

# RANCANG BANGUN APLIKASI SMART CARD INTERFACE

I Putu Agus Swastika \*, Siti Saibah Pua Luka \*, Yanno Dwi Ananda \*\*

\*) Staff Pengajar PS. Teknik Informatika, Sekolah Tinggi Ilmu Teknik Jembrana

\*\*) Mahasiswa PS. Teknik Informatika, Sekolah Tinggi Ilmu Teknik Jembrana

## Abstrak

*Peluang pengembangan aplikasi berbasis smart card cukup besar seiring kebutuhan akan teknologi smart card di berbagai bidang baik dunia bisnis maupun instansi pemerintahan dan BUMN. Salah satu merek smart card yang banyak digunakan adalah smart card jenis ACR122U NFC produksi dari vendor ACS(Advanced Card System Limited) karena harganya relatif terjangkau namun cukup tangguh .*

*Saat tugas akhir ini dikerjakan, ACS(Advanced Card System Limited) sebagai vendor produsen smart card jenis ACR122U NFC baru mendukung beberapa bahasa pemrograman untuk pengembangannya, antara lain Borland Delphi 7, Java, Microsoft Visual Basic 6.0, Microsoft Visual Basic.NET 2005, Microsoft Visual C#.2005, Microsoft Visual C++ 6.0 Microsoft Visual C++ 2005. Lalu bagaimana dengan bahasa pemrograman lain?, hal tersebut tentu akan menjadi kendala bagi pengembang aplikasi smart card untuk mengembangkan aplikasi berbasis smart card ACR 122U NFC, terutama bagi pengembang yang kompetensi bahasa pemrograman yang dimiliki diluar bahasa pemrograman yang didukung ACR 122U NFC saat ini. Kendala tersebut terutama kompatibilitas serta waktu pengembangan akan relatif lebih lama.*

*Untuk itu pengembangan aplikasi interface (penghubung) antara PC dengan Smart Card yang fleksibel, yang mampu mengkomunikasikan smart card, smart card reader dengan aplikasi yang akan dikembangkan dengan berbagai bahasa pemrograman (multiprogramming) yang berbeda akan membantu kemudahan dalam pengembangan aplikasi-aplikasi lain berbasis smart card ACR 122U.*

**Kata kunci:** smart card, smart card reader, ACR 122U, interface, multiprogramming.

## Abstract

*Opportunity to development smart card -based application is quite large as the need for smart card technology in various fields of both business and government agencies and BUMN. One brand that is widely used smart card is a smart card type from a vendor production NFC ACR122U ACS (Advanced Card System Limited) because prices are relatively affordable, but fairly tough.*

*At the Thesis is done, the ACS (Advanced Card System Limited) as a manufacturer of smart card vendor type ACR122U NFC only supports several programming languages for development, among other Borland Delphi 7, Java, Microsoft Visual Basic 6.0, Microsoft Visual Basic.NET 2005, Microsoft .2005 Visual C #, Microsoft Visual C + + 6.0 Microsoft Visual C + + 2005. And what about other programming languages?, It will certainly be an obstacle for smart card application developers to develop applications based NFC smart cards 122U ACR, especially for developers who owned the programming language competence beyond the programming language that supported the current ACR NFC 122U. Constraints are primarily compatibility as well as the development time will be relatively longer.*

*For that application development interface between a PC with Smart Card is flexible, able to communicate the smart card, smart card reader with the application to be developed*

*with various programming languages (multiprogramming) that differ will help ease in the development of other applications based on smart ACR card 122U.*

**Key words:** *smart card, smart card reader, ACR 122U, interfaces, multiprogramming.*

## 1. Pendahuluan

Salah satu tuntutan yang dihadapi dunia bisnis maupun institusi pelayanan publik saat ini adalah menciptakan sistem pelayanan publik yang memberikan kemudahan dalam tukar-menukar informasi, transaksi secara cepat mudah, dan transparan, serta memiliki mobilitas tinggi. Hal tersebut memegang peranan penting dalam kelangsungan hidup bisnis maupun keefektifan kinerja institusi karena berpengaruh terhadap mutu produk atau pelayanan yang diberikan.

*Smart card* diciptakan untuk menjadi solusi bagi problem tersebut. Teknologi ini menawarkan banyak manfaat signifikan bagi para penyedia dan pengguna jasa. Mobilitas tinggi didapatkan dari ukuran fisik yang kecil dengan dimensi chip hanya 85,6 mm x 54 mm. Kestabilan dan kecepatan dapat dioptimalkan dengan makin banyaknya bahasa pemrograman yang mendukungnya. Pemanfaatan *smart card* dapat diaplikasikan diberbagai bidang baik itu pendidikan, kesehatan, pelayanan umum, keamanan komputer, transaksi keuangan dan sebagainya. Ruang pemanfaatan *smart card* yang mencakup bidang yang sangat luas tersebut akan memberikan peluang besar bagi pengembang aplikasi untuk mengembangkan aplikasi-aplikasi berbasis *smart card* di berbagai sektor yang belum menggunakan teknologi tersebut.

Permasalahan yang kemudian timbul adalah fleksibilitas dan kemudahan untuk pengembangan aplikasi berbasis *smart card*. Para pengembang aplikasi yang ingin mengembangkan aplikasi berbasis *smart card* yang sebelumnya belum pernah mengembangkan aplikasi berbasis *smart card* akan membutuhkan waktu untuk mempelajari teknik pemrograman untuk menghubungkan antara aplikasi yang akan di buatnya dengan sebuah perangkat *smart card*. Masing-masing bahasa pemrograman tentunya memiliki perintah program yang berbeda untuk mengembangkan aplikasi berbasis smart card. Sehingga akan membutuhkan waktu yang cukup untuk mempelajarinya terutama yang belum menguasai bahasa yang mendukung pemrograman *smart card* ACR122U seperti Borland Delphi 7, Java, Microsoft Visual Basic 6.0, Microsoft Visual Basic.NET 2005, Microsoft Visual C#.2005, Microsoft Visual C++ 6.0 Microsoft Visual C++ 2005, sehingga pengembangan aplikasi berbasis *smart card* akan membutuhkan waktu yang lebih panjang apalagi jika kompetensi pengembang adalah dibidang bahasa pemrograman yang tidak atau belum mendukung pemrograman *smart card* ACR 122U tentunya akan kesulitan untuk mengembangkan aplikasi yang melibatkan *smart card* dengan kompetensi bahasa pemrograman yang dimilikinya tersebut.

Untuk itu pengembangan aplikasi *interface* (penghubung) antara PC dengan Smart Card yang fleksibel, yang mampu mengkomunikasikan *smart card*, *smart card reader* dengan aplikasi yang akan dikembangkan dengan berbagai bahasa pemrograman yang berbeda akan membantu kemudahan dalam pengembangan aplikasi-aplikasi lain berbasis *smart card*.

## 2. Tinjauan Pustaka

### 2.1. Smart Card

*Smart card*, sering disebut pula sebagai *chip card* atau *integrated circuit(s) card(ICC)*, dapat didefinisikan sebagai kartu seukuran kantong (dapat lebih kecil lagi) dengan *integrated circuit* yang *embedded* dengannya. Ada dua jenis ICCs, yaitu *memory card* dan *microprocessor card*. *Memory card* hanya terdiri dari *non volatile memory storage* dan mungkin terdiri dari fitur keamanan pula. Sedangkan *memory card* terdiri dari *memory* dan komponen *microprocessor*.

*Smart card* dapat dikategorikan dalam dua jenis, yaitu :

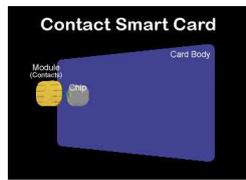
**A. Contact Smart card**

*Contact smart card* memiliki sebuah chip emas yang berukuran sekitar 0.5 inchi di bagian depan, tidak seperti kartu kredit yang memiliki *magnetic strip* di bagian belakang. *Contact Smart card* membutuhkan aplikasi *smart card reader* untuk membaca dan menulis data dari dan ke dalam chip tersebut.

Standar PIN koneksi berdasarkan ISO7816:



Gambar 1. Standar PIN *Contact Smart Card*



Gambar 2. Bentuk *Contact Smart Card*

**B. Contactless Smart card**

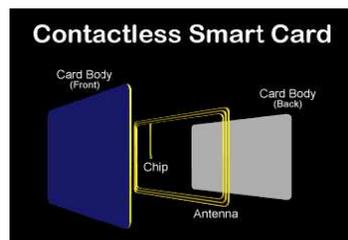
*Contactless Smart card* tampak seperti kartu kredit plastik dengan chip computer dan *antenna coil* di dalamnya. *Contactless smart card* dapat ditulis dan dibaca hanya dengan didekatkan pada antenna luar. *Contactless Smart card* digunakan bila membutuhkan transaksi yang harus diproses dengan cepat.

Dua kategori tambahan lainnya merupakan turunan dari kedua tipe yang telah dijelaskan sebelumnya. Kedua kategori tersebut adalah *Combi Card* dan *Hybrid Card*.

*Hybrid card* memiliki dua chips, masing-masing merepresentasikan antarmuka *contact* dan *contactless*. Kedua chip tersebut tidak saling berhubungan, tetapi untuk sebagian besar aplikasi digunakan secara bersamaan dalam melayani kebutuhan *consumer* dan *card issuer*. Berbeda dengan *Hybrid Card*, *Combi card* hanya memiliki sebuah chip yang merepresentasikan antarmuka *contact* dan *contactless*.

Chip yang digunakan pada kedua jenis kategori tersebut di atas dapat dikategorikan ke dalam dua bagian, yaitu: chip *microprocessor* dan chip *memory*.

Chip *memory* dapat dilihat sebagai *floppy disk* kecil dengan layanan keamanan(*optional*). *Memory card* dapat menyimpan 103 hingga 16000 bits data. *Memory card* lebih murah bila dibandingkan dengan *chip microprocessor*, hanya saja fasilitas keamanan yang dimiliki pun lebih sedikit. Keamanan *memory card* bergantung pada keamanan yang diberikan *card reader* saat pemrosesan data. Chip *microprocessor* dapat menambahkan, menghapus, ataupun memanipulasi informasi yang tersimpan dalam *memory*. Chip *microprocessor* dapat dianggap sebagai *miniature computer* dengan *input/ouput port*, sistem operasi, dan *hard disk*.



Gambar 3. Bentuk *Contactless Smart Card*

Standar Internasional untuk protokol yang paling banyak digunakan dalam teknologi *smart card* ini adalah ISO 7816. Meskipun demikian, terdapat pula beberapa standar lain yang digunakan. *Smart card* menyimpan dan memproses informasi melalui rangkaian elektronik yang ada di dalam silikon yang dalam substrat plastik dari *body*-nya. Terdapat dua jenis *smart card* yang paling banyak digunakan yaitu *intelligent smart card* yang terdiri atas sebuah mikroprosesor dan mampu membaca, menulis, dan menghitung, seperti halnya sebuah mikrokomputer. Serta *memory card*, tidak memiliki mikroprosesor dan hanya digunakan untuk menyimpan informasi saja. *memory card* menggunakan *security logic* untuk mengatur akses dari memori. Meskipun demikian, sesungguhnya terdapat lima buah jenis *smart card* yang ada saat ini yaitu :

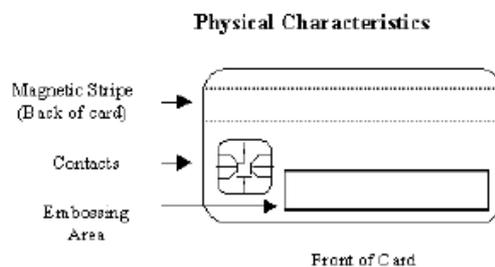
- *memory card*
- *processor card*
- *electronic purse card*
- *security card*
- *JavaCard*

Semua *smart card* berisi tiga tipe memori yaitu :

- *persistent non-mutable memory*;
- *persistent mutable memory*;
- dan *non-persistent mutable memory*.

ROM, EEPROM, dan RAM merupakan memori yang paling banyak digunakan untuk ketiga tipe memori tersebut .

Karakter fisik dari *smart card* berdasar standar ISO 7816 adalah :



Gambar 4. Karakter fisik dari smart card berdasar standar ISO 7816

Normalnya, sebuah *smart card* tidak memiliki *power supply*, *display*, atau *keyboard*. *Smart card* berinteraksi dengan dunia luar menggunakan *serial communication interface* via delapan *contact point* yang ada.

Sebuah *smart card* dimasukkan ke dalam *Card Acceptance Device (CAD)*, yang memungkinkan untuk berhubungan dengan komputer lain. Bentuk lain dari penggunaan *Card Acceptance Device* adalah *terminal*, *reader*, dan IFD (*interface device*). Perangkat-perangkat tersebut menyediakan berbagai fungsi dasar yang sama, diantaranya untuk men-supply smart card dengan power dan untuk menghasilkan koneksi untuk pertukaran data.

Ketika dua buah komputer berkomunikasi satu dengan yang lain, akan terjadi pertukaran paket data yang disusun berdasarkan sekumpulan protokol tertentu. Demikian pula smart card, berkomunikasi dengan dunia luar menggunakan paket data tersendiri yang dinamakan APDU (*Application Protocol Data Units*). APDU berisi baik perintah-perintah maupun pesan respon. Dalam *smart card* tersebut, digunakan model *masterslave* dimana *smart card* sendiri berperan secara pasif sebagai slave. smart card selalu menunggu perintah APDU dari sebuah terminal. Selanjutnya perintah tersebut dieksekusi dan dikembalikan ke terminal sebagai respon APDU. Perintah APDU dan respon APDU tersebut dipertukarkan secara antara *card* dan terminal.

Untuk membangun sebuah aplikasi *smart card* diperlukan beberapa komponen yaitu : *smart card reader*, *software* untuk berkomunikasi dengan *reader* dan *card* dipasangkan pada *reader*, dan *smart card* dan *smartcard hardware*.

## 2.2. XML

*Extensible Markup Language* (XML) merupakan sebuah *toolkit* penyimpanan data (*data storage toolkit*), sebuah kendaraan yang dapat dikonfigurasi untuk semua jenis informasi, sebuah pengembangan dan *open standard* yang digunakan oleh semua orang mulai dari *banker* sampai *webmaster*. Dalam beberapa tahun ini, XML diterapkan dan diadopsi secara luas oleh dunia industri. Hal ini dikarenakan *feature-feature* yang dimilikinya. Fitur-fitur yang ditawarkan XML :

- XML dapat menyimpan dan mengorganisir semua jenis informasi dalam bentuk yang kita suka (dapat disesuaikan dengan kebutuhan)
- Sebagai sebuah *open standard*, XML tidak terikat dengan perusahaan atau perangkat lunak manapun.
- Dengan *Unicode* sebagai karakter set standar, XML mendukung berbagai macam system penulisan (*scripts*) dan simbol. dari karakter Skandinavia sampai ideograf bangsa China Han.
- XML menawarkan berbagai cara untuk memeriksa kualitas sebuah dokumen, dengan aturan *syntax*, *internal link checking*, perbandingan dengan modul dokumen, dan *datatyping*.
- *Syntax* XML sederhana dan tidak mempunyai struktur yang ambigu. Sehingga mudah dibaca oleh manusia maupun program.
- XML mudah untuk dikombinasikan dengan *stylesheet* untuk membuat format dokumen sesuai dengan *style* yang kita inginkan.
- Sebuah file teks (kadang-kadang dieja "textfile": sebuah nama alternatif tua adalah "flatfile") adalah jenis file komputer yang terstruktur sebagai urutan garis . Sebuah file teks ada dalam sebuah sistem file komputer . Akhir dari file teks seringkali dinotasikan dengan menempatkan satu atau lebih karakter khusus, dikenal sebagai end-of-file marker, setelah baris terakhir dalam sebuah file teks.
- "File teks" mengacu pada jenis wadah, sementara teks biasa mengacu pada jenis konten. File teks dapat berisi teks biasa, tetapi mereka tidak terbatas pada hal tersebut.

## 2.3. File Teks

Sebuah file teks (kadang-kadang dieja "textfile": sebuah nama alternatif tua adalah "flatfile") adalah jenis file komputer yang terstruktur sebagai urutan garis . Sebuah file teks ada dalam sebuah sistem file komputer . Akhir dari file teks seringkali dinotasikan dengan menempatkan satu atau lebih karakter khusus, dikenal sebagai end-of-file marker, setelah baris terakhir dalam sebuah file teks.

"File teks" mengacu pada jenis wadah, sementara teks biasa mengacu pada jenis konten. File teks dapat berisi teks biasa, tetapi mereka tidak terbatas pada hal tersebut.

## 2.4. File Properties

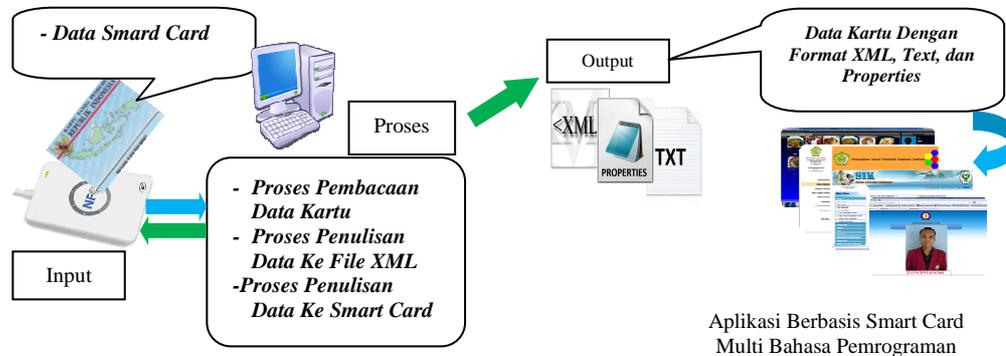
.properties adalah ekstensi file untuk file terutama digunakan di teknologi Java yang terkait untuk menyimpan parameter dikonfigurasi dari sebuah aplikasi . Mereka juga dapat digunakan untuk menyimpan string untuk Internasionalisasi dan lokalisasi , ini dikenal sebagai Kumpulan Sumber Daya Properti.

Setiap parameter disimpan sebagai sepasang string , salah satu menyimpan nama parameter (disebut kunci), dan yang lainnya menyimpan nilai.

### 3. Pemodelan Sistem

#### 3.1 Arsitektur Sistem

Gambaran mengenai kinerja pada Aplikasi Smart Card Interface dapat dilihat pada gambar dibawah ini:



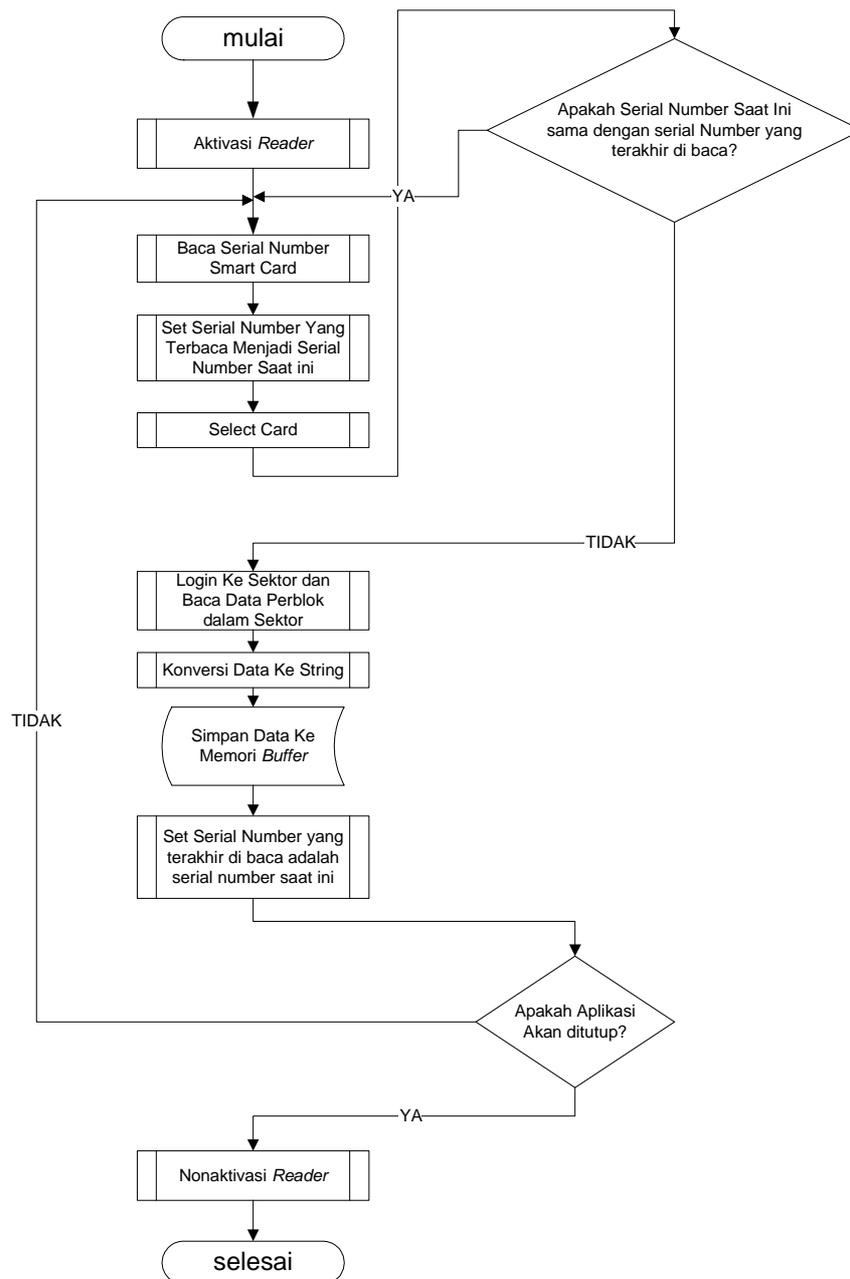
Gambar 5. Arsitektur Sistem Aplikasi Smart Card Interface.

Pada gambar diatas dapat dilihat mengenai alur proses dari jalannya Aplikasi Smart Card Interface, dimana inputannya berupa data smart card yang dibaca dari smart card oleh Aplikasi Smart Card Interface. Data inputan kemudian diproses oleh Aplikasi Smart Card Interface kemudian mem-parsing data hasil pembacaan smart card yang output-nya dalam bentuk file xml, text dan properties dengan tujuan agar data dapat dibaca bahasa pemrograman yang mendukung pembacaan data berformat xml, text, atau properties dan pada kenyataannya tipe data tersebut hampir didukung oleh semua bahasa pemrograman, data-data inilah yang nantinya akan digunakan untuk mengembangkan aplikasi-aplikasi berbasis smart card tanpa membuat program lagi untuk membaca data dari smart card. Selain melakukan pembacaan data dari smart card dan mem-parsing data ke xml, text, dan properties, aplikasi juga akan melakukan penulisan data ke smart card.

#### 3.2 Sistem flowchart

##### A. Flowchart Pembacaan data dari smart card

Gambar di bawah menunjukkan aliran mengenai proses yang terjadi pada untuk pembacaan data dari smart card. Aplikasi Smart Card interface ini akan membaca data yang tersimpan dalam chip smart card sesuai spesifikasi yang distandarkan pada smart card reader tipe ACR 122U kemudian hasil pembacaan kartu ini akan disimpan dalam sebuah data model di memori buffer.

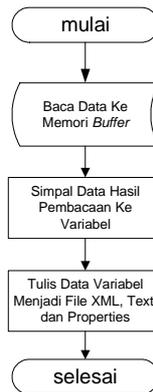


Gambar 6. System Flowchart Pembacaan data dari smart card.

Setelah aplikasi berhasil membaca data yang terdapat pada chips smart card dan menyimpannya ke dalam sebuah data model di memori buffer proses selanjutnya adalah mem-parsing atau menyimpannya kedalam format file yang fleksibel dan portable yang artinya file tersebut nantinya dapat di akses atau dibaca dengan multi bahasa pemrograman komputer dimana dalam hal ini file yang dihasilkan bertipe XML, Text, dan Properties.

### B. Flowchart Penulisan data ke Data XML, Text, dan Properties

Gambar di di bawah menunjukkan proses penulisan data ke data xml, text, dan properties. Dalam proses ini sumber data berasal dari data hasil pembacaan smart card yang sebelumnya pada proses pembacaan data smart card telah disimpan di memori buffer.



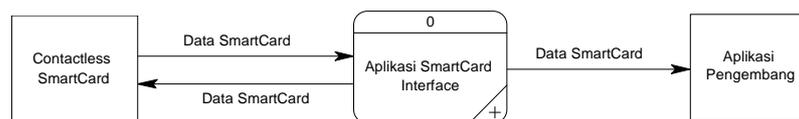
Gambar 8. System Flowchart Penulisan data ke Data XML, Text, dan Properties.

Data tersebut kemudian dibaca kembali dan ditampung ke dalam sebuah variable untuk kemudian menjadi data yang dituliskan dalam bentuk file berjenis xml, text, dan properties. File-file diataslah yang kemudian dapat dibaca oleh aplikasi pengembang(aplikasi berbasis smart card ACR122U NFC), sehingga pengembang tidak direpotkan lagi dengan urusan pembacaan data dari smart card. Di pilih file berjenis tersebut karena sebagian bahasa pemrograman mendukung jenis tersebut.

### 3.3 Data Flow Diagram (DFD)

Dalam proses pengembangan sebuah aplikasi dibutuhkan perencanaan terlebih dahulu. Hal ini bertujuan agar aplikasi yang dibuat dapat berfungsi dengan baik (sesuai dengan yang diharapkan).

Dalam perancangan aplikasi ini terdapat beberapa tahapan pengembangan yang harus dilakukan dengan tujuan agar aplikasi yang dirancang menjadi lebih mudah untuk dibangun, perancangan Aplikasi Smart Card Interface akan dijelaskan dalam bentuk Diagram Konteks dan Data Flow Diagram (DFD). Adapun uraian dari Diagram Konteks DFD adalah sebagai berikut:



Gambar 9. Diagram Konteks Aplikasi Smart Card Interface.

Dari gambar di atas dapat dilihat bahwa diagram konteks adalah diagram yang menggambarkan secara umum konteks yang terjadi dalam sistem antara dunia internal dan dunia eksternal. Diagram konteks dari Aplikasi Smart Card Interface adalah gambaran suatu proses hubungan input / output antara Aplikasi Smart Card Interface dengan entitas luarnya, yaitu : Contactless Smart Card dan Aplikasi Pengembang yang tentunya berbasis smart card.

Diagram konteks ini menunjukkan suatu interaksi antara aplikasi dengan entitas luarnya. Contactless Smart Card merupakan sumber data yang menyimpan data yang kemudian menjadi sumber input ke aplikasi untuk selanjutnya data tersebut di parsing ke dalam bentuk data yang fleksibel yang dapat di baca oleh semua bahasa pemrograman yang mendukung pembacaan data XML, properties, dan Text.

Mengenai uraian masukan dan pengeluaran data akan dijelaskan sebagai berikut : Smart Card Reader akan membaca data dari smart card dimana hasil pembacaan tersebut akan menjadi masukan yang akan diproses pada Aplikasi Smart Card Interface dan menghasilkan keluaran berupa file fleksibel yaitu bertipe XML, properties, dan text yang di

dalamnya berisikan data smart card yang berhasil dibaca Aplikasi Smart Card Interface, kemudian data dari file-file tersebut akan dimanfaatkan aplikasi pengembang.

#### 4. Hasil dan Pembahasan

##### 4.1. Pengujian Kompabilitas Aplikasi Dengan Bahasa Pemrograman Lain

Pengujian ini bertujuan untuk mengetahui kompabilitas aplikasi yang dibuat dengan bahasa pemrograman selain java. Untuk pengujiannya sebenarnya penulis tidak mengujikan langsung dengan bahasa pemrograman yang ada pada tabel 2, namun penulis hanya mencari referensi dukungan bahasa pemrograman pada tabel 2 terhadap pembacaan data dari *file* berjenis xml, text, dan properties karena asalkan bahasa pemrograman apapun itu jika mendukung pembacaan data dari *file* xml, text, dan properties maka aplikasi yang dibuat dengan bahasa pemrograman tersebut dapat diintegrasikan dengan aplikasi *Smart Card Interface*. Hasil tersebut adalah seperti dibawah ini:

**Tabel 1.** Hasil Pengujian Kompabilitas Aplikasi Dengan Bahasa Pemrograman Lain

NO	Bahasa Pemrograman	Kompabilitas dengan Smart Card ACR 122U	
		Tanpa SCI	Dengan SCI
1.	Java	√	√
2.	PHP	-	√
3.	VB	√	√
4.	C	√	√
5.	C++	√	√
6.	Delphi	√	√
7.	C#	-	√
8.	Rubby	-	√
9.	Groovy	-	√
10	Phyton	-	√

Dari hasil pengujian tersebut, didapatkan bahwa 4 dari bahasa pemrograman pada tabel 2 di atas kompatibel dengan *Smart Card* ACR 122U tanpa SCI namun SCI tetap dapat diintegrasikan dengan bahasa pemrograman tersebut. Dan semua bahasa pemrograman di atas dapat diintegrasikan dengan aplikasi SCI (*Smart Card Interface*). Kesimpulannya aplikasi SCI kompatibel dengan berbagai bahasa pemrograman dan aplikasi SCI dapat menjadi penghubung antara bahasa pemrograman yang tidak didukung *Smart Card* ACR 122U untuk mengembangkan aplikasi berbasis *Smart Card* khususnya dengan ACR122U.

##### 4.2. Pengujian Kecepatan Aplikasi Untuk Membaca Data *Smart Card*

Pengujian ini bertujuan untuk mengetahui kecepatan membaca data *smart card* dari aplikasi yang dikembangkan tanpa dan dengan aplikasi *Smart Card Interface*. Hasil tersebut adalah seperti dibawah ini:

**Tabel 2.** Hasil Pengujian Kecepatan Aplikasi Untuk Membaca Data *Smart Card*

NO	Teknik Pengembang an	Waktu Pembacaan data(dalam detik)				Rata-rata(dalam detik)
		Pengujian Ke-1	Pengujian Ke-2	Pengujian Ke-3	Pengujian Ke-4	
1.	Tanpa SCI	2,18	2,14	2,20	1,86	2,10
2.	Dengan SCI	1,78	2,66	2,10	1,43	1,99

Dari hasil pengujian tersebut, didapatkan bahwa kecepatan rata-rata pembacaan aplikasi berbasis *Smart Card* ACR 122U yang dikembangkan tanpa SCI adalah 2,10 detik sedangkan aplikasi yang dikembangkan dengan aplikasi SCI adalah 1,99 detik. Kesimpulan aplikasi yang dikembangkan dengan SCI memiliki kecepatan membaca data *Smart Card* lebih cepat daripada aplikasi yang dikembangkan tanpa SCI (pembacaan data langsung dari chip *smart card*), hanya saja pembacaan awal aplikasi dengan SCI lebih lama karena aplikasi memerlukan waktu untuk me-load data file yang dihasilkan SCI.

#### 4.3. Pengujian Kemudahan Penggunaan Sistem Untuk Pengembang/Programmer

Pengujian ini bertujuan untuk mengetahui apakah aplikasi yang dibuat secara umum dapat dengan mudah dimengerti oleh pengguna atau tidak. Untuk pengujiannya, diberikan kesempatan kepada 5 orang untuk menguji aplikasi. Hasil tersebut adalah seperti dibawah ini:

**Tabel 3.** Hasil Pengujian Kemudahan Penggunaan Sistem Untuk Pengembang/Programmer

NO	Pengguna	Tanpa SCI		Dengan SCI	
		Mudah	Sulit	Mudah	Sulit
1.	Pengguna 1	-	√	√	-
2.	Pengguna 2	-	√	√	-
3.	Pengguna 3	-	√	√	-
4.	Pengguna 4	-	√	√	-
5.	Pengguna 5	-	√	√	-

Dari hasil pengujian tersebut, didapatkan bahwa keseluruhan dari pengguna mengatakan bahwa mengembangkan aplikasi berbasis *smart card* ACR 122U tanpa SCI lebih sulit dibandingkan mengembangkan aplikasi *smart card* ACR 122U dengan SCI.

## 5. Penutup

### 5.1. Kesimpulan

Dari hasil uji coba Aplikasi *Smart Card Interface* yang dilakukan, dapat ditarik beberapa simpulan, antara lain:

1. Dengan teknologi *smart card* banyak aplikasi praktis yang dapat dikembangkan.
2. Aplikasi *Smart Card Interface* ini membantu para *programmer/software developer* dengan berbagai kompetensi bahasa pemrograman komputer dalam mengembangkan aplikasi berbasis *Smart Card* dengan mudah karena tidak lagi memikirkan program untuk membaca kartu *smart card*.
3. Fleksibilitas aplikasi dapat ditunjukkan dengan dukungan berbagai bahasa pemrograman komputer, terlebih karena pengguna tidak dipersulit dengan pembuatan program pembaca *smart card*.
4. Efisiensi aplikasi *smart card* yang dikembangkan dengan SCI dapat diandalkan karena waktu pembacaan data yang dibutuhkan lebih cepat daripada aplikasi *smart card* tanpa SCI, hanya saja pembacaan diawal lebih lama karena aplikasi me-load data dari file yang dihasilkan oleh SCI.

### 5.2. Saran

Hal yang perlu diperhatikan untuk mengembangkan aplikasi ini lebih lanjut yaitu perlunya pengembangan aplikasi *smart card* ini di *platform/Operating System* lain seperti linux, Macintos dan sebagainya sehingga memungkinkan ketergantungan sistem operasi dapat diatasi sehingga akan terbentuk sebuah sistem yang benar-benar fleksibel karena selain dukungan *multi programming* juga bisa *multi platform*.

## 6. Daftar pustaka

- Advanced Card Systems Ltd. 2008. *Application Programming Interfaces ACR 122U NFC Reader*. Hongkong: Advanced Card Systems Ltd.
- Advanced Card Systems Ltd. 2008. *Peer to Peer Demo Manual ACR 122U NFC Reader*. Hongkong : Advanced Card Systems Ltd.
- Advanced Card Systems Ltd. 2008. *Technical Specification ACR 122U NFC Reader*. Hongkong : Advanced Card Systems Ltd.
- Advanced Card Systems Ltd. 2008. *Visitor Management System Manual ACR 122U NFC Reader*. Hongkong : Advanced Card Systems Ltd.
- Anonim. 2010. *XML dan Java*. Diakses pada : 05 Oktober 2010, URL : <http://www.icoen.co.cc>
- Brian Benz, John R. Durant. 2003. *XML Programming Bible*. New York: Wiley Publishing, Inc.
- Dwi Cahyo Margoselo, Bambang. 2003. *Tinjauan Smart Card Untuk Pengamanan Database Berbasis Komputer*. Bandung : Institut Teknologi Bandung.
- Dwi Prasetyo, Didik. 2007. *150 Rahasia Pemrograman Java*. Jakarta: PT. Elex Media Komputindo
- Harold, Elliotte Rusty 2001, 2002. *Processing XML with Java*.
- Hermawan, Benny. 2004. *Menguasai Java 2 & Object Oriented Programming*. Yogyakarta: Penerbit ANDI.
- McLaughlin, Brett. 2001. *Java & XML, 2<sup>nd</sup> Edition*. Newyork : O'Reilly.
- Noprianto. 2004. *Mengenal XML*. Jakarta: Majalah INFOLINUX edisi Desember 2004.
- Pemda Jembrana. 2009. *J-ID (Jembrana Identitas Diri)*. Diakses Pada : 20 Oktober 2010, URL : <http://www.jembranakab.go.id>
- Pemda Jembrana. 2009. *J-Smart*. Diakses Pada : 20 Oktober 2010, URL : <http://www.jembranakab.go.id>
- Philips Semiconductors. 2001. *Mifare Standard Card IC MF1 IC S50 Specification*. Philips Semiconductors.
- Swastika, Agus. 2009. *Jembrana Smart Schoo(JSS)*. Diakses Pada : 20 Oktober 2010, URL : <http://guslong.wordpress.com>