

APLIKASI KOMPRESI DATA TEKS SMS MENGGUNAKAN ALGORITMA HUFFMAN DI BLACKBERRY

Juli Yanto¹⁾ Rachmat Suryadi²⁾

¹⁾Juli yanto

Sekolah Tinggi Teknologi Informa si NIIT I-Tech
Jl. Asem 2 No.22, Cipete – Jakarta Selatan
<http://www.i-tech.ac.id>
juliyanto@i-tech.ac.id

²⁾Rachmat Suryadi

Sekolah Tinggi Teknologi Informasi NIIT I-Tech
Jl. Asem 2 No.22, Cipete – Jakarta Selatan
<http://www.i-tech.ac.id>
black_karen002@yahoo.com

ABSTRAKSI

SMS menjadi salah satu layanan yang banyak diminati dan digunakan oleh user. Aplikasi SMS yang tersedia pada pada ponsel, menetapkan pengiriman teks dengan ukuran 160 karakter untuk setiap satu kali pengiriman. Apabila user mengirimkan lebih dari 160 karakter maka biaya yang dibebankan kepada user terhitung sebagai kelipatannya.

Untuk mengatasi hal-hal tersebut, sebenarnya ponsel dapat dibekali dengan aplikasi yang berfungsi sebagai kompresi teks SMS. Salah satu algoritma yang digunakan untuk mengkompresi teks adalah algoritma Huffman. Dalam aplikasi kompresi SMS ini, setiap pesan yang dikirimkan dikompresi terlebih dahulu dengan cara setiap karakter yang diketikkan akan dirubah menjadi bentuk bit sesuai dengan tabel static Huffman. Setelah diubah menjadi bit, selanjutnya rangkaian bit tersebut akan dikirimkan ke nomor tujuan.

Aplikasi kompresi SMS ini dibangun dengan menggunakan bahasa pemrograman J2ME untuk perangkat Blackberry dan bertujuan untuk menghemat biaya pengiriman SMS. yang cepat.

PENDAHULUAN

Layanan pesan singkat atau *SMS (Short Message Service)* merupakan salah satu jenis layanan teknologi nirkabel yang memungkinkan dikirimkan atau ditransmisikan pesan teks pendek dari dan ke ponsel. Dalam perkembangannya, *SMS* menjadi salah satu layanan yang banyak diminati dan digunakan oleh user.

Aplikasi *SMS* yang tersedia pada pada ponsel, menetapkan pengiriman teks dengan ukuran 160 karakter untuk setiap satu kali pengiriman. Apabila user mengirimkan lebih dari 160 karakter maka biaya yang dibebankan kepada user terhitung sebagai kelipatannya.

Untuk mengatasi hal-hal tersebut, sebenarnya ponsel dapat dibekali dengan aplikasi yang berfungsi sebagai kompresi teks *SMS* menggunakan metode *Huffman*. Artinya pesan yang dikirimkan, dikurangi terlebih dahulu karakternya, sehingga biaya yang dibebankan user tidak terhitung sebagai kelipatannya.

Dengan menggabungkan antara metode kompresi *Huffman* dan bahasa pemrograman java maka dapat dibuat sebuah aplikasi yang dapat mengurangi karakter teks yang dikirimkan pada setiap pengiriman pesan. Hal inilah yang mendasari pembuatan aplikasi kompresi teks *SMS* menggunakan metode *Huffman* pada tugas akhir ini.

Batasan masalah agar dapat memberikan gambaran yang jelas dari masalah yang dikaji dalam Tugas Akhir ini yaitu:

1. Aplikasi yang dibuat dijalankan pada perangkat *Blackberry OS 5* dan *Blackberry OS 6*.
2. Aplikasi hanya bisa mengirim *SMS* ke satu no tujuan.
3. Kedua pengguna (pengirim dan penerima *SMS*) harus sudah terlebih dahulu menginstall dan menggunakan aplikasi ini.
4. Aplikasi tidak bisa mengambil *SMS* dari *inbox* bawaan ponsel untuk di dekomresi.
5. Input berupa nomor telepon tujuan dan isi pesan.
6. Jumlah karakter pesan dibatasi hingga 165 karakter.
7. Aplikasi tidak terintegrasi dengan nomor kontak bawaan ponsel.

TEORI DASAR

Aplikasi

Aplikasi adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Microsoft Word*, *Microsoft Excel*.

Dari pengertian di atas, dapat disimpulkan bahwa aplikasi merupakan *software* yang berfungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan

Kompresi Data

Kompresi data adalah proses yang dapat mengubah sebuah aliran data masukan (sumber atau data asli) ke dalam aliran data lain (keluaran atau data yang dimampatkan) yang memiliki ukuran lebih kecil. Dalam ilmu komputer dan teori informasi, kompresi data atau *source coding* adalah proses meng-encode informasi dengan menggunakan lebih sedikit bit dari suatu sumber yang belum di *encode* melalui penggunaan skema pengkodean yang spesifik.

Tujuan dari kompresi data adalah untuk merepresentasikan suatu data digital dengan sesedikit mungkin bit, tetapi tetap mempertahankan kebutuhan minimum untuk membentuk kembali data aslinya. Data digital ini hanya berupa teks, gambar, suara dan kombinasi dari ketiganya, seperti video.

Contoh kompresi sederhana yang biasa kita lakukan misalnya adalah menyingkat kata-kata yang sering digunakan tapi sudah memiliki konvensi

umum. Untuk membuat suatu data menjadi lebih kecil ukurannya daripada data asli diperlukan tahap-tahapan (algoritma) untuk mengolah data tersebut. Algoritma adalah suatu prosedur komputasi yang didefinisikan secara baik, membutuhkan sebuah atau sekumpulan nilai sebagai *output*. Dalam algoritma kompresi data, tidak ada algoritma yang cocok untuk semua jenis data. Hal ini disebabkan karena data yang akan dimampatkan harus dianalisa terlebih dahulu dan berharap menemukan pola tertentu yang dapat digunakan untuk memperoleh data dalam ukuran yang lebih kecil. Karena itu, muncul banyak algoritma-algoritma kompresi data.

Kompresi data sangat populer sekarang ini karena dua alasan, yaitu:

1. Orang-orang lebih suka mengumpulkan data. Tidak peduli seberapa besar media penyimpanan yang dimilikinya, akan tetapi cepat atau lambat akan terjadi *overflow*.
2. Orang-orang benci menunggu waktu yang lama untuk memindahkan data. Misalnya ketika duduk di depan komputer untuk menunggu halaman website terbuka atau mengunduh sebuah file.

Kriteria algoritma dan aplikasi kompresi data:

1. Kualitas data hasil encoding: ukuran lebih kecil, data tidak rusak untuk kompresi lossy.
2. Kecepatan, ratio dan efisiensi proses kompresi dan dekomresi.
3. Ketepatan proses dekomresi data: data hasil dekomresi tetap sama dengan data sebelum dikompres (kompresi lossless)

Kompresi Lossless

Algoritma kompresi dikatakan lossless atau juga disebut juga reversible jika dimungkinkan untuk membentuk data asli yang tepat sama dari data yang sudah dikompresi. Tidak ada informasi yang hilang selama proses kompresi dan dekompresi. Teknik ini digunakan jika data tersebut sangat penting, jadi tidak di mungkinkan untuk menghilangkan beberapa detail.

Short Message Service (SMS)

Short Message Service disingkat dengan *SMS*, merupakan pesan singkat berupa teks yang dikirim dan diterima antar sesama pengguna telepon, pada awalnya pesan ini digunakan antar telepon genggam, namun dengan berkembangnya teknologi, pesan tersebut bisa dilakukan melalui komputer ataupun telepon rumah.

SMS merupakan sebuah layanan yang banyak diaplikasikan pada sistem komunikasi tanpa kabel, dikembangkan dan distandarisasi oleh suatu badan yang bernama European Telecommunication Standards Institute (ETSI) sebagai bagian dari pengembangan GSM fase 2, yang terdapat pada dokumentasi GSM 03.40 dan GSM 03.38. Fitur SMS ini memungkinkan perangkat Stasiun Digital (Digital Cellular Terminal, seperti ponsel) untuk dapat mengirim pesan dalam bentuk alphanumeric dan menerima pesan-pesan teks dengan panjang sampai dengan 160 karakter melalui jaringan GSM.

Isu SMS pertama kali muncul dibelahan Eropa pada sekitar tahun 1991 bersama sebuah teknologi komunikasi *wireless* yang saat ini cukup banyak penggunaannya, yaitu *Global System for Mobile communication* (GSM). Dipercaya bahwa pesan yang pertama dikirimkan menggunakan SMS dilakukan pada bulan Desember 1992, dikirimkan dari sebuah *Personal Computer* (PC) ke telepon *mobile* (bergerak) dalam jaringan GSM milik Vodafone Inggris. Perkembangannya kemudian merambah ke benua Amerika, dipelopori oleh beberapa operator komunikasi bergerak berbasis digital seperti *BellSouth Mobility*, *PrimeCo*, *Nextel* dan beberapa operator lain.

Layanan SMS merupakan yang bersifat *nonreal time* dimana sebuah short message dapat di submit ke suatu tujuan, tidak peduli apakah tujuan tersebut aktif atau tidak. SMS merupakan sebuah sistem pengiriman data dalam paket yang bersifat *out-of-band* dengan *bandwidth* kecil, dengan karakteristik ini pengiriman suatu *burst* data yang pendek dapat dilakukan dengan efisiensi yang sangat tinggi.

Algoritma

Algoritma adalah cara yang dapat ditempuh oleh komputer dalam mencapai suatu tujuan, terdiri atas langkah-langkah yang terdefinisi dengan baik menerima input, melakukan proses, dan menghasilkan output.

Dengan demikian bila disimpulkan algoritma adalah susunan langkah untuk menyelesaikan suatu masalah yang disusun secara sistematis untuk mencapai suatu tujuan dengan cara menerima input, melakukan proses dan menghasilkan output.

Algoritma Huffman

Huffman adalah himpunan yang berisi sekumpulan kode biner yang diberi nilai atau label. Huffman dibuat oleh seorang mahasiswa yang bernama *David Huffman* pada tahun 1952, merupakan salah satu metode paling lama dan paling terkenal dalam kompresi teks. Kode Huffman pada dasarnya merupakan kode prefik.

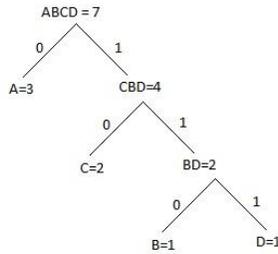
Dalam kode Huffman, panjang blok dari keluaran sumber dipetakan dalam blok biner berdasarkan panjang variabel. Cara seperti ini disebut sebagai *fixed to variable-length coding*. Cara pengkodean seperti ini disebut pemampatan (*compression*) data. Pemampatan data dilakukan dengan mengkodekan setiap karakter di dalam pesan atau di dalam arsip dikodekan dengan kode yang lebih pendek.

Untuk meminimumkan jumlah bit yang dibutuhkan, panjang kode untuk setiap karakter sedapat mungkin diperpendek, terutama untuk karakter yang kekerapan (*frekuensi*) kemunculannya besar. Pemikiran seperti inilah yang mendasari munculnya kode Huffman.

Cara Kerja Huffman

Cara kerja Huffman ini adalah memetakan mulai simbol yang paling banyak terdapat pada sebuah urutan sumber sampai dengan yang jarang muncul menjadi urutan biner. Ini berarti harus terdapat satu cara untuk memecahkan urutan biner yang diterima ke dalam suatu codeword. Seperti yang disebutkan di atas, bahwa ide dari Huffman Coding dalam memilih panjang codeword dari yang paling besar probabilitasnya sampai dengan urutan codeword yang paling kecil probabilitasnya. Apabila dapat memetakan setiap keluaran sumber dari probabilitas $1/p_i$ ke sebuah codeword dengan panjang $1/p_i$ dan pada saat yang bersamaan dapat memastikan bahwa dapat dikodekan secara unik, dapat mencari rata-rata panjang kode $H(x)$. Kode Huffman dapat mendekodekan secara unik dengan $H(x)$ minimum dan optimum pada keunikan dari kode-kode tersebut. Contoh :

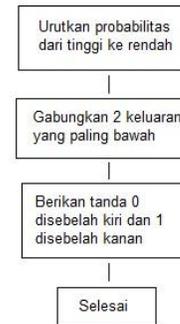
Pohon Huffman untuk pesan "ABACCCA"



Gambar 1 Pohon Huffman Untuk Pesan ABACCDA

Langkah-langkah cara pembentukan kode Huffman adalah:

1. Pilih simbol dengan peluang (probability) paling kecil (simbol B dan D). Kedua simbol tadi dikombinasikan sebagai simpul orang tua dari simbol B dan D sehingga menjadi simbol BD dengan peluang $1/7+1/7=2/7$, yaitu jumlah peluang kedua anaknya. Simbol baru ini diperlakukan sebagai simpul baru dan diperhitungkan dalam mencari simbol selanjutnya yang memiliki peluang paling kecil.
2. Pilih dua simbol berikutnya, termasuk simbol baru yang mempunyai peluang terkecil, dua simbol tersebut adalah C (peluang = $2/7$) dan BD (peluang = $2/7$). Lakukan hal yang sama seperti langkah sebelumnya sehingga dihasilkan simbol baru CBD dengan kekerapan $2/7+2/7=4/7$, prosedur yang sama dilakukan pada dua simbol berikutnya yang mempunyai peluang terkecil, yaitu A (peluang = $3/7$) dan CBD (peluang = $4/7$) sehingga menghasilkan simpul ABCD, yang merupakan akar pohon Huffman dengan peluang $4/7+3/7=7/7$, memberikan nilai 0 dan 1 untuk kedua keluaran.
3. Apabila sebuah keluaran merupakan hasil dari penggabungan 2 keluaran dari langkah sebelumnya, maka berikan tanda 0 dan 1 untuk codewordnya, ulangi sampai keluaran merupakan satu keluaran yang berdiri sendiri.
4. Untuk mendapatkan kode dari huruf yang dimaksud, maka buatlah lintasan tunggal dari akar menuju bobot huruf yang sesuai.



Gambar 2 Alur Kode Huffman

Use Case Diagram (UCD)

Use case diagram (UCD) menjelaskan apa yang akan dilakukan oleh sistem yang akan dibangun dan siapa yang berinteraksi dengan sistem. UCD menjadi dokumen kesepakatan antara Customer, user, dan Developer. User menggunakan dokumen UCD ini untuk memahami sistem dan mengevaluasi bahwa benar yang dilakukan sistem adalah untuk memecahkan masalah yang diajukan user atau sedang dihadapi. Developer menggunakan dokumen UCD ini sebagai rujukan yang benar dalam pengembangan sistem. Use case diagram terdiri dari komponen-komponen berikut:

1. Actor: sebuah entitas manusia atau mesin yang berinteraksi dengan sistem. Untuk melakukan pekerjaan-pekerjaan tertentu. Gambar berikut adalah simbol dari actor:



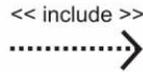
Gambar 3 Simbol Actor

2. Use Case: digunakan untuk memodelkan dan menyatakan unit fungsi/layanan yang disediakan oleh sistem untuk pemakai. Use case dapat dilingkupi dengan batasan sistem yang diberi label nama sistem. Gambar berikut adalah simbol dari Use case.



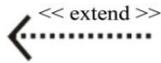
Gambar 4 Simbol Use Case

3. <<include>>
 yaitu kelakuan yang harus terpenuhi agar sebuah event dapat terjadi, dimana pada kondisi ini sebuah use case adalah bagian dari use case lainnya.

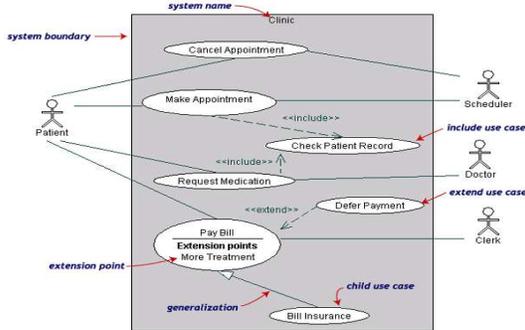


Gambar 5 Simbol Include

4. <<extends>>
 kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.



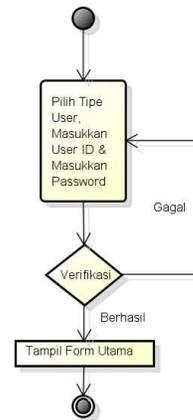
Gambar 6 Simbol Extends



Gambar 7 Contoh Use Case Diagram

Activity Diagram

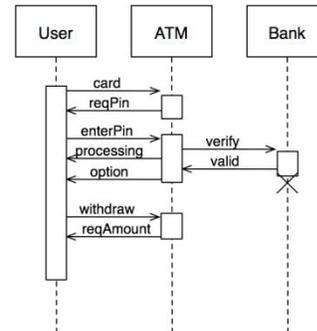
Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Activity diagram merupakan state diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karena itu activity diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Berikut ini adalah tabel simbol-simbol dari diagram activity:



Gambar 8 Contoh Activity Diagram

Sequence Diagram

Sequence diagram menjelaskan secara detail urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case, interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi. Pembuatan sequence diagram merupakan aktifitas yang paling kritical dari proses disain karena artifak inilah yang menjadi pedoman dalam proses pemrograman nantinya dan berisi aliran kendali dari program. Namun, diagram ini kurang mampu menjelaskan detail dari sebuah algoritma, seperti loop. Tabel 2.3 merupakan simbol-simbol yang digunakan pada sequence diagram.

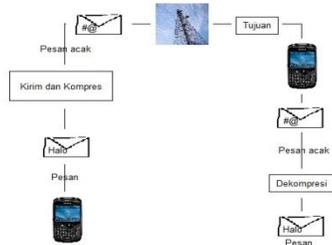


Gambar 9 Contoh Sequence Diagram

Deskripsi Sistem

Berdasarkan kendala yang terdapat pada analisis sistem di atas, maka dibuat lah *Aplikasi Kompresi Data Teks SMS dengan Menggunakan Algoritma Huffman di Blackberry*. Aplikasi ini berfungsi untuk mengkompresi pesan sebelum dikirimkan dan di kompresi menggunakan metode algoritma *Huffman*. Sebelum mengirimkan pesan,

baik pengirim maupun penerima harus sudah mengoperasikannya pada perangkat blackberry. Setelah itu, pengirim pesan dapat mengirimkan pesan yang terkompresi. Pesan yang diterima dalam keadaan terkompres, maka harus ada mekanisme pendekompresian pesan sehingga pesan asli dapat dibaca oleh penerima pesan. Untuk lebih jelasnya dapat dilihat pada ilustrasi pengiriman pesan dibawah ini:

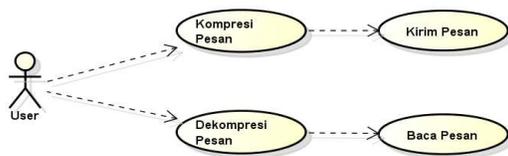


Gambar 10 Proses Pengiriman Pesan Kompresi

Berdasarkan gambar di atas, maka cara kerja sistem dapat di jelaskan sebagai berikut:

1. Pengguna atau pengirim akan mengirimkan pesan kepada penerima melalui telepon selular yang telah ter-*install* dan sedang menjalankan aplikasi.
2. Pengguna memasukkan no tujuan dan isi pesan. Setelah selesai, pengguna dapat mengirimkan pesan dan otomatis pesan dikompresi oleh sistem. Pesan yang dikirimkan adalah pesan acak berdasarkan metode Huffman, Setelah itu notifikasi pesan terkirim muncul.
3. Ketika pesan tersebut masuk pada telepon selular penerima, aplikasi menampilkan kepada pengguna bahwa terdapat pesan masuk dan pesan yang diterima adalah pesan acak.
4. Jika pengguna ingin membaca pesan, maka sebelumnya dilakukan proses dekompresi pada pesan tersebut, sehingga menampilkan pesan asli yang dapat dibaca oleh pengguna.

Use Case Diagram

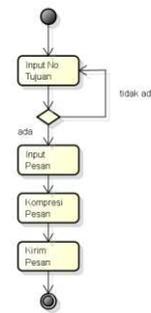


Gambar 11 Use Case Diagram

Activity Diagram



Gambar 12 Activity Diagram Kompresi Pesan



Gambar 13 Activity Diagram Kirim Pesan

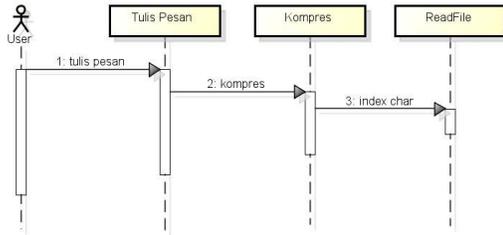


Gambar 14 Activity Diagram Dekompresi Pesan

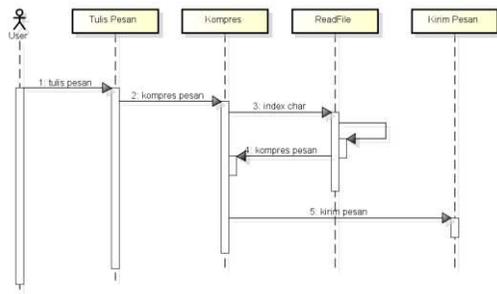


Gambar 15 Activity Diagram Baca Pesan

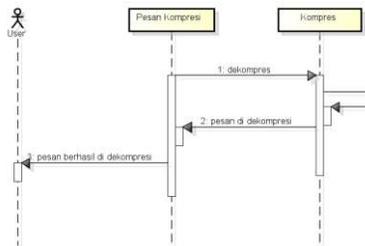
Sequence Diagram



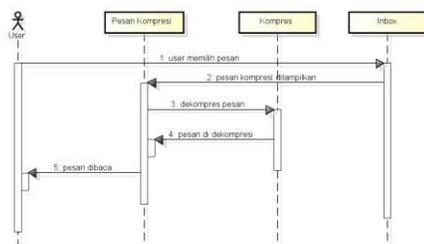
Gambar 16 Sequence Diagram Kompresi Pesan



Gambar 17 Sequence Diagram Kirim Pesan



Gambar 18 Sequence Diagram Dekompresi Pesan



Gambar 19 Sequence Diagram Baca Pesan

Lingkungan Implementasi

Dalam pengembangan *Aplikasi Keamanan dan Kompresi Data SMS Menggunakan Algoritma*

Huffman di Blackberry ini, akan dijelaskan beberapa hal yang berhubungan dengan *hardware*, *software*, serta lingkungan pada saat aplikasi ini dikembangkan.

Perangkat Keras (Hardware)

Perangkat keras yang digunakan untuk menjalankan aplikasi ini adalah sebagai berikut:

1. Laptop Acer Aspire 4740G dengan spesifikasi *hardware* sebagai berikut:
 - a) Processor Intel (R) Core (TM) i5 CPU M430 @ 2.27GHz (4CPUs), ~2.3GHz
 - b) RAM 6GB
 - c) Harddrive 500GB
 - d) nVidia GeForce 310M 512MB
2. Blackberry Gemini Curve 8520
3. Blackberry Gemini Curve 3G 9300
4. Kabel data micro USB

Perangkat Lunak (Software)

1. Sistem Operasi Microsoft Windows 8 Professional 64Bit
2. Sistem Operasi Blackberry versi 5 & versi 6
3. Blackberry Java Plug-in Eclipse SDK

Implementasi Sistem

Aplikasi Kompresi Data Teks SMS Menggunakan Algoritma Huffman di Blackberry ini memiliki 6 halaman, yaitu:

Halaman Intro

Halaman intro merupakan halaman pertama yang muncul ketika user membuka aplikasi.

: SMS Kompresi Huffman :



Gambar 20 Halaman Intro

Pada halaman ini terdapat beberapa komponen yang menyusunnya, berikut merupakan penjelasan dari masing-masing komponen:

- a. *Command Button* Lanjut, untuk menuju ke halaman selanjutnya.
- b. *Command Button* Kembali, untuk keluar dari aplikasi.
- c. *Ticker* animasi "Sekolah Tinggi Teknologi Informasi I-Tech 2013"

Halaman Menu Utama

Halaman menu utama merupakan halaman yang muncul setelah halaman intro. Berikut merupakan tampilan halaman menu utama.



Informasi I-TECH 2013

Gambar 21 Halaman Menu Utama

Pada halaman ini terdapat beberapa komponen yang menyusunnya, berikut merupakan penjelasan dari masing-masing komponen:

- Command Button* Pilih, untuk menuju ke halaman selanjutnya, sesuai dengan pilihan pada *Radio Button*.
- Command Button* Kembali, untuk kembali ke halaman intro.
- Ticker* animasi "Sekolah Tinggi Teknologi Informasi I-Tech 2013".
- Radio Button* Pesan Baru, untuk menentukan halaman yang akan dituju adalah halaman pesan baru.
- Radio Button* Kotak Masuk, untuk menentukan halaman yang akan dituju adalah halaman kotak masuk.
- Radio Button* Kotak Keluar, untuk menentukan halaman yang akan dituju adalah halaman kotak keluar.
- Radio Button* Profil, untuk menentukan halaman yang akan dituju adalah halaman profil.
- Radio Button* Keluar, untuk menentukan halaman yang akan dituju adalah keluar.

Halaman Pesan Baru

Halaman pesan baru merupakan halaman yang muncul setelah dipilih pada halaman menu utama. Berikut merupakan tampilan halaman pesan baru.



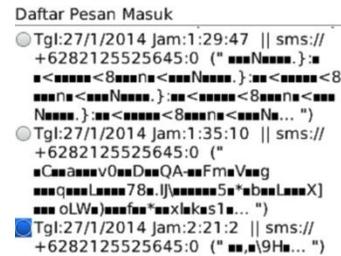
Gambar 22 Halaman Pesan Baru

Pada halaman ini terdapat beberapa komponen yang menyusunnya, berikut merupakan penjelasan dari masing-masing komponen:

- Textfield* dengan *label* Nomor Tujuan, sebagai tempat memasukkan nomor yang akan di tuju dalam pengiriman pesan ini.
- Label* Port 0, untuk memberitahu, bahwa proses pengiriman pesan menggunakan Port 0.
- Textfield* dengan *label* Isi Pesan, sebagai tempat memasukkan isi pesan yang akan dikirimkan.
- Command Button* Kirim Huffman, untuk mengirim pesan sekaligus mengkompresi pesan.
- Command Button* Kembali, untuk kembali ke menu utama.

Halaman Kotak Masuk

Halaman kotak masuk merupakan halaman yang muncul setelah dipilih pada halaman menu utama. Berikut merupakan tampilan halaman kotak masuk.



Gambar 23 Halaman Kotak Masuk

Pada halaman ini terdapat beberapa komponen yang menyusunnya, berikut merupakan penjelasan dari masing-masing komponen:

- Radio Button* pesan masuk, untuk menentukan pesan mana yang akan dibaca.
- Command Button* Detail, untuk melihat pesan secara detail, berdasarkan pilihan pada *Radio Button*.
- Command Button* Kembali, untuk kembali ke menu utama.
- Command Button* Hapus, untuk menghapus pesan berdasarkan pilihan pada *Radio Button*.

Halaman Kotak Keluar

Halaman kotak masuk merupakan halaman yang muncul setelah dipilih pada halaman menu utama. Berikut merupakan tampilan halaman kotak masuk.

Daftar Pesan Keluar

been the industry's standard dummy text ever since the 1500s, when an unkn... ")

● Tgl:27/1/2014 Jam:1:37:3 || 087880510018 (" Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unkn... ")

■ Tgl:27/1/2014 Jam:2:20:59 || 082125525645 (" Huffman SMS... ")

Gambar 24 Halaman Kotak Keluar

Pada halaman ini terdapat beberapa komponen yang menyusunnya, berikut merupakan penjelasan dari masing-masing komponen:

- Radio Button* pesan keluar, untuk menentukan pesan mana yang akan dibaca.
- Command Button* Detail, untuk melihat pesan secara detail, berdasarkan pilihan pada *Radio Button*.
- Command Button* Kembali, untuk kembali ke menu utama.
- Command Button* Hapus, untuk menghapus pesan berdasarkan pilihan pada *Radio Button*.

Halaman Profil

Halaman kotak masuk merupakan halaman yang muncul setelah dipilih pada halaman menu utama. Berikut merupakan tampilan halaman kotak masuk.

:: Profil ::



Nama: Rachmat Suryadi
 NIM: 42077009
 Email: rachmat_suryadi@yahoo.com
 Judul: Aplikasi keamanan dan kompresi data SMS dengan algoritma Huffman pada BlackBerry
 Teknologi Informasi I-TECH 2013

Gambar 25 Halaman Profil

Pada halaman ini terdapat beberapa komponen yang menyusunnya, berikut merupakan penjelasan dari masing-masing komponen:

- Command Button* Kembali, untuk kembali ke menu utama.
- Label* Nama, NIM, Email, Judul.
- Ticker* animasi "Sekolah Tinggi Teknologi Informasi I-Tech 2013".

Pengujian Sistem

Pada sub bab ini, kemudian akan dilakukan tahapan yang bertujuan untuk menguji terhadap ketahanan dan keakuratan dari sistem yang telah

dikembangkan. Pengujian ini menggunakan pengujian *white box* dan pengujian *black box*. Spesifikasi penggunaan perangkat keras dan perangkat lunak yang digunakan sama dengan yang digunakan pada tahap implementasi.

Analisa Hasil Pengujian

Analisa terhadap hasil pengujian tersebut adalah sebagai berikut: Pesan berhasil terkirim ke nomor tujuan dan berhasil mengeluarkan notifikasi pesan terkirim.

- Pesan yang dikirim berupa pesan acak dengan metode Kompresi Huffman dan karakter pesan terkirim tidak sesuai dengan pesan asli (lebih sedikit).
- Pesan yang dikirim, tersimpan pada halaman kotak keluar.
- Pada ponsel penerima, bila ada pesan masuk, akan menampilkan notifikasi yang menandakan bahwa terdapat pesan masuk.
- Pesan acak berhasil di dekomposisi dan sehingga menampilkan pesan asli.
- Pesan berhasil dihapus pada menu kotak masuk dan kotak keluar.
- Karakter maksimal pada kolom isi pesan adalah 165 karakter dan pesan hanya dapat mengirim ke satu nomor tujuan.

Kesimpulan

Setelah melakukan penelitian dan penulisan pada tugas akhir ini, maka dapat disimpulkan bahwa:

- Dalam proses kompresi dan dekomposisi, aplikasi dapat menyampaikan informasi atau data *SMS*, sesuai dengan informasi yang dikirimkan.
- Jika pengirim, mengirimkan pesan yang terkompresi dengan menggunakan aplikasi ini, maka pada penerima yang sudah terinstal aplikasi ini dapat mendekomposisi pesan, sehingga pesan dapat dibaca sesuai dengan pesan aslinya, sedangkan pada ponsel penerima yang tidak menginstal aplikasi ini, maka pesan akan masuk pada inbox penerima berupa pesan acak *Huffman*.
- Karakter pesan terkirim tidak sesuai dengan pesan asli (lebih sedikit), sehingga memungkinkan untuk mengurangi biaya *SMS*.
- Pada aplikasi ini dapat mengirimkan pesan maksimal sebanyak 165 karakter.
- Aplikasi hanya dapat mengirimkan pesan kepada sesama pengguna *Blackberry*.

6. Metode algoritma *Huffman* cukup efektif bila digunakan untuk proses kompresi dan dekompresi data teks *SMS*.

Saran

Terdapat beberapa saran dari hasil penelitian yang diharapkan dapat dikembangkan hal serupa teori-teori yang ada maupun yang belum ada, yaitu:

1. Pada pengembangan selanjutnya, diharapkan pada proses pengiriman sms bisa dilakukan ke banyak nomor tujuan.
2. Di harapkan mampu mengkreasikan tampilan program menjadi *user friendly*, seperti contohnya menambahkan background, dll.
3. Aplikasi dapat beroperasi di ponsel lain, tidak hanya berbasis blackberry.

Disarankan melakukan penambahan algoritma, sehingga keamanan lebih terjamin.

DAFTAR PUSTAKA

- [MUN, 05] dengan *Metodologi Berorientasi Objek*". Bandung: Informatika
- Munawar. 2005. "*Pemodelan Visual dengan UML*". Yogyakarta: Graha Ilmu
- [CAH, 06] Cahyono, Setiyo. 2006. "*Panduan Praktis Pemrograman Database Menggunakan MySQL & Java + CD*". Bandung: Informatika
- [MEN, 06] Mengyi Pu, Ida. 2006. "*Fundamental Data Compression*", Linacre House, Jordan Hill, Oxford
- [SAL, 07] Salomon, David. 2007. "*Data Compression*" Edisi 4th, Computer Science Department, California State, University, Northridge, Springer-Verlag, London
- [PUR, 08] Purnama, Rangsang. 2008. "*Pemrograman J2ME Tingkat Dasar*". Surabaya: Gitamedia Press
- [HER, 04] Hermawan, Julius. 2004. "*Analisa Desain & Pemrograman Berorientasi Objek dengan UML dan Visual Basic.Net*". Yogyakarta: ANDI Yogyakarta
- [ENT, 08] Enterprise, Jubilee. 2008. "*101 Tip dan Trik Blackberry*". Yogyakarta: Elex Media Komputindo
- [SET, 08] Setiadi, Robert. 2008. "*Algoritma itu Mudah*". Jakarta: PT Prima Infosarana Media
- [DHA, 09] Dhanta, Rizky. 2009. "*Kamus Istilah Komputer Grafis &*
- [MUN, 05] Munir, Rinaldi. 2005. "*Diktat Kuliah IF2251 Strategi Algoritmik*". Bandung: Institut Teknologi Bandung
- [NUG, 05] Nugroho, Adi. 2005. "*Analisis dan Perancangan Sistem Informasi*