

BAB IV

HASIL DAN UJI COBA

IV.1. Jalannya Uji Coba

Untuk uji coba program pada *handphone* harus memasukkan file hasil *compiler* yang berekstensi *.jar yang ada di dalam folder proyek *NetBeans* menggunakan *CardReader* atau *Bluetooth*. Jika aplikasi berhasil dijalankan maka pertama akan muncul layar *SplashScreen*, *Login*, dll.

1. Layar *SplashScreen*

Layar *SplashScreen* merupakan tampilan awal yang muncul pertama kali jika program berhasil di jalankan.

2. Layar *Login*

Layar login digunakan untuk masuk ke dalam menu utama, untuk masuk ke dalam menu utama terlebih dahulu harus memasukkan *username*, dan *password* yang benar.

3. Menu Utama

Menu ini akan muncul beberapa saat setelah layar *Login*. Menu ini merupakan semua tampilan Menu program yang meliputi Enkripsi, Dekripsi, Bantuan, Tentang, dan Keluar.

4. Menu Enkripsi

Setelah masuk ke dalam menu ini maka pilih menu enkripsi yang ada di menu utama. Menu ini digunakan untuk memasukkan kata ataupun pesan yang akan di enkripsikan ke dalam algoritma *Base64*.

5. Menu Hasil Enkripsi

Setelah memasukkan pesan maka untuk masuk ke menu ini pengguna hanya perlu menekan tombol enkripsi. Jika sudah menekan tombol enkripsi maka akan masuk ke menu ini. Dan pesan yang tadi dimasukkan sudah terenkripsi ke dalam algoritma base64.

6. Menu Kirim

Setelah pesan terenkripsi tekan tombol kirim, maka akan masuk ke menu kirim pesan. Masukkan nomor tujuan pada *textfield* di menu ini dan kemudian tekan tombol kirim, maka pesan akan terkirim ke nomor yang anda tulis tersebut.

7. Menu Dekripsi

Menu ini digunakan untuk mendekripsikan pesan yang terenkripsi. Jika kembali ke menu utama pilih menu dekripsi. Setelah masuk ke menu dekripsi maka terdapat sebuah *textbox*, *textbox* ini digunakan untuk masukan pesan yang terenkripsi.

8. Menu Hasil Dekripsi

Setelah memasukkan pesan yang terenkripsi pada menu enkripsi maka tekan tombol dekripsi untuk melakukan pendekripsian pesan yang terenkripsi.

9. Menu Bantuan

Menu ini terdapat di menu utama. Untuk membuka menu ini hanya perlu memilih menu bantuan yang terdapat pada menu utama. Menu ini berfungsi untuk mengetahui cara-cara menjalankan aplikasi ini.

10. Menu tentang

Cara masuk ke menu ini sama halnya dengan masuk ke menu bantuan, yaitu dengan cara memilih menu tentang yang terdapat di menu utama. Layar ini berfungsi untuk menampilkan informasi dari versi program dan pembuat program.

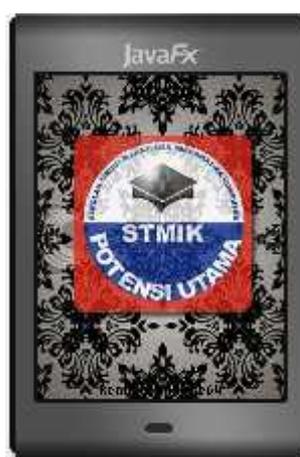
IV.2. Tampilan Layar

Tampilan layar merupakan hasil *output* dari pembuatan aplikasi keamanan pesan *sms* yang di tampilkan untuk mengetahui hasil dari tampilan program.

Berikut ini hasil dari aplikasi keamanan pesan *sms*.

1. Layar *Splash Screen*

Layar *SplashScreen* merupakan tampilan awal yang hanya berdurasi sebentar. Fungsi *Splash screen* ini biasanya digunakan untuk menampilkan informasi yang bersifat sekilas pada user.. Tampilan ini dapat di lihat pada gambar IV.1.



Gambar IV.1. Layar *Splash Screen*

2. Menu Login

Menu ini digunakan untuk masuk ke menu utama, sebelum masuk ke menu utama harus memasukkan *username* dan *password* yang benar agar bisa masuk ke dalam menu utama. Untuk Tampilan dapat di lihat pada gambar IV.2.



Gambar IV.2. Menu Login

3. Menu Utama

Menu ini digunakan untuk memilih menu-menu enkripsi, dekripsi, bantuan, tentang dan keluar. Tampilan dapat di lihat pada gambar IV.3.



Gambar IV.3. Menu Utama

4. Menu Enkripsi

Menu ini digunakan untuk melakukan pengenkripsian ke algoritma *Base64*. Tampilan dapat di lihat pada gambar IV.4.



Gambar IV.4. Menu Enkripsi

5. Menu Hasil Enkripsi

Menu hasil enkripsi ini digunakan untuk menampilkan hasil dari pengenkripsian algoritma *Base64*. Tampilan dapat di lihat pada gambar IV.5.



Gambar IV.5. Menu Hasil Dekripsi

6. Menu Kirim

Menu ini digunakan untuk melakukan pengiriman pesan yang sudah terenkripsi. Tampilan dapat di lihat pada gambar IV.6.



Gambar IV.6. Menu Kirim

7. Menu Dekripsi

Menu ini digunakan untuk melakukan proses pendekripsian ke algoritma *Base64*. Tampilan dapat di lihat pada gambar IV.7.



Gambar IV.7. Menu Dekripsi

8. Menu Hasil Dekripsi

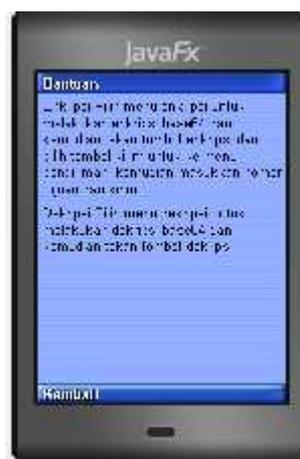
Menu hasil dekripsi ini digunakan untuk menampilkan hasil dari proses pendekripsian algoritma *Base64*. Tampilan dapat di lihat pada gambar IV.8.



Gambar IV.8. Menu Hasil Dekripsi

9. Menu Bantuan

Menu ini digunakan untuk memberikan petunjuk dari penggunaan aplikasi keamanan pesan ini. Tampilan dapat di lihat pada gambar IV.9.



Gambar IV.9. Menu Bantuan

10. Menu Tentang

Menu ini digunakan untuk menampilkan informasi pembuat program dan versi dari program. Tampilan dapat di lihat pada gambar IV.10.



Gambar IV.10. Menu Tentang

IV.3. Hardware/ Software yang dibutuhkan

Hardwar/software dalam pembuatan dan perancangan aplikasi ini yang dibutuhkan yaitu sebuah Laptop yang bersistem operasi *Windows 7* dengan spesifikasi *hardware intel core i3 M30 @2.53Ghz*, memori 2Gb, Compiler Java yaitu *JDK 1.6*. Untuk *Editor* penulis menggunakan *Netbean IDE 6.8* dengan *platform Micro Edition SDK 3.0*, Dengan Konfigurasi *device CLCD-1.1* dan profil *MIDP-2.1* serta *Handphone* bertipe *Java* dan kabel data.

IV.4. Analisa Hasil

Analisa hasil merupakan sebuah pembuktian hasil kebenaran dari algoritma *Base64* yang di aplikasikan ke aplikasi keamanan pesan *sms* ini. Berikut langkah-langkah enkripsi dan dekripsi menggunakan algoritma *Base64*:

1. Langkah-langkah enkripsi menggunakan *Base64*

Pada hasil enkripsi di Table IV.1 kata “Hai” diganti menjadi “SGFp”. Pada table *ascii* huruf H, a, i disimpan sebagai 72, 97, 105 atau dengan kata lain 01001000, 01100001, 01101001 pada bilangan berbasis 2. Apabila ketiga *byte* tersebut digabungkan, maka akan dihasilkan 24 *bit buffer* yaitu 010010000011000001101001. Angka tersebut harus dikonversi sehingga berbasis 64, caranya dengan membagi 24 *bit* tersebut dengan 6. Maka dihasilkan 4 bagian dengan masing-masing 6 *bit*. Kemudian masing-masing bagian tersebut dikonversi ke nilai yang ada di *Base64*. Berikut enkripsi *Base64* pada Tabel IV.1.

Tabel IV.1 Pesan yang akan di enkripsi untuk kata “Hai”

Text Content	H								a							i															
Ascii	72								97							105															
Bit Pattern(8bit)	0	1	0	0	1	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	1	0	0	1							
Bit Pattern(6bit)	0	1	0	0	1	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	1	0	0	1							
Index Base64	18								6							5								41							
Base64-Encoding	S								G							F								p							

Jika bilangan biner tidak bisa dibagi menjadi 6 blok maka ditambahkan nilai kosong agar mencukupi blok bilangan biner dan nilai kosong tersebut bernilai *index* “=”. Agar lebih jelas bisa dilihat pada Tabel IV.2.

Tabel IV.2 Pesan yang akan di enkripsi untuk karakter “H”

Text Content	H																																																							
Ascii	72																																																							
Bit Pattern(8bit)	0	1	0	0	1	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-																												
Bit Pattern(6bit)	0	1	0	0	1	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-																													
Index Base64	18								0																-																-															
Base64-Encoding	S								A																=																=															

2. Langkah-langkah dekripsi menggunakan algoritma *Base64*.

Sedangkan contoh penggunaan dari *Base64* dalam melakukan dekripsi karakter terdapat pada tabel IV.2. Pada hasil dekripsi di table IV.2 kata “SGFp”

diganti menjadi “Hai”. Selanjutnya karakter “Hai” disimpan sebagai index *base64* yaitu 18, 6, 5, 41. Kemudian diubah menjadi 6-bit yaitu “010010,000110,000101,101001” lalu 6-bit diubah menjadi 8-bit yaitu “01001000,01100001,01101001”. Maka akan dihasilkan angka desimal 72, 97, 41 dimana hasil dari karakter tersebut adalah kata “Hai” seperti karakter semula bila dilihat pada tabel *ascii*. Agar lebih jelas bisa dilihat pada Tabel IV.3

Tabel IV.3. Hasil dekripsi *Base64* untuk kata “SGFp”

Text Content	S						G				F					p								
Index Base64	18						6				5					41								
Bit Pattern(6bit)	0	1	0	0	1	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	1	0	0	1
Bit Pattern(8bit)	0	1	0	0	1	0	0	0	0	1	1	0	0	0	0	1	0	1	1	0	1	0	0	1
Ascii	72						97				105													
Text Content	H						a				i													

Jika hasil enkripsi berupa “=” maka nilai sama dengan kosong. Untuk penjelasan lebih lanjut dapat dilihat pada Tabel IV.4.

Tabel IV.4 Hasil Dekripsi *Base64* untuk kata “SA==”

Text Content	S						A				=				=									
Index Base64	18						0				-				-									
Bit Pattern(6bit)	0	1	0	0	1	0	0	0	0	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-
Bit Pattern(8bit)	0	1	0	0	1	0	0	0	0	0	0	0	-	-	-	-	-	-	-	-	-	-	-	-
Ascii	72																							
Text Content	H																							

3. Pengujian panjang karakter terhadap waktu eksekusi

Berikut ini adalah hasil pengujian pengaruh panjang karakter terhadap waktu eksekusi enkripsi dan dekripsi pada aplikasi keamanan pesan *sms*. Untuk melakukan penghitungan waktu eksekusi penulis menggunakan rumus “waktu eksekusi=waktu selesai – waktu selesai”. Untuk tabel pengujian bisa dilihat pada Tabel IV.5.

Tabel IV.5. Pengujian waktu eksekusi panjang karakter

No	Panjang Karakter	Waktu Eksekusi
1	1 Karakter	0.03 Detik
2	26 karakter	0.04 Detik
3	52 Karakter	0.04 Detik
4	78 Karakter	0.06 Detik
5	104 Karakter	0.07 Detik
6	130 Karakter	0.07 Detik
7	156 Karakter	0.09 Detik
8	182 Karakter	0.09 Detik
9	208 Karakter	0.10 Detik
10	234 Karakter	0.11 Detik

Setelah di uji hasil pengujian pengaruh waktu eksekusi panjang karakter maka dapat disimpulkan aplikasi keamanan pesan *sms* menggunakan algoritma *Base64* ini sangat cepat dalam melakukan proses pengenkripsian pesan.

IV.5. Kelebihan dan kekurangan

Kelebihan dari program keamanan pesan *sms* ini adalah dapat melakukan *enkripsi* dan *dekripsi* pada pengiriman pesan *sms* sehingga akan lebih aman jika ingin melakukan pengiriman pesan penting kepada seseorang dan agar terhindar dari pihak-pihak yang tidak diinginkan. Sedangkan kekurangan dari program ini yaitu belum menggunakan penyimpanan pesan *sms*, maka dari itu untuk melakukan pengenkripsian pesan harus *copy* pesan secara manual dari *inbox* atau kotak masuk pada *handphone* untuk melakukan pendekripsian pesan.