRahasia;
34                  int idxPesanAsli = 0;
35
36                  // TODO Auto-generated method stub
37                  arrayKarakter[0] = pesanRahasia;
38                  for (int i=1; i<pesanRahasia.length(); i++){
39                          pesanRahasia = pesanRahasia.substring(1) +
40  pesanRahasia.charAt(0);
41                          arrayKarakter[i]= pesanRahasia;
42                  }
43
44                  Arrays.sort(arrayKarakter);
45
46                  for(int i=0; i<pesanRahasia.length(); i++){
47                          if(arrayKarakter[i].equals(pesanAsli)) {
48                                  idxPesanAsli=i;
49                          }
50                          pesanBWT=
51  pesanBWT+arrayKarakter[i].charAt(arrayKarakter[i].length()-1);
52                  }
53                  indexPesanAsli = (char)idxPesanAsli;
54                  return pesanBWT;
55          }
```

©UKDW

```
56
57          public String forwardMTF(String pesanBWT){
58                  String pesanMTF="";
59                  int indexKarakter=0;
60                  int kar;
61
62                  ArrayList<Character> arrayKarakterMTF = new ArrayList<Character>();
63                  int[] dict = new int[128];
64                  for(int i = 0; i < 128; i++)
65                          dict[i] = i;
66
67                  for(int i=0; i < pesanBWT.length(); i++)
68                          arrayKarakterMTF.add(pesanBWT.charAt(i));
69
70                  for(char x : arrayKarakterMTF)
71                  {
72                          kar = (int) x;
73                          for(int i = 0; i < 128; i++){
74                                  if(dict[i] == kar)
75                                          indexKarakter = i;
76                          }
77                          for(int idx = indexKarakter; idx != 0; idx--){
78                                  dict[idx] = dict[idx-1];
79                          }
80                          dict[0] = kar;
81                          pesanMTF = pesanMTF + (char)indexKarakter;
82                  }
83                  pesanMTF = pesanMTF + indexPesanAsli;
84                  return pesanMTF;
85          }
86
87          public String compRLE(String pesanMTF){
88                  String pesanRLE= pesanMTF.substring(0, 1);
89                  for(int i=1;i<pesanMTF.length();i++){
90                          if (pesanMTF.charAt(i)==pesanMTF.charAt(i-1)){
91                                  binRLE= binRLE+"1";
92                          } else {
93                                  binRLE = binRLE+"0";
94                                  pesanRLE = pesanRLE + pesanMTF.charAt(i);
95                          }
96                  }
97                  return pesanRLE;
98          }
99
100 /*      ======================== fungsi string to biner
101 ========================*/
102
103         @Override
104         public String pesanBiner(String pesanKompresi) {
105                 // TODO Auto-generated method stub
106                 String psnBiner = pesanKompresi + "11";
107                 return psnBiner;
108         }
109
110 /*      ========================== fungsi2 decompresi
111 ==========================*/
```

©UKDW

```
112
113         @Override
114         public String decompress(String binCode) {
115                 String binCode2="", byteKompresi, pesanBWT, pesanRahasia, pesanMTF;
116                 // TODO Auto-generated method stub
117                 byteKompresi = binCode.substring(0, binCode.length()-2);
118
119                 pesanMTF = decompRLE(byteKompresi);
120                 pesanBWT = reverseMTF (pesanMTF);
121                 pesanRahasia = reverseBWT(pesanBWT);
122                 binCode2=pesanRahasia;
123                 return binCode2;
124         }
125
126         public String decompRLE(String byteKompresi){
127                 String pesanRLE="", pesanMTF="", kodeRLE, binToSring;
128
129                 for (int i=0; i<byteKompresi.length(); i+=8){
130
131                         if(byteKompresi.length()-i < 9){
132                                 binToSring = byteKompresi.substring(i,
133 byteKompresi.length());
134                         }else binToSring = byteKompresi.substring(i, i+8);
135
136
137                         if (binToSring.substring(0, 1).equals("0")){
138                                 byte isi = Byte.parseByte(binToSring, 2);
139                                 String binDecimal = Byte.toString(isi);
140                                 char karakter = (char) Integer.parseInt(binDecimal);
141                                 pesanRLE = pesanRLE+karakter;
142                         } else break;
143                 }
144                 int idxKodeRLE = pesanRLE.length()*8 + 1;
145                 kodeRLE = byteKompresi.substring(idxKodeRLE);
146                 int j=0;
147                 for(int i=0;i<kodeRLE.length();i++){
148                         String tamp = kodeRLE.substring(i, i+1);
149                         if (tamp.equals("0")) {
150                                 pesanMTF= pesanMTF + pesanRLE.charAt(j) + " ";
151                                 j++;
152                         }else if (tamp.equals("1")){
153                                 pesanMTF= pesanMTF + pesanRLE.charAt(j-1) + " ";
154                         }
155                 }
156                 return pesanMTF;
157         }
158
159         public String reverseMTF(String pesanMTF){
160                 String mtfOnly = pesanMTF.substring(0, pesanMTF.length()-2);
161                 char idxBWT = pesanMTF.charAt(pesanMTF.length()-2);
162
163                 String[] dataString = mtfOnly.split(" ", -1);
164
165                 ArrayList<Integer> data = new ArrayList<Integer>();
166                 for(int i= 0; i<dataString.length-1;i++){
167                         data.add((int) dataString[i].charAt(0));
```

©UKDW

```
168                 }
169
170                 int[] dict = new int[128];
171                 for(int i = 0; i < 128; i++)
172                         dict[i] = i;
173                 String pesanBWT = "";
174                 //Performs the Move-To-Front transform on the ArrayList
175                 for(int dataVal : data)
176                 {
177                         char b = (char)dict[dataVal];
178                         for(int check = dataVal; check != 0; check--)
179                         {
180                                 dict[check] = dict[check-1];
181                         }
182                         dict[0] = b;
183                         pesanBWT = pesanBWT+ b;
184                 }
185                 return pesanBWT+idxBWT;
186         }
187
188         public String reverseBWT(String pesanBWT){
189                 String pesanRahasia="", bwtOnly, tamp;
190                 char idxAsli = pesanBWT.charAt(pesanBWT.length()-1);
191
192                 bwtOnly = pesanBWT.substring(0, pesanBWT.length()-1);
193                 String[] arrayKarakter = new String[bwtOnly.length()];
194                 String[] arrayKarakter2 = new String[bwtOnly.length()];
195
196                 for(int i=0;i<bwtOnly.length();i++){
197                         tamp = ""+bwtOnly.charAt(i);
198                         arrayKarakter[i] = tamp;
199                         arrayKarakter2[i]= tamp;
200                 }
201
202                 for(int i=0;i<bwtOnly.length()-1;i++){
203                         Arrays.sort(arrayKarakter2);
204                         for(int j=0;j<bwtOnly.length();j++){
205                                 String tamp1 = arrayKarakter[j];
206                                 String tamp2 = arrayKarakter2[j];
207                                 arrayKarakter[j] = tamp1 +
208 tamp2.charAt(tamp2.length()-1);
209                                 arrayKarakter2[j] = arrayKarakter[j];
210                         }
211                 }
212
213                 Arrays.sort(arrayKarakter);
214                 pesanRahasia = arrayKarakter[idxAsli];
215
216                 return pesanRahasia;
217         }
218
219 }
```

©UKDW

Steganografi.java

```
1   package com.ukdw.steganosms;
2
3   public class Steganografi {
4
5           public String encodeSteganografi (String binerKompresi){
6
7                   String pesanTersembunyi = "";
8                   for (int i=0; i<binerKompresi.length(); i++){
9                           char temp = binerKompresi.charAt(i);
10                          if (temp=='0')
11                                  pesanTersembunyi = pesanTersembunyi + (char)9;
12                          else if (temp=='1')
13                                  pesanTersembunyi = pesanTersembunyi + (char)9 +
14  (char)32;
15                  }
16                  //String pesanTersembunyi = binerKompresi;
17          return pesanTersembunyi;
18          }
19
20          public String decodeSteganografi (String isismsStego){
21                  String binCode="";
22                  String[] arraySpasi = isismsStego.split("\t", -1);
23
24                  for (int i=0; i < arraySpasi.length; i++){
25                          int jmlKarakter = arraySpasi[i].length();
26                          if (jmlKarakter == 0){
27                                  binCode = binCode + "0";
28                                  arraySpasi[i]= "0";
29                                  /*int value = 0;
30                                  int offset = (b.length - 1 - i) *8;
31                                  b[i] = (byte) ((value >>> offset) & 0xFF);*/
32                          }
33                          else if (jmlKarakter == 1){
34                                  binCode = binCode + "1";
35                                  arraySpasi[i]= "1";
36                                  /*int value = 1;
37                                  int offset = (b.length - 1 - i) *8;
38                                  b[i] = (byte) ((value >>> offset) & 0xFF);*/
39                          }
40                  }
41                  return binCode;
42          }
43  }
```

©UKDW

StegoSMS.java

```
1   package com.ukdw.steganosms;
2
3   import java.util.ArrayList;
4
5   import android.net.Uri;
6   import android.os.Bundle;
7   import android.app.Activity;
8   import android.app.AlertDialog;
9   import android.app.Dialog;
10  import android.app.PendingIntent;
11  import android.content.BroadcastReceiver;
12  import android.content.ContentValues;
13  import android.content.Context;
14  import android.content.DialogInterface;
15  import android.content.Intent;
16  import android.content.IntentFilter;
17  import android.content.SharedPreferences;
18  import android.telephony.SmsManager;
19  import android.view.KeyEvent;
20  import android.view.Menu;
21  import android.view.MenuItem;
22  import android.view.View;
23  import android.view.Window;
24  import android.widget.Button;
25  import android.widget.EditText;
26  import android.widget.TextView;
27  import android.widget.Toast;
28
29  public class StegoSMS extends Activity {
30
31          TextView textViewPhone, textViewSMS;
32          Button btnBatalPesanRahasia, btnKirimSMS;
33
34      @Override
35      public void onCreate(Bundle savedInstanceState) {
36          super.onCreate(savedInstanceState);
37          setContentView(R.layout.activity_stego_sms);
38      }
39
40      @Override
41         protected void onResume() {
42                 super.onResume();
43
44                 btnBatalPesanRahasia = (Button)
45  findViewById(R.id.btnBatalPesanRahasia);
46          btnKirimSMS = (Button) findViewById(R.id.btnKirimSMS);
47          textViewPhone = (TextView) findViewById(R.id.textViewPhone);
48          textViewSMS = (TextView) findViewById(R.id.textViewSMS);
49
50          Intent tambah = getIntent();
51          textViewPhone.setText(tambah.getStringExtra("no_HP"));
52          textViewSMS.setText(tambah.getStringExtra("isi_SMS"));
53
54          btnKirimSMS.setOnClickListener(new View.OnClickListener() {
55
```

©UKDW

```
56                          @Override
57                          public void onClick(View v) {
58                                 String ponsel = textViewPhone.getText().toString();
59                                 String isisms = textViewSMS.getText().toString();
60                                 kirimSMS(ponsel, isisms);
61                                 saveSendItem(ponsel, isisms);
62                          }
63                   });
64
65             btnBatalPesanRahasia.setOnClickListener(new View.OnClickListener() {
66
67                          @Override
68                          public void onClick(View v) {
69                                 String ponsel = textViewPhone.getText().toString();
70                                 String isisms = textViewSMS.getText().toString();
71                                 batalPesan(ponsel, isisms);
72                          }
73
74                   });
75             }
76
77          @Override
78          public boolean onCreateOptionsMenu(Menu menu) {
79              // Inflate the menu; this adds items to the action bar if it is present.
80              getMenuInflater().inflate(R.menu.main_sm, menu);
81              return true;
82          }
83
84          public boolean onOptionsItemSelected(MenuItem item) {
85              switch (item.getItemId()) {
86              case R.id.ubahpass:
87                    // TODO Auto-generated method stub
88                    final Dialog ubah = new Dialog(StegoSMS.this);
89                    ubah.requestWindowFeature(Window.FEATURE_NO_TITLE);
90                    ubah.setContentView(R.layout.ubah_pass);
91
92                    Button ubahpass = (Button)ubah.findViewById(R.id.ubahPassword);
93                    ubahpass.setOnClickListener(new View.OnClickListener() {
94                          @Override
95                          public void onClick(View v) {
96                                 // TODO Auto-generated method stub
97                                 EditText pass_lama =
98    (EditText)ubah.findViewById(R.id.passwordLama);
99                                 EditText pass_baru =
100   (EditText)ubah.findViewById(R.id.passwordBaru);
101                                String kunci_lama = pass_lama.getText().toString();
102                                String kunci_baru = pass_baru.getText().toString();
103                   SharedPreferences passPreference = getSharedPreferences("Preference
104                     Password",Context.MODE_PRIVATE);
105                                SharedPreferences.Editor passPrefEdit =
106   passPreference.edit();
107                                String kunci_tersimpan=
108   passPreference.getString("pass_rahasia", "ukdw");
109                                if(kunci_tersimpan.equals(kunci_lama)){
110                                       passPrefEdit.putString("pass_rahasia",
111   kunci_baru);
```

©UKDW

```
112                                     passPrefEdit.commit();
113                                     Toast.makeText(getBaseContext(),"Password Baru
114   Tersimpan",
115                                             Toast.LENGTH_SHORT).show();
116                                     ubah.dismiss();
117                                 }else {
118                                     Toast.makeText(getBaseContext(),"Password Lama
119   Salah!",
120                                             Toast.LENGTH_SHORT).show();
121                                     pass_lama.setText("");
122                                     pass_baru.setText("");
123                                 }
124                             }
125                 });
126                 ubah.show();
127             return true;
128         case R.id.about:
129                 AlertDialog.Builder alertMax = new
130   AlertDialog.Builder(StegoSMS.this);
131          alertMax.setMessage("SMS Rahasia adalah aplikasi sms yang menerapkan
132             kompresi" + " dan steganografi pada pesan yang ingin dirahasiakan.
133             Aplikasi ini dibuat" + " oleh Yudhi Putra Tama (22084583) untuk memenuhi
134             Tugas Akhir program studi" + " Teknik Informatika UKDW.");
135                         alertMax.setTitle("Tentang Aplikasi");
136
137          alertMax.setPositiveButton("Kembali", new
138             DialogInterface.OnClickListener() {
139
140                                 @Override
141                                 public void onClick(DialogInterface dialog, int id) {
142                                     dialog.cancel();
143                                 }
144                         });
145                 AlertDialog dialog = alertMax.create();
146                 dialog.show();
147          return true;
148          default:
149              return super.onOptionsItemSelected(item);
150          }
151      }
152
153      public void buatSMS (View view){
154         Intent buat = new Intent(this, MainSMS.class);
155         startActivity(buat);
156      }
157
158      public void masukSMS (View view){
159         Intent masuk = new Intent(this, MasukSMS.class);
160         startActivity(masuk);
161      }
162
163      public void terkirimSMS (View view){
164         Intent terkirim = new Intent(this, TerkirimSMS.class);
165         startActivity(terkirim);
166      }
167
```

©UKDW

```
168       public void batalPesan (String ponsel, String isisms){
169          String smsAsli = isisms.substring(0, isisms.indexOf("      "));
170          Intent batal = new Intent(this, MainSMS.class);
171          batal.putExtra("no_HP", ponsel);
172          batal.putExtra("isi_SMS", smsAsli);
173          startActivity(batal);
174       }
175
176       public void kirimSMS (String ponsel, String isisms){
177
178          String SENT = "SMS_SENT";
179          String DELIVERED = "SMS_DELIVERED";
180
181          Intent mSentIntent = new Intent(SENT);
182          Intent mDeliveryIntent = new Intent(DELIVERED);
183
184                SmsManager smsMulti = SmsManager.getDefault();
185                ArrayList<String> bagian = smsMulti.divideMessage(isisms);
186
187                int numParts = bagian.size();
188                if (numParts <= 5){
189                ArrayList<PendingIntent> sentIntents = new
190    ArrayList<PendingIntent>();
191                ArrayList<PendingIntent> deliveryIntents = new
192    ArrayList<PendingIntent>();
193                for (int i=0; i < numParts; i++){
194                      sentIntents.add(PendingIntent.getBroadcast(this, 0,
195    mSentIntent, 0));
196                      deliveryIntents.add(PendingIntent.getBroadcast(this, 0,
197    mDeliveryIntent, 0));
198                      }
199
200         smsMulti.sendMultipartTextMessage(ponsel, null, bagian, sentIntents,
201            deliveryIntents);
202                }
203                else{
204                      Toast.makeText(getBaseContext(),"SMS lebih dari 5 bagian",
205                            Toast.LENGTH_SHORT).show();
206                }
207
208          registerReceiver(new BroadcastReceiver(){
209              @Override
210              public void onReceive(Context arg0, Intent arg1) {
211
212                  switch (getResultCode())
213                  {
214                      case Activity.RESULT_OK:
215                        Intent kembali = new Intent(StegoSMS.this, MainSMS.class);
216                              kembali.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
217                              startActivity(kembali);
218                        Toast.makeText(getBaseContext(), "SMS dikirim",
219                                Toast.LENGTH_SHORT).show();
220                        break;
221                      case SmsManager.RESULT_ERROR_GENERIC_FAILURE:
222                        Toast.makeText(getBaseContext(), "Generic failure",
223                                Toast.LENGTH_SHORT).show();
```

©UKDW

```
                        break;
                    case SmsManager.RESULT_ERROR_NO_SERVICE:
                        Toast.makeText(getBaseContext(), "No service",
                                Toast.LENGTH_SHORT).show();
                        break;
                    case SmsManager.RESULT_ERROR_NULL_PDU:
                        Toast.makeText(getBaseContext(), "Null PDU",
                                Toast.LENGTH_SHORT).show();
                        break;
                    case SmsManager.RESULT_ERROR_RADIO_OFF:
                        Toast.makeText(getBaseContext(), "Radio off",
                                Toast.LENGTH_SHORT).show();
                        break;
                }

            }
        }, new IntentFilter(SENT));



        registerReceiver(new BroadcastReceiver(){
            @Override
            public void onReceive(Context arg0,Intent arg1) {
                switch (getResultCode())
                {
                    case Activity.RESULT_OK:
                      Intent keterkirim = new Intent(StegoSMS.this, TerkirimSMS.class);

        keterkirim.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                                    startActivity(keterkirim);
                        Toast.makeText(getBaseContext(), "SMS terkirim",
                                Toast.LENGTH_SHORT).show();
                        break;
                    case Activity.RESULT_CANCELED:
                        Toast.makeText(getBaseContext(), "SMS gagal",
                        Toast.LENGTH_SHORT).show();
                                break;
                }
            }
        }, new IntentFilter(DELIVERED));

    }


    public void saveSendItem(String ponsel, String isisms) {
        ContentValues values = new ContentValues();
        values.put("address", ponsel);
        values.put("body", isisms);
        getContentResolver().insert(Uri.parse("content://sms/sent"), values);
    }

    public boolean onKeyDown(int keyCode, KeyEvent event){
              if (keyCode == KeyEvent.KEYCODE_BACK && event.getRepeatCount() == 0){
                    AlertDialog.Builder alertMax = new AlertDialog.Builder(this);
                    alertMax.setMessage("Anda yakin ingin keluar dari
aplikasi?");
```

©UKDW

```
280                         alertMax.setTitle("Tutup Aplikasi");
281                         alertMax.setPositiveButton("Ya", new
282     DialogInterface.OnClickListener() {
283                                 @Override
284                                 public void onClick(DialogInterface dialog, int id) {
285                                         Intent exit = new Intent(Intent.ACTION_MAIN);
286                                         exit.addCategory(Intent.CATEGORY_HOME);
287                                         exit.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
288                                         startActivity(exit);
289                                 }
290                         });
291
292                         alertMax.setNegativeButton("Tidak", new
293     DialogInterface.OnClickListener() {
294                                 @Override
295                                 public void onClick(DialogInterface dialog, int id) {
296                                         dialog.cancel();
297                                 }
298                         });
299
300                 AlertDialog dialog = alertMax.create();
301                 dialog.show();
302
303                         return true;
304                 }
305                 return super.onKeyDown(keyCode, event);
306         }
307     }
```

©UKDW

AmbilSMS.java

```
1   package com.ukdw.steganosms;
2
3   import android.net.Uri;
4   import android.app.Activity;
5   import android.database.Cursor;
6
7   public class AmbilSMS extends Activity {
8
9           public String getNo(int i, String lokasi){
10                  String no = "";
11                  int n = i;
12                  Uri uriSMSURI = Uri.parse("content://sms/"+lokasi);
13                  Cursor cur = getContentResolver().query(uriSMSURI, null, null, null,
14  null);
15                          if (cur.getCount() > n){
16                          cur.moveToPosition(n);
17                                  no = cur.getString(cur.getColumnIndex("address"));
18                          }
19                  return no;
20          }
21
22          public String getSMS(int i, String lokasi){
23                  String sms = "";
24                  int n = i;
25                  Uri uriSMSURI = Uri.parse("content://sms/"+lokasi);
26                  Cursor cur = getContentResolver().query(uriSMSURI, null, null, null,
27  null);
28                          if (cur.getCount() > n){
29                          cur.moveToPosition(n);
30                          sms = cur.getString(cur.getColumnIndexOrThrow("body"));
31                          }
32                  return sms;
33          }
34
35  }
```

©UKDW

MasukSMS.java

```
1   package com.ukdw.steganosms;
2
3   import java.util.ArrayList;
4   import java.util.HashMap;
5   import java.util.List;
6   import java.util.Map;
7   import android.os.Bundle;
8   import android.app.AlertDialog;
9   import android.app.Dialog;
10  import android.content.Context;
11  import android.content.DialogInterface;
12  import android.content.Intent;
13  import android.content.SharedPreferences;
14  import android.view.KeyEvent;
15  import android.view.Menu;
16  import android.view.MenuItem;
17  import android.view.View;
18  import android.view.Window;
19  import android.widget.AdapterView;
20  import android.widget.Button;
21  import android.widget.EditText;
22  import android.widget.ListView;
23  import android.widget.SimpleAdapter;
24  import android.widget.Toast;
25  import android.widget.AdapterView.OnItemClickListener;
26
27  public class MasukSMS extends AmbilSMS {
28
29          ListView listMasukSMS;
30
31          @Override
32          protected void onCreate(Bundle savedInstanceState) {
33                  super.onCreate(savedInstanceState);
34                  setContentView(R.layout.activity_masuk_sms);
35
36                  listMasukSMS = (ListView) findViewById(R.id.ListMasukSMS);
37
38                  final List<Map<String, String>> data = new ArrayList<Map<String,
39  String>>();
40                              for (int i=0; i<20; i++){
41                                      Map<String, String> data2 = new HashMap<String,
42  String>(2);
43                                      data2.put("penerima", getNo(i, "inbox"));
44                                      String isi = getSMS(i, "inbox");
45                                      data2.put("isi", isi);
46                                      if (isi.contains("\t"))
47  data2.put("viewIsi",isi.substring(0,isi.indexOf("\t")));
48                                      else data2.put("viewIsi", isi);
49                                      data.add(data2);
50                          }
51
52          SimpleAdapter adapter = new SimpleAdapter(this, data,
53             android.R.layout.simple_list_item_2, new String[] {"penerima","viewIsi"},
54             new int[] {android.R.id.text1, android.R.id.text2});
55                  listMasukSMS.setAdapter(adapter);
```

©UKDW

```
56
57                  listMasukSMS.setOnItemClickListener(new OnItemClickListener(){
58                        @Override
59                        public void onItemClick (AdapterView<?> parent, View view, int
60    position, long id){
61                              Map<String, String> map = data.get(position);
62                              String ponsel = map.get("penerima");
63                              String isisms = map.get("isi");
64
65                              Intent lihat = new Intent(MasukSMS.this,
66    LihatSMS.class);
67                        lihat.putExtra("no_HP", ponsel);
68                        lihat.putExtra("isi_SMS", isisms);
69                        startActivity(lihat);
70                        }
71                  });
72          }
73
74          @Override
75          public boolean onCreateOptionsMenu(Menu menu) {
76                  // Inflate the menu; this adds items to the action bar if it is
77    present.
78                  getMenuInflater().inflate(R.menu.masuk_sm, menu);
79                  return true;
80          }
81
82          public boolean onOptionsItemSelected(MenuItem item) {
83           switch (item.getItemId()) {
84           case R.id.ubahpass:
85                  // TODO Auto-generated method stub
86                  final Dialog ubah = new Dialog(MasukSMS.this);
87                  ubah.requestWindowFeature(Window.FEATURE_NO_TITLE);
88                  ubah.setContentView(R.layout.ubah_pass);
89
90                  Button ubahpass = (Button)ubah.findViewById(R.id.ubahPassword);
91                  ubahpass.setOnClickListener(new View.OnClickListener() {
92                        @Override
93                        public void onClick(View v) {
94                              // TODO Auto-generated method stub
95                              EditText pass_lama =
96    (EditText)ubah.findViewById(R.id.passwordLama);
97                              EditText pass_baru =
98    (EditText)ubah.findViewById(R.id.passwordBaru);
99                              String kunci_lama = pass_lama.getText().toString();
100                             String kunci_baru = pass_baru.getText().toString();
101                 SharedPreferences passPreference = getSharedPreferences("Preference
102                    Password",Context.MODE_PRIVATE);
103                             SharedPreferences.Editor passPrefEdit =
104   passPreference.edit();
105                             String kunci_tersimpan=
106   passPreference.getString("pass_rahasia", "ukdw");
107                             if(kunci_tersimpan.equals(kunci_lama)){
108                                   passPrefEdit.putString("pass_rahasia",
109   kunci_baru);
110                                   passPrefEdit.commit();
```

©UKDW

```
111                                     Toast.makeText(getBaseContext(),"Password Baru
112    Tersimpan",
113                                             Toast.LENGTH_SHORT).show();
114                                     ubah.dismiss();
115                                 }else {
116                                     Toast.makeText(getBaseContext(),"Password Lama
117    Salah!",
118                                             Toast.LENGTH_SHORT).show();
119                                     pass_lama.setText("");
120                                     pass_baru.setText("");
121                                 }
122                             }
123                 });
124                 ubah.show();
125             return true;
126         case R.id.about:
127                 AlertDialog.Builder alertMax = new
128    AlertDialog.Builder(MasukSMS.this);
129          alertMax.setMessage("SMS Rahasia adalah aplikasi sms yang menerapkan
130             kompresi" + " dan steganografi pada pesan yang ingin dirahasiakan.
131             Aplikasi ini dibuat" + " oleh Yudhi Putra Tama (22084583) untuk memenuhi
132             Tugas Akhir program studi" + " Teknik Informatika UKDW.");
133                         alertMax.setTitle("Tentang Aplikasi");
134                         alertMax.setPositiveButton("Kembali", new
135    DialogInterface.OnClickListener() {
136
137                                 @Override
138                                 public void onClick(DialogInterface dialog, int id) {
139                                     dialog.cancel();
140                                 }
141                         });
142                 AlertDialog dialog = alertMax.create();
143                 dialog.show();
144             return true;
145          default:
146             return super.onOptionsItemSelected(item);
147          }
148
149      }
150
151         public void buatSMS (View view){
152         Intent buat = new Intent(this, MainSMS.class);
153         startActivity(buat);
154      }
155
156      public void masukSMS (View view){
157         Intent masuk = new Intent(this, MasukSMS.class);
158         startActivity(masuk);
159      }
160
161      public void terkirimSMS (View view){
162         Intent terkirim = new Intent(this, TerkirimSMS.class);
163         startActivity(terkirim);
164      }
165
166      public boolean onKeyDown(int keyCode, KeyEvent event){
```

©UKDW

```
167                 if (keyCode == KeyEvent.KEYCODE_BACK && event.getRepeatCount() == 0){
168                         AlertDialog.Builder alertMax = new AlertDialog.Builder(this);
169                         alertMax.setMessage("Anda yakin ingin keluar dari
170   aplikasi?");
171                         alertMax.setTitle("Tutup Aplikasi");
172                         alertMax.setPositiveButton("Ya", new
173   DialogInterface.OnClickListener() {
174                                 @Override
175                                 public void onClick(DialogInterface dialog, int id) {
176                                         Intent exit = new Intent(Intent.ACTION_MAIN);
177                                         exit.addCategory(Intent.CATEGORY_HOME);
178                                         exit.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
179                                         startActivity(exit);
180                                         }
181                         });
182
183                         alertMax.setNegativeButton("Tidak", new
184   DialogInterface.OnClickListener() {
185                                 @Override
186                                 public void onClick(DialogInterface dialog, int id) {
187                                         dialog.cancel();
188                                 }
189                         });
190
191                 AlertDialog dialog = alertMax.create();
192                 dialog.show();
193
194                         return true;
195                 }
196                 return super.onKeyDown(keyCode, event);
197         }
198
199   }
```

©UKDW

TerkirimSMS.java

```
1   package com.ukdw.steganosms;
2
3   import java.util.ArrayList;
4   import java.util.HashMap;
5   import java.util.List;
6   import java.util.Map;
7   import android.os.Bundle;
8   import android.app.AlertDialog;
9   import android.app.Dialog;
10  import android.content.Context;
11  import android.content.DialogInterface;
12  import android.content.Intent;
13  import android.content.SharedPreferences;
14  import android.view.KeyEvent;
15  import android.view.Menu;
16  import android.view.MenuItem;
17  import android.view.View;
18  import android.view.Window;
19  import android.widget.AdapterView;
20  import android.widget.Button;
21  import android.widget.EditText;
22  import android.widget.ListView;
23  import android.widget.SimpleAdapter;
24  import android.widget.Toast;
25  import android.widget.AdapterView.OnItemClickListener;
26
27  public class TerkirimSMS extends AmbilSMS {
28
29          ListView listTerkirimSMS;
30
31          @Override
32          protected void onCreate(Bundle savedInstanceState) {
33                  super.onCreate(savedInstanceState);
34                  setContentView(R.layout.activity_terkirim_sms);
35
36                  listTerkirimSMS = (ListView) findViewById(R.id.listTerkirimSMS);
37
38                  final List<Map<String, String>> data = new ArrayList<Map<String,
39  String>>();
40                  for (int i=0; i<20; i++){
41                                  Map<String, String> data2 = new HashMap<String,
42  String>(2);
43                                  data2.put("penerima", getNo(i, "sent"));
44                                  String isi = getSMS(i, "sent");
45                                  data2.put("isi", isi);
46                                  if (isi.contains("\t")) data2.put("viewIsi", isi.substring(0,
47  isi.indexOf("\t")));
48                                  else data2.put("viewIsi", isi);
49                                  data.add(data2);
50                          }
51
52        SimpleAdapter adapter = new SimpleAdapter(this, data,
53           android.R.layout.simple_list_item_2, new String[] {"penerima","viewIsi"},
54           new int[] {android.R.id.text1, android.R.id.text2});
55                  listTerkirimSMS.setAdapter(adapter);
```

©UKDW

```
56                  listTerkirimSMS.setOnItemClickListener(new OnItemClickListener(){
57                          @Override
58                          public void onItemClick (AdapterView<?> parent, View view, int
59      position, long id){
60                                  Map<String, String> map = data.get(position);
61                                  String ponsel = map.get("penerima");
62                                  String isisms = map.get("isi");
63
64                                  Intent lihat = new Intent(TerkirimSMS.this,
65      LihatSMS.class);
66                          lihat.putExtra("no_HP", ponsel);
67                          lihat.putExtra("isi_SMS", isisms);
68                          startActivity(lihat);
69                          }
70                  });
71          }
72
73          @Override
74          public boolean onCreateOptionsMenu(Menu menu) {
75                  // Inflate the menu; this adds items to the action bar if it is
76      present.
77                  getMenuInflater().inflate(R.menu.terkirim_sm, menu);
78                  return true;
79          }
80
81          public boolean onOptionsItemSelected(MenuItem item) {
82           switch (item.getItemId()) {
83           case R.id.ubahpass:
84                  // TODO Auto-generated method stub
85                  final Dialog ubah = new Dialog(TerkirimSMS.this);
86                  ubah.requestWindowFeature(Window.FEATURE_NO_TITLE);
87                  ubah.setContentView(R.layout.ubah_pass);
88
89                  Button ubahpass = (Button)ubah.findViewById(R.id.ubahPassword);
90                  ubahpass.setOnClickListener(new View.OnClickListener() {
91                          @Override
92                          public void onClick(View v) {
93                                  // TODO Auto-generated method stub
94                                  EditText pass_lama =
95      (EditText)ubah.findViewById(R.id.passwordLama);
96                                  EditText pass_baru =
97      (EditText)ubah.findViewById(R.id.passwordBaru);
98                                  String kunci_lama = pass_lama.getText().toString();
99                                  String kunci_baru = pass_baru.getText().toString();
100                 SharedPreferences passPreference = getSharedPreferences("Preference
101                    Password",Context.MODE_PRIVATE);
102                                 SharedPreferences.Editor passPrefEdit =
103     passPreference.edit();
104                                 String kunci_tersimpan=
105     passPreference.getString("pass_rahasia", "ukdw");
106                                 if(kunci_tersimpan.equals(kunci_lama)){
107                                         passPrefEdit.putString("pass_rahasia",
108     kunci_baru);
109                                         passPrefEdit.commit();
110                                         Toast.makeText(getBaseContext(),"Password Baru
111     Tersimpan",
```

©UKDW

```
112                                                     Toast.LENGTH_SHORT).show();
113                                     ubah.dismiss();
114                                 }else {
115                                     Toast.makeText(getBaseContext(),"Password Lama
116   Salah!",
117                                                     Toast.LENGTH_SHORT).show();
118                                     pass_lama.setText("");
119                                     pass_baru.setText("");
120                                 }
121                             }
122                 });
123                 ubah.show();
124             return true;
125         case R.id.about:
126                 AlertDialog.Builder alertMax = new
127   AlertDialog.Builder(TerkirimSMS.this);
128          alertMax.setMessage("SMS Rahasia adalah aplikasi sms yang menerapkan
129             kompresi" + " dan steganografi pada pesan yang ingin dirahasiakan.
130             Aplikasi ini dibuat" + " oleh Yudhi Putra Tama (22084583) untuk memenuhi
131             Tugas Akhir program studi" + " Teknik Informatika UKDW.");
132                         alertMax.setTitle("Tentang Aplikasi");
133                         alertMax.setPositiveButton("Kembali", new
134   DialogInterface.OnClickListener() {
135
136                                 @Override
137                                 public void onClick(DialogInterface dialog, int id) {
138                                     dialog.cancel();
139                                 }
140                             });
141                 AlertDialog dialog = alertMax.create();
142                 dialog.show();
143             return true;
144          default:
145             return super.onOptionsItemSelected(item);
146          }
147      }
148
149         public void buatSMS (View view){
150         Intent buat = new Intent(this, MainSMS.class);
151         startActivity(buat);
152      }
153
154      public void masukSMS (View view){
155         Intent masuk = new Intent(this, MasukSMS.class);
156         startActivity(masuk);
157      }
158
159      public void terkirimSMS (View view){
160         Intent terkirim = new Intent(this, TerkirimSMS.class);
161         startActivity(terkirim);
162      }
163
164      public boolean onKeyDown(int keyCode, KeyEvent event){
165                 if (keyCode == KeyEvent.KEYCODE_BACK && event.getRepeatCount() == 0){
166                         AlertDialog.Builder alertMax = new AlertDialog.Builder(this);
```

©UKDW

```
167                         alertMax.setMessage("Anda yakin ingin keluar dari
168     aplikasi?");
169                         alertMax.setTitle("Tutup Aplikasi");
170                         alertMax.setPositiveButton("Ya", new
171     DialogInterface.OnClickListener() {
172                                 @Override
173                                 public void onClick(DialogInterface dialog, int id) {
174                                         Intent exit = new Intent(Intent.ACTION_MAIN);
175                                         exit.addCategory(Intent.CATEGORY_HOME);
176                                         exit.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
177                                         startActivity(exit);
178                                 }
179                         });
180
181                         alertMax.setNegativeButton("Tidak", new
182     DialogInterface.OnClickListener() {
183                                 @Override
184                                 public void onClick(DialogInterface dialog, int id) {
185                                         dialog.cancel();
186                                 }
187                         });
188
189                 AlertDialog dialog = alertMax.create();
190                 dialog.show();
191
192                         return true;
193                 }
194                 return super.onKeyDown(keyCode, event);
195         }
196     }
```

©UKDW

LihatSMS.java

```
1   package com.ukdw.steganosms;
2
3   import android.os.Bundle;
4   import android.app.Activity;
5   import android.app.Dialog;
6   import android.content.Context;
7   import android.content.Intent;
8   import android.content.SharedPreferences;
9   import android.view.Menu;
10  import android.view.View;
11  import android.view.Window;
12  import android.widget.Button;
13  import android.widget.EditText;
14  import android.widget.TextView;
15  import android.widget.Toast;
16
17  public class LihatSMS extends Activity {
18
19          Button btnLihatPesanRahasia;
20          String ponsel, isisms;
21          TextView textViewPhone, textViewSMS;
22
23          @Override
24          protected void onCreate(Bundle savedInstanceState) {
25                  super.onCreate(savedInstanceState);
26                  setContentView(R.layout.activity_lihat_sms);
27                  btnLihatPesanRahasia = (Button)
28  findViewById(R.id.btnLihatPesanRahasia);
29          }
30
31          @Override
32          protected void onResume() {
33                  super.onResume();
34                  Intent tambah = getIntent();
35                  ponsel = tambah.getStringExtra("no_HP");
36                  isisms = tambah.getStringExtra("isi_SMS");
37                  textViewPhone = (TextView) findViewById(R.id.textViewPhone);
38                  textViewSMS = (TextView) findViewById(R.id.textViewSMS);
39
40                  textViewPhone.setText("Pengirim : " + ponsel);
41                  textViewSMS.setText(isisms);
42
43                  btnLihatPesanRahasia.setOnClickListener(new View.OnClickListener() {
44                          @Override
45                          public void onClick(View v) {
46                                  loginDialog();
47                          }
48                  });
49          }
50
51          @Override
52          public boolean onCreateOptionsMenu(Menu menu) {
53                  // Inflate the menu; this adds items to the action bar if it is
54  present.
55                  getMenuInflater().inflate(R.menu.lihat_sm, menu);
```

©UKDW

```
56                  return true;
57          }
58          public void kembali (View view){
59          finish();
60      }
61
62          protected void loginDialog() {
63                  // TODO Auto-generated method stub
64                  final Dialog login = new Dialog(LihatSMS.this);
65                  login.requestWindowFeature(Window.FEATURE_NO_TITLE);
66                  login.setContentView(R.layout.login_form);
67
68                  Button masuk = (Button)login.findViewById(R.id.login);
69                  masuk.setOnClickListener(new View.OnClickListener() {
70                          @Override
71                          public void onClick(View v) {
72                                  // TODO Auto-generated method stub
73                                  EditText pass =
74  (EditText)login.findViewById(R.id.password);
75                                  String kunci = pass.getText().toString();
76              SharedPreferences passPreference = getSharedPreferences("Preference
77                 Password",Context.MODE_PRIVATE);
78                                  String kunci_tersimpan =
79  passPreference.getString("pass_rahasia", "ukdw");
80                                  if(kunci_tersimpan.equals(kunci)){
81                                          Intent isiRahasia = new Intent(LihatSMS.this,
82  LihatRahasia.class);
83                                          isiRahasia.putExtra("no_HP", ponsel);
84                                          isiRahasia.putExtra("isi_SMS", isisms);
85                                  startActivity(isiRahasia);
86                                  login.dismiss();
87                                  }else {
88                                          Toast.makeText(getBaseContext(),"Password
89  Salah!",
90                                                          Toast.LENGTH_SHORT).show();
91                                  pass.setText("");
92                          }
93                  }
94                  });
95
96                  login.show();
97          }
98
99  }
```

©UKDW

LihatRahasia.java

```
1   package com.ukdw.steganosms;
2
3   import android.os.Bundle;
4   import android.app.Activity;
5   import android.app.AlertDialog;
6   import android.content.DialogInterface;
7   import android.content.Intent;
8   import android.view.Menu;
9   import android.view.View;
10  import android.widget.TextView;
11
12  public class LihatRahasia extends Activity {
13
14          String pesan, ponsel, isisms, pesanRahasia, detail;
15          TextView textViewPhone, textViewSMS, textViewPesanRahasia;
16
17          @Override
18          protected void onCreate(Bundle savedInstanceState) {
19                  super.onCreate(savedInstanceState);
20                  setContentView(R.layout.activity_lihat_rahasia);
21          }
22
23          @Override
24          protected void onResume() {
25                  super.onResume();
26                  Intent isiRahasia = getIntent();
27                  pesan = isiRahasia.getStringExtra("isi_SMS");
28                  ponsel = isiRahasia.getStringExtra("no_HP");
29                  lihatPesan(pesan);
30
31                  textViewPhone = (TextView) findViewById(R.id.textViewPhone);
32                  textViewSMS = (TextView) findViewById(R.id.textViewSMS);
33                  textViewPesanRahasia = (TextView)
34  findViewById(R.id.textViewPesanRahasia);
35
36                  textViewPhone.setText("Pengirim : " + ponsel);
37                  textViewSMS.setText(isisms);
38                  textViewPesanRahasia.setText(pesanRahasia);
39          }
40
41          @Override
42          public boolean onCreateOptionsMenu(Menu menu) {
43                  // Inflate the menu; this adds items to the action bar if it is
44  present.
45                  getMenuInflater().inflate(R.menu.lihat_rahasia, menu);
46                  return true;
47          }
48
49          public void detail (View view){
50                  AlertDialog.Builder alertMax = new AlertDialog.Builder(this);
51                  alertMax.setMessage(detail);
52                  alertMax.setTitle("Detail Pesan");
53                  alertMax.setPositiveButton("Kembali", new
54  DialogInterface.OnClickListener() {
55
```

©UKDW

```
56                      @Override
57                      public void onClick(DialogInterface dialog, int id) {
58                              dialog.cancel();
59                      }
60              });
61          AlertDialog dialog = alertMax.create();
62          dialog.show();
63      }
64
65          public void lihatPesan (String pesan){
66                  int pjgTotalSMS = pesan.length(),
67                          pjgTeksSMS = 0,
68                          pjgPesanRahasia = 0,
69                          pjgPesanStego = 0,
70                          pjgBinerPesanStego = 0;
71
72                  if (pesan.contains("\t")){
73                          String isismsStego, binCode, algoritma="";
74                          isisms = pesan.substring(0, pesan.indexOf("\t"));
75                          isismsStego = pesan.substring(pesan.indexOf("\t") + 1);
76                          pjgTeksSMS = isisms.length();
77
78                          Steganografi stegano = new Steganografi();
79                          binCode = stegano.decodeSteganografi(isismsStego);
80                          pjgBinerPesanStego = binCode.length()-2;
81                          String algoKompresi = binCode.substring(binCode.length()-2,
82  binCode.length());
83                          pesanRahasia = "";
84
85                          if (algoKompresi.equals("00")) {
86                                  algoritma = "Tanpa Kompresi";
87                                  pjgPesanStego = pjgTotalSMS - pjgTeksSMS - 2;
88                          }
89                          else if (algoKompresi.equals("01")) {
90                                  algoritma = "Huffman";
91                                  pjgPesanStego = pjgTotalSMS - pjgTeksSMS - 3;
92                          }
93                          else if (algoKompresi.equals("10")) {
94                                  algoritma = "LZW";
95                                  pjgPesanStego = pjgTotalSMS - pjgTeksSMS - 3;
96                          }
97                          else if (algoKompresi.equals("11")) {
98                                  algoritma = "BWT";
99                                  pjgPesanStego = pjgTotalSMS - pjgTeksSMS - 4;
100                         }
101
102                         pjgTotalSMS = pjgTeksSMS+pjgPesanStego;
103                         Kompresi kompresi = KompresiFactory.algoKompresi(algoritma);
104                         pesanRahasia = kompresi.decompress(binCode);
105                         pjgPesanRahasia = pesanRahasia.length();
106          detail = detailPesan(algoritma, pjgTotalSMS, pjgTeksSMS, pjgPesanStego,
107          pjgPesanRahasia, pjgBinerPesanStego);
108
109                 } else {
110                         isisms = pesan;
111                         pesanRahasia = "tidak ada pesan rahasia.";
```

©UKDW

```
112                         detail ="Panjang SMS : " + pjgTotalSMS;
113                 }
114         }
115
116     public String detailPesan (String metode, int pjgTotalSMS, int pjgTeksSMS, int
117        pjgPesanStego, int pjgPesanRahasia, int pjgBinerPesanStego ){
118                 String detailPesan = "Algoritma Kompresi : " + metode + (char)10 +
119                                 "Panjang total SMS : " + pjgTotalSMS + (char)10 +
120                                 "Panjang teks SMS : " + pjgTeksSMS + (char)10 +
121                                 "Panjang pesan steganografi: " + pjgPesanStego +
122   (char)10 + (char)10 +
123                                 "Panjang pesan rahasia: " + pjgPesanRahasia + (char)10
124   +
125                                 "Panjang biner pesan steganografi: " +
126   pjgBinerPesanStego;
127                 return detailPesan;
128         }
129
130         public void kembali (View view){
131         finish();
132     }
133   }
```

©UKDW

activity_main_sms.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="0dp"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"

    tools:context=".MainSMS" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingBottom="5dp"
        android:background="#ffde00" >

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnBuatSMS"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="#ffde00"
            android:onClick="buatSMS"
            android:text="@string/buat_sms"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />


        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSMasuk"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="masukSMS"
            android:text="@string/title_activity_masuk_sms"
            android:textColor="#ffffff"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSTerkirim"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="terkirimSMS"
            android:text="@string/title_activity_terkirim_sms"
            android:textColor="#ffffff"
```

©UKDW

```
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />
    </LinearLayout>

    <EditText
        android:id="@+id/textPhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#ffffff"
        android:ems="10"
        android:hint="@string/nomor_tujuan"
        android:inputType="phone"
        android:padding="5dp"
        android:textAppearance="?android:attr/textAppearanceLarge" />

    <EditText
        android:id="@+id/textSMS"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_marginTop="5dp"
        android:layout_weight="1"
        android:background="#ffffff"
        android:ems="10"
        android:gravity="center_vertical|top"
        android:hint="@string/isi_sms"
        android:inputType="textMultiLine"
        android:padding="5dp">
        <requestFocus />
    </EditText>

    <Button
        android:id="@+id/btnTambahPesanRahasia"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="5dp"
        android:layout_marginTop="5dp"
        android:background="@drawable/button_selector"
        android:textColor="#000000"
        android:textSize="@dimen/button"
        android:text="@string/tambah_pesan" />

    <Button
        android:id="@+id/btnKirimSMS"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="0dp"
        android:layout_marginLeft="5dp"
        android:layout_marginRight="5dp"
        android:layout_marginTop="2dp"
        android:background="@drawable/button_selector"
        android:textColor="#000000"
        android:textSize="@dimen/button"
        android:text="@string/kirim_sms" />
</LinearLayout>
```

©UKDW

activity_pesan_tersembunyi.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".PesanTersembunyi" >

    <TextView
        android:id="@+id/viewTulisPesanRahasia"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#ffde00"
        android:padding="5dp"
        android:text="@string/tulis_pesan"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textStyle="bold" />

    <EditText
        android:id="@+id/textRahasia"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#ffffff"
        android:padding="10dp"
        android:ems="10"
        android:gravity="center_vertical|top"
        android:layout_below="@+id/viewTulisPesanRahasia"
        android:hint="@string/isi_pesan"
        android:inputType="text">

        <requestFocus />
    </EditText>

    <TextView
        android:id="@+id/viewAlgoKompresi"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textRahasia"
        android:layout_marginTop="5dp"
        android:background="#ffde00"
        android:padding="5dp"
        android:text="@string/kompresi"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textStyle="bold" />

    <RadioGroup
        android:id="@+id/radio_kompresi"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/viewAlgoKompresi"
        android:background="#ffffff"
        android:orientation="vertical" >

```

©UKDW

```
56            <RadioButton
57                android:id="@+id/rbtnTanpaKompresi"
58                android:layout_width="wrap_content"
59                android:layout_height="wrap_content"
60                android:checked="true"
61                android:text="@string/tanpa_kompresi" />
62
63            <RadioButton
64                android:id="@+id/rbtnHuffman"
65                android:layout_width="wrap_content"
66                android:layout_height="wrap_content"
67                android:text="@string/Huffman" />
68
69            <RadioButton
70                android:id="@+id/rbtnLZW"
71                android:layout_width="wrap_content"
72                android:layout_height="wrap_content"
73                android:text="@string/LZW" />
74
75            <RadioButton
76                android:id="@+id/rbtnBWT"
77                android:layout_width="wrap_content"
78                android:layout_height="wrap_content"
79                android:text="@string/BWT" />
80        </RadioGroup>
81
82        <Button
83            android:id="@+id/btnSembunyikanPesan"
84            android:layout_width="match_parent"
85            android:layout_height="wrap_content"
86            android:layout_marginTop="5dp"
87            android:background="@drawable/button_selector"
88            android:textColor="#000000"
89            android:layout_below="@+id/radio_kompresi"
90            android:text="@string/sembunyikan_pesan" />
91
92    </RelativeLayout>
```

©UKDW

activity_stego_sms.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:padding="0dp"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".StegoSMS" >

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingBottom="5dp"
        android:background="#ffde00" >

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnBuatSMS"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="#ffde00"
            android:onClick="buatSMS"
            android:text="@string/buat_sms"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSMasuk"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="masukSMS"
            android:text="@string/title_activity_masuk_sms"
            android:textColor="#ffffff"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSTerkirim"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="terkirimSMS"
            android:text="@string/title_activity_terkirim_sms"
            android:textColor="#ffffff"
            android:padding="5dp"
            android:textSize="@dimen/menu"
```

©UKDW

```
56                android:textStyle="bold" />
57
58        </LinearLayout>
59
60        <TextView
61            android:id="@+id/textViewPhone"
62            android:layout_width="match_parent"
63            android:layout_height="wrap_content"
64            android:background="#ffffff"
65            android:ems="10"
66            android:hint="@string/nomor_tujuan"
67            android:padding="5dp"
68            android:textAppearance="?android:attr/textAppearanceLarge" />
69
70        <TextView
71            android:id="@+id/textViewSMS"
72            android:layout_width="match_parent"
73            android:layout_height="0dp"
74            android:layout_marginTop="5dp"
75            android:layout_weight="1"
76            android:background="#ffffff"
77            android:ems="10"
78            android:gravity="center_vertical|top"
79            android:hint="@string/isi_sms"
80            android:padding="5dp" />
81
82        <Button
83           android:id="@+id/btnBatalPesanRahasia"
84           android:layout_width="match_parent"
85           android:layout_height="wrap_content"
86           android:layout_marginLeft="5dp"
87           android:layout_marginRight="5dp"
88           android:layout_marginTop="5dp"
89           android:background="@drawable/button_selector"
90           android:textColor="#000000"
91           android:textSize="@dimen/button"
92           android:text="@string/batal_pesan"/>
93
94        <Button
95           android:id="@+id/btnKirimSMS"
96           android:layout_width="match_parent"
97           android:layout_height="wrap_content"
98           android:layout_marginBottom="5dp"
99           android:layout_marginLeft="5dp"
100          android:layout_marginRight="5dp"
101          android:layout_marginTop="2dp"
102          android:background="@drawable/button_selector"
103          android:textColor="#000000"
104          android:textSize="@dimen/button"
105          android:text="@string/kirim_sms"/>
106
107   </LinearLayout>
```

©UKDW

activity_masuk_sms.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="0dp"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MasukSMS" >

    <LinearLayout
        android:id="@+id/linearLayout1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingBottom="5dp"
        android:background="#ffde00">

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnBuatSMS"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="buatSMS"
            android:text="@string/buat_sms"
            android:textColor="#ffffff"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSMasuk"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="#ffde00"
            android:onClick="masukSMS"
            android:text="@string/title_activity_masuk_sms"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSTerkirim"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="terkirimSMS"
            android:text="@string/title_activity_terkirim_sms"
            android:textColor="#ffffff"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />
```

©UKDW

```
56
57          </LinearLayout>
58
59          <ListView
60              android:id="@+id/listMasukSMS"
61              android:layout_width="match_parent"
62              android:layout_height="wrap_content"
63              android:layout_alignLeft="@+id/linearLayout1"
64              android:layout_below="@+id/linearLayout1"
65              android:clickable="true"
66              android:padding="5dp" >
67
68          </ListView>
69
70      </RelativeLayout>
```

©UKDW

activity_terkirim_sms.xml

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="0dp"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".TerkirimSMS" >

    <LinearLayout
        android:id="@+id/linearLayout1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:paddingBottom="5dp"
        android:background="#ffde00">

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnBuatSMS"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="buatSMS"
            android:text="@string/buat_sms"
            android:textColor="#ffffff"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSMasuk"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/button_selector"
            android:onClick="masukSMS"
            android:text="@string/title_activity_masuk_sms"
            android:textColor="#ffffff"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />

        <Button style="?android:attr/borderlessButtonStyle"
            android:id="@+id/btnSMSTerkirim"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="#ffde00"
            android:onClick="terkirimSMS"
            android:text="@string/title_activity_terkirim_sms"
            android:padding="5dp"
            android:textSize="@dimen/menu"
            android:textStyle="bold" />
```

©UKDW

```
56
57          </LinearLayout>
58
59          <ListView
60              android:id="@+id/listTerkirimSMS"
61              android:layout_width="match_parent"
62              android:layout_height="wrap_content"
63              android:layout_alignLeft="@+id/linearLayout1"
64              android:layout_below="@+id/linearLayout1"
65              android:clickable="true"
66              android:padding="5dp" >
67
68          </ListView>
69
70      </RelativeLayout>
```

©UKDW

activity_lihat_sms.xml

```
<?xml version="1.0" encoding="utf-8"?><LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".LihatSMS" >



    <TextView
        android:id="@+id/textViewPhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#ffde00"
        android:ems="10"
        android:padding="10dp"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textViewSMS"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="#ffffff"
        android:ems="10"
        android:gravity="center_vertical|top"
        android:padding="10dp"
        android:textAllCaps="false"
        android:textAppearance="?android:attr/textAppearanceMedium" />

    <Button
        android:id="@+id/btnLihatPesanRahasia"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:background="@drawable/button_selector"
        android:textColor="#000000"
        android:textSize="@dimen/button"
        android:text="@string/lihat_pesan_rahasia"/>

    <Button
        android:id="@+id/btnKembali"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="5dp"
        android:layout_marginTop="2dp"
        android:background="@drawable/button_selector"
        android:textColor="#000000"
        android:textSize="@dimen/button"
```

©UKDW

```
56            android:text="@string/kembali"
57            android:onClick="kembali"/>
58
59    </LinearLayout>
```

©UKDW

activity_lihat_rahasia.xml

```
<?xml version="1.0" encoding="utf-8"?><LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".LihatRahasia" >

    <TextView
        android:id="@+id/textViewPhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#ffde00"
        android:ems="10"
        android:padding="5dp"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/textViewSMS"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="#ffffff"
        android:ems="10"
        android:gravity="center_vertical|top"
        android:padding="5dp"
        android:textAllCaps="false"
        android:textAppearance="?android:attr/textAppearanceMedium" />

    <TextView
        android:id="@+id/labelPesanRahasia"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="5dp"
        android:background="#ffde00"
        android:padding="5dp"
        android:text="@string/isi_pesan_rahasia"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textStyle="bold" />

    <TextView
                android:id="@+id/textViewPesanRahasia"
                android:layout_width="match_parent"
                android:layout_height="0px"
                android:layout_weight="1"
                android:background="#ffffff"
                android:padding="5dp"
                android:textAppearance="?android:attr/textAppearanceMedium" />

    <Button
```

©UKDW

```
56            android:id="@+id/btnDetail"
57            android:layout_width="match_parent"
58            android:layout_height="wrap_content"
59            android:layout_marginTop="5dp"
60            android:background="@drawable/button_selector"
61            android:textColor="#000000"
62            android:textSize="@dimen/button"
63            android:text="@string/detail"
64            android:onClick="detail"/>
65
66        <Button
67            android:id="@+id/btnKembali"
68            android:layout_width="match_parent"
69            android:layout_height="wrap_content"
70            android:layout_marginBottom="5dp"
71            android:layout_marginTop="2dp"
72            android:background="@drawable/button_selector"
73            android:textColor="#000000"
74            android:textSize="@dimen/button"
75            android:text="@string/kembali"
76            android:onClick="kembali"/>
77
78    </LinearLayout>
```

©UKDW

login_form.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:layout_width="match_parent"
4      android:layout_height="wrap_content"
5      android:padding="5dip"
6      android:orientation="vertical" >
7
8      <EditText
9          android:id="@+id/password"
10         android:layout_width="fill_parent"
11         android:layout_height="wrap_content"
12         android:ems="10"
13         android:hint= "@string/masukkan_password"
14         android:inputType="textPassword"
15         android:singleLine="true" >
16         <requestFocus />
17     </EditText>
18
19     <Button
20         android:id="@+id/login"
21         android:layout_width="300dp"
22         android:layout_height="40dip"
23         android:layout_marginTop="20dp"
24         android:background="@drawable/button_selector"
25         android:text="@string/lihat_pesan_rahasia"
26         android:textSize="@dimen/button" />
27 </LinearLayout>
```

©UKDW

ubah_pass.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="5dip"
    android:orientation="vertical" >

    <TextView
        android:id="@+id/label_passLama"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="@string/pass_lama"
        android:textAppearance="?android:attr/textAppearanceMedium" />

    <EditText
        android:id="@+id/passwordLama"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPassword"
        android:singleLine="true" >
    </EditText>

    <TextView
        android:id="@+id/label_passBaru"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="@string/pass_baru"
        android:textAppearance="?android:attr/textAppearanceMedium" />

    <EditText
        android:id="@+id/passwordBaru"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:ems="10"
        android:inputType="textPassword"
        android:singleLine="true" >
    </EditText>

    <Button
        android:id="@+id/ubahPassword"
        android:layout_width="300dp"
        android:layout_height="40dip"
        android:layout_marginTop="20dp"
        android:background="@drawable/button_selector"
        android:text="@string/ubah_password"
        android:textSize="@dimen/button" />
</LinearLayout>
```

©UKDW

strings.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">SMS Rahasia</string>
    <string name="action_settings">Settings</string>
    <string name="kirim_sms">Kirim SMS</string>
    <string name="buat_sms">Buat SMS</string>
    <string name="title_activity_masuk_sms">SMS Masuk</string>
    <string name="title_activity_terkirim_sms">SMS Terkirim</string>
    <string name="title_activity_pesan_tersembunyi">Pesan Tersembunyi</string>
    <string name="tambah_pesan">Tambah Pesan Rahasia</string>
    <string name="Huffman">Huffman</string>
    <string name="LZW">LZW</string>
    <string name="BWT">BWT</string>
    <string name="tanpa_kompresi">Tanpa Kompresi</string>
    <string name="kompresi">Pilihan Kompresi:</string>
    <string name="tulis_pesan">Tulis pesan rahasia:</string>
    <string name="nomor_tujuan">Nomor Tujuan</string>
    <string name="isi_sms">Isi SMS</string>
    <string name="sembunyikan_pesan">Sembunyikan Pesan</string>
    <string name="isi_pesan">Isi pesan rahasia</string>
    <string name="lihat_pesan_rahasia">Lihat Pesan Rahasia</string>
    <string name="title_activity_stego_sms">StegoSMS</string>
    <string name="title_activity_lihat_sms">LihatSMS</string>
    <string name="isi_pesan_rahasia">Isi Pesan Rahasia:</string>
    <string name="title_activity_main">MainActivity</string>
    <string name="kembali">Kembali</string>
    <string name="detail">Detail Pesan</string>
    <string name="title_activity_lihat_rahasia">Lihat Rahasia</string>
    <string name="batal_pesan">Batalkan Pesan Rahasia</string>
    <string name="masukkan_password">Masukkan Password</string>
    <string name="ubah_pass">Ubah Password</string>
    <string name="about">Tentang Aplikasi</string>
    <string name="pass_lama">Masukkan password lama:</string>
    <string name="pass_baru">Masukkan password baru:</string>
    <string name="ubah_password">Ubah password:</string>
</resources>
```

©UKDW

AndroidManifest.xml

```
1   <?xml version="1.0" encoding="utf-8"?>
2   <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3       package="com.ukdw.steganosms"
4       android:versionCode="1"
5       android:versionName="1.0" >
6
7       <uses-sdk
8           android:minSdkVersion="14"
9           android:targetSdkVersion="17" />
10
11      <uses-permission android:name="android.permission.SEND_SMS" />
12      <uses-permission android:name="android.permission.RECEIVE_SMS" />
13      <uses-permission android:name="android.permission.READ_SMS" />
14      <uses-permission android:name="android.permission.WRITE_SMS" />
15      <uses-permission android:name="android.permission.READ_LOGS" />
16
17      <application
18          android:allowBackup="true"
19          android:icon="@drawable/ic_launcher"
20          android:label="@string/app_name"
21          android:theme="@style/AppTheme" >
22          <activity
23              android:name="com.ukdw.steganosms.MainSMS"
24              android:label="@string/app_name"
25              android:screenOrientation="portrait"
26              >
27              <intent-filter>
28                  <action android:name="android.intent.action.MAIN" />
29
30                  <category android:name="android.intent.category.LAUNCHER" />
31              </intent-filter>
32          </activity>
33          <activity
34              android:name="com.ukdw.steganosms.MasukSMS"
35              android:label="@string/app_name"
36              android:screenOrientation="portrait">
37          </activity>
38          <activity
39              android:name="com.ukdw.steganosms.TerkirimSMS"
40              android:label="@string/app_name"
41              android:screenOrientation="portrait">
42          </activity>
43          <activity
44              android:name="com.ukdw.steganosms.PesanTersembunyi"
45              android:label="@string/app_name"
46              android:screenOrientation="portrait">
47          </activity>
48          <activity
49              android:name="com.ukdw.steganosms.StegoSMS"
50              android:label="@string/app_name"
51              android:screenOrientation="portrait">
52          </activity>
53          <activity
54              android:name="com.ukdw.steganosms.LihatSMS"
55              android:label="@string/app_name"
```

©UKDW

```
56              android:screenOrientation="portrait">
57          </activity>
58          <activity
59              android:name="com.ukdw.steganosms.LihatRahasia"
60              android:label="@string/app_name"
61              android:screenOrientation="portrait">
62          </activity>
63      </application>
64
65  </manifest
```

©UKDW

©UKDW

LAMPIRAN D

- Surat-Surat Kelengkapan

Universitas Kristen Duta Wacana
Fakultas Teknologi Informasi Program Studi Teknik Informatika
Jl. Dr. Wahidin Sudirahusada 5-25 Yogyakarta 55224
Telp.: (0274)563929 ext 321 Faks.: (0274)513235

FORMULIR CATATAN UJIAN SKRIPSI

(Diisi oleh Ketua Tim Penguji)

Pada hari ini : **Senin, 9 Juni 2014**
Telah dilakukan Ujian Skripsi untuk mahasiswa tersebut dibawah ini:

Nama Mahasiswa	:	**YUDHI PUTRA TAMA**
No. Induk Mahasiswa	:	**22084583**
Judul Skripsi	:	PERBANDINGAN ALGORITMA HUFFMAN, LZW, DAN BWT DALAM KOMPRESI PESAN TERSEMBUNYI PADA SMS BERBASIS ANDROID
Dosen Pembimbing I	:	**Antonius Rachmat C., SKom.,M.Cs**
Dosen Pembimbing II	:	**Junius Karel, M.T.**
Keterangan (coret yang tidak terpilih)	:	**LULUS** / ~~TIDAK LULUS~~

Beberapa perubahan/catatan yang harus dilakukan oleh mahasiswa tersebut diatas terkait dengan skripsi yang dikerjakannya:

NO.	CATATAN PERBAIKAN
1	Lihat catatan jurnal.
2	Perubahan menu dari ~~kalimat~~ bullet ke number.
3	
4	
5	
6	
7	
8	
9	
10	

Perubahan diatas harus sudah diselesaikan paling lambat tanggal : **Rabu, 9 Juli 2014**

Yogyakarta, 9 Juni 2014
Ketua Tim Penguji

JUNIUS KAREL, S.Si, M.T.

Catatan:
* 1 (satu) lembar untuk mahasiswa
* 1 (satu) lembar untuk arsip

©UKDW

Universitas Kristen Duta Wacana
Fakultas Teknologi Informasi Program Studi Teknik Informatika
Jl. Dr. Wahidin Sudirahusada 5-25 Yogyakarta 55224
Telp.: (0274)563929 ext. 321 Faks.: (0274)513235

FORMULIR PERBAIKAN (REVISI) SKRIPSI

Dicetak tanggal: 12-06-2014 11:45:38

Yang bertanda tangan di bawah ini:

Nama	: YUDHI PUTRA TAMA
N I M	: 22084583
Judul Skripsi	: PERBANDINGAN ALGORITMA HUFFMAN, LZW, DAN BWT DALAM KOMPRESI PESAN TERSEMBUNYI PADA SMS BERBASIS ANDROID
Tanggal Pendadaran	: Senin, 9 Juni 2014 pukul 08:00 WIB

Telah melakukan perbaikan tugas akhir dengan lengkap.

Demikian pernyataan kami agar dapat dipergunakan sebagaimana mestinya.

Yogyakarta, Kamis, 12 Juni 2014

Dosen Pembimbing I	Dosen Pembimbing II
Antonius Rachmat C., SKom.,M.Cs	Junius Karel, M.T.

©UKDW