

Aplikasi Point Of Sales Yang Terhubung Dengan Electronic Data Capture

Bobby Loardy, Budi Bunawan, Petrus Hartono

Abstrak—Tujuan dari penelitian adalah pengembangan aplikasi *Point Of Sales* (POS) yang terhubung dengan *Electronic Data Capture* adalah mengurangi kesalahan yang terjadi dalam penggunaan POS dan EDC dalam pembayaran dengan kartu. Hasil yang dicapai adalah aplikasi POS yang terhubung dengan EDC yang memungkinkan komunikasi data di antara keduanya. Pengguna POS tidak perlu lagi memasukkan jumlah pembayaran secara manual ke EDC dengan adanya perangkat komunikasi tersebut. Pengguna tidak perlu lagi memasukkan bukti transaksi dari EDC ke POS.

Kata kunci— *Point Of Sales*, *Electronic Data Capture*, Kartu pintar, UML.

I. PENDAHULUAN

Pembayaran non-tunai menggunakan kartu semakin berkembang pesat. Dalam penggunaan kartu sebagai alat pembayaran diperlukan alat lain untuk melaksanakan proses pembayaran. Alat tersebut adalah *Electronic Data Capture* (EDC). EDC berfungsi untuk memvalidasi kartu, memproses pembayaran, menangkap data transaksi dan mengirim data transaksi ke bank. Penggunaan EDC dalam proses pembayaran dengan kartu menjadi sangat vital. Masalah yang umum terjadi adalah kesalahan input data baik ke EDC ataupun ke sistem komputer mereka. Masalah tersebut seringkali terjadi karena faktor kesalahan operator yang mengoperasikan EDC. Kesalahan input tersebut memberikan dampak yang cukup besar baik ke pihak perusahaan maupun ke bank penerbit kartu serta kepada pemilik kartu itu sendiri karena diperlukan proses koreksi kembali yang terkadang merepotkan dan menyulitkan pihak-pihak yang terkait.

Masalah kesalahan input tersebut perlu ditangani dengan baik karena dampak yang ditimbulkannya. Salah satu cara penanganan masalah tersebut adalah dengan menghubungkan *Point Of Sales* (POS) dengan EDC. Dengan menghubungkan POS dengan EDC maka POS dan EDC mampu berkomunikasi serta saling mengirimkan informasi yang dibutuhkan. Hubungan tersebut juga mampu mengurangi faktor kesalahan input data dengan demikian dampak kesalahan dapat dikurangi bahkan dapat dihilangkan.

II. LANDASAN TEORI

A. Kartu Pintar

Kartu pintar adalah sebuah kartu yang di-embed dengan

sirkuit *microcontroller* yang terintegrasi yang dapat memproses data karena memiliki *internal memory* atau *memory chip* [1]. Kartu pintar terhubung ke pembaca kartu baik melalui kontak langsung maupun tidak langsung melalui *contactless radio frequency interface*. Dengan adanya sirkuit *microcontroller* memungkinkan kartu pintar untuk dapat menyimpan data dan menjalankan fungsi yang berada dalam kartu seperti enkripsi dan autentikasi data serta dapat berinteraksi secara cerdas dengan pembaca kartu pintar.

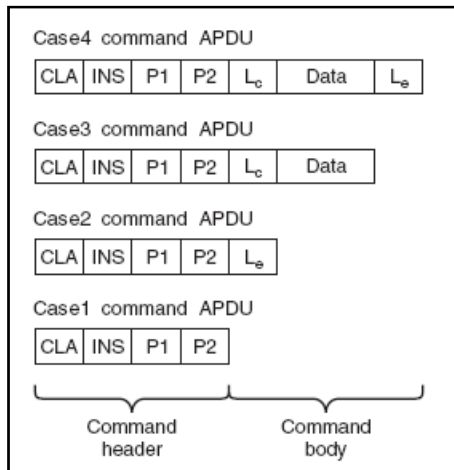
Berdasarkan cara transmisi data, kartu pintar terbagi menjadi 2 yaitu *contact smart card* dan *contactless smart card*. *Contact smart card* bekerja dengan komunikasi melalui kontak langsung secara fisik antara chip kartu dengan pembaca kartu. *Contact smart card* diatur dengan standar internasional ISO/IEC 7816 [2]. *Contactless smart card* berkomunikasi dengan pembaca kartu dengan memancarkan Radio Frequency ID (RFID). *Contactless smart card* diatur dengan standar internasional ISO/IEC 14443 [3].

Komunikasi dengan kartu pintar dilakukan dengan menggunakan suatu protokol yaitu *Application Protocol Data Unit* (APDU) [4] [5]. APDU didefinisikan sebagai suatu data paket yang berisi perintah yang dikirim ke kartu pintar dan respon yang dihasilkan.

Sebuah perintah APDU terdiri dari sebuah *command header* dan sebuah *command body*. *Command header* bersifat wajib tetapi *command body* bersifat opsional.

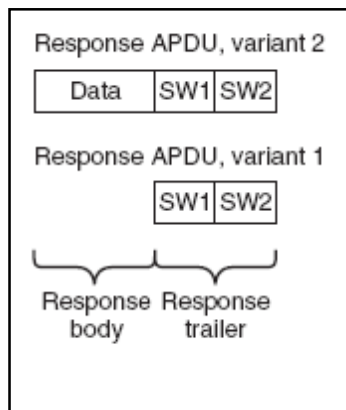
Sebuah *command header* terdiri dari 4 byte data yang memiliki format sebagai berikut: *Class* (CLA), *Instruction* (INS), *Parameter1* (P1) dan *Parameter2* (P2) yang masing-masing berukuran 1 byte. CLA berfungsi untuk mengidentifikasi aplikasi atau program yang ada pada kartu pintar. INS menyatakan kode instruksi/perintah. P1 dan P2 merupakan parameter yang dibutuhkan.

Sebuah *command body* terdiri dari maksimum 3 elemen data yaitu: *Length command* (Lc) yang berisi panjang dari data dalam perintah APDU, *Length expected* (Le) menyatakan jumlah maksimal dari byte yang diharapkan di dalam *data field* dari respon APDU dan *Data field* menyatakan data yang diperlukan oleh perintah APDU. Kombinasi dari perintah APDU seperti pada Gambar 1.



Gambar 1 Kombinasi Perintah APDU

Respon APDU adalah suatu respon atau balikan atas perintah APDU yang diberikan. Sebuah respon APDU terdiri dari sebuah *response body* yang bersifat opsional dan *response trailer* yang bersifat wajib. *Response body* berisi data yang diperlukan oleh respon APDU. *Response trailer* berisi *status byte* SW1 dan SW2 yang menyatakan status proses dari perintah APDU yang dikirim. Kombinasi dari format respon APDU seperti pada Gambar 2.



Gambar 2 Kombinasi Respon APDU

B. Electronic Data Capture

Electronic Data Capture (EDC) adalah sebuah sistem terkomputerisasi yang dibuat untuk mengumpulkan data dalam bentuk data elektronik [6]. Penggunaan EDC saat ini lebih sering digunakan dalam pembayaran menggunakan kartu baik kartu kredit maupun kartu debit. Ketika kartu digesekkan atau dihubungkan dengan pembaca kartu elektronik atau terminal penerima data dari kartu yang ada dalam EDC maka informasi yang tersimpan dalam kartu dimasukkan/ ditangkap oleh EDC. Secara umum, EDC menyediakan:

- Komponen *Graphical User Interface* (GUI) untuk memasukkan data.
- Komponen validasi untuk mengecek data.
- Alat untuk pembuatan laporan untuk mendukung analisis dari data yang telah dikumpulkan.

C. Point Of Sales

Point Of Sales (POS) dapat berupa sebuah *checkout counters* dalam sebuah toko atau tempat usaha dimana transaksi penjualan terjadi [7]. Secara lebih spesifik, POS umumnya merujuk pada perangkat keras dan perangkat lunak yang digunakan oleh *checkout counter* seperti *electronic cash register*, *barcode scanner*, pembaca kartu dan perangkat lainnya. POS banyak digunakan pada *supermarket*, restoran, hotel, dan stadion, seperti pada Gambar 3.

Penggunaan POS pada industri restoran telah merevolusi industri restoran khususnya dalam bidang *fast food*. Penggunaan POS telah meningkatkan efisiensi kerja karena dalam mempercepat proses order pesanan oleh *customer*, dapat membuat laporan secara cepat, mempercepat proses perhitungan, memungkinkan pencarian data, serta melacak transaksi harian.

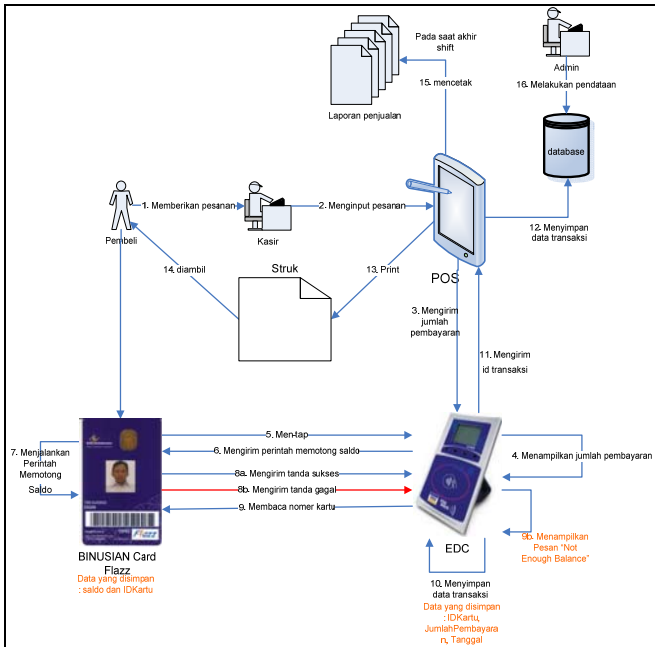


Gambar 3 Perangkat Keras POS

III. SOFTWARE DEVELOPMENT

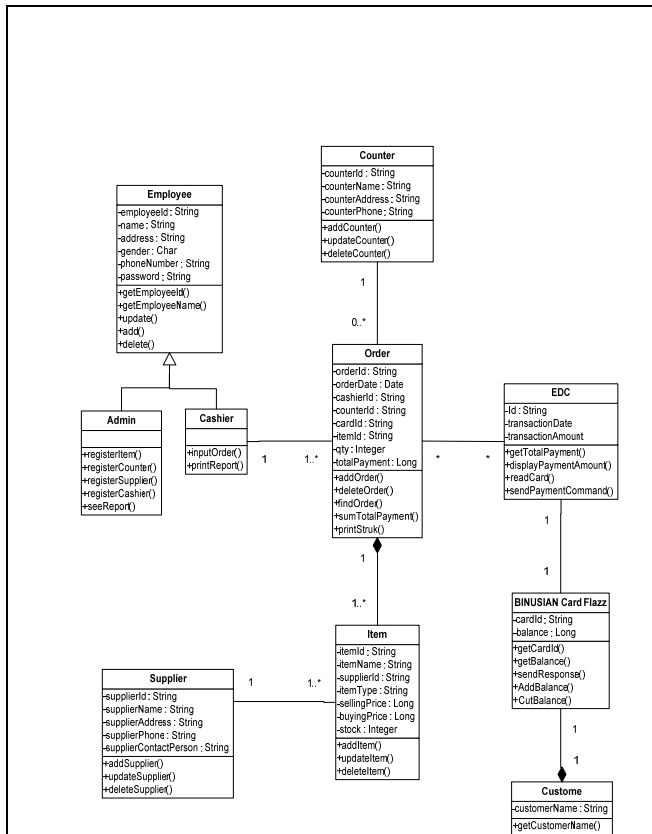
A. Model Konseptual

Model Konseptual dari aplikasi POS yang terhubung dengan EDC seperti pada Gambar 4, Gambar 5 dan Gambar 6. Cara kerja dari aplikasi POS yang terhubung dengan EDC digambarkan pada Gambar 4. *Class diagram* yang dibangun digambarkan pada Gambar 5. Gambar 6 menggambarkan kebutuhan komponen perangkat lunak yang dibutuhkan.

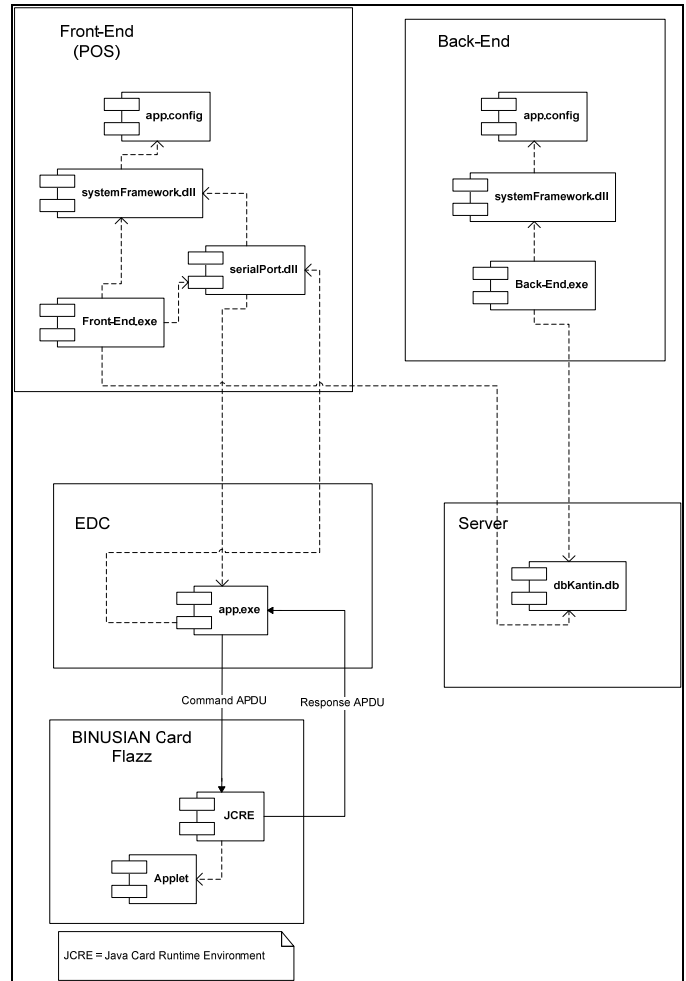


Gambar 4 Model Konseptual

B. Design Model



Gambar 5 Class Diagram



Gambar 6 Component Diagram

C. Kebutuhan Perangkat Keras

Untuk dapat menjalankan aplikasi dengan baik, perangkat keras yang dibutuhkan adalah sebagai berikut.

1. Komputer Server

Kebutuhan perangkat keras pada komputer server adalah server dengan minimum spesifikasi:

- Prosesor Pentium IV 3.0 GHz.
- Memory RAM 512 MB.
- Harddisk Drive berkapasitas 40 GB.
- Monitor, keyboard, dan mouse.

2. Komputer Client

Komputer client terbagi menjadi 2 yaitu client yang menjalankan aplikasi Front-End (POS) dan client yang menjalankan aplikasi Back-End. Kebutuhan perangkat keras pada masing-masing client minimum memiliki spesifikasi sebagai berikut.

- Client untuk aplikasi Front-End (POS), yaitu:
 - Prosesor Pentium IV 2.0 GHz.
 - Memory RAM 256 MB.
 - Hard Disk Drive berkapasitas 10 GB.
 - Monitor Touch Screen 12 inci.
 - Printer khusus untuk struk.
- Client untuk aplikasi Back-End, yaitu:
 - Prosesor Pentium IV 2.0 GHz.

- ii. Memory RAM 256 MB.
- iii. *Hard Disk Drive* berkapasitas 10 GB.
- iv. Monitor, *keyboard*, *mouse*.
- v. Printer.

Selain komputer *client*, untuk dapat menjalankan aplikasi diperlukan EDC dengan spesifikasi minimum:

- a. Memiliki *contactless reader*.
- b. Mampu membaca kartu pintar dengan ISO/IEC 7816 dan ISO/IEC 14443.
- c. Memiliki *display panel* dengan ukuran 128*64 dengan *backlight*.
- d. Memiliki 1 port RS232, 1 USB, dan 4 SAM slot.
- e. Memiliki 4 *LED indicators* dan 1 MB SRAM

Kartu yang digunakan dalam aplikasi merupakan kartu pintar dengan tipe *java card* yang memiliki spesifikasi minimum sebagai berikut:

- a. Mendukung JavaCard 2.1.1 OS
- b. Memiliki 32KB EEPROM
- c. Mendukung DES dan RSA

IV. EVALUASI

Hasil evaluasi yang didapatkan adalah sebagai berikut.

1. Dapat mengurangi waktu antrian yang dihabiskan pada transaksi pembayaran.

Dengan penggunaan aplikasi, proses pembayaran menjadi lebih cepat sehingga dapat mengurangi waktu antrian yang dibutuhkan, seperti pada Tabel 1 dan Tabel 2.

Tabel 1 Rata-Rata Waktu Transaksi Pada Proses Bisnis Lama

Transaksi Jumlah item yang dibeli	Waktu yang dibutuhkan	
	Tunai	<i>BINUSIAN Card-Flazz</i>
1	50 detik	35 detik
3	1 menit 10 detik	45 detik
5	2 menit	58 detik

Tabel 2 Rata-Rata Waktu Transaksi Menggunakan Aplikasi

Transaksi Jumlah item yang dibeli	Waktu yang dibutuhkan <i>BINUSIAN Card-Flazz</i>
1	25 detik
3	35 detik
5	48 detik

2. Mengurangi kemungkinan terjadinya *human error*

Kasir sebagai pengguna aplikasi tidak perlu meng-input jumlah pembayaran secara manual ke EDC dan *id* bukti transaksi secara manual ke POS. Aplikasi POS secara langsung mengirim jumlah yang harus dibayar ke EDC. EDC secara langsung mengirim *id* bukti transaksi ke POS. Dengan proses yang tanpa melibatkan manusia dapat mengurangi bahkan menghilangkan kemungkinan terjadinya *human error*.

3. Mempercepat waktu pembuatan laporan penjualan.

Kasir tidak lagi perlu membuat laporan penjualan secara manual. Aplikasi akan secara langsung mencetak laporan penjualan. Perbandingan rata-rata waktu yang diperlukan

untuk membuat laporan penjualan secara manual dan yang dibutuhkan aplikasi adalah sebagai berikut.

Secara manual : 10 menit
Secara komputerisasi : 3 menit

REFERENSI

- [1] Smart Card Alliance. 2007. *Smart Card FAQ*.
<http://www.smartcardalliance.org/pages/smart-cards-faq>.
- [2] Wikipedia. 2008. *ISO 7816*.
http://en.wikipedia.org/wiki/ISO_7816.
- [3] Wikipedia. 2008. *ISO 14443*.
http://en.wikipedia.org/wiki/ISO_14443.
- [4] Rankl, Wolfgang. 2007. *Smart Card Applications Design Models For Using And Programming Smart Cards*. Wiley. England.
- [5] Ortiz, Enrique. 2003. *An Introduction to Java Card Technology –Part 1*.
<http://developers.sun.com/mobility/javacard/articles/javacard1/>.
- [6] Wikipedia. 2008. *Electronic Data Capture*.
http://en.wikipedia.org/wiki/Electronic_data_capture.
- [7] Wikipedia. 2008. *Point Of Sales*.
http://en.wikipedia.org/wiki/Point_of_sale.