

**RANCANG BANGUN APLIKASI BILLING SMART CARD
UNTUK MEMBER WARNET**

SKRIPSI



Disusun Oleh :
RENDY SYAHPUTRA
12.18.129

PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2016

LEMBAR PERSETUJUAN

**RANCANG BANGUN APLIKASI BILLING SMART CARD UNTUK
MEMBER WARNET**

SKRIPSI

*Disusun dan Diajukan untuk melengkapi dan memenuhi persyaratan
guna mencapai Gelar Sarjana Komputer Strata Satu (S-1)*

Disusun Oleh :

RENDY SYAHPUTRA

12.18.129

Diperiksa dan Disetujui,

Dosen Pembimbing I

Dosen Pembimbing II

Sonny Prasctio, ST, MT

Febriana Santi W, S.Kom, M.Kom

NIP.P 1031000433

NIP.P 1031000425

Mengetahui,

Program Studi Teknik Informatika S-1

Ketua



Joseph Dedy Irawan, ST, MT

NIP. 197404162005011002

**PROGRAM STUDI TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2016

LEMBAR KEASLIAN
PERNYATAAN KEASLIAN SKRIPSI

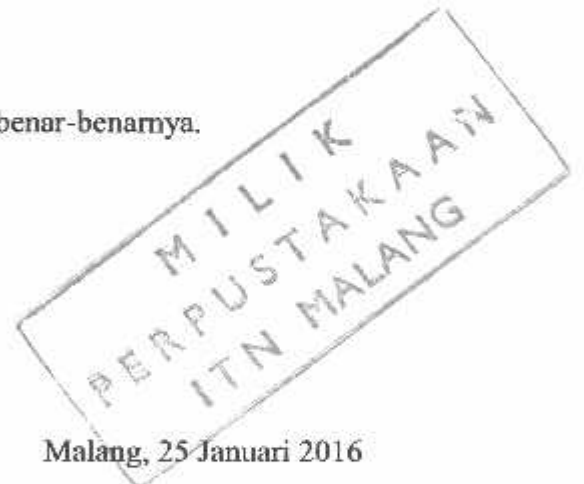
Sebagai mahasiswa Program Studi Teknik Informatika S-1 Fakultas Teknologi Industri Institut Teknologi Nasional Malang, yang bertanda tangan di bawah ini, saya:

Nama : Rendy Syahputra

NIM : 12.18.129

Menyatakan dengan sesungguhnya bahwa skripsi saya dengan judul *"Rancang Aplikasi Billing Smart Card Untuk Member Warnet"* merupakan karya asli dan bukan merupakan duplikat dan mengutip seluruhnya karya orang lain. Apabila di kemudian hari, karya asli saya disinyalir bukan merupakan karya asli saya, maka saya akan bersedia menerima segala konsekuensi apa pun yang diberikan Program Studi Teknik Informatika S-1 Fakultas Teknologi Industri Institut Teknologi Nasional Malang.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya.



Malang, 25 Januari 2016

Yang membuat pernyataan



Rendy Syahputra

NIM. 1218129

RANCANG BANGUN APLIKASI BILLING SMART CARD UNTUK MEMBER WARNET

Rendy Syahputra (1218129)

Program Studi Teknik Informatika S-1

Institut Teknologi Nasional Malang

Email : rendysyahputra99@gmail.com

Dosen Pembimbing: 1. Sonny Prasectio, ST, MT.

2. Febriana Santi Wahyuni, S.Kom,M.Kom.

Abstrak

Rancang Bangun Billing Smart Card Untuk Member Warnet. Aplikasi ini dirancang sebagai proses penghitungan waktu sekaligus transaksi pembayaran pemakaian warnet menggunakan teknologi smart card. Karena di warnet inilah para pengguna jasa warnet mencari informasi, bermedia sosial, atau hanya bermain game dengan bersenang-senang saja.

Aplikasi Billing Smartcard ini menggunakan Borland Delphi 7 yang terhubung dengan smart card yang nantinya akan menjadi perantara antara server admin dengan PC client. Aplikasi ini dapat menangani proses penghitungan waktu dan biaya pemakaian warnet, dapat mengisikan data member yang menggunakan smartcard ataupun menambah saldo pelanggan yang menambah saldonya dengan memilih data pelanggan yang tampil pada tabel pelanggan yang ada pada admin. Warnet yang ada di masyarakat saat ini dalam melakukan suatu transaksi pembayaran kebanyakan masih menggunakan dengan cara manual.

Dalam Aplikasi ini tipe Reader ACR38 dan sangat kompatibel dengan 4 versi Windows yang banyak digunakan saat ini, seperti Windows Xp, Windows 7, Windows 8.1 serta Windows 10, selain itu untuk pengaturan alamat IP dan nama komputer pada client dapat diubah pada file "set.ini" pada file client.

Kata Kunci : *Smartcard, Warnet, Billing, Borland Delphi 7.*

KATA PENGANTAR

Puji Syukur Kehadirat Tuhan Yang Maha Esa, karena atas kasih dan karuniaNya yang telah diberikan selama ini sehingga penulis dapat menyelesaikan skripsi ini dengan Judul "**Rancang Bangun Aplikasi Billing Smart Card Untuk Member Warnet**".

Penyusunan skripsi ini merupakan salah satu persyaratan untuk menyelesaikan program pendidikan Strata Satu (S-1) Teknik Informatika, Fakultas Teknologi Industri di Institut Teknologi Nasional Malang.

Pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Bapak Dr. Ir. Lalu Mulyadi, MT selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Anang Subardi, MT selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Joseph Dedy Irawan, ST, MT selaku Ketua Program Studi Teknik Informatika S-1 Institut Teknologi Nasional Malang.
4. Bapak Sonny Prasetyo, ST, MT selaku Dosen pembimbing I.
5. Ibu Febriana Santi Wahyuni, S.Kom, M.Kom selaku Dosen pembimbing II.
6. Bapak dan Ibu Dosen Teknik Informatika S-1 selaku pengamat dan penguji.
7. Kedua Orang Tua tercinta yaitu Bapak Suwono dan Ibu Ninik Purwanti.
8. Semua teman-teman seperjuangan Teknik Informatika yang tidak bisa disebutkan satu-persatu yang telah memberikan semangat, dukungan, saran dan bantuan.
9. Serta semua pihak yang telah membantu dalam penyelesaian skripsi ini.

Penulis menyadari bahwa skripsi masih jauh dari sempurna, oleh karena itu penulis mengharapkan kritik dan saran dari pembaca. Sehingga skripsi ini bisa bermanfaat bagi para pembaca sekalian.

Malang, 25 Januari 2016



Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	
LEMBAR KEASLIAN -----	iii
ABSTRAK -----	iv
KATA PENGANTAR -----	v
DAFTAR ISI -----	vi
DAFTAR GAMBAR -----	viii
DAFTAR TABEL -----	xi
BAB I PENDAHULUAN -----	1
1.1 Latar Belakang -----	1
1.2 Rumusan Masalah -----	1
1.3 Tujuan -----	2
1.4 Batasan Masalah -----	2
1.5 Metode Penelitian -----	2
1.6 Sistematika Penulisan -----	3
BAB II LANDASAN TEORI -----	5
2.1 Warnet -----	5
2.2 <i>Smart Card</i> -----	5
2.2.1 Pengertian Smart Card -----	5
2.2.2 Tipe-Tipe Smart Card -----	9
2.3 <i>Smart Card Reader</i> -----	10
BAB III ANALISIS DAN PERANCANGAN -----	13
3.1 Spesifikasi Harware Dan Software Sistem -----	13
3.1.1 Sistem Saat Ini -----	13
3.1.2 Sistem Yang Akan Dibangun -----	13
3.1.3 Spesifikasi <i>Hardware</i> dan <i>Software</i> -----	14
3.2 <i>Blok Diagram Sistem</i> -----	14
3.2.1 <i>Blok Diagram Sistem</i> -----	14
3.2.2 Desain Struktur <i>Menu</i> -----	15
3.2.3 Flowchart Sistem -----	16
3.2.4 Perancangan <i>Interface</i> -----	18

BAB IV IMPLEMENTASI DAN PENGUJIAN -----	22
4.1 Implementasi -----	22
4.2 Pengujian -----	46
BAB V PENUTUP -----	52
5.1 Kesimpulan -----	52
5.2 Saran -----	52
DAFTAR PUSTAKA -----	53
LAMPIRAN -----	

DAFTAR GAMBAR

Gambar 2.1 <i>Smart Card</i>	5
Gambar 2.2 Standar <i>PIN Contact Smart Card</i>	6
Gambar 2.3 Bentuk <i>Contact Smart Card</i>	6
Gambar 2.4 Bentuk <i>Contactless Smart Card</i>	7
Gambar 2.5 Karakter Fisik Dari <i>Smart Card</i> Berdasarkan Standar <i>ISO 7816</i>	8
Gambar 2.6 <i>Smart Card Reader ACR 38</i>	11
Gambar 3.1 Blok Diagram Sistem	14
Gambar 3.2 Desain Struktur Menu	15
Gambar 3.3 <i>Flowchart</i>	17
Gambar 3.4 <i>Form Login Admin</i>	18
Gambar 3.5 Pemberitahuan Cek <i>Login</i>	18
Gambar 3.6 <i>Form Menu</i>	19
Gambar 3.7 <i>Form Saldo</i>	19
Gambar 3.8 <i>Form Pelanggan</i>	20
Gambar 3.9 <i>Form Billing</i>	21
Gambar 3.10 <i>Form Billing Client</i>	21
Gambar 4.1 Tampilan <i>Login Admin</i>	22
Gambar 4.2 <i>Login Berhasil</i>	23
Gambar 4.3 <i>Login Salah Password</i>	23
Gambar 4.4 <i>Login Salah Username</i>	24
Gambar 4.5 <i>Form Menu</i>	24
Gambar 4.6 <i>Form Tambah</i>	25
Gambar 4.7 <i>Form Pelanggan Simpan</i>	25
Gambar 4.8 <i>Form Pelanggan Ubah</i>	26
Gambar 4.9 <i>Form Pelanggan Update</i>	26
Gambar 4.10 <i>Form Pelanggan Hapus</i>	27
Gambar 4.11 Konfirmasi Hapus	27
Gambar 4.12 Data Pelanggan Terhapus	28
Gambar 4.13 <i>Menu Utama</i>	28

Gambar 4.14 Tambah Saldo	29
Gambar 4.15 Saldo Bertambah	29
Gambar 4.16 Data Belum Lengkap	30
Gambar 4.17 Pilih <i>Billing</i>	30
Gambar 4.18 <i>Form Billing</i>	31
Gambar 4.19 <i>Form Client</i>	32
Gambar 4.20 <i>Form Client</i> Mulai	32
Gambar 4.21 <i>Form Billing (Client</i> Mulai).....	33
Gambar 4.22 <i>Client</i> Selesai	33
Gambar 4.23 Tampilan <i>Billing</i> Saat <i>Client</i> 6 selesai	34
Gambar 4.24 <i>Client</i> Stanby	34
Gambar 4.25 <i>Start Client</i> Dari <i>Admin</i>	35
Gambar 4.26 <i>Client</i> Started	35
Gambar 4.27 <i>Start Client</i> Dari <i>Admin</i>	36
Gambar 4.28 <i>USB Smart Card Reader</i> Masuk	36
Gambar 4.29 <i>Admin</i> Start <i>Member</i>	37
Gambar 4.30 <i>USB Smart Card Reader</i> Belum Masuk	37
Gambar 4.31 Koneksi Dengan <i>Smart Card Reader</i> Berhasil	38
Gambar 4.32 Informasi Untuk Memasukkan Kartu <i>Smart Card</i>	38
Gambar 4.33 <i>USB Smart Card Reader</i> Masuk.....	39
Gambar 4.34 Informasi Nama <i>Client</i> Pada <i>Admin</i>	39
Gambar 4.35 Informasi Saldo Awal <i>Client</i> Pada <i>Admin</i>	40
Gambar 4.36 <i>Client</i> Online	40
Gambar 4.37 Komputer <i>Client</i> Sudah Siap Dan <i>Online</i>	41
Gambar 4.38 <i>Client Smart Card</i> Selesai	41
Gambar 4.39 Saldo Pelanggan Berkurang	42
Gambar 4.40 Koneksi Dengan <i>Server</i> Terputus	42
Gambar 4.41 <i>Smart Card ToolSet PRO v3.4</i>	43
Gambar 4.42 Tampilan Aplikasi Saat Dibuka	43
Gambar 4.43 Konfigurasi <i>Path</i>	44

Gambar 4.44 <i>Component</i> Pada Delphi 7	44
Gambar 4.45 Hasil <i>Component</i> Yang Diambil.....	44
Gambar 4.46 <i>Source Code</i> Diisikan Pada <i>Event</i>	45
Gambar 4.47 <i>Connect to the smart card service</i>	45
Gambar 4.48 <i>Open the card explorer window</i>	45
Gambar 4.49 <i>Command</i> pada <i>Single APDU</i>	46
Gambar 4.50 <i>Command</i> melalui <i>APDU Batch File</i>	46

DAFTAR TABEL

Tabel Pengujian 4.1 Hasil Pengujian Fungsi	47
Tabel Pengujian 4.2 Pengujian <i>Smart Card Reader ACR38</i>	48
Tabel Pengujian 4.3 Pengujian Fungsional <i>Compability Smart Card</i>	49
Tabel Pengujian 4.4 Pengujian Fungsional <i>APDU Command</i> pada <i>Smart Card</i> -	50
Tabel Pengujian 4.5 Pengujian <i>User Admin</i>	50
Tabel Pengujian 4.6 Pengujian <i>User Member</i>	51

BAB I

PENDAHULUAN

1.1 Latar Belakang

Semakin semaraknya perkembangan teknologi khususnya di bidang internet di Indonesia, membawa dampak positif terhadap perkembangan bisnis jasa di bidang ini. Salah satunya adalah warung internet (warnet), di warnet inilah para pengguna jasa warnet mencari informasi, bermedia sosial, dan mengolah kemampuannya di dunia maya.

Warnet adalah tempat kebanyakan orang mengakses internet. Di negara-negara atau daerah-daerah maju sudah sangat banyak yang menggunakan jasa warnet untuk mengakses internet. Namun umumnya warnet yang ada sekarang dalam melakukan transaksi itu rata-rata masih manual. Pada saat pengguna selesai menggunakan komputer pada warnet, pengguna akan melakukan transaksi pembayaran dengan mendatangi penjaga warnet atau server secara langsung dan menunggu pengembalian uang jika yang dibayarkan lebih dari jumlah harga sewa.

Smart card merupakan sebuah kartu yang memiliki chip memori untuk menyimpan sebuah data, sehingga data yang ada pada kartu dapat dibaca oleh komputer untuk diambil data yang akan dibutuhkan untuk proses selanjutnya. Seperti halnya kartu sehat, kartu Indonesia pintar dan kartu sejenisnya.

Pada skripsi ini penulis akan memilih mengimplementasikan smart card sebagai pengganti transaksi manual dari pembayaran warnet yang diharapkan dapat menambah efisiensi bagi masyarakat serta penjaga warnet sebagai admin dan juga nantinya para member yang menggunakan smart card akan mendapatkan bonus minuman atau bonus saldo ketika menambah berapa saldo tergantung yang di tentukan admin.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan sebelumnya, maka penulis akan merumuskan masalah yang akan dibahas adalah Bagaimana

merancang aplikasi billing dengan memanfaatkan Smart Card untuk member warnet menggunakan Delphi 7 sebagai software pengembang.

1.3 Tujuan

Adapun tujuan yang ingin dicapai dari pembuatan aplikasi ini adalah Merancang aplikasi billing yang dapat mengakses Smart Card melalui Smart Card Reader yang dihubungkan dengan komputer sehingga dapat berfungsi sebagai alat transaksi penyewaan warnet.

1.4 Batasan Masalah

Agar pembahasan dalam skripsi ini tidak meluas, maka ditentukan beberapa batasan masalah, yaitu :

1. Aplikasi ini hanya fokus dibuat untuk transaksi penggunaan warnet dan tidak dilengkapi fitur *chatting*.
2. Informasi yang terdapat pada *smart card* berisi data *id user*.
3. *Smart Card Reader* hanya ada pada komputer *admin*.
4. Pada Skripsi ini tidak membahas aspek bisnis, jaringan komputer, TCP/IP, dan aspek keamanan data dalam *Smart Card*.
5. Tipe *Smart Card* yang digunakan adalah *ISO 7816* dan tipe *ACR 38* untuk *Smart Card Reader*.
6. Aplikasi dibuat menggunakan *Delphi 7*.

1.5 Metode Penelitian

Metode yang digunakan dalam penulisan tugas akhir ini adalah sebagai berikut:

1. Pengumpulan bahan-bahan dan hasil penelitian sebelumnya dan mempelajari teknologi *Smart Card*.
 2. Analisis kebutuhan. Tahap ini merupakan tahap awal dari perancangan sistem yaitu dengan analisis kebutuhan umum yang diperlukan oleh aplikasi *Smart Card* untuk rekam medis. Berdasarkan analisis kebutuhan umum tersebut
-

ditetapkan kriteria-kreteria sesuai dengan peraturan dan kebutuhan di rumah sakit.

3. *Preliminary design*. Perancangan tahap awal sistem dari segi perangkat lunak maupun perangkat keras berdasarkan analisis kebutuhan yang telah dilakukan.
4. *Detailed design*. Perancangan selanjutnya, untuk menuju perancangan tahap akhir sistem per modul.
5. Implementasi. Setelah *detailed design* maka akan dimulai tahap pembuatan perangkat lunak dan perangkat keras sesuai dengan spesifikasi yang dihasilkan oleh *detailed design*.
6. Pengujian sistem yang dibangun.
7. Analisis hasil pengujian. Dilakukan setelah hasil pengujian terhadap sistem secara keseluruhan.
8. Kesimpulan. Selanjutnya diambil kesimpulan rancangan aplikasi smartcard untuk rekam medis apakah sudah dapat memenuhi semua spesifikasi yang sesuai dengan kondisi di rumah sakit yang ada di Indonesia.

1.6 Sistematika Penulisan

Untuk mempermudah memahami pembahasan pada penulisan skripsi ini, maka sistematika penulisan diperoleh sebagai berikut :

BAB I : Pendahuluan

Berisi latar belakang, perumusan masalah, batasan masalah, tujuan penelitian, metode penelitian, dan sistematika penulisan.

BAB II : Landasan Teori

Berisi dasar teori mengenai permasalahan yang berhubungan dengan penelitian ini.

BAB III : Analisa dan Perancangan Sistem

Berisi mengenai perancangan rancang bangun aplikasi billing smart card untuk member warnet yang akan di buat.

BAB VI : Implementasi dan Pengujian

Berisi implementasi terhadap proses rancang bangun aplikasi billing smart card untuk member warnet yang akan di buat, serta melakukan pengujian terhadap aplikasi tersebut.

BAB V : Penutup

Berisi kesimpulan dari hasil penelitian yang dilakukan dan saran yang dapat digunakan untuk bahan pengembangan penelitian berikutnya.

BAB II

LANDASAN TEORI

Dalam penyusunan laporan skripsi ini diperlukan landasan teori yang memiliki relevansi dengan masalah yang dibahas. Landasan teori ini untuk memberikan arah, persepsi dan landasan untuk menentukan solusi terhadap permasalahan yang sedang dibahas. Landasan teori tersebut diperoleh dengan membahas beberapa literatur yang mempublikasikan pendapat beberapa ilmuwan yang dipakai sebagai penunjang pembahasan masalah.

2.1 Warnet

Warung Internet adalah sebuah tempat yang menyediakan akses infrastruktur internet dengan berbagai koneksi dan komputer sebagai perangkat akses sehingga pengguna bisa mengakses internet dengan biaya yang lebih murah (Ahmadjayadi, 2007).

2.2 Smartcard

2.2.1 Pengertian Smartcard

Smart card atau sering juga disebut *ICC (Integrated Circuit Card)* adalah kartu *plastic* yang berukuran sama seperti kartu kredit dan di dalamnya terdapat suatu *chip* silikon yang disebut *microcontroller* seperti pada gambar 2.1



Gambar 2.1. *Smart Card*

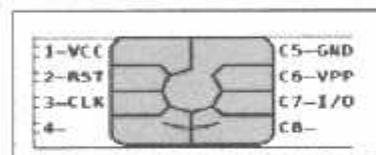
Chip merupakan rangkaian terintegrasi (*integrated circuit*) yang terdiri dari prosesor dan memori. *Chip*, seperti layaknya CPU (*Central Processing Unit*) di komputer, bertugas melaksanakan perintah dan menyediakan *power* ke *smart card*. (Pamungkas,2011)

Smart Card dikategorikan menjadi dua jenis, yaitu:

1. *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 2.2. Standar *PIN Contact Smart Card*



Gambar 2.3. Bentuk *Contact Smart Card*

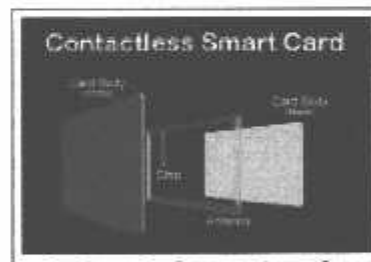
2. *Contactless Smart card*

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/output port, sistem operasi, dan hard disk.



Gambar 2.4. 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 bentuknya. 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.

Semua *smart card* berisi tiga tipe memori yaitu :

1. *Persistent Non-Mutable Memory*

Persistent Non-Mutable Memory atau biasa disebut ROM (*Read Only-Memory*) merupakan jenis memori yang tidak memungkinkan penggunanya untuk mengubah data yang ada didalamnya.

2. *Persistent Mutable Memory*

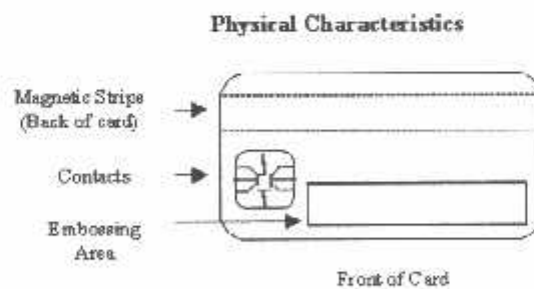
Persistent Mutable Memory atau EEPROM (*Electronical Erasable Programmable Read Only Memory*), seperti ROM, dapat menjaga isi data walaupun daya tidak ada. Perbedaannya adalah isi memori ini dapat dimodifikasi selama penggunaan kartu yang normal.

3. *Non-Persistent Mutable Memory*

Non-Persistent Mutable Memory atau RAM (*Random Acces Memory*) digunakan sebagai ruang bekerja yang bertempo (*temporary*) untuk menyimpan dan memodifikasi data. RAM adalah memori *nonpersistent*, sehingga isi informasi tidak dijaga ketika daya tidak ada.

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 2.5. Karakter Fisik Dari *Smart Card* Berdasarkan 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 juga *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*. (Swastika dkk, 2010)

2.2.2 Tipe-Tipe Smartcard

Secara komersial, industri membuat *smart card* dalam beberapa tipe, yaitu:

1. *Memory card*. *Smart card* tipe ini tidak mempunyai *processor* atau sistem keamanan yang canggih melainkan hanya perlindungan fisik (karena *smart card* bersifat *tamper proof*). *Smart card* ini merupakan tipe pertama yang dikenal orang dan digunakan pertama kali untuk kartu telepon. Tipe kartu ini menyimpan data yang telah di-*preload* oleh manufakturnya, kemudian mesin pembaca akan mengurangi isi variabel yang disimpannya
-

2. *Memory protected cards*. *Smart card* tipe ini mempunyai system keamanan yang lebih canggih daripada *memory cards*, misalnya mekanisme *password* untuk mengakses *smart card*.
3. *Microprocessor cards*. *Smart card* tipe ini mempunyai *processor* sehingga dapat melakukan komputasi walaupun terbatas. Keterbatasannya ada pada ukuran ROM yang dimiliki dan fungsi aritmatika yang masih sederhana. Kemampuannya antara lain mengorganisasikan berkas (file) yang dilindungi dengan *password*.
4. *Java cards*. *Smart card* ini dilengkapi dengan *Java Virtual Machine* sedemikian hingga dapat dimasukkan berbagai program ke dalamnya.
5. *Public key cards*. *Smart card* ini mendukung *public key cryptography* (kriptografi asimetris) sehingga proses *enkripsi/dekripsi* dapat dilakukan secara internal dan dapat menyimpan *key*.

2.3 Smartcard Reader

Smart card reader atau biasa disebut *reader* adalah antarmuka yang menghubungkan komunikasi antara *smart card* dengan PC. Istilah lain yang digunakan untuk *smart card reader* adalah terminal, *CAD (Card Acceptance Device)* dan *IFD (interface device/perangkat antarmuka)*.

Smart card tidak berarti tanpa adanya *smart card reader*, yang berfungsi sebagai perantara komunikasi antara *smart card* dengan peralatan lain seperti komputer. Komputer membaca atau menulis data melalui *smart card reader*, kemudian *smart card reader* mengubah perintah membaca/menulis tersebut ke dalam bahasa yang dimengerti *Smart Card*.

Reader membangun hubungan dengan mikroprosesor *smart card* melalui kontak elektrik pada permukaan *smart card*. Melalui hubungan elektrik ini, *reader* menyediakan *power* ke *smart card* dan membangun hubungan suatu pertukaran data.

Reader dapat berhubungan dengan PC biasanya melalui *port keyboard*, *port serial* atau *port PCMCIA*. Namun pada saat ini kebanyakan *reader* berhubungan dengan PC melalui *port USB (Universal Serial Bus)*.



Gambar 2.6. *Smart Card Reader ACR 38*

2.4 Komponen yang Digunakan

1. ActiveX

ActiveX adalah standar yang memungkinkan komponen perangkat lunak untuk berinteraksi dengan satu sama lain dalam lingkungan jaringan, tanpa memandang bahasa yang digunakan untuk membuatnya. Sebagian besar pengguna World Wide Web (WWW) akan mengalami *ActiveX* teknologi dalam bentuk kontrol ActiveX, dokumen *ActiveX* dan skrip *ActiveX*.

a. Kontrol *ActiveX*

Kontrol *ActiveX*, sebelumnya dikenal sebagai kontrol OLE atau kontrol OCX, adalah komponen (atau objek) bahwa Anda dapat memasukkan ke halaman Web atau program lainnya sehingga Anda dapat menggunakan kembali dikemas fungsionalitas yang seseorang diprogram. Sebagai contoh, kontrol ActiveX yang disertakan dengan Internet Explorer memungkinkan Anda untuk meningkatkan halaman Web Anda dengan fitur canggih format dan animasi.

Manfaat utama dari kontrol ActiveX melalui program Java dan plug-in Netscape adalah bahwa kontrol ActiveX juga dapat digunakan dalam program-program yang ditulis dalam banyak bahasa pemrograman, termasuk semua bahasa pemrograman dan pangkalan data Microsoft. (<https://support.microsoft.com>, 2016)

b. *ActiveX* dokumen

Saat Anda menjelajah dengan browser Web ActiveX-aware, seperti Internet Explorer ActiveX dokumen memungkinkan Anda untuk membuka program dengan Toolbar dan menu yang tersedia. Ini berarti Anda dapat membuka berkas non-IHTML, seperti berkas Microsoft Excel atau Microsoft Word, menggunakan penjelajah Web mengetahui ActiveX.

c. *ActiveX Scripting*

ActiveX scripting mendukung bahasa scripting paling populer, termasuk Microsoft Visual Basic Script dan JavaScript. ActiveX scripting dapat digunakan untuk menggabungkan perilaku beberapa kontrol ActiveX atau Java program dari Web browser atau server, memperluas fungsi mereka.

2. *Smart Card ToolSet PRO*

Aplikasi smart card toolset pro mampu memproses pengiriman APDU dari ke smart card. Aplikasi ini berkerja dengan menggunakan Winscard.dll yang mengirimkan APDU ke CAP. Jika Anda ingin bekerja dengan Smart Card ToolSet PRO melalui antarmuka COM, anda hanya perlu drag komponen server dari palet komponen Delphi dan menempatkannya pada form. Anda dapat mengakses server acara menggunakan komponen event. Kemudian anda juga dapat mengakses server metode menggunakan metode komponen yang sama.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis

3.1.1 Sistem Saat Ini

Warnet adalah tempat kebanyakan orang mengakses internet. Di negara-negara atau daerah-daerah maju sudah sangat banyak yang menggunakan jasa warnet untuk mengakses internet. Namun umumnya warnet yang ada sekarang dalam melakukan transaksi itu rata-rata masih manual. Pada saat pengguna selesai menggunakan komputer pada warnet, pengguna akan melakukan transaksi pembayaran dengan mendatangi penjaga warnet atau *admin* secara langsung dan menunggu pengembalian uang jika uang yang dibayarkan lebih dari jumlah harga sewa.

3.1.2 Sistem Yang Akan Dibangun

Pada skripsi ini penulis akan memilih mengimplementasikan smart card sebagai pengganti transaksi manual dari pembayaran warnet yang diharapkan dapat menambah efisiensi bagi masyarakat serta penjaga warnet sebagai admin dan juga nantinya para member yang menggunakan smart card akan mendapatkan bonus minuman atau bonus saldo ketika menambah berapa saldo tergantung yang di tentukan admin. Spesifikasi perangkat lunak yang akan dibangun adalah sebagai berikut:

1. Pada admin terdapat *form billing, form pelanggan, form saldo*.
2. *Smart card reader* terletak pada komputer admin.
3. Data yang ada pada kartu adalah *id* pelanggan yang akan mewakili data pada *database* pelanggan.
4. Pendaftaran member dilakukan dengan mendaftarkan pada admin untuk mendapatkan *id* pelanggan dan saldo.
5. Pengisian saldo peanggan dilakukan dengan mendatangi admin.
6. Pelanggan yang selesai menggunakan jasa warnet akan langsung meninggalkan warnet tanpa datang lagi pada admin atau penjaga warnet.

3.1.3 Spesifikasi Hardware dan Software.

Untuk mendukung sistem yang dibuat, maka diperlukan *hardware* dan juga *software* yang diperlukan. Berikut ini adalah spesifikasi *hardware* yang digunakan:

1. Hardware

- | | |
|----------------------|---------------------------|
| a. Processor | : Intel Dual Core 2.0 Ghz |
| b. Harddisk | : 500 GB |
| c. Memori | : 2 GB RAM. |
| d. Input | : Keyboard & Mouse |
| e. Smart Card | : ISO 7816 |
| f. Smart Card Reader | : ACR 38 |

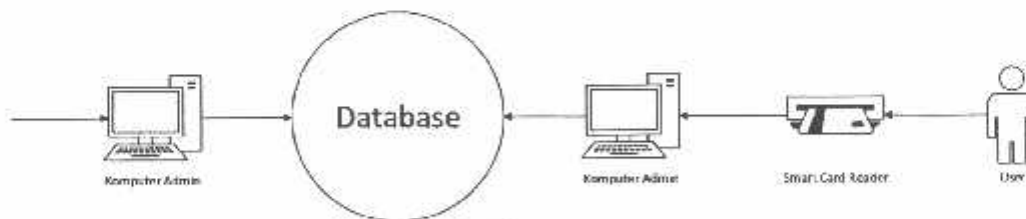
2. Software

- | | |
|-------------------|---------------------------------|
| a. Sistem Operasi | : Windows 10 (32bit atau 64bit) |
| b. Pengembang | : Borland Delphi 7.0 |

3.2 Perancangan

3.2.1 Blok Diagram Sistem

Gambar 3.1 merupakan gambaran desain sistem yang akan dibuat dimana aplikasi komputer *admin* digunakan untuk pendataan, pengisian saldo, dan pengecekan data, sedangkan *smart card* digunakan untuk penyimpanan id pelanggan dan *Smart Card Reader* digunakan untuk membaca *id* yang terdapat pada *Smart Card*.



Gambar 3.1. Blok Diagram Sistem

Penjelasan Gambar 3.1:

Admin – Komputer Admin – Database :

Admin dapat melakukan penginputan data untuk pelanggan baru, penambahan saldo pelanggan jika pelanggan menambah saldo yang akan

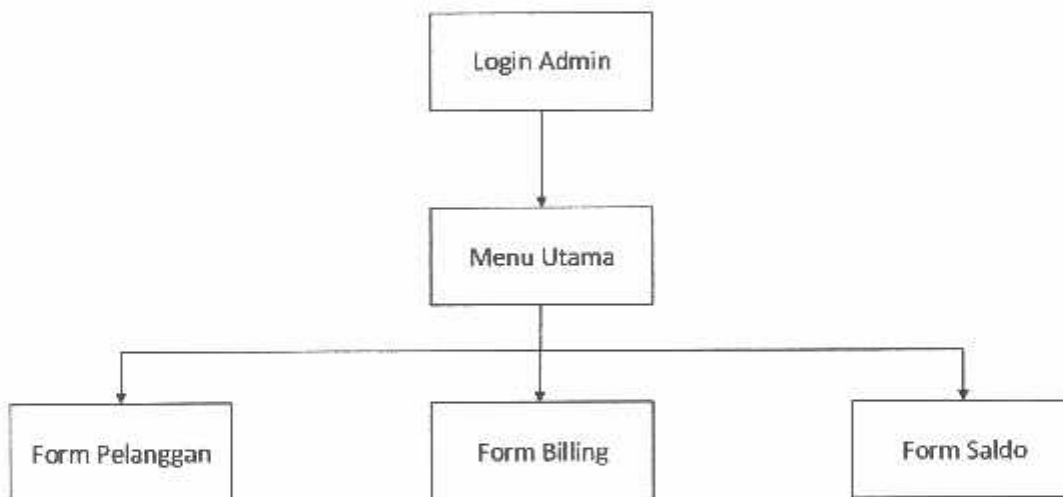
disimpan pada *database*, dan pengoperasian *interface Billing* pada komputer *admin*.

User – Smart Card – Komputer Admin – Database :

User akan memasukkan *smartcard* ke *smartcard reader* yang ada pada *server* dan terhubung pada komputer *admin* kemudian *billing* akan dimulai dari *server* sampai *user* selesai dan saldo yang dimiliki akan terpotong dari saldo *user* dikurangi penggunaan komputer *user* kemudian data pada *database* akan *terupdate*.

3.2.2 Desain Struktur Menu

Dalam aplikasi ini terdapat *menu-menu* yang ada pada *Admin*. Penulis mencoba menggambarkan *menu-menu* tersebut ke dalam sebuah gambar struktur *menu* seperti pada gambar 3.2.



Gambar 3.2. Desain struktur menu

Penjelasan Gambar 3.2:

Login Admin: Merupakan *form* tampilan untuk *login admin* pada awal aplikasi dibuka.

Menu Utama: Merupakan *form* pilihan menu yang terdiri dari *Form* Pelanggan, *Form Billing*, dan *Form Saldo*.

Form Pelanggan: Merupakan *form* yang digunakan untuk melakukan *input data* pelanggan atau *member*.

Form Billing: Merupakan *form* yang menampilkan *list* dari pengguna komputer yang ada pada warnet berisi waktu, durasi, dan biaya.

Form Saldo: Merupakan *form* yang digunakan untuk menambahkan saldo dari pelanggan atau *member*.

3.2.3 Flowchart Sistem

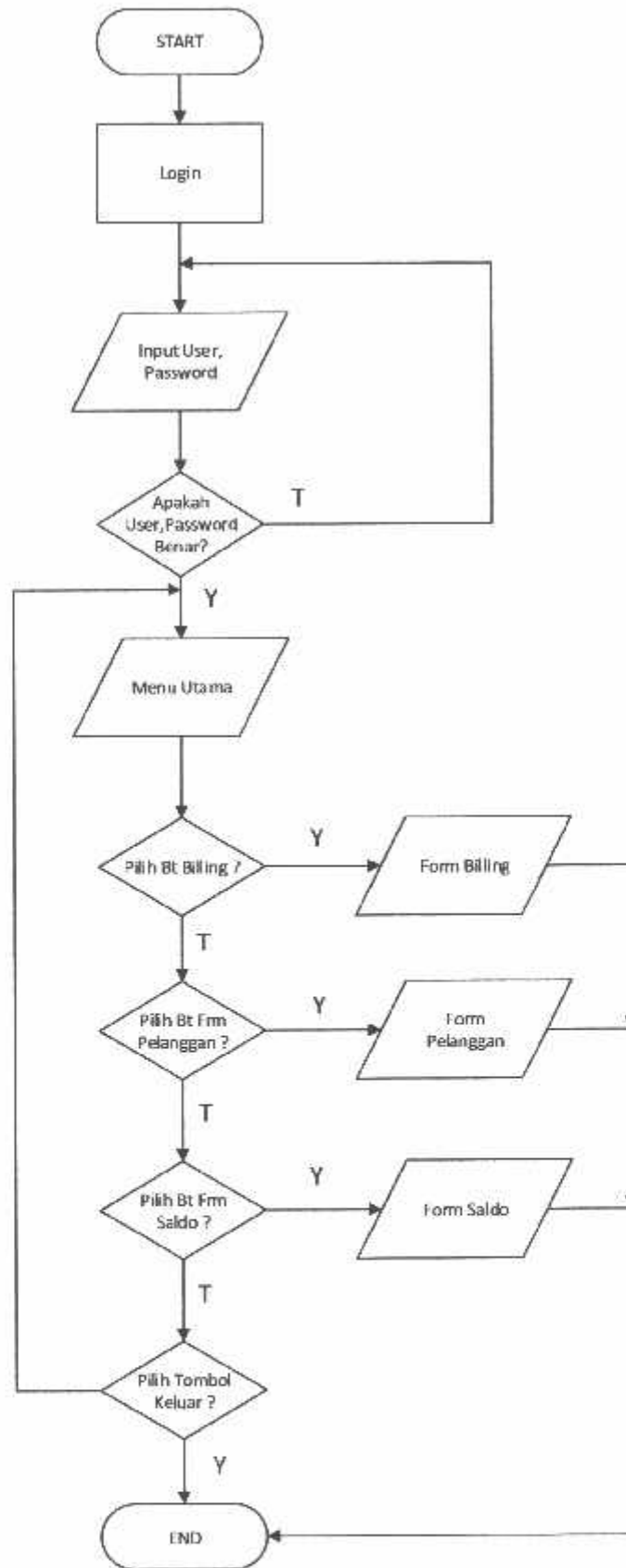
Bagan alir sistem (*flowchart*) adalah bagan-bagan yang mempunyai arus dan menggambarkan beberapa langkah-langkah penyelesaian suatu masalah. Adapun tujuan memakai bagan alir *system (flowchart)*:

1. Menggambarkan suatu tahapan penyelesaian masalah.
2. Secara sederhana, terurai, rapi dan jelas.
3. Menggunakan simbol-simbol standar.

Berikut merupakan penjelasan dan flowchart dari *aplikasi billing smartcard* pada Gambar 3.3.

Penjelasan Gambar 3.3

Pada awal program dimulai, admin akan diharuskan melakukan *login* dengan memasukkan *username* dan *password admin*, jika salah memasukkan *username* atau *password* maka akan diharuskan memasukkan data lagi dengan benar. *Form billing* digunakan sebagai billing dengan *client*. *Form Pelanggan* digunakan untuk menambah data pelanggan. *Form saldo* untuk menambah saldo pelanggan.



Gambar 3.3. Flowchart

3.2.4 Desain Interface

Design Interface adalah desain untuk komputer, peralatan, mesin, perangkat komunikasi *mobile*, aplikasi perangkat lunak, dan situs web yang berfokus pada pengalaman dan interaksi pengguna.

a. *Form Login*

Pada *form* ini akan menampilkan kotak berisi *Username* dan *Password* digunakan sebagai kunci masuk admin pada awal membuka aplikasi ini seperti pada gambar 3.4.



The image shows a rectangular box representing a login form. At the top center, the text "FORM LOGIN ADMIN" is displayed in a bold, sans-serif font. Below this title, there are two rows of input fields. The first row is labeled "Username" on the left and has a single-line text input field on the right. The second row is labeled "Password" on the left and has a single-line text input field on the right. Below the input fields, there are two buttons side-by-side. The left button is labeled "Login" and the right button is labeled "Keluar".

Gambar 3.4. *Form Login Admin.*

Form login ini terdapat pada *admin* pada awal mula aplikasi dibuka dengan memasukkan *username* dan *password*, kemudian dengan klik tombol *login* untuk masuk dan keluar jika membatalkan jalannya aplikasi.

Pada Gambar 3.5 adalah tampilan saat cek *login*



The image shows a rectangular dialog box. In the center, the text "Login Berhasil" is displayed. Below this text, there is a single button labeled "Ok".

Gambar 3.5. Pemberitahuan Cek *Login*.

Pemberitahuan cek *login* ini tampil ketika *admin* benar atau salah ketika memasukkan *username* dan *password*.

b. Form Menu

Pada gambar 3.6 terdapat tampilan menu yang tersedia untuk admin.

The image shows a rectangular window titled "FORM MAIN MENU". Inside the window, there are four buttons arranged in two rows. The top row contains three buttons: "Billing", "Form Pelanggan", and "Form Saldo". The bottom row contains one button, "Keluar", positioned towards the right side.

Gambar 3.6. *Form Menu*

Form Main Menu terdapat tombol menu pilihan yaitu Billing, Form Pelanggan, Form Saldo dan tombol keluar untuk menutup form dan membatalkan jalannya aplikasi.

c. Form Saldo

Pada gambar 3.7 adalah tampilan form saldo, disini kita bisa menambahkan jumlah saldo pelanggan untuk mendapatkan saldonya.

The image shows a rectangular window titled "FORM SALDO". On the left side, there are three labels: "Nama", "Sisa Saldo", and "Jumlah Tambah". Each label is followed by a horizontal input field. Below these input fields are two buttons: "Tambah" and "Batal". To the right of the input fields and buttons is a large, empty rectangular area.

Gambar 3.7. *Form Saldo*

Form Saldo merupakan form admin menambah saldo ketika pelanggan ingin menambahkan saldonya. Pada tabel yang ada pada form dengan klik 2x kemudian nama pelanggan akan tampil pada label nama, sisa saldo pada label sisa saldo dan mengisi jumlah saldo yang akan ditambahkan kemudian pilih tambah atau Batal.

d. Form Pelanggan

Pada gambar 3.8 kita dapat menambahkan data pelanggan yang menjadi member warnet.

Gambar 3.8. *Form Pelanggan*

Form Pelanggan adalah form dimana data pelanggan dibuat pada form ini, yang harus diisikan adalah id, nama pelanggan, alamat, dan saldo. Untuk tombol terdapat banyak tombol yaitu Cari, Tambah, Simpan, Hapus, Batal, Ubah, Reset, dan Keluar.

e. Form Billing

Pada Gambar 3.9 merupakan form billing dimana ini adalah pusat kontrol dan pemantauan pengguna yang sedang aktif.

Gambar 3.9. *Form Billing*

Form Billing yang merupakan *form* dimana *admin* dapat mengatur biaya permenit warnet, mengetahui *client* aktif beserta kalkulasi waktu dan biaya.

f. **Form Billing Client**

Form ini merupakan tampilan dari komputer yang berada pada *client* dan hanya tampil berapa waktu dan biaya penggunaanya seperti pada gambar 3.10.

Gambar 3.10. *Form Billing Client*

Form Client adalah *interface* pada *client* yang digunakan untuk memulai *billing* dan masuk *desktop* agar dapat digunakan untuk mengoperasikan internet. Pada *form* ini terdapat pengisian *username*, tombol mulai, selesai, keluar, dan informasi waktu mulai, waktu selesai, durasi pemakaian, dan biaya pemakaian.

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Dalam bab ini akan dijelaskan mengenai hasil uji coba perangkat *Smart Card*. Setelah itu hasil uji coba yang telah dilakukan, akan dianalisa apakah rancangan ini dapat memenuhi tujuan yang akan dicapai seperti yang dipaparkan pada Bab 1.

4.1 Implementasi

Merupakan penerapan dari aplikasi *Smart Card* berisi tampilan – tampilan *form* beserta konfigurasi *Smart Card* yang ada dalam aplikasi *billing* ini.

1. *Form Login*

Halaman ini merupakan halaman yang pertama kali tampil pada saat *admin* menjalankan aplikasi ini seperti pada gambar 4.1.



Gambar 4.1. Tampilan *Login Admin*

Awal aplikasi *running*, *Admin* mengisikan *user* dan *password* untuk *login* aplikasi.

2. *Login Sukses*

Pemberitahuan akan tampil “Login Berhasil” jika admin berhasil masuk seperti pada Gambar 4.2



Gambar 4.2. *Login Berhasil*

Pemberitahuan menampilkan informasi login berhasil setelah mengisi username dan *password* benar.

3. *Login Gagal Password*

Jika admin salah memasukkan password maka akan tampil seperti pada gambar 4.3.

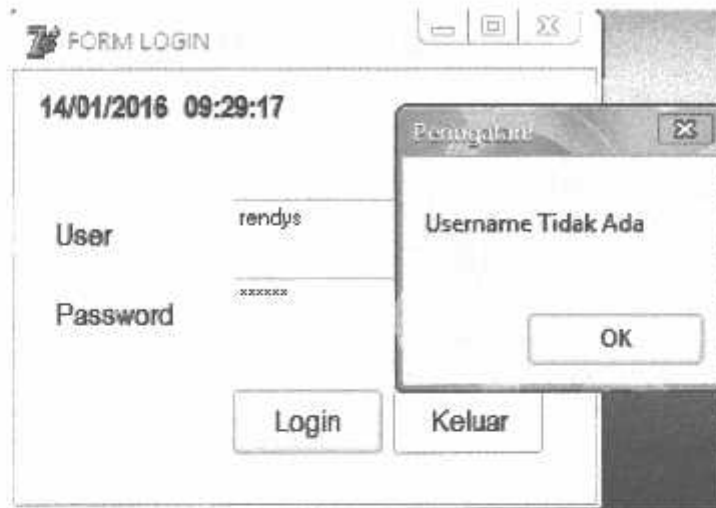


Gambar 4.3 *Login Salah Password*

Peringatan muncul saat password salah ketika admin menekan tombol ok maka akan dikembalikan pada form login untuk memasukkan username atau password yang benar untuk melanjutkan masuk aplikasi.

4. *Login Salah Username*

Jika *admin* salah memasukkan *username* maka akan muncul pemberitahuan seperti pada gambar 4.4.



Gambar 4.4. *Login Salah Username*

5. *Form Menu*

Gambar 4.5. Pada *form* ini menampilkan *menu* yang akan dipilih oleh *admin*.



Gambar 4.5. *Form Menu*

Pada *form* ini *admin* memilih *form* pelanggan untuk menampilkan *form* selanjutnya.

6. *Form Pelanggan (Tambah)*

Gambar 4.6. *Form* ini berfungsi untuk menambahkan data *member* warnet.

13/12/2018 18:56:17

ID: 5, Nama: Doni, Alamat: Batu, Saldo: 15000, No HP: 087564287198

Buttons:

ID	NAMA	ALAMAT	SALDO	NO_HP
1	Randy	Batu	8300	0817942
2	Rizki	Batu	10000	0814284
3	Clara	Medang	9000	0827788
4	Milani	Medang	80000	0877948

Gambar 4.6. *Form Tambah*

Admin mengisikan Id = 5, Nama = Doni, Alamat = Batu, Saldo = 15000, No HP = 087564287198. Kemudian memilih tombol simpan.

7. *Form Pelanggan (Simpan)*

Gambar 4.7 Merupakan data pelanggan yang sudah disimpan.

13/12/2018 18:57:29

ID: 5, Nama: Doni, Alamat: Batu, Saldo: 15000, No HP: 087564287198

Buttons:

ID	NAMA	ALAMAT	SALDO	NO_HP
1	Randy	Batu	8300	0817942
2	Rizki	Batu	10000	0814284
3	Clara	Medang	9000	0827788
4	Milani	Medang	80000	0877948
5	Doni	Batu	15000	0875642

Gambar 4.7. *Form Pelanggan Simpan*

Data yang sudah diisikan pada gambar 4.6 akan tersimpan pada *database* dan ditampilkan pada *list view* seperti gambar 4.7.

8. *Form Pelanggan (Ubah)*

Gambar 4.8 Merupakan data pelanggan yang sedang diubah data pada nama pelanggan.

12/12/2016 18.08.28

Id: 1
 Nama: Doni Arodo
 Alamat: Batu
 Saldo: 10000
 No HP: 08175042

Tambah Simpan Ubah
 Batal Hapus Reset Keluar

ID	NAMA	ALAMAT	SALDO	NO_HP
1	Renly	Batu	0000	0817041
2	Rio	Batu	10000	0818284
3	Quin	Ubung	0000	0817700
4	Ikram	Ubung	50000	0817700
5	Doni	Batu	10000	08175042

Gambar 4.8. *Form Pelanggan Ubah*

Admin mengubah pelanggan dengan nama Doni dengan cara klik 2x, data akan tampil pada *textbox* kemudian nama Doni diubah menjadi Doni Arodo.

9. *Form Pelanggan (Update Selesai)*

Gambar 4.9 Merupakan data pelanggan yang berhasil diubah data namanya.

12/12/2016 18.08.02

Id: 1
 Nama: Doni Arodo
 Alamat: Batu
 Saldo: 10000
 No HP: 08175042

Tambah Simpan Ubah
 Batal Hapus Reset Keluar

ID	NAMA	ALAMAT	SALDO	NO_HP
1	Renly	Batu	0000	0817041
2	Rio	Batu	10000	0818284
3	Quin	Ubung	0000	0817700
4	Ikram	Ubung	50000	0817700
5	Doni Arodo	Batu	10000	08175042

Gambar 4.9. *Form Pelanggan Update*

Data yang sebelumnya bernama Doni telah berubah menjadi Doni Arodo dan ditampilkan pada *list view* seperti pada gambar 4.9.

10. *Form Pelanggan (Hapus)*

Gambar 4.10 Adalah data yang terpilih akan dihapus dan data tidak akan tampil lagi pada tabel *list view* pelanggan

13/12/2016 19.00.20

Id: 5
 Nama: Doni Arodo
 Alamat: Duku
 Saldo: 10000
 No HP: 08155438715E

Tambah Simpan Ubah
 Batal Hapus Reset

ID	NAMA	ALAMAT	SALDO	NO_HP
1	Reandy	Batu	6000	0897543
2	Riba	Batu	10000	0818284
3	Quati	Malang	9000	08947760
4	Rizwan	Malang	80000	0827648
5	Doni Arodo	Batu	15000	0879443

Keluar

Gambar 4.10. Form Pelanggan Hapus

Pemilihan pada *list view* untuk data pelanggan Doni Arodo ketika di klik 2x, data akan tampil pada *textbox* dan terdapat pilihan tombol Ubah, Hapus, Batal, Reset.

11. Konfirmasi Hapus

Gambar 4.11 Merupakan konfirmasi jika data akan terhapus.

Konfirmasi

Data Akan Terhapus

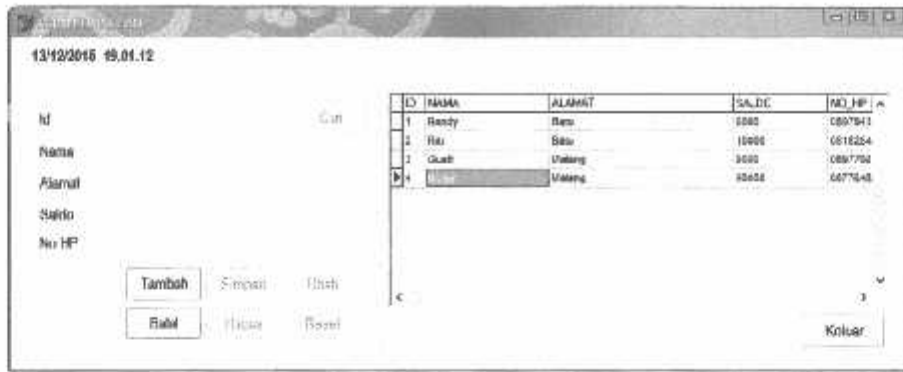
Yes No

Gambar 4.11. Konfirmasi Hapus

Pesan ini akan tampil ketika akan menghapus data Doni Arodo untuk meyakinkan *admin* yang akan menghapus data, jika dipilih "yes" data akan terhapus jika "no" penghapusan akan dibatalkan.

12. Member Terhapus

Gambar 4.12 Merupakan data pelanggan yang sebelumnya sudah terhapus dan tidak tampil lagi.



Gambar 4.12. Data Pelanggan Terhapus

Data untuk pelanggan Doni Arodo yang dihapus oleh admin sudah terhapus pada tabel dan *database*.

13. *Form* Menu Utama (*Form* Tambah Saldo)

Gambar 4.13 Merupakan tampilan menu utama dan akan dipilih pada *Form* Tambah Saldo.



Gambar 4.13. *Menu* Utama

Pada menu utama ini *admin* memilih tombol *Form* Tambah Saldo untuk menampilkan *form* selanjutnya.

14. *Form* Saldo

Gambar 4.14 Merupakan tampilan nilai saldo yang akan ditambahkan.

13/12/2016 19.03.32

Nama Rendy

Sisa Saldo 6000

Jumlah Tambah

ID	NAMA	ALAMAT	SALDO
1	Rendy	Batu	6000
2	Riki	Batu	10000
3	Gusti	Malang	9000
4	Nizam	Malang	90000

Gambar 4.14. Tambah Saldo

Admin menambahkan saldo dari pelanggan Rendy, kemudian ditambahkan Jumlah Tambah 1000.

15. *Form Saldo (Tambah)*

Gambar 4.15. Merupakan tampilan data yang sudah bertambah saldonya setelah diinputkan jumlah data tambahannya.

13/12/2016 19.04.28

Nama Rendy

Sisa Saldo

Jumlah Tambah

ID	NAMA	ALAMAT	SALDO
1	Rendy	Batu	7000
2	Riki	Batu	10000
3	Gusti	Malang	9000
4	Nizam	Malang	90000

Gambar 4.15. Saldo Bertambah

Saldo dari pelanggan Rendy yang sebelumnya 6000 menjadi 7000 setelah ditambahkan 1000.

16. *Form Saldo (Tambah Ketika Textbox Kosong)*

Pada gambar 4.16 adalah ketika data belum dilengkapi dan menekan tombol Tambah Saldo.

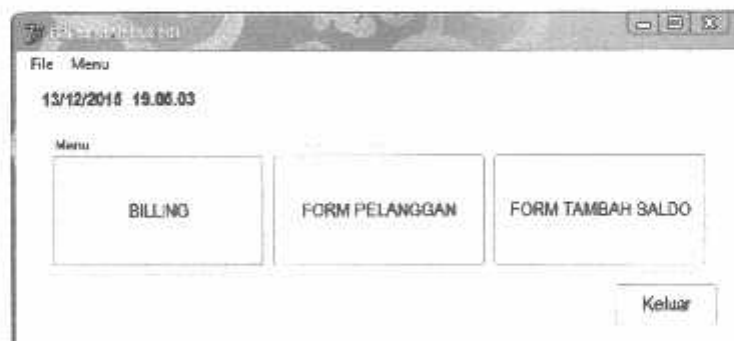


Gambar 4.16. Data Belum Lengkap

Peringatan muncul ketika data belum terlengkapi dan menekan tombol Tambah Saldo, kemudian dapat diulangi dengan menekan tombol *Ok*.

17. *Form Menu (Billing)*

Gambar 4.17 Merupakan tampilan menu utama yang akan terpilih pada *Form Billing*.



Gambar 4.17. Pilih *Billing*

Pada *form main menu* ini *admin* memilih *Billing* untuk ditampilkan pada form selanjutnya.

18. Form Billing

Gambar 4.18 Merupakan tampilan dari *form billing* dimana daftar *client* yang aktif dan *standby* akan tampil sekaligus kalkulasi biaya dan waktu sesuai konfigurasi biaya permenit dari *admin* dan penggunaan komputer *client*.

No Komp	Username	Waktu Mulai	Waktu Selesai	Durasi	Biaya	Status
Client-5	Kosong	00:00:00	00:00:00	00:00:00	Rp.0	Offline

Gambar 4.18. *Form Billing*

Pada *form billing* akan menampilkan pengaturan biaya permenit dan daftar *client* secara keseluruhan yang sedang aktif dan tidak. Pada *list view* ini terdapat satu komputer yang *standby* dan belum *online* sehingga tampilan masih berupa "*Username*" yang bertuliskan teks "Kosong", "Waktu Mulai 00:00:00", "Durasi 00:00:00", "Biaya Rp.0", dan berstatus "*Offline*" seperti pada gambar 4.18.

19. Form Client

Gambar 4.19 Adalah sebuah tampilan yang berada pada komputer *client* masing-masing. Yang berfungsi sebagai tampilan pertama sebelum pengguna dapat menggunakan koneksi internet pada komputer agar pengguna diharuskan mengisi nama dan memulai waktu sewa warnet. *Form client* terdapat *textbox username* untuk mengisi nama *client*, tombol mulai, selesai, keluar. Dan informasi Waktu Mulai, Selesai, Durasi, dan Biaya.

Tanggal 13/12/2016 Jam 19.09.03

Username

internet billing 

Detail

Mulai : 00.00.00

Selesai : 00.00.00

Durasi : 00.00.00

Biaya : 0

Gambar 4.19. Form *Client*

20. *Form Client (Mulai)*

Gambar 4.20 Adalah tampilan *form client* setelah *running*.



Tanggal 14/01/2016 Jam 09:51:48

Username

internet billing 

Detail

Mulai : 09:51:44

Selesai : 09:51:48

Durasi : 00:00:04

Biaya : 60

Gambar 4.20. Form *Client* Mulai

21. *Form Billing (Client Mulai)*

Gambar 4.21 Merupakan tampilan yang pada billing *admin* setelah *client* 6 *running* tampil daftar kalkulasi biaya dan waktu.

The screenshot shows a web browser window titled "internet billing" with a date and time of 14/01/2016 09:52:43. The main heading is "internet billing" with a logo of three overlapping cards. Below the heading, there is a "Biaya Per Menit" field with the value "50" and a "Set" button. A table displays client information:

No Komp	Username	Waktu Mulai	Waktu Selesai	Durasi	Biaya	Status
Client-6	Rendy	09:51:44	09:52:43	00:00:59	50	Online

At the bottom right, there is a "Disediakan: 50" label and a "Keluar" button.

Gambar 4.21. *Form Billing (Client Mulai)*

Setelah *client running* maka pada *billing admin* akan tampil pada *list view client online* beserta *username* waktu mulai, durasi, dan biaya.

22. *Form Client (Selesai)*

Pada saat *client* selesai akan kembali pada tampilan seperti pada gambar 4.22 dengan informasi kalkulasi waktu dan biaya yang tertera.

The screenshot shows a web browser window titled "internet billing" with a date and time of Tanggal 14/01/2016 Jam 09:53:44. The main heading is "internet billing" with a logo of three overlapping cards. Below the heading, there is a "Username" field with the value "Rendy" and buttons for "Mula" and "Selesai". A "Detail" section displays client information:

Mulai	: 09:51:44
Selesai	: 09:53:39
Durasi	: 00:01:55
Biaya	: 95

Gambar 4.22. *Client Selesai*

23. Form Billing (Client Selesai)

Pada saat *client* 6 selesai kalkulasi akan berhenti dan muncul waktu selesai dan status *offline* seperti pada gambar 4.23



Gambar 4.23. Tampilan *Billing* Saat *Client* 6 selesai

Client selesai waktu yang di kalkulasi berhenti sehingga menampilkan Waktu Mulai, Waktu Selesai, Durasi, Biaya, Dan Status.

24. Form Client (Standby)

Kondisi *client* yang sedang *stanby* sebelum *client* mulai running seperti pada gambar 4.24



Gambar 4.24. *Client Standby*

25. Form Billing (Start Client)

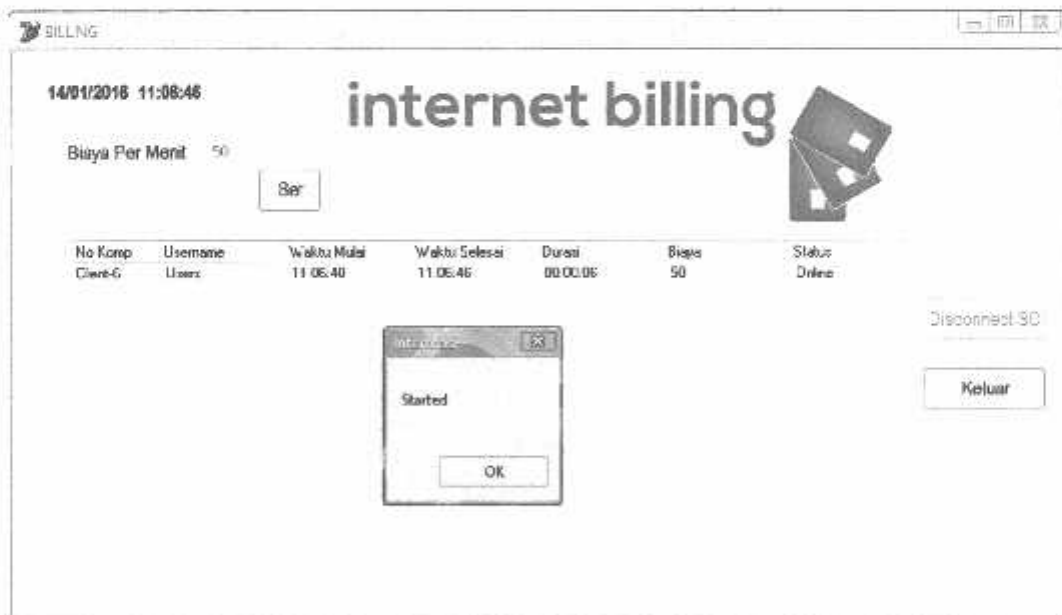
Pada Gambar 4.25 *admin* memulai *client* yang langsung meremote *client* memulai *billing*.



Gambar 4.25. Start Client Dari Admin

26. Form Billing (Started Client)

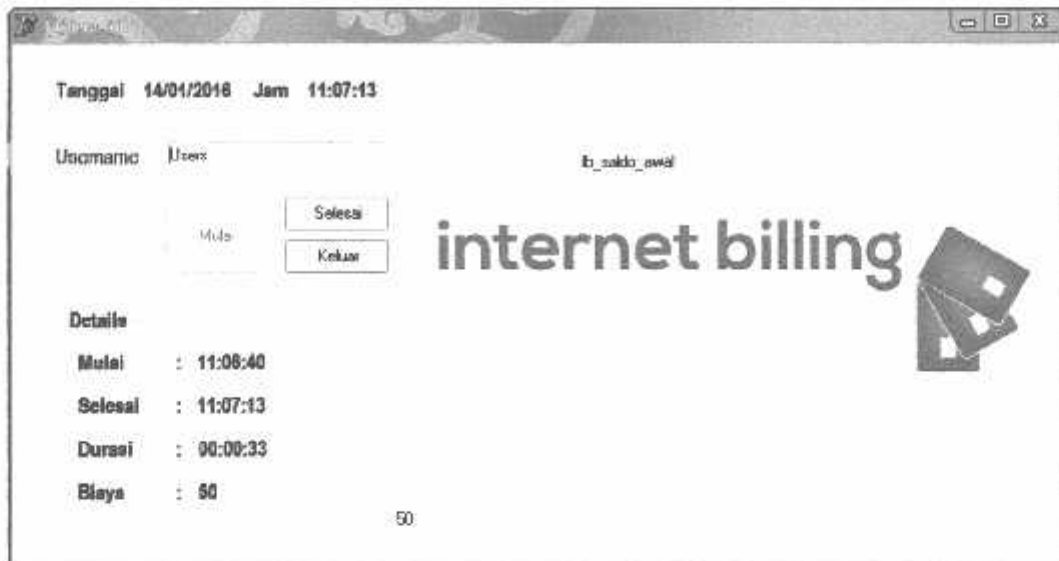
Setelah *admin* memulai *client* akan muncul tampilan jika *client* sudah berjalan terlihat seperti pada gambar 4.26.



Gambar 4.26. Client Started

27. *Form Client (Start Dari Admin)*

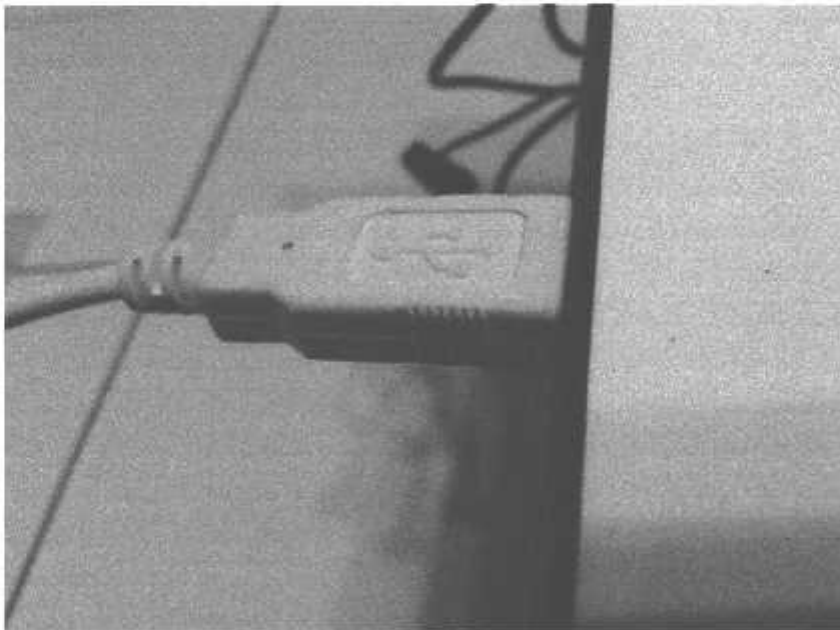
Setelah *admin start client*, pada *client* akan langsung mulai otomatis dan *username* otomatis terisi *Users* seperti pada gambar 4.27.



Gambar 4.27. Start Client Dari Admin

28. *USB Smart Card Reader Di Masukkan*

Pada gambar 4.28 *USB Smart Card Reader* di masukkan untuk *member* yang menggunakan *smart card*.



Gambar 4.28. USB Smart Card Reader Masuk

29. *Form Billing (Admin Start Member)*

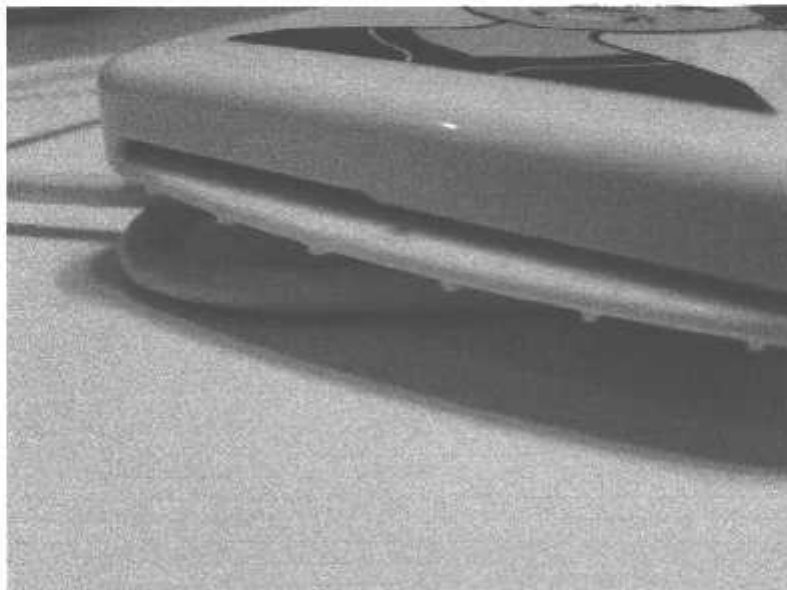
Admin start member yang akan menggunakan *smart card* seperti pada gambar 4.29 yang nantinya pelanggan di berikan komputer yang kosong dan *smart card* akan dimasukkan untuk memulai komputer *client*.



Gambar 4.29. *Admin Start Member*

30. *Smart Card Reader Belum Terpasang Dengan Smart Card*

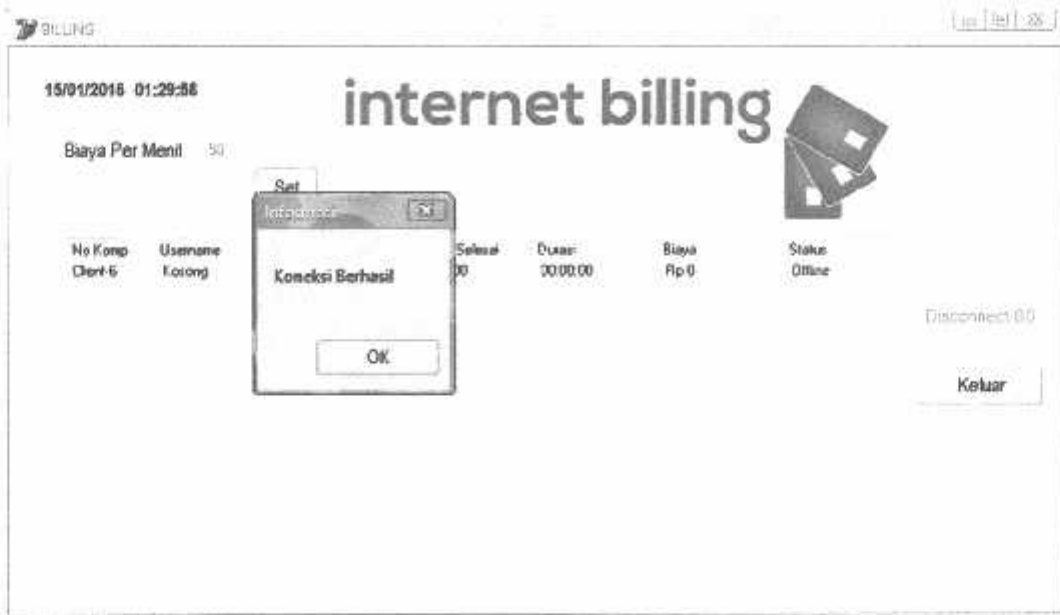
Pada gambar 4.30 terlihat kondisi *smart card reader* yang belum terpasang dengan *smart card* namun hanya terpasang *USB* dengan komputer admin.



Gambar 4.30. *USB Smart Card Reader* Belum Masuk

31. Form Billing (Admin Start Member)

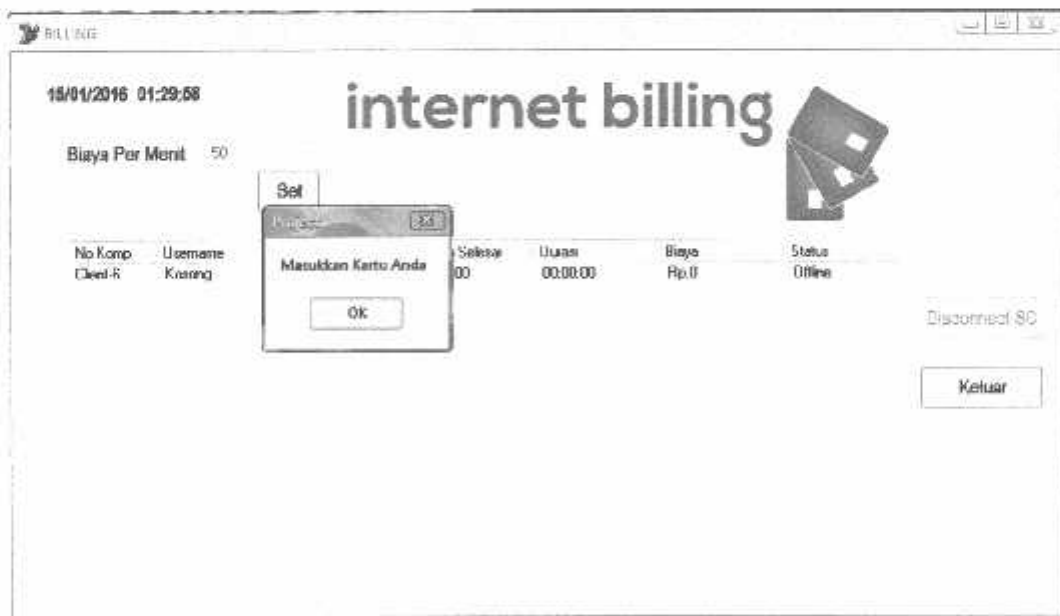
Setelah *admin start – member* dan *reader* terpasang, maka tampil informasi jika koneksi dengan *reader* berhasil terhubung seperti pada gambar 4.31.



Gambar 4.31. Koneksi Dengan *Smart Card Reader* Berhasil

32. Form Billing (Admin Start Member)

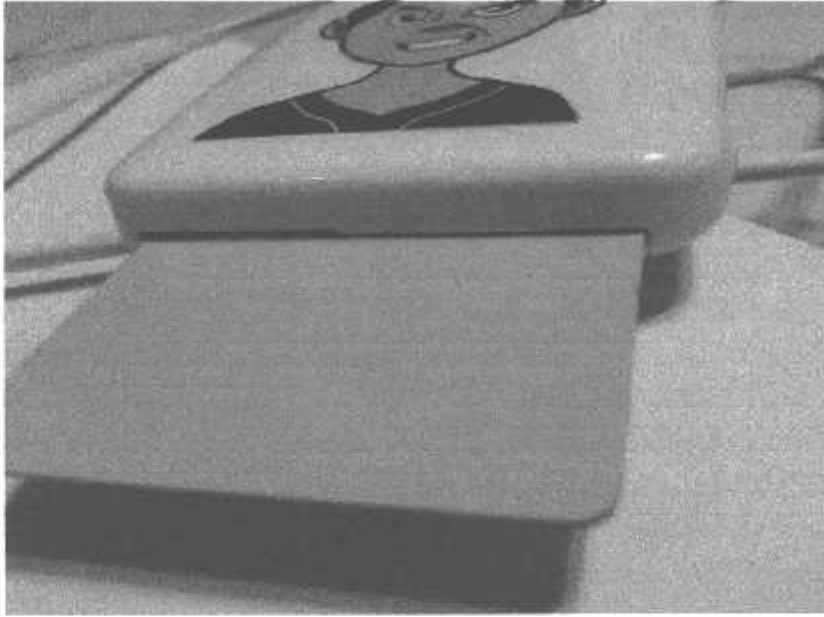
Setelah berhasil terhubung dengan *smart card reader*, akan muncul informasi perintah untuk memasukkan kartu *smart card* seperti pada gambar 4.32



Gambar 4.32. Informasi Untuk Memasukkan Kartu *Smart Card*

33. *Smart Card* Dimasukkan Pada *Smart Card Reader*

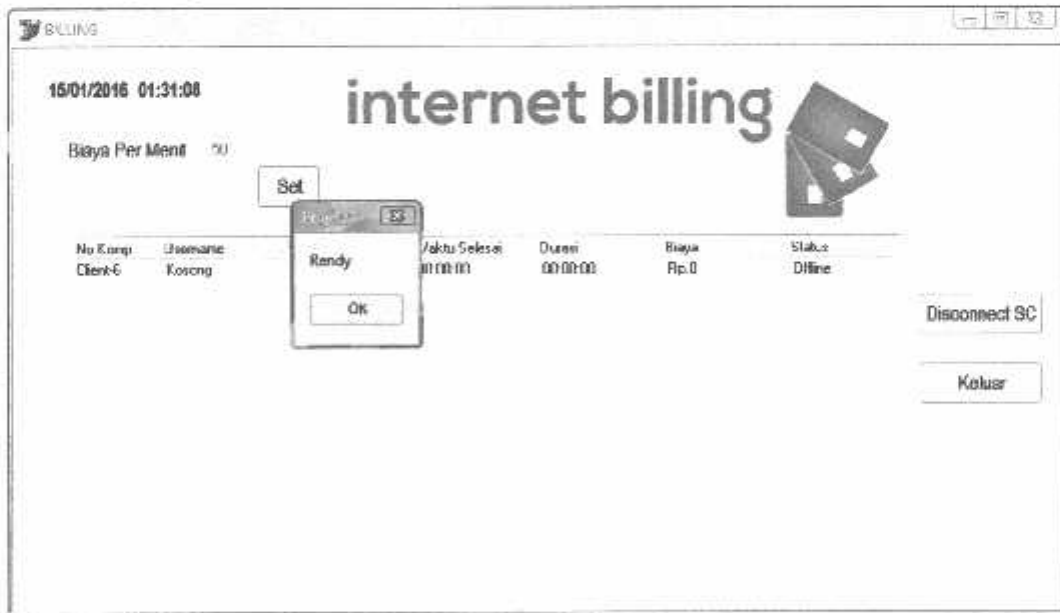
Smart card dimasukkan pada *smart card reader* yang telah terhubung dengan komputer *admin* terlihat seperti pada gambar 4.33.



Gambar 4.33. *USB Smart Card Reader* Masuk

34. *Form Billing* (Informasi Nama *Client*)

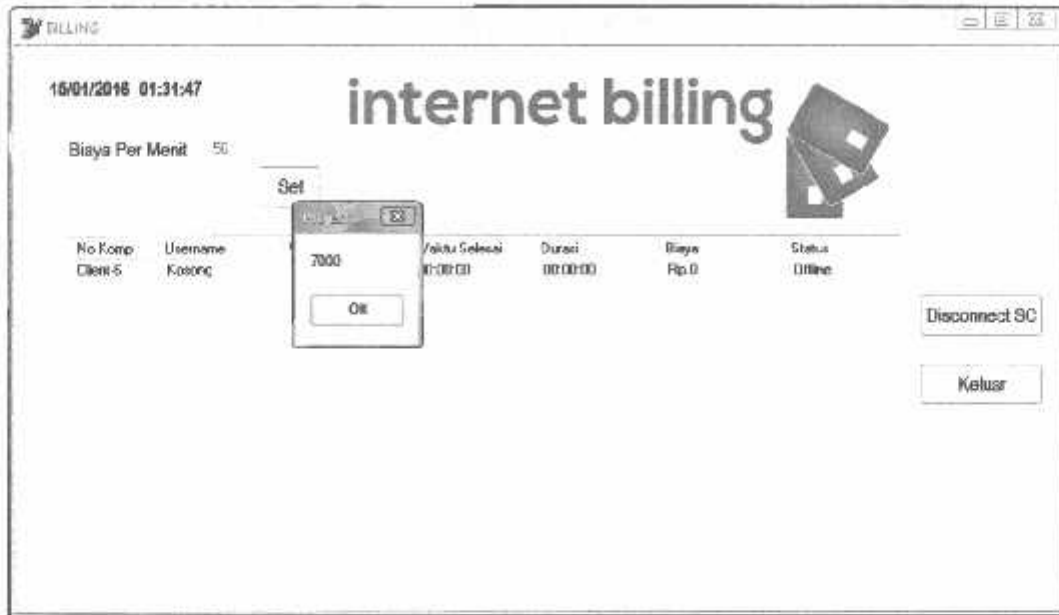
Setelah *smart card* masuk maka akan tampil nama pengguna pada komputer *admin* seperti pada gambar 4.34



Gambar 4.34. Informasi Nama *Client* Pada *Admin*

35. Form Billing (Informasi Saldo Client)

Setelah informasi nama *client* maka akan tampil saldo awal *client* sehingga seperti pada gambar 4.35



Gambar 4.35. Informasi Saldo Awal *Client* Pada *Admin*

36. Form Billing (Client Online)

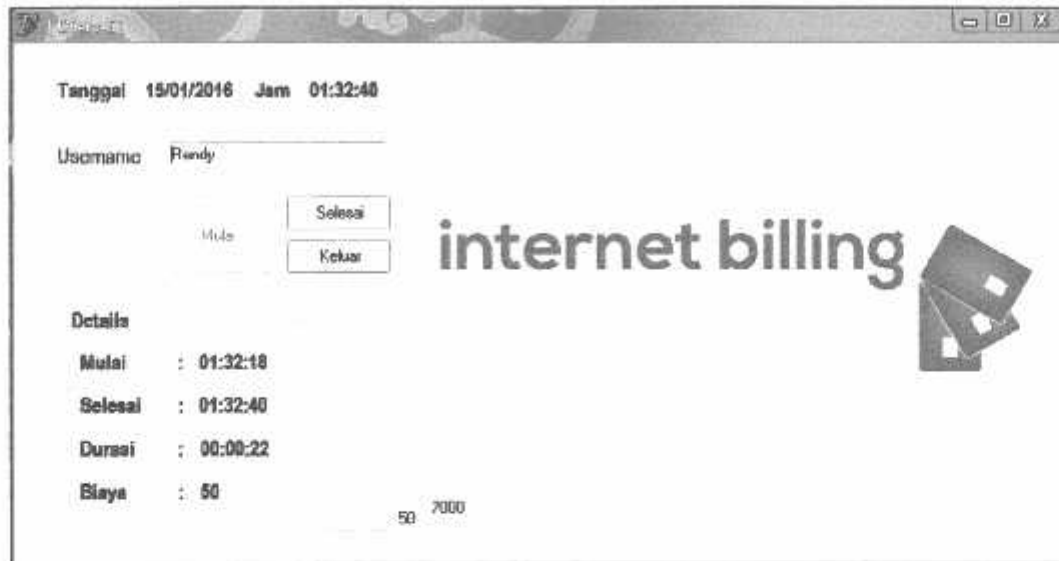
Setelah informasi nama dan saldo *client* tampil maka komputer *client* siap dan sudah *online* seperti pada gambar 4.36



Gambar 4.36. *Client Online*

37. Form Client (Online Smart Card)

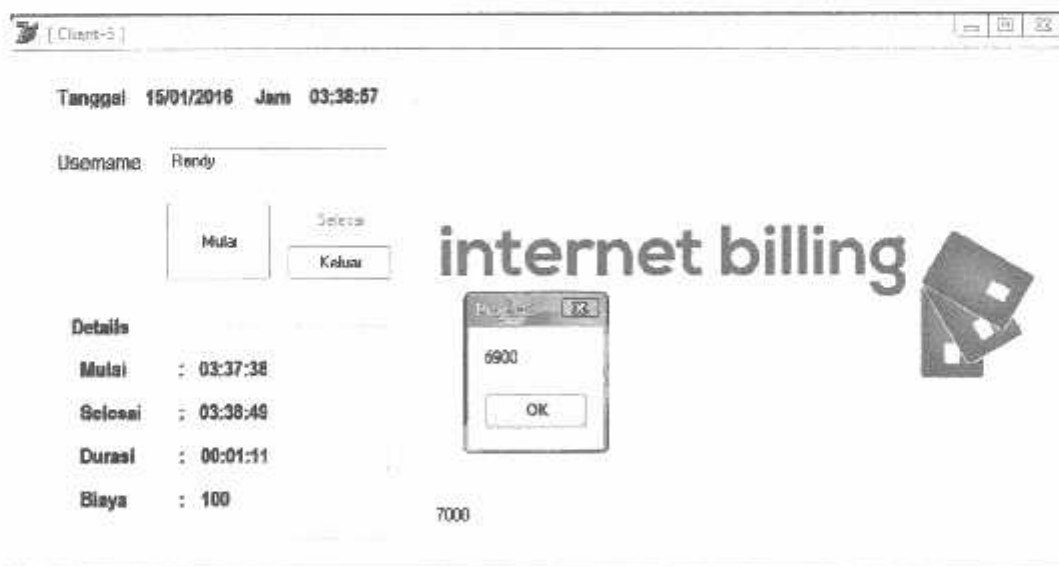
Pada gambar 4.37 *client* sudah siap dan *online* setelah *start member smart card* dari *admin*.



Gambar 4.37. Komputer *Client* Sudah Siap Dan *Online*

38. Form Client (Selesai)

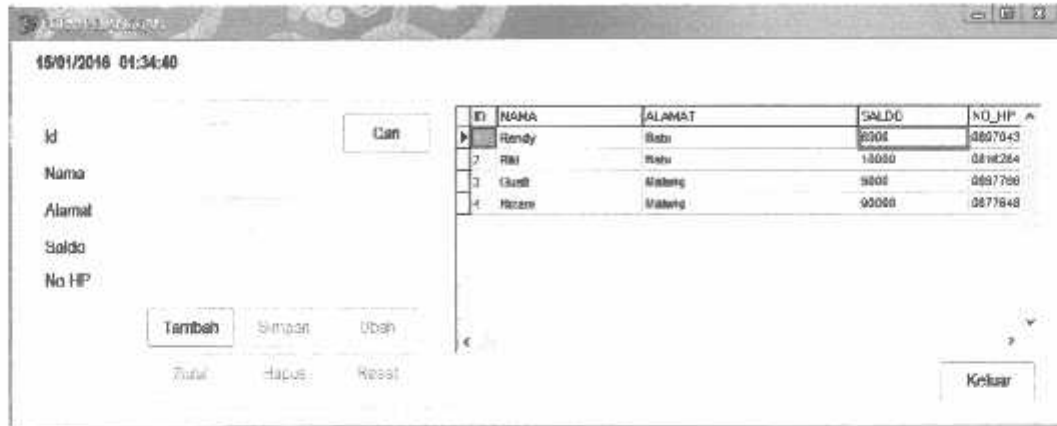
Client sudah selesai dan tampil informasi saldo berkurang dengan sisa berapa seperti pada gambar 4.38



Gambar 4.38. *Client Smart Card* Selesai

39. Form Pelanggan (Saldo Berkurang)

Setelah pelanggan *smart card* selesai maka saldo akan dikurangi sesuai biaya yang telah dikenakan seperti pada gambar 4.39.



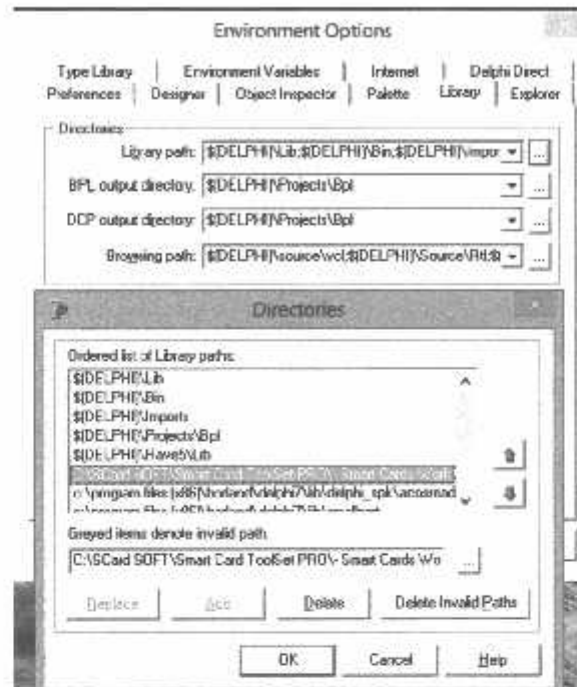
Gambar 4.39. Saldo Pelanggan Berkurang

40. Form Client (Koneksi Terputus)

Pada gambar 4.40 merupakan tampilan yang menandakan peringatan jika koneksi dengan *server* terputus dan *client* akan otomatis *stop*.



Gambar 4.40. Koneksi Dengan *Server* Terputus



Gambar 4.43. Konfigurasi *Path*.

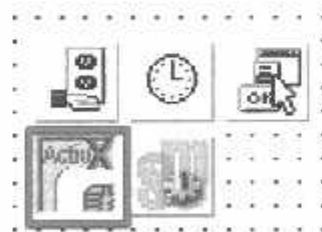
Setelah *component* diinstal pada delphi, akan *component* yang akan digunakan terletak pada *ActiveX* seperti pada gambar 4.44.



Gambar 4.44. *Component* Pada Delphi 7

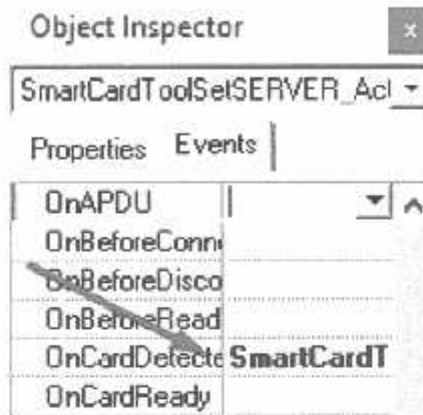
Pilih *ActiveX* kemudian pilih icon centang hijau seperti gambar 4.44.

Kemudian jika *component* di masukkan kedalam *form* akan tampil seperti pada gambar 4.45.



Gambar 4.45. Hasil *Component* Yang Diambil

Kemudian untuk *source code* diketik pada *object inspector* dan *tab events* dengan event “*On Card Detected*” seperti pada gambar 4.46.



Gambar 4.46. *Source Code* Diisikan Pada *Event*

Klik 2x pada *OnCardDetected* dan *code* di isikan didalamnya.

43. Verify, Select, Write, Read pada Smart Card dengan Aplikasi Plugin Smart Card ToolSet PRO.

Buka aplikasi plugin *Smart Card ToolSet PRO*. Selanjutnya klik ikon seperti gambar 4.47. *Connect to the smart card service* agar dapat dibuka *card explorer*.



Gambar 4.47. *Connect to the smart card service*

Kemudian klik ikon seperti pada gambar 4.48. *Open the card explorer window*.



Gambar 4.48. *Open the card explorer window*

Isikan *command* pada *form single APDU* yaitu pertama “*Verify*” seperti gambar 4.49 terdapat “*Cla*” : *Class Reader* yang cocok, “*Ins*” : Kode instruksi

atau command seperti *Verify* yang berisi “P1” : Parameter 1, “P2” : Parameter 2, “Lc” : Panjang karakter Heksadesimal dari *Data In*, *Data In* : Heksadesimal data verifikasi (sudah ditentukan), untuk “Le” dikosongkan saja. Untuk melakukan *command* yang lain hanya mengubah kode heksadesimal “Ins” Instruksi dan *Data In*.

Single APDU		Cl	Ins	P1	P2	Lc	Data In	Hasil
	SEND	80	20	07	00	08	41 43 4F 53 54 45 53 54	<input checked="" type="checkbox"/>

Gambar 4.49. *Command* pada *Single APDU*

Pada gambar 4.50 terdapat *command* melalui *batch file* di bawah *Single APDU* yang sudah diisi *command* dan dapat disimpan mulai dari “*Verify*” (1), “*Select*” (2), “*Write*” (3), dan “*Read*” (4). *Data In* pada “*Select*” – FF 01 merupakan pemilihan alamat data yang akan di isikan. Kemudian “*Data In*” pada “*Write*” = 00 00 00 01 merupakan *heksadesimal* dari angka 1 dan dimasukkan pada alamat data FF 01. Untuk melakukan *read* data hanya mengisi panjang data dari *Data In* atau “*Le*” dengan syarat melakukan *command* “*Select*” pada alamat FF 01 sehingga akan jika melakukan “*Read*” data akan memilih pada data yang berada pada alamat FF 01.

Use	Command	Cl	Ins	P1	P2	Lc	Data IN (Hex)	Le	Comments	
<input checked="" type="checkbox"/>	Verify	1	80	20	07	00	08	41 43 4F 53 54 45 53 54		VERIFY
<input checked="" type="checkbox"/>	Select3	2	80	A4	00	00	02	FF 01		id (Ready)
<input checked="" type="checkbox"/>	Write3	3	80	02	00	00	04	00 00 00 01		id (Ready)
<input checked="" type="checkbox"/>	Read3	4	80	B2	00	00	04			id (Ready)

Gambar 4.50. *Command* melalui *APDU Batch File*.

Pengiriman *command* melalui *Single APDU* juga sama tidak terdapat perbedaan cara pengiriman *command*. Sesuaikan kode *command* yang dibutuhkan seperti pada gambar 4.50

4.2 Pengujian

Pada pengujian ini akan ditampilkan tabel-tabel pengujian berhasil dan berfungsi atau tidak suatu fungsi program, *software*, dan perangkat yang digunakan.

1. Pengujian Fungsional

Tabel Pengujian 4.1. Hasil Pengujian Fungsi

No.	Fungsi	Berhasil	
		Ya	Tidak
1.	Form Login	√	-
	a. Bt_Login Admin	√	-
	b. Bt_Keluar	√	-
2.	Form Menu	√	-
	a. Bt_Billing	√	-
	b. Bt_Pelanggan	√	-
	c. Bt_Saldo	√	-
3.	Form Pelanggan	√	-
	a. CRUD (Create, Read, Update, Delete) Data Pclanggan	√	-
	b. Bt_Pencarian	√	-
	c. Bt_Reset	√	-
	d. Bt_Keluar		-
4.	Form Saldo	√	-
	a. Lv_pclanggan	√	-
	b. Bt_Saldo	√	-
	c. Bt_Batal	√	-
5.	Form Billing	√	-
	a. Bt_Set	√	-
	b. Bt_Keluar	√	-
	c. Lv_Client	√	-
	d. PopUpReguler	√	-
	e. PopUpMember	√	-
	f. PopUpStop	√	-

No.	Fungsi	Berhasil	
		Ya	Tidak
6.	Form Client	√	-
	a. Bt_Mulai	√	-
	b. Bt_Selesai	√	-
	c. Bt_Keluar	√	-
	d. Lb_Mulai	√	-
	e. Lb_Selesai	√	-
	f. Lb_Durasi	√	-
	g. Lb_Biaya	√	-
7.	Form Billing	√	-
	a. Select Data Kartu	√	-
	b. Read Data Kartu	√	-

Dari tabel pengujian fungsional 4.1, diatas menunjukkan semua fungsi pokok meliputi *login*, *logout*, CRUD, koneksi dapat berjalan dengan baik secara keseluruhan.

2. Pengujian *Smart Card Reader ACR38* Dengan Tabel Fungsional

Pengujian *Reader ACS38* dilakukan dengan 4 macam sistem operasi yang diujikan untuk mengetahui tingkat kompatibilitas dari *reader ACS38*. Berikut hasil pengujian pada Tabel 4.2.

Tabel Pengujian 4.2 *Pengujian Smart Card Reader ACR38*

No.	Sistem Operasi	Berfungsi	
		Ya	Tidak
1.	Windows XP	√	-
2.	Windows 7	√	-
3.	Windows 8.1	√	-
4.	Windows 10	√	-

Dari tabel fungsional *Reader ACS38* diatas menunjukkan bahwa *Reader ACS38* memiliki tingkat kompatibilitas yang baik dengan 4 sistem operasi yang diuji cobakan.

3. Pengujian *Compability Smart Card Dengan Smart Card Reader ACR 38 Melalui Sistem Operasi Berbeda.*

Pengujian *Smart Card* dilakukan dengan 2 macam *Smart Card* yaitu tipe *ISO 7816* dan *SLE 4442* serta dengan menggunakan Bahasa pemrograman *Delphi 7* dan *Reader ACS38* untuk mengetahui kompatibilitas dari *smart card* dengan beberapa Sistem Operasi yang digunakan. Berikut hasil pengujian pada Tabel 4.3.

Tabel Pengujian 4.3 Pengujian Fungsional *Compability Smart Card*

Smart Card Reader	Sistem Operasi	Smartcard	
		ISO 7816	SLE4442
ACR 38	Windows XP	√	-
	Windows 7	√	-
	Windows 8.1	√	-
	Windows 10	√	-

Dari tabel fungsional *Smart Card* diatas menunjukkan bahwa *Smart Card* dengan tipe *ISO 7816* memiliki tingkat kompatibilitas yang sangat baik dengan 4 sistem operasi yang diuji coba sedangkan tipe *SLE4442* sama sekali tidak kompatibel baik dengan *reader ACS38*.

4. Pengujian *APDU Command* pada *Smart Card*.

Pengujian *APDU Command* pada *Smart Card* dilakukan dengan menguji beberapa fungsi diantaranya *Verifikasi, Select, Write, Read* seperti hasil pada tabel pengujian 4.4 Pengujian Fungsional *Smart Card* yaitu berhasil 100% seperti yang diharapkan.

Tabel Pengujian 4.4 Pengujian Fungsional *APDU Command* pada *Smart Card*

Fungsi	ISO 7816	
	Berhasil	Gagal
<i>Verify Smart Card</i>	√	-
<i>Select Smart Card</i>	√	-
<i>Write Smart Card</i>	√	-
<i>Read Smart Card</i>	√	-

5. Pengujian User Admin

Pengujian *User Admin* diterapkan kepada 2 pengguna yang menjadi *admin* warnet di “Warnet Pemuda” Perumahan Ardimulyo Singosari dengan rentan usia mulai dari 21 - 32 tahun. Pengujian dilakukan untuk mengetahui apakah aplikasi dapat dengan mudah diakses dan digunakan dengan 4 pilihan kriteria yakni mudah, cukup mudah, sulit dan sangat sulit. Untuk hasil pengujian dapat dilihat pada Tabel 4.5.

Tabel 4.5 Tabel Pengujian *User Admin*

Pengguna (<i>Admin</i>)	Kriteria			
	Sangat Mudah	Mudah	Sulit	Sangat Sulit
Pengguna 1	-	√	-	-
Pengguna 2	-	√	-	-

6. Pengujian User Member

Pengujian *User Member* diterapkan kepada 4 pengguna yang menjadi *user* warnet di “Warnet Pemuda” Perumahan Ardimulyo Singosari dengan rentan usia mulai dari 19 - 35 tahun. Pengujian dilakukan untuk mengetahui apakah aplikasi dapat dengan mudah diakses dan digunakan dengan 4 pilihan kriteria yakni mudah, cukup mudah, sulit dan sangat sulit. Untuk hasil pengujian dapat dilihat pada Tabel 4.6.

Tabel 4.6 Tabel Pengujian *User Member*

Pengguna (<i>Member</i>)	Kriteria			
	Sangat Mudah	Mudah	Sulit	Sangat Sulit
Pengguna 1	√	-	-	-
Pengguna 2	-	√	-	-
Pengguna 3	-	√	-	-
Pengguna 4	-	√	-	-

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan analisa dan pengujian, kesimpulan yang dapat diambil yaitu :

1. Dari hasil pengujian dapat diketahui aplikasi ini dapat berfungsi baik untuk deteksi *smart card menggunakan tipe ISO 7816* dan koneksi *Reader ACR38*.
2. Fungsi *CRUD Form Pelanggan* berfungsi 100%.
3. Dari tabel pengujian fungsional *Pengujian Smart Card Reader ACR38* tingkat kompatibilitas dengan 4 sistem operasi yang diuji, *Reader ACR38* menunjukkan tingkat kompatibilitas 100% .
4. Hasil pengujian fungsional sistem dengan akses sebagai operator berhasil diujikan dengan baik pada *Windows 10 64bit*.

5.2 Saran

Pada Rancang Bangun Aplikasi *Billing Smart Card* untuk *Member Warnet* ini, adapun pengembangan lebih lanjut yang disarankan :

1. Untuk kedepannya mungkin aplikasi ini bisa dibuat menggunakan perangkat *RFID (Radio Frequency Identification)* sebagai pengganti *Smart Card*.
2. Aplikasi *billing* dapat dilengkapi dengan fitur-fitur tambahan seperti *chatting*.

DAFTAR PUSTAKA

Ahmadjayadi, Cahyana and Soeprijanto, Bambang. 2007. Buku Panduan Sederhana Memulai Bisnis Warnet Legal. Departemen Komunikasi dan Informatika Direktorat Jenderal Aplikasi Telematika, Jakarta.

Pamungkas, Dedy, 2011, Aplikasi Smart Card Sebagai Kartu Pra Bayar, Semarang.

Swastika, I Putu Agus., Sisti Saibah Pua Luka., Yanno Dwi Ananda, 2010, Rancang Bangun Aplikasi Smart Card Interface; Vol No.1 2010, (Desember), 32-35.

<https://support.microsoft.com/id-id/kb/154544>, Deskripsi Tentang Teknologi ActiveX, di Akses Pada 25 Januari 2016.

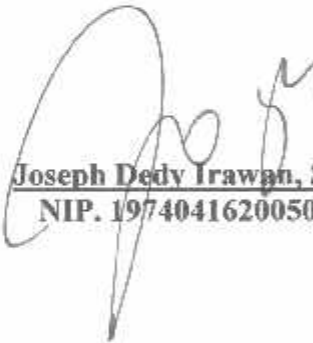
LAMPIRAN

BERITA ACARA UJIAN KOMPRESI
FAKULTAS TEKNOLOGI INDUSTRI

Nama : Rendy Syahputra
NIM : 12.18.129
Program Studi : Teknik Informatika S-1
Judul Skripsi : Rancang Bangun Aplikasi Billing Smart Card Untuk
Member Warnet

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S1)
pada:
Hari : Sabtu
Tanggal : 16 Januari 2016
Nilai : 81.89 (A)

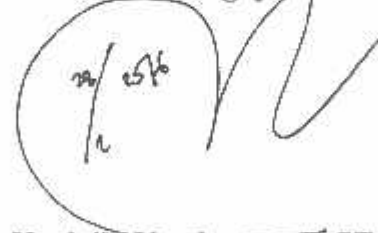
Panitia Ujian Skripsi
Ketua Majelis Penguji


Joseph Dedy Irawan, ST. MT.
NIP. 197404162005011002

Dosen Penguji I


Suryo Adi Wibowo, ST. MT
NIP. P. 1013000438

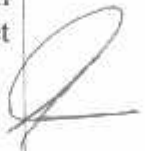

Dosen Penguji II


Nurlaily Vandyansyah, ST



FORMULIR PERBAIKAN SKRIPSI

Nama : Rendy Syahputra
NIM : 12.18.129
Program Studi : Teknik Informatika S-1
Judul Skripsi : Rancang Bangun Aplikasi Billing Smart Card Untuk
Member Warnet

Tanggal	Penguji	Uraian	Paraf
16 Januari 2016	I	<ul style="list-style-type: none">- Bab II Landasan Teori Tentang Komponen ActiveX dan Aplikasi Smart Card ToolSet PRO.- Pengujian Terhadap User.- Write Smart Card.	
16 Januari 2016	II	<ul style="list-style-type: none">- Latar Belakang Tentang Permasalahan.- Batasan Masalah.- Paparkan Sistem yang Sedang Berjalan Sebelumnya.- Sistem yang Akan Dibangun.	

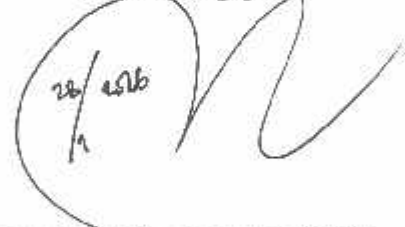
Anggota Penguji

Dosen Penguji I



Survo Adi Wibowo, ST. MT
NIP. P. 1013000438

Dosen Penguji II



Nurlaily Vendyansyah, ST

Mengetahui,

Dosen Pembimbing I



Sonny Prasetio, ST. MT
NIP. P. 1031000433

Dosen Pembimbing II



Febriana Santi W, S.Kom, M.Kom
NIP.P 1031000425



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 23 Oktober 2015

Nomor : ITN-593/I.INF/TA/2015

Lampiran : ---

Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu Sonny prasetio ST,MT
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,

Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : RENDY SYAHPUTRA
Nim : 1218129
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

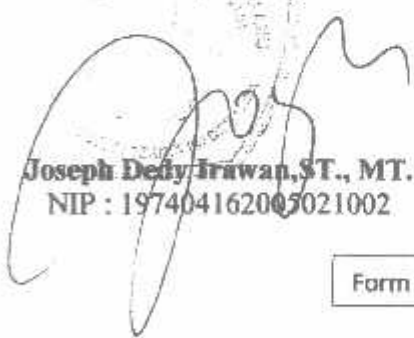
Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

23 Oktober 2015 S/D 23 Maret 2016

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.

Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua


Joseph Dedy Irawan, ST., MT.
NIP : 197404162005021002

Form S-4a

BUKTI BIMBINGAN SKRIPSI

Nama : Rendy Syahputra
NIM : 12.18.129
Masa Bimbingan : 23 Oktober 2015 s/d 23 Maret 2015
Judul Skripsi : Rancang Bangun Aplikasi Billing Smart Card Untuk
Member Warnet

No	Tanggal	Uraian	Paraf Pembimbing
1.	20/11/2015	- Revisi Bab I	
2.	21/11/2015	- Revisi Bab II	
3.	23/11/2015	- Revisi Bab III Penjelasan Flowchart	
4.	24/11/2015	- Acc Seminar Progress	
5.	18/11/2015	- Acc Seminar Hasil	
6.	08/01/2016	- Acc Bab I, II, III	
7.	09/01/2016	- Revisi Bab IV	
8.	11/01/2016	- Revisi Bab V Kesimpulan	
9.	12/01/2016	- Acc Bab IV & V	
10.	13/02/2016	- Acc Kompre.	

Malang, 25 Januari 2016

Dosen Pembimbing I



Sonny Prasetio, ST. MT
NIP. P. 1031000433



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

: BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 23 Oktober 2015

Nomor : ITN-593/L.INF/TA/2015
Lampiran : —
Perihal : Bimbingan Skripsi

Kepada : Yth. Bpk/Ibu Febriana Santi W, S.Kom, M.Kom
Dosen Pembina Program Studi Teknik Informatika S-1
Institut Teknologi Nasional
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :


Nama : RENDY SYAHPUTRA
Nim : 1218129
Prodi : Teknik Informatika S-1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan kami serahkan sepenuhnya kepada Saudara/i selama waktu 6 (enam) bulan, terhitung mulai tanggal :

23 Oktober 2015 S/D 23 Maret 2016

Sebagai satu syarat untuk menempuh Ujian Akhir Sarjana Teknik, Program Studi Teknik Informatika S-1.
Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S-1
Ketua,


Joseph Dedy Irawan, ST., MT.
NIP : 197404152005021002

Form S-4a

BUKTI BIMBINGAN SKRIPSI

Nama : Rendy Syahputra
NIM : 12.18.129
Masa Bimbingan : 23 Oktober 2015 s/d 23 Maret 2015
Judul Skripsi : Rancang Bangun Aplikasi Billing Smart Card Untuk
Member Warnet

No	Tanggal	Uraian	Paraf Pembimbing
1.	18/11/2015	- Revisi Program "Secondary Key"	hi
2.	21/11/2015	- Revisi Program "No Hp Pelanggan"	hi
3.	23/11/2015	- Acc Seminar Progress	hi
4.	09/12/2015	- Landasan Teori	hi
5.	16/12/2015	- Abstrak	hi
6.	08/01/2016	- Kutipan Daftar Pustaka	hi
7.	13/01/2016	- Revisi Laporan	hi
8.	14/01/2016	- Acc Kompre	hi
9.			
10.			

Malang, 25 Januari 2016

Dosen Pembimbing II



Febriana Santi W, S.Kom, M.Kom
NIP.P 1031000425

Source Code Form Pelanggan

```
procedure Tfrm_pelanggan.Timer1Timer(Sender: TObject);
begin
lb_tanggal.Caption:=DateToStr(date);
lb_jam.Caption:=TimeToStr(time);
end;

procedure Tfrm_pelanggan.bt_cariClick(Sender:
TObject);
var
    cari:string;
begin
    cari:=InputBox('Nama Yang Dicari ??','Nama :
','');
    // Prose Pencarian Data
    if tb_pelanggan.Locate('NAMA',cari,[]) then
with tb_pelanggan do
begin
ed_id.Text:=FieldValues['ID'];
ed_nama.Text:=FieldValues['NAMA'];
ed_alamat.Text:=FieldValues['ALAMAT'];
ed_saldo.Text:=FieldValues['SALDO'];
ed_no.Text:=FieldValues['NO_HP'];
end
else
begin
    Application.MessageBox('Tidak Ada Data
!!','Perhatian!');
end;
end;
```

```

procedure Tfrm_pelanggan.ed_idKeyPress(Sender:
TObject; var Key: Char);
begin
if not ( key in['0'..'9', #8, #13]) then
begin
key:= #0 ;
Application.MessageBox('Masukkan
Angka!', 'Peringatan!');
end
end;

procedure Tfrm_pelanggan.ed_noKeyPress(Sender:
TObject; var Key: Char);
begin
if not ( key in['0'..'9', #8, #13]) then
begin
key:= #0 ;
Application.MessageBox('Masukkan
Angka!', 'Peringatan!');
end
end;

procedure Tfrm_pelanggan.bt_tambahClick(Sender:
TObject);
begin
dbq_pelanggan.Enabled:=false;
ed_id.Enabled:=true;
ed_nama.Enabled:=true;
ed_alamat.Enabled:=true;
ed_saldo.Enabled:=true;
ed_no.Enabled:=true;
bt_tambah.Enabled:=false;

```



```

bt_simpan.Enabled:=true;
bt_hapus.Enabled:=false;
bt_ubah.Enabled:=false;
bt_reset.Enabled:=true;
bt_batal.Enabled:=true;
bt_cari.Enabled:=false;
end;

procedure Tfrm_pelanggan.bt_hapusClick(Sender:
TObject);
begin
if (Application.MessageBox('Data Akan
Terhapus', 'Konfirmasi?', MB_YESNO)=ID_Yes) then
    BEGIN
    tb_pelanggan.Delete;
    END;
    ed_id.Text:='';
    ed_nama.Text:='';
    ed_alamat.Text:='';
    ed_saldo.Text:='';
    ed_no.Text:='';
    ed_id.Enabled:=false;
    ed_nama.Enabled:=false;
    ed_alamat.Enabled:=false;
    ed_saldo.Enabled:=false;
    ed_no.Enabled:=false;
    bt_simpan.Enabled:=false;
    bt_hapus.Enabled:=false;
    bt_ubah.Enabled:=false;
    bt_reset.Enabled:=false;
    bt_tambah.Enabled:=true;
end;

```

```
procedure Tfrm_pelanggan.bt_ubahClick(Sender:
TObject);
begin
tb_pelanggan.Edit;
tb_pelanggan.FieldName('ID').AsString:=ed_id.Text;
tb_pelanggan.FieldName('NAMA').AsString:=ed_nama.Text;
tb_pelanggan.FieldName('ALAMAT').AsString:=ed_alamat
.Text;
tb_pelanggan.FieldName('SALDO').AsString:=ed_saldo.T
ext;
tb_pelanggan.FieldName('NO_HP').AsString:=ed_no.Text
;
tb_pelanggan.Post;

ed_id.Text:='';
ed_nama.Text:='';
ed_alamat.Text:='';
ed_saldo.Text:='';
ed_no.Text:='';
ed_id.Enabled:=false;
ed_nama.Enabled:=false;
ed_alamat.Enabled:=false;
ed_saldo.Enabled:=false;
ed_no.Enabled:=false;
bt_tambah.Enabled:=true;
bt_simpan.Enabled:=false;
bt_hapus.Enabled:=false;
bt_ubah.Enabled:=false;
bt_reset.Enabled:=false;
end;
```

```

procedure Tfrm_pelanggan.bt_simpanClick(Sender:
TObject);
begin
if (ed_id.GetTextLen=0) or (ed_nama.GetTextLen=0) or
(ed_alamat.GetTextLen=0) or (ed_saldo.GetTextLen=0) or
(ed_no.GetTextLen=0) then
begin
Application.MessageBox('Lengkapi Data Terlebih
Dahulu', 'Peringatan!');
end

else
begin
if (ed_id.GetTextLen>0) or (ed_nama.GetTextLen>0) or
(ed_alamat.GetTextLen>0) or (ed_saldo.GetTextLen>0) or
(ed_no.GetTextLen>0) then
begin
tb_pelanggan.Append;
tb_pelanggan.FieldName('ID').AsString:=ed_id.Text;

tb_pelanggan.FieldName('NAMA').AsString:=ed_nama.Text;

tb_pelanggan.FieldName('ALAMAT').AsString:=ed_alamat.
Text;

tb_pelanggan.FieldName('SALDO').AsString:=ed_saldo.T
ext;

tb_pelanggan.FieldName('NO_HP').AsString:=ed_no.Text
;
tb_pelanggan.Insert;

```

```

    bt_tambah.Enabled:=True;
    bt_simpan.Enabled:=False;
    ed_id.Enabled:=false;
    ed_nama.Enabled:=false;
    ed_alamat.Enabled:=false;
    ed_saldo.Enabled:=false;
    ed_no.Enabled:=false;
    bt_hapus.Enabled:=false;
    bt_ubah.Enabled:=false;
    bt_reset.Enabled:=false;
    dbg_pelanggan.Enabled:=true;
    tb_pelanggan.Next;
    ed_id.Text:='';
    ed_nama.Text:='';
    ed_alamat.Text:='';
    ed_saldo.Text:='';
    ed_no.Text:='';
    end
end;

procedure Tfrm_pelanggan.bt_resetClick(Sender:
TObject);
begin
    ed_id.Text:='';
    ed_nama.Text:='';
    ed_alamat.Text:='';
    ed_saldo.Text:='';
    ed_no.Text:='';
end;

procedure Tfrm_pelanggan.bt_batalClick(Sender:

```

```

TObject);
begin
ed_id.Enabled:=false;
ed_nama.Enabled:=false;
ed_alamat.Enabled:=false;
ed_saldo.Enabled:=false;
ed_no.Enabled:=false;
bt_simpan.Enabled:=false;
bt_hapus.Enabled:=false;
bt_ubah.Enabled:=false;
bt_reset.Enabled:=false;
bt_tambah.Enabled:=true;
bt_batal.Enabled:=false;
bt_cari.Enabled:=true;
dbg_pelanggan.Enabled:=true;
end;

procedure Tfrm_pelanggan.ed_idExit(Sender: TObject);
begin
if bt_simpan.Enabled then
begin
if tb_pelanggan.Locate('ID',ed_id.Text,[])=true then
begin
Application.MessageBox('ID Sudah
Terpakai!', 'Peringatan!');
ed_id.text:='';
ed_id.SetFocus;
end;
end;
end;

procedure Tfrm_pelanggan.bt_keluarClick(Sender:

```

```

TObject);
begin
frm_menu.show;
ed_id.Text:='';
ed_nama.Text:='';
ed_alamat.Text:='';
ed_saldo.Text:='';
ed_no.Text:='';
close;
end;

procedure Tfrm_pelanggan.FormCreate(Sender: TObject);
begin
ed_id.Enabled:=false;
ed_nama.Enabled:=false;
ed_alamat.Enabled:=false;
ed_saldo.Enabled:=false;
ed_no.Enabled:=false;
bt_simpan.Enabled:=false;
bt_hapus.Enabled:=false;
bt_ubah.Enabled:=false;
bt_reset.Enabled:=false;
bt_batal.Enabled:=false;
end;

procedure Tfrm_pelanggan.ed_noExit(Sender: TObject);
begin
if bt_simpan.Enabled then
begin
if tb_pelanggan.Locate('NO_HP',ed_no.Text, [])=true
then
begin

```

```

Application.MessageBox('Nomor Sudah
Ada!', 'Peringatan!');
ed_no.text:='';
ed_no.SetFocus;
end;
end;
end;
end.

```

Source Code Form Tambah Saldo

```

procedure Tfrm_saldo.bt_tambah_saldoClick(Sender:
TObject);
var TMBH:Integer;
begin
if ed_jumlah_saldo.GetTextLen=0 then
ShowMessage('Lengkapi data terlebih dahulu');
If (ed_jumlah_saldo.GetTextLen>0) then
begin
TMBH:=(StrToInt (lb_sisa.Caption)+StrToInt (ed_jumlah_sa
ldo.Text));
tb_pelanggan.Edit;
tb_pelanggan.FieldName ('SALDO').AsInteger:=TMBH;
tb_pelanggan.Post;

cd_jumlah_saldo.Text:='';
lb_sisa.Caption:='-';
end
end;

procedure Tfrm_saldo.bt_batalClick(Sender: TObject);

```

```

begin
frm menu.show;
lb_nama.Caption:='';
lb_sisa.Caption:='';
ed_jumlah_saldo.Text:='';
close;
end;

procedure Tfrm_saldo.dbg_jumlah_tambahDbClick(Sender:
TObject);
begin
with tb_pelanggan do
begin
lb_nama.Caption:=FieldValues['NAMA'];
lb_sisa.Caption:=FieldValues['SALDO'];
end;
ed_jumlah_saldo.Enabled:=true;
bt_tambah_saldo.Enabled:=true;
end;

procedure Tfrm_saldo.FormCreate(Sender: TObject);
begin
bt_tambah_saldo.Enabled:=false;
ed_jumlah_saldo.Enabled:=false;
end;

procedure Tfrm_saldo.Timer1Timer(Sender: TObject);
begin
lb_tanggal.Caption:=-DateToStr(date);
lb_jam.Caption:=TimeToStr(time);
end;
end.

```


Source Code Form Login

```
procedure Tfrm_login.bt_keluarClick(Sender: TObject);
begin
Application.Terminate;
end;

procedure Tfrm_login.bt_loginClick(Sender: TObject);
begin
  if
dat_mod.tb_log.Locate('USERNAME',ed_user.Text,[locasei
nsensitive]) then
    if dat_mod.tb_log.FieldByName('PASSWORD').AsString
= ed_password.Text then
      begin
        Application.MessageBox('Login
Berhasil', 'Informasi');
        hide;
        frm_menu.Show;
      end
    else
      begin
        Application.MessageBox('Password
Salah', 'Peringatan!');
      end
    else
      Application.MessageBox('Username Tidak
Ada', 'Peringatan!');
end;

procedure Tfrm_login.Timer1Timer(Sender: TObject);
```

```

begin
lb_jam.Caption:=TIMETOSTR(TIME);
lb_tgl.Caption:=DATEETOSTR(DATE);
end;

procedure Tfrm_login.ed_passwordKeyPress(Sender:
TObject; var Key: Char);
begin
if key = #13 then
begin
bt_login.Click;
end;
end;

procedure Tfrm_login.ed_userKeyPress(Sender: TObject;
var Key: Char);
begin
if key = #13 then
begin
bt_login.Click;
end;
end;

end.

```

Source Code Form Menu

```

procedure Tfrm_menu.bt_frm_billingClick(Sender:
TObject);
begin
frm_menu.Hide;

```

```
frm_billing.Show;
end;

procedure Tfrm_menu.bt_frm_saldoClick(Sender:
TObject);
begin
frm_saldo.Show;
frm_menu.Hide;
end;

procedure Tfrm_menu.Button5Click(Sender: TObject);
begin
Application.Terminate;
end;

procedure Tfrm_menu.bt_frm_pelangganClick(Sender:
TObject);
begin
frm_pelanggan.Show;
frm_menu.Hide;
end;

procedure Tfrm_menu.Timer1Timer(Sender: TObject);
begin
lb_tanggal.Caption:=DateToStr(date);
lb_jam.Caption:=TimeToStr(time);
end;

procedure Tfrm_menu.Exit1Click(Sender: TObject);
begin
Application.Terminate;
end;
```

```

procedure Tfrm_menu.Billing1Click(Sender: TObject);
begin
frm_menu.Hide;
frm_billing.Show;
end;

procedure Tfrm_menu.FormPelanggan1Click(Sender:
TObject);
begin
frm_menu.Hide;
frm_pelanggan.Show;
end;

procedure Tfrm_menu.FormSaldo1Click(Sender: TObject);
begin
frm_menu.Hide;
frm_saldo.Show;
end;

end.

```

Source Code Form Billing

```

var
frm_billing: Tfrm_billing;
total Menit: Integer;

waktuMulai: TTime;
waktuSelesai: Ttime;
TotalWaktu :Word;
TotalBiaya: Currency;

```

```

Jl,Ml,Dl,MDl:word;
BiayaAwal:Currency ;

Opened : Boolean;
Connected : Boolean;
Koneksi : WordBool;
implementation

uses Unit5, Unit4, Unit6;

{$R *.dfm}
function hexaToInt(s : string) : Int64;
begin

    if (s <> '') and (s[1] <> '$') then
        result := strToInt64('$' + s )
    else
        result := strToInt64(s);
end;

function Tfrm_billing.WaktuKeDetik(waktu: TTime):
Double;
var
    ms, ss, mm, hh: word;
    ret:double;
begin
    DecodeTime (waktu, hh, mm, ss, ms) ;
    ret:=(hh*3600) + (mm*60) + (ss) ;
    Result:=ret;
end;

function WaktuKeMenit(waktu: TTime): Double;

```

```

var
  ms, ss, mm, hh: word;
  ret:double;
begin
  DecodeTime(waktu, hh, mm, ss, ms);
  ret:=(hh*60)+mm;
  Result:=ret;
end;

function Tfrm_billing.DetikKeWaktu(const jumDetik:
Cardinal): Double;
var
  ms, ss, mm, hh: Cardinal;
begin
  hh := (jumDetik mod 86400) div 3600;
  mm := ((jumDetik mod 86400) mod 3600) div 60;
  ss := ((jumDetik mod 86400) mod 3600) mod 60;
  ms := 0;
  Result :=EncodeTime(hh, mm, ss, ms);
end;

function findCaption(lv: TListView; const S: string;
column: Integer):integer;
var
  i: Integer;
  found: Boolean;
  lvItem: TListItem;
  indexfound:integer;
begin
  indexfound:=-1;
  Assert (Assigned(lv));

```

```

Assert((lv.ViewStyle = vsReport) or (column = 0));
Assert(S <> '');
for i := 0 to lv.Items.Count - 1 do
begin
    lvItem := lv.Items[i];
    if column = 0 then
        found := AnsiCompareText(lvItem.Caption, S) = 0
    else if column > 0 then
        begin
            if lvItem.SubItems.Count >= Column then
                found :=
AnsiCompareText(lvItem.SubItems[column - 1], S) = 0
            else
                found := False;
        end
    else
        found := False;
    if found then
        begin
            lv.Selected := lvItem;
            indexfound :=lv.Selected.Index;
        end;
    end;
    Result:=indexfound;
end;

procedure Tfrm_billing.bt_keluarClick(Sender:
TObject);
begin
    frm_menu.show;
close;
end;

```

```

procedure Tfrm_billing.Timer1Timer(Sender: TObject);
var
  i:integer;
  waktul:TDateTime;
begin
lb_jam.Caption:=TIMETOSTR(TIME);
lb_tgl.Caption:=DATETOSTR(DATE);

try
  for i:=0 to lv_client.Items.Count-1 do
  begin
    if lv_client.Items[i].SubItems.Strings[5]='Online'
  then
    begin
waktuMulai:=StrToTime(lv_client.Items[i].SubItems[1]);
      waktuSelesai:=now;
      DecodeTime(waktuSelesai-
waktuMulai, J1, M1, D1, MD1);
      TotalWaktu:=(J1*60)+M1+1;
      TotalBiaya :=((TotalWaktu div
1)*StrToCurr(ed_biaya.Text));

waktul:=StrToTime(lv_client.Items[i].SubItems[1])-Now;

lv_client.Items[i].SubItems.Strings[2]:=TimeToStr(now)
;

lv_client.Items[i].SubItems.Strings[3]:=TimeToStr(wakt
ul);

lv_client.Items[i].SubItems.Strings[4]:=CurrToStr(Tota
lBiaya);

```



```

        end
    end;
except
    end;
end;

procedure Tfrm_billing.FormCreate(Sender: TObject);
begin
    ServerSocket1.Active:=true;
    bt_diconnect.Enabled:=false;
end;

procedure
Tfrm_billing.ServerSocket1ClientError(Sender: TObject;
    Socket: TCustomWinSocket; ErrorEvent: TErrorEvent;
    var ErrorCode: Integer);
begin
    ErrorCode:=0;
end;

procedure Tfrm_billing.ServerSocket1ClientRead(Sender:
TObject;
    Socket: TCustomWinSocket);
var
    row:TListItem;
    command: TStringList;
    i,found:integer;
    waktu:String;
begin
    command:=TStringList.Create;
    command.Delimiter:='@';
    command.DelimitedText:=Socket.ReceiveText;

```

```

if command[0] = 'CON' then
begin
    waktu:=TimeToStr(Now);
    found:=findCaption(lv_client,command[1],0);

lv_client.Items[found].SubItems.Strings[0]:=command[2]
;
    lv_client.Items[found].SubItems.Strings[1]:=waktu;

lv_client.Items[found].SubItems.Strings[2]:=TimeToStr(
time);

lv_client.Items[found].SubItems.Strings[3]:='00:00:00'
;
    lv_client.Items[found].SubItems.Strings[4]:='Rp.0';

lv_client.Items[found].SubItems.Strings[5]:=command[3]
;

Socket.SendText(command[0]+'@start@'+TimeToStr(Now)+'@
'+ed_biaya.Text);
    end

    else if command[0]= 'DSC' then
begin
    for i:-0 to ServerSocket1.Socket.ActiveConnections-
1 do

ServerSocket1.Socket.Connections[i].SendText(lv_client
.Selected.Caption+'@stop');

lv_client.Selected.SubItems.Strings[5]:='Offline';

```

```
end

else if command[0]= 'MDSC' then
begin
    frm_pelanggan.lb_pelanggan.Edit;

frm_pelanggan.tb_pelanggan.FieldName('SALDO').AsString:=command[2];
    frm_pelanggan.tb_pelanggan.Post;

lv_client.Selected.SubItems.Strings[5]:='Offline';
    end
else
begin
    row:=lv_client.Items.Add;
    row.Caption:=command[1];
    row.SubItems.Add('Kosong');
    row.SubItems.Add('00:00:00');
    row.SubItems.Add('00:00:00');
    row.SubItems.Add('00:00:00');
    row.SubItems.Add('Rp.0');
    row.SubItems.Add(command[2]);
end;
end;

procedure Tfrm_billing.bt_diconnectClick(Sender:
TObject);
begin
SmartCardToolSetSERVER_ActiveX1.DisconnectSCARD;
bt_diconnect.Enabled:=false;
end;
```

```

procedure Tfrm_billing.bt_setClick(Sender: TObject);

begin
if bt_set.Caption='Set' then
begin
ed_biaya.Enabled:=true;
bt_set.Caption:='Done';
end
else
begin
Next;

if bt_set.Caption='Done' then
begin
ed_biaya.Enabled:=false;
bt_set.Caption:='Set';
end;
end;
end;

procedure Tfrm_billing.Stop1Click(Sender: TObject);
var
i:integer;
begin
for i:=0 to ServerSocket1.Socket.ActiveConnections-1
do

ServerSocket1.Socket.Connections[i].SendText(lv_client
.Selected.Caption+'@stop');

lv_client.Selected.SubItems.Strings[5]:='Offline';
end;

```
