

BAB IV

HASIL DAN UJI COBA

IV.1. Jalannya Uji Coba

1. Halaman Utama

Halaman utama adalah halaman *validasi user* sebelum *user* tertentu dapat melakukan enkripsi dan dekripsi *file* bahan ajar. Halaman ini bertujuan agar *user* yang tidak diinginkan dapat memakai program ini. Interaksi dengan halaman ini adalah dengan memasukkan *username* dan *password* yang telah ada terlebih dahulu. Setelah *username* dan *password valid* maka *user* dapat menuju halaman berikutnya yaitu halaman utama.

2. Halaman Menu

Halaman utama adalah halaman menu utama untuk dapat melakukan enkripsi dan dekripsi bahan ajar. Pada halaman ini terdapat dua *menu* tombol yang disediakan yaitu tombol enkripsi dan tombol dekripsi. Interaksi yang dilakukan pada halaman ini adalah dengan memilih tombol enkripsi bila ingin melakukan enkripsi *file* bahan ajar (*plain text*) dan memilih tombol dekripsi bila ingin melakukan dekripsi *file* bahan ajar (*chipper text*). Setelah memilih salah satu tombol tadi maka *user* akan menuju halaman enkripsi atau dekripsi sesuai yang ingin dilakukan.

3. Halaman Enkripsi

Pada halaman enkripsi, terdapat *file field* dan daftar *file* yang telah di *upload* untuk di enkripsi atau dekripsi. Ada dua interaksi yang dapat dilakukan pada halaman ini. Pertama pada *file field*, pada tombol ini kita dapat memasukkan *file* bahan ajar (plain text) untuk di *enkripsi*, setelah *user* memilih *file* bahan ajar yang diinginkan maka *user* menekan tombol *upload* untuk memasukkan *file* tersebut ke *database*. Kedua, pada daftar *file user* dapat memilih *file* yang telah di *upload* untuk di *enkripsi*.

4. Halaman Proses Enkripsi

Pada halaman proses enkripsi, terdapat tombol enkripsi dan *back* didalamnya. Interaksi yang dilakukan adalah dengan menekan tombol *enkripsi* untuk melakukan enkripsi dan mengeluarkan *output chipper text* berbentuk dokumen. Dan menekan tombol *back* bila tidak jadi melakukan *enkripsi*.

5. Halaman Dekripsi

Pada halaman dekripsi, terdapat *file field* dan daftar *file* yang telah di *upload* untuk di enkripsi atau dekripsi. Ada dua interaksi yang dapat dilakukan pada halaman ini. Pertama pada *file field*, pada tombol ini kita dapat memasukkan *file* bahan ajar (plain text) untuk di enkripsi, setelah *user* memilih *file* bahan ajar yang diinginkan maka *user* menekan tombol *upload* untuk memasukkan *file* tersebut ke *database*. Kedua, pada daftar *file user* dapat memilih *file* yang telah di *upload* untuk di dekripsi.

6. Halaman Proses Dekripsi

Pada halaman proses dekripsi, terdapat tombol enkripsi dan *back* didalamnya. Interaksi yang dilakukan adalah dengan menekan tombol dekripsi untuk melakukan *enkripsi* dan mengeluarkan *output plain text* berbentuk dokumen. Dan menekan tombol *back* bila tidak jadi melakukan dekripsi.

IV.2. Tampilan Layar

1. Halaman Utama

Pada halaman utama ini terdapat menu login, user diminta memasukan username dan password untuk dapat ke halaman berikutnya. Halaman ini bertujuan untuk validasi user yang tidak diinginkan agar dapat masuk dan memakai program ini, dapat dilihat pada gambar IV.1 berikut ini :



Gambar IV. 1. Tampilan halaman utama

2. Halaman Menu

Pada halaman menu, user terdapat dua tombol enkripsi dan dekripsi. Dua tombol memiliki fungsi yaitu enkripsi untuk menuju halaman enkripsi file bahan ajar sedangkan dekripsi untuk menuju halaman dekripsi file, dapat dilihat pada gambar IV.2 berikut ini :



Gambar IV. 2. Tampilan halaman menu

3. Halaman Enkripsi

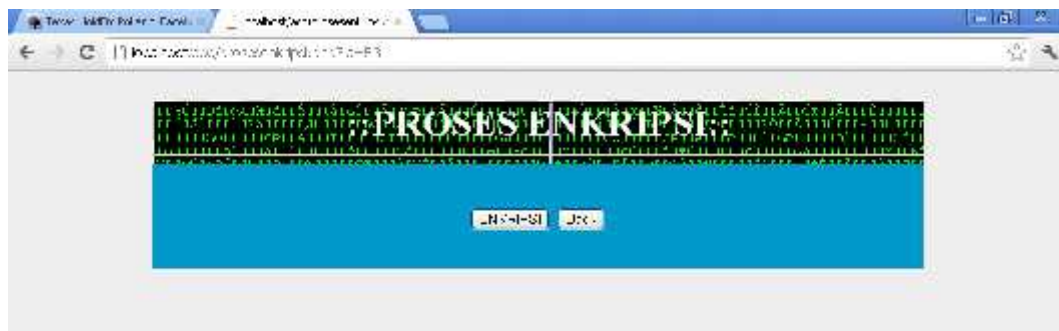
Pada halaman enkripsi ini terdapat *field file* untuk memilih *file* yang akan di *upload* ke *database* dan daftar *file* yang telah di *upload* sebelumnya ke *database*, dapat dilihat pada gambar IV.2 berikut ini :



Gambar IV. 3. Tampilan halaman enkripsi

4. Halaman Proses Enkripsi

Pada halaman ini adalah dimana *user* hanya menekan tombol enkripsi bila ingin enkripsi dan akan mengeluarkan hasil enkripsi ke dokumen berekstensi *.doc*, dapat dilihat pada gambar IV.4 berikut ini :



Gambar IV. 4. Tampilan halaman proses enkripsi

5. Halaman Dekripsi

Pada halaman dekripsi ini terdapat *field file* untuk memilih *file* yang akan di *upload* ke *database* dan daftar *file* yang telah di *upload* sebelumnya ke *database*, dapat dilihat pada gambar IV.5 berikut ini :



Gambar IV. 5. Tampilan halaman dekripsi

6. Halaman Proses Dekripsi

Pada halaman ini adalah dimana *user* hanya menekan tombol enkripsi bila ingin enkripsi dan akan mengeluarkan hasil enkripsi ke dokumen berekstensi *.doc*, dapat dilihat pada gambar IV.6 berikut ini :



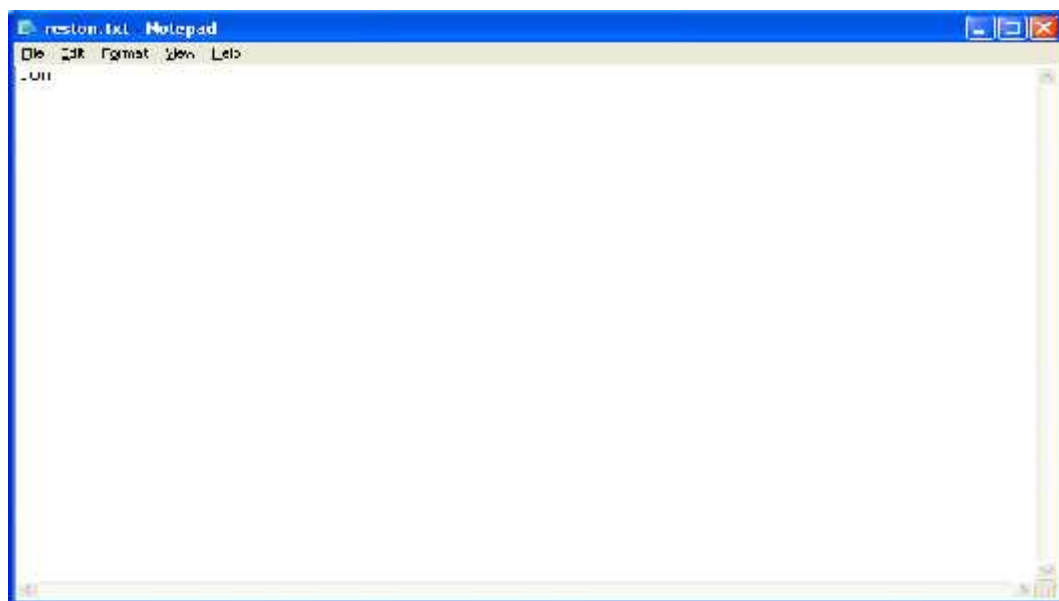
Gambar IV. 6. Tampilan halaman proses dekripsi

IV.3. *Hardware/ Software yang dibutuhkan*

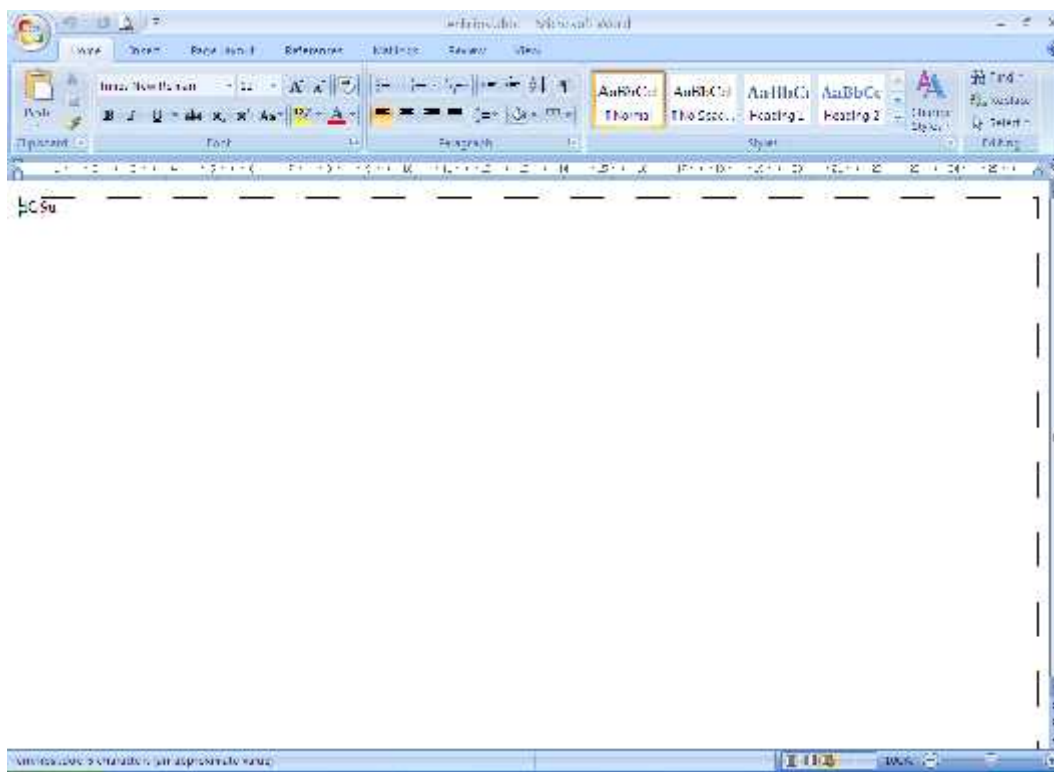
Hardwar/software dalam pembuatan dan perancangan aplikasi ini yang dibutuhkan yaitu sebuah komputer yang bersistem operasi *Windows XP* dengan spesifikasi *hardware* minimum *intel pentium 2.8Ghz*, memori 512Mb, dan *software* yaitu *Web Server* seperti *AppServ 2.5.9* dan *Web Browser* seperti *Mozilla Firefox, Internet Explorer, Opera, dll*, sedangkan *editor* penulis menggunakan *Macromedia Dreamweaver*.

IV.4. *Analisa Hasil*

Program ini dapat meng-enkripsi file bahan ajar yang ingin di amankan oleh pihak-pihak yang tidak diinginkan. Pada gambar IV.7 merupakan gambar sebelum di *enkripsi (plain text)* sedangkan gambar IV. 8 merupakan gambar hasil (*chiper text*) setelah di *enkripsi*.



Gambar IV.7. File sebelum di enkripsi



Gambar IV.8. Hasil setelah di enkripsi

IV.5. Kelebihan dan kekurangan

Kelebihan dari program keamanan bahan ajar ini adalah dapat melakukan *enkripsi* dan *dekripsi* bahan ajar juga dapat menyimpan kedalam *database* sehingga dapat mengamankan *file* tersebut dari pihak-pihak yang tidak diinginkan. Sedangkan kekurangan dari program ini adalah *file* bahan ajar yang seharusnya dokumen harus terlebih dahulu di *copy* isinya ke *notepad* agar bisa di *enkripsi* atau *dekripsi* sehingga membutuhkan dua kali kerja.

IV.5.1 Pembahasan Hasil

Disini penulis membahas proses enkripsi dan dekripsi pada bahan ajar yang akan diamankan dari pihak-pihak yang tidak diinginkan. Berikut ini adalah contoh penggunaan dari *MIME Base 64* dalam melakukan enkripsi karakter.

Pada hasil enkripsi di table IV.1 kata “ton” diganti menjadi “dG9u”. Pada table ASCII huruf t, o, n disimpan sebagai 116, 111, 110 atau dengan kata lain 01110100, 01101111, 01101110 pada bilangan berbasis 2. Apabila ketiga byte tersebut digabungkan, maka akan dihasilkan 24 *bit buffer* yaitu 011101000110111101101110. 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*.

Tabel IV.1. Hasil enkripsi *Base64* untuk kata “ton”

Huruf	T								o								n									
ASCII	116								111								110									
Bit	0	1	1	1	0	1	0	0	0	1	1	0	1	1	1	1	0	1	1	0	1	1	1	0		
Index	29								6						61						46					
Base 64 Encoded	d								G						9						u					

Dari hasil analisa percobaan enkripsi yang telah dilakukan pada beberapa *file text*, maka dapat dibuat tabel pengujian dari hasil enkripsi yang telah dilakukan pada beberapa file tersebut. Tabel pengujian dapat dilihat pada Tabel IV.2 berikut ini :

Tabel IV.2. Tabel Pengujian Enkripsi

No	Nama File	Input	Output	Ukuran Asli (Byte)	Ukuran Terenkripsi (Byte)	Selisih	Persentase (%)
1	Test1.txt	reston	cmVzdG9u	6	8	2	33,33
2	Test2.txt	rendi	cmVuZGk=	5	8	3	60
3	Test3.txt	algoritma	YWxnb3JpdG1h	9	12	3	33,33
4	Test4.txt	potensi	cG90ZW5zaQ==	7	12	5	71,42
5	Test5.txt	utama	dXRhbWE=	5	8	3	60
6	Test6.txt	belawan	YmVsYXdhbg==	7	12	5	71,42
7	Test7.txt	pendidikan	cGVuZGlkaWthbg==	10	16	6	60
8	Test8.txt	kriptografi	a3JpcHRvZ3JhZmk=	11	16	5	45,45
9	Test9.txt	keamanan	a2VhbWFuYW4=	8	12	4	50
10	Test10.txt	komputerisasi	a29tcHV0ZXJpc2FzaQ==	13	20	7	53,84

Sedangkan contoh penggunaan dari *MIME Base 64* dalam melakukan dekripsi karakter terdapat pada tabel IV.2. Pada hasil dekripsi di table IV.2 kata “dG9u” diganti menjadi “ton”. Selanjutnya karakter “ton” disimpan sebagai index *base64* yaitu 29, 6, 61, 46. Kemudian diubah menjadi 6-bit yaitu “011101,000110,111101,101110” lalu 6-bit diubah menjadi 8-bit yaitu “01110100,01101111,01101110”. Maka akan dihasilkan angka desimal 116, 111, 110 dimana hasil dari karakter tersebut adalah kata “Ton” seperti karakter semula bila dilihat pada tabel ASCII.

Tabel IV.3. Hasil dekripsi Base64 untuk kata “dG9u”

Base64	d						G						9						u									
Encoded																												
Index	29						6						61						49									
Bit	0	1	1	1	0	1	0	0	0	0	1	1	0	1	1	1	1	0	1	1	0	1	1	0	1	1	1	0
ASCII	116												111						110									
Huruf	T						o						n															

Dari hasil analisa percobaan dekripsi yang telah dilakukan pada beberapa *file text*, maka dapat dibuat tabel pengujian dari hasil dekripsi yang telah dilakukan pada beberapa file tersebut. Tabel pengujian dapat dilihat pada Tabel IV.4 berikut ini :

Tabel IV.4. Tabel Pengujian Dekripsi

No	Nama File	Input	Output	Ukuran Asli (Byte)	Ukuran Terenkripsi (Byte)	Selisih	Persentase (%)
1	Test1.txt	cmVzdG9u	reston	8	6	2	25
2	Test2.txt	cmVuZGk=	rendi	8	5	3	37,5
3	Test3.txt	YWxnb3JpdG1h	algoritma	12	9	3	25
4	Test4.txt	cG90ZW5zaQ==	potensi	12	7	5	41,66
5	Test5.txt	dXRhbWE=	utama	8	5	3	37,5
6	Test6.txt	YmVsYXdhbg==	belawan	12	7	5	41,66
7	Test7.txt	cGVuZGlkaWthbg==	pendidikan	16	10	6	37,55
8	Test8.txt	a3JpcHRvZ3JhZmk=	kriptografi	16	11	5	31,25
9	Test9.txt	a2VhbWFuYW4=	keamanan	12	8	4	33,33
10	Test10.txt	a29tcHV0ZXJpc2FzaQ==	komputerisasi	20	13	7	35