



TUGAS AKHIR - KI091391

**RANCANG BANGUN SISTEM KEAMANAN
KELUAR MASUK PARKIR DENGAN KARTU
CERDAS MIFARE DAN TEKNOLOGI NEAR
FIELD COMMUNICATION (NFC) STUDI KASUS
PARKIR JURUSAN TEKNIK INFORMATIKA**

FAHMI HIDAYATULLAH
NRP 5110 100 199

Dosen Pembimbing I
Dwi Sunaryono, S.Kom., M.Kom.

Dosen Pembimbing II
Ridho Rahman Hariadi, S.Kom., M.Sc.

JURUSAN TEKNIK INFORMATIKA
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2014



FINAL PROJECT - KI091391

**SOFTWARE DESIGN AND IMPLEMENTATION
PARKING AREA SECURITY SYSTEM WITH
MIFARE SMART CARD AND NEAR FIELD
COMMUNICATION TECHNOLOGY USE CASE
PARKING AREA INFORMATICS DEPARTMENT**

**FAHMI HIDAYATULLAH
NRP 5110 100 199**

Advisor
Dwi Sunaryono, S.Kom., M.Kom.
Ridho Rahman Hariadi, S.Kom., M.Sc.

**INFORMATICS ENGINEERING DEPARTEMENT
Information Technology Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2014**

**Rancang Bangun Sistem Keamanan Keluar Masuk Parkir
dengan Kartu Cerdas *Mifare* dan Teknologi *Near Field
Communication* (NFC) Studi Kasus Parkir Jurusan
Teknik Informatika.**

Nama Mahasiswa : Fahmi Hidayatullah
NRP : 5110 100 199
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing I : Dwi Sunaryono, S.Kom., M.Kom.
Dosen Pembimbing II : Ridho Rahman Hariadi, S.Kom., M.Sc.

ABSTRAK

Institut Teknologi Sepuluh Nopember (ITS) sebagai salah satu perguruan tinggi yang unggul di bidang teknologi terus berbenah menyediakan fasilitas terbaiknya. Namun dalam pelaksanaannya, kepuasan terhadap fasilitas itu bisa dibilang kurang, terutama dimasalah kepuasan penjagaan keamanan parkir yang menyatakan 87% mahasiswa ITS tidak puas hingga cukup puas.

Near Field Communication adalah pengembangan dari Radio Frequency Identification (RFID). NFC memberikan kemudahan dalam bertransaksi dengan mudah. Membuat NFC dapat diimplementasikan di banyak tempat. Salah satunya adalah pemanfaatan NFC dengan kartu cerdas mifare. Dimana Kartu Tanda Mahasiswa ITS adalah kartu cerdas jenis mifare dan masih belum termanfaatkan.

Sistem Keamanan Kampus adalah aplikasi berbasis desktop dengan teknologi NFC dan kartu cerdas Mifare. Tugas Akhir ini memadukan teknologi kartu cerdas mifare yang merupakan salah satu bagian dari contactless card yang memiliki kemampuan untuk berkomunikasi dengan menggunakan gelombang radio dengan alat pembaca NFC tanpa kontak langsung secara fisik. Sistem Keamanan Kampus dapat melakukan pencatatan setiap orang yang masuk ke dalam parkir kendaraan roda dua dan melakukan pengecekan ketika akan keluar dari tempat parkir. Sehingga dapat diketahui siapa saja orang yang telah masuk dan keluar dari tempat parkir dengan kendaraan tertentu.

Pengujian dilakukan langsung selama beberapa hari di parkir jurusan Teknik Informatika terhadap beberapa orang yang masuk tempat parkir, dilakukan dengan mendekatkan kartu cerdas baik itu Kartu Tanda Mahasiswa dan/atau Kartu Tanda Penduduk yang dimiliki pengguna ke alat pembaca NFC. Hasil uji coba aplikasi memberikan performa yang baik. Untuk melakukan pencatatan ketika pengguna masuk dibutuhkan waktu rata-rata 132,6 milidetik dan untuk melakukan verifikasi ketika pengguna keluar dibutuhkan waktu rata-rata 1.729 milidetik.

Kata kunci: *Kartu Cerdas, Keamanan Parkir, Mifare, NFC*

**Software Design and Implementation Parking Area
Security System with Mifare Smart Card And Near Field
Communication Technology Use Case Parking Area
Informatics Department**

Student Name : Fahmi Hidayatullah
NRP : 5110 100 199
Major : Informatics Engineering FTIf-ITS
Advisor I : Dwi Sunaryono, S.Kom., M.Kom.
Advisor II : Ridho Rahman Hariadi, S.Kom, M.sc.

ABSTRACT

Institut Teknologi Sepuluh Nopember (ITS) as one of the leading university in the field of technology continues to improve his best facilities. But in practice, the rate of satisfaction in his facility is low. Especially in the satisfaction in parking guard security. It's shown that 87% of Student of ITS are not satisfied until quite satisfied in parking guard security.

Near Field Communication (NFC) is the development of Radio Frequency Identification (RFID). NFC to easily transact with ease. Make NFC can be implemented in many places. One is the use of NFC with smartcard mifare. Where the Student Identity Card of ITS is a smart card with Mifare type and still untapped.

Campus Security System is a desktop based application with NFC Technology and Mifare Smart Card. This Final Project blends Mifare Smart Card technology which is one part of a contactless card that has the ability to communicate using radio waves with NFC Reader without direct physical contact. Campus Security System can keeps records of each person who entered into parking and check when it will be out of the parking lot. So it can be seen that anyone who has been in and out of the parking lot with a certain vehicles.

The experimental performed directly for a few days at the parking area of Informatics Department for a few people who entered the parking area, done with either a smartcard closer of Student Identity Card and/or National Identity Card owned by the use to NFC Reader.

The experimental result of this application shows a good performance. To record when the user enters the average time it takes 132.6 milliseconds and to verify when the user exits the average time it takes 1,729 milliseconds.

Keyword: *Smart Card, Security of Parking lot, Mifare, NFC.*

LEMBAR PENGESAHAN

**Rancang Bangun Sistem Keamanan Keluar Masuk Parkir
dengan Kartu Cerdas *Mifare* dan Teknologi *Near Field
Communication (NFC)* Studi Kasus Parkir Jurusan
Teknik Informatika.**

TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Rekayasa Perangkat Lunak
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

FAHMI HIDAYATULLAH

NRP : 5110 100 199

Disetujui oleh Dosen Pembimbing Tugas akhir

Dwi Sunaryono, S.Kom., M.Kom.
NIP: 197205281997021001

(pembimbing 1)

Ridho Rahman Hariadi, S.Kom., M.Sc.
NIP: 051100123

(pembimbing 2)



**SURABAYA
JULI 2014**

DAFTAR GAMBAR

Gambar 1. 1 Diagram grafik hasil survei tingkat kepuasan mahasiswa ITS	2
Gambar 2. 1 Perangkat ACR 122 U, kartu cerdas dan SDK	13
Gambar 2. 2 Diagram Alir Komunikasi ACR122U	14
Gambar 2. 3 Arsitektur Kartu Cerdas Mifare	17
Gambar 2. 4 Kartu Cerdas Mifare	17
Gambar 2. 5 Kartu Tanda Mahasiswa	18
Gambar 2. 6 Kartu Tanda Penduduk	19
Gambar 2. 7 Format Perintah APDU sesuai ISO 7816	20
Gambar 2. 8 Format Respon APDU	21
Gambar 2. 9 Penjelasan Respon APDU SW1 dan SW2	21
Gambar 3. 1 Diagram Kasus Penggunaan	29
Gambar 3. 2 Diagram Aktivitas Kasus Penggunaan UC-001	32
Gambar 3. 3 Diagram Sekuensial Kasus Penggunaan UC-001	33
Gambar 3. 4 Diagram Aktivitas Kasus Penggunaan UC-002	35
Gambar 3. 5 Diagram Sekuensial Kasus Penggunaan UC-002	36
Gambar 3. 6 Diagram Aktivitas Kasus Penggunaan UC-003	39
Gambar 3. 7 Diagram Sekuensial Kasus Penggunaan UC-003	40
Gambar 3. 8 Diagram Aktivitas Kasus Penggunaan UC-004	43
Gambar 3. 9 Diagram Sekuensial Kasus Penggunaan UC-004	44
Gambar 3. 10 Diagram Aktivitas Kasus Penggunaan UC-005	47
Gambar 3. 11 Diagram Sekuensial Kasus Penggunaan UC-005	48
Gambar 3. 12 Diagram Model Konsep Basis Data (CDM) Aplikasi Sistem Keamanan Kampus	49
Gambar 3. 13 Diagram Model Fisik Basis Data (PDM) Aplikasi Sistem Keamanan Kampus	50
Gambar 3. 14 Arsitektur Sistem	52
Gambar 3.15 Rancangan Halaman Login	54
Gambar 3. 16 Rancangan Halaman Admin Mengelola Data Pengguna	56

Gambar 3. 17 Rancangan Halaman Admin Mengelola Rekapitulasi Keluar Masuk Kendaraan	58
Gambar 3. 18 Rancangan Halaman Utama	60
Gambar 3. 19 Diagram Alir Proses Login	62
Gambar 3. 20 Diagram Alir Proses Pendeteksian kartu Cerdas ..	63
Gambar 3. 21 Diagram Alir Proses Pengambilan Gambar.....	64
Gambar 3. 22 Diagram Alir Proses Pencatatan Kendaraan Masuk Parkir	65
Gambar 3. 23 Diagram Alir Proses verifikasi kendaraan yang akan Keluar.....	66
Gambar 3. 24 Diagram Alir Prose Pencarian Data Pengguna	67
Gambar 3. 25 Diagram Alir Proses Menambahkan Data Pengguna	68
Gambar 3. 26 Proses Pengubahan Data Pengguna	69
Gambar 3. 27 Diagram Alir Proses Menghapus Data Pengguna.	70
Gambar 3. 28 Diagram Alir Mengubah Data Transaksi.....	71
Gambar 3. 29 Diagram Alir Proses Menghapus Data Transaksi.	72
Gambar 4. 1 Implementasi Halaman Login.....	74
Gambar 4. 2 Implementasi Halaman Admin Mengelola Data Pengguna	75
Gambar 4. 3 Implementasi Halaman Admin Mengelola Rekapitulasi Keluar Masuk Kendaraan	76
Gambar 4. 4 Implementasi Halaman Utama.....	77
Gambar 5. 1 Pengujian Menambahkan Data Pengguna	105
Gambar 5. 2 Pengujian Mengubah Data Pengguna	107
Gambar 5. 3 Pengujian Mengubah Data Transaksi	109
Gambar 5. 4 Hasil Penyimpanan Pada Saat Transaksi	112
Gambar 5. 5 Beberapa Transaksi Terakhir yang Tersimpan	112
Gambar 5. 6 Pengujian Melihat data transaksi keluar masuk kendaraan.....	115
Gambar 5. 7 Hasil Ujicoba Login.....	117

Gambar 5. 8 Jenis Kartu Cerdas yang Digunakan saat Pengujian	118
Gambar 5. 9 Grafik Hasil Pengujian Performa (dalam milidetik)	124

DAFTAR KODE SUMBER

Kode Sumber 4. 1 Implementasi Struktur Tabel Akses	79
Kode Sumber 4. 2 Implementasi Struktur Tabel Pengguna	79
Kode Sumber 4. 3 Implementasi Struktur Tabel Transaksi.....	80
Kode Sumber 4. 4 Implementasi <i>Query</i> Login.....	81
Kode Sumber 4. 5 Implementasi <i>Query</i> Mencari Data Pengguna	82
Kode Sumber 4. 6 Implementasi <i>Query</i> Menambahkan Data Pengguna	83
Kode Sumber 4. 7 Implementasi <i>Query</i> Mengubah Data Pengguna	83
Kode Sumber 4. 8 Implementasi <i>Query</i> Menghapus Data Pengguna	84
Kode Sumber 4. 9 Implementasi <i>Query</i> Mencari Data Transaksi	85
Kode Sumber 4. 10 Implementasi <i>Query</i> Mengubah Data Transaksi	86
Kode Sumber 4. 11 Implementasi <i>Query</i> Pencatatan Kendaraan Masuk Parkir	87
Kode Sumber 4. 12 Implementasi <i>Query</i> Menambahkan Data Pengguna	88
Kode Sumber 4. 13 Implementasi Proses Login	89
Kode Sumber 4. 14 Implementasi Proses Pendeteksian Kartu Cerdas.....	94
Kode Sumber 4. 15 Implementasi Proses Pengambilan Gambar dengan Webcam	94
Kode Sumber 4. 16 Implementasi Proses Pencatatan Kendaraan Masuk Parkir	95
Kode Sumber 4. 17 Implementasi Verifikasi kendaraan yang Keluar Parkir	96

Kode Sumber 4. 18 Implementasi Proses Mencari Data Pengguna	97
Kode Sumber 4. 19 Implementasi Proses Menambah Data Pengguna	98
Kode Sumber 4. 20 Implementasi Proses Mengubah Data Pengguna	99
Kode Sumber 4. 21 Implementasi Proses Menghapus Data Pengguna	99
Kode Sumber 4. 22 Implementasi Proses Mencari Data Transaksi	101
Kode Sumber 4. 23 Implementasi Proses Mengubah Data Transaksi.....	102
Kode Sumber 4. 24 Implementasi Proses Menghapus Data Transaksi.....	102
Kode Sumber A. 1 Implementasi API ACR122U	143

KATA PENGANTAR

Alhamdulillah, Puji syukur kepada Allah Yang Maha Esa atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan Tugas Akhir ini. Shalawat serta salam tak lupa diucapkan kepada Baginda Agung Rasulullah SAW, Keluarga, Sahabat dan para pengikutnya hingga akhir zaman.

Penulis ingin menyampaikan penghormatan dan ucapan terima kasih yang sebesar-besarnya kepada pihak-pihak yang telah membantu dalam menyelesaikan Tugas Akhir ini, terutama kepada.

1. Bapak Sudjarni, Ibu Sukariyati, Mas Fathul, Dek Fitri, Dek Faridl dan keluarga penulis lainnya yang tiada henti-hentinya mencurahkan kasih sayang, perhatian, dan doa kepada penulis selama ini demi kelancaran penulis untuk terus berjuang dan pantang menyerah.
2. Bapak Dwi Sunaryono dan Bapak Ridho Rahman selaku Dosen Pembimbing yang telah memberikan bimbingan, motivasi, dan meluangkan waktu untuk membantu pengerjaan Tugas Akhir ini.
3. Bapak Wahyu Suadi dan Bapak Supeno Djanali sebagai Dosen Wali penulis dan Bapak dan Ibu Dosen Teknik Informatika ITS yang telah membina dan memberikan ilmu dan saran kepada penulis selama menempuh studi di Teknik Informatika ITS.
4. Seluruh staf dan karyawan Teknik Informatika, FTIF dan ITS yang banyak memberikan kelancaran administrasi kepada penulis.
5. Sahabat dekat yang selalu memberikan dukungan dan inspirasi untuk terus berkarya: Hani, Fadli, Muamar, Fadjar, Amin, Riduwan, Awal, Dhila, Ika, Anita, Luluk dan Eka.

6. Kabinet BEM ITS Muda Bersahabat Mukhlis, Umam, Sidratu, Hanum, Brahm, Solving, Avis, para menteri dan ‘sekmen’ lainnya yang membuat tahun terakhir penulis dikampus menjadi lebih ‘seru’.
7. Armada Dagri Kety, Ian, Ashar, Fajar, Myta, Marsha dan semua staff tercinta yang membuat keseharian penulis lebih hidup dan memahami apa itu arti sebuah keluarga.
8. Punakawan Dagri dan Kabinet BEM ITS Mahakarya Mas Widy, Mbak Ken, Mas Ary, Mas Sony, Mas Yoga, Rifqi, dan lainnya yang telah membimbing penulis dan mengajarkan apa arti sebuah karya dan kebermanfaatannya.
9. Rekan-rekan Administrator Laboratorium Pengembangan Game Mas Rohmad, Mas Misbach, Mbak Yola, Maria, Monika, Yasin, Andrie dan Miranti.
10. Teman-teman Eufouria, Pemandu FTIF dan Keluarga Biru yang membuat pengalaman penulis lebih berwarna.
11. KPU ITS 2012 Mas Rivan, Mas Vonda, Mas Restu, Azzah, Yosia, PPU dan Panwaslu atas cerita indahny di PEMILU.
12. Keluar besar angkatan TC 2010 dan C1A tercinta yang sudah membantu, mendukung dan menemani keseharian penulis dalam kampus perjuangan.
13. Serta pihak-pihak lain yang namanya tidak dapat penulis sebutkan satu–persatu, semuanya terimakasih banyak.

Bagaimanapun juga penulis telah berusaha sebaik-baiknya dalam menyusun Tugas Akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan Saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juli 2014

Fahmi Hidayatullah

DAFTAR ISI

LEMBAR PENGESAHAN.....	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI.....	xv
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxiii
DAFTAR KODE SUMBER	xxv
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Permasalahan.....	4
1.3 Batasan Permasalahan	4
1.4 Tujuan.....	5
1.5 Metodologi	5
1.6 Sistematika	8
BAB II TINJAUAN PUSTAKA.....	11
2.1 Profil ITS	11
2.2 Identifikasi dan Verifikasi	12
2.3 Rancang Bangun Perangkat Lunak.....	12
2.4 <i>Near Field Communication</i> (NFC).....	12
2.5 <i>Smartcard Reader/Writer ACR122U NFC</i>	13

2.6	Microsoft Visual Studio.....	15
2.7	Microsoft Visual C# (C Sharp).....	16
2.8	Mifare	16
2.9	MySQL	20
2.10	<i>Application Protocol Data Unit (APDU)</i>	20
2.11	EasyWebCam Library	21
BAB III ANALISIS DAN PERANCANGAN		23
3.1.	Analisis	23
2.1.1	Analisis Permasalahan.....	23
3.1.2	Deskripsi Umum Sistem.....	25
3.1.3	Spesifikasi Kebutuhan Perangkat Lunak	26
3.1.4	Identifikasi Pengguna	28
3.2	Perancangan.....	28
3.2.1	Skenario Kasus Penggunaan.....	28
3.2.2	Perancangan Basis Data	49
3.2.3	Perancangan Arsitektur	52
3.2.4	Perancangan Antarmuka Perangkat Lunak.....	54
3.2.5	Perancangan Proses Perangkat Lunak	61
BAB IV IMPLEMENTASI		73
4.1	Lingkungan Implementasi	73
4.1.1	Lingkungan Implementasi Perangkat Keras.....	73
4.1.2	Lingkungan Implementasi Perangkat Lunak.....	73
4.2	Lingkungan Implementasi Antarmuka	74
4.2.1	Implementasi Antarmuka Halaman Login	74

4.2.2 Implementasi Antarmuka Halaman Admin Mengelola Data Pengguna.....	75
4.2.3 Implementasi Antarmuka Halaman Admin Mengelola Rekapitulasi Keluar Masuk Kendaraan	76
4.2.4 Implementasi Antarmuka Halaman Utama	77
4.3. Implementasi Basis Data	78
4.3.1 Implementasi Struktur Basis Data.....	78
4.3.2 Implementasi <i>Query</i>	80
4.4. Implementasi Proses Aplikasi	88
4.4.1 Implementasi Proses Login	88
4.4.2 Implementasi Proses Pendeteksian Kartu Cerdas.....	89
4.4.3 Implementasi Proses Pengambilan Gambar dengan Webcam.....	94
4.4.4 Implementasi Proses Pencatatan Kendaraan Masuk Parkir	94
4.4.5 Implementasi Proses Verifikasi Kendaraan yang Keluar Parkir	95
4.4.6 Implementasi Proses Mencari Data Pengguna	96
4.4.7 Implementasi Proses Menambah Data Pengguna.....	97
4.4.8 Implementasi Proses Mengubah Data Pengguna	98
4.4.9 Implementasi Proses Menghapus Data Pengguna.....	99
4.4.10 Implementasi Proses Mencari Data Transaksi Keluar Masuk Kendaraan.....	100
4.4.11 Implementasi Proses Mengubah Data Transaksi Keluar Masuk Kendaraan	101

4.4.12 Implementasi Proses Menghapus Data Transaksi Keluar Masuk Kendaraan	102
BAB V PENGUJIAN DAN EVALUASI	103
5.1. Lingkungan Pengujian	103
5.2. Skenario Pengujian	103
5.2.1. Pengujian Fungsionalitas	104
5.2.2. Pengujian Kegunaan	117
5.2.3 Pengujian Performa	123
5.3. Evaluasi Pengujian.....	126
5.3.1 Evaluasi Pengujian Fungsionalitas	126
5.3.2 Evaluasi Pengujian Kegunaan	127
BAB VI PENUTUP.....	129
6.1. Kesimpulan.....	129
6.2. Saran	130
DAFTAR PUSTAKA.....	131
LAMPIRAN A – KODE SUMBER.....	133
LAMPIRAN B – HASIL UJI COBA	145
BIODATA PENULIS.....	151

DAFTAR TABEL

Tabel 3. 1 Daftar Kebutuhan Fungsional	26
Tabel 3. 2 Daftar Kebutuhan Non Fungsional.....	27
Tabel 3. 3 Daftar Kasus Penggunaan	29
Tabel 3. 4 Rincian Alur Kasus Penggunaan UC-001	31
Tabel 3. 5 Rincian Alur Kasus Penggunaan UC-002	34
Tabel 3. 6 Rincian Alur Kasus Penggunaan UC-003	37
Tabel 3. 7 Rincian Alur Kasus Penggunaan UC-004	41
Tabel 3. 8 Rincian Alur Kasus Penggunaan UC-005	45
Tabel 3. 9 Atribut Tabel Akses	50
Tabel 3. 10 Atribut Tabel Pengguna (<i>User</i>).....	51
Tabel 3. 11 Atribut Tabel Transaksi.....	51
Tabel 5. 1 Skenario Pengujian Menambah Data Pengguna	104
Tabel 5. 2 Skenario Pengujian Mengubah Data Pengguna	106
Tabel 5. 3 Skenario Pengujian Menghapus Data Pengguna.....	107
Tabel 5. 4 Skenario Mengubah Data Transaksi.....	108
Tabel 5. 5 Skenario Menghapus Data Transaksi	110
Tabel 5. 6 Skenario Pengujian Mencatat Kendaraan Masuk Parkir	111
Tabel 5. 7 Skenario Pengujian Melakukan Verifikasi Kendaraan Keluar Parkir	113
Tabel 5. 8 Skenario Pengujian Melihat Transaksi Keluar Masuk Kendaraan.....	114
Tabel 5. 9 Skenario Pengujian Login Aplikasi.....	115
Tabel 5. 10 Pelaksanaan Pengujian	118
Tabel 5. 11 Rekapitulasi Pengujian Aplikasi	118
Tabel 5. 12 Tabel Basis Data Pengujian.....	123
Tabel 5. 13 Hasil Pengujian Perfoma (dalam detik).....	125
Tabel B. 1 Hasil Pengujian.....	145

BAB I

PENDAHULUAN

Bagian ini akan dijelaskan hal-hal yang menjadi latar belakang, permasalahan yang dihadapi, batasan masalah, tujuan dan manfaat, metodologi, dan sistematika penulisan yang digunakan dalam pembuatan Tugas Akhir.

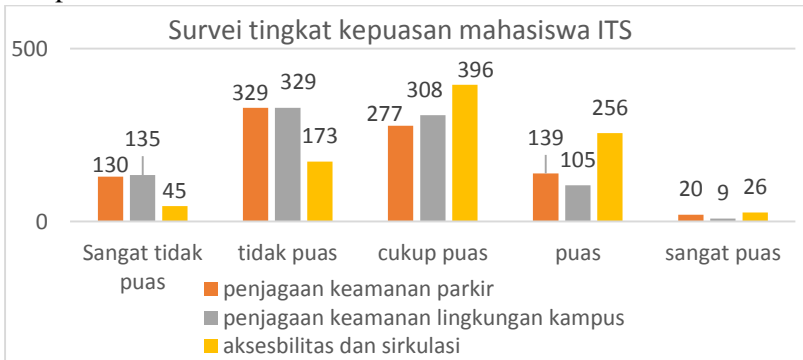
1.1 Latar Belakang

Institut Teknologi Sepuluh Nopember Surabaya (ITS) merupakan kampus teknik yang ternama di Indonesia, khususnya di wilayah timur Indonesia. Aktivitas kampus yang cukup padat membuat sirkulasi keluar masuk area kampus sangat tinggi, baik itu pejalan kaki maupun pengguna kendaraan bermotor. Selain itu, kondisi akses pintu keluar masuk yang ada ITS cukup banyak membuat akses ke dalam kampus sangat mudah dan gampang juga dimasuki oleh orang-orang yang tidak berkepentingan dengan segala aktivitas-aktivitas yang terkait dengan kegiatan kampus sekalipun. Selain itu, wilayah kampus ITS ini juga menjadi satu dengan kawasan perumahan dosen.

Fakta-fakta ini membuktikan bahwa keamanan di kampus ITS bisa dikatakan sangat rawan. Berdasarkan informasi yang didapat dari Satuan Keamanan Kampus (SKK-ITS) dalam tahun 2014, kasus kehilangan yang terjadi di wilayah kampus ITS sudah sangat banyak. Contohnya kasus kehilangan dua motor sekaligus di parkir pusat KPA-ITS pada awal bulan Februari 2014. Ketika korban dan di bantu oleh Badan Eksekutif Mahasiswa ITS meminta pertanggung jawabannya dari pihak ITS, Prof. Drs. Nur Iriawan, Mkom., Ph.D. selaku wakil rektor 3 (tiga) bidang sumber daya manusia dan bidang organisasi menjelaskan bahwa tidak ada bukti yang menguatkan bahwa korban pada saat itu benar-benar parkir di tempat tersebut.

Berikutnya, kasus kehilangan dua motor kembali terjadi secara bersamaan di jurusan Arsitektur, yang pada saat itu bertepatan dengan hari libur ITS (Sabtu dan Minggu). Kedua kendaraan tersebut diparkir tidak pada tempat yang telah disediakan. Selain itu, di kampus ITS ini tidak hanya kehilangan kendaraan bermotor saja yang terjadi, tetapi juga terjadi cukup banyak kasus kehilangan tas, laptop, telepon genggam dan yang lainnya. Dari informasi yang disampaikan oleh salah satu polisi dari polsek Sukolilo bahwa kampus ITS ini adalah ladangnya para pelaku kriminal khususnya di wilayah polsek Sukolilo itu sendiri.

Banyak pihak yang mengeluhkan keadaan ini, khususnya dari kalangan mahasiswa ITS sendiri. Karena berdasarkan survei yang dilakukan oleh Kementerian Dalam Negeri Badan Eksekutif Mahasiswa (BEM) ITS melalui kuesioner yang dibagikan secara acak kepada 890 mahasiswa ITS, didapatkan bahwa 82% mahasiswa ITS menyatakan sangat tidak puas hingga cukup puas terhadap tingkat kepuasan penjagaan keamanan parkir (kendaraan bermotor), 87% menyatakan sangat tidak puas hingga cukup puas terhadap tingkat kepuasan penjagaan keamanan lingkungan kampus.



Gambar 1. 1 Diagram grafik hasil survei tingkat kepuasan mahasiswa ITS

Sebelumnya Muhammad sudah melakukan penelitian dan implementasi sebuah perangkat lunak yang digunakan untuk sistem pengecekan keluar masuk ITS dan manajemen kendaraan dengan menggunakan QR Code yang dicetak di dalam sebuah sticker yang ditempel di kendaraan bermotor dan terhubung dengan internet dan layanan basis data yang terpusat [1]. Kekurangan dari penggunaan QR Code ini adalah sifatnya yang statis dan permanen, tidak bisa diganti-ganti, dan untuk pembuatannya dibutuhkan waktu yang lama karena QR Code dibuat menggunakan media sticker.

Teknologi Near Field Communication (NFC) adalah sebuah teknologi baru yang merupakan pengembangan dari teknologi RFID yang sudah banyak dikenal dan digunakan dalam aktivitas sehari-hari. NFC merupakan teknologi yang digunakan untuk transfer data dari satu perangkat ke perangkat lainnya. Di Indonesia, penggunaan NFC dalam bidang transaksi elektronik masih tergolong suatu teknologi yang baru dan masih sedikit yang menggunakannya. NFC memungkinkan pengguna mentransfer informasi antara perangkat dengan sentuhan sederhana secara cepat dan mudah. NFC merupakan teknologi yang sangat menarik untuk dikembangkan karena beberapa keunggulan-keunggulannya. Mengingat teknologi NFC saat ini juga sangat berkembang pesat di perangkat bergerak.

Di ITS sendiri, sebenarnya kartu cerdas (*smartcard*) sudah diterapkan. Untuk mahasiswa diterapkan sebagai Kartu Tanda Mahasiswa (KTM) yang sebenarnya sudah memiliki teknologi yang terpasang di dalamnya. Bahkan sebelum tahun 2010, KTM yang dipegang oleh mahasiswa juga sudah dimanfaatkan sebagai kartu ATM di salah satu bank yang ada di Indonesia. Namun belakangan ini, pemanfaatan teknologi yang ada di KTM itu sudah tidak termanfaatkan dengan baik. Bahkan, tidak ada kegunaan khusus dengan kartu tersebut. KTM itu merupakan jenis kartu cerdas Mifare dengan ukuran data yang dapat disimpan

sebanyak 1 Kilobyte (KB) dengan sudah memiliki UID yang unik di setiap kartunya. Sebaiknya KTM tersebut dapat dikelola dan dimanfaatkan kembali dengan baik, khususnya sebagai akses khusus yang dimiliki oleh mahasiswa. Selain itu, di dalam masterplan pembangunan kampus ITS, ada rencana untuk menerapkan sistem satu pintu untuk keluar masuk ITS dan sistem parkir terpusat.

Berdasarkan studi kasus yang telah dijelaskan sebelumnya, dalam Tugas Akhir ini akan dikembangkan perangkat lunak Sistem Keamanan Keluar Masuk Parkir dengan Kartu Cerdas Mifare dan Teknologi Near Field Communication (NFC) Studi Kasus Parkir Jurusan Teknik Informatika. Aplikasi ini diharapkan muncul sebuah karya yang dapat menyelesaikan permasalahan-permasalahan yang telah dijelaskan dengan kelemahan yang minimal.

1.2 Rumusan Permasalahan

Rumusan masalah yang diangkat dalam Tugas Akhir ini dapat dipaparkan sebagai berikut.

1. Bagaimana membuat sistem keamanan kendaraan yang melakukan aktivitas keluar masuk tempat parkir?
2. Bagaimana membuat sistem yang dapat melakukan identifikasi dan verifikasi dengan memanfaatkan kartu cerdas yang memiliki fitur NFC?

1.3 Batasan Permasalahan

Permasalahan yang dibahas dalam Tugas Akhir ini memiliki beberapa batasan, di antaranya adalah sebagai berikut.

1. Perangkat media Mifare yang digunakan berukuran 1K.
2. Aplikasi dibangun menggunakan pustaka tambahan dari Software Development Kit (SDK) ACR122U NFC.
3. Sistem hanya menangani pencatatan keluar masuknya orang di tempat parkir.

1.4 Tujuan

Tujuan dari Tugas Akhir ini adalah sebagai berikut.

1. Merancang dan membangun perangkat lunak menggunakan teknologi NFC dengan perangkat media kartu cerdas Mifare.
2. Mengimplementasikan teknologi baru NFC untuk meningkatkan keamanan dan kenyamanan semua pihak yang terlibat di aktivitas kampus ITS.
3. Merancang dan membangun perangkat lunak untuk identifikasi dan verifikasi keluar masuk parkir di jurusan Teknik Informatika ITS.

1.5 Metodologi

Langkah yang ditempuh dalam pengerjaan Tugas Akhir ini adalah sebagai berikut.

1. Studi Literatur

Studi literatur dilakukan dengan mengumpulkan informasi yang diperlukan untuk perancangan sistem meliputi perencanaan dan pembangunan perangkat lunak, pengembangan arsitektur dan cara kerja NFC melalui media kartu cerdas, pengembangan aplikasi desktop yang dapat berkomunikasi dengan teknologi NFC. Informasi-informasi tersebut diperoleh dari beberapa literatur berupa jurnal maupun buku-buku yang terkait. Analisis kebutuhan juga dilakukan dengan cara menggali kebutuhan dari pihak-pihak yang terlibat seperti SKK ITS dan survei kepada mahasiswa ITS. Beberapa eksplorasi dalam studi literatur adalah eksplorasi pengoperasian NFC dan SDK, eksplorasi proses dan mekanis baca tulis dengan alat baca/tulis ACR122U, dan eksplorasi perangkat media kartu cerdas yang berteknologi NFC, serta eksplorasi penggunaan webcam.

2. Analisis Sistem

Melakukan analisa kebutuhan sistem sebagai solusi atas permasalahan yang dihadapi pengguna. Dari proses tersebut selanjutnya dirumuskan rancangan sistem yang akan dibangun dan dapat menangani permasalahan. Langkah pada tahap ini antara lain:

- a. Analisa aktor yang terlibat dalam sistem.
- b. Perancangan diagram kasus penggunaan sebagai analisa kebutuhan fungsional sistem.
- c. Analisa kebutuhan non-fungsional.
- d. Analisa arsitektur sistem.

3. Perancangan Sistem

Melakukan perancangan sistem dari hasil analisa terhadap sistem. Proses analisa digambarkan dalam bentuk diagram atau bagan untuk mempermudah gambaran rancangan sistem. Langkah perancangan pada tahap ini antara lain:

- a. Rancangan basis data sistem.
- b. Rancangan kelas-kelas yang terlibat dalam aplikasi
- c. Rancangan aplikasi berbasis desktop untuk sistem keamanan keluar masuk parkir.

4. Implementasi Perangkat Lunak

Tahap implementasi merupakan tahap realisasi dari analisis dan perancangan yang telah dilakukan sebelumnya menjadi sebuah baris kode program sehingga perangkat lunak yang dapat digunakan untuk menyelesaikan permasalahan-permasalahan yang telah dijelaskan pada latar belakang. Tahap implementasi meliputi.

- a. Implementasi pembuatan basis data menggunakan sistem manajemen basis data relasional MySQL.

- b. Implementasi pembuatan aplikasi berbasis desktop yang dapat terintegrasi dengan basis data dan teknologi NFC serta webcam.
- c. Implementasi integrasi antara aplikasi sebelumnya dengan perangkat media kartu cerdas dengan teknologi NFC melalui bantuan alat pembaca ACR122U.

5. Pengujian dan Evaluasi

Pada tahap ini akan dilakukan uji coba dan evaluasi terhadap perangkat lunak yang dibuat. Pengujian dan evaluasi akan dilakukan dengan melihat kesesuaian dengan perencanaan. Pengujian dan evaluasi dimaksudkan untuk menemukan kesalahan-kesalahan dalam pengembangan aplikasi secara langsung pada perangkat lunak dan segera dilakukan perbaikan jika terdapat kesalahan. Rencana tahapan-tahapan dari uji coba yang akan dilakukan adalah pengujian *blackbox* dan pengujian performa dan akurasi. Pengujian *blackbox* adalah pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, penguji dapat mendefinisikan kumpulan kondisi masukan dan melakukan pengujian pada spesifikasi fitur atau fungsional program [2]. Pengujian ini dilakukan untuk melihat apakah data yang diminta pengguna dapat muncul secara tepat atau tidak.

6. Penyusunan Buku Tugas Akhir

Pada tahap ini ditulis buku yang bertujuan untuk mendokumentasikan seluruh konsep, rancangan, dasar teori, literatur, proses yang dilakukan dan hasil yang diperoleh selama pengerjaan Tugas Akhir. Buku Tugas Akhir yang ditulis bertujuan untuk memberikan gambaran dari pengerjaan Tugas Akhir dan dapat berguna bagi pembaca yang tertarik untuk melakukan pengembangan aplikasi ini lebih lanjut.

1.6 Sistematika

Pendokumentasian seluruh konsep, rancangan, dasar teori, literatur, proses yang dilakukan dan hasil yang diperoleh selama pengerjaan Tugas Akhir. Buku yang ditulis bertujuan untuk memberikan gambaran dari pengerjaan Tugas Akhir dan berguna untuk pembaca yang tertarik untuk melakukan pengembangan aplikasi ini lebih lanjut.

Buku Tugas Akhir akan terdiri dari beberapa bagian, yaitu:

Bab I Pendahuluan

Bab ini membahas latar belakang masalah, tujuan pembuatan tugas akhir, rumusan permasalahan, batasan permasalahan, metodologi yang digunakan, dan sistematika penyusunan Tugas Akhir.

Bab II Tinjauan Pustaka

Bab ini membahas teori pendukung dan literatur yang berkaitan dengan bahasan dan mendasari pembuatan Tugas Akhir ini.

Bab III Analisis dan Perancangan

Bab ini membahas tentang desain dan rancangan dari perangkat lunak. Rancangan dan desain meliputi data, proses dan arsitektur.

Bab IV Implementasi

Bab ini membahas tentang implementasi hasil analisis dan perancangan dalam bentuk *coding*. Bab ini membahas proses pembangunan perangkat lunak.

Bab V Pengujian dan Evaluasi

Bab ini membahas tentang pengujian aplikasi berdasarkan skenario yang telah ditentukan. Melakukan evaluasi fitur aplikasi apakah telah memenuhi kebutuhan fungsional.

Bab VI Kesimpulan dan Saran

Bab ini berisi kesimpulan dan saran baik dari proses pengembangan perangkat lunak dan hasil uji coba.

Daftar Pustaka

Merupakan daftar referensi yang digunakan dalam pembuatan Tugas Akhir.

Lampiran

Merupakan bab tambahan yang berisi kode-kode sumber yang penting pada aplikasi dan hasil pengujian yang telah dilakukan.

BAB II

TINJAUAN PUSTAKA

Pada bab ini akan dibahas mengenai dasar teori dan literatur yang menjadi dasar pembuatan Tugas Akhir.

2.1 Profil ITS

Kampus ITS Sukolilo menempati area seluas 180 hektar dengan luas bangunan seluruhnya kurang lebih 150.000 m². Selain itu, terdapat kampus Manyar yang dipergunakan oleh program D-3 Teknik Sipil dengan luas bangunan 5.176 m² dan kampus ITS Cokroaminoto yang dipergunakan untuk magister manajemen serta beberapa lembaga kerjasama dengan luas bangunan 4.000 m².

Sampai tahun 2013, ITS memiliki 5 fakultas dengan 12 Program Doktorat, 16 Program Magister, 29 jurusan/program studi tingkat Sarjana, 6 Program Studi D-3, 1 Program Studi D-4. Jumlah mahasiswa ITS yang terdaftar pada tahun ajaran 2012/2013 berjumlah kurang lebih 20.000 mahasiswa baik itu tingkat Sarjana, Diploma, Program Doktorat, dan Program Magister.

Selain itu, ITS mempunyai staf pengajar sebanyak 960 orang baik itu tingkat profesor, doktor, master, dan lainnya yang merupakan lulusan perguruan tinggi terkemuka di dalam dan luar negeri serta profesional di bidangnya, menjadikan ITS sebagai acuan perguruan tinggi lain di kawasan Indonesia Timur. Staf non akademik ITS berjumlah 1194 orang (PNS dan Non-PNS) yang tersebar pada berbagai bagian administrasi mulai dari rektorat, pustakawan, sampai jurusan-jurusan dan laboratorium-laboratorium yang ada di ITS [3].

2.2 Identifikasi dan Verifikasi

Identifikasi menurut Kamus Besar Bahasa Indonesia adalah penentuan atau penetapan identitas seseorang. Identifikasi juga bisa diartikan sebagai kecenderungan atau keinginan dalam diri seseorang untuk menjadi sama dengan individu lain yang ditiru.

Verifikasi menurut Kamus Besar Bahasa Indonesia adalah pemeriksaan tentang kebenaran laporan, pernyataan, dan sebagainya. Verifikasi juga bisa diartikan sebagai proses pengecekan data apakah sudah sesuai dengan aturan atau tidak.

2.3 Rancang Bangun Perangkat Lunak

Rancang bangun perangkat lunak merupakan tahapan-tahapan teknis untuk membangun sebuah perangkat lunak yang melingkupi tahap analisis permasalahan dan kebutuhan, tahap perencanaan, tahap analisis sistem dan tahap implementasi, serta tahapan aktivitas perawatan dan pemeliharaan perangkat lunak.

Rancang bangun perangkat lunak diperlukan untuk menentukan konsep, strategi, dan praktik yang baik dan diterapkan untuk menciptakan sebuah perangkat lunak yang berkualitas tinggi, sesuai anggaran biaya, mudah dalam pemeliharannya, serta tidak membutuhkan waktu yang lama dalam pembangunannya [4]. Beberapa contoh model rancang bangun perangkat lunak yang cukup terkenal dan banyak dipakai antara lain adalah model air terjun dan model iterasi.

2.4 *Near Field Communication* (NFC)

NFC adalah teknologi baru yang merupakan pengembangan dari teknologi Radio Frequency Identification (RFID). NFC adalah suatu set dari standar untuk seluler pintar dan perangkat serupa untuk melakukan komunikasi radio antara satu dan lainnya dengan menyentuhkan kedua perangkat secara bersamaan atau mendekatkan kedua perangkat dalam jarak tertentu, dan biasanya tidak lebih dari beberapa sentimeter [5]. NFC beroperasi pada frekuensi 13.56 MHz dan hanya memiliki jarak maksimal 20 cm

dalam melakukan transfer data. NFC memiliki kecepatan transfer data dari 106 kbit/s, 212 kbit/s dan 424 kbit/s dan 848 kbit/s (keterangan 8 bit = 1 karakter, 1 kilobit = 1024 bit) [6].

2.5 *Smartcard Reader/Writer ACR122U NFC*

Smartcard Reader/Writer ACR122U NFC Adalah *contactless smartcard reader/writer* yang dapat terhubung ke komputer dan dikembangkan berdasarkan teknologi RFID. Sesuai dengan standar ISO/IEC 18092 untuk NFC, yang bisa mendukung tidak hanya teknologi kartu Mifare dan ISO 14443 A dan B, tetapi juga semua empat jenis Smartcard Reader ACR122U NFC yang kompatibel dengan kedua versi komunikasi, yaitu CCID dan PC/SC. Perangkat USB yang dapat digunakan ketika alat dipasang (*plug and play*) yang memungkinkan interoperabilitas atau kemampuan untuk digunakan dan kecocokan dengan perangkat dan aplikasi yang berbeda.

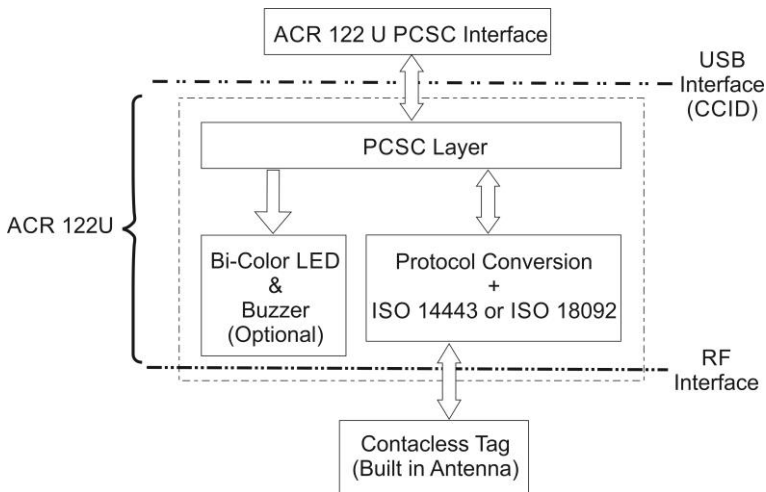
Satu perangkat Smartcard Reader/Writer ACR122U NFC terdiri dari alat pembaca/penulis NFC, beberapa kartu cerdas yang sudah mendukung penerapan teknologi NFC, serta Software Development Kit (SDK). Contoh perangkat Smartcard Reader/Writer ACR122U NFC seperti pada Gambar 2.1.



Gambar 2. 1 Perangkat ACR 122 U, kartu cerdas dan SDK

Dengan kecepatan akses hingga 424 kbps dan kecepatan penuh USB yaitu hingga 12 Mbps, ACR122U juga dapat membaca dan menulis lebih cepat dan efisien. Operasi kedekatan jarak alat pembaca ACR122U adalah sampai 5 cm, tergantung dari tipe kartu cerdas yang digunakan. Untuk meningkatkan tingkat keamanan, ACR122 dapat diintegrasikan dengan ISO 7816-3. Selanjutnya, Smartcard Reader ACR122U NFC Reader/Writer yang juga tersedia dalam bentuk modul, memungkinkan adanya integrasi yang mudah ke mesin atau perangkat yang lebih besar [7].

Gambar 2.2 merupakan gambaran diagram alir komunikasi yang terjadi di dalam ACR122U. ada 2 antarmuka yang berbeda. Yang pertama antarmuka dari kartu cerdas menuju alat pembaca ACR122U itu sendiri yang disebut dengan *RF Interface*. Yang kedua adalah antar muka USB atau yang biasa disebut *USB Interface/CCID* yang menghubungkan antara alat pembaca dengan PC.



Gambar 2. 2 Diagram Alir Komunikasi ACR122U

Fitur yang dimiliki oleh ACR122U NFC adalah sebagai berikut.

1. kompatibel CCID,
2. kompatibel PC/SC,
3. baca/tulis dengan kecepatan 424 Kbps,
4. *built-in antena* dengan jarak baca 5 cm (tergantung dari tipe kartu),
5. mendukung kartu cerdas dengan berbagai tipe seperti mifare, ISO 14443A, ISO 14443B, Felica dan semua jenis tag NFC (ISO/IEC 18092), dan
6. mendukung 3 mode NFC: pembaca, emulsi kartu dan *peer-to-peer*.

2.6 Microsoft Visual Studio

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (suite) yang dapat digunakan untuk melakukan pengembangan aplikasi baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya dalam bentuk aplikasi konsol, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup compiler, *Software Development Kit* (SDK), *Integrated Development Environment* (IDE) dan dokumentasinya (umumnya berupa MSDN Library). Compiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, dan masih banyak yang lainnya.

Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam *native code* (dalam bahasa mesin yang berjalan di atas Windows) ataupun *managed code* (dalam bentuk Microsoft Intermediate Language di atas .NET Framework). Selain itu, Visual Studio juga dapat digunakan untuk mengembangkan aplikasi Silverlight, aplikasi Windows Mobile (yang berjalan di atas .NET Compact Framework).

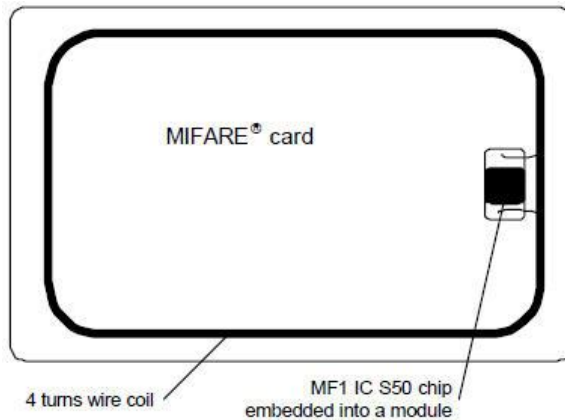
2.7 Microsoft Visual C# (C Sharp)

Microsoft Visual C# adalah sebuah program alat bantu pemrograman (*Rapid Application Development Tool*) yang dibuat oleh Microsoft Corporation dan dapat digunakan untuk membuat program berbasis grafis dengan menggunakan bahasa pemrograman mirip C++. Program ini telah dimasukkan ke dalam produk Microsoft Visual Studio, bersama-sama dengan Visual C++, Visual Basic, dan yang lainnya. Sejauh ini, program dengan menggunakan Visual C# adalah yang paling banyak digunakan oleh para programmer untuk membuat program. Selain itu, Microsoft Visual C# ini merupakan sebuah bahasa pemrograman yang bersifat Pemrograman Berorientasi Obyek (OOP) yang membuat bahasa pemrograman ini lebih dinamis.

2.8 Mifare

Mifare adalah merek dagang serangkaian chip produk dari NXP Semiconductors, yang secara luas diaplikasikan pada *contactless smart cards* dan *proximity cards*. Nama Mifare berasal dari *Mikron Fare Collection System*, mencakup teknologi *proprietary* berdasarkan berbagai tingkat standar *contactless smart cards* yaitu ISO/IEC 14443 Tipe A dengan 13,56 MHz.

Mifare sering dianggap sebagai teknologi “kartu pintar”. Hal ini didasarkan pada kemampuan untuk membaca dan menulis ke kartu. Membaca dengan jarak yang dekat, Mifare cocok untuk melakukan fungsi *Increment/Decrement*. Berikut 7 jenis kartu Mifare *contactless card* yaitu Mifare Classic, Ultralight, Ultralight C, SmartMX, DESFire, DESFire EV1, Mifare Plus, dan Mifare SAM AV2 [8].



Gambar 2. 3 Arsitektur Kartu Cerdas Mifare

Di dalam kartu cerdas *Mifare* ini terdapat beberapa informasi yang bisa didapatkan. Berikut adalah beberapa informasi yang berhasil didapatkan dari kartu cerdas ini dengan menggunakan instruksi Application Protocol Data Unit (APDU) dan diambil dari 3 jenis kartu cerdas, yaitu kartu cerdas *Mifare*, Kartu Tanda Mahasiswa ITS, dan Kartu Tanda Penduduk. Dan ketiga kartu cerdas ini nantinya yang akan dapat digunakan dalam Tugas Akhir ini.



Gambar 2. 4 Kartu Cerdas Mifare

Dari kartu cerdas Mifare 1K ini diperoleh data-data sebagai berikut.

- Error Code: 00
- Field: 00 (*RF Field is NOT Present and NOT Detected*)
- Number of Target: 01
- Logical Number: 01
- Bit Rate in Reception: 00 (106 kbps)
- Bit Rate in Transmission: 00 (106 kbps)
- Modulation Type: 00 (ISO14443 or Mifare)
- Active Protocol: T=CL
- Card Type: ISO 14443A, Part3 (Mifare 1K)
- UID: 96 39 00 F9
- ATS Not Supported



Gambar 2. 5 Kartu Tanda Mahasiswa

Dari Kartu Tanda Mahasiswa, diperoleh data-data sebagai berikut.

- Error Code: 00
- Field: 00 (*RF Field is NOT Present and NOT Detected*)
- Number of Target: 01
- Logical Number: 01

- Bit Rate in Reception: 00 (106 kbps)
- Bit Rate in Transmission: 00 (106 kbps)
- Modulation Type: 00 (ISO14443 or Mifare)
- Active Protocol: T=CL
- Card Type: ISO 14443A, Part3 (Mifare 1K)
- UID: 6C 35 FD 9A
- ATS Not Supported



Gambar 2. 6 Kartu Tanda Penduduk

Dari Kartu Tanda Penduduk, diperoleh data-data sebagai berikut.

- Error Code: 01
- Field: 00 (RF Field is NOT Present and NOT Detected)
- Number of Target: 01
- Logical Number: 01
- Bit Rate in Reception: 00 (106 kbps)
- Bit Rate in Transmission: 00 (106 kbps)
- Modulation Type: 00 (ISO14443 or Mifare)
- Active Protocol: T=CL
- UID: 04 1E 7D 8A 86 26 80
- ATS: 10 78 B3 C4 02 65 4B 54 50 30 44 32 65 4B 54 50

2.9 MySQL

MySQL merupakan sistem manajemen basis data yang bersifat *opensource*. MySQL dikembangkan, didistribusikan dan didukung oleh perusahaan Oracle. MySQL merupakan sistem manajemen basis data yang berfungsi untuk menambah, menyimpan dan mengakses data yang tersimpan di dalam basis data yang ada di komputer. Basis data MySQL bersifat relasional.

Keuntungan dari MySQL adalah mendukung untuk standar SQL, dapat berjalan di berbagai sistem operasi seperti Windows, Linux, FreeBSD, dan Mac OS. Antar muka MySQL juga tersedia untuk berbagai bahasa pemrograman antara lain C/C++, C#, Java, PHP, Python dan Ruby [9].

2.10 Application Protocol Data Unit (APDU)

APDU (*Application Protocol Data Unit*) merupakan komunikasi antara sebuah pembaca dan kartu. Struktur dari APDU ini mengikuti standar dari ISO 7816. Ada dua kategori APDU yaitu Perintah APDUs dan APDUs Respon. Perintah APDU dikirim oleh pembaca untuk kartu, dan wajib mengandung 4 byte header yaitu (CLASS, INS, P1, P2) dan data byte mulai dari 0 sampai 255 byte data. Sebuah APDU Respon dikirim oleh kartu untuk pembaca yang mengandung 0-65536 byte data, dan 2 byte status wajib (SW1, SW2) [10]. Untuk lebih jelasnya bisa dilihat pada Gambar 2.3 sampai Gambar 2.5.

Command	Class	INS	P1	P2	Lc	Data In	Le
ISO 7816 Part 4 Command	-	-	-	-	Length of the Data In	-	Expected length of the Response Data

Gambar 2. 7 Format Perintah APDU sesuai ISO 7816

Response	Data Out		
Result	Response Data	SW1	SW2

Gambar 2. 8 Format Respon APDU

Results	SW1 SW2	Meaning
Success	90 00h	The operation completed successfully.
Error	63 00h	The operation failed.

Gambar 2. 9 Penjelasan Respon APDU SW1 dan SW2

Contoh Perintah APDU untuk mendapatkan UID yang ada di dalam kartu cerdas.

Perintah APDU: FF CA 00 00 00 → Respon APDU [UID] 90 00.
Respon APDU 90 00 berarti Berhasil.

2.11 EasyWebCam Library

EasyWebCam Library adalah pustaka tambahan yang dapat digunakan untuk dapat menggunakan fungsi menampilkan video dan mengambil gambar dari webcam dengan C# atau Visual Basic .NET.

Dengan menggunakan pustaka ini, dapat ditambahkan beberapa baris kode untuk menampilkan video dari webcam di dalam aplikasi yang sedang dibuat. Selain itu ada fungsi untuk mengubah aturan resolusi webcam, dan aturan lebih lanjut mengenai penggunaan webcam seperti kontras, kecerahan, memutar, dll [11].

BAB III

ANALISIS DAN PERANCANGAN

Pada bab ini dibahas analisa kebutuhan, proses bisnis dan desain dari perangkat lunak yang dibangun dalam Tugas Akhir ini. Bagian awal bab akan dibahas tentang analisa permasalahan dan kebutuhan masyarakat. Berikutnya dibahas fungsional sistem yang berdasarkan hasil analisis kebutuhan. Bagian berikutnya akan dibahas rancangan perangkat lunak yang ditujukan untuk memberikan gambaran tentang aplikasi yang akan dibuat.

3.1. Analisis

Pada subbab berikut akan dijelaskan analisis pembuatan aplikasi Sistem Keamanan Kampus dengan menggunakan teknologi NFC dan *Mifare*. Analisis yang dilakukan meliputi analisis permasalahan, deskripsi umum sistem, spesifikasi kebutuhan sistem dan Identifikasi pengguna sistem.

2.1.1 Analisis Permasalahan

ITS merupakan salah satu perguruan tinggi teknik yang ternama di Indonesia, khususnya di wilayah timur Indonesia. Aktivitas kampus yang cukup padat membuat sirkulasi keluar masuk ke area kampus sangat tinggi. Terlebih lagi wilayah kampus ITS ini yang juga menjadi satu bagian dengan kawasan Perumahan Dosen dan Karyawan.

Berdasarkan informasi yang didapat dari Satuan Keamanan Kampus (SKK-ITS) dalam tahun 2014 saja, kasus kehilangan yang terjadi di wilayah Kampus ITS sudah sangat banyak. Contohnya kasus kehilangan dua motor sekaligus di parkir pusat KPA-ITS pada awal Februari ini. Berikutnya kasus kehilangan dua motor kembali terjadi secara bersamaan di parkir Jurusan Arsitektur, yang pada saat itu bertepatan dengan hari libur ITS

(Sabtu dan Minggu). Selain itu, di Kampus ITS ini juga banyak terjadi kasus kehilangan seperti tas, laptop, telepon genggam dan barang berharga lainnya.

Banyak pihak yang mengluhkan keadaan ini, khususnya dari kalangan mahasiswa ITS sendiri. Karena berdasarkan survei yang dilakukan oleh Kementerian Dalam Negeri BEM ITS melalui kuesioner yang dibagikan secara acak kepada 890 mahasiswa ITS. Didapatkan bahwa 82% mahasiswa ITS menyatakan tidak puas hingga cukup puas terhadap tingkat kepuasan penjagaan keamanan parkir (kendaraan bermotor), 87% menyatakan tidak puas hingga cukup puas terhadap tingkat kepuasan penjagaan keamanan lingkungan kampus.

Teknologi NFC merupakan sebuah teknologi baru yang merupakan pengembangan dari teknologi RFID yang sudah banyak dikenal dan digunakan dalam aktivitas sehari-hari. NFC merupakan teknologi yang digunakan untuk transfer data dari satu perangkat ke perangkat lainnya. NFC memungkinkan pengguna mentransfer informasi antara perangkat dengan sentuhan sederhana secara cepat dan mudah. Selain itu, NFC juga dapat diintegrasikan dengan kartu cerdas dan perangkat lainnya.

Di ITS sudah menerapkan kartu cerdas sebagai Kartu Tanda Mahasiswa. Namun sangat disayangkan pemanfaatan kartu cerdas itu dalam kehidupan sehari-hari bisa dibilang tidak ada. Selain itu, di dalam Rencana pembangunan ITS, ada rencana untuk menerapkan sistem satu pintu untuk keluar masuk ITS dan sistem parkir terpusat.

Dengan perkembangan teknologi, dan kondisi-kondisi tersebut. Permasalahan ini dapat diselesaikan dengan memanfaatkan Kartu Tanda Mahasiswa yang merupakan kartu cerdas sebagai alat bantu untuk membuat sistem keamanan kampus sehingga nantinya diharapkan ada peningkatan keamanan yang ada di kawasan kampus ITS.

3.1.2 Deskripsi Umum Sistem

Aplikasi yang dibangun pada Tugas Akhir ini bernama Sistem Keamanan Kampus ITS. Sistem Keamanan Kampus ini adalah aplikasi berbasis *desktop* yang dibedakan menjadi 2 hak akses, yaitu hak akses untuk Administrator, dan hak akses untuk petugas. Aplikasi ini memanfaatkan teknologi NFC dan kartu cerdas jenis Mifare untuk melakukan identifikasi dan verifikasi setiap orang yang keluar masuk dengan kendaraan di pos penjagaan tempat parkir. Kemudian setiap pengguna yang akan keluar dari tempat parkir akan dilakukan pengecekan kembali yang memastikan bahwa orang yang keluar itu benar sesuai dengan pada saat dia masuk ke tempat parkir sebelumnya. Semua transaksi itu diintegrasikan dengan basis data MySQL.

Untuk masuk ke dalam tempat parkir, pengguna harus mendekatkan kartu cerdas ke alat pembaca NFC, kemudian nantinya alat pembaca itu mencatat setiap transaksi masuk pengguna itu ke dalam sistem. Pada Setiap transaksi dilakukan pengambilan gambar pada saat itu siapa yang masuk dan kendaraan apa yang dikendarainya. Ketika pengguna hendak keluar dari tempat parkir, pengguna harus kembali mendekatkan kartu cerdas ke alat pembaca yang tersedia, untuk dilakukan verifikasi apakah benar pengguna yang keluar ini sesuai data pengguna yang telah disimpan sebelumnya pada saat dia masuk.

Administrator dapat melakukan pengisian data baru civitas akademika dan melihat laporan rekapitulasi kendaraan dan orang yang telah keluar masuk di tempat parkir tersebut.

Diharapkan dengan adanya Sistem Keamanan Kampus ini, dapat dengan lebih mudah mengetahui siapa saja yang masuk ke dalam tempat parkir dan orang yang keluar dari tempat parkir. Sehingga, akan lebih meningkatkan keamanan yang ada di dalam tempat parkir itu sendiri.

3.1.3 Spesifikasi Kebutuhan Perangkat Lunak

Berdasarkan uraian mengenai cakupan-cakupan permasalahan yang telah dijelaskan sebelumnya, maka terdapat beberapa spesifikasi perangkat lunak yang harus dipenuhi dalam pembangunan aplikasi ini agar sistem mampu mengakomodasi serta menyelesaikan permasalahan yang menjadi domain utama dalam Tugas Akhir ini. Spesifikasi kebutuhan sistem ini dibagi menjadi dua, yaitu kebutuhan fungsional dan kebutuhan non fungsional.

3.1.3.1 Spesifikasi Kebutuhan Fungsional Sistem

Kebutuhan fungsional berisi semua kebutuhan utama yang harus dipenuhi di dalam aplikasi agar sistem mampu berjalan dengan baik. Kebutuhan fungsional secara tidak langsung mendeskripsikan fungsionalitas dasar perangkat lunak. Kebutuhan fungsional juga mendefinisikan layanan yang harus disediakan dan reaksi sistem terhadap masukan atau pada situasi tertentu. Daftar kebutuhan fungsional dapat dilihat pada Tabel 3.1.

Tabel 3. 1 Daftar Kebutuhan Fungsional

No.	Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
1.	F-0001	Mencatat kendaraan masuk	Pengguna dapat mencatat kendaraan yang masuk ke dalam basis data
2.	F-0002	Melakukan verifikasi kendaraan keluar	Pengguna dapat mencatat dan melakukan verifikasi setiap kendaraan yang keluar.
3.	F-0003	Melihat transaksi keluar masuk	Pengguna dapat melihat transaksi kendaraan yang telah keluar atau masuk ke dalam tempat parkir

No.	Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
4.	F-0004	Mengelola data pengguna	Pengguna dapat mengelola data pengguna, termasuk mengubah dan menambahkan data pengguna yang baru.
5.	F-0005	Mengelola rekapitulasi keluar masuk kendaraan	Pengguna dapat mengelola laporan/rekapitulasi kendaraan yang telah keluar masuk

3.1.3.2 Spesifikasi Kebutuhan Non Fungsional

Kebutuhan non fungsional berisi kebutuhan-kebutuhan sampingan yang mendukung aplikasi agar sistem dapat berjalan secara lebih efektif maupun efisien. Kebutuhan non fungsional yang diimplementasikan pada sistem yang akan dibangun dalam Tugas Akhir ini adalah permasalahan otentikasi untuk menggunakan atau mengakses subsistem-subsistem yang akan dibangun. Otentikasi ini berdasarkan peran pengguna (*user's role*) yang dijelaskan pada subbab Identifikasi Pengguna. Daftar kebutuhan non fungsional dapat dilihat pada Tabel 3.2.

Tabel 3. 2 Daftar Kebutuhan Non Fungsional

No.	Kode Kebutuhan	Kebutuhan Non Fungsional	Deskripsi
1.	NF-001	Login	Membedakan hak akses aktor berdasarkan peran pengguna (<i>user's role</i>)

3.1.4 Identifikasi Pengguna

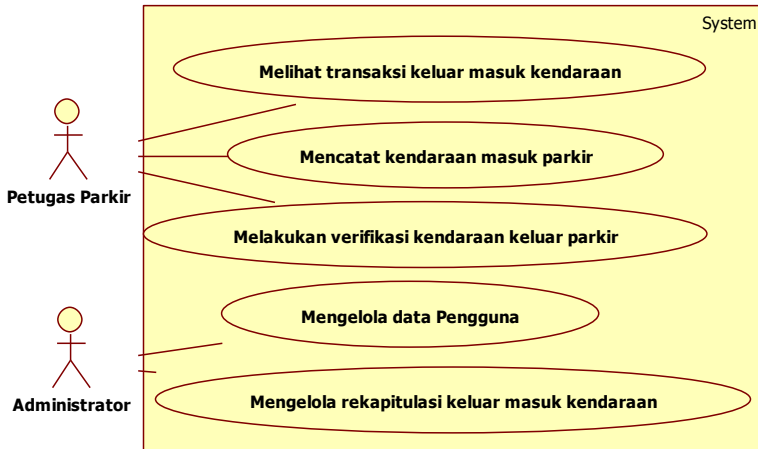
Identifikasi Pengguna atau aktor mendefinisikan entitas-entitas yang terlibat dan berinteraksi langsung dengan sistem. Entitas ini bisa berupa manusia maupun sistem atau perangkat lunak yang lain. Berdasarkan deskripsi umum yang telah dijelaskan pada bagian sebelumnya, maka dapat diketahui bahwa pengguna yang akan menggunakan aplikasi ini ada dua, yang bertanggung jawab dalam masalah Data Mahasiswa yang disebut Administrator. Dia yang bertanggung jawab terhadap data pengguna yang ada. Yang kedua adalah Petugas Parkir yang bertugas langsung pada setiap transaksi keluar masuk kendaraan.

3.2 Perancangan

Subbab berikut membahas tentang perancangan dari aplikasi Sistem Keamanan Kampus ini. Tahap perancangan dalam subbab ini dibagi menjadi beberapa bagian, yaitu perancangan skenario kasus penggunaan, perancangan basis data, perancangan arsitektur, perancangan antarmuka dan perancangan proses. Pembahasan lebih detail akan dibahas berikut ini.

3.2.1 Skenario Kasus Penggunaan

Berdasarkan analisis spesifikasi kebutuhan fungsional dan identifikasi pengguna dari sistem, dibuat kasus penggunaan sistem. Kasus penggunaan sistem digambarkan dalam bentuk diagram kasus penggunaan dan dalam tabel penjelasan kasus penggunaan. Terdapat 5 kasus penggunaan, seperti yang akan dijelaskan pada Gambar 3.1. dan Tabel 3.3.



Gambar 3. 1 Diagram Kasus Penggunaan

Tabel 3. 3 Daftar Kasus Penggunaan

No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
1.	UC-001	Melihat transaksi keluar masuk kendaraan.	Aktor dapat melihat semua transaksi keluar masuk kendaraan yang telah terjadi dan tersimpan di dalam basis data berdasarkan pencarian yang dilakukan oleh aktor.
2.	UC-002	Mencatat kendaraan masuk parkir.	Aktor dapat mencatat setiap kendaraan yang masuk ke dalam parkir dengan menambah data dari kartu cerdas dan webcam

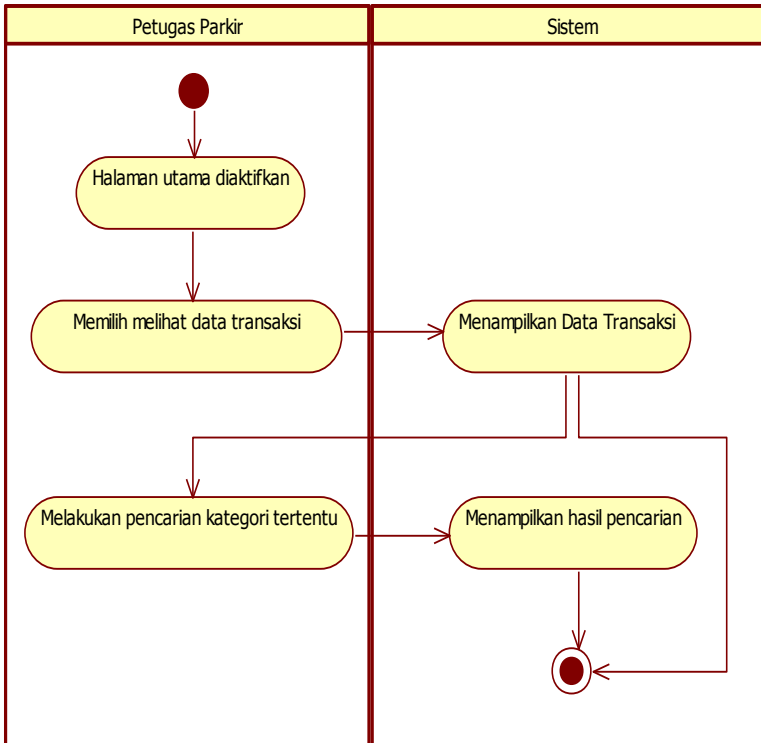
No.	Kode Kasus Penggunaan	Nama Kasus Penggunaan	Keterangan
3.	UC-003	Melakukan verifikasi kendaraan keluar parkir.	Aktor dapat melakukan verifikasi setiap kendaraan yang akan keluar berdasarkan data pada saat penggunaan masuk.
4.	UC-004	Mengelola data pengguna.	Aktor dapat menambah, mengubah dan menghapus data pengguna
5.	UC-005	Mengelola rekapitulasi keluar masuk kendaraan.	Aktor dapat melihat, menambah, mengubah dan menghapus data transaksi keluar masuknya kendaraan.

3.2.1.1 Kasus Penggunaan Melihat Transaksi Keluar Masuk Kendaraan

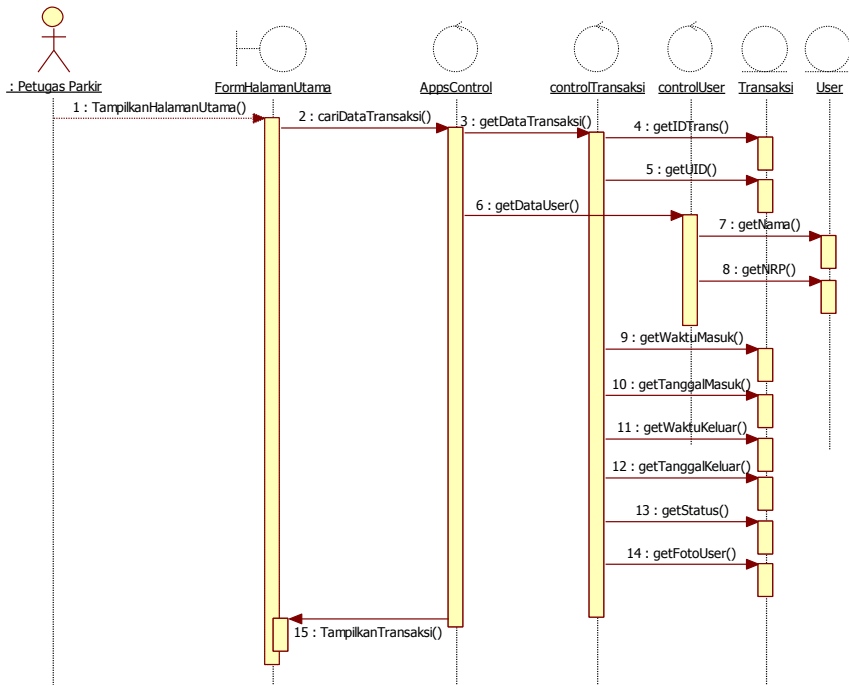
Pada kasus penggunaan UC-001, pengguna dapat melihat semua transaksi keluar masuk kendaraan yang telah terjadi dan disimpan dalam basis data. Pengguna pada kasus ini hanya dapat melihat data transaksi keluar masuk saja. Tidak bisa melakukan perubahan data kecuali pada saat nanti melakukan verifikasi kendaraan ketika akan keluar dari Parkir. Rincian alur kasus penggunaan UC-001 dijelaskan pada Tabel 3.4. Diagram aktifitas kasus penggunaan UC-001 dijelaskan pada Gambar 3.2. Diagram sekuensial atau diagram alir kasus penggunaan UC-001 diejelaskan pada Gambar 3.3.

Tabel 3. 4 Rincian Alur Kasus Penggunaan UC-001

Nama	Melihat Transaksi Keluar Masuk Kendaraan
Kode	UC-001
Aktor	Petugas Parkir
Kondisi Awal	Data Transaksi keluar masuk kendaraan belum diketahui dan Pengguna masuk ke dalam sistem
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna meminta tampilkan data transaksi keluar masuk. A1. Pengguna meminta melihat data transaksi tertentu 2. Pengguna menekan tombol cari 3. Sistem menampilkan data transaksi yang telah terjadi. 4. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Pengguna meminta melihat data transaksi tertentu</p> <ol style="list-style-type: none"> 1. Pengguna memasukkan pencarian berdasarkan kategori tertentu 2. Kembali ke aliran kejadian normal nomor 2
Kondisi Akhir	Data transaksi keluar masuk ditampilkan
Kebutuhan Khusus	-



Gambar 3. 2 Diagram Aktivitas Kasus Penggunaan UC-001



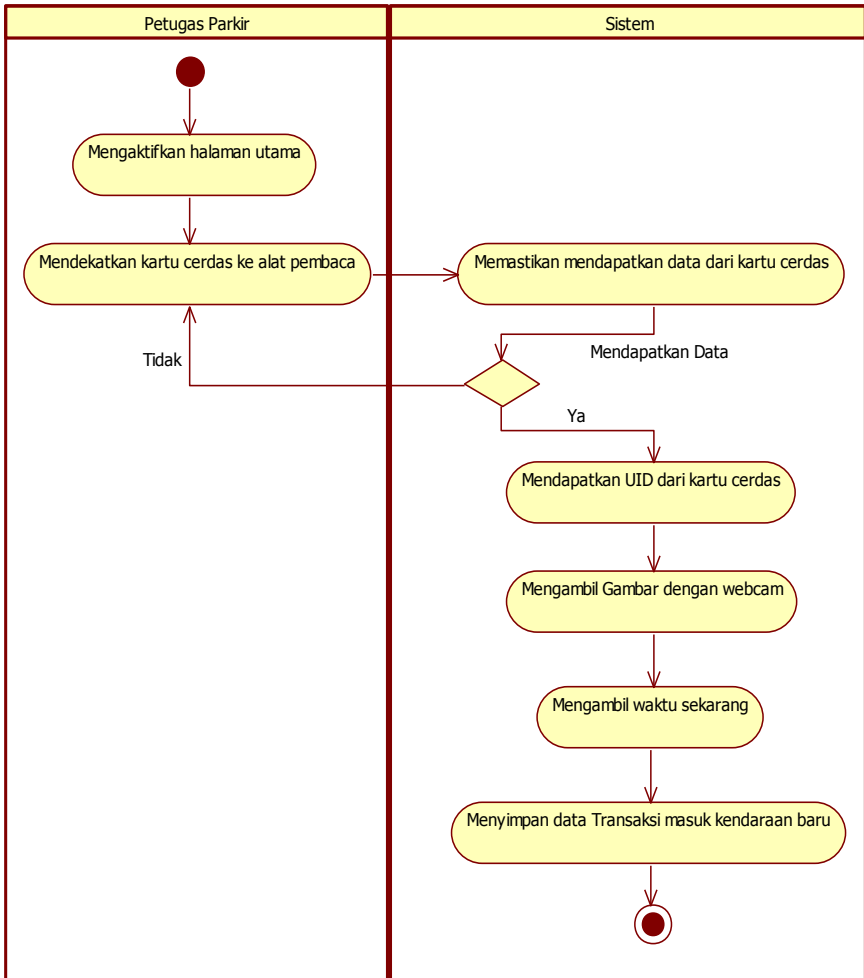
Gambar 3. 3 Diagram Sekuensial Kasus Penggunaan UC-001

3.2.1.2 Kasus Penggunaan Mencatat Kendaraan Masuk Parkir

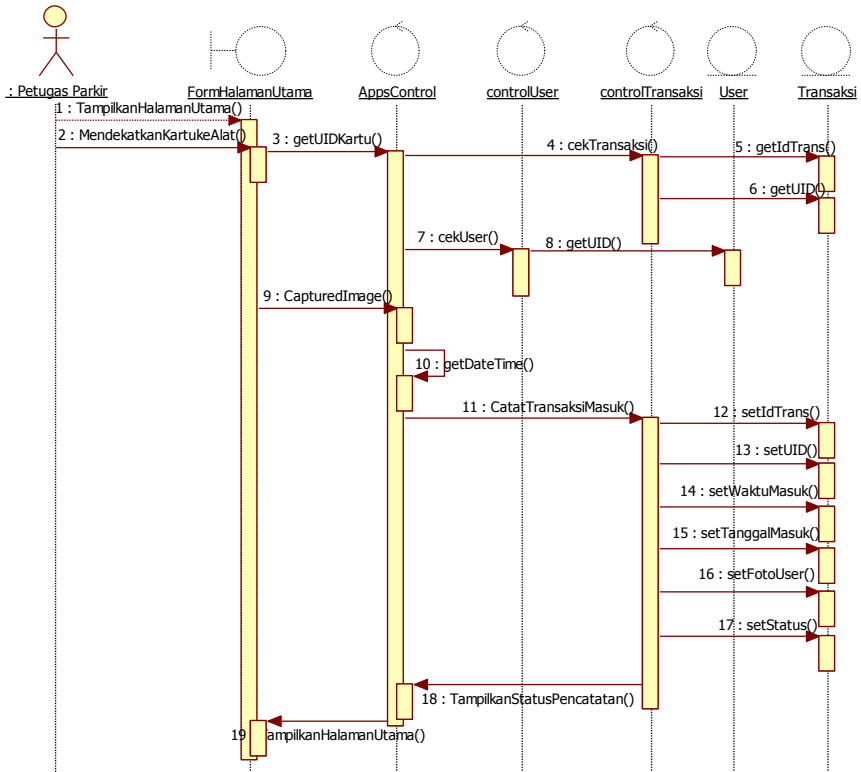
Pada kasus penggunaan UC-002, aktor dapat melakukan pencatatan setiap pengguna yang menggunakan kendaraan yang akan masuk ke dalam tempat parkir. Untuk melakukan pencatatan dibutuhkan alat pembaca NFC dan webcam. Rincian alur kasus penggunaan UC-002 dijelaskan pada Tabel 3.5. Diagram aktifitas kasus penggunaan UC-002 dijelaskan pada Gambar 3.4. Diagram sekuensial atau diagram alir kasus penggunaan UC-002 diejelaskan pada Gambar 3.5.

Tabel 3. 5 Rincian Alur Kasus Penggunaan UC-002

Nama	Mencatat Kendaraan Masuk Parkir
Kode	UC-002
Aktor	Petugas Parkir
Kondisi Awal	Belum ada transaksi masuk yang disimpan dan Petugas sudah masuk ke dalam sistem
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. User mendekatkan kartu cerdas ke alat pembaca NFC 2. UID dari kartu cerdas terbaca oleh alat pembaca NFC <ol style="list-style-type: none"> A1. UID dari kartu cerdas tidak terbaca oleh alat pembaca NFC 3. Webcam mengambil gambar pada saat itu 4. Sistem mengambil waktu kejadian pada saat itu 5. Sistem menyimpan data-data kendaraan masuk termasuk gambar dari webcam. 6. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. UID dari kartu cerdas tidak terbaca oleh alat pembaca NFC</p> <ol style="list-style-type: none"> 1. Kembali ke aliran kejadian normal nomor 1.
Kondisi Akhir	Data Transaksi pengguna saat masuk dapat disimpan ke dalam basis data
Kebutuhan Khusus	Webcam dan alat pembaca NFC sudah terpasang dan berfungsi dengan baik



Gambar 3. 4 Diagram Aktivitas Kasus Penggunaan UC-002



Gambar 3. 5 DiagramSekuensial Kasus Penggunaan UC-002

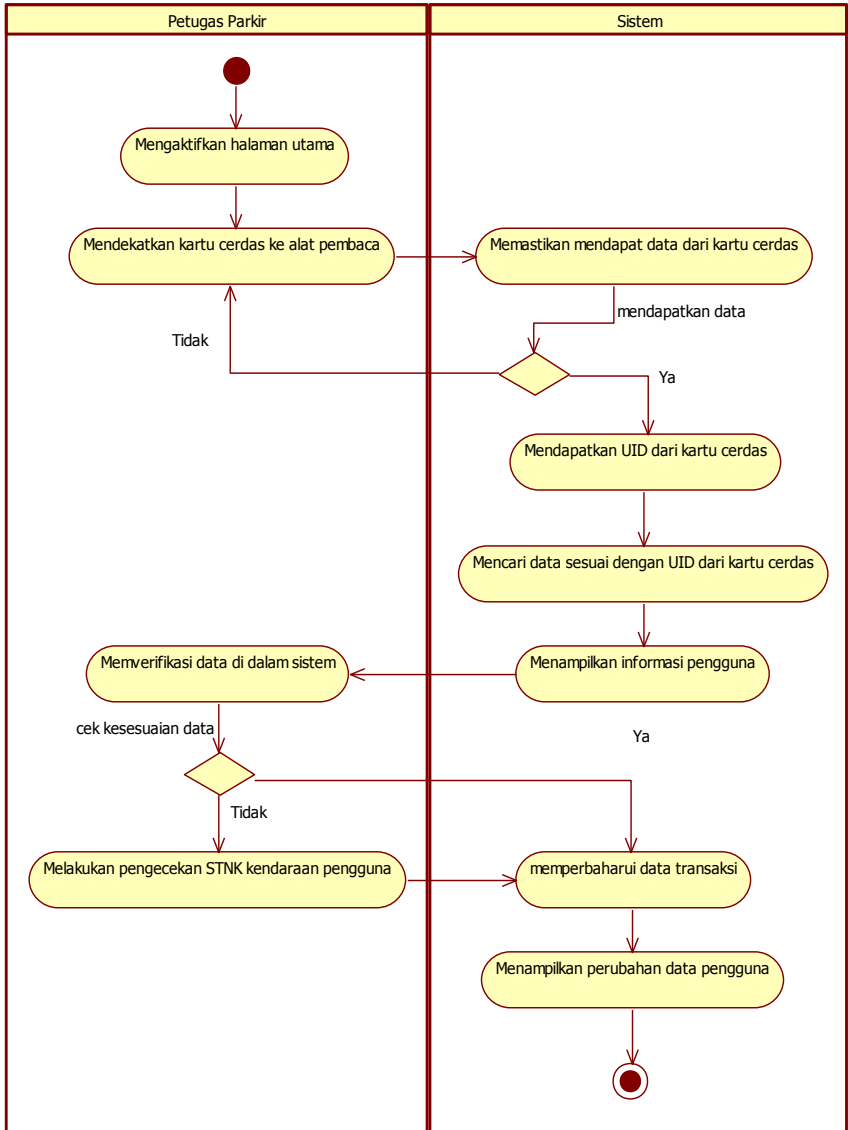
3.2.1.3 Kasus Penggunaan Melakukan Verifikasi Kendaraan Keluar Parkir

Pada kasus penggunaan UC-003, Petugas Parkir dapat melakukan verifikasi terhadap pengguna yang akan keluar dari parkir terhadap data yang telah disimpan sebelumnya. Untuk melakukan verifikasi, juga diperlukan alat pembaca NFC untuk mendapatkan data pengguna yang terdapat di kartu cerdas dan kemudian sistem mencari data yang ada di dalam basis data sistem, kemudian petugas melakukan verifikasi terhadap pengguna tersebut. Apakah data pada saat awal masuk itu sama dengan data ketika pengguna akan keluar dari parkir. Rincian alur kasus penggunaan UC-003 dijelaskan pada Tabel 3.6. Diagram aktifitas kasus penggunaan UC-003 dijelaskan pada Gambar 3.6. Diagram sekuensial kasus penggunaan UC-003 dijelaskan pada Gambar 3.7.

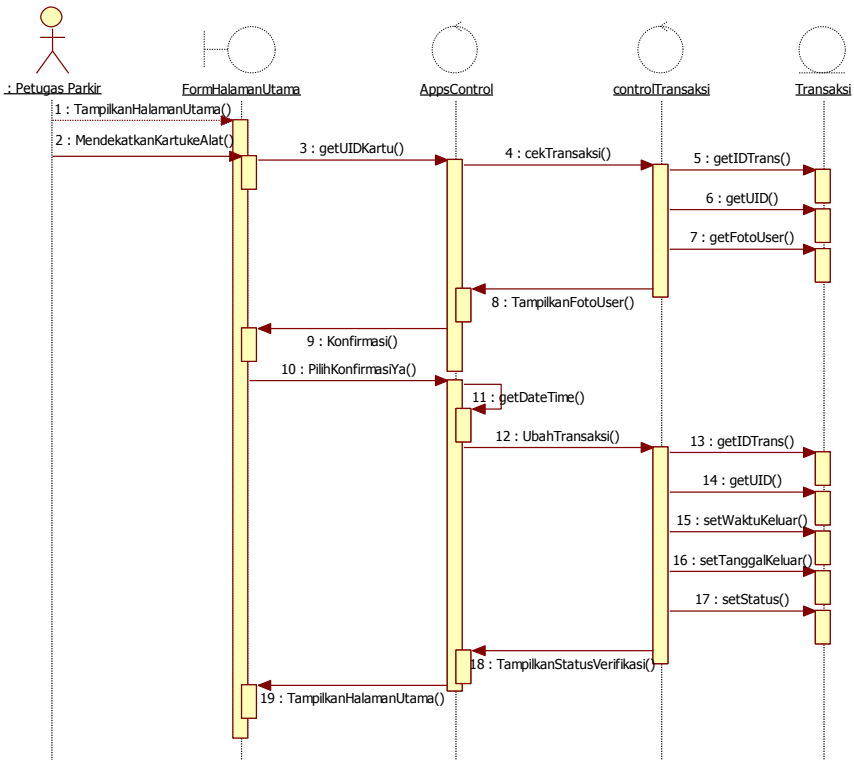
Tabel 3. 6 Rincian Alur Kasus Penggunaan UC-003

Nama	Melakukan Verifikasi Kendaraan Keluar Parkir
Kode	UC-003
Aktor	Petugas Parkir
Kondisi Awal	Pengguna telah dicatat saat masuk dan aktor telah masuk ke dalam sistem
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna mendekatkan kartu cerdas ke alat pembaca NFC 2. Sistem mencari data transaksi berdasarkan UID yang didapat dari kartu cerdas. 3. Sistem menampilkan data transaksi berdasarkan UID termasuk gambar pada saat pengguna masuk 4. Petugas memverifikasi pengguna yang akan keluar dari data yang ada di dalam basis data dengan kondisi aslinya

	<p>(kondisi nyata saat itu)</p> <p>A1. Pengguna yang keluar tidak sesuai dengan data pada saat masuk</p> <p>A2. Data tidak ada di dalam basis data</p> <p>5. Data transaksi pengguna diperbarui dengan mengubah status menjadi Keluar</p> <p>6. Kasus penggunaan berakhir</p>
Aliran Kejadian Alternatif	<p>A1. Pengguna yang keluar tidak sesuai dengan data pada saat masuk</p> <ol style="list-style-type: none"> 1. Petugas memeriksa STNK Kendaraan 2. Berlanjut ke aliran kejadian normal nomor 6 <p>A2. Data tidak ada di dalam basis data</p> <ol style="list-style-type: none"> 1. Petugas memeriksa STNK Kendaraan 2. Berlanjut ke aliran kejadian normal nomor 6
Kondisi Akhir	Data transaksi dicatat dengan mengubah data pengguna saat masuk.
Kebutuhan Khusus	Alat pembaca NFC sudah terhubung dan berfungsi dengan baik



Gambar 3. 6 Diagram Aktivitas Kasus Penggunaan UC-003



Gambar 3. 7 Diagram Sekuensial Kasus Penggunaan UC-003

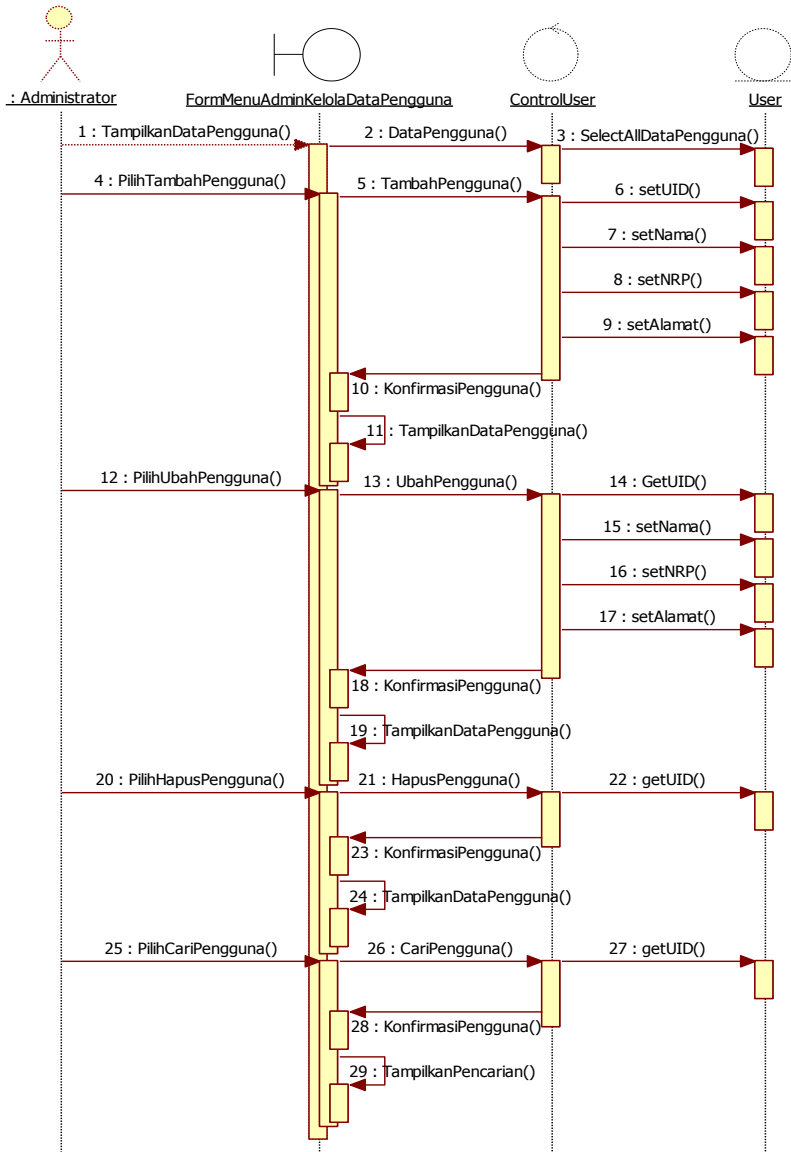
3.2.1.4 Kasus Penggunaan Mengelola Data Pengguna

Pada kasus penggunaan UC-004 Administrator dapat melakukan pengelolaan terhadap data pengguna yang telah tersimpan di dalam basis data. Administrator dapat melihat data, melakukan pencarian, penambahan, pengubahan, dan penghapusan data pengguna sesuai dengan kebutuhan. Rincian alur kasus penggunaan UC-004 dijelaskan pada Tabel 3.7. Diagram aktifitas kasus penggunaan UC-004 dijelaskan pada Gambar 3.8. Diagram sekuensial kasus penggunaan UC-004 dijelaskan pada Gambar 3.9.

Tabel 3. 7 Rincian Alur Kasus Penggunaan UC-004

Nama	Mengelola Data Pengguna
Kode	UC-004
Aktor	Administrator
Kondisi Awal	Administrator sudah masuk ke dalam sistem. Data pengguna belum terisi dan form isian belum terisi
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Data Pengguna ditampilkan secara keseluruhan 2. Administrator memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Administrator melakukan pencarian pengguna A2. Administrator memilih menambahkan data pengguna baru A3. Administrator memilih mengubah data pengguna A4. Administrator menghapus data pengguna 3. Sistem menampilkan kembali data pengguna setelah mengalami perubahan sesuai yang dilakukan oleh Administrator 4. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<ol style="list-style-type: none"> A1. Administrator melakukan pencarian pengguna <ol style="list-style-type: none"> 1. Administrator mengisi isian untuk memilih pengguna berdasarkan isian pencarian 2. Administrator menekan tombol cari 3. Sistem menampilkan hasil pencarian pengguna 4. Berlanjut ke aliran kejadian normal nomor 3 A2. Administrator memilih menambahkan data pengguna baru <ol style="list-style-type: none"> 1. Administrator mendekatkan kartu cerdas ke alat pembaca NFC 2. Alat mendapatkan UID kartu cerdas yang akan disimpan sebagai <i>Primary Key</i> Pengguna 3. Administrator menambahkan isian untuk pengguna 4. Administrator menekan tombol tambah

	<p>5. Sistem menyimpan data pengguna baru</p> <p>6. Berlanjut ke aliran kejadian normal nomor 3</p> <p>A3. Administrator memilih mengubah data pengguna</p> <ol style="list-style-type: none"> 1. Administrator memilih pengguna tertentu yang akan diubah 2. Administrator mengubah isian data yang baru dari data pengguna 3. Administrator menekan tombol ubah 4. Sistem mengubah data pengguna dengan data baru dan menyimpan data ke dalam basis data <p>5. Berlanjut ke aliran kejadian normal nomor 3</p> <p>A4. Adminsitrator memilih menghapus data pengguna</p> <ol style="list-style-type: none"> 1. Administrator memilih pengguna tertentu yang akan dihapus 2. Administrator menekan tombol hapus 3. Sistem menghapus data pengguna yang telah dipilih 4. Berlanjut ke aliran kejadian normal nomor 3
Kondisi Akhir	Informasi data pengguna tersimpan
Kebutuhan Khusus	Alat pembaca NFC telah terpasang dan berfungsi dengan baik



Gambar 3. 9 Diagram Sekuensial Kasus Penggunaan UC-004

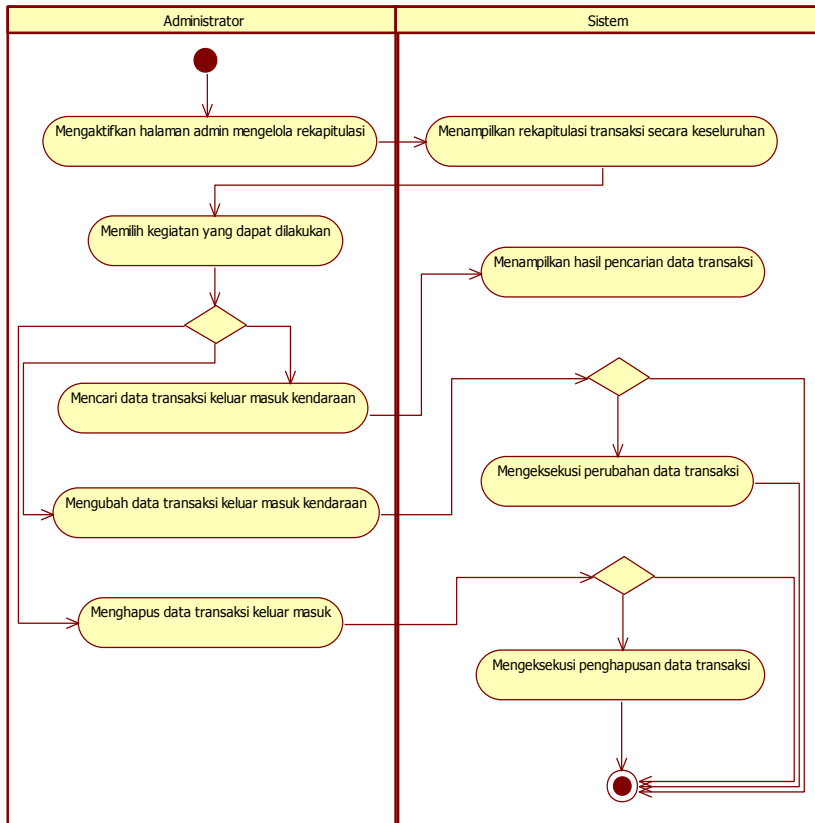
3.2.1.5 Kasus Penggunaan Mengelola Rekapitulasi Keluar Masuk Kendaraan

Pada kasus penggunaan UC-005, Administrator dapat melakukan pengelolaan terhadap data rekapitulasi keluar masuk kendaraan. Administrator dapat melihat data, mencari data, mengubah data, dan menghapus data. Disini tidak disediakan fungsi untuk menambahkan data dan data hanya bisa ditambahkan oleh Petugas Parkir yang ada di pos parkir secara langsung. Rincian alur kasus penggunaan UC-005 dijelaskan pada Tabel 3.8. Diagram aktifitas kasus penggunaan UC-005 dijelaskan pada Gambar 3.10. Diagram sekuensial kasus penggunaan UC-006 dijelaskan pada Gambar 3.11.

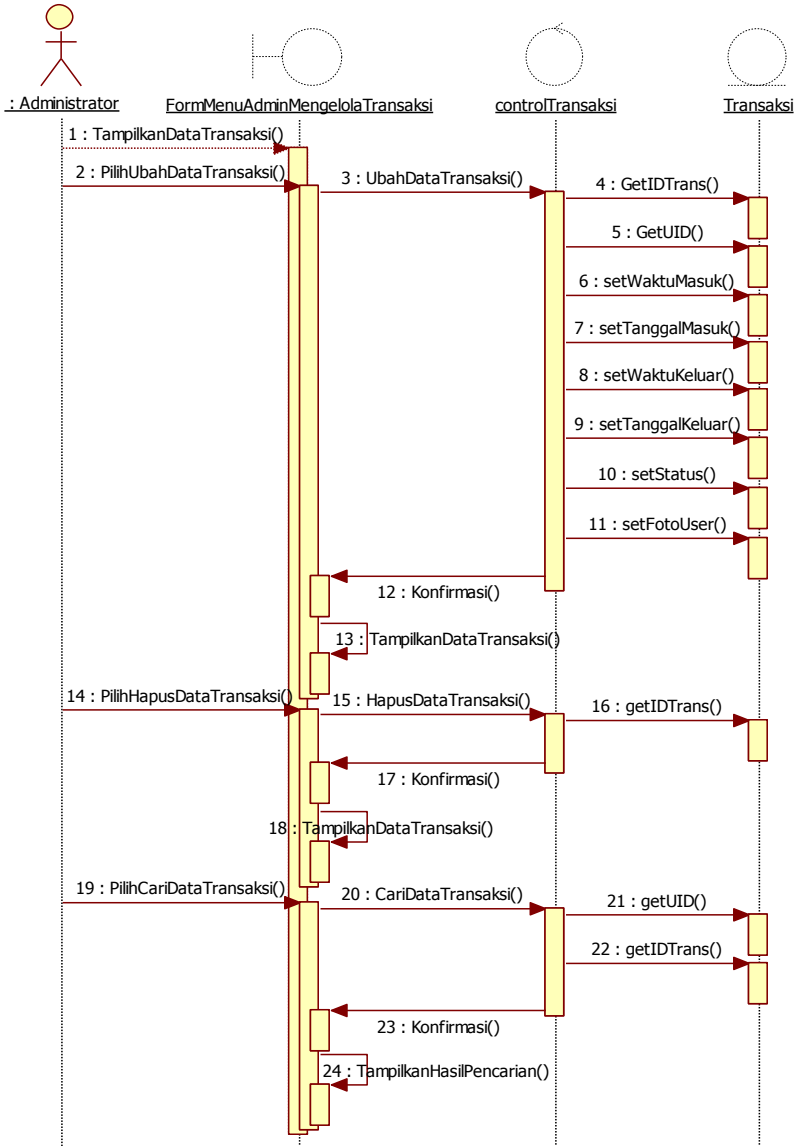
Tabel 3. 8 Rincian Alur Kasus Penggunaan UC-005

Nama	Mengelola Rekapitulasi Keluar Masuk Kendaraan
Kode	UC-005
Aktor	Administrator
Kondisi Awal	Administrator sudah masuk ke dalam sistem. Data transaksi keluar masuk kendaraan belum terisi dan form isian belum terisi
Aliran Kejadian Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan keseluruhan data transaksi keluar masuk kendaraan 2. Administrator memilih kegiatan yang dapat dilakukan <ol style="list-style-type: none"> A1. Administrator melakukan pencarian transaksi keluar masuk kendaraan A2. Administrator memilih mengubah data transaksi keluar masuk kendaraan A3. Administrator menghapus data transaksi keluar masuk kendaraan 3. Sistem menampilkan kembali data transaksi keluar masuk kendaraan setelah mengalami perubahan sesuai yang dilakukan oleh Administrator

	4. Kasus penggunaan berakhir
Aliran Kejadian Alternatif	<p>A1. Administrator melakukan pencarian transaksi keluar masuk kendaraan</p> <ol style="list-style-type: none"> 1. Administrator mengisi isian untuk memilih data transaksi keluar masuk kendaraan berdasarkan isian pencarian 2. Administrator menekan tombol cari 3. Sistem menampilkan hasil pencarian data transaksi keluar masuk kendaraan 4. Berlanjut ke aliran kejadian normal nomor 3 <p>A2. Administrator memilih mengubah data transaksi keluar masuk kendaraan</p> <ol style="list-style-type: none"> 1. Administrator memilih data transaksi keluar masuk kendaraan tertentu yang akan diubah 2. Administrator mengisi data transaksi keluar masuk kendaraan yang baru yang akan diubah 3. Administrator menekan tombol ubah 4. Sistem menyimpan data transaksi keluar masuk kendaraan yang baru ke dalam basis data 5. Berlanjut ke aliran kejadian normal nomor 3 <p>A3. Administrator menghapus data transaksi keluar masuk kendaraan</p> <ol style="list-style-type: none"> 1. Administrator memilih data transaksi keluar masuk kendaraan tertentu yang akan diubah 2. Administrator menekan tombol hapus 3. Sistem menghapus data transaksi keluar masuk kendaraan yang telah dipilih dari basis data 4. Berlanjut ke aliran kejadian normal nomor 3
Kondisi Akhir	Informasi data transaksi keluar masuk kendaraan tersimpan
Kebutuhan Khusus	Alat pembaca NFC telah terpasang dan berfungsi dengan baik



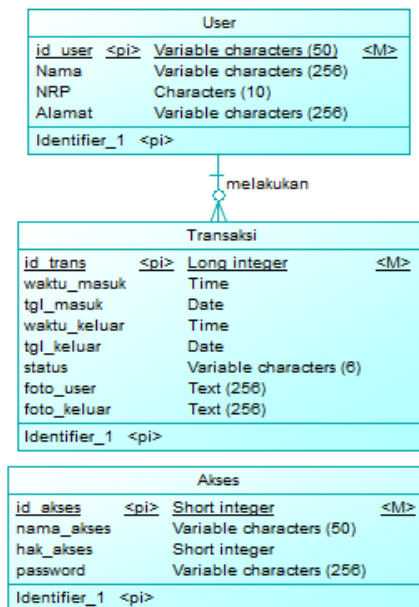
Gambar 3. 10 Diagram Aktivitas Kasus Penggunaan UC-005



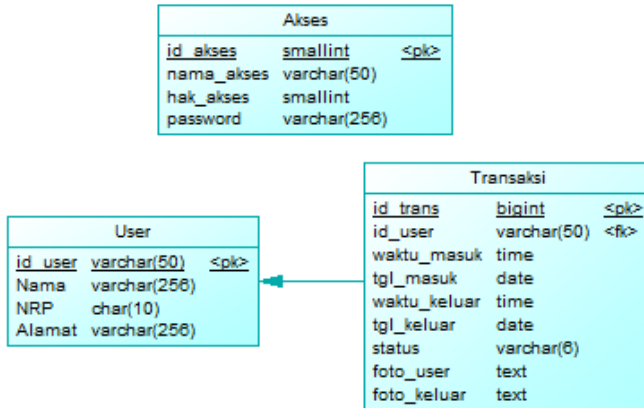
Gambar 3. 11 Diagram Sekuensial Kasus Penggunaan UC-005

3.2.2 Perancangan Basis Data

Pada subbab ini akan dijelaskan bagaimana rancangan basis data yang digunakan pada aplikasi Sistem Keamanan Kampus. Basis data pada sistem yang dibangun pada Tugas Akhir ini menggunakan sistem manajemen basis data relasional MySQL. MySQL digunakan untuk penyimpanan data pengguna, data transaksi dan data aturan peran pengguna (*user's role*). Perancangan basis data dimodelkan dengan diagram model konsep basis data atau yang biasa disebut *Conceptual Data Model* (CDM) dan diagram model fisik basis data atau *Physical Data Model* (PDM). Keduanya dapat dilihat pada Gambar 3.12 dan Gambar 3.13.



Gambar 3. 12 Diagram Model Konsep Basis Data (CDM) Aplikasi Sistem Keamanan Kampus



Gambar 3. 13 Diagram Model Fisik Basis Data (PDM) Aplikasi Sistem Keamanan Kampus

3.2.2.1 Rancangan Tabel Akses

Tabel Akses digunakan untuk menyimpan data pengguna (aktor) aplikasi yang digunakan sebagai akses untuk masuk ke dalam aplikasi. Detail atribut tabel akses dijelaskan pada Tabel 3.9.

Tabel 3. 9 Atribut Tabel Akses

Nama Kolom	Tipe Data	Keterangan
Id_akses	Smallint	<i>Primary Key</i> dari tabel akses
Nama_akses	Varchar (50)	Nama aktor
Hak_akses	Smallint	Hak akses untuk pengguna sesuai aturan peran pengguna.
Password	Varchar (256)	Kata kunci pengguna

3.2.2.2 Rancangan Tabel Pengguna (*User*)

Tabel Pengguna digunakan untuk menyimpan data pengguna yang nantinya akan digunakan agar dia bisa melakukan transaksi

keluar masuk parkir. Detail atribut tabel pengguna (*User*) dijelaskan pada Tabel 3.10.

Tabel 3. 10 Atribut Tabel Pengguna (*User*)

Nama Kolom	Tipe Data	Keterangan
Id_user	Varchar (50)	<i>Primary Key</i> dari tabel pengguna (<i>user</i>).
Nama	Varchar (256)	Nama pengguna
NRP	Char (10)	NRP pengguna
Alamat	Varchar (256)	Alamat pengguna

3.2.2.3 Rancangan Tabel Transaksi

Tabel Transaksi digunakan untuk menyimpan data transaksi yang dilakukan oleh pengguna (*user*) yang masuk ke tempat dan kemudian keluar dari tempat parkir. Tabel transaksi memiliki relasi ke tabel lainnya yaitu sebagai berikut:

1. Tabel Pengguna (*User*)

Hubungan dengan tabel ini adalah tabel transaksi menyimpan data transaksi yang dilakukan oleh pengguna, sedangkan tabel pengguna ini untuk menyimpan detail dari pengguna yang sedang melakukan transaksi. Karena satu orang pengguna dapat melakukan transaksi lebih dari satu kali. Detail atribut tabel transaksi dijelaskan pada Tabel 3.11.

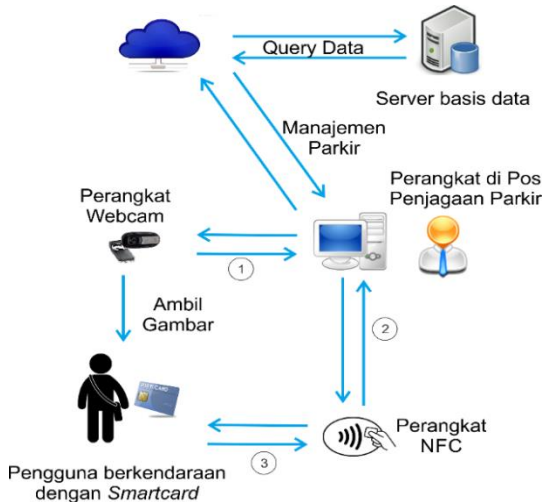
Tabel 3. 11 Atribut Tabel Transaksi

Nama Kolom	Tipe Data	Keterangan
Id_trans	Bigint	<i>Primary Key</i> dari tabel transaksi
Id_user	Varchar (50)	<i>Foreign Key</i> identitas pengguna yang dimiliki oleh pengguna
Waktu_masuk	Time	Waktu di saat pengguna masuk ke tempat parkir.
Tgl_masuk	Date	Tanggal ketika pengguna masuk

		ke tempat parkir
Waktu_keluar	Time	Waktu ketika pengguna keluar dari parkir
Tgl_keluar	Date	Tanggal ketika pengguna keluar dari parkir
Status	Varchar (6)	Status dari pengguna yang sedang atau telah melakukan transaksi. Status masuk, atau status keluar.
Foto_user	text	Tempat alamat foto transaksi disaat masuk parkir disimpan
Foto_keluar	text	Tempat alamat foto transaksi di saat pengguna keluar parkir disimpan

3.2.3 Perancangan Arsitektur

Pada subbab ini akan dibahas mengenai perancangan arsitektur aplikasi Sistem Keamanan Kampus ini. Arsitektur aplikasi digambarkan pada Gambar 3.14 sebagai berikut.



Gambar 3. 14 Arsitektur Sistem

Berikut penjelasan alur komunikasi antar elemen yang terlibat pada Gambar 3.14.

1. Perangkat yang ada di pos penjagaan parkir akan mengambil gambar kendaraan dan wajah dari pengendara yang kemudian disebut pengguna. Pengambilan gambar dilakukan dengan menggunakan webcam saat pengguna akan masuk ke dalam tempat parkir. Disaat hendak keluar, dilakukan pengambilan gambar dan pencocokan dengan data saat pengguna masuk.
2. Dilakukan pembacaan kartu cerdas oleh pembaca NFC. Data identitas yang dibaca kemudian disimpan ke dalam server basis data. Data alamat gambar dan waktu saat masuk juga kemudian disimpan ke dalam basis data.
3. Pengguna dengan menggunakan kendaraan bermotor yang akan masuk ke tempat parkir membawa kartu cerdas. Kartu cerdas yang dapat digunakan disini adalah Kartu tanda Mahasiswa atau Kartu Tanda Penduduk. Kemudian pengguna mendekatkan kartu tersebut ke alat pembaca NFC.

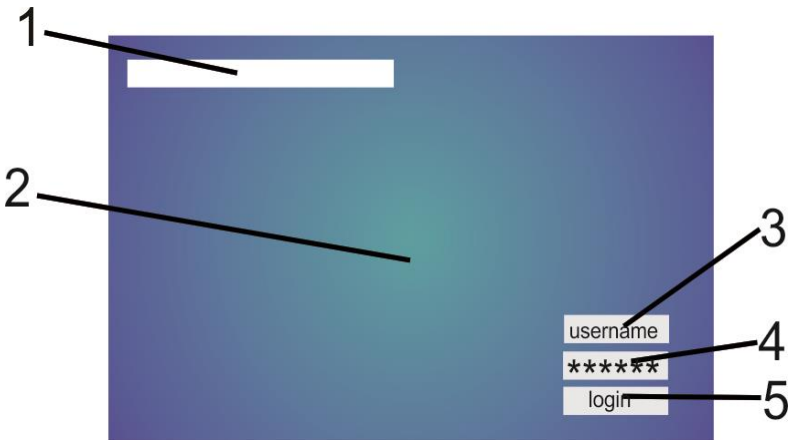
Proses identifikasi dan verifikasi ketika kendaraan tersebut masuk dan kemudian keluar dari tempat parkir yang benar adalah pengguna dan kendaraan yang masuk sama dengan pengguna yang akan keluar. Ketika terjadi kondisi tidak sama, maka Petugas Parkir akan secara manual memeriksa kelengkapan kendaraan yaitu Surat Tanda Nomor Kendaraan (STNK) yang sedang dibawa. Kemudian setelah memastikan bahwa kendaraan itu sesuai dengan STNK yang dia bawa, maka Petugas Parkir akan melakukan verifikasi terhadap pengguna itu bahwa pengguna benar dan dinyatakan telah keluar. Kemudian disimpan ke dalam basis data.

3.2.4 Perancangan Antarmuka Perangkat Lunak.

Pada subbab ini akan dibahas secara mendetail dari rancangan antarmuka yang berhubungan dengan pengguna aplikasi pada Tugas Akhir ini.

3.2.4.1 Perancangan Halaman Login

Halaman ini merupakan halaman ketika perangkat lunak pertama kali dijalankan. Aktor harus melakukan login terlebih dahulu ke dalam sistem. Login ini juga yang nantinya memberikan aturan peran aktor ke dalam sistem. Terdapat masukan berupa nama akun dan kata kunci dari pengguna. Dan terdapat satu tombol untuk melakukan login. Untuk melakukan login, pengguna harus terlebih dahulu mengisi nama akun dan kata kuncinya kemudian memilih tombol login untuk kemudian mengakses ke halaman berikutnya sesuai aturan peran. Gambar 3.15 merupakan rancangan antarmuka untuk halaman login perangkat lunak.



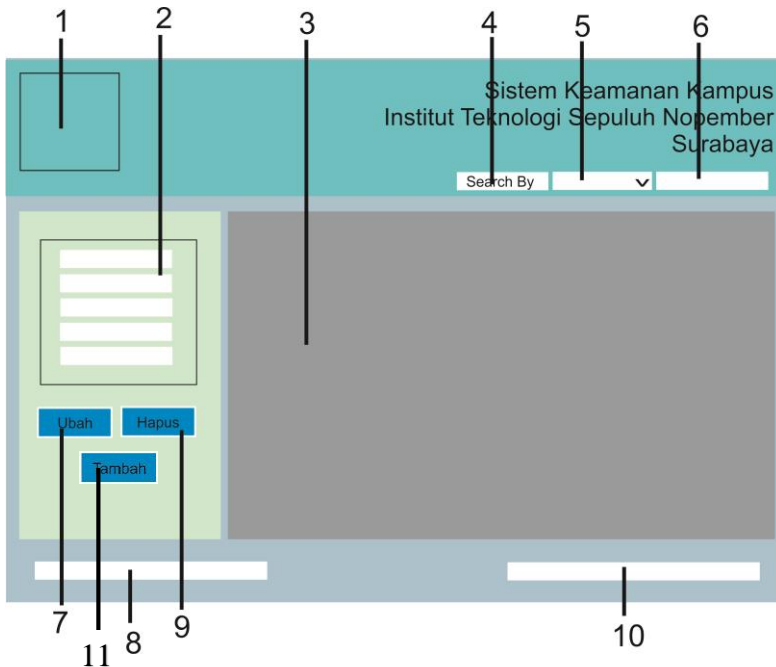
Gambar 3.15 Rancangan Halaman Login

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3.15.

1. Berupa teks yang menampilkan pesan dari hasil login yang dilakukan oleh pengguna.
2. Gambar latar belakang halaman.
3. Berupa *textbox* untuk menampung nama akun pengguna ketika akan melakukan login.
4. Berupa *textbox* untuk menampung kata kunci pengguna ketika akan melakukan login.
5. Tombol yang digunakan untuk memproses login yang dilakukan oleh pengguna

3.2.4.2 Perancangan Halaman Admin Mengelola Data Pengguna

Halaman ini merupakan halaman yang dapat diakses ketika login aktor Administrator dipilih. Kemudian admin memilih halaman Mengelola Data Pengguna. Pada halaman ini terdiri dari tampilan data tabel dari pengguna (*user*) yang dapat melakukan transaksi keluar masuk kendaraan. Beberapa isian dan tombol untuk melakukan penambahan, pengubahan atau penghapusan data pengguna. Dan untuk mencari data pengguna yang lebih spesifik.



Gambar 3. 16 Rancangan Halaman Admin Mengelola Data Pengguna

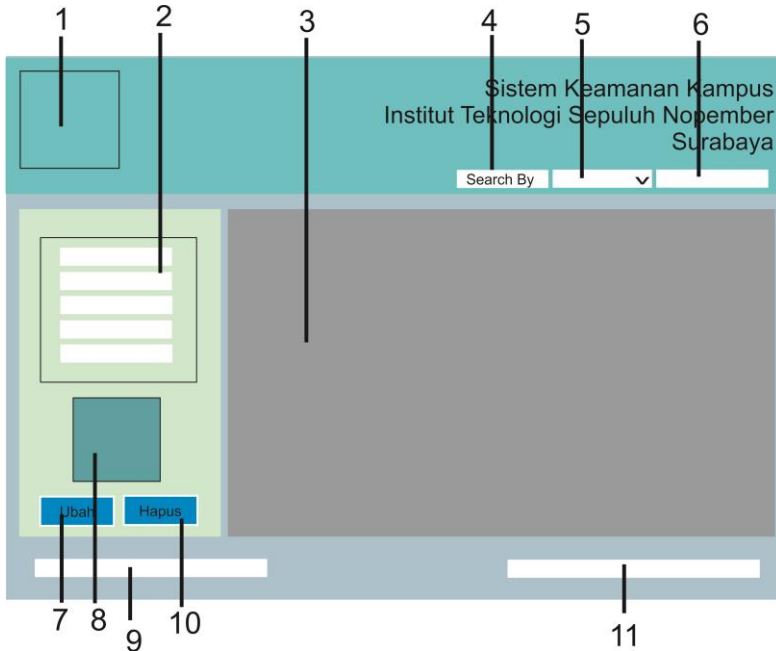
Gambar 3.16 merupakan gambaran rancangan antarmuka halaman mengelola data transaksi keluar masuk. Berikut masing-masing penjelasan nomor yang tertera pada Gambar 3.16.

1. Berupa *icon* untuk logo aplikasi Sistem Keamanan Kampus ITS.
2. Berupa beberapa *textbox* dan label teks serta *checkbox* untuk menampung data-data pengguna (*user*) seperti *id_user*, nama, *nrp* dan alamat yang nantinya dapat digunakan ketika hendak melakukan penambahan data, pengubahan data maupun penghapusan data.
3. Berupa tabel untuk menampilkan data pengguna

4. Tombol untuk melakukan pencarian yang dapat dilakukan oleh aktor ketika akan melihat data pengguna (*user*) dengan pilihan tertentu.
5. Berupa *combobox* yang menampung pilihan pencarian berdasarkan pilihan tertentu yang disimpan disini.
6. Berupa *textbox* untuk mengisi teks tertentu yang lebih spesifik untuk melakukan pencarian.
7. Berupa Tombol untuk melakukan pengubahan data pengguna tertentu yang telah dipilih.
8. Berupa teks yang menampilkan informasi ketersediaan alat pembaca NFC.
9. Berupa Tombol untuk melakukan penghapusan data pengguna tertentu yang telah dipilih.
10. Berupa teks untuk menampilkan waktu terkini ketika aplikasi dijalankan.
11. Berupa Tombol untuk melakukan penambahan data pengguna yang baru.

3.2.4.3 Perancangan Halaman Admin Mengelola Rekapitulasi Keluar Masuk Kendaraan

Halaman ini merupakan halaman yang dapat diakses ketika login Administrator dipilih. Kemudian administrator memilih halaman Mengelola Rekapitulasi Keluar Masuk Kendaraan. Pada halaman ini terdiri dari tampilan data tabel dari transaksi keluar masuk yang telah terjadi. Beberapa isian dan tombol untuk dapat melakukan pengubahan dan penghapusan data. Serta untuk mencari transaksi yang lebih spesifik.



Gambar 3. 17 Rancangan Halaman Admin Mengelola Rekapitulasi Keluar Masuk Kendaraan

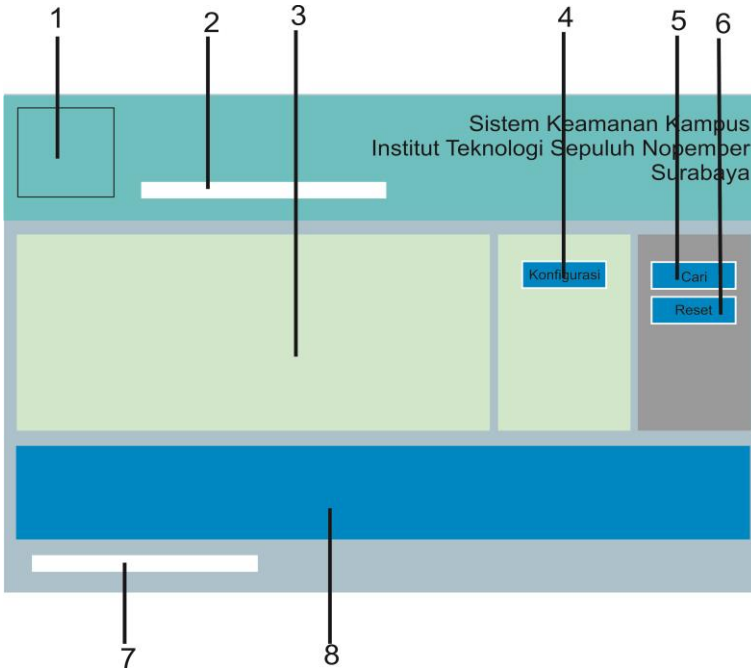
Gambar 3.17 merupakan gambaran rancangan antarmuka halaman mengelola data transaksi keluar masuk. Berikut masing-masing penjelasan nomor yang tertera pada Gambar 3.17.

1. Berupa *icon* untuk logo aplikasi Sistem Keamanan Kampus ITS.
2. Berupa beberapa *textbox* dan label teks serta *checkbox* untuk menampung data-data transaksi seperti *id_user*, *waktu_masuk*, *status*, *waktu keluar*, dll yang nantinya dapat digunakan ketika hendak melakukan perubahan data maupun penghapusan data.
3. Berupa tabel untuk menampilkan data transaksi keluar masuk kendaraan.

4. Tombol untuk melakukan pencarian yang dapat dilakukan oleh aktor ketika akan melihat transaksi dengan pilihan tertentu.
5. Berupa *combobox* yang menampung pilihan pencarian berdasarkan pilihan tertentu yang disimpan disini.
6. Berupa *textbox* untuk mengisi teks tertentu yang lebih spesifik untuk melakukan pencarian.
7. Berupa Tombol untuk melakukan pengubahan data transaksi tertentu yang telah dipilih.
8. Berupa gambar atau tempat gambar yang menampilkan foto pengguna saat masuk ke dalam parkir.
9. Berupa teks yang menampilkan informasi ketersediaan alat pembaca NFC.
10. Berupa Tombol untuk melakukan penghapusan data transaksi tertentu yang telah dipilih.
11. Berupa teks untuk menampilkan waktu terkini ketika aplikasi dijalankan.

3.2.4.4 Perancangan Halaman Utama

Halaman ini merupakan halaman utama yang dapat diakses ketika login Petugas Parkir dipilih. Halaman ini terdiri dari tampilan gambar yang diambil dari webcam yang terhubung dengan sistem, beberapa pilihan yang dapat dilakukan oleh pengguna, tabel transaksi keluar masuk kendaraan yang terjadi dan beberapa informasi yang diberikan oleh sistem kepada petugas, seperti waktu sekarang dan informasi ketersediaan alat pembaca NFC. Gambar 3.18 adalah tampilan halaman Utama perangkat lunak.



Gambar 3. 18 Rancangan Halaman Utama

Berikut penjelasan masing-masing nomor yang tertera pada Gambar 3.18.

1. Berupa *icon* untuk logo aplikasi Sistem Keamanan Kampus ITS.
2. Berupa teks untuk menampilkan waktu terkini ketika aplikasi dijalankan.
3. Tampilan gambar yang diambil dari webcam dan menampilkan gambar yang diambil dari basis data ketika ada pengguna yang keluar dari parkir.
4. Tombol untuk memilih konfigurasi webcam.

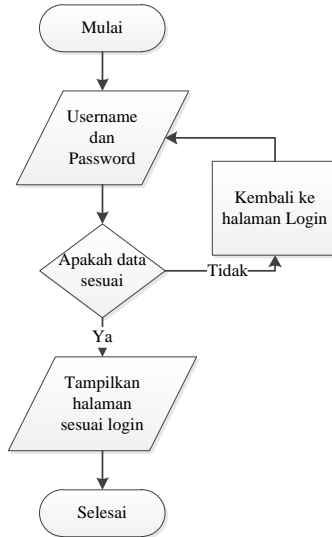
5. Tombol untuk melakukan pencarian yang dapat dilakukan oleh petugas ketika akan melihat transaksi dengan pilihan tertentu.
6. Tombol untuk memilih melakukan *reset* terhadap semua konfigurasi yang telah dipilih sebelumnya.
7. Berupa teks yang menampilkan informasi ketersediaan alat pembaca NFC.
8. Berupa tabel untuk menampilkan data transaksi keluar masuk kendaraan.

3.2.5 Perancangan Proses Perangkat Lunak

Pada subbab ini akan dijelaskan mengenai rancangan proses-proses yang terjadi untuk pencapaian suatu fungsionalitas pada program.

3.2.5.1 Proses Login

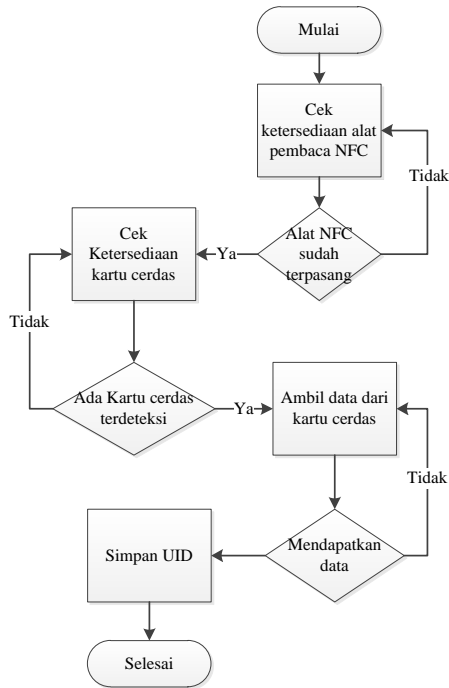
Proses login dilakukan oleh aktor aplikasi pada saat awal menjalankan aplikasi. Proses ini bertujuan untuk membedakan dan memberikan peran pengguna terhadap aplikasi. Sehingga untuk masing-masing aktor hanya akan dapat melakukan perintah sesuai dengan peran yang diberikan kepada aktor tersebut. Ketika aktor melaksanakan login ke dalam sistem, aktor harus memasukkan nama akun dan kata kunci. Kemudian sistem akan melakukan pencarian terhadap nama akun tersebut dan kemudian memberikan respon yaitu menampilkan halaman yang hanya diakses oleh akun tersebut. Ketika data tidak ditemukan, sistem menampilkan peringatan dan aktor tidak dapat melanjutkan ke halaman yang berikutnya. Atau dengan kata lain aktor tidak masuk ke dalam sistem. Berikut adalah diagram alir proses login yang dijelaskan pada Gambar 3.19.



Gambar 3. 19 Diagram Alir Proses Login

3.2.5.2 Proses Pendeteksian Kartu Cerdas

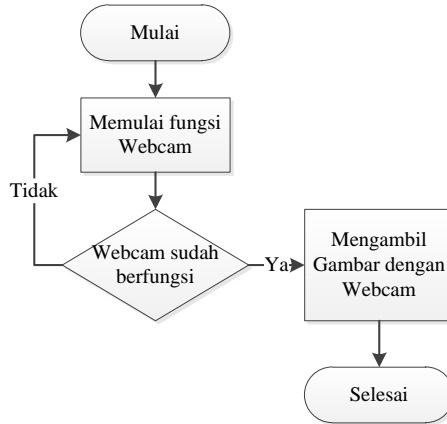
Proses pendeteksian kartu cerdas ini dilaksanakan ketika setiap pengguna yang akan masuk ke dalam tempat parkir. Proses ini dilakukan oleh alat pembaca NFC yang terhubung ke dalam sistem dan dapat berfungsi sebagai alat pembaca perangkat kartu cerdas yang dibawa oleh pengguna. Proses pendeteksian ini diawali dengan memastikan bahwa alat pembaca sudah terhubung dengan baik ke dalam sistem. Kemudian alat itu secara otomatis akan menunggu adanya kartu cerdas yang didekatkan menuju alat ini. Setiap ada kartu cerdas yang terdeteksi oleh alat pembaca ini, secara otomatis alat akan membaca data-data yang ada di dalam kartu cerdas ini. Kemudian menyimpan data tersebut untuk kemudian diolah untuk proses-proses yang lain. Misal pencatatan data masuk atau keluar. Berikut ada diagram alir proses pendeteksian kartu cerdas yang akan dijelaskan pada Gambar 3.20.



Gambar 3. 20 Diagram Alir Proses Pendeteksian kartu Cerdas

3.2.5.3 Proses Pengambilan Gambar dengan Webcam

Proses pengambilan gambar ini dilakukan ketika ada pengguna yang akan masuk ke dalam tempat parkir. Proses ini berlangsung secara otomatis ketika kartu cerdas telah terdeteksi oleh alat pembaca. Sebelumnya perangkat webcam ini harus sudah terhubung dengan sistem dan dapat berfungsi dengan baik. Setiap ada kartu cerdas yang terdeteksi untuk melakukan proses masuk ke dalam tempat parkir, secara otomatis webcam akan mengambil gambar pada saat itu, dan kemudian menyimpan ke dalam basis data. Berikut adalah diagram alir yang menjelaskan proses pengambilan gambar dengan webcam yang dijelaskan pada Gambar 3.21.



Gambar 3. 21 Diagram Alir Proses Pengambilan Gambar

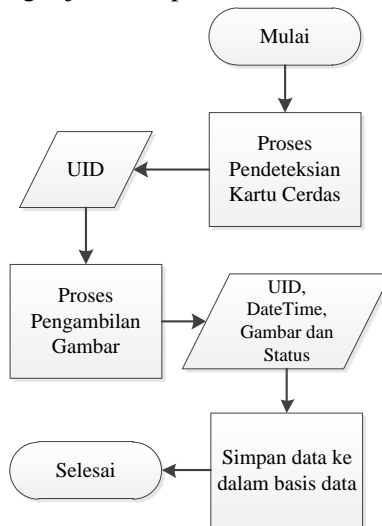
3.2.5.4 Proses Pencatatan Kendaraan Masuk Parkir

Proses pencatatan kendaraan masuk parkir bertujuan untuk mencatat setiap kendaraan yang masuk ke dalam tempat parkir ke dalam sistem menggunakan perangkat NFC dan kartu cerdas yang dimiliki oleh pengguna (*user*). Pencatatan dipengaruhi oleh ID yang terdapat pada kartu cerdas yang dibaca oleh alat pembaca NFC kemudian dijadikan pemicu untuk mendapatkan data yang lain yang dibutuhkan seperti gambar dan waktu pada saat itu. Secara umum, urutan proses pencatatan kendaraan masuk parkir adalah sebagai berikut.

1. Aplikasi Sistem Keamanan Kampus menampilkan halaman utama.
2. Perangkat webcam dan alat pembaca NFC telah terpasang dan terhubung ke dalam sistem dengan baik.
3. Pengguna yang masuk mendekati kartu cerdas ke alat pembaca.
4. Secara otomatis, alat pembaca akan membaca isi dari kartu cerdas tersebut dan menyimpannya ke dalam basis data

5. Secara bersamaan webcam juga mengambil gambar pada saat itu untuk disimpan ke dalam basis data.
6. Sistem juga mengambil waktu kejadian pada saat itu dan kemudian disimpan juga ke dalam basis data.

Berikut adalah diagram alir proses pencatatan kendaraan masuk parkir yang dijelaskan pada Gambar 3.22.

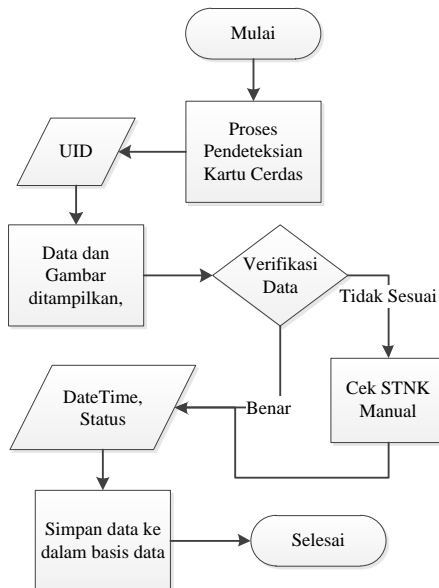


Gambar 3. 22 Diagram Alir Proses Pencatatan Kendaraan Masuk Parkir

3.2.5.5 Proses Verifikasi Kendaraan yang Keluar Parkir

Proses verifikasi dilakukan setiap pengguna yang telah masuk ke dalam tempat parkir akan keluar dari tempat parkir. Pengguna mendekati kartu cerdas yang dimilikinya menuju alat pembaca NFC. Secara otomatis alat pembaca akan membaca data-data yang ada di dalam kartu cerdas untuk kemudian di proses oleh sistem untuk melakukan pencarian data transaksi masuk tersebut. Setelah itu sistem menampilkan data hasil pencarian.

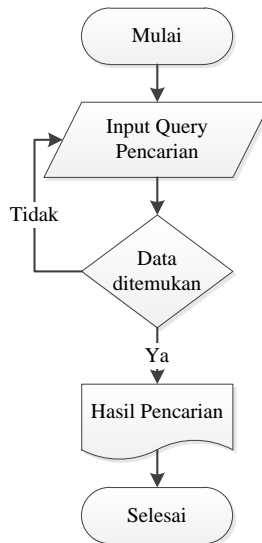
Petugas parkir kemudian melakukan verifikasi terhadap data tersebut dengan kondisi nyata, apakah data yang disimpan ketika pengguna masuk sama dengan kondisi ketika pengguna akan keluar. Ketika data sama, maka petugas memastikan verifikasi pengguna dengan benar dan kemudian sistem mengubah status transaksi tersebut menjadi telah keluar. Ketika data tidak sama, maka petugas secara manual akan memeriksa kelengkapan kendaraan yang digunakan oleh pengguna, yaitu dengan memeriksa Surat Tanda Nomor Kendaraan (STNK). Apabila dinyatakan benar, maka pengguna dapat keluar dari tempat parkir. Berikut adalah diagram alir dari proses verifikasi kendaraan yang keluar parkir dijelaskan pada Gambar 3.23.



Gambar 3. 23 Diagram Alir Proses verifikasi kendaraan yang akan Keluar

3.2.5.6 Proses Mencari Data Pengguna

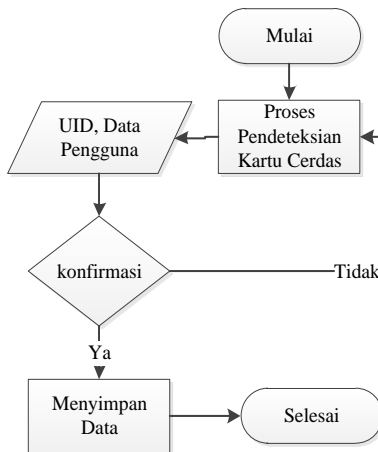
Proses mencari data pengguna merupakan proses yang dilakukan Administrator atau Petugas Parkir untuk mencari pengguna tertentu berdasarkan kategori tertentu. Data pengguna didapatkan dari basis data. Data yang telah ditemukan akan ditampilkan di antarmuka aplikasi. Pencarian ini dilakukan ketika aktor ingin mencari pengguna tertentu dengan mencari berdasarkan kategori tertentu yang dibutuhkan. Bisa juga dilakukan pencarian berdasarkan UID pengguna dengan cara mendekatkan kartu cerdas pengguna ke alat pembaca NFC. Secara otomatis nantinya sistem akan menampilkan pengguna berdasarkan UID yang terdeteksi dari kartu cerdas. Berikut adalah diagram alir dari proses mencari data pengguna dijelaskan pada Gambar 3.24.



Gambar 3. 24 Diagram Alir Prose Pencarian Data Pengguna

3.2.5.7 Proses Menambah Data Pengguna

Proses menambah data pengguna merupakan proses yang dilakukan ketika data pengguna belum terdaftar di dalam sistem. Proses ini dilakukan oleh Administrator dengan mengisikan informasi-informasi pengguna, kemudian informasi itu disimpan di dalam basis data. Untuk menambahkan data, diperlukan sebuah kartu cerdas yang dimiliki oleh pengguna yang kemudian di dekatkan ke alat pembaca NFC. Kemudian alat pembaca NFC tersebut mendapatkan UID dari kartu cerdas yang nantinya UID ini akan dijadikan *primary key* dari data pengguna yang akan ditambahkan. Gambar 3.25 merupakan penjelasan dari diagram alir proses menambahkan data pengguna.

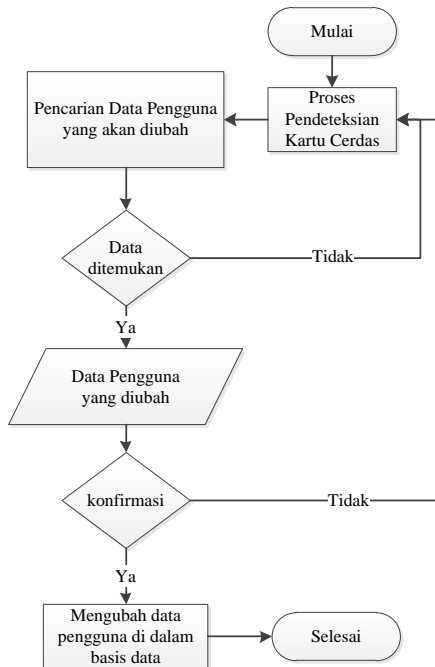


Gambar 3. 25 Diagram Alir Proses Menambakan Data Pengguna

3.2.5.8 Proses Mengubah Data Pengguna

Proses mengubah data pengguna merupakan proses yang dilakukan oleh Administrator ketika dibutuhkan sebuah perubahan terhadap data pengguna. Data pengguna yang didapat dari basis data diubah melalui aplikasi dan kemudian Administrator mengisikan atau memilih perubahan-perubahan apa

yang dibutuhkan. Data yang telah diubah kemudian disimpan kembali di dalam basis data. Perubahan pengguna bisa dilakukan dengan mencari berdasarkan kategori tertentu atau melalui UID yang didapatkan dari kartu cerdas yang dibaca oleh alat pembaca NFC. Secara umum proses diagram alir proses mengubah data pengguna dijelaskan pada Gambar 3.26.

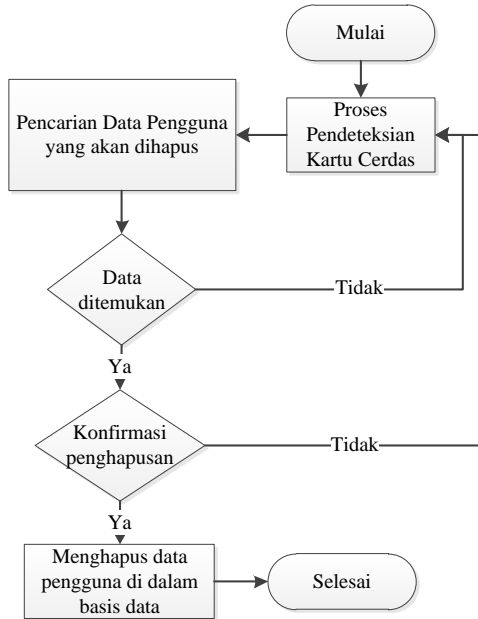


Gambar 3. 26 Proses Perubahan Data Pengguna

3.2.5.9 Proses Menghapus Data Pengguna

Proses menghapus data pengguna merupakan proses yang dilakukan ketika data pengguna akan dihapus. Data pengguna yang telah didapatkan dari basis data dihapus sesuai dengan perintah yang dilakukan dari aplikasi. Administrator terlebih dahulu memilih pengguna mana yang akan dihapus, kemudian

Administrator memilih menghapus data. Secara umum, proses menghapus data pengguna dijelaskan pada diagram alir pada Gambar 3.27 berikut.



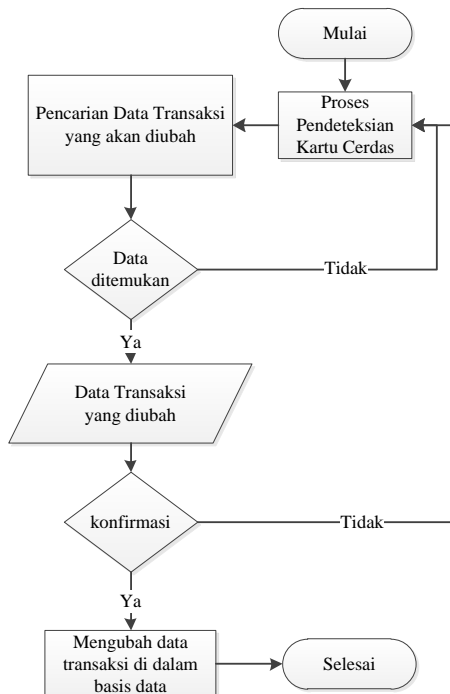
Gambar 3. 27 Diagram Alir Proses Menghapus Data Pengguna

3.2.5.10 Proses Mencari Data Transaksi Keluar Masuk Kendaraan

Proses mencari data transaksi keluar masuk kendaraan dapat dilakukan ketika Administrator atau Petugas Parkir ingin mengetahui informasi spesifik dari sebuah transaksi berdasarkan kategori tertentu. Data transaksi yang didapat kemudian ditampilkan ke halaman antarmuka aplikasi. Diagram alir proses mencari data transaksi keluar masuk kendaraan sama dengan pencarian data pengguna dan telah dijelaskan sebelumnya pada Gambar 3.24.

3.2.5.11 Proses Mengubah Data Transaksi Keluar Masuk Kendaraan

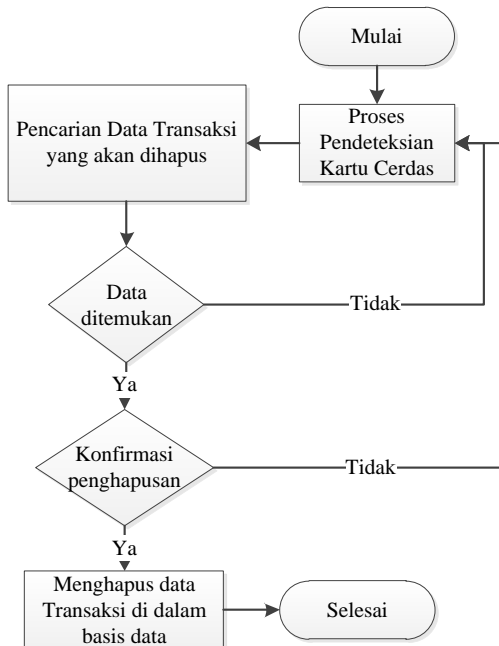
Proses mengubah data transaksi keluar masuk kendaraan merupakan proses yang dilakukan oleh Administrator atau Petugas Parkir ketika dibutuhkan sebuah perubahan terhadap data pengguna. Data transaksi yang didapat dari basis data diubah melalui aplikasi dan kemudian mengisikan atau memilih perubahan-perubahan apa yang dibutuhkan. Data yang telah diubah kemudian disimpan kembali di dalam basis data. Diagram alir proses mengubah data transaksi keluar masuk kendaraan dijelaskan pada Gambar 3.28.



Gambar 3. 28 Diagram Alir Mengubah Data Transaksi

3.2.5.12 Proses Menghapus Data Transaksi Keluar Masuk Kendaraan

Proses menghapus data transaksi merupakan proses yang dilakukan oleh Administrator ketika data transaksi akan dihapus. Data transaksi yang telah didapatkan dari basis data dihapus sesuai dengan perintah yang dilakukan dari aplikasi. Administrator terlebih dahulu memilih transaksi mana yang akan dihapus, kemudian Administrator memilih menghapus data. Secara umum, proses menghapus data transaksi keluar masuk kendaraan dijelaskan pada diagram alir pada Gambar 3.29 berikut.



Gambar 3. 29 Diagram Alir Proses Menghapus Data Transaksi

BAB IV IMPLEMENTASI

Pada bab ini akan dibahas mengenai implementasi dari rancangan perangkat lunak. Implementasi yang dijelaskan meliputi lingkungan pembangunan perangkat lunak, implementasi antarmuka pengguna, dan implementasi proses-proses yang terjadi pada masing-masing kasus penggunaan pada perangkat lunak. Implementasi sistem mengacu pada analisis dan perancangan yang ditulis pada Bab 3. Namun, tidak menutup kemungkinan adanya perubahan-perubahan dari rancangan tersebut apabila memang diperlukan.

4.1 Lingkungan Implementasi

Dalam merancang perangkat lunak ini digunakan beberapa perangkat pendukung sebagai berikut.

4.1.1 Lingkungan Implementasi Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan sistem adalah komputer dengan spesifikasi *notebook* Compaq Presario CQ-42 258 VX, Intel® Core™ i5 M450 @2,4 Ghz dan 4GB memory.

4.1.2 Lingkungan Implementasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan untuk pengembangan sistem adalah sebagai berikut:

- Microsoft Windows 8.1 sebagai sistem operasi.
- Microsoft Visual Studio Ultimate 2012 dan Microsoft .NET Framework 4.5 untuk implementasi aplikasi
- MySQL seri 5 sebagai aplikasi manajemen basis data.
- Power Designer 16.0 untuk merancang basis data.

- StarUML versi 5.0.2.1570 dan Microsoft Visio 2010 untuk merancang diagram perencanaan perangkat lunak
- Kakas Bantu SDK ACR 122U untuk menghubungkan alat pembaca ACR122U dengan project aplikasi
- Corel Draw X6 untuk merancang desain aplikasi

4.2 Lingkungan Implementasi Antarmuka

Pada subbab ini akan dibahas implementasi antarmuka berdasarkan rancangan antarmuka yang telah dibahas pada Bab 3. Antarmuka yang akan dibahas adalah antarmuka aplikasi berbasis *desktop*.

4.2.1 Implementasi Antarmuka Halaman Login

Antarmuka halaman login ditunjukkan pada Gambar 4.1. Halaman diakses pada saat aplikasi pertama kali dijalankan. Pengguna yang login nantinya akan diberikan tampilan atau akses yang sesuai dengan peran penggunaannya. Berikut implementasi dari halaman login aplikasi Sistem Keamanan Kampus.



Gambar 4. 1 Implementasi Halaman Login

Pada Gambar 4.1 pengguna diminta untuk memasukkan nama akun dan kata kunci dan kemudian pengguna menekan tombol login. Apabila nama akun dan kata kunci yang dimasukkan tidak sesuai, maka sistem akan menampilkan pesan peringatan, dan pengguna tidak dapat melanjutkan ke halaman selanjutnya.

4.2.2 Implementasi Antarmuka Halaman Admin Mengelola Data Pengguna

Antarmuka halaman admin mengelola data pengguna merupakan halaman yang dapat diakses dengan peran pengguna sebagai Administrator setelah proses login pada halaman login. Halaman berikut ini merupakan implementasi dari antarmuka halaman admin mengelola data pengguna yang ditunjukkan pada Gambar 4.2.

Menu Administrator Sistem Keamanan Kampus

Data Keluar Masuk: Data Pengguna

System Keamanan Kampus
Institut Teknologi Sepuluh Nopember
Surabaya

Search by:

ID Pengguna:

Nama: Orysa Nur Permata Sari

NRP: 51101001

Alamat: Sidoarjo

Ubah Hapus

Tambah

ID_USER	NAMA	NRP	ALAMAT
0411520A932980	Orysa Nur Permata Sari	51101001	Sidoarjo
BCTCFBA	Hani Ramadhan	5110100043	Perumahan Makarya Binangun, Waru
969200F9	Uji Coba	5110100000	Lab RPL
9C62009B	Fadika Dita Nurjanto	5110100192	Keputih Gg 2 No 14, Surabaya
6C35FD9A	Fahmi Hidayatullah	5110100199	Jalan Arif Rahman Hakim 2B, Keputih Surabaya
9252E43D	Eka Miftahul	5110100174	Gg H.Davel
AC45F89A	Ainatul Maulida	5110100	Bangkalan
041E7DBA862680	Pengguna 041E7DBA862680		
0422385A272980	Pengguna 0422385A272980		
04306D1A742880	Pengguna 04306D1A742880		
048A3602302A80	Pengguna 048A3602302A80		
06132556	Pengguna 06132556		
3CA1FC9A	Pengguna 3CA1FC9A		
6C17FC9A	Pengguna 6C17FC9A		
76AEE83A	Pengguna 76AEE83A		
9C84FC9A	Pengguna 9C84FC9A		
9C94FD9A	Pengguna 9C94FD9A		
AC10FC9A	Pengguna AC10FC9A		
CC8BF99A	Pengguna CC8BF99A		
D2CC6A80	Pengguna D2CC6A80		
D6142656	Pengguna D6142656		

Reader Belum Terbung

Tuesday, 24 June, 2014 jam 11:02:12 AM

Gambar 4. 2 Implementasi Halaman Admin Mengelola Data Pengguna

Pada halaman Admin mengelola data pengguna, terdapat beberapa tombol untuk melakukan pengelolaan data pengguna baik itu tambah, ubah, dan hapus. Selain itu juga diimplementasikan tombol untuk melakukan pencarian berdasarkan kategori tertentu.

4.2.3 Implementasi Antarmuka Halaman Admin

Mengelola Rekapitulasi Keluar Masuk Kendaraan

Antarmuka halaman admin mengelola rekapitulasi keluar masuk kendaraan merupakan halaman yang dapat diakses dengan peran pengguna sebagai Administrator setelah proses login pada halaman login. Berikut adalah implementasi antarmuka halaman admin mengelola rekapitulasi keluar masuk kendaraan ditunjukkan pada Gambar 4.3.

The screenshot shows the 'Menu Administrator Sistem Keamanan Kampus' interface. The main content area displays a table of vehicle entry and exit records. The table has columns for ID_TRANS, ID_USER, WAKTU_MASUK, TGL_MASUK, WAKTU_KELUAR, TGL_KELUAR, and STAT. The data is as follows:

ID_TRANS	ID_USER	WAKTU_MASUK	TGL_MASUK	WAKTU_KELUAR	TGL_KELUAR	STAT
20	BC7CF9A	05:53:25	22-Jun-14			Masuk
19	6C35FD9A	05:16:21	22-Jun-14			Keluar
18	9C64FC9A	05:09:30	22-Jun-14			Masuk
17	04300D1A742...	05:00:26	22-Jun-14			Masuk
16	963900F9	12:48:43	22-Jun-14	12:48:54	22-Jun-14	Keluar
15	5CA1FC9A	04:30:09	21-Jun-14	04:31:13	21-Jun-14	Keluar
14	9C94FD9A	03:54:07	21-Jun-14	03:54:53	21-Jun-14	Keluar
13	0422385A272...	03:49:05	21-Jun-14			Masuk
12	CC88FF9A	02:55:59	21-Jun-14	03:49:30	21-Jun-14	Keluar
11	AC10FC9A	02:39:23	21-Jun-14			Masuk
10	D6142636	02:10:05	21-Jun-14			Masuk
9	048A36D2302...	02:06:58	21-Jun-14			Masuk
8	ECEEF9A	01:55:28	21-Jun-14			Masuk
7	06132556	01:49:15	21-Jun-14	01:55:00	21-Jun-14	Keluar
6	E6C72556	01:45:19	21-Jun-14	01:46:27	21-Jun-14	Keluar
5	D6202556	01:40:39	21-Jun-14	01:40:40	21-Jun-14	Keluar
4	D6202556	01:40:17	21-Jun-14	01:40:37	21-Jun-14	Keluar
2	ACA5FB9A	01:00:36	21-Jun-14			Masuk
1	BC7CF9A	12:58:55	21-Jun-14	12:59:24	21-Jun-14	Keluar

The interface also includes a sidebar with filters for User ID (BC7CF9A), Waktu Masuk (05:53:25), Tanggal Masuk (Sunday, 22 June, 20), Waktu Keluar, and Tanggal Keluar (Tuesday, 24 June, 20). There are checkboxes for 'Masuk' and 'Keluar' status. A camera feed is visible below the filters. At the bottom, there are 'Ubah' and 'Hapus' buttons. The footer shows 'Tertubung dengan alat pembaca ACS ACR122 0' and the date 'Tuesday, 24 June, 2014 jam 12:41:00 PM'.

Gambar 4. 3 Implementasi Halaman Admin Mengelola Rekapitulasi Keluar Masuk Kendaraan

Halaman admin mengelola rekapitulasi keluar masuk kendaraan bertujuan untuk pengelolaan terhadap data rekapitulasi keluar masuk kendaraan ini. Sehingga, nantinya Administrator dapat melihat data, mencari data berdasarkan kategori tertentu, mengubah, dan menghapus data rekapitulasi keluar masuk kendaraan. Terdapat beberapa tombol dan isian yang dapat digunakan oleh Administrator untuk memenuhi tujuan yang diinginkan oleh Administrator.

4.2.4 Implementasi Antarmuka Halaman Utama

Antarmuka halaman utama merupakan halaman utama yang diakses oleh Petugas Parkir setelah proses login dilakukan pada halaman login. Berikut ini merupakan implementasi dari antarmuka halaman utama yang ditunjukkan pada Gambar 4.4

ID Transaksi	NAMA	NRP	Waktu Masuk	Tanggal Masuk	Waktu Keluar	Tanggal Keluar	STATUS	Foto User
8	Tirta Taruna R...	5110100701	01:55:28	21-Jun-14			Masuk	C:\Users\Fah...
9	Pengguna 04...		02:06:58	21-Jun-14			Masuk	C:\Users\Fah...
10	Wiby Mahan ...	5111100025	02:10:05	21-Jun-14			Masuk	C:\Users\Fah...
11	Hardika Khus...	5110100206	02:39:23	21-Jun-14			Masuk	C:\Users\Fah...
17	Antonia Cahy...	5110100187	05:00:26	22-Jun-14			Masuk	C:\Users\Fah...
18	Astris Dyah P...	5110100178	05:09:30	22-Jun-14			Masuk	C:\Users\Fah...

Gambar 4. 4 Implementasi Halaman Utama

Pada halaman utama, terdapat beberapa tombol untuk melakukan konfigurasi webcam, pencarian terhadap data transaksi, tampilan untuk gambar yang diambil ketika pengguna masuk, dan tampilan gambar ketika pengguna keluar. Serta tabel untuk menampilkan data transaksi.

4.3. Implementasi Basis Data

Pada subbab ini akan dibahas implementasi dari rancangan basis data yang telah dibahas pada Bab 3. Terdapat dua implementasi pada subbab ini yakni implementasi struktur basis data dan implementasi *query* yang digunakan aplikasi Sistem Keamanan Kampus.

4.3.1 Implementasi Struktur Basis Data

Implementasi struktur basis data adalah implementasi sintaks yang digunakan untuk membuat tabel-tabel di dalam basis data yang dibutuhkan pada aplikasi Sistem Keamanan Kampus. Implementasi basis data menggunakan sintaks SQL dan dibedakan menjadi beberapa tabel, yaitu tabel Akses, tabel Pengguna (*User*), dan tabel Transaksi. Berikut penjelasan masing-masing implementasi dari ketiga tabel tersebut.

4.3.1.1 Implementasi Tabel Akses

Tabel Akses merupakan tabel yang digunakan untuk menyimpan data akses pengguna. Menyimpan aturan peran pengguna (*user's role*) ke dalam sistem. Implementasi tabel Akses ditunjukkan pada Kode Sumber 4.1.

```
CREATE TABLE IF NOT EXISTS `akses` (
  `ID_AKSES` smallint(6) NOT NULL,
  `NAMA_AKSES` varchar(50) DEFAULT NULL,
  `HAK_AKSES` smallint(6) DEFAULT NULL,
  `PASSWORD` varchar(256) DEFAULT NULL,
```

```

PRIMARY KEY (`ID_AKSES`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4. 1 Implementasi Struktur Tabel Akses

4.3.1.2 Implementasi Tabel Pengguna (*User*)

Tabel Pengguna (*User*) digunakan untuk menyimpan data pengguna yang nantinya akan digunakan agar dia bisa melakukan transaksi keluar masuk parkir. Dimana tabel ini akan menjadi tabel master untuk tabel Transaksi. Implementasi tabel Pengguna (*User*) ditunjukkan pada Kode Sumber 4.2.

```

CREATE TABLE IF NOT EXISTS `user` (
  `ID_USER` varchar(50) NOT NULL,
  `NAMA` varchar(256) DEFAULT NULL,
  `NRP` char(10) DEFAULT NULL,
  `ALAMAT` varchar(256) DEFAULT NULL,
  PRIMARY KEY (`ID_USER`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

Kode Sumber 4. 2 Implementasi Struktur Tabel Pengguna

4.3.1.3 Implementasi Tabel Transaksi

Tabel Transaksi digunakan untuk menyimpan data transaksi yang dilakukan oleh pengguna (*user*) yang masuk ke tempat dan kemudian keluar dari tempat parkir. Tabel transaksi memiliki relasi ke tabel lainnya yaitu tabel Pengguna (*User*). Karena memungkinkan bahwa satu pengguna dapat melakukan transaksi keluar masuk lebih dari satu kali. Implementasi tabel Transaksi ditunjukkan pada Kode Sumber 4.3.

```

CREATE TABLE IF NOT EXISTS `transaksi` (
  `ID_TRANS` bigint(20) NOT NULL AUTO_INCREMENT,
  `ID_USER` varchar(50) NOT NULL,
  `WAKTU_MASUK` time DEFAULT NULL,
  `TGL_MASUK` date DEFAULT NULL,
  `WAKTU_KELUAR` time DEFAULT NULL,
  `TGL_KELUAR` date DEFAULT NULL,
  `STATUS` varchar(6) DEFAULT '0',

```



```

`FOTO_USER` text,
`FOTO_KELUAR` text,
PRIMARY KEY (`ID_TRANS`),
KEY `FK_MELAKUKAN` (`ID_USER`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1
AUTO_INCREMENT=1;

```

Kode Sumber 4.3 Implementasi Struktur Tabel Transaksi

4.3.2 Implementasi *Query*

Query yang digunakan pada aplikasi ini adalah MySQL. Implementasi dari beberapa proses *query* akan dibahas dalam Subbab berikut ini.

4.3.2.1 Implementasi *Query Login*

Query login merupakan *query* yang dieksekusi ketika aktor menjalani proses otentikasi sebelum masuk ke dalam sistem. *Query* ini diimplementasikan dan ditunjukkan pada Kode Sumber 4.4.

```

public void GetPegguna(int _id_akses)
{
    UpdateClass();
    try
    {
        MySqlConnection mycon = new
        MySqlConnection(connectionString);
        mycon.Open();
        MySqlCommand mycommand = new MySqlCommand();
        mycommand.Connection = mycon;
        mycommand.CommandText = "select * from akses where
        id_akses = @_id_akses";
        mycommand.Prepare();
        mycommand.Parameters.AddWithValue("@_id_akses",
        _id_akses);
        MySqlDataReader rdr = mycommand.ExecuteReader();
        while (rdr.Read()){
            . . .
            break;
        }
    }
}

```

```

    }
    }
    catch (Exception e)
    { . . . } }

```

Kode Sumber 4. 4 Implementasi *Query* Login

4.3.2.2 Implementasi *Query* Mencari Data Pengguna

Query mencari data pengguna dilakukan untuk mendapatkan data pengguna tertentu dari basis data. *Query* ini diimplementasikan pada *controller*. Pencarian dapat berupa mencari ketersediaan data pengguna di dalam basis data maupun untuk mendapatkan data pengguna. Implementasi *query* mencari data pengguna ditunjukkan pada Kode Sumber 4.5.

```

public int cekUser(string _UID) {
    int _count = 0;
    MySqlConnection mycon = new
MySqlConnection(connectionString);
    mycon.Open();
    MySqlCommand mycommand = new MySqlCommand();
    mycommand.Connection = mycon;
    mycommand.CommandText = "select count(*) AS
count_transaksi from user where id_user = @_id";
    mycommand.Prepare();
    mycommand.Parameters.AddWithValue("@_id",
_UID);
    MySqlDataReader rdr =
mycommand.ExecuteReader();
    while (rdr.Read())
    { . . . }
    mycon.Close();
    return _count;
}

public void GetUser(string _UID) {
    UpdateClass();
    try{
        MySqlConnection mycon = new

```

```

MySQLConnection(connectionString);
mycon.Open();
MySQLCommand mycommand = new MySQLCommand();
mycommand.Connection = mycon;
mycommand.CommandText = "select * from user
where id_user = @a";
mycommand.Prepare();
mycommand.Parameters.AddWithValue("@a",
_UID);
MySQLDataReader rdr =
mycommand.ExecuteReader();
while (rdr.Read())
{ . . . }
mycon.Close();
}
catch (Exception e)
{ . . . }

```

Kode Sumber 4. 5 Implementasi *Query* Mencari Data Pengguna

4.3.2.3 Implementasi *Query* Menambahkan Data Pengguna

Query menambahkan data pengguna ini dilakukan ketika aktor ingin menambahkan data pengguna ke dalam basis data. *Query* ini diimplementasikan di dalam kelas *controller*. Kode Sumber 4.6 adalah implementasi dari *query* menambahkan data pengguna.

```

public void InsertUser() {
try{
MySQLConnection mycon = new
MySQLConnection(connectionString);
mycon.Open();
MySQLCommand mycommand = new MySQLCommand();
mycommand.Connection = mycon;
mycommand.CommandText = "insert into user
(id_user, nama, nrp, alamat) values(@a, @b,
@c, @d)";
mycommand.Prepare();
mycommand.Parameters.AddWithValue("@a",
User.IDUser);

```

```

. . .
int x = mycommand.ExecuteNonQuery();
mycon.Close();
}
catch (Exception e)
{ . . . }

```

Kode Sumber 4. 6 Implementasi *Query* Menambahkan Data Pengguna

4.3.2.4 Implementasi *Query* Mengubah Data Pengguna

Query mengubah data pengguna dilakukan ketika aktor akan melakukan perubahan terhadap data pengguna. Berikut implementasi dari *query* mengubah data pengguna ditunjukkan pada Kode Sumber 4.7

```

public void UpdateUser(string _UID) {
try{
    MySqlConnection mycon = new
MySqlConnection(connectionString);
    mycon.Open();
    MySqlCommand mycommand = new MySqlCommand();
    mycommand.Connection = mycon;
    mycommand.CommandText = "update user set nama
= @a, nrp = @b, alamat = @c where id_user=@d";
    mycommand.Prepare();
    mycommand.Parameters.AddWithValue("@a",
User.naMa);
    int x = mycommand.ExecuteNonQuery();
}
catch (Exception e)
{ . . . }
}

```

Kode Sumber 4. 7 Implementasi *Query* Mengubah Data Pengguna

4.3.2.5 Implementasi *Query* Menghapus Data Pengguna

Query menghapus data pengguna dilakukan ketika aktor menginginkan menghapus data pengguna. Implementasi *query* ditunjukkan pada Kode Sumber 4.8.

```

public void DeleteUser (string _UID) {
    try{
        MySqlConnection mycon = new
MySqlConnection (connectionString) ;
        mycon.Open () ;
        MySqlCommand mycommand = new MySqlCommand () ;
        mycommand.Connection = mycon;
        mycommand.CommandText = "delete from user
where id_user = @a";
        mycommand.Prepare () ;
        mycommand.Parameters.AddWithValue ("@a",
_UID) ;
        int x = mycommand.ExecuteNonQuery () ;
    }
    catch (Exception e)
    { . . . } }

```

Kode Sumber 4. 8 Implementasi *Query* Menghapus Data Pengguna

4.3.2.6 Implementasi *Query* Mencari Data Transaksi Keluar Masuk Kendaraan

Query mencari dapat dilakukan oleh aktor maupun sistem secara otomatis. Sistem secara otomatis melakukan pencarian data transaksi ketika ada pengguna yang akan masuk dan keluar dari tempat parkir. Ini dilakukan untuk mengecek ada tidaknya data pengguna tersebut sebelum dia melakukan transaksi kembali. Kode Sumber 4.9 merupakan implementasi dari *query* mencari data transaksi keluar masuk kendaraan.

```

public void GetTransaksi (string _UID)
{
    UpdateClass () ;
    try{
        MySqlConnection mycon = new
MySqlConnection (connectionString) ;
        mycon.Open () ;
        MySqlCommand mycommand = new MySqlCommand () ;
        mycommand.Connection = mycon;

```

```

        mycommand.CommandText = "select * from
transaksi where id_user = @_id";
        mycommand.Prepare();
        mycommand.Parameters.AddWithValue("@_id",
        _UID);
        MySqlDataReader rdr =
mycommand.ExecuteReader();
        bool adaData = false;
        while (rdr.Read()){ . . . }
        if (adaData == false)
        {
            _transaksi.STATUS = "0";
        }
        mycon.Close();
    catch (Exception e)
    { . . . } }

```

Kode Sumber 4. 9 Implementasi *Query* Mencari Data Transaksi

4.3.2.7 Implementasi *Query* Mengubah Data Transaksi Ketika Keluar Parkir

Query mengubah data transaksi ini merupakan *query* untuk mengubah data transaksi tertentu sesuai kebutuhan dari aktor ketika ingin dilakukan perubahan. Namun *query* ini juga dilakukan ketika pengguna akan keluar dari tempat parkir. Setelah Petugas Parkir melakukan verifikasi terhadap data transaksi pengguna tersebut. Sehingga nantinya dilakukan perubahan bahwa pengguna sudah keluar dari tempat parkir. Implementasi *query* mengubah data transaksi ketika keluar parkir ditunjukkan di Kode Sumber 4.10.

```

public void UpdateTransaksi(string _UID, int id)
{
    try
    {
        MySqlConnection mycon = new
        MySqlConnection(connectionString);

```

```

mycon.Open();
MySQLCommand mycommand = new MySQLCommand();
mycommand.Connection = mycon;
mycommand.CommandText = "UPDATE transaksi
SET waktu_keluar = @a, tgl_keluar = @b, STATUS
= @c WHERE id_user = @d AND id_trans = @e";
mycommand.Prepare();
mycommand.Parameters.AddWithValue("@a",
Transaksi.waktuKeluar);
. . .
int x = mycommand.ExecuteNonQuery();
MySQLCommand mcom = new MySQLCommand();
mcom.Connection = mycon;
mcom.CommandText = "SELECT foto_user from
transaksi where id_trans = @f";
mcom.Prepare();
mcom.Parameters.AddWithValue("@f", id);
MySQLDataReader rdr =
mycommand.ExecuteReader();
while (rdr.Read()){
    Transaksi.fotoUser =
rdr.GetString("foto_user");
    break;
}
mycon.Close();
}
catch (Exception e)
{. . .} }

```

Kode Sumber 4. 10 Implementasi *Query* Mengubah Data Transaksi

4.3.2.8 Implementasi *Query* Pencatatan Kendaraan Masuk Parkir

Query pencatatan kendaraan masuk parkir sama halnya dengan penambahan data kendaraan keluar masuk yang dilakukan secara otomatis oleh sistem ketika ada pengguna yang akan masuk ke dalam parkir. *Query* ini diimplementasikan dan ditunjukkan pada Kode Sumber 4.11.

```

public void InsertTransaksi () {
try{
    MySqlConnection mycon = new
MySqlConnection (connectionString) ;
    mycon.Open () ;
    MySqlCommand mycommand = new MySqlCommand () ;
    mycommand.Connection = mycon ;
    mycommand.CommandText = "insert into
transaksi (id_user, waktu_masuk, tgl_masuk,
status, foto_user) values (@a, @b, @c, @d, @e)";
    mycommand.Prepare () ;
    mycommand.Parameters.AddWithValue ("@a",
Transaksi.idUser) ;
    . . .
    int x = mycommand.ExecuteNonQuery () ;
    mycon.Close () ;
}
catch (Exception e){. . .} }

```

**Kode Sumber 4. 11 Implementasi *Query* Pencatatan Kendaraan
Masuk Parkir**

4.3.2.9 Implementasi *Query* Menambahkan Data Pengguna

Query menambahkan data pengguna dilakukan ketika aktor ingin menambahkan data pengguna ke dalam basis data. Berikut adalah implementasi dari *query* penambahan data pengguna ditunjukkan pada Kode Sumber 4.12.

```

public void InsertUser () {
try{
    MySqlConnection mycon = new
MySqlConnection (connectionString) ;
    mycon.Open () ;
    MySqlCommand mycommand = new MySqlCommand () ;
    mycommand.Connection = mycon ;
    mycommand.CommandText = "insert into user
(id_user, nama, nrp, alamat) values (@a, @b,
@c, @d)";
    mycommand.Prepare () ;
    mycommand.Parameters.AddWithValue ("@a",

```



```

User.IDUser);
    . . .
    int x = mycommand.ExecuteNonQuery();
    mycon.Close();
}
catch (Exception e)
{. . .}

```

Kode Sumber 4. 12 Implementasi *Query* Menambahkan Data Pengguna

4.4. Implementasi Proses Aplikasi

Pada subbab ini akan dibahas tentang fungsi atau proses yang berjalan pada aplikasi Sistem Keamanan Kampus. Proses yang terjadi karena ada permintaan dari pengguna aplikasi maupun proses yang terjadi secara otomatis dilakukan aplikasi tanpa perlu adanya permintaan dari pengguna aplikasi. Proses-proses yang diimplementasikan disini ada yang secara langsung menjadi implementasi dari keseluruhan kasus penggunaan yang ada, dan ada juga proses yang diimplementasikan memenuhi fungsi-fungsi yang dibutuhkan oleh aplikasi secara tidak langsung.

4.4.1 Implementasi Proses Login

Proses login aktor diminta untuk memasukkan nama akun dan kata kunci ke dalam aplikasi. Apabila nama akun dan kata kunci tidak sesuai, maka aplikasi akan memberikan pesan peringatan dan aktor tidak dapat masuk ke dalam sistem. Kode Sumber 4.13 merupakan implementasi dari proses login dan diimplementasikan pada *controller* halaman login.

```

private void btnLogin_Click(object sender,
EventArgs e){
    AksesController log = new AksesController();
    if(log.GetPengguna(tbUserLogin.Text,
tbPassLogin.Text))

```

```

    {
        Session.Id =
            log.Akses.idAkses.ToString();
        Session.Login = true;

        log.GetPegguna(Convert.ToInt32(Session.Id));
        if (Session.Id == "1")
        {
            warning.Text = "Selamat Datang, " +
log.Akses.namaAkses.ToString() + " !";
            warning.Show();
            this.Hide();
            MainNFC sistem = new MainNFC();
            sistem.ShowDialog();
        }
        else if (Session.Id == "0")
        {
            warning.Text = "Selamat Datang, " +
log.Akses.namaAkses.ToString() + " !";
            warning.Show();
            this.Hide();
            MenuAdmin ma = new MenuAdmin();
            ma.ShowDialog();
        }
        else
        { . . . }
    }
    else
    { . . . }
}

```

Kode Sumber 4. 13 Implementasi Proses Login

4.4.2 Implementasi Proses Pendeteksian Kartu Cerdas

Berikut ini merupakan implementasi dari proses pendeteksian kartu cerdas. Proses ini merupakan proses yang terjadi ketika ada pengguna akan masuk ke dalam parkir dan ketika kemudian pengguna mendekatkan kartu cerdas ke alat pembaca NFC. Proses dimulai dari melakukan pengecekan

terhadap ketersediaan alat, mengecek adanya kartu yang masuk dan kemudian mengambil data yang ada di dalam kartu cerdas. Kode Sumber 4.14 merupakan implementasi dari proses pendeteksian kartu cerdas.

```

public void clearBuffers() {
    long indx;
    for (indx = 0; indx <= 262; indx++){
        RecvBuff[indx] = 0;
        sendBuff[indx] = 0;
    }
}

public List<String> getLoader() {
    try{
        int retCodetemp = establishContext();
        if (retCodetemp !=
ModWinsCard.SCARD_S_CARDLIST_READERS(this.hContext,
null, null, ref pcchReaders))
        { }
        else{
            retCodetemp = listPCSC();
            if (retCode !=
ModWinsCard.SCARD_S_SUCCESS){ }
            else{
                byte[] readersList = new
byte[pcchReaders];
                retCodetemp = listReader(readersList);
                if (retCode !=
ModWinsCard.SCARD_S_SUCCESS)
                { }
                else{
                    rName = "";
                    index = 0;

                    while (readersList[index] != 0){
                        while (readersList[index] != 0){
                            rName = rName +
(char) readersList[index];

```

```

        index = index + 1;
    }

    readerNFC.Add(rName);

    rName = "";
    index = index + 1;
}
}
}
}
}
return readerNFC;
}
catch (Exception e){return readerNFC;}
}

public int getConnect(){
    if (readerNFC.Count>0){
        retCode =
ModWinsCard.SCardConnect(hContext, readerNFC[0],
ModWinsCard.SCARD_SHARE_SHARED,
ModWinsCard.SCARD_PROTOCOL_T0 |
ModWinsCard.SCARD_PROTOCOL_T1, ref hCard, ref
Protocol);
        return retCode;
    }
    return -2146435063;
}

public int establishContext(){
    retCode =
ModWinsCard.SCardEstablishContext(ModWinsCard.SC
ARD_SCOPE_USER, 0, 0, ref hContext);
    return retCode;
}

public int listPCSC(){
    retCode =
ModWinsCard.SCardListReaders(this.hContext,

```

```

null, null, ref pcchReaders);
    return retCode;
}

public int listReader(byte[] readersList){
    retCode =
ModWinsCard.SCardListReaders(this.hContext,
null, readersList, ref pcchReaders);
    return retCode;
}

public int sendAPDU(){
    int index;
    string tmpStr;
    pioSendRequest.dwProtocol = aProtocol;
    pioSendRequest.cbPciLength = 8;
    tmpStr = "";
    for (index = 0; index <= sendLen - 1;
index++){
        tmpStr = tmpStr + " " +
string.Format("{0:X2}", sendBuff[index]);
    }

    retCode = ModWinsCard.SCardTransmit(hCard,
ref pioSendRequest, ref sendBuff[0], sendLen,
ref pioSendRequest, ref RecvBuff[0], ref
recvLen);
    if (retCode ==
ModWinsCard.SCARD_E_INSUFFICIENT_BUFFER){
        pioSendRequest.cbPciLength = 10;
        int retco =
ModWinsCard.SCardTransmit(hCard, ref
pioSendRequest, ref sendBuff[0], sendLen, ref
pioSendRequest, ref RecvBuff[0], ref recvLen);
    }
    tmpStr = "";
    for (index = 0; index <= recvLen - 1;
index++){
        tmpStr = tmpStr + " " +

```

```

string.Format("{0:X2}", RecvBuff[index]);
    }
    return retCode;
}

public string getUID() {
    string tmpStr;
    int length = 14;
    int index;
    clearBuffers();
    sendBuff[0] = 0xFF;
    sendBuff[1] = 0xCA;
    sendBuff[2] = 0x00;
    sendBuff[3] = 0x00;
    sendBuff[4] = 0x00;
    sendLen = 5;
    recvLen = (byte)length;
    try{
        retCode = sendAPDU();
        if (retCode !=
ModWinsCard.SCARD_S_SUCCESS) {
            return "-";
        }
        else{
            tmpStr = "";
            for (index = recvLen - 2; index <=
recvLen - 1; index++)
            {
                tmpStr = tmpStr + " " +
string.Format("{0:X2}", RecvBuff[index]);
            }
            tmpStr = "";
            for (index = 0; index <= recvLen - 3;
index++){
                tmpStr = tmpStr + " " +
string.Format("{0:X2}", RecvBuff[index]);
            }
            uid = tmpStr;

```

```

clearBuffers ();
return uid;
}
catch (Exception e){. . .} }

```

Kode Sumber 4. 14 Implementasi Proses Pendeteksian Kartu Cerdas

4.4.3 Implementasi Proses Pengambilan Gambar dengan Webcam

Berikut merupakan implementasi dari proses pengambilan gambar dengan webcam. Proses pengambilan webcam ini merupakan proses kelanjutan dari proses pencatatan kendaraan ketika pengguna akan masuk ke dalam parkir kemudian mendekati kartu cerdas ke alat pembaca NFC. Kode Sumber 4.15 merupakan implemenatasi dari proses pengambilan gambar dengan webcam.

```

public static void
SaveImageCapture(System.Drawing.Image image,
string nilai, int id){
    string pathfile =
Application.StartupPath.ToString() +
"\\SaveImage\\" + id.ToString() + "Image" +
nilai + ".jpg";
    FileStream fstream2 = new
FileStream(pathfile, FileMode.Create);
    image.Save(fstream2,
System.Drawing.Imaging.ImageFormat.Jpeg);
    fstream2.Close();
}

```

Kode Sumber 4. 15 Implementasi Proses Pengambilan Gambar dengan Webcam

4.4.4 Implementasi Proses Pencatatan Kendaraan Masuk Parkir

Berikut merupakan implementasi dari proses pencatatan kendaraan masuk parkir. Proses ini merupakan proses setelah

dilakukannya proses pendeteksian kartu cerdas, dan disaat bersamaan ketika proses ini dijalankan, melakukan proses pengambilan gambar oleh webcam. Implementasi dijelaskan pada Kode Sumber 4.16.

```

public void InsertTransaksi () {
    try{
        DateTime masuk = DateTime.Now;
        int nextID =
controlTrans.Transaksi.idTrans + 1;

        AppsController.SaveImageCapture (pb_live.Image,
uid, nextID);

        . . .
        controlTrans.InsertTransaksi ();
        MessageBox.Show ("Data Telah Disimpan");
    }
    catch (Exception ex){
System.Console.WriteLine (ex.StackTrace.ToString (
));
    }
    controlApps.getDGV (dgvMain, "select * from
transaksi order by id_trans desc limit 10");
    waktu.Start ();
    statuskartu = false;
}

```

Kode Sumber 4. 16 Implementasi Proses Pencatatan Kendaraan Masuk Parkir

4.4.5 Implementasi Proses Verifikasi Kendaraan yang Keluar Parkir

Proses verifikasi dilakukan ketika pengguna akan keluar dari tempat parkir. Sebelumnya pengguna yang akan keluar mendekati kembali kartu cerdasnya untuk kemudian mendapatkan data-data dari kartu cerdas. Setelah itu kemudian dilakukan pencarian terhadap data masuk dari pengguna tersebut.

Setelah dilakukan verifikasi, dilakukan perubahan data masuk menjadi data keluar. Kode Sumber 4.17 merupakan implementasi dari proses verifikasi kendaraan yang keluar dari parkir.

```

public void GetUpdate () {
    try {
        string pathkeluar =
Application.StartupPath.ToString () +
"\\SaveImage\\" +
controlTrans.Transaksi.idTrans.ToString () +
"Image" + uid.ToString () + ".jpg";
        pb_show.Image = Image.FromFile (pathkeluar);
        . . .
        if (MessageBox.Show ("Konfirmasi Pengguna? ",
"Konfirmasi", MessageBoxButtons.YesNo) ==
DialogResult.Yes) {
            controlTrans.UpdateTransaksi (uid,
controlTrans.Transaksi.idTrans);
            MessageBox.Show ("Pengguna Keluar dan Data
Telah Disimpan");
        }
    }
    . . .
}

```

Kode Sumber 4. 17 Implementasi Verifikasi kendaraan yang Keluar Parkir

4.4.6 Implementasi Proses Mencari Data Pengguna

Berikut merupakan implementasi dari proses mencari data pengguna. Ketika Administrator ingin mencari pengguna tertentu. Administrator memilih mencari pengguna berdasarkan kategori tertentu. Kode Sumber 4.18 merupakan implementasi dari proses pencarian data pengguna.

```

private void btnSearchPengguna_Click (object
sender, EventArgs e) {
    if (tbSearchPengguna.Text.Equals ("")) {
        string searchItem =

```

```

cbSearchPegguna.SelectedItem.ToString();
    string id;
    switch (searchItem){
        case "ID Pengguna":
            id = "select * from user order by
id_user"; break;
            . . .
        default:
            id = "select * from user order by
id_user"; break;
    }
    controlApps.getDGV(dgvPengguna, id);
}
else{
    string searchItem =
cbSearchPegguna.SelectedItem.ToString();
    string id = tbSearchPegguna.Text;
    string query;
    switch (searchItem){
        case "ID Pengguna":
            query = "select * from user where
id_user like '%" + id + "%'"; break;
            . . .
        default:
            query = "select * from user order by
id_user"; break;
    } } }

```

Kode Sumber 4. 18 Implementasi Proses Mencari Data Pengguna

4.4.7 Implementasi Proses Menambah Data Pengguna

Penambahan data pengguna dapat dilakukan oleh Administrator dan Petugas Parkir ketika data pengguna yang akan masuk ke tempat parkir belum terdaftar di dalam basis data. Proses ini dilakukan dengan mendapatkan ID dari kartu cerdas, dan kemudian mengisikan isian yang harus ditambahkan ke dalam basis data. Kode Sumber 4.19 merupakan implementasi dari proses menambah data pengguna

```

private void btnInsertPengguna_Click(object
sender, EventArgs e){
    getConnect();
    if (MessageBox.Show("Apakah Anda Yakin akan
Memasukkan Data? ", "Konfirmasi",
MessageBoxButtons.YesNo) == DialogResult.Yes){
        controlUser.UpdateClass();
        controlUser.User.IDUser = uid;
        controlUser.User.naMa = tbNama.Text;
        controlUser.User.nRP = tbNRP.Text;
        controlUser.User.aLamat = tbAlamat.Text;
        controlUser.InsertUser();
        string id = "select * from user order by
id_user";
        controlApps.getDGV(dgvPengguna, id);
        MessageBox.Show("Data Pengguna berhasil
ditambahkan");
        clear();
    } }

```

Kode Sumber 4.19 Implementasi Proses Menambah Data Pengguna

4.4.8 Implementasi Proses Mengubah Data Pengguna

Pengubahan data pengguna dapat dilakukan oleh Administrator ketika dibutuhkan sebuah perubahan. Berikut adalah implementasi dari proses mengubah data pengguna yang ditunjukkan pada Kode Sumber 4.20.

```

private void btnUpdatePengguna_Click(object
sender, EventArgs e){
    if (MessageBox.Show("Apakah Anda Yakin akan
Mengubah Data? ", "Konfirmasi",
MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        controlUser.UpdateClass();
        controlUser.User.IDUser = tbIDUser.Text;
        controlUser.User.naMa = tbNama.Text;
        controlUser.User.nRP = tbNRP.Text;
        controlUser.User.aLamat = tbAlamat.Text;
        controlUser.UpdateUser(tbIDUser.Text);
    }

```

```

        string id = "select * from user order by
id_user";
        controlApps.getDGV(dgvPegguna, id);
        MessageBox.Show("Data Pegguna telah
diperbaharui");
        clear();
    } }

```

Kode Sumber 4. 20 Implementasi Proses Mengubah Data Pengguna

4.4.9 Implementasi Proses Menghapus Data Pengguna

Berikut implementasi dari proses menghapus data pengguna yang dapat dilakukan oleh Administrator di dalam aplikasi Sistem Keamanan ini. Penghapusan dengan memilih pengguna mana yang akan dihapus dari basis data. Kode Sumber 4.21 merupakan implementasi dari proses menghapus data pengguna.

```

private void btnDeletePengguna_Click(object
sender, EventArgs e)
{
    if (MessageBox.Show("Apakah Anda Yakin akan
Menghapus Data? ", "Konfirmasi",
MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        controlUser.UpdateClass();
        controlUser.User.IDUser = tbIDUser.Text;
        controlUser.DeleteUser(controlUser.User.IDUser);
        string id = "select * from user order by
id_user";
        controlApps.getDGV(dgvPegguna, id);
        MessageBox.Show("Data Pegguna telah
dihapus");
        clear();
    }
}

```

Kode Sumber 4. 21 Implementasi Proses Menghapus Data Pengguna

4.4.10 Implementasi Proses Mencari Data Transaksi Keluar Masuk Kendaraan

Proses implementasi data transaksi keluar masuk kendaraan merupakan proses yang dapat dilakukan oleh Administrator dan Petugas Parkir. Baik itu untuk mencari keseluruhan transaksi keluar masuk maupun untuk mencari berdasarkan kategori tertentu. Kode Sumber 4.22 merupakan implementasi dari proses mencari data transaksi keluar masuk.

```
private void btnSearchTransaksi_Click(object
sender, EventArgs e)
{
    if (tbSearchTransaksi.Text.Equals(""))
    {
        string searchItem =
cbSearchTrans.SelectedItem.ToString();
        string id = tbSearchTransaksi.Text;
        switch (searchItem)
        {
            case "User ID":
                id = "SELECT * FROM transaksi ORDER BY
id_user"; break;
                . . .
            default:
                id = "SELECT * FROM transaksi ORDER BY
id_user"; break;
        }
        controlApps.getDGV(dgvTransaksi, id);
    }
    else
    {
        string searchItem =
cbSearchTrans.SelectedItem.ToString();
        string id = tbSearchTransaksi.Text;
        string query;
        switch (searchItem)
        {
            case "User ID":
```

```

        query = "SELECT * FROM transaksi where
id_user like '%" + id + "%'"; break;
        . . .
        default:
            query = "SELECT * FROM transaksi order
by id_trans"; break;
    }
    controlApps.getDGV(dgvTransaksi, query);
}

```

Kode Sumber 4. 22 Implementasi Proses Mencari Data Transaksi

4.4.11 Implementasi Proses Mengubah Data Transaksi Keluar Masuk Kendaraan

Proses mengubah data transaksi keluar masuk kendaraan merupakan proses yang bisa dilakukan oleh Administrator. Proses ini dilakukan dengan cara memilih terlebih dahulu data mana yang akan diubah. Setelah itu lakukan pengubahan sesuai kebutuhan. Kemudian menekan tombol ubah untuk mengeksekusi proses ini. Berikut Implementasi dari proses mengubah data transaksi keluar masuk kendaraan ditunjukkan pada Kode Sumber 4.23.

```

private void btnUpdateTransaksi_Click(object
sender, EventArgs e)
{
    if (MessageBox.Show("Apakah Anda Yakin akan
Mengubah Data? ", "Konfirmasi",
MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        if (cbKeluar.Checked == false)
        {
            controlTrans.Transaksi.waktuKeluar = null;
        }
        else
        {
            DateTime keluar = DateTime.Now;
            controlTrans.Transaksi.tanggalKeluar = keluar;
        }
    }
}

```

```

        controlTrans.Transaksi.waktuKeluar =
        keluar.ToLongTimeString();

controlTrans.Transaksi.STATUS = "Keluar";

controlTrans.UpdateTransaksi(controlTrans.Transaksi.idUser, controlTrans.Transaksi.idTrans);
    }
}
}
}

```

Kode Sumber 4. 23 Implementasi Proses Mengubah Data Transaksi

4.4.12 Implementasi Proses Menghapus Data Transaksi Keluar Masuk Kendaraan

Berikut implementasi dari proses menghapus data transaksi keluar masuk kendaraan. Proses ini dapat dilakukan oleh Administrator dengan memilih data mana yang akan dihapus dari basis data. Kode Sumber 4.24 merupakan implementasi dari proses menghapus data transaksi keluar masuk kendaraan.

```

private void btnHapus_Click(object sender,
EventArgs e)
{
    if (MessageBox.Show("Apakah Anda Yakin akan
Menghapus Data? ", "Konfirmasi",
MessageBoxButtons.YesNo) == DialogResult.Yes)
    {
        controlTrans.deleteTransaksi(idtrans);
        string id = "select * from transaksi order
by id_trans desc";
        controlApps.getDGV(dgvTransaksi, id);
        clear();
    }
}
}

```

**Kode Sumber 4. 24 Implementasi Proses Menghapus Data
Transaksi**

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas rangkaian pengujian dan evaluasi terhadap perangkat lunak yang dibangun pada Tugas Akhir ini. Pengujian dilakukan secara menyeluruh untuk memeriksa apakah fungsional yang dibangun telah berjalan sesuai kebutuhan perangkat lunak. Bab ini mencakup lingkungan pengujian, skenario pengujian, dan evaluasi terhadap pengujian.

5.1. Lingkungan Pengujian

Pengujian terhadap aplikasi Sistem Keamanan Kampus dilakukan pada dua jenis perangkat, yaitu:

1. Personal Computer dengan spesifikasi sebagai berikut.
 - a. Processor Intel® Core™ i3 CPU 530 @2.93GHz,
 - b. Memori (RAM) 4,00 GB, dan
 - c. Sistem Operasi Windows 8 Pro 64-bit.
2. Laptop HP Compaq Presario CQ42-258VX dengan spesifikasi sebagai berikut.
 - a. Processor Intel® Core™ i5 M450 @2,4 Ghz,
 - b. Memori (RAM) 4,00 GB, dan
 - c. Sistem Operasi Windows 8.1 Pro 64-bit.

5.2. Skenario Pengujian

Pada subbab ini akan dibahas pengujian perangkat lunak perangkat lunak Sistem Keamanan Kampus untuk menguji fungsionalitas dari perangkat lunak tersebut. Pengujian fungsionalitas dilakukan dengan model *blackbox* untuk masing-masing fungsi dari perangkat lunak. Pengujian dilakukan untuk menguji apakah fungsionalitas yang diidentifikasi pada tahap analisis kebutuhan benar-benar diimplementasikan dan berfungsi seperti yang diharapkan. Pengujian didokumentasikan secara sistematis sebagai tolok ukur keberhasilan perangkat lunak.

5.2.1. Pengujian Fungsionalitas

Pada subbab ini akan dibahas pengujian pada aplikasi Sistem Keamanan Kampus yang digunakan untuk melihat, memasukkan dan mengolah data yang dibutuhkan aplikasi. Pengguna pada aplikasi adalah Administrator dan Petugas Parkir. Pengujian fungsionalitas dilakukan dengan menyiapkan sejumlah skenario sebagai tolok ukur keberhasilan pengujian. Pengujian fungsionalitas dengan mengacu pada kasus penggunaan yang telah dijelaskan pada subbab 3.2.1. Pengujian fungsionalitas dijabarkan pada subbab berikut.

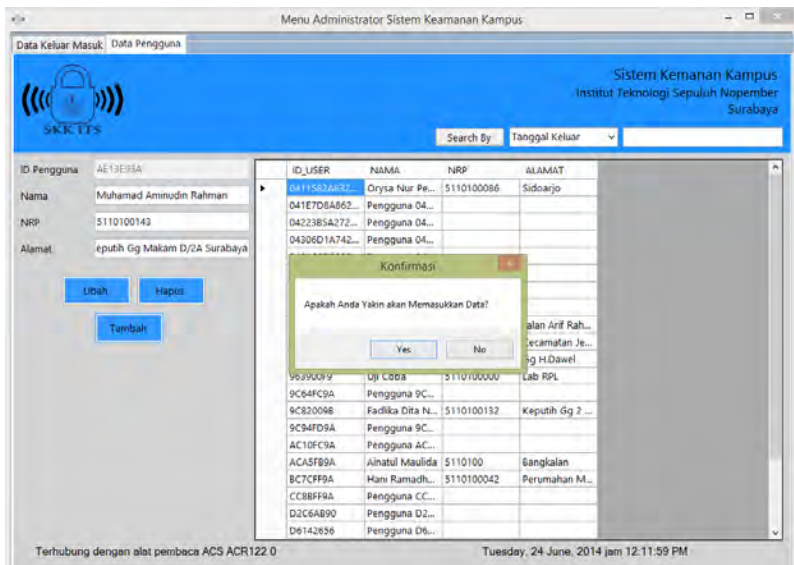
5.2.1.1 Pengujian Mengelola Data Pengguna

Pengujian pengelolaan data pengguna dilakukan dengan skenario Administrator memilih menu mengelola data pengguna. Pengelolaan data pengguna dibagi menjadi beberapa proses pengujian yaitu menambah data pengguna, mengubah data pengguna, dan menghapus data pengguna. Skenario pengujian menambah data pengguna dijelaskan pada Tabel 5.1. Skenario pengujian mengubah data pengguna dijelaskan pada Tabel 5.2. Skenario pengujian menghapus data tempat dijelaskan pada Tabel 5.3. Ilustrasi proses pengujian pengelolaan data pengguna dijelaskan pada Gambar 5.1 dan Gambar 5.2.

Tabel 5. 1 Skenario Pengujian Menambah Data Pengguna

Kode Pengujian	PUJ.UC-001
Nama	Pengujian menambahkan data pengguna
Referensi Kasus Penggunaan	UC-004
Tujuan	Memeriksa fungsi untuk menambahkan data pengguna berfungsi atau tidak
Kondisi awal	Data pengguna belum terdaftar, alat pembaca NFC telah terhubung dan berfungsi dengan baik

Skenario	<ol style="list-style-type: none"> 1. Administrator melakukan login. 2. Administrator memilih menu mengelola data pengguna. 3. Administrator mendekatkan kartu cerdas baru ke alat pembaca NFC untuk mendapatkan ID kartu cerdas. 4. Administrator mengisikan data pengguna yang baru. 5. Administrator menyimpan data pengguna dengan menekan tombol tambah.
Masukan	Data pengguna baru, UID Kartu
Keluaran yang diharapkan	Data pengguna tersimpan ke dalam basis data
Hasil Pengujian	Data pengguna baru berhasil ditambahkan ke dalam basis data. Pengujian berhasil

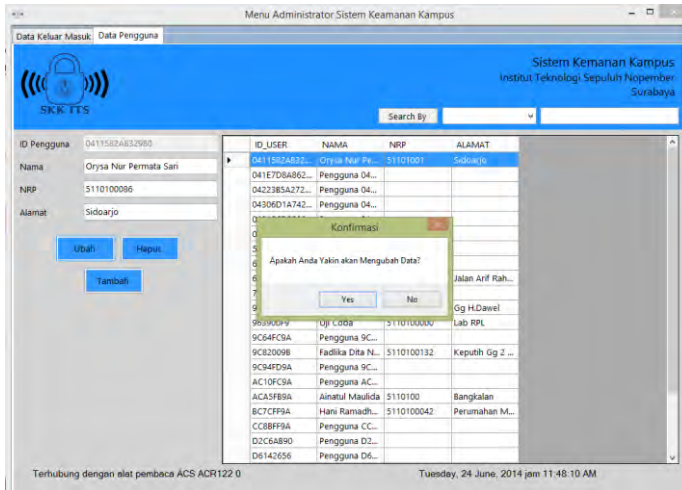


Gambar 5. 1 Pengujian Menambahkan Data Pengguna

Dalam pengujian menambahkan data pengguna baru, data pengguna yang baru saja dimasukkan secara langsung akan muncul di dalam tabel dan secara langsung dapat melakukan fungsi yang lainnya atau ingin menambahkan kembali data pengguna.

Tabel 5. 2 Skenario Pengujian Mengubah Data Pengguna

Kode Pengujian	PUJ.UC-002
Nama	Pengujian mengubah data pengguna
Referensi Kasus Penggunaan	UC-004
Tujuan	Memeriksa fungsi untuk mengubah data pengguna berfungsi atau tidak
Kondisi awal	Data pengguna sudah terdaftar, alat pembaca NFC telah terhubung dan berfungsi dengan baik
Skenario	<ol style="list-style-type: none"> 1.Administrator melakukan login 2.Administrator memilih menu mengelola data pengguna 3.Administrator memilih salah satu pengguna 4.Administrator mengisikan data pengguna yang diubah. 5.Administrator menyimpan data pengguna dengan menekan tombol ubah.
Masukan	Data pengguna baru yang diubah, UID
Keluaran yang diharapkan	Data pengguna berubah dan tersimpan ke dalam basis data
Hasil Pengujian	Data pengguna telah berubah dan tersimpan di dalam basis data. Pengujian berhasil.



Gambar 5. 2 Pengujian Mengubah Data Pengguna

Dalam pengujian mengubah data pengguna, data pengguna yang diubah secara langsung akan muncul di dalam tabel dan secara langsung dapat melakukan fungsi yang lainnya atau ingin menambahkan kembali data pengguna.

Tabel 5. 3 Skenario Pengujian Menghapus Data Pengguna

Kode Pengujian	PUJ.UC-003
Nama	Pengujian menghapus data pengguna
Referensi Kasus Penggunaan	UC-004
Tujuan	Memeriksa fungsi untuk menghapus data pengguna berfungsi atau tidak
Kondisi awal	Data pengguna sudah terdaftar, alat pembaca NFC telah terhubung dan berfungsi dengan baik
Skenario	1. Administrator melakukan login 2. Administrator memilih menu mengelola data pengguna

	3.Administrator memilih salah satu pengguna 4.Administrator menghapus data pengguna dengan menekan tombol hapus.
Masukan	Penghapusan data pengguna
Keluaran yang diharapkan	Data pengguna berhasil dihapus dan data di dalam tabel hilang.
Hasil Pengujian	Data di dalam basis data terhapus dan data di dalam tabel menghilang, pengujian berhasil.

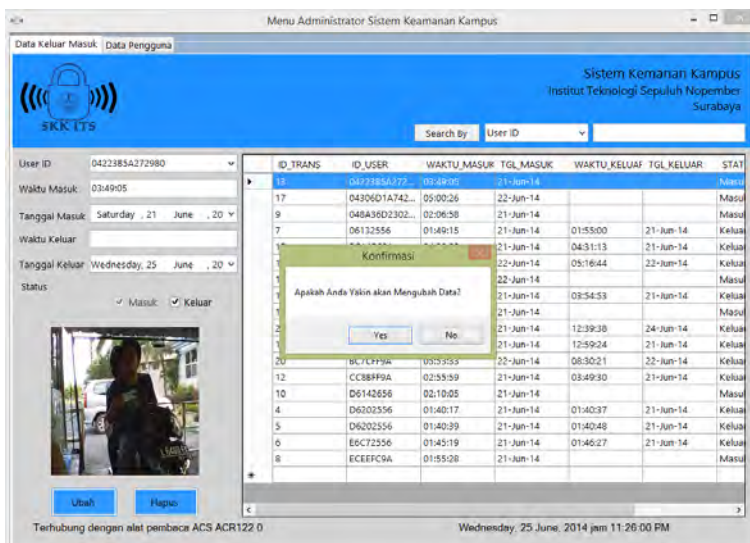
5.2.1.2 Pengujian Mengelola Rekapitulasi Keluar masuk kendaraan

Pengujian pengelolaan rekapitulasi keluar masuk kendaraan dilakukan dengan skenario Administrator memilih menu mengelola data keluar masuk. Pengelolaan rekapitulasi keluar masuk kendaraan dibagi menjadi beberapa proses pengujian yaitu mengubah data transaksi dan menghapus data transaksi. Skenario pengujian mengubah data transaksi dijelaskan pada Tabel 5.4. Skenario pengujian menghapus data transaksi dijelaskan pada Tabel 5.5. Ilustrasi proses pengujian pengelolaan data pengguna dijelaskan pada Gambar 5.3.

Tabel 5. 4 Skenario Mengubah Data Transaksi

Kode Pengujian	PUJ.UC-004
Nama	Pengujian mengubah data transaksi keluar masuk kendaraan
Referensi Kasus Penggunaan	UC-005
Tujuan	Memeriksa fungsi untuk mengubah data transaksi keluar masuk kendaraan berfungsi atau tidak
Kondisi awal	Data transaksi sudah terdaftar di dalam basis data dan alat pembaca NFC telah terhubung

	dan berfungsi dengan baik
Skenario	<ol style="list-style-type: none"> 1. Administrator melakukan login 2. Administrator memilih menu mengelola Rekapitulasi Keluar Masuk Kendaraan 3. Administrator memilih salah satu transaksi 4. Administrator mengisikan data transaksi yang diubah. 5. Administrator menyimpan data transaksi dengan menekan tombol ubah.
Masukan	Perubahan data transaksi keluar masuk kendaraan.
Keluaran yang diharapkan	Data transaksi berhasil diubah dan tersimpan ke dalam basis data
Hasil Pengujian	Data transaksi berubah, pengujian berhasil.



Gambar 5. 3 Pengujian Mengubah Data Transaksi

Tabel 5. 5 Skenario Menghapus Data Transaksi

Kode Pengujian	PUJ.UC-005
Nama	Pengujian menghapus data transaksi keluar masuk kendaraan.
Referensi Kasus Penggunaan	UC-005
Tujuan	Memeriksa fungsi untuk menghapus data transaksi keluar masuk kendaraan berfungsi atau tidak.
Kondisi awal	Data transaksi keluar masuk kendaraan sudah terdaftar di dalam basis data dan alat pembaca NFC telah terhubung dan berfungsi dengan baik.
Skenario	<ol style="list-style-type: none"> 1.Administrator melakukan login. 2.Administrator memilih menu mengelola. rekapitulasi keluar masuk kendaraan. 3.Administrator memilih salah satu data transaksi 4.Administrator menghapus data transaksi dengan menekan tombol hapus.
Masukan	Penghapusan data transaksi keluar masuk kendaraan
Keluaran yang diharapkan	Data transaksi di dalam basis data terhapus, dan data di tabel hilang.
Hasil Pengujian	Data transaksi terhapus dari basis data, data di dalam tabel hilang, pengujian berhasil.

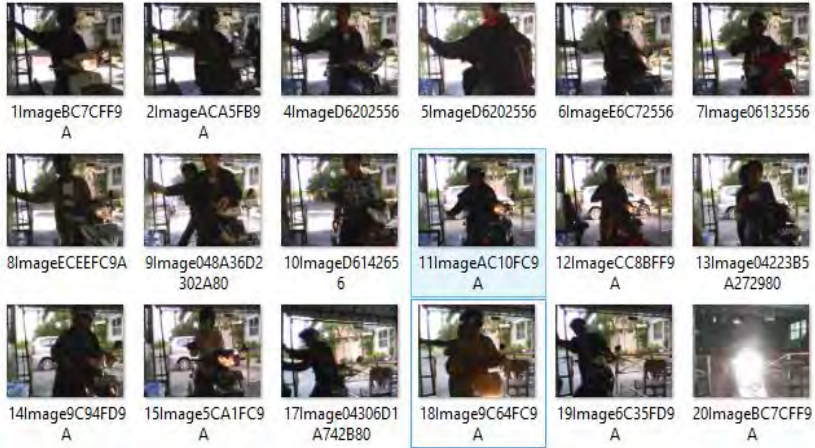
5.2.1.3 Pengujian Mencatat Kendaraan Masuk Parkir

Berikut ini pembahasan pengujian mencatat kendaraan ketika masuk parkir. Pengujian ini dilakukan oleh Petugas parkir Skenario rinci pengujian ini dijelaskan pada Tabel 5.6 dan hasil pengujian ditunjukkan pada Gambar 5.4.

Tabel 5. 6 Skenario Pengujian Mencatat Kendaraan Masuk Parkir

Kode Pengujian	PUJ.UC-006
Nama	Pengujian menambahkan data pengguna.
Referensi Kasus Penggunaan	UC-002
Tujuan	Memeriksa fungsi untuk mencatat kendaraan masuk parkir berfungsi atau tidak.
Kondisi awal	Data transaksi belum terdaftar, alat pembaca NFC dan webcam telah terhubung dan berfungsi dengan baik.
Skenario	<ol style="list-style-type: none"> 1. Petugas Parkir melakukan login. 2. Petugas Parkir mendekatkan kartu cerdas ke alat pembaca NFC. 3. Sistem mendapatkan data dari kartu cerdas dan mendapatkan gambar pada saat itu dari webcam. 4. Sistem secara otomatis akan menambah data transaksi yang baru.
Masukan	UID dari kartu cerdas, foto webcam, waktu saat ini.
Keluaran yang diharapkan	Data transaksi baru tersimpan ke dalam basis data
Hasil Pengujian	Transaksi berhasil ditambahkan, pengujian berhasil.

Gambar 5.4 adalah beberapa foto yang berhasil diambil di saat transaksi dilaksanakan. Foto tersebut diambil setiap ada kendaraan yang masuk ke dalam parkir. Pengambilan foto bersamaan dengan mencatat data-data lain yang dibutuhkan untuk proses pencatatan kendaraan masuk parkir. Foto itu diletakkan di dalam folder tertentu yang telah didefinisikan di dalam aplikasi.



Gambar 5. 4 Hasil Penyimpanan Pada Saat Transaksi

Transaksi	ID Pengguna	Waktu Masuk	Tanggal Masuk	Waktu Keluar	Tanggal Keluar	status	Foto Masuk
15	5CA1FC9A	04:30:09	21-Jun-14	04:31:13	21-Jun-14	Keluar	C:\Users\Fah...
14	9C94FD9A	03:54:07	21-Jun-14	03:54:53	21-Jun-14	Keluar	C:\Users\Fah...
▶ 13	04223B5A272...	03:49:05	21-Jun-14			Masuk	C:\Users\Fah...
12	CC88FF9A	02:55:59	21-Jun-14	03:49:30	21-Jun-14	Keluar	C:\Users\Fah...
11	AC10FC9A	02:39:23	21-Jun-14			Masuk	C:\Users\Fah...
*							

Terhubung dengan alat pembaca ACS ACR122 0

Gambar 5. 5 Beberapa Transaksi Terakhir yang Tersimpan

Gambar 5.5 diatas merupakan hasil beberapa transaksi yang baru saja ditambahkan ketika dilakukan pencatatan kendaraan masuk parkir yang ditampilkan di halaman utama.

5.2.1.4 Pengujian Melakukan Verifikasi Kendaraan Keluar Parkir

Berikut ini pembahasan pengujian melakukan verifikasi kendaraan keluar parkir. Petugas parkir melakukan verifikasi

setiap ada pengguna yang akan keluar parkir dengan mendekatkan kartu cerdas terlebih dahulu ke alat pembaca NFC. Kemudian, sistem menampilkan data berdasarkan ID yang ada di dalam kartu cerdas untuk di tampilkan ke dalam di halaman utama aplikasi. Kemudian, petugas parkit melakukan verifikasi dengan mencocokkan dengan data yang telah disimpan sebelumnya. Skenario rinci pengujian ini dijelaskan pada Tabel 5.7.

Tabel 5. 7 Skenario Pengujian Melakukan Verifikasi Kendaraan Keluar Parkir

Kode Pengujian	PUJ.UC-007
Nama	Pengujian melakukan verifikasi kendaraan keluar parkir
Referensi Kasus Penggunaan	UC-003
Tujuan	Memeriksa fungsi untuk melakukan verifikasi kendaraan keluar parkir berfungsi atau tidak.
Kondisi awal	Data transaksi sudah terdaftar, alat pembaca NFC telah terhubung dan berfungsi dengan baik.
Skenario	<ol style="list-style-type: none"> 1. Petugas Parkir melakukan login. 2. Petugas Parkir mendekatkan kartu cerdas ke alat pembaca NFC. 3. Sistem mendapatkan data dari kartu cerdas. 4. Sistem melakukan pencarian data transaksi yang sesuai dengan data yang didapat dari kartu cerdas 5. Petugas Parkir melakukan verifikasi terhadap pengguna yang akan keluar 6. Petugas Parkir menekan tombol OK yang menandakan verifikasi pengguna benar.

Masukan	ID dari kartu cerdas dan data transaksi di dalam basis data
Keluaran yang diharapkan	Data transaksi diubah dan tersimpan ke dalam basis data
Hasil Pengujian	Berhasil

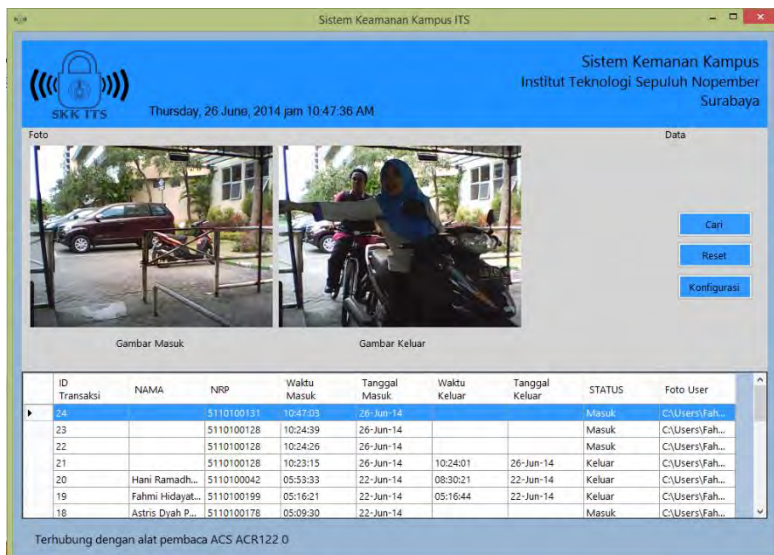
5.2.1.5 Pengujian Melihat Transaksi Keluar Masuk Kendaraan

Berikut ini pengujian melihat transaksi keluar masuk kendaraan. Petugas Parkir dapat melihat data transaksi berdasarkan berbagai kategori sesuai dengan yang diinginkan. Skenario rinci pengujian ini dijelaskan pada Tabel 5.8 dan hasil pengujian ditunjukkan pada Gambar 5.6.

Tabel 5. 8 Skenario Pengujian Melihat Transaksi Keluar Masuk Kendaraan

Kode Pengujian	PUJ.UC-008
Nama	Pengujian melihat transaksi keluar masuk kendaraan.
Referensi Kasus Penggunaan	UC-001
Tujuan	Memeriksa fungsi untuk melihat data transaksi keluar masuk kendaraan berfungsi atau tidak.
Kondisi awal	Data transaksi keluar masuk telah tersimpan di dalam basis data
Skenario	<ol style="list-style-type: none"> 1. Petugas Parkir melakukan login. 2. Petugas Parkir memilih pencarian data dengan kategori tertentu. 3. Petugas Parkir memilih menampilkan dengan menekan tombol cari
Masukan	Kategori pencarian, dan kata pencarian

Keluaran yang diharapkan	Data transaksi ditampilkan dalam tabel.
Hasil Pengujian	Berhasil



Gambar 5. 6 Pengujian Melihat data transaksi keluar masuk kendaraan

5.2.1.6 Pengujian Login Aplikasi

Berikut ini merupakan pengujian yang dilakukan untuk menguji aktor dapat menggunakan aplikasi sesuai dengan peran pengguna atau hak akses yang diberikan kepadanya. Skenario rinci pengujian ini dijelaskan pada Tabel 5.9.

Tabel 5. 9 Skenario Pengujian Login Aplikasi

Kode Pengujian	PUJ.NF-001
Nama	Pengujian melihat transaksi keluar masuk kendaraan.

Referensi Kasus Penggunaan	-
Tujuan	Menguji fungsi login pada pengguna
Kondisi awal	-
Skenario 1	<ol style="list-style-type: none"> 1. Pengguna aplikasi menjalankan aplikasi 2. Pengguna memasukkan nama akun dan kata kunci yang sesuai 3. Pengguna memilih login dengan menekan tombol login
Masukan	Data Nama akun dan kata kunci yang benar
Keluaran	Berhasil masuk ke halaman sesuai dengan perannya.
Hasil Pengujian	Berhasil
Skenario 2	<ol style="list-style-type: none"> 1. Pengguna aplikasi menjalankan aplikasi 2. Pengguna memasukkan nama akun dan kata kunci yang tidak sesuai 3. Pengguna memilih login dengan menekan tombol login
Masukan	Nama akun dan kata kunci yang tidak sesuai
Keluaran	Tidak dapat masuk ke halaman sesuai dengan perannya dan pengguna tetap berada di halaman login
Hasil Pengujian	Berhasil

Gambar 5.7 merupakan hasil pengujian ketika dilakukan pengujian dengan memasukkan kata kunci yang salah. Aktor tidak dapat masuk ke dalam sistem dan tetap di halaman login. Terdapat pesan yang menyatakan bahwa kata kunci atau nama akun yang dimasukkan aktor salah.



Gambar 5.7 Hasil Ujicoba Login

5.2.2. Pengujian Kegunaan

Pengujian kegunaan dilakukan dengan menguji aplikasi secara langsung dengan melibatkan pengguna secara langsung selama beberapa hari dan rentang waktu tertentu.

5.2.2.1 Kriteria Pengguna

Selama uji coba berlangsung, beberapa pengguna yang masuk parkir kendaraan roda dua diminta untuk mendekatkan kartu cerdas yang dimiliki oleh pengguna. Sesuai dengan penjelasan sebelumnya, kartu cerdas yang dapat digunakan oleh pengguna berupa Kartu Tanda Mahasiswa atau jika tidak ada Kartu Tanda Mahasiswa, pengguna bisa menggunakan Kartu Tanda Penduduk. Contoh kartu cerdas yang dapat digunakan dijelaskan pada Gambar 5.8 berikut.



Gambar 5. 8 Jenis Kartu Cerdas yang Digunakan saat Pengujian

5.2.2.2 Lingkungan Pengujian Kegunaan

Pengujian dilakukan langsung di pos parkir kendaraan roda dua Jurusan Teknik Informatika ITS. Pelaksanaan pengujian dijelaskan pada Tabel 5.10 berikut.

Tabel 5. 10 Pelaksanaan Pengujian

No	Hari/Tanggal	Waktu
1.	Sabtu/21 Juni 2014	12.00 – 18.00
2.	Minggu/22 Juni 2014	15.00 – 18.00
3.	Kamis/26 Juni 2014	09.00 – 12.00

Dalam pengujian kali ini, pengujian dinyatakan berhasil ketika setiap pengguna yang masuk dapat dicatat oleh sistem dan ketika keluar dari tempat parkir dapat diverifikasi dengan benar. Berikut hasil dari pengujian kegunaan aplikasi langsung dari tempat parkir dijelaskan pada Tabel 5.11 berikut.

Tabel 5. 11 Rekapitulasi Pengujian Aplikasi

No	UID Kartu	Nama	Foto User	Keterangan
1	BC7CFF9A	Hani Ramadhan	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\ SaveImage\1ImageBC7CFF9 A.jpg	Berhasil

No	UID Kartu	Nama	Foto User	Keterangan
2	ACA5FB9A	Ainatul Maulida	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\2ImageACA5FB9A.jpg	Berhasil
3	0411582A832980	Orysa Nur Permata S	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\3Image0411582A832980.jpg	Gagal, tidak dapat mengambil gambar dengan baik
4	D6202556	Didik Purwanto	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\4ImageD6202556.jpg	Berhasil
5	D6202556	Didik Purwanto	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\5ImageD6202556.jpg	Gagal, Terjadi penyimpanan pengguna masuk ke tempat parkir ketika pengguna melakukan verifikasi keluar
6	E6C72556	M Chaqiqi Mudafi	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\6ImageE6C72556.jpg	Berhasil
7	06132556	Nabilla	C:\Users\Fahmi\Desktop\TA	Berhasil

No	UID Kartu	Nama	Foto User	Keterangan
		Sabbaha AP	NFC\TA NFC\bin\Debug\ SaveImage\7Image06132556. jpg	
8	ECEEFC9 A	Tirta Taruna R	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\ SaveImage\8ImageECEEFC9 A.jpg	Berhasil
9	048A36D2 302A80	Pengguna	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\ SaveImage\9Image048A36D 2302A80.jpg	Berhasil
10	D6142656	Wiby Mahan F	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\ SaveImage\10ImageD614265 6.jpg	Berhasil
11	AC10FC9A	Hardika Khusnulia	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\ SaveImage\11ImageAC10FC 9A.jpg	Berhasil
12	CC8BFF9A	M Satryo P	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\ SaveImage\12ImageCC8BFF 9A.jpg	Berhasil
13	04223B5A 272980	Ramadhan Satya Putra	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\ SaveImage\13Image04223B5 A272980.jpg	Berhasil

No	UID Kartu	Nama	Foto User	Keterangan
14	9C94FD9A	Rizqi Hidayatullah	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\14Image9C94FD9A.jpg	Berhasil
15	5CA1FC9A	Ines Dwi Andini	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\15Image5CA1FC9A.jpg	Berhasil
17	04306D1A742B80	Antonio Cahyadi L	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\17Image04306D1A742B80.jpg	Berhasil
18	9C64FC9A	Astris Dyah Perwita	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\18Image9C64FC9A.jpg	Berhasil
19	6C35FD9A	Fahmi Hidayatullah	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\19Image6C35FD9A.jpg	Berhasil
20	BC7CFF9A	Hani Ramadhan	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\20ImageBC7CFF9A.jpg	Berhasil
21	6EC4E93A	Mir'atul Mahmudah	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\21Image6EC4E93A.jpg	Berhasil

No	UID Kartu	Nama	Foto User	Keterangan
22	66912556	Wati Marpaung	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\22Image66912556.jpg	Berhasil
23	5CA1FC9A	Ines Dwi Andini	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\23Image5CA1FC9A.jpg	Berhasil
24	BCD4FB9A	Mohammad Faridl Naufal	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\24ImageBCD4FB9A.jpg	Berhasil
25	6C35FD9A	Fahmi Hidayatullah	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\25Image6C35FD9A.jpg	Gagal, Tidak dapat mengambil gambar
26		Abdurrazak Baihaqi	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\26Image.jpg	Gagal, Tidak mendapatkan UID dari kartu
27		Linda Nur Faizah	C:\Users\Fahmi\Desktop\TA NFC\TA NFC\bin\Debug\SaveImage\27Image'.jpg	Gagal, Tidak mendapatkan UID dari kartu

Berdasarkan Tabel 5.11 didapatkan hasil bahwa dari 26 pengujian berhasil melakukan penyimpanan ketika orang dengan kendaraan roda dua akan masuk ke dalam tempat parkir dan kemudian keluar dari tempat parkir. Terdapat beberapa pengujian

yang tidak berhasil dikarenakan kesalahan dari pengguna ketika mendekatkan kartu cerdas ke alat.

5.2.3 Pengujian Performa

Pengujian performa ini dilakukan untuk mengetahui kemampuan atau kecepatan (waktu) yang dimiliki oleh perangkat lunak dalam beberapa proses. Pada pengujian ini, dilakukan pengujian performa setiap pembacaan data dari kartu cerdas oleh alat, performa setiap transaksi masuk ke dalam tempat parkir, dan performa ketika akan keluar dari tempat parkir.

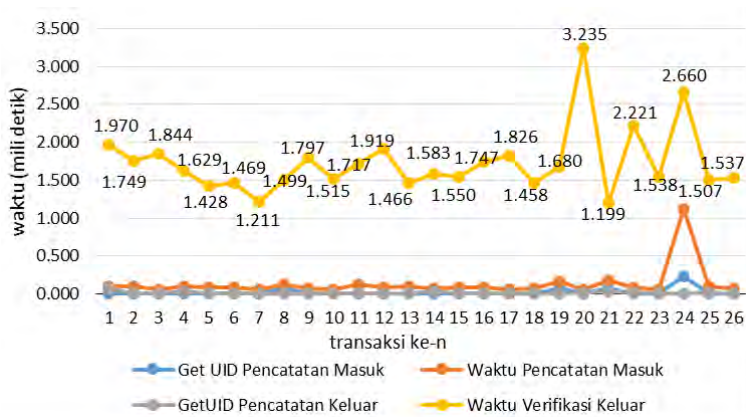
Pada pengujian ini, dibuatlah sebuah basis data baru ke dalam sistem untuk menampung kecepatan setiap kondisi pengujian yang telah dijelaskan sebelumnya. Tabel basis data Pengujian dijelaskan pada Tabel 5.12 berikut.

Tabel 5. 12 Tabel Basis Data Pengujian

Nama Kolom	Tipe Data	Keterangan
Id_uji	Bigint (20)	<i>Primary Key</i> dari tabel Pengujian
Id_trans	Bigint (20)	<i>Foreign Key</i> ID Transaksi yang dimiliki oleh Tabel Transaksi
Time_uid_insert	Double	Waktu yang dibutuhkan ketika mencatat kendaraan masuk yang dihitung sejak kartu didekatkan ke alat pembaca sampai didapatkannya data dari kartu cerdas.
Time_insert	Double	Waktu yang dibutuhkan dari mendapatkan data yang ada di kartu cerdas sampai pencatatan kendaraan masuk berhasil tersimpan di dalam basis data
Time_uid_update	Double	Waktu yang dibutuhkan ketika melakukan verifikasi kendaraan

Nama Kolom	Type Data	Keterangan
		keluar yang dihitung sejak kartu didekatkan ke alat pembaca sampai didapatkannya data dari kartu cerdas.
Time_update	Double	Waktu yang dibutuhkan dari mendapatkan data yang ada di kartu cerdas sampai proses verifikasi kendaraan berhasil dilakukan dan perubahan data berhasil disimpan di dalam basis data

Dan dari setiap pengujian kegunaan yang sudah dilakukan didapatkan performa waktu yang dibutuhkan setiap proses yang dilakukan. Hasil rekapitulasi pengujian performa yang akan dijelaskan dalam bentuk grafik dari masing-masing jenis pengujian performa yang dilakukan digambarkan pada Gambar 5.9 dan Tabel 5.13 berikut.



Gambar 5. 9 Grafik Hasil Pengujian Performa (dalam milidetik)

Tabel 5. 13 Hasil Pengujian Performa (dalam detik)

ID Uji	ID Trans	Time_UID_Insert	Time_Insert	Time_UID_Update	Time_Update
1	1	0,003	0,106	0,076	1,970
2	2	0,003	0,102	0,004	1,749
3	3	0,004	0,064	0,003	1,844
4	4	0,004	0,105	0,034	1,629
5	5	0,003	0,088	0,003	1,428
6	6	0,003	0,087	0,003	1,469
7	7	0,002	0,065	0,004	1,211
8	8	0,077	0,131	0,003	1,499
9	9	0,003	0,078	0,003	1,797
10	10	0,002	0,062	0,003	1,515
11	11	0,003	0,132	0,003	1,717
12	12	0,005	0,082	0,004	1,919
13	13	0,003	0,101	0,003	1,466
14	14	0,003	0,077	0,036	1,583
15	15	0,004	0,085	0,004	1,550
17	17	0,002	0,081	0,005	1,747
18	18	0,002	0,063	0,004	1,826
19	19	0,003	0,076	0,005	1,458
20	20	0,078	0,160	0,003	1,680
21	21	0,004	0,060	0,018	3,235
22	22	0,068	0,181	0,034	1,199
23	23	0,003	0,092	0,015	2,221
24	24	0,003	0,065	0,004	1,538
25	25	0,235	1,126	0,005	2,660
26	26	0,002	0,099	0,022	1,507
27	27	0,003	0,079	0,003	1,537

Berdasarkan Tabel 5.13 bahwa waktu untuk mendapatkan data dari kartu cerdas diambil dari waktu rata-rata `time_uid_insert` dan `time_uid_update` adalah 0,0159 detik atau 15,9 milidetik. Sedangkan waktu yang diperlukan untuk melakukan pencatatan

kendaraan masuk parkir adalah 0,1326 detik atau 132,6 milidetik. Sedangkan waktu yang diperlukan untuk melakukan verifikasi kendaraan yang akan keluar tempat parkir adalah 1,729 detik.

5.3. Evaluasi Pengujian

Berdasarkan hasil pengujian fungsionalitas, maka dapat dilakukan evaluasi sebagai berikut.

5.3.1 Evaluasi Pengujian Fungsionalitas

Pengujian fungsionalitas yang telah dilakukan memberikan hasil yang sesuai dengan skenario yang telah direncanakan. Evaluasi pengujian pada masing-masing fungsionalitas dijelaskan sebagai berikut.

1. Pengujian mengelola data pengguna dengan menambah data pengguna baru sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.001 yang memberikan informasi bahwa proses menambahkan data pengguna baru berjalan dengan benar.
2. Pengujian mengelola data pengguna dengan mengubah data pengguna sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.002 yang memberikan informasi bahwa pengubahan data pengguna berhasil dilakukan dan berjalan dengan benar.
3. Pengujian mengelola data pengguna dengan menghapus data pengguna sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.003 yang memberikan informasi bahwa penghapusan data pengguna berhasil dilakukan dan berjalan dengan benar.
4. Pengujian mengelola rekapitulasi keluar masuk kendaraan dengan mengubah data transaksi sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.004 yang memberikan informasi bahwa pengubahan data transakis berhasil dilakukan

5. Pengujian mengelola rekapitulasi keluar masuk kendaraan dengan melakukan penghapusan data transaksi berhasil sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.005 yang memberikan informasi bahwa pengujian penghapusan data transaksi keluar masuk kendaraan berhasil dilakukan dan berjalan dengan benar.
6. Pengujian mencatat kendaraan masuk parkir sudah sesuai dengan apa yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.006 yang memperlihatkan bahwa pengujian pencatatan kendaraan masuk berhasil dilakukan dan berjalan dengan benar.
7. Pengujian melakukan verifikasi kendaraan keluar parkir sudah sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.007 yang memperlihatkan bahwa pengujian melakukan verifikasi ketika kendaraan keluar parkir berhasil dilakukan dan berjalan dengan benar.
8. Pengujian melihat transaksi keluar masuk kendaraan yang dilakukan oleh Petugas Parkir sudah sesuai dengan yang diharapkan. Kondisi ini diperlihatkan pada pengujian PUJ.UC.008 yang menunjukkan bahwa Petugas Parkir dapat melihat transaksi keluar masuk kendaraan dan proses pengujian berjalan dengan benar.

5.3.2 Evaluasi Pengujian Kegunaan

Pengujian kegunaan yang telah dilakukan langsung di tempat parkir Jurusan Teknik Informatika memberikan hasil yang diharapkan. Namun beberapa kegagalan yang terjadi dikarenakan ketidaktahuan pengguna yang masuk terhadap perangkat lunak yang diuji coba. Sehingga dilakukan perbaikan pada perangkat lunak yang dapat menangani permasalahan-permasalahan setelah pengujian ini dilaksanakan. Agar kesalahan-kesalahan yang telah terjadi ketika pengujian tidak terjadi lagi.

BAB VI PENUTUP

Pada bab ini akan dijelaskan kesimpulan yang dapat diambil dalam pengerjaan Tugas Akhir dan saran tentang kemungkinan pengembangan lebih lanjut dari aplikasi Sistem Keamanan Kampus yang dapat dilakukan di masa yang akan datang.

6.1. Kesimpulan

Dalam proses pengerjaan Tugas Akhir dari tahap pendahuluan, kajian pustaka, analisis, perancangan, implementasi dan pengujian aplikasi Sistem Keamanan Kampus ini diperoleh kesimpulan sebagai berikut.

1. Aplikasi Sistem Keamanan Kampus yang dibangun berbasis Desktop dengan menggunakan bahasa pemrograman C# dan SDK dari alat pembaca NFC (ACR122U) dapat melakukan pencatatan orang yang masuk ke tempat parkir dengan kendaraan roda dua, dan dapat diverifikasi ketika pengguna itu akan keluar dari tempat parkir. Diharapkan dengan adanya identifikasi orang dengan kendaraan tersebut, tingkat keamanan kendaraan di ITS dapat ditingkatkan.
2. Aplikasi Sistem Keamanan Kampus ini dapat melakukan pembacaan dan mendapatkan data yang tersimpan di dalam kartu cerdas jenis Mifare dan jenis kartu cerdas lain yang mendukung ISO 14443 tipe A (E-KTP) dengan kecepatan pembacaan data rata-rata adalah 15,9 milidetik.
3. Aplikasi Sistem Keamanan Kampus dapat menyimpan dan melakukan pencatatan orang yang masuk dengan kendaraan, serta mencatat waktu kejadian pengguna masuk

ke dalam tempat parkir dan mengambil gambar pengguna dengan kecepatan rata-rata adalah 132,6 milidetik.

4. Rata-rata waktu yang dibutuhkan setiap verifikasi pengguna yang akan keluar dari tempat parkir adalah 1.729 detik. Waktu ini sudah termasuk waktu yang dibutuhkan Petugas Parkir memastikan bahwa pengguna yang keluar itu benar dengan kondisi ketika pengguna masuk.
5. Aplikasi Sistem Keamanan Kampus dapat melakukan monitoring terhadap pengguna yang telah masuk di tempat parkir atau telah keluar dari tempat parkir.
6. Aplikasi Sistem Keamanan Kampus dapat memanfaatkan Kartu Tanda Mahasiswa ITS dan/atau Kartu Tanda Penduduk sebagai kartu cerdas.

6.2. Saran

Berikut ini beberapa saran untuk kemungkinan pengembangan lebih lanjut aplikasi Sistem Keamanan Kampus di masa mendatang berdasarkan hasil rancangan, implementasi, dan uji coba yang telah dilakukan.

1. Pengambilan gambar hanya sebatas mengambil foto saja. Dan Verifikasi yang dilakukan oleh Petugas Parkir dilakukan secara manual. Aplikasi tidak dapat mendeteksi dan melakukan pencocokan secara langsung orang yang masuk atau keluar. Sehingga memungkinkan pengembangan sistem menuju pengolahan pengenalan citra wajah atau pengenalan nomor polisi kendaraan.
2. Pembuatan aplikasi yang dapat terhubung dengan alat bantu berupa pintu otomatis.

DAFTAR PUSTAKA

- [1] Muhammad, Perancangan dan Implementasi Perangkat Lunak Untuk Identifikasi dan Manajemen Kendaraan di ITS Pada Perangkat Bergerak dengan Teknologi QR Code, Surabaya, 2013.
- [2] P. Roger S, "Blackbox Testing," in *Software Engineering A Practitioner's Approach 7th Edition*, New York, McGraw-Hill, 2010, p. 495.
- [3] R. ITS, "Profil ITS," [Online]. Available: <https://www.its.ac.id/article/profil-its/id>. [Diakses 4 Maret 2014].
- [4] I. Sommerville, *Software Engineering 9th Edition*, United states of America: Pearson Education, Inc, 2011.
- [5] N. Forum, "NFC and Contactless Technologies," [Online]. Available: <http://nfc-forum.org/what-is-nfc/about-the-technology/>. [Diakses Februari 2014].
- [6] J. Satria, "NFC, Teknologi Baru Pengubah Gaya Hidup," 16 Januari 2012. [Online]. Available: <http://intisari-online.com/read/nfc-teknologi-baru-pengubah-gaya-hidup>. [Diakses 2 Maret 2014].
- [7] Advanced Card System Ltd., "ACR 122U USB NFC Reader," 2013. [Online]. Available: <http://www.acs.com.hk/en/products/3/acr122u-usb-nfc-reader>. [Diakses Januari 2014].
- [8] bahamutsky, "Artikel Mifare," [Online]. Available: <http://www.id.scribd.com/doc/62618350/Artiket-MIFARE>. [Diakses Maret 2014].

- [9] W. Hastomo, "Pengertian dan Kelebihan Database MySQL," Hastomo Network, 2012. [Online]. [Diakses Maret 2014].
- [10] J. Simon, "What is an APDU," JGuru, 07 May 2012. [Online]. Available: <http://www.jguru.com/faq/view.jsp?EID=470744>. [Diakses 22 June 2014].
- [11] Pongsakorn, "WebCam Library for WinForm and WPF with C# and VB.NET," CodePlex, 13 December 2009. [Online]. Available: <http://easywebcam.codeplex.com>. [Accessed 22 June 2014].

BIODATA PENULIS



Penulis lahir di Surabaya, 31 Maret 1992. Merupakan anak kedua dari empat bersaudara. Penulis menempuh pendidikan di SD Muhammadiyah 1 Banyuwangi, SMP Negeri 2 Banyuwangi, dan SMA Negeri 1 Glagah, Banyuwangi. Penulis melanjutkan pendidikan sarjana di Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya. Dalam menyelesaikan pendidikan S1, penulis mengambil bidang minat Rekayasa Perangkat Lunak (*Software Engineering*). Selama kuliah penulis aktif dalam berbagai kegiatan dan organisasi. Di bidang akademik, Penulis beberapa kali aktif sebagai asisten praktikum sistem digital, asisten dosen mata kuliah Grafika Komputer, Pemrograman Framework.NET, Excel untuk Bisnis (PIKTI). Pernah aktif juga menjadi Administrator Laboratorium Pengembangan Game.

Di bidang non-akademik, penulis pernah menjadi Menteri Dalam Negeri BEM ITS 2013-2014. Sebelumnya juga menjadi Direktur Jenderal Humas Internal Kementerian Dalam Negeri BEM ITS periode 2012-2013. Penulis juga aktif menjadi seorang Pemandu LKMM, staff Departemen Ristek HMTC, staff Departemen Humas Keluarga Muslim Informatika dan Ketua Divisi Kepemanduan BEM FTIF pada kepengurusan 2011 hingga 2013. Dan beberapa kegiatan social di luar kampus. Penulis memiliki ketertarikan dalam pengembangan aplikasi perangkat bergerak, SQA, basis data, dan *software evolution*. Penulis dapat dihubungi melalui alamat surel fahmihidayatullah31@gmail.com.

LAMPIRAN A – KODE SUMBER

```

public class ModWinsCard
{
    [StructLayout(LayoutKind.Sequential)]
    public struct SCARD_IO_REQUEST
    {
        public int dwProtocol;
        public int cbPciLength;
    }
    [StructLayout(LayoutKind.Sequential)]
    public struct APDUREC
    {
        public byte bCLA;
        public byte bINS;
        public byte bP1;
        public byte bP2;
        public byte bP3;
        [MarshalAs(UnmanagedType.ByValArray,
SizeConst=256)]
        public byte[] Data;
        [MarshalAs(UnmanagedType.ByValArray,
SizeConst=3)]
        public byte[] SW;
        public bool IsSend;
    }
    [StructLayout(LayoutKind.Sequential)]
    public struct SCARD_READERSTATE
    {
        public string RdrName;
        public int UserData;
        public int RdrCurrState;
        public int RdrEventState;
        public int ATRLength;
        [MarshalAs(UnmanagedType.ByValArray,
SizeConst=37)]
        public byte ATRValue;
    }
}

```

```

public const int SCARD_S_SUCCESS = 0;
public const int SCARD_ATR_LENGTH = 33;
public const int CT_MCU = 0x00;
// MCU
public const int CT_IIC_Auto = 0x01;
// IIC (Auto Detect Memory Size)
public const int CT_IIC_1K = 0x02;
public const int CT_IIC_2K = 0x03;
public const int CT_IIC_4K = 0x04;
public const int CT_IIC_8K = 0x05;
public const int CT_IIC_16K = 0x06;
public const int CT_IIC_32K = 0x07;
public const int CT_IIC_64K = 0x08;
public const int CT_IIC_128K = 0x09;
public const int CT_IIC_256K = 0x0A;
public const int CT_IIC_512K = 0x0B;
public const int CT_IIC_1024K = 0x0C;
public const int CT_AT88SC153 = 0x0D;
public const int CT_AT88SC1608 = 0x0E;
public const int CT_SLE4418 = 0x0F;
public const int CT_SLE4428 = 0x10;
public const int CT_SLE4432 = 0x11;
public const int CT_SLE4442 = 0x12;
public const int CT_SLE4406 = 0x13;
public const int CT_SLE4436 = 0x14;
public const int CT_SLE5536 = 0x15;
public const int CT_MCU0 = 0x16;
public const int CT_MCU1 = 0x17;
public const int CT_MCU_Auto = 0x18;

public const int SCARD_SCOPE_USER = 0;
public const int SCARD_SCOPE_TERMINAL = 1;
public const int SCARD_SCOPE_SYSTEM = 2;

public const int SCARD_STATE_UNAWARE = 0x00;
public const int SCARD_STATE_IGNORE = 0x01;
public const int SCARD_STATE_CHANGED = 0x02;
public const int SCARD_STATE_UNKNOWN = 0x04;
public const int SCARD_STATE_UNAVAILABLE =

```

```

0x08;
    public const int SCARD_STATE_EMPTY = 0x10;
    public const int SCARD_STATE_PRESENT = 0x20;
    public const int SCARD_STATE_ATRMATCH = 0x40;
    public const int SCARD_STATE_EXCLUSIVE =
0x80;
    public const int SCARD_STATE_INUSE = 0x100;
    public const int SCARD_STATE_MUTE = 0x200;
    public const int SCARD_STATE_UNPOWERED =
0x400;
    public const int SCARD_SHARE_EXCLUSIVE = 1;
    public const int SCARD_SHARE_SHARED = 2;
    public const int SCARD_SHARE_DIRECT = 3;
    public const int SCARD_LEAVE_CARD = 0;
    public const int SCARD_RESET_CARD = 1;
    public const int SCARD_UNPOWER_CARD = 2;
    public const int SCARD_EJECT_CARD = 3;

    public const long FILE_DEVICE_SMARTCARD =
0x310000; // Reader action IOCTLs
    public const long IOCTL_SMARTCARD_DIRECT =
FILE_DEVICE_SMARTCARD + 2050 * 4;
    public const long IOCTL_SMARTCARD_SELECT_SLOT
= FILE_DEVICE_SMARTCARD + 2051 * 4;
    public const long IOCTL_SMARTCARD_DRAW_LCDBMP
= FILE_DEVICE_SMARTCARD + 2052 * 4;
    public const long IOCTL_SMARTCARD_DISPLAY_LCD
= FILE_DEVICE_SMARTCARD + 2053 * 4;
    public const long IOCTL_SMARTCARD_CLR_LCD =
FILE_DEVICE_SMARTCARD + 2054 * 4;
    public const long IOCTL_SMARTCARD_READ_KEYPAD
= FILE_DEVICE_SMARTCARD + 2055 * 4;
    public const long IOCTL_SMARTCARD_READ_RTC =
FILE_DEVICE_SMARTCARD + 2057 * 4;
    public const long IOCTL_SMARTCARD_SET_RTC =
FILE_DEVICE_SMARTCARD + 2058 * 4;
    public const long IOCTL_SMARTCARD_SET_OPTION
= FILE_DEVICE_SMARTCARD + 2059 * 4;
    public const long IOCTL_SMARTCARD_SET_LED =

```



```

FILE_DEVICE_SMARTCARD + 2060 * 4;
    public const long IOCTL_SMARTCARD_LOAD_KEY =
FILE_DEVICE_SMARTCARD + 2062 * 4;
    public const long IOCTL_SMARTCARD_READ_EEPROM
= FILE_DEVICE_SMARTCARD + 2065 * 4;
    public const long
IOCTL_SMARTCARD_WRITE_EEPROM =
FILE_DEVICE_SMARTCARD + 2066 * 4;
    public const long IOCTL_SMARTCARD_GET_VERSION
= FILE_DEVICE_SMARTCARD + 2067 * 4;
    public const long
IOCTL_SMARTCARD_GET_READER_INFO =
FILE_DEVICE_SMARTCARD + 2051 * 4;
    public const long
IOCTL_SMARTCARD_SET_CARD_TYPE =
FILE_DEVICE_SMARTCARD + 2060 * 4;
    public const long
IOCTL_SMARTCARD_ACR128_ESCAPE_COMMAND =
FILE_DEVICE_SMARTCARD + 2079 * 4;
    public const int SCARD_F_INTERNAL_ERROR =
-2146435071;
    public const int SCARD_E_CANCELLED =
-2146435070;
    public const int SCARD_E_INVALID_HANDLE =
-2146435069;
    public const int SCARD_E_INVALID_PARAMETER =
-2146435068;
    public const int SCARD_E_INVALID_TARGET =
-2146435067;
    public const int SCARD_E_NO_MEMORY =
-2146435066;
    public const int SCARD_F_WAITED_TOO_LONG =
-2146435065;
    public const int SCARD_E_INSUFFICIENT_BUFFER
= -2146435064;
    public const int SCARD_E_UNKNOWN_READER
= -2146435063;
    public const int SCARD_E_TIMEOUT =
-2146435062;

```

```

public const int SCARD_E_SHARING_VIOLATION =
-2146435061;
public const int SCARD_E_NO_SMARTCARD =
-2146435060;
public const int SCARD_E_UNKNOWN_CARD =
-2146435059;
public const int SCARD_E_CANT_DISPOSE =
-2146435058;
public const int SCARD_E_PROTO_MISMATCH =
-2146435057;

public const int SCARD_E_NOT_READY =
-2146435056;
public const int SCARD_E_INVALID_VALUE =
-2146435055;
public const int SCARD_E_SYSTEM_CANCELLED =
-2146435054;
public const int SCARD_F_COMM_ERROR =
-2146435053;
public const int SCARD_F_UNKNOWN_ERROR =
-2146435052;
public const int SCARD_E_INVALID_ATR =
-2146435051;
public const int SCARD_E_NOT_TRANSACTED =
-2146435050;
public const int SCARD_E_READER_UNAVAILABLE =
-2146435049;
public const int SCARD_P_SHUTDOWN =
-2146435048;
public const int SCARD_E_PCI_TOO_SMALL =
-2146435047;

public const int SCARD_E_READER_UNSUPPORTED =
-2146435046;
public const int SCARD_E_DUPLICATE_READER =
-2146435045;
public const int SCARD_E_CARD_UNSUPPORTED =
-2146435044;
public const int SCARD_E_NO_SERVICE =

```

```

-2146435043;
    public const int SCARD_E_SERVICE_STOPPED      =
-2146435042;
    public const int SCARD_W_UNSUPPORTED_CARD    =
-2146435041;
    public const int SCARD_W_UNRESPONSIVE_CARD   =
-2146435040;
    public const int SCARD_W_UNPOWERED_CARD     =
-2146435039;
    public const int SCARD_W_RESET_CARD         =
-2146435038;
    public const int SCARD_W_REMOVED_CARD       =
-2146435037;

    public const int SCARD_PROTOCOL_UNDEFINED =
0x00;          // There is no active protocol.
    public const int SCARD_PROTOCOL_T0 = 0x01;
// T=0 is the active protocol.
    public const int SCARD_PROTOCOL_T1 = 0x02;
// T=1 is the active protocol.
    public const int SCARD_PROTOCOL_RAW =
0x10000;      // Raw is the active protocol.
    public const int SCARD_UNKNOWN = 0;
    public const int SCARD_ABSENT = 1;
    public const int SCARD_PRESENT = 2;
    public const int SCARD_SWALLOWED = 3;
    public const int SCARD_POWERED = 4;
    public const int SCARD_NEGOTIABLE = 5;
    public const int SCARD_SPECIFIC = 6;

[DllImport("winscard.dll")]
    public static extern int
SCardEstablishContext(int dwScope, int
pvReserved1, int pvReserved2, ref int
phContext);

[DllImport("winscard.dll")]
    public static extern int
SCardReleaseContext(int phContext);

```

```

    [DllImport("winscard.dll")]
    public static extern int SCardConnect(int
hContext, string szReaderName, int dwShareMode,
int dwPrefProtocol, ref int phCard, ref int
ActiveProtocol);

    [DllImport("winscard.dll")]
    public static extern int
SCardBeginTransaction (int hCard);

    [DllImport("winscard.dll")]
    public static extern int SCardDisconnect(int
hCard, int Disposition);

    [DllImport("winscard.dll")]
    public static extern int
SCardListReaderGroups(int hContext, ref string
mzGroups, ref int pcchGroups);

    [DllImport("winscard.DLL",
EntryPoint="SCardListReadersA",
CharSet=CharSet.Ansi)]
    public static extern int SCardListReaders(
int hContext, byte[] Groups, byte[] Readers,
ref int pcchReaders);

    [DllImport("winscard.dll")]
    public static extern int SCardStatus(int
hCard, string szReaderName, ref int
pcchReaderLen, ref int State, ref int Protocol,
ref byte ATR, ref int ATRLen);

    [DllImport("winscard.dll")]
    public static extern int SCardEndTransaction
(int hCard, int Disposition);

    [DllImport("winscard.dll")]
    public static extern int SCardState (int

```

```

hCard, ref uint State, ref uint Protocol, ref
byte ATR, ref uint ATRLen);

[DllImport("winscard.dll")]
public static extern int SCardTransmit (int
hCard, ref SCARD_IO_REQUEST pioSendRequest, ref
byte SendBuff, int SendBuffLen, ref
SCARD_IO_REQUEST pioRecvRequest, ref byte
RecvBuff, ref int RecvBuffLen);

public static string GetScardErrMsg(int
ReturnCode)
{
    switch(ReturnCode)
    {
        case SCARD_E_CANCELLED:
            return("The action was canceled by an
SCardCancel request.");
        case SCARD_E_CANT_DISPOSE:
            return("The system could not dispose of
the media in the requested manner.");
        case SCARD_E_CARD_UNSUPPORTED:
            return("The smart card does not meet
minimal requirements for support.");
        case SCARD_E_DUPLICATE_READER:
            return("The reader driver didn't produce
a unique reader name.");
        case SCARD_E_INSUFFICIENT_BUFFER:
            return("The data buffer for returned data
is too small for the returned data.");
        case SCARD_E_INVALID_ATR:
            return("An ATR string obtained from the
registry is not a valid ATR string.");
        case SCARD_E_INVALID_HANDLE:
            return("The supplied handle was
invalid.");
        case SCARD_E_INVALID_PARAMETER:
            return("One or more of the supplied
parameters could not be properly interpreted.");
    }
}

```

```

    case SCARD_E_INVALID_TARGET:
    return("Registry startup information is
missing or invalid.");
    case SCARD_E_INVALID_VALUE:
    return("One or more of the supplied
parameter values could not be properly
interpreted.");
    case SCARD_E_NOT_READY:
    return("The reader or card is not ready
to accept commands.");
    case SCARD_E_NOT_TRANSACTED:
    return("An attempt was made to end a
non-existent transaction.");
    case SCARD_E_NO_MEMORY:
    return("Not enough memory available to
complete this command.");
    case SCARD_E_NO_SERVICE:
    return("The smart card resource manager
is not running.");
    case SCARD_E_NO_SMARTCARD:
    return("The operation requires a smart
card, but no smart card is currently in the
device.");
    case SCARD_E_PCI_TOO_SMALL:
    return("The PCI receive buffer was too
small.");
    case SCARD_E_PROTO_MISMATCH:
    return("The requested protocols are
incompatible with the protocol currently in use
with the card.");
    case SCARD_E_READER_UNAVAILABLE:
    return("The specified reader is not
currently available for use.");
    case SCARD_E_READER_UNSUPPORTED:
    return("The reader driver does not meet
minimal requirements for support.");
    case SCARD_E_SERVICE_STOPPED:
    return("The smart card resource manager
has shut down.");
```

```

    case SCARD_E_SHARING_VIOLATION:
    return("The smart card cannot be accessed
because of other outstanding connections.");
    case SCARD_E_SYSTEM_CANCELLED:
    return("The action was canceled by the
system, presumably to log off or shut down.");
    case SCARD_E_TIMEOUT:
    return("The user-specified timeout value
has expired.");
    case SCARD_E_UNKNOWN_CARD:
    return("The specified smart card name is
not recognized.");
    case SCARD_E_UNKNOWN_READER:
    return("The specified reader name is not
recognized.");
    case SCARD_F_COMM_ERROR:
    return("An internal communications error
has been detected.");
    case SCARD_F_INTERNAL_ERROR:
    return("An internal consistency check
failed.");
    case SCARD_F_UNKNOWN_ERROR:
    return("An internal error has been
detected, but the source is unknown.");
    case SCARD_F_WAITED_TOO_LONG:
    return("An internal consistency timer has
expired.");
    case SCARD_S_SUCCESS:
    return("No error was encountered.");
    case SCARD_W_REMOVED_CARD:
    return ("The smart card has been removed,
so that further communication is not
possible.");
    case SCARD_W_RESET_CARD:
    return("The smart card has been reset, so
any shared state information is invalid.");
    case SCARD_W_UNPOWERED_CARD:
    return("Power has been removed from the
smart card, so that further communication is not




```







```
possible.");  
    case SCARD_W_UNRESPONSIVE_CARD:  
        return("The smart card is not responding  
to a reset.");  
    case SCARD_W_UNSUPPORTED_CARD:  
        return("The reader cannot communicate  
with the card, due to ATR string configuration  
conflicts.");  
    default:  
        return("?");  
    }  
}
```







Kode Sumber A. 1 Implementasi API ACR122U






LAMPIRAN B – HASIL UJI COBA







Tabel B. 1 Hasil Pengujian

No	UID Kartu	Waktu Masuk	Tanggal Masuk	Waktu Keluar	Tanggal Keluar	Status	Foto User	Foto Keluar
1	BC7CFF9A	12:58:55	2014-06-21	12:59:24	2014-06-21	Keluar		Null
2	ACA5FB9A	01:00:36	2014-06-21	12:39:38	2014-06-24	Keluar		Null
3	0411582A832980	01:05:31	2014-06-21	04:22:13	2014-06-21	Keluar		Null

4	D6202556	01:40:17	2014-06-21	01:40:37	2014-06-21	Keluar		Null
5	D6202556	01:40:39	2014-06-21	01:40:48	2014-06-21	Keluar		Null
6	E6C72556	01:45:19	2014-06-21	01:46:27	2014-06-21	Keluar		Null
7	06132556	01:49:15	2014-06-21	01:55:00	2014-06-21	Keluar		Null
8	ECEEFC9A	01:55:28	2014-06-21	10:17:00	2014-06-26	Keluar		Null
9	048A36D2302A80	02:06:58	2014-06-21	05:12:11	2014-07-21	Keluar		Null

10	D6142656	02:10:05	2014-06-21	07:01:43	2014-07-21	Keluar		Null
11	AC10FC9A	02:39:23	2014-06-21	06:31:14	2014-07-21	Keluar		Null
12	CC8BFF9A	02:55:59	2014-06-21	03:49:30	2014-06-21	Keluar		Null
13	04223B5A272980	03:49:05	2014-06-21	11:27:16	2014-06-25	Keluar		Null
14	9C94FD9A	03:54:07	2014-06-21	03:54:53	2014-06-21	Keluar		Null
15	5CA1FC9A	04:30:09	2014-06-21	04:31:13	2014-06-21	Keluar		Null

16	04306D1A742B80	05:00:26	2014-06-22	09:01:31	2014-07-22	Keluar		Null
17	9C64FC9A	05:09:30	2014-06-22	10:14:42	2014-07-22	Keluar		Null
18	6C35FD9A	05:16:21	2014-06-22	05:16:44	2014-06-22	Keluar		Null
19	BC7CFF9A	05:53:33	2014-06-22	08:30:21	2014-06-22	Keluar		Null
20	6EC4E93A	10:47:03	2014-06-26	13:11:14	2014-07-26	Keluar		Null

21	66912556	10:57:22	2014-06-26	15:04:11	2014-07-26	Keluar		Null
22	5CA1FC9A	11:05:35	2014-06-26	11:05:38	2014-06-26	Keluar		Null
23	BCD4FB9A	11:05:41	2014-06-26	11:05:41	2014-06-26	Keluar		Null
24	6C35FD9A	11:10:43	2014-06-26	23:57:58	2014-06-26	Keluar		Null
25		11:11:17	2014-07-01	16:03:45	2014-07-01	Keluar		Null
26	'	11:13:44	2014-07-01	14:23:56	2014-07-01	Keluar		Null