IMPLEMENTASI ALGORITMA RC4 PADA APLIKASI *SMART CARD* LAYANAN TRANSAKSI PEMINJAMAN BUKU PERPUSTAKAAN

SKRIPSI

oleh:
DEWI CHUMAIROH
NIM. 10650104

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2014

HALAMAN PENGAJUAN

IMPLEMENTASI ALGORITMA RC4 PADA APLIKASI *SMART CARD* LAYANAN TRANSAKSI PEMINJAMAN BUKU PERPUSTAKAAN

SKRIPSI

Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

oleh :
DEWI CHUMAIROH
NIM. 10650104 / S-1

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2014

HALAMAN PERSETUJUAN

IMPLEMENTASI ALGORITMA RC4 PADA APLIKASI *SMART CARD* LAYANAN TRANSAKSI PEMINJAMAN BUKU PERPUSTAKAAN

SKRIPSI

Oleh :

Nama : Dewi Chumairoh
NIM : 10650104
Jurusan : Teknik Informatika
Fakultas : Sains Dan Teknologi

Telah Diperiksa dan Disetujui untuk Diuji :
Tanggal : 13 November 2014

Dosen Pembimbing I	Dosen Pembimbing II
Dr. Muhammad Faisal, M.T.	**Yunifa Miftachul Arif, M.T.**
NIP.19740510 200501 1 007	**NIP. 19830616 201101 1 004**

Mengetahui,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

HALAMAN PENGESAHAN

IMPLEMENTASI ALGORITMA RC4 PADA APLIKASI *SMART CARD* LAYANAN TRANSAKSI PEMINJAMAN BUKU PERPUSTAKAAN

SKRIPSI

Oleh :

DEWI CHUMAIROH
NIM. 10650104

Telah Dipertahankan Di Depan Dewan Penguji Skripsi
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S. Kom)

Tanggal 25 November 2014

Susunan Dewan Penguji :			**Tanda Tangan**
1. Penguji Utama	:	**Totok Chamidy, M.Kom** **NIP. 19691222 200604 1 001**	**( )**
2. Ketua Penguji	:	**Fresy Nugroho, M.T** **NIP. 19710722 201101 1 001**	**( )**
3. Sekretaris	:	**Dr. M. Faisal, M.T** **NIP. 19740510 200501 1 007**	**( )**
4. Anggota Penguji	:	**Yunifa Miftachul Arif, M.T** **NIP. 19830616 201101 1 004**	**( )**

Mengetahui,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

HALAMAN PERNYATAAN
ORISINALITAS PENELITIAN

Saya yang bertandatangan di bawah ini:

Nama	: Dewi Chumairoh
NIM	: 10650104
Fakultas/Jurusan	: Sains Dan Teknologi / Teknik Informatika
Judul Penelitian	:*Implementasi Algoritma RC4 pada Aplikasi Smart Card Layanan Transaksi Peminjaman Buku Perpustakaan*

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 13 November 2014
Yang membuat pernyataan,

Dewi Chumairoh
NIM. 10650104

MOTTO

"Always be your self and never be anyone else even if they look better than you."

Selalu jadi diri sendiri dan jangan pernah menjadi orang lain meskipun mereka tampak lebih baik dari Anda.

"Do whatever you like, be consistent, and success will come naturally."

Lakukan apapun yang kamu sukai, jadilah konsisten, dan sukses akan datang dengan sendirinya.

PERSEMBAHAN

Alhamdulillah...
Segala Puji bagi Allah SWT Tuhan semesta alam, kupersembahkan sebuah karya sederhana untuk orang-orang yang paling aku sayangi dan aku banggakan,

Ayah dan Ibu
Subandi dan Siti Chusnah

Atas Keikhlasannya dalam mendo'akanku,
Atas Kerendahan hatinya dalam membimbingku,
Atas ketulusannya mengingatkan ku dalam kebaikan,
Atas Segalanya yang tak mungkin semua ku ungkapkan,
hanya dalam tulisan sederhan

Kakak - kakakku
Luluk Maslucha dan Anna Fauziah
Yang selalu membuatku semangat dan tak pernah lupa untuk mendoakanku
Seluruh keluarga besarku
Nenek, kakak ipar, dan keponakan
Semoga Allah SWT melindungi dan menjaga kalian, Amin...

Dan ucapan terimakasih yang sebanyak-banyaknya kepada :

- Sahabat seperjuangan, suka dan duka bersama Ade Durotun Nisa' dan Anugrah Widiasari, akhirnya kita bisa menyelesaikan karya ini
- Pak Aziz, Taufan, dan Juniardi yang mau berbagi ilmu dan tak lelah mengajari, terima kasih banyak
- Sahabat - sahabatku tersayang Puspita, Aldita, Puspa, dan Vivid yang selalu memberi semangat, dukungan, dan doa
- Teman – teman seperjuangan Sari, Vina, Kiki, Balqis, Zakiya, Aeny, Wati, Riris, Amel, Fuad, Amru, An'im, dan mas Prima, sukses untuk kita semua
- Teman – teman MMM (Novi, Naufal, Afif, Zaenal, Syafei, Gery, Dzikri, Agus, Dian, dan Haris) yang selalu memberi semangat, dukungan, doa, serta menjadi tempat berbagi suka dan duka
- Teman – teman 4GTI serta alm. Muslich yang telah mendahului kita, bersama-sama kita belajar dari awal perkuliahan hingga saat ini, terima kasih telah menjadi sahabat terbaikku
- Teman – teman Infinity (TI 2010), kalian sudah menjadi keluarga bagiku
- Seluruh warga jurusan Teknik Informatika, terima kasih atas semuanya

KATA PENGANTAR

Bismillahirrohmaanirrohim

Segala puji bagi Allah SWT yang telah melimpahkan rahmat serta karuniaNya kepada penulis sehingga bisa menyelesaikan skripsi dengan judul "*Implementasi Algoritma RC4 pada Aplikasi Smart Card Layanan Transaksi Peminjaman Buku Perpustakaan*" dengan baik

Shalawat serta salam semoga tercurah kepada Nabi Besar Muhammad SAW yang telah membimbing umatnya dari gelapnya kekufuran menuju cahaya Islam yang terang benderang.

Penulis menyadari keterbatasan pengetahuan yang penulis miliki, karena itu tanpa keterlibatan dan sumbangsih dari berbagai pihak, sulit bagi penulis untuk menyelesaikan skripsi ini. Maka dari itu dengan segenap kerendahan hati patutlah penulis ucapkan terima kasih kepada:

1. Dr. Muhammad Faisal, M.T, selaku dosen pembimbing I yang telah meluangkan waktu untuk membimbing, memotivasi, mengarahkan dan memberi masukan dalam pengerjaan skripsi ini.
2. Yunifa Miftachul Arif, M.T., selaku dosen pembimbing II, yang selalu memberikan masukan, nasehat serta petunjuk dalam penyusunan laporan skripsi ini.
3. Dr. Cahyo Crysdian, selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang, yang mendukung dan mengarahkan dalam pengerjaan skripsi ini.
4. Segenap Dosen dan Laboran Teknik informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.
5. Prof. Dr. H. Mudjia Rahardjo, M.Si selaku rektor UIN Maulana Malik Ibrahim Malang, yang telah banyak memberikan pengetahuan dan pengalaman yang berharga.
6. Dr. drh. Hj. Bayyinatul Muchtaromah,M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
7. Semua pihak yang tidak mungkin penulis sebutkan satu-persatu, atas segala yang telah diberikan kepada penulis dan dapat menjadi pelajaran.

Sebagai penutup, penulis menyadari dalam skripsi ini masih banyak kekurangan dan jauh dari sempurna. Semoga apa yang menjadi kekurangan bisa disempurnakan oleh peneliti selanjutnya. Apa yang menjadi harapan penulis, semoga karya ini bermanfaat bagi kita semua. Amin.

Malang, 13 November 2014

Penulis

DAFTAR ISI

HALAMAN JUDUL i
HALAMAN PENGAJUAN ii
HALAMAN PERSETUJUAN iii
HALAMAN PENGESAHAN iv
HALAMAN PERNYATAAN v
HALAMAN MOTTO vi
HALAMAN PERSEMBAHAN vii
KATA PENGANTAR viii
DAFTAR ISI ix
DAFTAR GAMBAR xiii
DAFTAR TABEL xv
ABSTRAK xvi
ABSTRACT xvii
الخلاصة xviii

BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Identifikasi Masalah 5
1.3 Batasan Masalah 5
1.4 Tujuan Penelitian 6
1.5 Manfaat Penelitian 6
1.6 Sistematika Penulisan 6

BAB II TINJAUAN PUSTAKA 8
2.1 *Smart Card* 8
2.1.1 Jenis Memori pada *Smart Card* 8

2.1.2 Standar Internasional *Smart Card* 9
2.1.3 Tipe-tipe *Smart Card* 10
2.1.4 *Card Reader* ACR38U 13
2.1.5 Kartu SLE4428 15
2.2 Kriptografi 16
2.2.1 Algoritma kriptografi 17
2.3 Algoritma RC4 20
2.3.1 Inisialisasi State-Array 21
2.3.2 Penghasilan Kunci Enkripsi & Pengenkripsian 22
2.4 Penelitian Terkait 23

BAB III PERANCANGAN DAN IMPLEMENTASI 26

3.1. Deskripsi Sistem 26
3.2. Perancangan Sistem 26
3.2.1 Diagram Konteks 26
3.2.2 DFD Level 1 28
3.2.3 DFD Level 2 28
3.2.3.1 DFD Level 2 Proses Login 28
3.2.3.2 DFD Level 2 Proses Registrasi 29
3.2.3.3 DFD Level 2 Proses Koleksi 29
3.2.3.4 DFD Level 2 Proses Peminjaman 30
3.2.3.5 DFD Level 2 Proses Pengembalian 30
3.2.3.6 DFD Level 2 Proses Denda 31
3.2.3.7 DFD Level 2 Proses Mahasiswa 31
3.3.Perancangan Database 32
3.4. Perancangan Flowchart 33
3.4.1 Flowchart Algoritma RC4 Proses Enkripsi 33
3.4.2 Flowchart Algoritma RC4 Proses Dekripsi 34
3.4.3 Flowchart Aplikasi Smart Card untuk Enkripsi Data 39
3.4.4 Flowchart Aplikasi Smart Card untuk Dekripsi Data 40
3.5. Perancangan Interface 41
3.5.1 Interface Halaman Depan 41

3.5.2 Interface Halaman Login 41
3.5.3 Interface Halaman Admin..... 42
3.5.4 Interface Halaman Input Koleksi 42
3.5.5 Interface Halaman Input Data Mahasiswa..... 43
3.5.6 Interface Halaman Input Peminjaman..... 44
3.5.7 Interface Halaman Input Pengembalian 44
3.5.8 Interface Halaman Laporan Peminjaman 45
3.5.9 Interface Halaman Laporan Pengembalian..... 45
3.5.10 Interface Halaman Data Pengunjung 46

BAB IV IMPLEMENTASI DAN UJI COBA 47
4.1. Implementasi Sistem 47
4.1.1 Interface 47
4.1.1.1 Form Halaman Depan 47
4.1.1.2 Form Halaman Login..... 48
4.1.1.3 Form Halaman Admin 48
4.1.1.4 Form Halaman Input Koleksi 49
4.1.1.5 Form Halaman Input Data Mahasiswa 50
4.1.1.6 Form Halaman Input Data Peminjaman..... 50
4.1.1.7 Form Halaman Input Data Pengembalian..... 51
4.1.1.8 Form Halaman Laporan Peminjaman..... 52
4.1.1.9 Form Halaman Laporan Pengembalian 52
4.1.1.10 Form Halaman Data Pengunjung..... 53
4.2.Implementasi Algoritma..... 53
4.2.1.Tahap Pembentukan Kunci 53
4.2.2.Tahap Proses Enkripsi Data 54
4.2.3.Tahap Proses Dekripsi Data..... 55
4.2.4.Perhitungan Manual Data Inputan..... 55
4.2.5.Hasil enkripsi Data 63
4.2.6.Hasil Dekripsi Data 63
4.3. Pengujian Sistem..... 64
4.3.1 Rencana Pengujian 64

4.3.2 Pengujian 64
4.3.2.1 Pengujian Autentifikasi Data pada Smart Card 65
4.3.2.2 Pengujian Algoritma RC4 66
4.4. Integrasi Kriptografi menurut Kajian dalam Al-Qur'an 70

BAB V PENUTUP 73
5.1. Kesimpulan 73
5.2. Saran 73
DAFTAR PUSTAKA 75

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM PUSAT PERPUSTAKAAN

DAFTAR GAMBAR

Gambar 2.1 Kartu Smart Card 10
Gambar 2.2 Chip pada Smart Card 11
Gambar 2.3 Smart Card Reader 13
Gambar 2.4 Kartu SLE4428 16
Gambar 2.5 Algoritma Simetrik 18
Gambar 2.6 Algoritma Asimetrik 20
Gambar 3.1 Context Diagram 27
Gambar 3.2 DFD Level 1 28
Gambar 3.3 DFD Level 2 Proses Login 28
Gambar 3.4 DFD Level 2 Proses Registrasi 29
Gambar 3.5 DFD Level 2 Proses Koleksi 29
Gambar 3.6 DFD Level 2 Proses Peminjaman 30
Gambar 3.7 DFD Level 2 Proses Pengembalian 30
Gambar 3.8 DFD Level 2 Proses Denda 31
Gambar 3.9 DFD Level 2 Proses Mahasiswa 31
Gambar 3.10 Rancangan Database 32
Gambar 3.11 Flowchart Algoritma RC4 Proses Enkripsi 33
Gambar 3.12 Flowchart Algoritma RC4 Proses Dekripsi 34
Gambar 3.13 Flowchart Aplikasi Smart Card Proses Enkripsi 39
Gambar 3.14 Flowchart Aplikasi Smart Card Proses Dekripsi 40
Gambar 3.15 Interface Halaman Depan 41
Gambar 3.16 Interface Halaman Login 41
Gambar 3.17 Interface Halaman Admin 42
Gambar 3.18 Interface Halaman Input Koleksi Buku 42
Gambar 3.19 Interface Halaman Input Data Mahasiswa 43
Gambar 3.20 Interface Halaman Input Data Peminjaman 44
Gambar 3.21 Interface Halaman Input Data Pengembalian 44
Gambar 3.22 Interface Halaman Laporan Peminjaman 45
Gambar 3.23 Interface Halaman Laporan Pengembalian 45
Gambar 3.24 Interface Halaman Data Pengunjung 46
Gambar 4.1 Form Halaman Depan 47

Gambar 4.2 Form Halaman Login 48
Gambar 4.3 Form Halaman Admin 48
Gambar 4.4 Form Halaman Koleksi Perpustakaan 49
Gambar 4.5 Form Halaman Data Mahasiswa 50
Gambar 4.6 Form Halaman Data Peminjaman 50
Gambar 4.7 Form Halaman Data Pengembalian 51
Gambar 4.8 Form Halaman Laporan Peminjaman 52
Gambar 4.9 Form Halaman Laporan Pengembalian 52
Gambar 4.10 Form Halaman Data Pengunjung 53
Gambar 4.11 Pseudo code Pembentukan Kunci 53
Gambar 4.12 Pseudo code Proses Enkripsi 54
Gambar 4.13 Pseudo code Proses Dekripsi 55
Gambar 4.14 Hasil Enkripsi Data 63
Gambar 4.15 Hasil Dekripsi Data 63

DAFTAR TABEL

Tabel 3.1 Kode ASCII untuk Tiap Karakter 38
Tabel 3.2 Proses XOR Kunci Enkripsi dengan Plainteks 38
Tabel 3.3 Proses XOR Kunci Dekripsi dengan Cipherteks 38
Tabel 4.1 Kode ASCII untuk Tiap Karakter 61
Tabel 4.2 Proses XOR Kunci Enkripsi dengan Plainteks 62
Tabel 4.3 Proses XOR Kunci Dekripsi dengan Cipherteks 62
Tabel 4.4 Rencana Pengujian 64
Tabel 4.5 Hasil Uji Smart Card 65
Tabel 4.6 Hasil Uji Proses Enkripsi & Dekripsi Data 66
Tabel 4.7 Hasil Uji Algoritma RC4 dengan Masukan Angka 66
Tabel 4.8 Hasil Uji Algoritma RC4 dengan Masukan Huruf..... 67
Tabel 4.9 Hasil Uji Algoritma RC4 dengan Masukan Berbagai Karakter..... 69

ABSTRAK

Chumairoh, Dewi. 2014. **Implementasi Algoritma RC4 pada Aplikasi Smart Card Layanan Transaksi Peminjaman Buku Perpustakaan.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
Pembimbing: (I) Dr. Muhammad Faisal, M.T. (II) Yunifa Miftachul Arif, M.T.

Kata Kunci : *Smart Card, Kriptografi, RC4*

Smart card merupakan salah satu teknologi yang saat ini sedang berkembang. Seiring dengan berkembangnya teknologi, permasalahan yang dihadapi adalah ketidakamanan data yang disimpan di dalam kartu biasa. Data tersebut masih bisa dibaca dan dimanipulasi oleh pihak yang tidak berwenang. Dengan menggunakan *smart card*, keuntungan yang dapat diperoleh adalah menyimpan kerahasiaan data pemilik *smart card*, menyediakan informasi data-data yang penting, dan penyimpanan data secara aman Penelitian ini dimaksudkan untuk membuat sebuah aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan. Aplikasi ini dibangun menggunakan algoritma *RC4* yang merupakan salah satu jenis *stream cipher*, algoritma RC4 memiliki kunci yang sama untuk proses enkripsi dan dekripsi.

Pengujian dilakukan dengan memasukkan data berupa NIM, ID pinjam, dan tanggal terakhir kunjungan perpustakaan. Data-data tersebut dilakukan proses enkripsi sehingga berubah bentuk menjadi kode rahasia, kemudian data disimpan dalam *smart card.* Berdasarkan penelitian yang telah dilakukan, diperoleh hasil bahwa algoritma RC4 dapat digunakan dalam pembuatan aplikasi *smart card* layanan transaksi peminjaman buku perpustakaan. Data yang disimpan dalam *smart card* tidak dapat dibuka pada program lain. Kunci hasil pembentukan pada algoritma RC4 memiliki panjang karakter yang sama dengan panjang karakter masukan data. Hasil pengujian algoritma RC4 dengan masukan berupa angka, huruf, dan kombinasi berbagai karakter menghasilkan nilai akurasi rata-rata sebesar 80%.

ABSTRACT

Chumairoh, Dewi. 2014. **Implementation of RC4 Algorithm on Smart Card Applications for Services Lending Transactions of Library Book**. Thesis. Informatics Department of Faculty of Science and Technology. Maulana Malik Ibrahim State Islamic University, Malang.
Adviser : (I) Dr. Muhammad Faisal, M.T. (II) Yunifa Miftachul Arif, M.T.

Keywords : *Smart Card, Cryptography, RC4*

Smart card technology is one that is currently being developed. Along with the development of technology, the problems faced is the insecurity of data stored in the regular card. The data can still be read and manipulated by unauthorized parties. By using smart cards, the advantages that can be obtained is to store data confidentiality owner of smart cards, providing information critical data, and secure data storage. This study is intended to create a smart card application for services lending transactions of library book. This application is built using the RC4 algorithm, which is one type of stream cipher, RC4 algorithm has the same key for encryption and decryption.

Testing is done by entering the data in the form of NIM, ID of loan, and date of last visit a library. These data encryption process is carried out so that transformed into a secret code, then the data stored in the smart card. Based on the research that has been conducted, the results showed that the RC4 algorithm can be used in the manufacture of smart card application service lending transactions of library book. The data stored in the smart card can not be opened in other programs. Key results in the formation of the RC4 algorithm has the same character length with length character data input. Test results RC4 algorithm with input in the form of numbers, letters, and a combination of various characters produces an average accuracy rate of 80%.

الخلاصة

خميرة. دوي.2014 . المقالة. تنفيذ خوارزمية على البطاقة الذكية طلبات القروض خدمات المعاملات مكتبة الكتاب. قسم المعلوماتية في كلية العلوم والتكنولوجي في جامعة الحكومية الإسلامية مولانا مالك إبراهيم مالانج.

المسرف : 1. محمد فيسل. 2 يونفة مفتاح العريف

كلمات البحث: بطاقة الذكية، والترميز, RC4

تكنولوجيا البطاقات الذكية هي تلك التي يجري حاليا تطويرها. جنبا إلى جنب مع تطور التكنولوجيا, والمشاكل التي تواجهها هي انعدام الأمن من البيانات المخزنة في البطاقة العادية. لا تزال البيانات أن تقرأ والتلاعب من قبل أطراف غير مصرح بها. باستخدام البطاقات الذكية, والمزايا التي يمكن الحصول عليها هي لتخزين صاحب سرية البيانات من البطاقات الذكية, وتوفير المعلومات والبيانات الحرجة, وتخزين البيانات بشكل آمن. ويهدف هذه الدراسة إلى إنشاء تطبيق البطاقة RC4 للكتاب من المكتبة خدمات المعاملات الإقراض RC4 ناء هذا التطبيق باستخدام خوارزمية ,وهو نوع واحد من تيار والشفرات, خوارزمية ,لديها نفس المفتاح لتشفير وفك التشفير. ويتم اختبار عن طريق إدخال البيانات في شكل رقم دفتر قيد, القرض, وتاريخ آخر زيارة مكتبة. تتم هذه العملية تشفير البيانات خارج بحيث تتحول إلى رمز سري, ثم البيانات المخزنة في البطاقة الذكية. واستنادا إلى RC4 حوث التي أجريت وأظهرت النتائج أن خوارزمية يمكن استخدامها في صناعة المعاملات الإقراض كتاب من المكتبة خدمة تطبيق البطاقة الذكية. البيانات المخزنة في البطاقة الذ RC4 مكن فتح في برامج أخرى. النتائج الرئيسية في تشكيل خوار RC4 يها طول مساو لطول حرف حرفا إدخال البيانات. نتائج الاختبار خوارزمية مع المدخلات في شكل الأرقام والحروف, ومزيج من مختلف الشخصيات ينتج متوسط معدل دقة 80%.

الخلاصة

خميرة. دوي.2014 . المقالة. تنفيذ خوارزمية على البطاقة الذكية طلبات القروض خدمات المعاملات مكتبة الكتاب. قسم المعلوماتية في كلية العلوم والتكنولوجي في جامعة الحكومية الإسلامية مولانا مالك إبراهيم مالانج.

المسرف : 1. محمد فيسل. 2 يونفة مفتاح العريف

كلمات البحث: بطاقة الذكية، والترميز RC4

تكنولوجيا البطاقات الذكية هي تلك التي يجري حاليا تطويرها. جنبا إلى جنب مع تطور التكنولوجيا, والمشاكل التي تواجهها هي انعدام الأمن من البيانات المخزنة في البطاقة العادية. لا تزال البيانات أن تقرأ والتلاعب من قبل أطراف غير مصرح بها. باستخدام البطاقات الذكية, والمزايا التي يمكن الحصول عليها هي لتخزين صاحب سرية البيانات من البطاقات الذكية, وتوفير المعلومات والبيانات الحرجة, وتخزين البيانات بشكل آمن. ويهدف هذه الدراسة إلى إنشاء تطبيق البطاقة الذكية للكتاب من المكتبة خدمات المعاملات الإقراض.

تم بناء هذا التطبيق باستخدام خوارزمية RC4 .ع واحد من تيار والشفرات, خوارزمية ,لديها ن RC4 فتاح لتشفير وفك التشفير. ويتم اختبار عن طريق إدخال البيانات في شكل رقم دفتر قيد, القرض, وتاريخ آخر زيارة مكتبة. تتم هذه العملية تشفير البيانات خارج بحيث تتحول إلى رمز سري, ثم البيانات المخزنة في البطاقة الذكية. واستنادا إلى البحوث التي أجريت وأظهرت النتائج أن خوارزمية RC4 استخدامها في صناعة المعاملات الإقراض كتاب من المكتبة خدمة تطبيق البطاقة الذكية. البيانات المخزنة في البطاقة الذكية لا يمكن فتح في برامج أخرى. النتائج الرئيسية في تشكيل خوارزمية لديها طو RC4 و لطول حرف حرفا إدخال البيانات. نتائج الاختبار خوارزمية م RC4 لات في شكل الأرقام والحروف, ومزيج من مختلف الشخصيات ينتج متوسط معدل دقة 80%.

ABSTRACT

Chumairoh, Dewi. 2014. **Implementation of RC4 Algorithm on Smart Card Applications for Services Lending Transactions of Library Book**. Thesis. Informatics Department of Faculty of Science and Technology. Maulana Malik Ibrahim State Islamic University, Malang.
Adviser : (I) Dr. Muhammad Faisal, M.T. (II) Yunifa Miftachul Arif, M.T.

Keywords : *Smart Card, Cryptography, RC4*

Smart card technology is one that is currently being developed. Along with the development of technology, the problems faced is the insecurity of data stored in the regular card. The data can still be read and manipulated by unauthorized parties. By using smart cards, the advantages that can be obtained is to store data confidentiality owner of smart cards, providing information critical data, and secure data storage. This study is intended to create a smart card application for services lending transactions of library book. This application is built using the RC4 algorithm, which is one type of stream cipher, RC4 algorithm has the same key for encryption and decryption.

Testing is done by entering the data in the form of NIM, ID of loan, and date of last visit a library. These data encryption process is carried out so that transformed into a secret code, then the data stored in the smart card. Based on the research that has been conducted, the results showed that the RC4 algorithm can be used in the manufacture of smart card application service lending transactions of library book. The data stored in the smart card can not be opened in other programs. Key results in the formation of the RC4 algorithm has the same character length with length character data input. Test results RC4 algorithm with input in the form of numbers, letters, and a combination of various characters produces an average accuracy rate of 80%.

ABSTRAK

Chumairoh, Dewi. 2014. **Implementasi Algoritma RC4 pada Aplikasi Smart Card Layanan Transaksi Peminjaman Buku Perpustakaan.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
Pembimbing : (I) Dr. Muhammad Faisal, M.T. (II) Yunifa Miftachul Arif, M.T.

Kata Kunci : *Smart Card, Kriptografi, RC4*

Smart card merupakan salah satu teknologi yang saat ini sedang berkembang. Seiring dengan berkembangnya teknologi, permasalahan yang dihadapi adalah ketidakamanan data yang disimpan di dalam kartu biasa. Data tersebut masih bisa dibaca dan dimanipulasi oleh pihak yang tidak berwenang. Dengan menggunakan *smart card*, keuntungan yang dapat diperoleh adalah menyimpan kerahasiaan data pemilik *smart card*, menyediakan informasi data-data yang penting, dan penyimpanan data secara aman. Penelitian ini dimaksudkan untuk membuat sebuah aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan. Aplikasi ini dibangun menggunakan algoritma *RC4* yang merupakan salah satu jenis *stream cipher*, algoritma RC4 memiliki kunci yang sama untuk proses enkripsi dan dekripsi.

Pengujian dilakukan dengan memasukkan data berupa NIM, ID pinjam, dan tanggal terakhir kunjungan perpustakaan. Data-data tersebut dilakukan proses enkripsi sehingga berubah bentuk menjadi kode rahasia, kemudian data disimpan dalam *smart card.* Berdasarkan penelitian yang telah dilakukan, diperoleh hasil bahwa algoritma RC4 dapat digunakan dalam pembuatan aplikasi *smart card* layanan transaksi peminjaman buku perpustakaan. Data yang disimpan dalam *smart card* tidak dapat dibuka pada program lain. Kunci hasil pembentukan pada algoritma RC4 memiliki panjang karakter yang sama dengan panjang karakter masukan data. Hasil pengujian algoritma RC4 dengan masukan berupa angka, huruf, dan kombinasi berbagai karakter menghasilkan nilai akurasi rata-rata sebesar 80%.

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Pada era globalisasi sekarang ini, menuntut manusia untuk mengembangkan berbagai macam ilmu pengetahuan dan teknologi untuk membantu meningkatkan kehidupannya. Seiring dengan perkembangan teknologi yang semakin pesat, hampir semua bidang kehidupan menggunakan teknologi informasi untuk mendukung pekerjaan operasional menjadi lebih baik. Pemanfaatan ilmu pengetahuan dan teknologi informasi yang maksimal akan menghasilkan sumber daya manusia potensial yang terdidik dan terpelajar yang dapat membangun bangsa.

Smart card merupakan salah satu teknologi yang saat ini sedang berkembang. *Smart car*d memiliki beberapa keunggulan, antara lain kemudahan pengaksesan data, keamanan penyimpanan data, perlindungan data dari pihak-pihak yang tidak berwenang, serta fleksibilitas untuk dibawa dengan mudah dalam kegiatan sehari-hari, telah mendorong penggunaan teknologi ini diterapkan di sektor akademik untuk menyimpan data. Keuntungan yang dapat diperoleh dengan penggunaan *smart card* ini adalah menyimpan kerahasiaan data pemilik *smart card*, menyediakan informasi data-data yang penting, dan penyimpanan data secara aman (Christine Sariasih, 2009).

Seiring dengan berkembangnya teknologi *smart card*, permasalahan yang dihadapi adalah ketidakamanan data yang disimpan di dalam *smart card*. Data tersebut masih bisa dibaca dan dimanipulasi oleh pihak yang tidak berwenang. Berdasarkan hal tersebut dibutuhkan suatu perangkat yang dapat menjaga data yang ada di dalam *smart card* dengan baik. Data tersebut akan dilakukan proses enkripsi sebelum disimpan di dalam *smart card* (Supriyono, 2009).

Pernyataan ini sesuai dengan ayat *Al-Qur-an* surat *Al-Hijr* ayat 9 yang berbunyi :

إِنَّا نَحْنُ نَزَّلْنَا الذِّكْرَ وَإِنَّا لَهُ لَحَافِظُونَ ﴿٩﴾

"Sesungguhnya Kami-lah yang menurunkan al-Qur'an, dan sesungguhnya Kami benar-benar memeliharanya." (QS. Al-Hijr : 9)

Allah SWT memelihara Al-Qur'an dari upaya setan yang ingin menambahkan kebatilan ke dalamnya dan mengurangi kebenarannya, sehingga Al-Qur'an tetap terpelihara.

Al-Qur'an terpelihara saat diturunkan maupun setelahnya. Saat diturunkan, Allah SWT memeliharanya dari upaya setan yang ingin mencuri-curi beritanya. Adapun setelah diturunkan, Allah SWT menyimpannya di hati Rasulullah SAW, kemudian di hati umatnya. Allah SWT menjaga lafadz-lafadznya dari perubahan, baik penambahan maupun pengurangan. Allah SWT juga menjaga makna-maknanya dari perubahan dan penggantian. Tidak seorang pun yang berusaha memalingkan salah satu makna pada Al-Qur'an, melainkan

Allah SWT pasti mendatangkan orang yang akan menjelaskan kebenaran yang nyata. Ini merupakan salah satu tanda keagungan ayat-ayat Allah SWT dan kenikmatan-Nya terhadap hamba-hamba-Nya yang mukmin. Di antara bentuk pemeliharaan Allah SWT terhadap Al-Qur'an juga adalah Dia memelihara ahlul Qur'an dari musuh-musuh mereka. Allah SWT menyelamatkan mereka dari gangguan musuh (Al-Imam Ibnu Kasir Ad-Dimasyqi, 2000).

Ayat lain yang semakna di antaranya firman Allah Subhanahu wata'ala:

لَّا يَأْتِيهِ ٱلْبَٰطِلُ مِنۢ بَيْنِ يَدَيْهِ وَلَا مِنْ خَلْفِهِۦ ۖ تَنزِيلٌ مِّنْ حَكِيمٍ حَمِيدٍ ﴿٤٢﴾

"Yang tidak datang kepadanya (Al-Qur'an) kebatilan, baik dari depan maupun dari belakangnya." (QS. Fushshilat : 42)

Maksudnya, memeliharanya dari penyimpangan, baik huruf maupun makna, dan penambahan maupun pengurangan. Allah SWT menjadikan Al-Qur'an sebagai suatu keajaiban (mukjizat), guna membedakan apa yang tertera padanya dengan ucapan manusia (Al-Imam Ibnu Kasir Ad-Dimasyqi, 2000).

Oleh sebab itu, penulis akan membuat sebuah aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan. Teknologi *smart card* menawarkan kemudahan dan keamanan penyimpanan data karena adanya mekanisme enkripsi data sebelum data tersebut disimpan di dalam memori, serta

adanya pin (kode rahasia) yang menjaga data tersebut agar tidak dibaca oleh pihak yang tidak berwenang.

Dalam menjaga keamanan data, *smart card* telah menggunakan beberapa metode yang ada pada kriptografi, salah satu metode tersebut adalah *Rivest Code 4* atau biasa dikenal dengan *RC4*. Algoritma *RC4* merupakan salah satu jenis *stream cipher* sehingga *RC4* memproses unit atau input data, pesan atau informasi pada satu saat. Unit atau data pada umumnya sebuah *byte* sehingga dengan cara ini enkripsi atau dekripsi dapat dilaksanakan pada panjang yang variabel. Algoritma ini tidak harus menunggu sejumlah input data, pesan atau informasi tertentu sebelum diproses, atau menambahkan *byte* tambahan untuk mengenkripsi (Yuri Ariyanto, 2009).

Algoritma RC4 termasuk dalam kriptografi kunci simetris, yaitu algoritma kriptografi yang menggunakan kunci enkripsi yang sama dengan kunci dekripsinya. Kelebihan dari kriptografi kunci simetris antara lain proses enkripsi / dekripsi membutuhkan waktu yang singkat, ukuran kunci simetris relatif pendek dan dapat digunakan untuk membangkitkan bilangan acak, kunci simetris dapat disusun untuk menghasilkan *cipher* yang lebih kuat, otentikasi pengirim pesan langsung diketahui dari *ciphertext* yang diterima karena kunci hanya diketahui oleh pengirim dan penerima pesan saja.

Dari pernyataan tersebut, maka penulis akan membangun aplikasi ini yang diimplementasikan pada perpustakaan, dengan harapan dapat memberikan

kemudahan bagi pengunjung perpustakaan dalam melakukan transaksi peminjaman buku sehingga lebih praktis dan waktu yang digunakan lebih efisien. Selain itu, keamanan data pada *smart card* lebih meningkat dibandingkan dengan kartu perpustakaan yang terbuat dari kertas. Sebagaimana sudah dijelaskan di atas, di dalam *smart card* telah dilakukan proses enkripsi dan dekripsi, maka kerahasiaan data lebih terjaga dan aman.

1.2 IDENTIFIKASI MASALAH

Berdasarkan penjelasan pada latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah bagaimana algoritma RC4 diimplementasikan dalam aplikasi *smart card* yang digunakan untuk layanan transaksi peminjaman buku perpustakaan.

1.3 BATASAN MASALAH

- Bahasa pemrograman yang digunakan adalah *Visual Basic*.
- Penggunaan algoritma *Rivest Code 4 (RC4)* adalah algoritma untuk keamanan data.
- *Card Reader* yang digunakan adalah ACR 38U dan kartu SLE 4428.
- Aplikasi ini berbasis desktop yang diimplementasikan pada perpustakaan Universitas Islam Negeri Malang.
- Aplikasi ini ditujukan untuk mahasiswa-mahasiswi Universitas Islam Negeri Malang.

1.4 TUJUAN PENELITIAN

Berdasarkan rumusan masalah di atas, maka tujuan penelitian ini adalah membuat aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan menggunakan algoritma RC4.

1.5 MANFAAT PENELITIAN

Manfaat pembuatan aplikasi *smart card* ini adalah membuat layanan transaksi peminjaman buku perpustakaan menjadi lebih mudah dan cepat bagi pengunjung perpustakaan serta penerapan algoritma RC4 untuk keamanan data pada *smart card*.

1.6 SISTEMATIKA PENULISAN

Penulisan skripsi ini tersusun dalam lima bab dengan sistematika penulisan sebagai berikut :

BAB I Pendahuluan

Pendahuluan, membahas tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penyusunan tugas akhir, metodologi, dan sistematika penyusunan tugas akhir.

BAB II Landasan Teori

Landasan teori berisikan beberapa teori yang mendasari dalam penyusunan tugas akhir ini. Adapun yang dibahas dalam bab ini adalah dasar teori yang berkaitan dengan pembahasan mengenai *smart card*.

BAB III Analisa dan Perancangan

Menganalisa kebutuhan sistem untuk membuat aplikasi meliputi langkah-langkah pembuatan aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan menggunakan *algoritma RC4*.

BAB IV Hasil dan Pembahasan

Menjelaskan tentang pengujian aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan menggunakan *algoritma RC4* yang telah diterapkan.

BAB V Penutup

Berisi kesimpulan dan saran.

BAB II

TINJAUAN PUSTAKA

2.1 SMART CARD

Smartcard adalah kartu plastik yang berukuran sama dengan kartu kredit yang di dalamnya terdapat *chip* silikon yang disebut *microcontroller*. *Chip* merupakan *integrated circuit* yang terdiri dari prosesor dan memori. *Chip*, seperti layaknya CPU (*Central Processing Unit*) di komputer, bertugas melaksanakan perintah dan menyediakan *power* ke *smartcard* (Christine Sariasih, 2009). *Smartcard* merupakan pengembangan dari kartu magnetis, namun berbeda dengan kartu magnetis yang hanya dipakai sebagai tempat penyimpanan data, *Smartcard* mempunyai kemampuan untuk memproses dan menginterpretasikan data, serta menyimpan data tersebut secara aman. Apalagi dengan perkembangan algoritma kriptografi, data yang disimpan akan dienkripsi terlebih dahulu, sehingga tidak mudah dibaca oleh pihak yang tidak berwenang/berhak. Hal ini akan mempersulit pemalsuan *smartcard*. Selain perbedaan dengan adanya *chip*, *smartcard* memiliki kapasitas memori yang lebih besar dari kartu magnetis (Christine Sariasih, 2009).

2.1.1 Jenis Memori Pada *Smartcard*

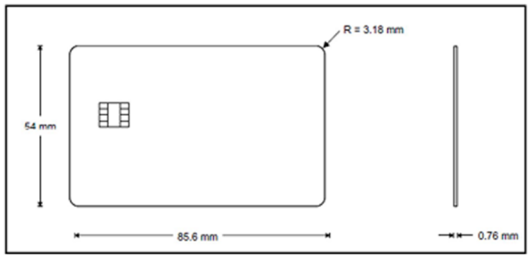
Secara umum ada 3 jenis memori [ISO7816-95] yang digunakan :

1. ROM (*Read Only Memory*), berfungsi untuk menyimpan program utama dan sifatnya permanen.

2. RAM (*Random Access Memory*), berfungsi untuk menyimpan data sementara ketika proses sedang berjalan atau hasil penghitungan selama mengeksekusi perintah. Data yang disimpan di dalamnya akan hilang begitu *smartcard* dicabut (*power* hilang).
3. EEPROM (*Electrically Erasable Programmable Read Only Memory*), berfungsi untuk menyimpan program dan data yang sewaktu-waktu bisa diubah. Seperti halnya *hard disk* pada komputer, jenis memori ini akan tetap menyimpan data meskipun tidak ada *power* (permanen).

2.1.2 Standar Internasional *Smart Card*

Standar ISO/IEC merupakan suatu standar internasional untuk menentukan standarisasi dari *smart card.* Kepanjangan dari ISO adalah *International Organization for Standarization*, sedangkan kepanjangan dari IEC adalah *International Electrotechnical Commission*. ISO/IEC 7816 dan ISO/IEC 7810 adalah sebuah standar untuk *smart card* untuk menentukan hal berikut ini: Bentuk fisik dari kartu, Karakteristik dari sirkuit elektronik, Posisi dan ukuran kartu dari konektor elektrikal kartu, Protokol komunikasi, Ketahanan kartu, Fungsionalitas kartu. Ukuran kartu *smart card* dengan format ID-1 berdasarkan ISO/IEC 7810 adalah 85.60mm x 54mm dan memiliki ketebalan kartu 0.76mm ± 0.08mm serta jari-jari sudut 3.18mm (Gintoro, 2005).



Gambar 2.1. Kartu *smart card* (*sumber: Gintoro, 2005*)

2.1.3 Tipe-tipe *Smartcard*

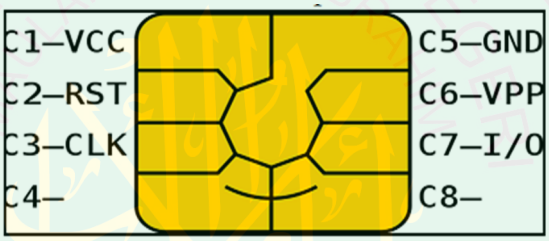
Secara umum ada dua macam tipe dari *smart card* yaitu *memory card* dan *microprocessor card*. *Memory card* hanya menyimpan dan memproteksi data secara lokal, namun tidak mengandung sebuah prosesor untuk melakukan perhitungan komputer pada data. Sedangkan *microprocessor card* adalah sebuah kartu yang memiliki memori dan *microprocessor* yang dapat melakukan kalkulasi pada data dan menyimpan data di dalam kartu secara aman (Muhammad Fauzan, 2012).

Sebuah *memory card* dapat menyimpan data sebesar 4kb dan keuntungan dari *memory card* terletak pada teknologinya yang sederhana. Sedangkan *microprocessor card,* menawarkan sekuritas dan kemampuan multifungsional. Data yang disimpan dalam *microprocessor card* tidak dapat diakses secara langsung melalui aplikasi di luar kartu. *Microprocessor* mengontrol data dan memori yang mengatur pengaksesan data menurut kondisi yang diberikan seperti *password,* enkripsi dan lainnya. Keuntungan *microprocessor card* adalah dapat diintegrasikan ke lebih dari satu aplikasi (Muhammad Fauzan, 2012).

Secara fisik *smart card* dapat dibedakan menjadi dua jenis yaitu:

1. ***Contact Smart Card***

   Contact Smart Card bekerja dengan cara berkomunikasi secara fisik antara *card reader* dan *smart card pin contact* yang berbentuk segi empat berukuran ± 1 cm. *Contact smart card* tidak membutuhkan baterai dan akan aktif ketika terhubung dengan *card reader*. Saat terhubung dengan *card reader,* maka *chip* akan menunggu perintah *request* dari klien dari aplikasi untuk membaca informasi dari *chip* atau menulis informasi ke *chip*.



Gambar 2.2. Chip pada smart card (*sumber: Mustafa Sarinanto, 2003*)

Berikut adalah fungsi dari masing-masing *contact* menurut ISO/IEC 7816, yaitu:

a. VCC (*Supply Voltage*) Berfungsi sebagai penyedia tegangan listrik, tegangan listrik yang tersedia sebesar 3 atau 5 volt dengan toleransi maksimum ± 10%.
b. RST (*Reset Input*) Berfungsi untuk mengirimkan sinyal untuk me-*reset microprocessor.*
c. CLK (*Clock Input*) Berfungsi sebagai *timing* atau *clocking signal* yang mengatur frekuensi waktu dan kecepatan *microprocessor.*

d. GND (*Ground*) Merupakan *reference voltage* yang berlawanan dengan VCC di mana energi potensialnya diukur dan nilainya dianggap 0 volt.

e. VPP (*Programming Voltage*) Berfungsi menyediakan tegangan listrik yang berbeda dari VCC.

f. I/O (*Input*/*Output*) Berfungsi untuk memonitor semua komunikasi yang keluar maupun yang masuk ke kartu.

g. C4 dan C8 *Contact* C4 dan C8 merupakan *contact* tambahan yang digunakan bila sewaktu-waktu diperlukan untuk operasi pada kartu atau penggunaan *interface* di masa yang akan datang.

2. Contactless Smart Card

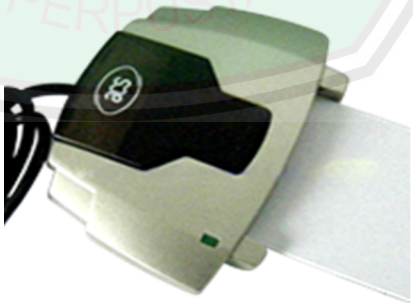
Contactless smart card berkomunikasi dengan terminal melalui sinyal frekuensi radio. *Contactless smart card* tidak memiliki baterai sehingga kartu ini memiliki *inductor* yang dapat menangkap sinyal frekuensi radio sebagai daya elektronik bagi kartu. *Contactless card* memerlukan jarak tertentu untuk melakukan pertukaran data dengan *card reader*. Standar komunikasi *contactless smart card* adalah ISO/IEC 14443.

Pada umumnya, *smartcard* tidak berisi *power supply*, *display* atau *keyboard*. *Smartcard* berinteraksi dengan dunia luar dengan menggunakan antarmuka komunikasi serial melalui 8 titik kontak. Ukuran dan letak dari kontak tersebut didefinisikan didalam ISO 7816, bagian kedua (Mustafa Sarinanto, 2003).

Smartcard dimasukkan ke dalam perangkat penerima *smartcard* (*Card Acceptance Device/CAD*), yang dapat dihubungkan dengan komputer. Istilah lain yang digunakan untuk CAD adalah terminal, *reader* dan IFD (*interface device*/perangkat antarmuka). Semuanya mempunyai fungsi dasar, yaitu menyediakan *power* ke *smartcard* dan membangun hubungan pertukaran data (Christine Sariasih, 2009).

2.1.4 *Card Reader* ACR 38U

Smart card ACR38U adalah pembaca kartu pintar (*smartcard)* yang memiliki kelebihan dalam hal teknologi membaca dan menulis pada semua jenis kartu *smart card. Smart card* ACR38U memberikan keseimbangan sempurna antara kehandalan dan fungsionalitas serta desain yang modern dalam memenuhi kebutuhan keamanan data pada *smart card. Smart card reader* telah banyak dipergunakan untuk keperluan seperti *e-Banking, e-Payment*, akses data jaringan dan instansi pemerintahan (www.acr38u.com).



Gambar 2.3. Smart Card Reader (*sumber: www.acr38u.com*)

Smart Card Reader ACR38 U memiliki Kecepatan koneksi penuh ke PC dengan sistem operasional yang sederhana dalam membaca dan menulis pada semua kartu *mikroprosesor smartcard reader* ACR38 U ini memiliki sirkuit perlindungan data yang sangat baik pada kartu pintar *smart card* ACR38U ini mendukung PPS (Protokol dan Parameter Seleksi) dengan 1.7 KBps dalam membaca dan menulis kartu pintar.

Fitur dari *Card reader* ACR38U adalah sebagai berikut :

a. Mendukung ISO 7816 Kelas A , B dan C (5 V , 3 V , 1,8 V) kartu
b. Mendukung CAC (*Common Access Card*)
c. Mendukung PIV (*Personal Identity Verification*) Kartu
d. Mendukung kartu mikroprosesor dengan T = 0 atau T = 1 protokol
e. Mendukung kartu memori : Kartu mengikuti protokol I2C bus (kartu memori bebas)
f. Kartu dengan cerdas 256 *byte* EEPROM dan menulis melindungi fungsi , termasuk: SLE4432 , SLE4442 , SLE5532 , SLE5542
g. Kartu dengan cerdas 1K *byte* EEPROM dan menulis - melindungi fungsi , termasuk: SLE4418 , SLE4428 , SLE5518 , SLE5528
h. Kartu dengan tipe ' 104 ' dari EEPROM non - reloadable kartu tanda *counter*, termasuk: SLE4406 , SLE4436 , SLE5536 , SLE6636
i. Kartu memori dengan aman IC dengan *password* dan otentikasi , termasuk: AT88SC153 , AT88SC1608

j. Kartu dengan Cerdas 416 - Bit EEPROM dengan cek PIN internal termasuk : SLE4404

k. Kartu dengan *Security Logic* dengan Zona Aplikasi , termasuk: AT88SC101 , AT88SC102 , AT88SC1003

l. Mendukung PPS (*Protocol* dan Parameter Seleksi)

m. Fitur Sirkuit Perlindungan pendek

n. *Application Programming Interface* : Mendukung PC / SC

o. Mendukung CT - API (melalui pembungkus di atas PC / SC)

2.1.5 Kartu SLE 4428

Memory Card SLE4428 adalah, standar EEPROM murah *smart card* memori dengan ruang penyimpanan 1K*byte*. SLE4428 adalah kartu pintar yang dilindungi yang terdiri dari 1.024 *byte* memori EEPROM. Setelah delapan entri yang salah berturut-turut konter kesalahan akan memblokir segala upaya berikutnya pada verifikasi PSC dan karenanya kemungkinan untuk menulis dan menghapus. Hal ini tidak diragukan lagi yang memberikan rasa aman bagi pengguna (http://kartuchip.com/chip_fm4428_sle4428.html).

Pembacaan memori seluruh selalu mungkin, dan memori yang dapat ditulis dan dihapus per *byte*. Kartu memori SLE4428 adalah ISO7816 kompatibel, dengan ukuran kartu langsing (sekitar 86 x 54 x 0,76 mm) menawarkan pengguna yang rasa kenyamanan. Kartu ini selalu siap untuk cetak *offset* dan cetak *silkscreen*, dengan retensi data hingga 10 tahun yang panjang.

Gambar 2.4. Kartu SLE4428 (*sumber: http://kartuchip.com*)

Spesifikasi dari kartu SLE4428 :

- 1K *byte* EEPROM
- Menulis-melindungi fungsi
- *Programmable logic* Keamanan Kode verifikasi
- Semua memori, kecuali untuk PSC, selalu dapat dibaca
- Memori dapat ditulis atau dihapus hanya setelah PSC memverifikasi
- Setelah 8 entri yang salah berturut-turut *counter error* akan memblokir segala upaya berikutnya pada verifikasi PSC dan maka kemungkinan apapun untuk menulis dan menghapus
- Dimensi: CR80 ukuran standar kartu kredit
- Bahan: PVC

2.2 KRIPTOGRAFI

Kriptografi (*cryptography*) berasal dari bahasa Yunani, terdiri dari dua suku kata yaitu *kripto* dan *graphia*. *Kripto* artinya menyembunyikan, sedangkan *graphia* artinya tulisan. Kriptografi adalah ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi, seperti

kerahasiaan data, keabsahan data, integritas data, serta autentikasi data. Tetapi tidak semua aspek keamanan informasi dapat diselesaikan dengan kriptografi (Eka Risky Firmansyah, 2012).

Kriptografi dapat pula diartikan sebagai ilmu atau seni untuk menjaga keamanan pesan. Ketika suatu pesan dikirim dari suatu tempat ke tempat lain, isi pesan tersebut mungkin dapat disadap oleh pihak lain yang tidak berhak untuk mengetahui isi pesan tersebut. Untuk menjaga pesan, maka pesan tersebut dapat diubah menjadi suatu kode yang tidak dapat dimengerti oleh pihak lain. Enkripsi adalah sebuah proses penyandian yang melakukan perubahan sebuah kode dari yang bisa dimengerti (plainteks) menjadi sebuah kode yang tidak bisa dimengerti (cipherteks). Sedangkan proses kebalikannya untuk mengubah cipherteks menjadi plainteks disebut dekripsi. Proses enkripsi dan dekripsi memerlukan suatu mekanisme dan kunci tertentu (Eka Risky Firmansyah, 2012).

Kriptoanalisis (*cryptanalysis*) adalah kebalikan dari kriptografi, yaitu suatu ilmu untuk memecahkan mekanisme kriptografi dengan cara mendapatkan kunci dari cipherteks yang digunakan untuk mendapatkan plainteks. Kriptologi (*cryptology*) adalah ilmu yang mencakup kriptografi dan kriptoanalisis (Eka Risky Firmansyah, 2012).

2.2.1 Algoritma Kriptografi

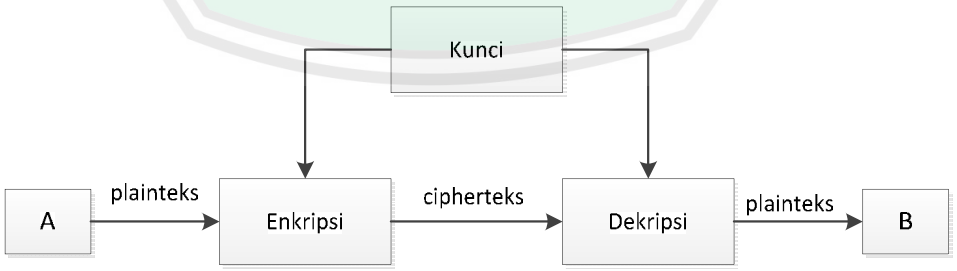
Algoritma kriptografi atau sering disebut dengan *cipher* adalah suatu fungsi matematis yang digunakan untuk melakukan enkripsi dan dekripsi. Ada

dua macam algoritma kriptografi, yaitu algoritma simetris (*symmetric algorithms*) dan algoritma asimetris (*asymmetric algorithms*).

a. Algoritma Simetris

Algoritma simetris adalah algoritma kriptografi yang menggunakan kunci enkripsi yang sama dengan kunci dekripsinya. Algoritma ini mengharuskan pengirim dan penerima menyetujui suatu kunci tertentu sebelum mereka saling berkomunikasi. Keamanan algoritma simetris tergantung pada kunci, membocorkan kunci berarti bahwa orang lain dapat mengenkripsi dan mendekripsi pesan. Agar komunikasi tetap aman, kunci harus tetap dirahasiakan. Algoritma simetris sering juga disebut dengan algoritma kunci rahasia, algoritma kunci tunggal, atau algoritma satu kunci (Oky Dwi Nurhayati, 2009).

Sifat kunci yang seperti ini membuat pengirim harus selalu memastikan bahwa jalur yang digunakan dalam pendistribusian kunci adalah jalur yang aman atau memastikan bahwa seseorang yang ditunjuk membawa kunci untuk dipertukarkan adalah orang yang dapat dipercaya.



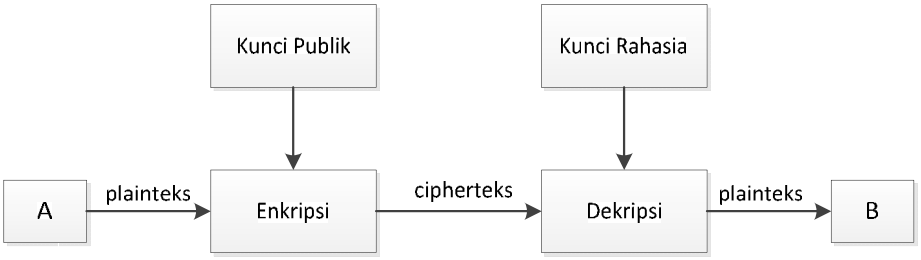
Gambar 2.5. Algoritma Simetrik

Contoh dari algoritma kriptografi simetris adalah Cipher Permutasi, Cipher Substitusi, Cipher Hill, OTP, RC6, Twofish, Magenta, FEAL, SAFER, LOKI, CAST, Rijndael (AES), Blowfish, GOST, A5, Kasumi, DES dan IDEA.

b. Algoritma Asimetris

Algoritma asimetris, sering juga disebut dengan algoritma kunci publik, menggunakan dua jenis kunci, yaitu kunci publik (*public key*) dan kunci rahasia (*secret key*). Kunci publik merupakan kunci yang digunakan untuk mengenkripsi pesan. Sedangkan kunci rahasia digunakan untuk mendekripsi pesan (Oky Dwi Nurhayati, 2009).

Kunci publik bersifat umum, artinya kunci ini tidak dirahasiakan sehingga dapat dilihat oleh siapa saja. Sedangkan kunci rahasia adalah kunci yang dirahasiakan dan hanya orang-orang tertentu saja yang boleh mengetahuinya. Keuntungan utama dari algoritma ini adalah memberikan jaminan keamanan kepada siapa saja yang melakukan pertukaran informasi meskipun di antara mereka tidak ada kesepakatan mengenai keamanan pesan terlebih dahulu maupun saling tidak mengenal satu sama lainnya (Stefanus Astrianto, 2008).



Gambar 2.6. Algoritma Asimetrik

Contoh dari algoritma asimetris adalah RSA, ElGamal, McEliece, LUC dan DSA (*Digital Signature Algorithm*).

Dalam melakukan proses enkripsi, sering digunakan plainteks berupa data ataupun pesan yang besar, sehingga membutuhkan waktu yang lama apabila dilakukan proses sekaligus pada plainteks tersebut. Oleh karena itu, plainteks dapat dipotong-potong menjadi beberapa blok-blok yang sama panjang. Kemudian dari blok-blok yang diperoleh tersebut dilakukan proses enkripsi, dan hasil cipherteksnya dapat didekripsi dan digabungkan kembali menjadi plainteks. Algoritma kriptografi yang menggunakan mekanisme seperti ini disebut dengan cipher blok (*block cipher*) (Oky Dwi Nurhayati, 2009).

2.3. ALGORITMA RC4

Algoritma kriptografi Rivest Code 4 (RC4) merupakan salah satu algoritma kunci simetris dibuat oleh RSA Data Security Inc (RSADSI) yang berbentuk *stream chipper*. Algoritma ini ditemukan pada tahun 1978 oleh Ronald Rivest dan menjadi simbol keamanan RSA (merupakan singkatan dari tiga nama penemu: Rivest Shamir Adleman). RC4 menggunakan panjang

kunci dari 1 sampai 256 *byte* yang digunakan untuk menginisialisasikan tabel sepanjang 256 *byte*. Tabel ini digunakan untuk generasi yang berikut dari *pseudo random* yang menggunakan XOR dengan plainteks untuk menghasilkan cipherteks. Masing-masing elemen dalam tabel saling ditukarkan minimal sekali (Febrian Wahyu, 2012).

RC4 merupakan salah satu jenis *stream cipher* sehingga RC4 memproses unit atau input data, pesan atau informasi pada satu saat. Unit atau data pada umumnya sebuah *byte* sehingga dengan cara ini enkripsi atau dekripsi dapat dilaksanakan pada panjang yang variabel. Algoritma ini tidak harus menunggu sejumlah input data, pesan atau informasi tertentu sebelum diproses, atau menambahkan *byte* tambahan untuk mengenkrip (Yuri Ariyanto, 2009).

Algoritma RC4 menggunakan dua buah S-Box yaitu *array* sepanjang 256 yang berisi permutasi dari bilangan 0 sampai 255, dan S-Box kedua, yang berisi permutasi merupakan fungsi dari kunci dengan panjang yang variable (Yuri Ariyanto, 2009).

Secara umum, algoritma RC4 terbagi menjadi dua, inisialisasi *state-array* dan penghasilan kunci enkripsi serta pengenkripsiannya.

2.3.1 Inisialisasi *State-Array*

Dalam penginisialisasian *state-array*, terdapat 2 *state-array* yang harus diinisialisasi, S dan K. *Array* S sebesar 256 bit diinisialisasi dengan

angka dari 0 sampai dengan 255. Sedangkan *array* K sebesar 256 bit diisi dengan *key* dengan panjang 1-256 bit secara berulang sampai seluruh *array* K terisi penuh. Setelah itu, dilakukan *Key Scheduling Algorithm* untuk menghasilkan permutasi dari *array* S berdasarkan *key* yang tersedia (Karina Novita Suryani, 2009).

2.3.2 Penghasilan kunci Enkripsi dan Pengenkripsiannya

Setelah memiliki *state array* yang telah teracak, maka kita akan menginisialisasi kembali i dan j dengan 0. Setelah itu, kita lakukan *pseudo-random generation algorithm* atau PRGA untuk menghasilkan kunci enkripsi (*cipher_byte*) yang akan di-XOR-kan dengan plainteks. Untuk menghasilkan kunci enkripsi, PRGA meng-*incremen* i, menambahkan nilai S[i] dan S[j] menukar nilai keduanya, dan nilai kunci yang dihasilkan adalah S dengan indeks yang sama dengan jumlah S[i] dan S[j] di-modulo dengan 256 (Karina Novita Suryani, 2009).

PRGA akan dilakukan sebanyak 4 kali dikarenakan plainteks yang akan dienkripsi berjumlah 4 karakter. Hal ini disebakan karena dibutuhkan 1 kunci dan 1 kali pengoperasian XOR untuk tiap-tiap karakter pada plainteks.

Setelah menemukan kunci untuk tiap karakter, maka dilakukan operasi XOR antara karakter pada plainteks dengan kunci yang dihasilkan.

2.4 PENELITIAN TERKAIT

Penelitian oleh Dedy Pamungkas(2005) mengenai *Aplikasi Smart Card Sebagai kartu Pra Bayar Internet*. Pada penelitian ini, peneliti bertujuan membuat aplikasi yang berfungsi sebagai kartu pra bayar internet. Sistem ini juga dapat melakukan pengisian ulang pulsa. Yang diujikan pada sistem ini adalah penulisan data ke memori utama *smart card* dan pembacaan data pada memori utama *smart card* serta pengujian efektifitas pemakaian aplikasi. Pada aplikasi ini terdapat PIN yang berfungsi untuk melindungi sistem dari penyalahgunaan data.

Penelitian oleh Dodon Yendri(2008) mengenai *Rancangan Model Layanan Transaksi Peminjaman Buku Menggunakan Teknologi Smart Card pada Perpustakaan Induk*. Pada penelitian ini, peneliti telah menjelaskan secara rinci rancangan-rancangannya, mulai dari membangkitkan kunci *public*, kunci *private*, dan nilai *hash*. Metode yang digunakan yaitu metode RSA, metode ini mengambil dua bilangan secara acak yang akan dijadikan kunci sehingga didapat dua kunci yaitu kunci *public* dan kunci *private*. Disini telah dijelaskan pihak-pihak yang bertugas untuk membangkitkan kunci, tetapi tidak dijelaskan bagaimana proses pengamanan datanya. Sehingga tidak diketahui hasil akhir dari penelitian ini.

Penelitian oleh Angga Rusdinar(2008) mengenai *Perancangan Sistem Kartu Identitas Berbasis Smart Card*. Pada penelitian ini, peneliti

menjelaskan mengenai pemrograman *hardware* pada *smart card*. Objek yang digunakan pada penelitian ini adalah data identitas dari akte kelahiran, KTP, dan SIM. Akan tetapi, peneliti tidak menjelaskan metode yang digunakan pada penelitian ini.

Penelitian oleh Karina Novita(2009) mengenai *Algoritma RC4 sebagai Metode Enkripsi*. Pada penelitian ini, peneliti menjelaskan mekanisme dasar kerja algoritma RC4 secara rinci dan jelas. Algoritma RC4 ini digunakan untuk meningkatkan keamanan data, melindungi data atau pesan agar tidak terbaca oleh pihak lain. Algoritma RC4 merupakan metode enkripsi tercepat dibandingkan dengan DES, Triple-DES, Blowfish-256, AES-128, dan AES-256. Namun, algoritma RC4 juga memiliki kelemahan antara lain, tingginya peluang untuk menghasilkan *array* S yang sama ataupun berulang dan *Bit-Flipping Attack*.

Penelitian oleh Yefriadi(2010) mengenai *Aplikasi Smart Card untuk Electronic Medical Record (EMR)*. Pada penelitian ini, peneliti membuat sebuah aplikasi yang digunakan untuk menyimpan rekam medis pasien di rumah sakit, mulai dari rawat jalan, rawat inap, dan hasil diagnose dokter. Akan tetapi, pada penelitian ini tidak dijelaskan metode yang digunakan oleh peneliti, peneliti hanya menjelaskan diagram alir dari proses login sampai pembuatan *smart card* baru.

Penelitian oleh Yanno Dwi Ananda(2010) mengenai *Rancang Bangun Aplikasi Smart Card Interface*. Penelitian ini menjelaskan mengenai beberapa bahasa pemrograman yang digunakan dalam *smart card*. Peneliti membahas mengenai kompabilitas *smart card* dengan beberapa bahasa pemrograman dan kecepatan aplikasi untuk membaca data *smart card*. Hasilnya, efisiensi aplikasi *smart card* dipengaruhi oleh bahasa pemrograman.

Penelitian oleh Febrian Wahyu dkk(2012) mengenai *Penerapan Algoritma Gabungan RC4 dan Base64 pada Sistem Keamanan E-Commerce*. Dalam penelitian ini dibahas tentang algoritma kriptografi RC4 dan *Encoding Base64* dalam menyelesaikan dan mengatasi masalah keamanan dalam *E-Commerce* yang dimodelkan melalui interaksi antara *Website* Rental Mobil dan *Bank*. Sistem yang dibuat dapat menjamin keamanan data transaksi pembayaran *online* yang dilakukan oleh pelanggan karena *password* pelanggan di *Bank* telah disamarkan dengan proses enkripsi dan sangat sulit dipecahkan apabila kunci dan perhitungan algoritma berbeda.

BAB III

PERANCANGAN DAN IMPLEMENTASI

Bab ini membahas mengenai perancangan sistem pembuatan aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan yang menerapkan algoritma RC4. Pada bab ini yang akan dibahas meliputi deskripsi sistem, perancangan sistem, perancangan database, perancangan flowchart, perancangan perangkat keras, dan perancangan *interface*.

3.1 DESKRIPSI SISTEM

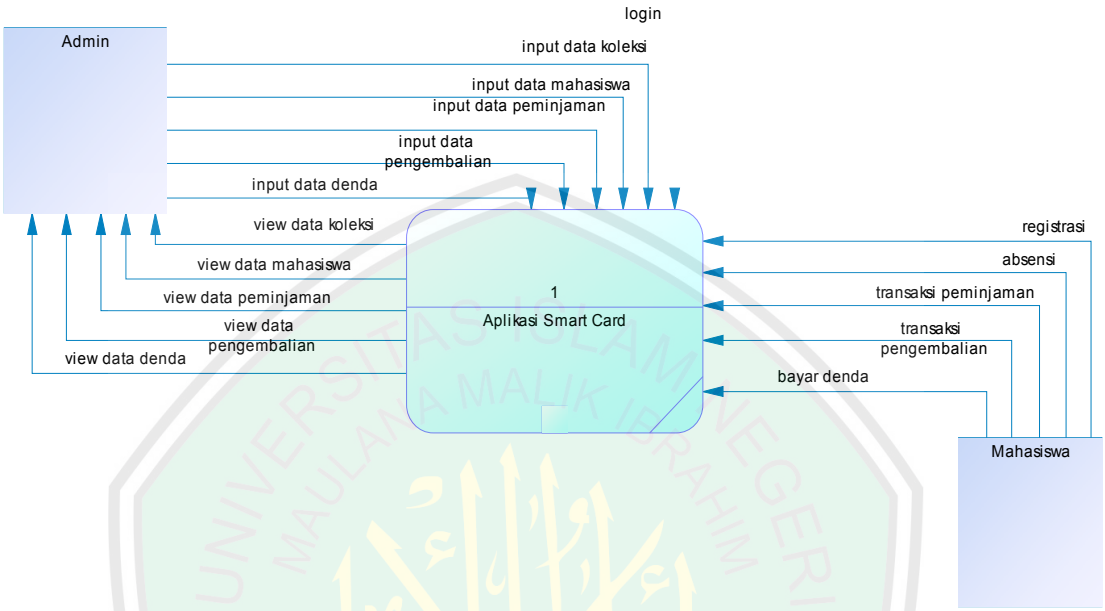
Aplikasi *smart card* ini akan dibangun dengan menggunakan algoritma RC4. Algoritma RC4 ini digunakan untuk proses enkripsi dan dekripsi data yang ada di dalam *smart card*. Tujuan dari penggunaan algoritma RC4 ini adalah untuk mengamankan data yang ada pada *smart card* sehingga data tersebut tidak dapat dibaca oleh pihak yang tidak berwenang. Algoritma RC4 merupakan algoritma simetrik, yaitu algoritma yang memiliki kunci sama untuk proses enkripsi dan dekripsi. Proses utama pada aplikasi ini adalah dengan melakukan enkripsi data yang telah diinputkan ke dalam *smart card* dengan kunci rahasia yang telah dibentuk.

3.2 PERANCANGAN SISTEM

3.2.1 Diagram Konteks

Diagram konteks memodelkan hubungan antara aplikasi dengan entitas. Selain itu, diagram konteks merupakan diagram yang paling awal yang menjelaskan

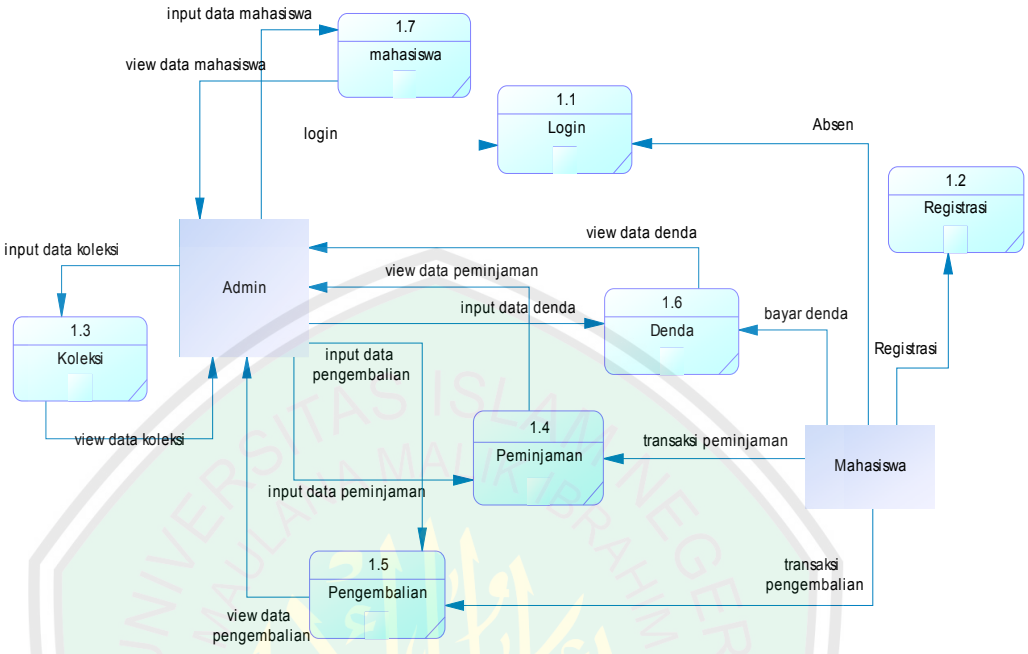
sistem secara umum. Aliran data dalam diagram konteks memodelkan masukan ke dalam sistem dan keluaran dari sistem.



Gambar 3.1. Context Diagram

Gambar 3.1 menjelaskan mengenai arus aplikasi *smart card* untuk layanan transaksi peminjaman buku perpustakaan, yang secara umum melibatkan dua entitas, yaitu : admin dan mahasiswa. Admin merupakan penyedia layanan aplikasi perpustakaan dan terdapat sebelas aliran data, dimana enam aliran data menuju aplikasi dan lima aliran data keluar dari aplikasi. Sedangkan mahasiswa merupakan pengguna dari aplikasi perpustakaan dan terdapat lima aliran data yang keseluruhannya menuju aplikasi.

3.2.2 Data Flow Diagram Level 1

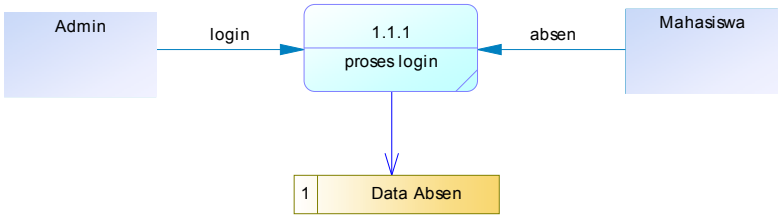


Gambar 3.2. DFD Level 1

Gambar 3.2 menjelaskan bagian proses yang lebih rinci dari proses secara umum yang ada pada diagram konteks sebelumnya. Terdapat tujuh proses pada data flow diagram level 1 diatas, antara lain yaitu: login, registrasi, koleksi, peminjaman, pengembalian, denda, dan mahasiswa.

3.2.3 Data Flow Diagram Level 2

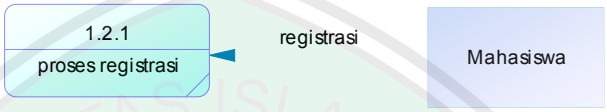
3.2.3.1 DFD Level 2 Proses Login



Gambar 3.3. DFD Level 2 Proses Login

Gambar 3.3 menjelaskan kegiatan arus data antara admin dengan mahasiswa. Pada proses tersebut terdapat dua aliran data yaitu admin login ke dalam aplikasi dan mahasiswa melakukan absen, kemudian data absen akan masuk pada database absen.

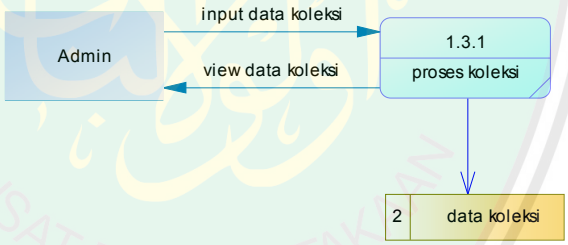
3.2.3.2 DFD Level 2 Proses Registrasi



Gambar 3.4. DFD Level 2 Proses Registrasi

Gambar 3.4 menjelaskan kegiatan arus data oleh mahasiswa. Pada proses tersebut terdapat satu aliran data saja, yaitu registrasi mahasiswa ke dalam aplikasi.

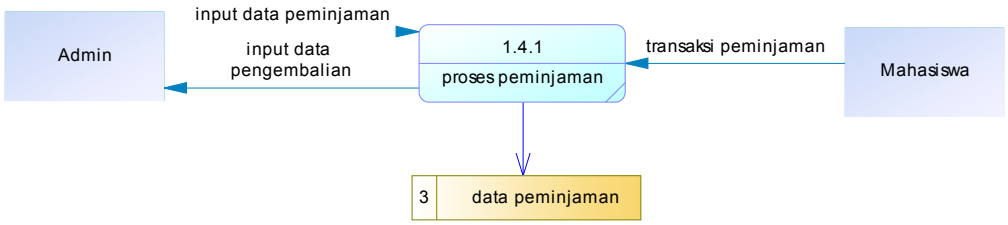
3.2.3.3 DFD Level 2 Proses Koleksi



Gambar 3.5. DFD Level 2 Proses Koleksi

Gambar 3.5 menjelaskan kegiatan arus data oleh admin. Pada proses tersebut terdapat dua aliran data, yaitu menginputkan data koleksi buku ke dalam aplikasi dan dapat melihat data koleksi yang ada di dalam aplikasi.

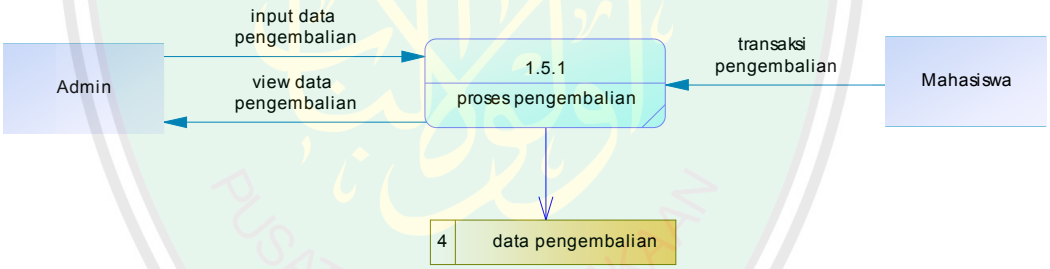
3.2.3.4 DFD Level 2 Proses Peminjaman



Gambar 3.6. DFD Level 2 Proses Peminjaman

Gambar 3.6 menjelaskan kegiatan arus data oleh admin dan mahasiswa. Pada proses tersebut terdapat tiga aliran data, yaitu mahasiswa melakukan transaksi peminjaman buku, admin menginputkan data peminjaman ke dalam aplikasi dan dapat melihat data peminjaman buku.

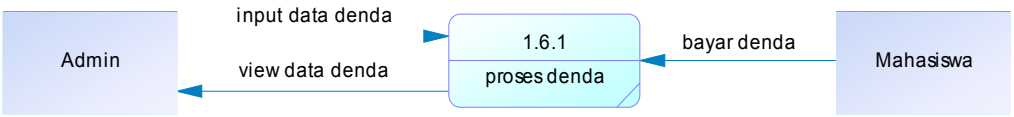
3.2.3.5 DFD Level 2 Proses Pengembalian



Gambar 3.7. DFD Level 2 Proses Pengembalian

Gambar 3.7 menjelaskan kegiatan arus data oleh admin dan mahasiswa. Pada proses tersebut terdapat tiga aliran data, yaitu mahasiswa melakukan transaksi pengembalian buku, admin menginputkan data pengembalian ke dalam aplikasi dan dapat melihat data pengembalian buku.

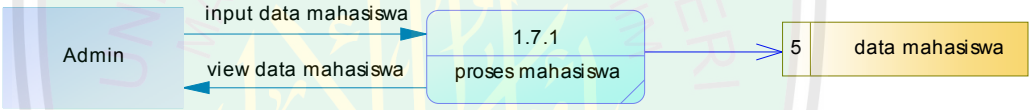
3.2.3.6 DFD Level 2 Proses Denda



Gambar 3.8. DFD Level 2 Proses Denda

Gambar 3.8 menjelaskan kegiatan arus data oleh admin dan mahasiswa. Pada proses tersebut terdapat tiga aliran data, yaitu mahasiswa melakukan pembayaran denda, admin menginputkan data pembayaran denda ke dalam aplikasi dan dapat melihat data denda.

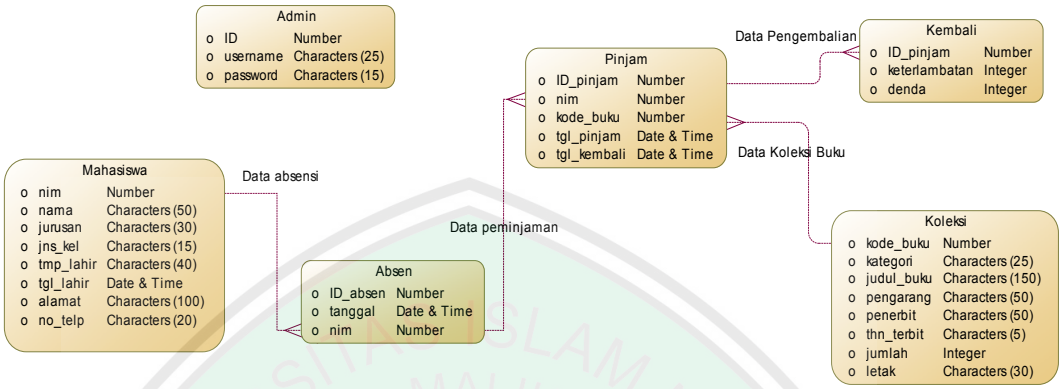
3.2.3.7 DFD Level 2 Proses Mahasiswa



Gambar 3.9. DFD Level 2 Proses Mahasiswa

Gambar 3.9 menjelaskan kegiatan arus data oleh admin. Pada proses tersebut terdapat dua aliran data, yaitu admin menginputkan data mahasiswa ke dalam aplikasi dan dapat melihat data mahasiswa.

3.3 PERANCANGAN DATABASE

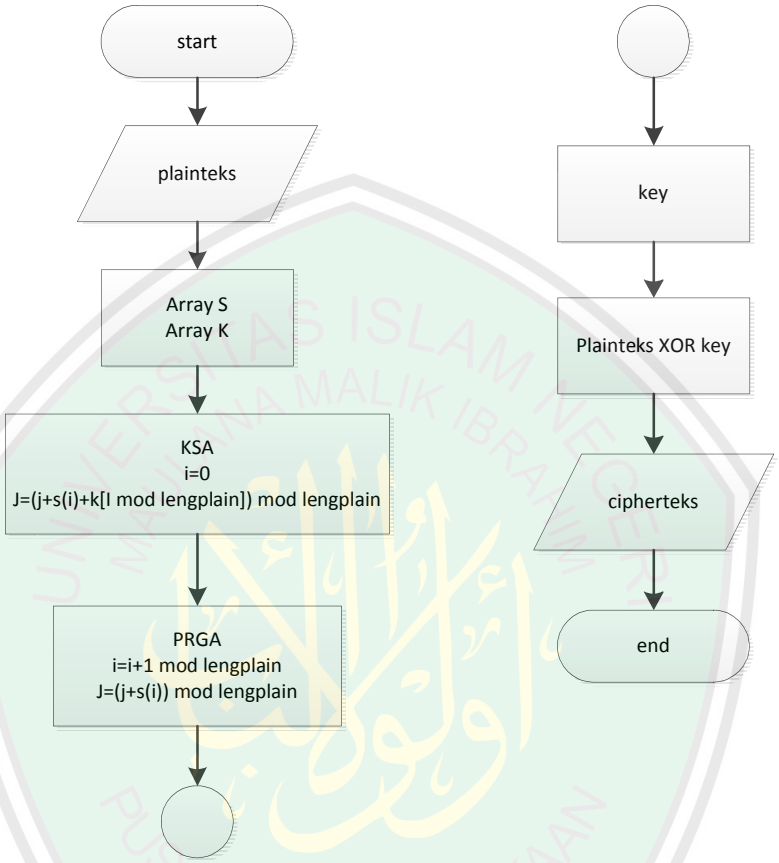


Gambar 3.10. Rancangan Database

Perancangan database pada Gambar 3.10 ini menggunakan *Database Access* yang merupakan salah satu program aplikasi basis data yang mempunyai kegunaan untuk merancang, membuat, dan mengelola database. Pada perancangan database tersebut, terdapat enam tabel yaitu: admin, koleksi, mahasiswa, absen, pinjam, dan kembali. Untuk tabel admin, tabel tersebut berdiri sendiri dan tidak terdapat relasi dengan tabel-tabel yang lain. Untuk tabel mahasiswa pada *field* nim terdapat relasi dengan tabel absen pada *field* nim juga. Tabel absen pada *field* nim tersebut juga terdapat relasi dengan tabel pinjam pada *field* nim. Untuk tabel pinjam pada *field* ID_pinjam terdapat relasi dengan tabel kembali pada *field* ID_pinjam. Sedangkan tabel pinjam pada *field* kode buku terdapat relasi dengan *field* kode buku pada tabel koleksi.

3.4 PERANCANGAN FLOWCHART

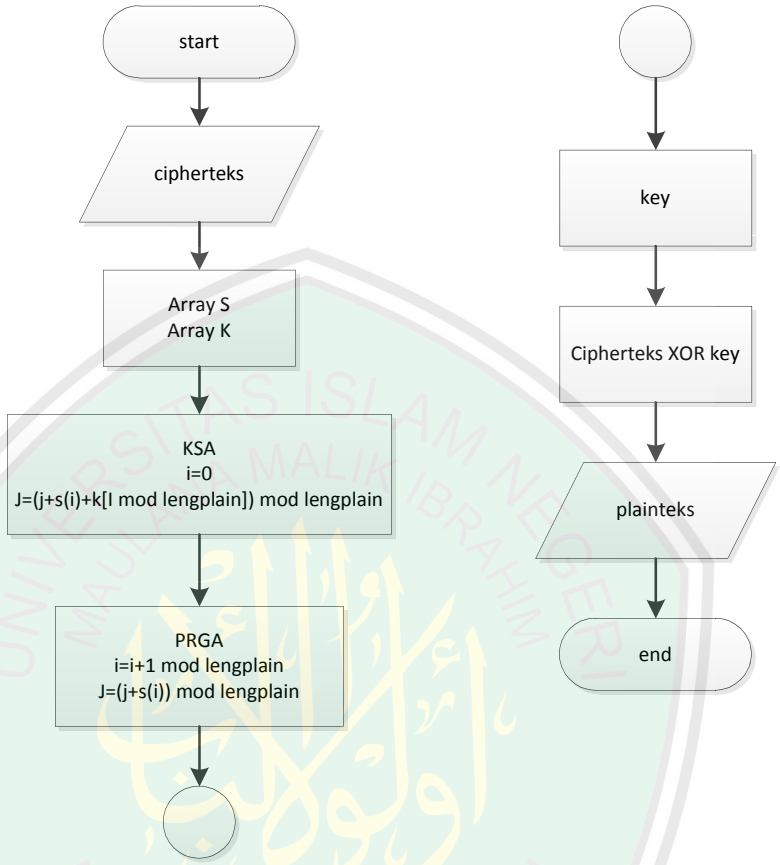
3.4.1 Flowchart Algoritma RC4 Proses Enkripsi



Gambar 3.11. Flowchart Algoritma RC4 Proses Enkripsi

Flowchart pada Gambar 3.11 dimulai dengan inputan yang diinisialisasikan dengan *array* S dan *array* K. Kemudian dilakukan proses KSA dan dilanjutkan dengan proses PRGA. Maka terbentuklah sebuah kunci. Proses selanjutnya yaitu proses enkripsinya yaitu meng-XOR-kan plainteks dengan kunci yang tadi telah terbentuk. Hasil akhirnya yaitu berupa cipherteks.

3.4.2 Flowchart Algoritma RC4 Proses Dekripsi



Gambar 3.12. Flowchart Algoritma RC4 Proses Dekripsi

Flowchart pada Gambar 3.12 sama halnya dengan flowchart enkripsi pada langkah pembentukan kunci. Setelah kunci terbetuk, maka akan dilakukan proses dekripsi yaitu meng-XOR-kan cipherteks dengan kunci. Maka keluarlah hasil plainteks.

Berikut adalah perhitungan secara manual algoritma RC4. Sebagai contoh, kita akan mengenkripsi kata HALO dengan kunci 2573. Pertama, kita menginisialisasi *array* S 4 bit sehingga terbentuk *state-array* S dan K sebagai berikut,

Array S	0	1	2	3
Array K	2	5	7	3

Inisialisasi i dan j dengan 0 kemudian dilakukan KSA agar tercipta *state-array* yang acak. Penjelasan iterasi lebih lanjut dapat dijelaskan sebagai berikut.

Iterasi 1

i = 0

j = (0 + S[0] + K [0 mod 4]) mod 4

= (0 + 0 + 2) mod 4 = 2

Swap (S[0],S[2])

Hasil Array S

2	1	0	3

Iterasi 2

i = 1

j = (2 + S[1] + K [1 mod 4]) mod 4

= (2 + 1 + 5) mod 4 =0

Swap (S[1],S[0])

Hasil Array S

1	2	0	3

Iterasi 3

i = 2

j = (0 + S[2] + K [2 mod 4]) mod 4

= (0 + 0 + 7) mod 4 = 3

Swap (S[2],S[3])

Hasil Array S

1	2	3	0

Iterasi 4

i = 3

j = (3 + S[3] + K [3 mod 4]) mod 4

= (3 + 0 + 3) mod 4 = 2

Swap (S[3],S[2])

Hasil Array S

1	2	0	3

Setelah melakukan KSA, akan dilakukan PRGA.

Array S

1	2	0	3

Inisialisasi

i = 0

j = 0

Iterasi 1

i = (0 + 1) mod 4 = 1

j = (0 + S[1]) mod 4 = (0+ 2) mod 4 = 2

swap (S[1],S[2])

1	0	2	3

K1 = S[(S[1]+S[2]) mod 4]

= S[2 mod 4] = 2

K1 = 00000010

Iterasi 2

i = (1 + 1) mod 4 = 2

j = (2 + S[2]) mod 4 = (2+ 2) mod 4 = 0

swap (S[2],S[0])

2	0	1	3

K2 = S[(S[2]+S[0]) mod 4]

= S[3 mod 4] = 3

K2 = 00000011

Iterasi 3

i = (2 + 1) mod 4 = 3

j = (0 + S[3]) mod 4 = (0+ 3) mod 4 = 3

swap (S[3],S[3])

2	0	1	3

K3 = S[(S[3]+S[3]) mod 4]

= S[6 mod 4] = 1

K3 = 00000001

Iterasi 4

i = (3 + 1) mod 4 = 0

j = (3 + S[0]) mod 4 = (3+ 2) mod 4 = 1

swap (S[0],S[1])

1	0	2	3

K1 = S[(S[0]+S[1])mod 4]

= S[2 mod 4] = 1

K1 = 00000001

Setelah menemukan kunci untuk tiap karakter, maka dilakukan operasi XOR antara karakter pada plainteks dengan kunci yang dihasilkan. Berikut adalah tabel ASCII untuk tiap-tiap karakter pada plainteks yang digunakan.

Kode ASCII untuk setiap karakter plainteks dapat dilihat pada **Tabel 3.1**

Huruf	**Kode ASCII (Binary 8 bit)**
H	01001000
A	01000001
L	01001100
O	01001111

Proses XOR antara plainteks dengan kunci bisa dilihat pada tabel berikut.

Proses XOR kunci enkripsi dengan plainteks dapat dilihat pada **Tabel 3.2**

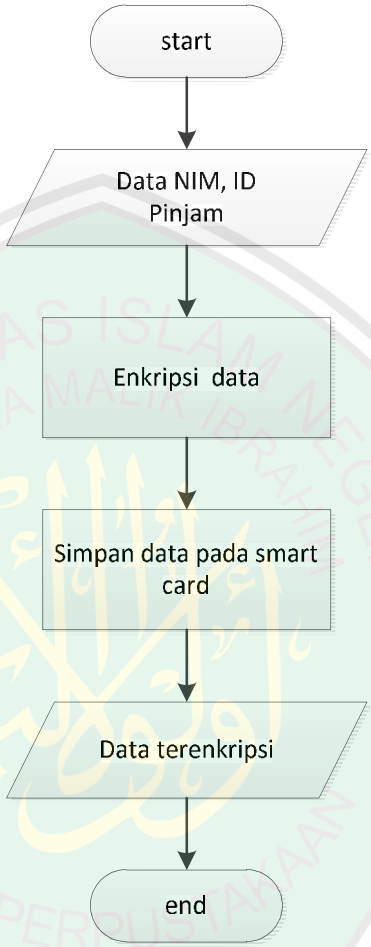
	H	A	L	O
Plainteks	01001000	01000001	01001100	01001111
Key	00000010	00000011	00000001	00000001
Cipherteks	01001010 (J)	01000010 (B)	01001101 (M)	01001110 (N)

Proses pendekripsian dilakukan dengan proses XOR antara cipherteks dengan kunci yang dapat dilihat pada table berikut.

Proses XOR kunci dekripsi dengan cipherteks dapat dilihat pada **Tabel 3.3**

	J	B	M	N
Cipherteks	01001010	01000010	01001101	01001110
Key	00000010	00000011	00000001	00000001
Plainteks	01001000 (H)	01000001 (A)	01001100 (L)	01001111 (O)

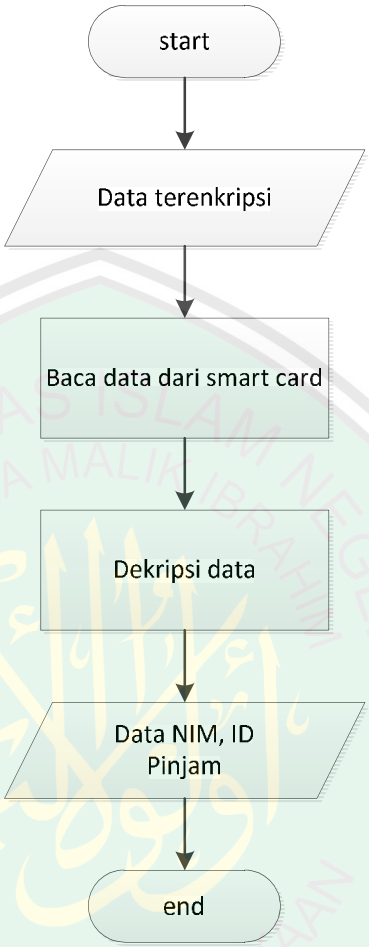
3.4.3 Flowchart Aplikasi Smart Card untuk Enkripsi Data



Gambar 3.13. Flowchart Aplikasi Smart Card Proses Enkripsi

Flowchart pada Gambar 3.13 dimulai dengan inputan berupa data NIM ataupun ID peminjaman. Kemudian inputan tersebut mengalami proses enkripsi dan proses selanjutnya data yang telah dienkripsi tersebut disimpan di dalam *smart card*. Output dari flowchart tersebut berupa data NIM atau ID peminjaman namun dalam bentuk terenkripsi.

3.4.4 Flowchart Aplikasi Smart Card untuk Dekripsi Data

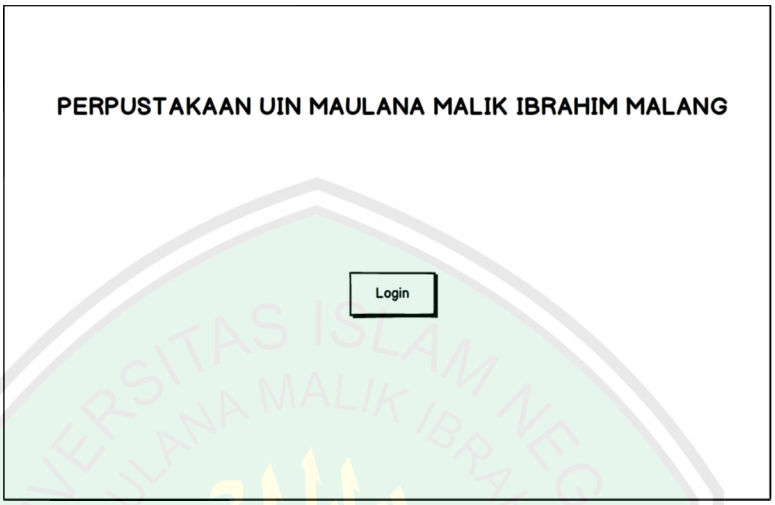


Gambar 3.14. Flowchart Aplikasi Smart Card Proses Dekripsi

Flowchart pada Gambar 3.14 dimulai dengan inputan berupa data yang telah terenkripsi.. Kemudian inputan tersebut dibaca yang diambil dari *smart card* dan dilanjutkan dengan proses dekripsi. Output dari flowchart tersebut berupa data NIM atau ID peminjaman, data yang sama seperti sebelum terjadi proses enkripsi.

3.5 PERANCANGAN INTERFACE

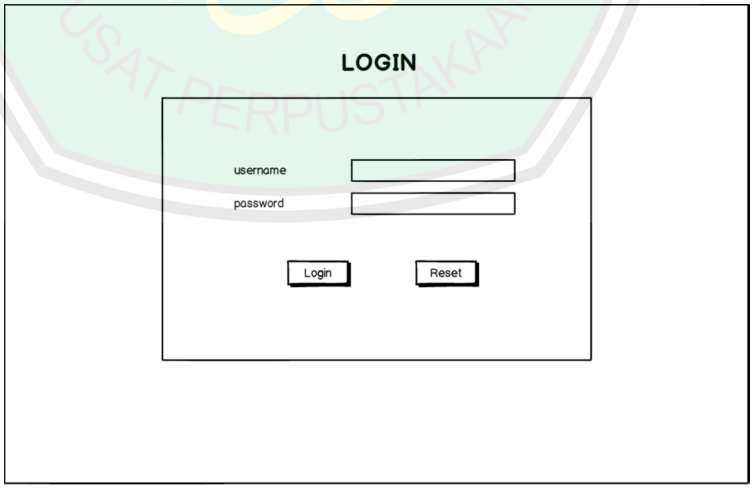
3.5.1 Interface Halaman Depan



Gambar 3.15. Interface Halaman Depan

Gambar 3.15 merupakan halaman depan pada aplikasi *smart card*, terdapat *button* login yang digunakan oleh admin untuk masuk ke dalam aplikasi.

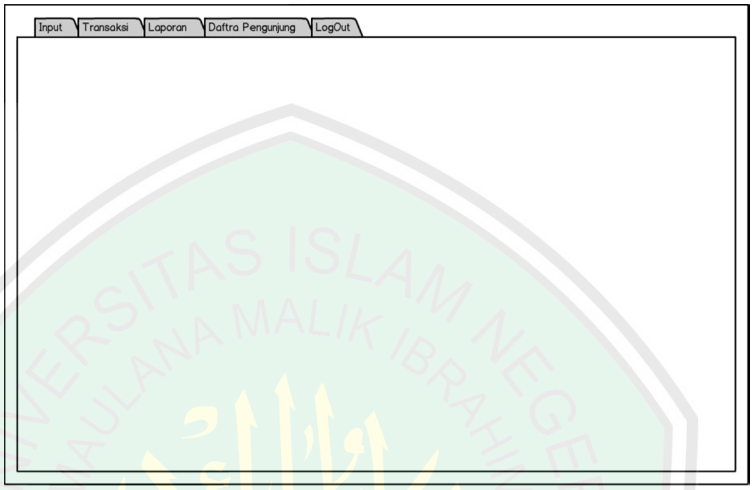
3.5.2 Interface Halaman Login



Gambar 3.16. Interface Halaman Login

Gambar 3.16 merupakan halaman login untuk admin. Untuk masuk ke dalam aplikasi, admin harus menginputkan *username* dan *password.*

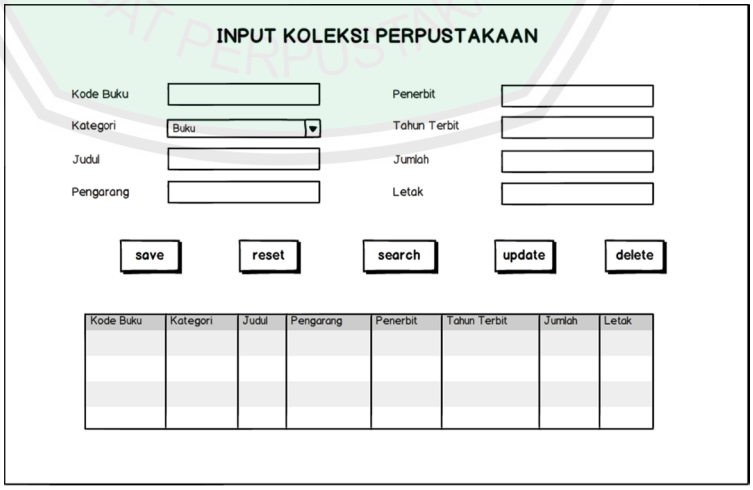
3.5.3 Interface Halaman Admin



Gambar 3.17. Interface Halaman Admin

Gambar 3.17 merupakan halaman admin. Pada halaman tersebut terdapat lima menu, yaitu: input, transaksi, laporan, data pengunjung, dan logout.

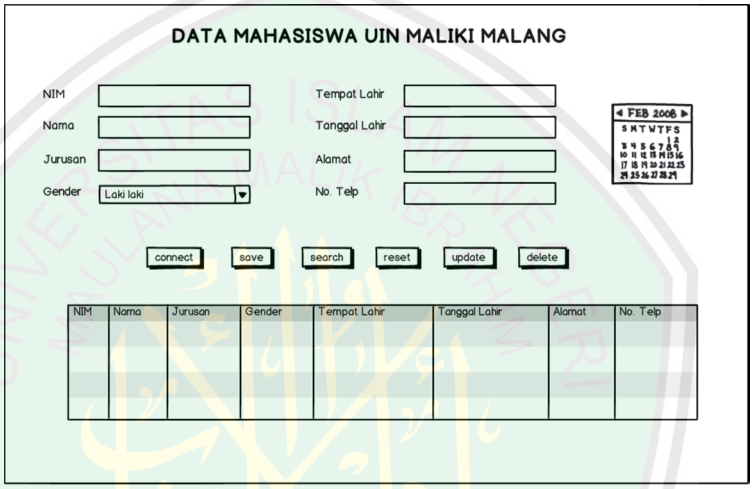
3.5.4 Interface Halaman Input Koleksi Perpustakaan



Gambar 3.18. Interface Input Koleksi Buku

Gambar 3.18 merupakan halaman untuk menginputkan koleksi buku, jurnal, dan skripsi pada aplikasi ini. Pada bagian bawah, terdapat tabel yang secara otomatis akan ter-*update* apabila admin telah menginputkan data dan data tersebut akan langsung muncul pada tabel tersebut.

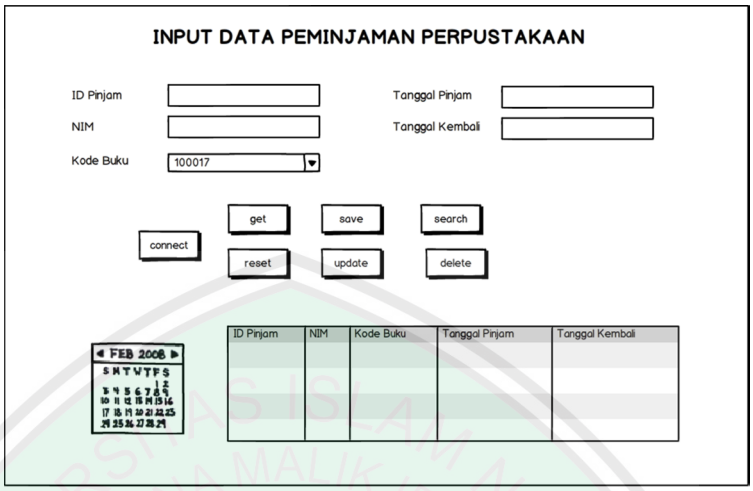
3.5.5 Interface Halaman Input Data Mahasiswa



Gambar 3.19. Interface Input DataMahasiswa

Gambar 3.19 merupakan halaman untuk menginputkan data mahasiswa di awal registrasi. Pada halaman tersebut terdapat *button connect* yang berfungsi untuk mengoneksikan *reader smart card* dengan aplikasi. Data yang telah diinputkan admin akan langsung ter-*update* pada tabel dibawahnya.

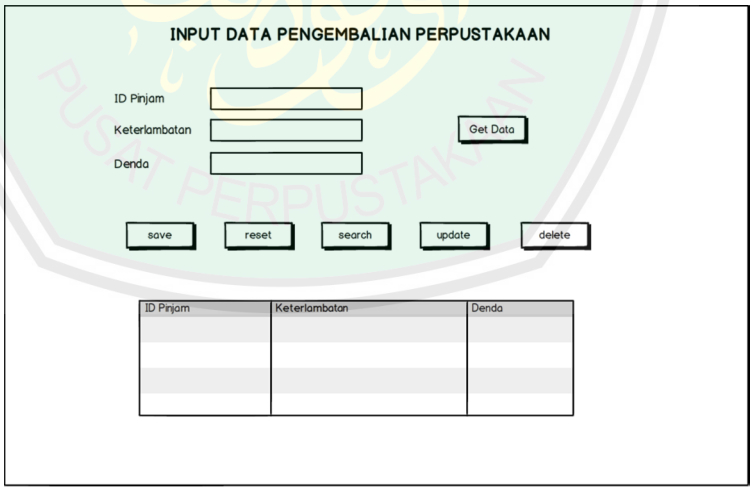
3.5.6 Interface Halaman Input Data Peminjaman Perpustakaan



Gambar 3.20. Interface Input Data Peminjaman

Gambar 3.20 merupakan halaman untuk menginputkan data peminjaman perpustakaan.

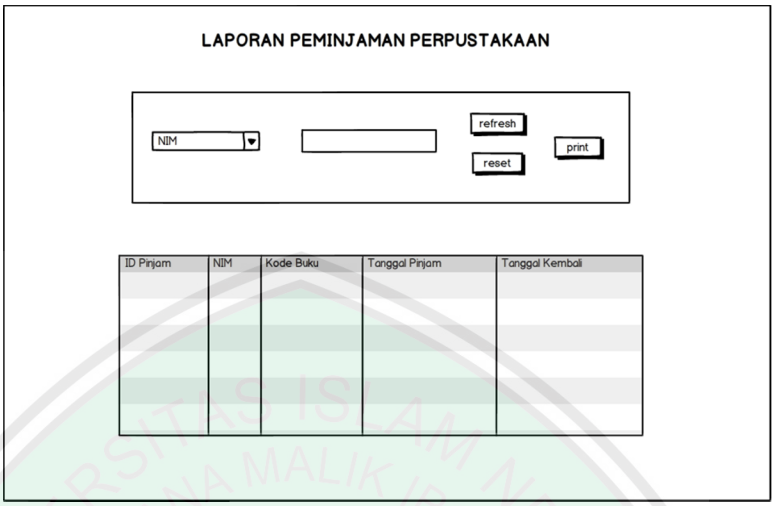
3.5.7 Interface Halaman Input Data Pengembalian Perpustakaan



Gambar 3.21. Interface Input Data Pengembalian

Gambar 3.21 merupakan halaman untuk menginputkan data pengembalian perpustakaan.

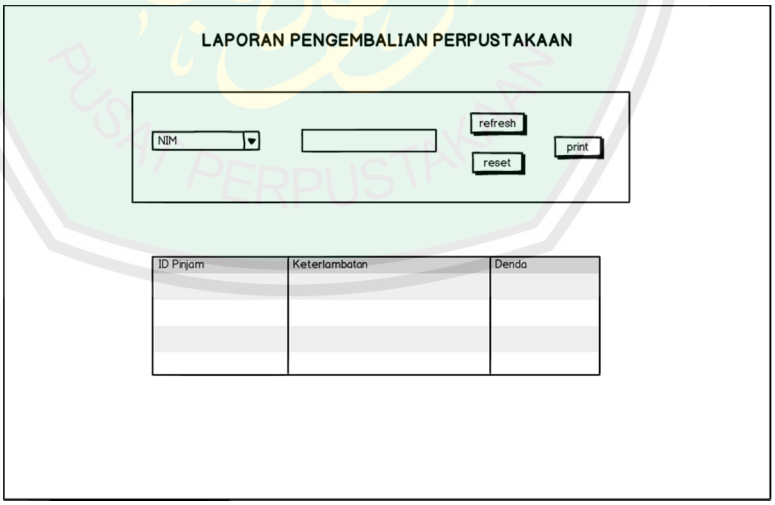
3.5.8 Interface Halaman Laporan Peminjaman Perpustakaan



Gambar 3.22. Interface Laporan Peminjaman

Gambar 3.22 merupakan halaman untuk menunjukkan laporan peminjaman perpustakaan dan hasilnya dapat langsung dicetak.

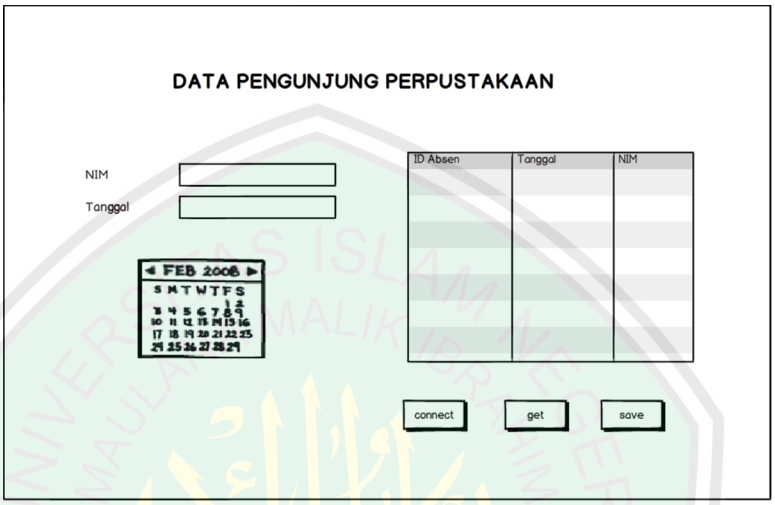
3.5.9 Interface Halaman Laporan Pengembalian Perpustakaan



Gambar 3.23. Inreface Laporan Pengembalian

Gambar 3.23 merupakan halaman untuk menunjukkan laporan pengembalian perpustakaan,dan hasilnya dapat langsung dicetak.

3.5.10 Interface Halaman Data Pengunjung Perpustakaan



Gambar 3.24. Interface Data Pengunjung

Gambar 3.24 merupakan halaman data pengunjung perpustakaan. Halaman tersebut berfungsi untuk menunjukkan siapa saja pengunjung yang mengunjungi perpustakaan tersebut.

BAB IV

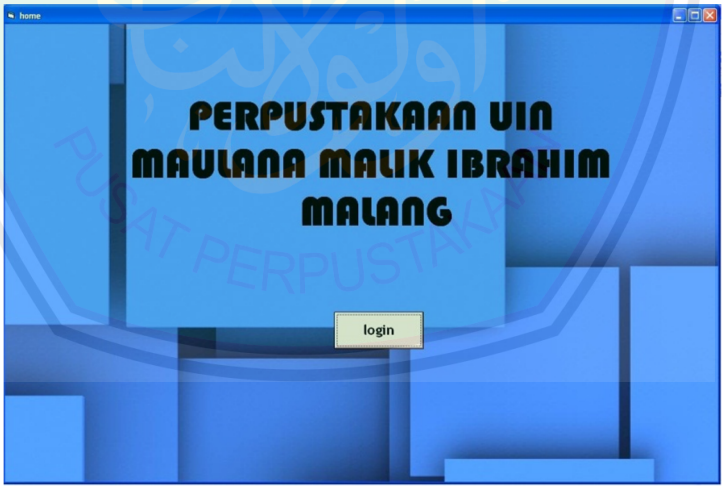
IMPLEMENTASI DAN HASIL UJI COBA

4.1 IMPLEMENTASI SISTEM

Bab ini menjelaskan mengenai implementasi dari aplikasi yang dibangun dan pengujian terhadap metode yang digunakan. Tujuan dilakukannya pengujian adalah untuk mengetahui apakah metode yang dipilih oleh penulis sesuai dan dapat berjalan dengan baik pada aplikasi tersebut. Selain itu juga untuk mengetahui apakah aplikasi yang dibangun sudah sesuai dengan apa yang sudah direncanakan.

4.1.1 Interface

4.1.1.1 Form Halaman Depan



Gambar 4.1. Form Halaman Depan

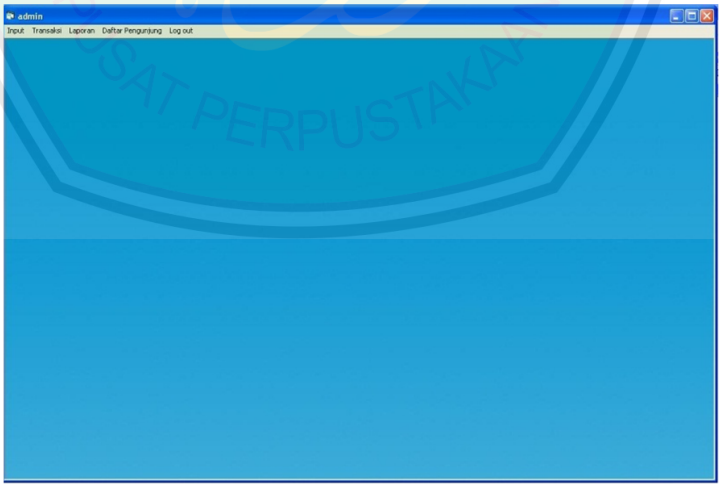
Gambar 4.1 merupakan halaman depan pada aplikasi *smart card*, terdapat *button* login yang digunakan oleh admin untuk masuk ke dalam aplikasi.

4.1.1.2 Form Halaman Login Admin

Gambar 4.2. Form Halaman Login Admin

Gambar 4.2 merupakan halaman login untuk admin. Untuk masuk ke dalam aplikasi, admin harus menginputkan *username* dan *password* yang sesuai dengan yang ada di *database*.

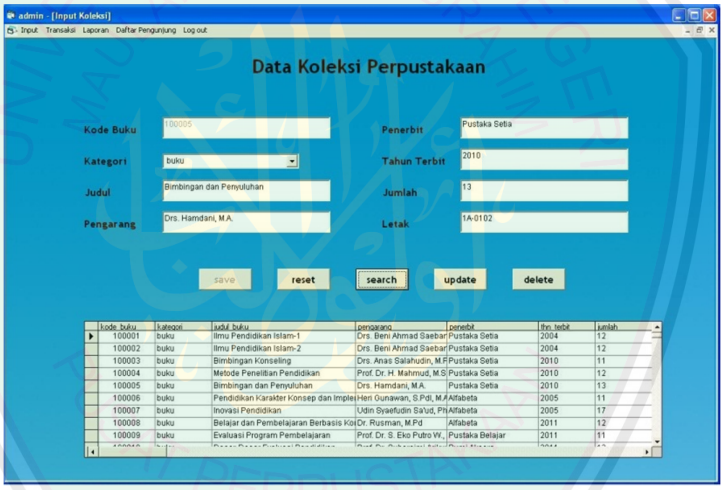
4.1.1.3 Form Halaman Depan Admin



Gambar 4.3. Form Halaman depan Admin

Gambar 4.3 merupakan halaman admin. Pada halaman tersebut terdapat lima menu, yaitu: input, transaksi, laporan, data pengunjung, dan logout. Untuk menu input terdapat dua submenu yaitu input koleksi dan input mahasiswa. Untuk menu transaksi terdapat dua submenu yaitu transaksi peminjaman dan transaksi pengembalian. Untuk menu laporan, terdapat dua submenu yaitu laporan peminjaman dan laporang pengembalian. Pada menu data pengunjung berisi daftar mahasiswa yang telah mengunjungi perpustakaan.

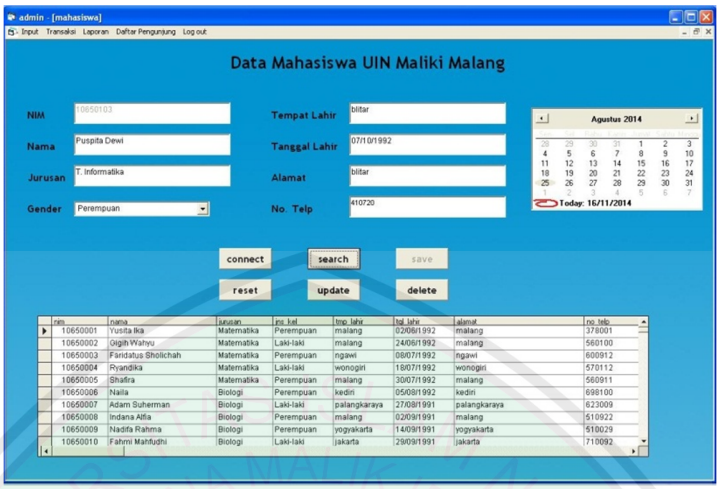
4.1.1.4 Form Halaman Input Koleksi Perpustakaan



Gambar 4.4. Form Halaman Koleksi Perpustakaan

Gambar 4.4 merupakan halaman untuk menginputkan koleksi buku, jurnal, dan skripsi. Pada bagian bawah, terdapat tabel yang secara otomatis akan ter-*update* apabila admin telah menginputkan data dan data tersebut akan langsung muncul pada tabel tersebut.

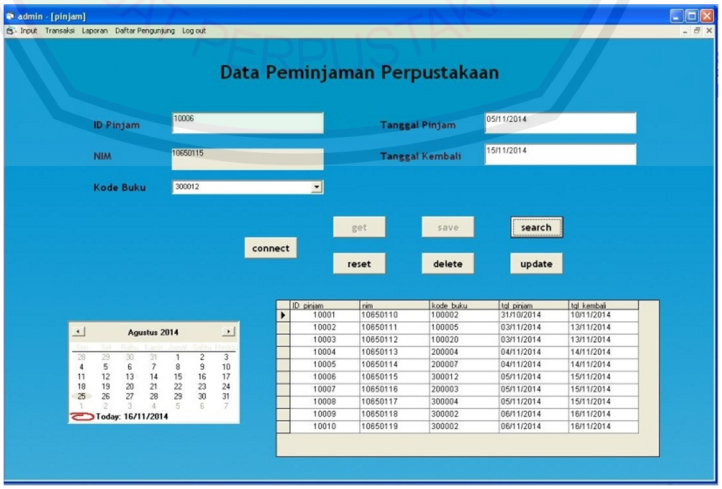
4.1.1.5 Form Halaman Input Data Mahasiswa



Gambar 4.5. Form Halaman Input Data Mahasiswa

Gambar 4.5 merupakan halaman untuk menginputkan data mahasiswa di awal registrasi. Pada halaman tersebut terdapat *button connect* yang berfungsi untuk mengoneksikan *reader smart card* dengan aplikasi. Data yang telah diinputkan admin akan langsung ter-*update* pada tabel dibawahnya.

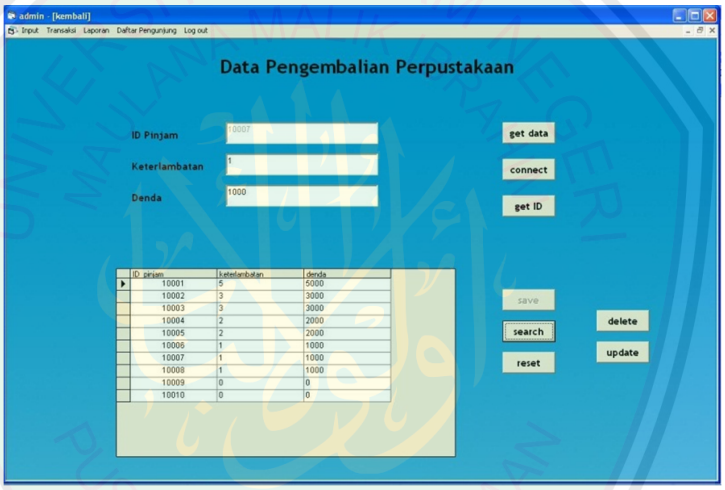
4.1.1.6 Form Halaman Input Data Peminjaman



Gambar 4.6. Form Halaman Input Data Peminjaman

Gambar 4.6 merupakan halaman untuk menginputkan data peminjaman perpustakaan. *Button connect* pada halaman tersebut berfungsi untuk mengoneksikan *reader smart card* dengan aplikasi. Sedangkan *button get* berfungsi untuk membaca data yang ada di *smart card* yaitu data nim, dan data tersebut akan muncul pada *textbox* nim. Kemudian, admin menginputkan semua data lalu disimpan dalam *database* dan *smart card*.

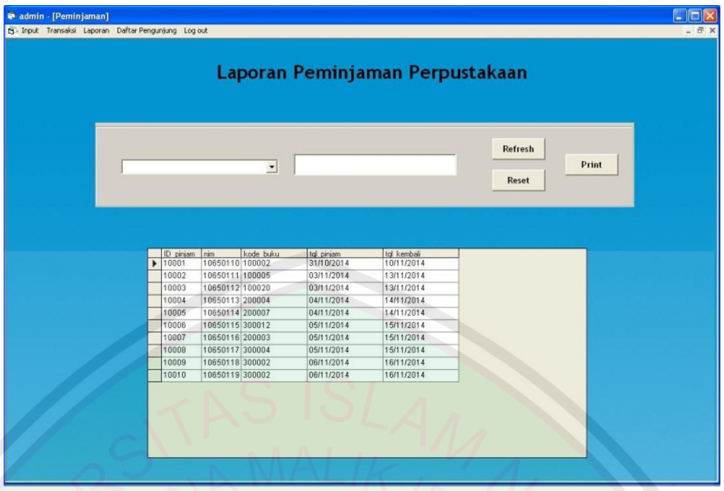
4.1.1.7 Form Halaman Input Data Pengembalian



Gambar 4.7. Form Halaman Input Data Pengembalian

Gambar 4.7 merupakan halaman untuk menginputkan data pengembalian perpustakaan. *Button get* data berfungsi untuk mengambil data ID pinjam pada *smart card*, lalu data tersebut dimunculkan pada *textbox* ID Pinjam pada halaman tersebut dan kemudian disimpan dalam *database*.

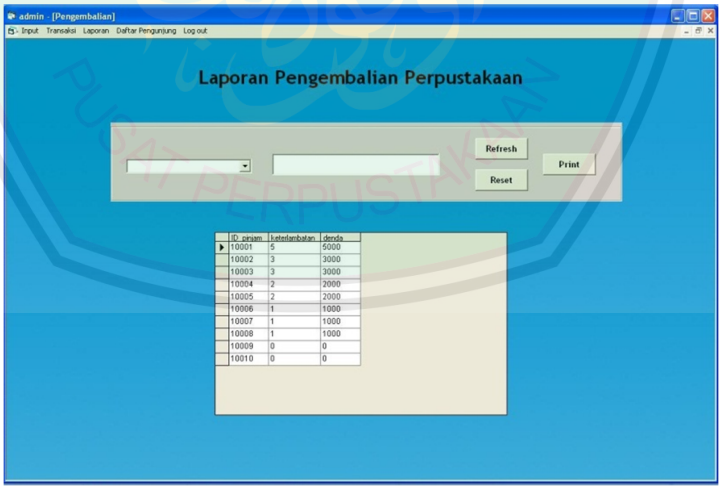
4.1.1.8 Form Halaman Laporan Peminjaman



Gambar 4.8. Form Halaman Laporan Peminjaman

Gambar 4.8 merupakan halaman untuk menunjukkan laporan peminjaman perpustakaan dan hasil pencarian dapat langsung dicetak.

4.1.1.9 Form Halaman Laporan Pengembalian



Gambar 4.9. Form Halaman Laporan Pengembalian

Gambar 4.9 merupakan halaman untuk menunjukkan laporan pengembalian perpustakaan, dan hasil pencarian dapat langsung dicetak.

4.1.1.10 Form Halaman Daftar Pengunjung Perpustakaan

Gambar 4.10. Form Halaman Daftar Pengunjung Perpustakaan

Gambar 4.10 merupakan halaman data pengunjung perpustakaan. Halaman tersebut berfungsi untuk menunjukkan siapa saja pengunjung yang mengunjungi perpustakaan tersebut.

4.2 IMPLEMENTASI ALGORITMA

4.2.1 Tahapan Pembentukan Kunci

Pembentukan kunci merupakan tahap awal sebelum dilakukannya proses enkripsi maupun dekripsi terhadap data yang disimpan di dalam *smart card*. Pertama yaitu inisialisasi *array* S 8 bit sehingga terbentuk *state-array* S dan K, kemudian dilakukan proses KSA dan PRGA.

Pseudo code pembentukan kunci :

```
Inisialisasi Array
For i = 0 To Len(sumber)  do
If i Mod 4 = 0 Then
k(i) = 2
   Else
```

```
k(i) = 5
End If

Proses KSA
Input j As Integer
For i = 0 To Len(sumber2) do
End for

Proses PRGA
Input i, j, As Integer
For a = 0 To Len(sumber2) do
End for
tmp = (s(i) + s(j)) Mod Len(sumber2)
   key(a) = s(tmp)
```

Gambar 4.11. *Pseudo code* pembentukan kunci

4.2.2 Tahapan Proses Enkripsi

Pada proses enkripsi dilakukan proses XOR antara plainteks dan *key* yang telah terbentuk.

```
Proses enkripsi
Input ciphertext(), dtASCII(), dtByte()  As Byte
Input ii As Integer
Input sEnkripsi As String

ii = 0
dtASCII = sumber
ReDim Preserve dtByte(0 To (UBound(dtASCII) / 2) + 1) As Byte
ReDim Preserve ciphertext(0 To (UBound(dtASCII) / 2) + 1) As Byte

For i = 0 To UBound(dtASCII) do
If (i Mod 2) = 0 Then
End if
End for

For j = 0 To UBound(dtByte) do
sEnkripsi = sEnkripsi + Chr(ciphertext(j))
End for
enkripsi = sEnkripsi
```

Gambar 4.12. *Pseudo code* proses enkripsi

4.2.3 Tahapan Proses Dekripsi

Pada proses dekripsi dilakukan proses XOR antara cipherteks dan *key* yang telah terbentuk.

```
Proses dekripsi
Input ciphertext(), dtASCII(), dtByte()  As Byte
Input ii As Integer
Input sEnkripsi As String

ii = 0
dtASCII = sumber3
ReDim Preserve dtByte(0 To UBound(dtASCII) / 2) As Byte
ReDim Preserve ciphertext(0 To UBound(dtASCII) / 2) As Byte

For i = 0 To UBound(dtASCII) do
If (i Mod 2) = 0 Then
End if
End for

For j = 0 To UBound(dtByte) do
sDekripsi = sDekripsi + Chr(ciphertext(j))
End for
dekripsi = sDekripsi
```

Gambar 4.13. *Pseudo code* proses dekripsi

4.2.4 Perhitungan Manual Data Inputan

Inputan berupa NIM : 10650104

Panjang karakter : 8

Kunci awal : 2573

Pertama, inisialisasi *array* S 8 bit sehingga terbentuk *state-array* S dan K sebagai berikut.

Array S

0	1	2	3	4	5	6	7

Array K

2	5	7	3	2	5	7	3

Inisialisasi i dan j dengan 0 kemudian dilakukan KSA agar tercipta *state-array* yang acak. Penjelasan iterasi lebih lanjut dapat dijelaskan sebagai berikut.

Iterasi 1

i = 0

j = (0 + S[0] + K [0 mod 8]) mod 8

= (0 + 0 + 2) mod 8 = 2

Swap (S[0],S[2])

Hasil Array S

2	1	0	3	4	5	6	7

Iterasi 2

i = 1

j = (2 + S[1] + K [1 mod 8]) mod 8

= (2 + 1 + 5) mod 8 =0

Swap (S[1],S[0])

Hasil Array S

1	2	0	3	4	5	6	7

Iterasi 3

i = 2

j = (0 + S[2] + K [2 mod 8]) mod 8

= (0 + 0 + 7) mod 8 = 7

Swap (S[2],S[7])

Hasil Array S

1	2	7	3	4	5	6	0

Iterasi 4

i = 3

j = (3 + S[3] + K [3 mod 8]) mod 8

= (7 + 3 + 3) mod 8 = 5

Swap (S[3],S[5])

Hasil Array S

1	2	7	5	4	3	6	0

Iterasi 5

i = 4

j = (5 + S[4] + K [4 mod 8]) mod 8

= (5 + 4 + 2) mod 8 = 3

Swap (S[4],S[3])

Hasil Array S

1	2	7	4	5	3	6	0

Iterasi 6

i = 5

j = (3 + S[5] + K [5 mod 8]) mod 8

= (3 + 3 + 5) mod 8 =3

Swap (S[5],S[3])

Hasil Array S

1	2	7	3	5	4	6	0

Iterasi 7

i = 6

j = (3 + S[6] + K [6 mod 8]) mod 8

= (3 + 6 + 7) mod 8 = 0

Swap (S[6],S[0])

Hasil Array S

6	2	7	3	5	4	1	0

Iterasi 8

i = 7

j = (0 + S[7] + K [7 mod 8]) mod 8

= (0 + 0 + 3) mod 8 = 3

Swap (S[7],S[3])

Hasil Array S

6	2	7	0	5	4	1	3

Setelah melakukan KSA, akan dilakukan PRGA.

Array S

6	2	7	0	5	4	1	3

Inisialisasi

i = 0

j = 0

Iterasi 1

i = (0 + 1) mod 8 = 1

j = (0 + S[1]) mod 8 = (0+ 2) mod 8 = 2

swap (S[1],S[2])

6	7	2	0	5	4	1	3

K1 = S[(S[1]+S[2]) mod 8]

= S[9 mod 8] = 7

K1 = 0111

Iterasi 2

i = (1 + 1) mod 8 = 2

j = (2 + S[2]) mod 8 = (2+ 2) mod 8 = 4

swap (S[2],S[4])

6	7	5	0	2	4	1	3

K2 = S[(S[2]+S[4]) mod 8]

= S[7 mod 8] = 3

K2 = 0011

Iterasi 3

i = (2 + 1) mod 8 = 3

j = (4 + S[3]) mod 8 = (4+ 0) mod 8 = 4

swap (S[3],S[4])

6	7	5	2	0	4	1	3

K3 = S[(S[3]+S[4]) mod 8]

= S[2 mod 8] = 5

K3 = 0101

Iterasi 4

i = (3 + 1) mod 8 = 4

j = (4 + S[4]) mod 8 = (4+ 0) mod 8 = 4

swap (S[4],S[4])

6	7	5	2	0	4	1	3

K4 = S[(S[4]+S[4])mod 8]

= S[0 mod 8] = 6

K4 = 0110

Iterasi 5

i = (4 + 1) mod 8 = 5

j = (4 + S[5]) mod 8 = (4+ 4) mod 8 = 0

swap (S[5],S[0])

4	7	5	2	0	6	1	3

K5 = S[(S[5]+S[0]) mod 8]

= S[10 mod 8] = 5

K5 = 0101

Iterasi 6

i = (5 + 1) mod 8 = 6

j = (0 + S[6]) mod 8 = (0+ 1) mod 8 = 1

swap (S[6],S[1])

4	1	5	2	0	6	7	3

K6 = S[(S[6]+S[1]) mod 8]

= S[8 mod 8] = 4

K6 = 0100

Iterasi 7

i = (6 + 1) mod 8 = 7

j = (1 + S[7]) mod 8 = (1+ 3) mod 8 = 4

swap (S[7],S[4])

4	1	5	2	3	6	7	0

K7 = S[(S[7]+S[4]) mod 8]
= S[3 mod 8] = 2

K7 = 0010

Iterasi 8

i = (7 + 1) mod 8 = 0

j = (4 + S[8]) mod 8 = (4+ 4) mod 8 = 0

swap (S[0],S[0])

4	1	5	2	3	6	7	0

K4 = S[(S[0]+S[0])mod 8]
= S[0 mod 8] = 4

K4 = 0100

Setelah menemukan kunci untuk tiap karakter, maka dilakukan operasi XOR antara karakter pada plainteks dengan kunci yang dihasilkan. Berikut adalah tabel ASCII untuk tiap-tiap karakter pada plainteks yang digunakan.

Kode ASCII untuk setiap karakter plainteks dapat dilihat pada **Tabel 4.1**

Huruf	**Kode ASCII (Binary 4 bit)**
1	0001
0	0000
6	0110
5	0101

0	0000
1	0001
0	0000
4	0100

Proses pengenkripsian dilakukan dengan proses XOR antara plainteks dengan kunci yang dapat dilihat pada tabel berikut.

Proses XOR kunci enkripsi dengan plainteks dapat dilihat pada **Tabel 4.2**

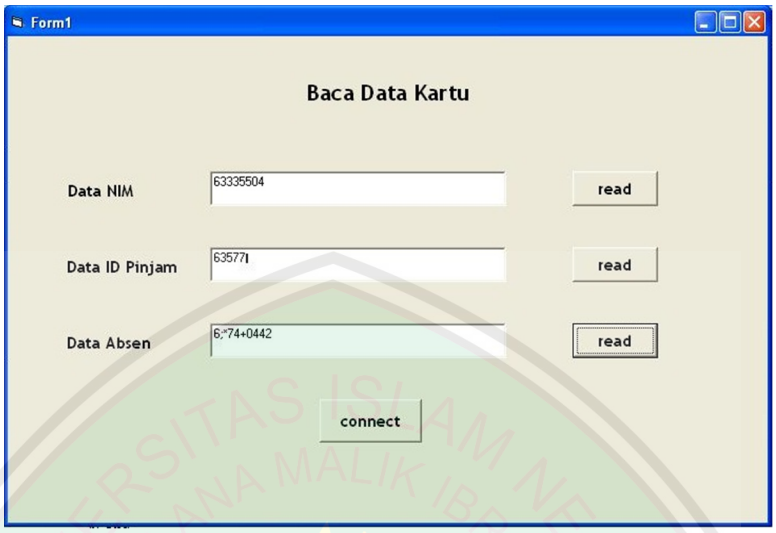
	1	0	6	5	0	1	0	4
Plainteks	0001	0000	0110	0101	0000	0001	0000	0100
Key	0111	0011	0101	0110	0101	0100	0010	0100
Cipherteks	0110 6	0011 3	0011 3	0011 3	0101 5	0101 5	0010 2	0000 0

Proses pendekripsian dilakukan dengan proses XOR antara cipherteks dengan kunci yang dapat dilihat pada tabel berikut.

Proses XOR kunci dekripsi dengan cipherteks dapat dilihat pada **Tabel 4.3**

	6	3	3	3	5	5	2	0
Plainteks	0110	0011	0011	0011	0101	0101	0010	0000
Key	0111	0011	0101	0110	0101	0100	0010	0100
Cipherteks	0001 1	0000 0	0110 6	0101 5	0000 0	0001 1	0000 0	0100 4

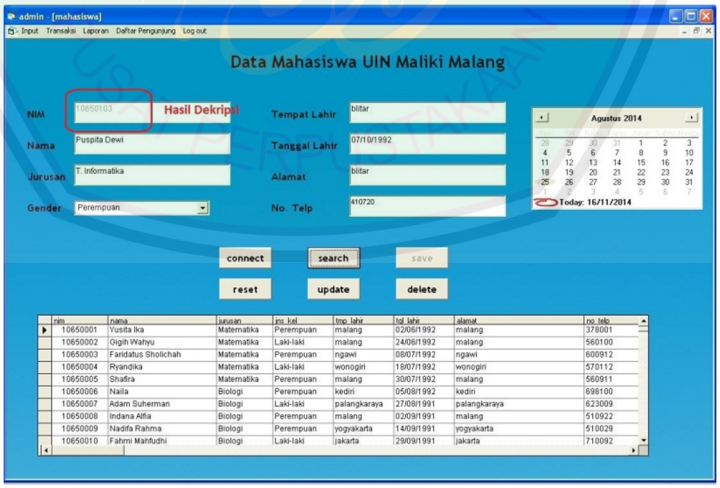
4.2.5 Hasil Enkripsi Data



Gambar 4.14. Hasil enkripsi data

Dari hasil Gambar 4.14 menunjukkan bahwa aplikasi telah berhasil melakukan proses enkripsi dari palinteks menjadi cipherteks.

4.2.6 Hasil Dekripsi Data



Gambar 4.15. Hasil dekripsi data

Dari hasil Gambar 4.15 menunjukkan bahwa aplikasi telah berhasil melakukan proses dekripsi data dari cipherteks ke plainteks.

4.3 PENGUJIAN SISTEM

Pengujian merupakan suatu langkah yang harus dilakukan untuk menjelaskan mengenai pengoperasian perangkat lunak yang telah dibangun. Pengujian tersebut terbagi menjadi beberapa item yang masing-masing item akan diuji.

4.3.1 Rencana Pengujian

Dalam perencanaan pengujian ini, pengujian akan dibagi dalam beberapa item. Dari hasil pengujian tersebut akan didapatkan hasil pengujian yang nantinya dapat ditarik kesimpulan dari pengujian tersebut. Pengujian ini dilakukan dengan tujuan apakah aplikasi yang dibangun sudah berjalan dengan baik dan sesuai dengan yang penulis harapkan.

4.3.2 Pengujian

Pengujian yang dilakukan menitikberatkan pada hasil keluaran dari kendali masukan. Pada kendali masukan, pengujian dikatakan berhasil apabila keluaran sesuai dengan kendali masukan. Berdasarkan rencana pengujian sistem ini yang telah disusun, maka dapat dilakukan pengujian sebagai berikut :

Tabel 4.4. Rencana pengujian

Item Uji	Butir Uji
Mengoneksikan *Card Reader*	Memasukkan kartu ke dalam *card reader*

Memasukkan Data ke Kartu	Menulis data dan disimpan di dalam kartu
Mengambil Data dari Kartu	Membaca data yang ada di dalam kartu
Mengenkripsi Data	Data yang telah dimasukkan dilakukan proses enkripsi
Mendekripsi Data	Data yang ada di dalam kartu dilakukan proses dekripsi untuk dibaca datanya

4.3.2.1 Pengujian Autentifikasi Data pada Smart Card

Hasil pengujian saat data pada smart card dapat dilihat pada **Tabel 4.5**

Tabel 4.5. Hasil uji smart card

Kasus dan Hasil Uji			
Butir Uji	**Hasil yang Diharapkan**	**Hasil Pengamatan**	**Kesimpulan**
Masukkan kartu	Sistem mampu mendeteksi adanya *reader* dan *smart card*	Kartu terkoneksi dengan card reader karena adanya data hexadesimal	[√] sukses [ ] gagal
Masukkan data ke dalam smart card	Menginputkan data ke dalam *smart card*	Data berupa NIM berhasil dimasukkan ke dalam *smart card*	[√] sukses [ ] gagal
Membaca data dari smart card	Membaca data dari *smart card*	Data yang ada di dalam smart card berhasil dibaca	[√] sukses [ ] gagal

Hasil pengujian saat proses enkripsi dan dekripsi data dalam penerapan algoritma RC4 dapat dilihat pada **Tabel 4.6**

Tabel 4.6. Hasil Uji Proses Enkripsi dan Dekripsi Data

Kasus dan Hasil Uji			
Butir Uji	**Hasil yang Diharapkan**	**Hasil Pengamatan**	**Kesimpulan**
Enkripsi data	Data dapat dienkripsi	Data NIM 10650104 berhasil dienkripsi menjadi 63335520	[√] sukses [ ] gagal
Dekripsi data	Data dapat didekripsi	Data 63335520 berhasil didekripsi Menjadi NIM 10650104	[√] sukses [ ] gagal

4.3.2.2 Pengujian Algoritma RC4

Berikut adalah hasil uji algoritma RC4 dengan berbagai inputan :

Tabel 4.7. Hasil Uji Algoritma RC4 dengan Masukan Angka

Isi Pesan	**Panjang Karakter**	**Kunci**	**Hasil**
10650101	8	73565424	63335525
10650102	8	70626416	60076514
10650103	8	32072454	22622557
10650012	8	73565424	63335436
10650119	8	35040523	2561043:
10650058	8	40072661	50622639
10650104	8	51104267	41754366
10650062	8	37043144	27613126

10650016	8	15725743	05175755
10650037	8	60615275	70045242
10650023	8	34051063	24601040
10650049	8	05052506	1560254?
10650075	8	11176777	01726702
10650086	8	71220376	614703?0
10650094	8	57533520	473635;4

Berdasarkan uji coba yang telah dilakukan, percobaan dilakukan dengan masukan angka sebanyak 15 kali. Akurasi yang didapatkan sebesar 100%. Percobaan ini dilakukan dengan masukan angka dengan panjang karakter 8. Kunci yang dihasilkan yaitu angka dengan panjang karakter 8. Keluaran yang dihasilkan yaitu angka pula dengan panjang karakter 8. Sehingga, semua masukan dapat dilakukan proses enkripsi dengan baik dengan menghasilkan keluaran yang memiliki panjang sama denga masukan.

Tabel 4.8. Hasil Uji Algoritma RC4 dengan Masukan Huruf

Isi Pesan	Panjang Karakter	Kunci	Hasil
ANUGRAH WIDYASARI	17	15 6 14 3 4 16 12 12 15 12 15 3 0 8 8 14 0	NH[DVQD,XEKZA[I\
ADE DUROTUN NISA'	17	15 6 14 3 4 16 12 12 15 12 15 3 0 8 8 14 0	NBK#@E^C[YA#NA[O
PUSPITA DEWI EKASARI	20	0 0 11 9 10 1 5 15 7 1 0 14 2 14 2 15 19 10 0 5	PUXYCUD/CDWG"KIN@KRL
NOVI ANTO	9	4 4 2 7 3 6 2 8 5	JKTN#GL\
NAUFAL WAFIQURRAHMA	20	0 0 11 9 10 1 5 15 7 1 0 14 2 14	NA^OKM%XFGI_W\PN[GAK

N		2 15 19 10 0 5	
ZAENAL ABIDIN	13	1 5 8 10 12 3 5 4 7 9 8 9 4	[DMDMO%EE@L@
ALDITA RAHMA MML	16	15 0 3 11 14 13 12 9 9 7 9 2 15 9 12 13	NLGBZL,[HODC/DAA
AGUS FADLUN NI'AM	17	15 6 14 3 4 16 12 12 15 12 15 3 0 8 8 14 0	NA[P$VMHCYA#NA/O
JOEHANI ABDILLAH ASLAMI	23	8 10 16 13 11 13 10 11 16 2 8 16 10 3 15 16 9 5 6 4 5 12 15	BEUEJCC+Q@LYFONX) DUHDA
AFIF SUBARKAH	13	1 5 8 10 12 3 5 4 7 9 8 9 4	@CAL,PPFF[CH
SYAFE'I KARIM	13	1 5 8 10 12 3 5 4 7 9 8 9 4	R\ILI$L$LHZ@
ALIF AKBARUL MUSLIM	19	8 5 16 1 4 4 17 2 5 13 3 3 15 11 4 4 13 10 0	IIYG$EZ@D_VO/FQWAC
VIVID ROHMANIYAH	16	2 10 6 7 13 9 3 5 8 1 10 4 2 2 6 12	TCPNI)QJ@LKJK[GD
PUSPA MILADIN NURAIDA SAFITRI AB	32	9 2 2 9 25 7 24 25 20 31 17 9 8 26 28 30 29 26 23 28 6 6 9 18 19 1 29 16 10 16 26 4	YWQYX'UPX^U@F:RKO[^XG&ZSUHIBC0[F
GERY ARIES SUKMA WIBAWA	23	16 20 5 8 9 12 22 12 13 6 3 11 18 11 4 12 17 5 8 11 19 20 7	WQWQ)MDEHU#XG@IM 1RAIRC

Berdasarkan uji coba yang telah dilakukan, percobaan dilakukan dengan masukan huruf sebanyak 15 kali. Akurasi yang didapatkan sebesar 80%. Percobaan ini dilakukan dengan masukan huruf dengan panjang karakter tertentu. Kunci yang

dihasilkan yaitu angka dengan panjang karakter yang sesuai dengan panjang masukan. Namun, terdapat beberapa keluaran yang dihasilkan tidak sama dengan masukan dan panjang kunci. Sehingga, tidak semua masukan dapat dilakukan proses enkripsi dengan baik.

Tabel 4.9. Hasil Uji Algoritma RC4 dengan Masukan Berbagai Karakter

Isi Pesan	Panjang Karakter	Kunci	Hasil
(A+B)(A-B)=	11	8 10 1 0 5 4 9 8 5 6 8	K*B,,H%G/
QW!E@R	6	5 4 3 3 0 2	TS"F@P
RT$%YU^IO&*	11	8 10 1 0 5 4 9 8 5 6 8	Z^%%\QWAJ
IO*(P)LK=JH_	12	10 5 7 0 11 4 11 11 1 11 5 2	CJ-([-G@<AM]
CJ-([-G@<AM]	11	8 10 1 0 5 4 9 8 5 6 8	@MG=.@OLV]
AS{}ZX,.??	10	1 7 2 4 5 4 9 9 3 0	@Tyy_\%'<?
VC\}[,JKL><	11	8 10 1 0 5 4 9 8 5 6 8	^I]}^(CCI8
MNB%&?ER/	9	8 7 3 5 5 5 6 8 4	EIA #:CZ
FGH%^(VFW\]{u	13	9 0 6 2 12 0 11 11 0 6 4 3 0	OGN'R(]MWZYx
OJN*>+WDB#":BU	14	1 8 0 8 8 8 11 8 9 7 10 0 8 8	NBN"6#\LK$(:J]

Berdasarkan uji coba yang telah dilakukan, percobaan dilakukan dengan masukan kombinasi berbagai karakter sebanyak 10 kali. Akurasi yang didapatkan sebesar 60%. Percobaan ini dilakukan dengan masukan berbagai dengan panjang tertentu. Kunci yang dihasilkan yaitu angka dengan panjang karakter yang sesuai

dengan panjang masukan. Namun, terdapat beberapa keluaran yang dihasilkan tidak sama dengan masukan dan panjang kunci. Sehingga, tidak semua masukan dapat dilakukan proses enkripsi dengan baik. Dari ketiga hasil uji coba di atas, maka didapatkan hasil rata-rata prosentase keakurasian sebesar 80%.

4.4 INTEGRASI KRIPTOGRAFI MENURUT KAJIAN DALAM AL-QUR'AN

Kriptografi diartikan sebagai ilmu atau seni untuk menjaga keamanan pesan. Ketika suatu data dikirim dari suatu tempat ke tempat lain, isi data tersebut mungkin dapat disadap oleh pihak lain yang tidak berhak untuk mengetahui isi pesan tersebut. Untuk menjaga data, maka data tersebut dapat diubah menjadi suatu kode yang tidak dapat dimengerti oleh pihak lain. Enkripsi adalah sebuah proses penyandian yang melakukan perubahan sebuah kode dari yang bisa dimengerti (*plainteks*) menjadi sebuah kode yang tidak bisa dimengerti (*cipherteks*). Sedangkan proses kebalikannya untuk mengubah cipherteks menjadi plainteks disebut dekripsi. Proses enkripsi dan dekripsi memerlukan suatu mekanisme dan kunci tertentu.

Di dalam Al-Quran disebutkan bahwa Allah SWT memelihara Al-Qur'an dari upaya syetan yang ingin menambahkan kebatilan ke dalamnya dan mengurangi kebenarannya, sehingga Al-Qur'an tetap terpelihara. Maksudnya adalah memeliharanya dari penyimpangan, baik huruf maupun makna, dan penambahan maupun pengurangan. Allah SWT menjadikan Al-Qur'an sebagai suatu keajaiban

(mukjizat), guna membedakan apa yang tertera padanya dengan ucapan manusia. Dalam konteks ini terdapat kesinambungan antara kriptografi dengan ajaran agama Islam, menjaga keamanan data dalam kriptografi sama halnya menjaga Al-Quran dari upaya setan yang ingin mengubah isi dari Al-Quran. Selain dalam al-Quran, di dalam hadits juga disebutkan bahwa orang yang hafal al-Quran juga ikut bertugas untuk menjaga isi dari al-Quran .

Pernyataan ini sesuai dengan hadits yang diriwayatkan oleh Muslim yang berbunyi :

و حَدَّثَنَا زُهَيْرُ بْنُ حَرْبٍ وَعُثْمَانُ بْنُ أَبِي شَيْبَةَ وَإِسْحَقُ بْنُ إِبْرَاهِيمَ قَالَ
إِسْحَقُ أَخْبَرَنَا وَقَالَ الْآخَرَانِ حَدَّثَنَا جَرِيرٌ عَنْ مَنْصُورٍ عَنْ أَبِي وَائِلٍ
عَنْ عَبْدِ اللَّهِ قَالَ قَالَ رَسُولُ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ بِئْسَمَا لِأَحَدِهِمْ
يَقُولُ نَسِيتُ آيَةَ كَيْتَ وَكَيْتَ بَلْ هُوَ نُسِّيَ اسْتَذْكِرُوا الْقُرْآنَ فَلَهُوَ أَشَدُّ
تَفَصِّيًا مِنْ صُدُورِ الرِّجَالِ مِنْ النَّعَمِ بِعُقُلِهَا

Hadis riwayat Abdullah bin Masud ra., ia berkata:
Alangkah buruknya seorang yg berkata, 'Aku lupa ayat ini & itu.' Akan tetapi ia telah dilupakan Allah. Sering-seringlah mengingat (membaca) Al Qur`an, karena ia lebih cepat hilangnya dari dada seseorang, daripada unta yg lepas dari ikatannya. [HR. Muslim No.1314].

Hasil yang diperoleh dari skripsi ini adalah perangkat lunak yang dibangun telah berhasil mengenkripsi data sesuai dengan aturan dalam metode RC4 pada kriptografi. Sehingga perangkat lunak ini dapat digunakan untuk merahasiakan sebuah data, dan telah sesuai dengan tujuan yang diharapkan dari pembangunan perangkat lunak ini yaitu membuat aplikasi *smart card* menggunakan metode RC4

untuk proses pengenkripsiannya. Kaitannya dengan keislaman dalam menjaga Al-Quran yang merupakan firman Allah sudah sesuai dengan yang telah diharapkan. Metode RC4 dapat bekerja dengan baik dalam mengenkripsi sebuah data dan hal ini sama dengan Al-Quran yang telah dijaga oleh Allah dari para setan yang ingin mengubah isinya.

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM PUSAT PERPUSTAKAAN

BABA V

PENUTUP

5.1 KESIMPULAN

Dari hasil implementasi dan uji coba yang telah diteliti dapat disimpulkan bahwa :

1. Algoritma RC4 dapat digunakan dalam pembuatan aplikasi *smart card* layanan transaksi peminjaman buku perpustakaan. Algoritma RC4 tersebut digunakan sebagai metode untuk proses enkripsi dan dekripsi data dalam *smart card*.
2. Kunci hasil pembentukan pada algoritma RC4 memiliki panjang karakter yang sama dengan panjang karakter masukan data.
3. Berdasarkan hasil pengujian algoritma RC4 dengan masukan berupa angka, huruf, dan kombinasi berbagai karakter menghasilkan nilai akurasi rata-rata sebesar 80%.

5.2 SARAN

Beberapa saran untuk penelitian dan pengembangan aplikasi selanjutnya adalah sebagai berikut :

1. Dapat digunakan algoritma lain untuk proses enkripsi dan dekripsi data. Selain algoritma simetris, algoritma asimetris juga dapat digunakan untuk mendapatkan hasil yang lebih variatif.
2. Perlu dilakukan pengembangan pada manajemen pengetahuan sehingga basis pengetahuan dapat di-*update* dengan metode yang lebih dinamis.

DAFTAR PUSTAKA

Ad-Dimasyqi, Al-Imam Ibnu Kasir. 2000. *Tafsir Ibnu Kasir.*

Al-Qur'anul Karim. 2006. *Al-Quran Tajwid dan Terjemahnya*. Bandung

Ariyanto, Yuri. 2009. *Algoritma RC4 dalam Proteksi Transmisi dan Hasil Query untuk ORDBMS PostgreSQL*. Jurusan Teknik Informatika, Fakultas Teknologi Informasi.

Astrianto, Stefanus. 2008. *Pembangunan Perangkat Lunak untuk Security pada Contactless Smart Card dengan Algoritma RC4.* Institut Teknologi Bandung.

Dwi Ananda, Yanno. 2010. *Rancang Bangun Aplikasi Smart Card Interface.* Teknik Informatika, Sekolah Tinggi Ilmu Teknik Jembrana.

Fauzan, Muhammad. 2012. *Implementasi Algoritma Kriptografi RC4 pada DSP TM320C6713 Sebagai Pendukung Sekuritas Jaringan Komunikasi Voice Over Internet Protocol(VoIP)*. Jurnal EECCIS.

Firmansyah, Eka Risky. 2012. *Algoritma Kriptografi*. UIN Syarif Hidayatullah, Jakarta.

Gintoro,dkk. 2005. *Sistem Smart Card Room Berbasis Smart Card dan Bahasa Pemrograman C++.*

http://kartuchip.com/chip_fm4428_sle4428.html / Diakses pada 6 Oktober 2014.

Nurhayati, Oky Dwi. 2009. *Keamanan Multimedia*. Universitas Diponegoro, Semarang.

Pamungkas, Dedy. 2005. *Aplikasi Smart Card Sebagai Kartu Pra Bayar Internet.* Teknik Elektro Universitas Diponegoro Semarang.

Rusdinar, Angga. 2008. *Perancangan Sisten Kartu Identitas Berbasis Smart Card.* Departemen Teknik Elektro, Institut Teknologi Telkom.

Sariasih, Christine. 2009. *Rancangan Keamanan Data Sistem Smart Card Kesehatan Sesuai Kebutuhan di Indonesia.* Fakultas Ilmu Komputer Universitas Indonesia.

Sarinanto, Mustafa. 2003. *Pengembangan Program Smart Card pada Rumah Sakit.* Universitas Gunadarma Jakarta.

Supriyono. 2009. *Penerapan Aplikasi RFID di Bidang Perpustakaan.*

Suryani, Karina Novita. 2009. *Algoritma RC4 Sebagai Metode Enkripsi.* Sekolah Tinggi Elektro dan Informatika ITB.

Wahyu, Febrian dkk. 2012. *Penerapan Algoritma Gabungan RC4 dan Base64 pada Sistem Keamanan E-Commerce*. Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana.

www.acr38u.com / Diakses pada 6 Oktober 2014

Yefriadi. 2010. *Aplikasi Smart Card untuk Electronic Medical Record (EMR)*. Jurusan Teknik Elektro, Politeknik Negeri Padang.

Yendri, Dodon. 2008. *Rancangan Model Layanan Transaksi Peminjaman Buku Menggunakan Teknologi Smart Card pada Perpustakaan Induk.* Universitas Andalas Padang.