

SKRIPSI

**PERANCANGAN PROTOTYPE PRESENSI MENGGUNAKAN
E-KTM BERBASIS RADIO FREQUENCY
IDENTIFICATION**



OLEH :

RIANTONO HARTAKU

DBC 115 070

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS PALANGKA RAYA

2020

SKRIPSI

**PERANCANGAN PROTOTYPE PRESENSI MENGGUNAKAN
E-KTM BERBASIS RADIO FREQUENCY
IDENTIFICATION**

Sebagai salah satu syarat menyelesaikan Program Strata I
pada Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya

OLEH :

RIANTONO HARTAKU
NIM. DBC 115 070

Disetujui untuk diajukan dalam Seminar Akhir Skripsi.

Pembimbing I


RONY TEGUH, S.Kom., M.T., Ph.D
NIP. 19760624 200501 1 015

Pembimbing II


NAHUMI NUGRAHANINGSIH, ST., MT., Ph.D
NIP. 19791009 200801 2 016

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA**

2020

**PERANCANGAN PROTOTYPE PRESENSI MENGGUNAKAN
E-KTM BERBASIS RADIO FREQUENCY
IDENTIFICATION**

SKRIPSI

Sebagai salah satu syarat untuk menyelesaikan Program Strata-1 pada Jurusan Teknik
Informatika Fakultas Teknik Universitas Palangka Raya

OLEH :

RIANTONO HARTAKU

DBC 115 070

Telah dipertahankan di depan tim penguji, pada :

Hari/Tanggal : Kamis, 04 Juni 2020

Waktu : 13:00 – 14:30

1. VIKTOR H. PRANATAWILAYA, S.T., MT
NIP. 19810606 200501 1 001 (Ketua)
2. ABERTUN SAGIT SAHAY, ST., M.Eng
NIP. 19751212 200312 1 002 (Anggota)
3. ARIESTA LESTARI, S.Kom., M.Cs., Ph.D
NIP. 19800322 200501 2 004 (Anggota)
4. RONY TEGUH, S.Kom., M.T., Ph.D
NIP. 19760624 200501 1 015 (Anggota)
5. NAHUMI NUGRAHANINGSIH, ST., MT., Ph.D
NIP. 19791009 200801 2 016 (Anggota)

Mengetahui :

Fakultas Teknik
Universitas Palangka Raya
Teknik

Ir. WALLYO SUDWANTORO, M.T.
NIP. 19631119 199302 1 001

Jurusan / Program Studi Teknik Informatika
Fakultas Teknik Universitas Palangka Raya
Ketua Jurusan,


ABERTUN SAGIT SAHAY, S.T., M.Eng
NIP. 19751212 200312 1 002

PERNYATAAN

Dengan ini saya menyatakan dengan sebenar - benarnya bahwa dalam Skripsi ini tidak terdapat karya ilmiah yang pernah diajukan untuk memperoleh gelar kesarjanaan disuatu Perguruan Tinggi, serta tidak terdapat karya ilmiah atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam Skripsi ini dan disebutkan dalam Tinjauan Pustaka.

Palangka Raya, Juni 2020

RIANTONO HARTAKU
DBC 115 070

RIWAYAT PENYUSUN

Data Diri

Nama : RIANTONO HARTAKU
NIM : DBC 115 070
Fakultas : Teknik
Jurusan/Program Studi : Teknik Informatika
Jenjang : Strata 1 (S-1)
Jenis Kelamin : LAKI-LAKI
Tempat, Tanggal Lahir : LENGGANG, 30 MEI 1996
Agama : KRISTEN
Status dalam Keluarga : Anak
Anak ke - : 3 (Tiga)
Alamat : Jl. Bukit Raya XV C, NO.1, Palangka Raya
No. Telpon/HP : -



Data Orang Tua

Nama Ayah : LUKMAN RUBIN
Pekerjaan Ayah : Wiraswasta
Nama Ibu : ECENI SALEN
Pekerjaan Ibu : Ibu Rumah Tangga
Alamat Orang Tua : Gg. Sudimampir, Ampah, Barito Timur
No. Telpon/HP : -

Riwayat Pendidikan *)

SD : SDN 4 Dusun Tengah (Tahun Lulus 2007)
SMP : SMPN 1 Dusun Tengah (Tahun Lulus 2010)
SMA : SMAN 1 Dusun Tengah (Tahun Lulus 2013)

Palangka Raya, Juni 2020

RIANTONO HARTAKU
DBC 115 070

HALAMAN PERSEMBAHAN

“Only God Can Judge Me”

Terimakasih kepada TUHAN YESUS KRISTUS yang telah mencurahkan kasih karunia-Nya kepada saya, sehingga saya berhasil menyelesaikan studi dan laporan Tugas Akhir ini.

Kupersembahkan skripsi ini untuk Ayahanda Lukman Rubin dan Ibunda Eceni Salen terkasih, tercinta dan terhormat.

Serta istri tercinta yang selalu senantiasa mendoakan dan menyemangati saya dalam setiap proses yang saya lalui.

... Terima Kasih Untuk Semuanya ...

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan kasih karunia-Nya maka Laporan Skripsi/Tugas Akhir ini dapat dikerjakan dengan baik dan diselesaikan tepat pada waktunya.

Adapun judul dari Skripsi/Tugas Akhir yang diambil yaitu “*Perancangan Prototype Presensi Menggunakan E-KTM Berbasis Radio Frequency Identification*”. Tujuan dari pengambilan judul ini diharapkan kedepannya dapat membantu pencatatan daftar hadir agar lebih mudah dan cepat bagi pengguna.

Penulis juga menyampaikan ucapan terimakasih kepada semua pihak yang telah membantu dalam menyelesaikan tugas ini, khususnya kepada:

1. Tuhan Yesus Kristus yang telah mencurahkan Berkat-Nya kepada Penulis.
2. Orang tua, Istri dan saudara-saudara saya yang telah memberikan semangat, motivasi dan doa kepada saya.
3. Bapak Rony Teguh S.Kom., MT., Ph.D dan Ibu Nahumi Nugrahaningsih, ST., MT., Ph.D selaku dosen pembimbing yang telah sabar membimbing dan memberikan motivasi dalam pembuatan program dan laporan.
4. Bapak Viktor H. Pranatawijaya, S.T., M.T, Bapak Abertun Sagit Sahay, S.T., M.Eng, dan Ibu Ariesta Lestari, S.Kom.,M.Cs., Ph.D, selaku dosen penguji yang telah banyak membantu dalam memberikan masukan guna penyempurnaan program dan laporan ini.
5. Rekan-rekan yang terlibat secara langsung maupun yang tidak langsung dalam pengerjaan projek skripsi/tugas akhir ini.

Besar harapan saya, semoga laporan ini dapat bermanfaat bagi kita semua, terkhusus bagi semua mahasiswa Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya.

Palangka Raya, Juni 2020

Riantono Hartaku

**PERANCANGAN *PROTOTYPE* PRESENSI MENGGUNAKAN
E-KTM BERBASIS *RADIO FREQUENCY*
*IDENTIFICATION***

Riantono Hartaku DBC 115 070

Jurusan Teknik Informatika Universitas Palangka Raya
Kampus Tunjung Nyaho Jl. Yos Sudarso Palangka Raya 73112
Email : rianhartaku@gmail.com

ABSTRAK

Prototype Presensi ini dibangun dengan tujuan bagaimana mengimplementasikan RFID sebagai sistem presensi dan dapat berfungsi sebagai pengganti presensi yang biasa digunakan yaitu presensi menggunakan kertas.

Pada perancangan *prototype* presensi ini menggunakan proses pengembangan SDLC (*System Development Life Cycle*) dengan metode *prototyping* yang telah dimodifikasi oleh penulis yang terbagi menjadi beberapa tahapan yaitu *Requirements Definition* (analisis), *System And Software Design* (desain), *Prototype Construction* (koding), *System Testing* (pengujian). Pada tahap analisis terdapat analisis sistem lama dan analisis sistem baru dan menggunakan metodologi UML dan desainnya menggunakan aplikasi *Edraw Max*, lalu ditahap pembangunan sistem menggunakan bahasa pemrograman *PHP*, *HTML* dan *Python*, kemudian pada tahap *testing* (pengujian) menggunakan *black box testing*.

Prototype ini mempunyai fitur rekap kehadiran yang lebih efisien dimana di dalamnya telah terekam data matakuliah, data mahasiswa, data kelas, data semester, pencatatan waktu saat mahasiswa melakukan absen, semua itu dapat langsung diunduh dan diprint menjadi rekap daftar hadir mahasiswa.

Kata kunci : *Prototype Presensi, Sistem Kehadiran, RFID, Raspberry PI*

***DESIGN PRESENCE PROTOTYPE USING E-KTM
BASED ON RADIO FREQUENCY
IDENTIFICATION***

Riantono Hartaku DBC 115 070

Department of Informatics Engineering Palangkaraya University
Tunjung Nyaho Campus Jl. Yos Sudarso Palangka Raya 73112
Email : rianhartaku@gmail.com

ABSTRACT

This Presence prototype was built with the aim of implementing RFID as a presence system and can be used as an implementation of the presence that is commonly used is the presence using paper.

In designing this presence prototype uses the process of developing SDLC (System Development Life Cycle) with a prototyping method that has been modified by the author which is divided into several stages, namely Requirements Definition (analysis), System And Software Design (design), Prototype Construction (coding), System Testing (testing).

This prototype has a more efficient attendance recap feature where in it has been recorded course data, student data, class data, semester data, time recording when students are absent, all of which can be directly downloaded and printed into a student attendance list.

Keyword : Prototype presence, Attendance system, RFID, Raspberry PI

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN	iv
HALAMAN RIWAYAT PENYUSUN	v
HALAMAN PERSEMBAHAN	vi
KATA PENGANTAR	vii
ABSTRAK	viii
DAFTAR ISI	x
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah.....	5
1.4 Tujuan Penelitian	6
1.5 Sistematika Penulisan	6
1.6 Jadwal Kegiatan	8
BAB II LANDASAN TEORI	9
2.1 Tinjauan Pustaka	9
2.2 Sistem Informasi	10
2.3 Presensi	12
2.4 Kartu Tanda Mahasiswa Elektronik.....	12
2.5 <i>Radio Frequency Identification</i> (RFID).....	13
2.6 <i>Raspberry Pi</i>	14
2.7 GPOI Raspberry Pi.....	15
2.8 <i>Database & Database Management System</i>	17
2.8.1 <i>Database</i>	17

2.8.2 Database Management System (DBMS).....	17
2.8.3 SQL	18
2.9 Unified Modeling Language (UML)	19
2.8.1 Use Case Diagram	19
2.8.2 Activity Diagram	21
2.8.3 Class Diagram.....	22
2.8.4 Flowchart	24
2.10 Bahasa Pemrograman.....	25
2.9.1 Hypertext Markup Language (HTML)	25
2.9.2 Personal Home Page (PHP)	27
2.7.3 Bahasa Pemrograman Python	31
2.11 MySql.....	32
BAB III METODOLOGI PENELITIAN	34
3.1 Alat dan Bahan.....	34
3.2 Metodologi Pengembangan	35
3.3 Metode Pengumpulan Data	37
3.4 Requirements Definition	38
3.4.1 Analisis Sistem Lama.....	38
3.4.2 Analisis Sistem Baru	40
3.5 Wire Diagram Rfid dan Raspberry Pi3	43
3.6 Diagram Sistem Rfid dan Raspberry Pi3	43
3.7 Desain Sistem.....	44
3.7.1 Use Case Diagram.....	45
3.7.2 Activity Diagram.....	47
3.7.3 Class Diagram	50
3.7.4 Desain Interface	51

BAB IV HASIL DAN PEMBAHASAN	55
4.1 <i>Protoypte Construction and system testing</i>	55
4.1.1 <i>Prototype Construction</i>	55
4.1.2 <i>System Testing</i>	62
BAB V KESIMPULAN DAN SARAN	73
5.1 Kesimpulan	73
5.2 Saran	74
DAFTAR PUSTAKA	
LEMBAR PERSETUJUAN	

DAFTAR TABEL

Tabel 1.1 Jadwal Kegiatan	8
Tabel 2.1 Keterangan Usecase Diagram	20
Tabel 2.2 Keterangan Activity Diagram	22
Tabel 2.3 Keterangan Class Diagram.....	23
Tabel 2.4 Flowchart (Diagram alir)	24
Table 3.1 Tabel Definisi Aktor	45
Table 3.2 Tabel Definisi <i>Use Case</i> Admin	45
Table 3.3 Tabel Definisi <i>Use Case</i> Mahasiswa	47
Table 4.1 Tabel <i>BlackBox Testing</i> Admin	62
Table 4.2 Tabel <i>BlackBox Testing</i> Mahasiswa	66

DAFTAR GAMBAR

Gambar 2.1 RFID <i>reader</i> dan RFID <i>tag</i>	14
Gambar 2.2 Raspberry Pi 3 model B+	16
Gambar 2.3 GPIO Raspberry Pi 3.....	17
Gambar 3.1 Metodologi Pengembangan SDLC	35
Gambar 3.2 <i>Flowchart</i> Sistem Lama	39
Gambar 3.3 <i>Flowchart</i> Sistem Baru Mahasiswa	41
Gambar 3.4 <i>Flowchart</i> Sistem Baru Admin.....	42
Gambar 3.5 Wire Diagram Rfid dan Raspberry Pi3	43
Gambar 3.6 Diagram Sistem Rfid dan Raspberry Pi3	43
Gambar 3.7 Use Case Admin.....	46
Gambar 3.8 Use Case Mahasiswa	47
Gambar 3.9 Activity Diagram Mahasiswa.....	48
Gambar 3.10 Activity Diagram Admin.....	49
Gambar 3.11 Class diagram	50
Gambar 3.12 Desain Interface Halaman Beranda admin.....	51
Gambar 3.13 Desain Interface Halaman Menu Mahasiswa.....	51
Gambar 3.14 Desain Interface Halaman edit Mahasiswa	52
Gambar 3.15 Desain Interface Halaman Mata Kuliah.....	52
Gambar 3.16 Desain Interface Halaman edit Mata Kuliah	53
Gambar 3.17 Desain Interface Halaman Jadwal (Pilih Semester).....	53
Gambar 3.18 Desain Interface Halaman Jadwal (Pilih Kelas).....	54
Gambar 3.19 Desain Interface Halaman Rekap Absen.....	54
Gambar 4.1 Halaman Home.....	56
Gambar 4.2 Halaman Mahasiswa	56
Gambar 4.3 Halaman Edit Mahasiswa.....	57
Gambar 4.4 Halaman Daftar Mata Kuliah	57
Gambar 4.5 Halaman Edit Mata Kuliah.....	58
Gambar 4.6 Halaman Jadwal (Pilih Semester)	58
Gambar 4.7 Halaman Jadwal (Pilih Kelas).....	59

Gambar 4.8 Halaman Jadwal (Daftar Jadwal Mata Kuliah)	59
Gambar 4.9 Hasil Keluaran Rekap Presensi Mahasiswa	60
Gambar 4.10 Alat Dinyalakan.....	60
Gambar 4.11 Mahasiswa Melakukan Absen.....	61
Gambar 4.12 Mahasiswa Melakukan Absen Beruntun.....	61
Gambar 4.13 Pengujian Pada Jarak 0-1 cm kartu terbaca.....	67
Gambar 4.14 Pengujian Pada Jarak 2 cm kartu terbaca	68
Gambar 4.15 Pengujian Pada Jarak 3 cm kartu terbaca	69
Gambar 4.16 Pengujian Pada Jarak 4 cm kartu tidak terbaca	70
Gambar 4.17 Pengujian Pada Jarak 5 cm kartu tidak terbaca	71

BAB I

PENDAHULUAN

1.1 Latar Belakang

RFID (*Radio Frequency identification*) adalah teknologi yang menggabungkan fungsi dari kopling elektromagnetik atau elektrostatik pada porsi frekuensi radio dari spectrum elektromagnetik, untuk mengidentifikasi sebuah objek. Teknologi RFID mudah digunakan dan sangat cocok untuk operasi otomatis. RFID mengkombinasikan keunggulan yang tidak tersedia pada teknologi identifikasi yang lain. RFID dapat disediakan dalam perangkat yang hanya dapat dibaca saja (*Read Only*) atau dapat dibaca dan ditulis (*Read/write*), tidak memerlukan kontak langsung maupun jalur cahaya untuk dapat beroperasi, dapat berfungsi pada berbagai variasi kondisi lingkungan, dan menyediakan tingkat integritas data yang tinggi. Pada sistem RFID umumnya, tag atau transporter ditempelkan pada suatu objek. Setiap tag dapat membawa informasi yang unik seperti serial number, model, warna, tempat perakitan, dan data lain dari objek tersebut. Ketika tag ini melalui medan yang dihasilkan oleh pembaca RFID yang kompatibel, tag akan mentransmisikan informasi yang ada pada tag kepada pembaca RFID, sehingga proses identifikasi objek dapat dilakukan.

Perancangan Sistem Absensi Kehadiran dengan Menggunakan *Radio Frequency Identification* (RFID), dijelaskan bahwa RFID merupakan suatu metode identifikasi objek yang menggunakan gelombang radio.

Proses identifikasi dilakukan oleh RFID *reader* dan RFID *transponder* (RFID tag). RFID tag dilekatkan pada suatu benda atau suatu objek yang akan diidentifikasi. Tiap-tiap RFID tag memiliki data angka identifikasi (*ID number*) yang unik, sehingga tidak ada RFID tag yang memiliki *ID number* yang sama.

Permasalahan yang sering timbul dalam presensi saat ini adalah banyak pegawai/mahasiswa terlambat dalam menghadiri kegiatan. Hal ini menimbulkan pegawai/mahasiswa melakukan penitipan presensi sehingga dosen tidak dapat mengetahui jumlah mahasiswa yang hadir sebenarnya. Harapan penulis membuat suatu alat *prototyping* presensi secara otomatis. Alat *prototyping* tersebut memanfaatkan RFID *Reader* sebagai tanda pengenal setiap pengguna.

Masing-masing pengguna yang hadir di ruangan pada suatu kegiatan akan diminta untuk menandatangani selembar kertas absensi oleh atasan/dosen sebagai bukti kehadiran. Selain hal tersebut, terkadang data absensi yang di masukkan ke dalam sistem juga sering terjadi kesalahan input dan ketidaksesuaian dengan data yang sebenarnya. Hal tersebut terjadi biasanya dikarenakan dosen tidak langsung setiap hari menginputkan data absensi ke dalam sistem absensi perkuliahan.

Teknologi ini merupakan suatu metode identifikasi objek yang menggunakan gelombang radio. Proses identifikasi dilakukan oleh RFID *reader* dan RFID *transponder* (RFID tag). RFID tag di letakkan pada suatu benda atau suatu objek yang akan di identifikasi. Setiap RFID tag memiliki data angka data identifikasi (*ID number*) yang unik, sehingga tidak ada RFID tag yang memiliki *ID number* yang sama.

Berdasarkan kamus Besar Bahasa Indonesia. Kata presensi adalah kehadiran seseorang, sedangkan Absensi adalah ketidakhadiran seseorang. Dalam bahasa Inggris, absen sering dipakai dalam bentuk istilah "*List of Absent*", yang artinya adalah daftar nama anggota yang tidak hadir. Sementara "Daftar Hadir" biasa ditulis "*List of Presence*" atau "*List of Participants*". Presensi suatu pendataan kehadiran yang merupakan bagian dari aktifitas pelaporan yang ada dalam sebuah institusi maupun sekolah/kampus. Presensi disusun dan diatur sehingga mudah untuk dicari dan dipergunakan ketika diperlukan oleh pihak yang berkepentingan.

Absensi disusun dan diatur sehingga mudah untuk dicari dan dipergunakan ketika diperlukan oleh pihak yang berkepentingan. Secara umum, jenis-jenis absensi menurut cara penggunaannya dapat dikelompokkan menjadi dua, yaitu:

- a. Absensi Manual, yang merupakan cara penulisan kehadiran dengan cara menggunakan pena berupa tanda tangan.
- b. Absensi non manual, yang merupakan cara penulisan kehadiran dengan menggunakan alat yang terkomputerisasi, bisa menggunakan kartu RFID ataupun fingerprint.

Untuk Sistem Absensi Berbasis Mikrokontroler, dijelaskan bahwa RFID adalah teknologi yang menggabungkan fungsi dari kopling elektromagnetik atau elektrostatik pada porsi frekuensi radio dari spectrum elektromagnetik, untuk mengidentifikasi sebuah objek. Pada sistem RFID umumnya, *tag* atau *transponder* ditempelkan pada suatu objek. Setiap *tag* dapat membawa informasi yang unik seperti nomor serial, model, warna, tempat perakitan, dan data lain dari

objek tersebut, ketika *tag* ini melalui medan yang dihasilkan oleh pembaca RFID yang kompatibel, *tag* akan mentransmisikan informasi yang ada pada *tag* ke RFID *reader*, sehingga proses identifikasi objek dapat dilakukan.

Metode identifikasinya menggunakan sarana yang disebut label RFID atau transponder (*tag*) untuk menyimpan dan mengambil data jarak jauh. Teknologi ini mampu mengidentifikasi berbagai objek secara simultan tanpa diperlukan kontak langsung (atau dalam jarak pendek). Implementasi RFID secara efektif digunakan pada lingkungan manufaktur atau industri yang memerlukan akurasi dan kecepatan identifikasi objek dalam jumlah yang besar serta berbeda di area yang luas.

1.2 RUMUSAN MASALAH

Berdasarkan latar belakang di atas, maka masalah-masalah yang ada dapat dirumuskan sebagai berikut :

1. Bagaimana membangun sebuah sistem presensi RFID menggunakan Raspberry Pi3 B+.
2. Bagaimana membuat sistem presensi menggunakan RFID *tag* sebagai kartu yang dideteksi dan RFID modul RC522 sebagai pembaca dari RFID *tag*.
3. Bagaimana mengirimkan data pada RFID *tag* melalui modul menuju server sebagai data presensi.

1.3 BATASAN MASALAH

Adapun batasan masalah dalam tugas akhir ini fokus kepada :

1. Tipe dari RFID *reader* yang digunakan adalah module Mifare RC522.
2. RFID *tag* yang digunakan merupakan jenis RC522 dengan frekuensi 13,45 Mhz berfungsi sebagai kartu pengenalan (alat presensi).
3. Dalam perancangan ini tidak membahas keamanan sistem.
4. Dalam perancangan ini tidak membahas tentang apa yang terjadi jika adanya halangan.
5. Pada percobaan alat dilakukan dengan pembacaan kartu satu persatu melewati *reader*.
6. Database yang dibuat menggunakan Mysql.
7. Sistem tidak menangani daftar hadir dosen.
8. Dalam perancangan ini tidak membahas Protokol yang digunakan.
9. Fitur dan menu yang terdapat pada website ini memiliki fitur-fitur sebagai berikut:

A. Admin.

- a) Halaman Status, merupakan halaman utama website setelah admin melakukan *login* ke website presensi.
- b) Kelola halaman Mahasiswa, merupakan halaman untuk mengelola data mahasiswa seperti menambah, mengedit dan menghapus data mahasiswa.

- c) Kelola halaman Mata Kuliah, merupakan halaman untuk mengelola data mata kuliah seperti menambah, mengedit dan menghapus data mata kuliah.
- d) Kelola halaman Jadwal / Absen, merupakan halaman untuk mengelolah jadwal mata kuliah / absen seperti menambah, mengedit, menghapus serta melakukan download rekap absen.

B. Mahasiswa

- a) Melakukan presensi pada *reader* RFID.

1.4 TUJUAN PENELITIAN

Tujuan dari projek tugas akhir ini antara lain :

1. Dapat mengimplementasikan RFID sebagai sistem presensi.
2. Dapat berfungsi sebagai pengganti absensi yang biasa digunakan yaitu absensi menggunakan kertas.

1.5 SISTEMATIKA PENULISAN

Penulisan Skripsi ini disusun dalam lima bab, dengan menggunakan sistematika sebagai berikut.

BAB I PENDAHULUAN

Pada bab ini, diuraikan mengenai latar belakang masalah, rumusan masalah, ruang lingkup/batasan masalah, maksud dan tujuan pengembangan, dan sistematika penulisan.

BAB II LANDASAN TEORI

Pada bab ini akan memuat landasan teori yang berkaitan dengan konsep dan prinsip dasar yang diperlukan untuk memecahkan masalah pekerjaan untuk merumuskan hipotesis. Landasan teori ini berbentuk permasalahan-permasalahan yang langsung berkaitan dengan permasalahan yang dikerjakan.

BAB III ANALISIS DAN DESAIN

Pada bab ini berisi tentang evaluasi masalah yang berlaku dalam menganalisis, merancang dan membuat sebuah desain *prototyping*.

BAB V IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan menjelaskan tentang tahapan dari implementasi pembuatan Aplikasi dan tahapan penggunaan serta testing program.

BAB V PENUTUP

Pada bab ini akan menguraikan kesimpulan beserta saran yang didapatkan dari awal pembuatan sampai terselesaikannya proyek Skripsi.

DAFTAR PUSTAKA

LAMPIRAN



1.6 JADWAL KEGIATAN

Tabel Jadwal kegiatan Skripsi

Tabel 1.1. Jadwal Skripsi

RENCANA KEGIATAN	Oktober				November					Maret				April				Juni			
	Minggu				Minggu					Minggu				Minggu				Minggu			
	I	II	III	IV	I	II	III	IV	V	I	II	III	IV	I	II	III	IV	I	II	III	IV
Penyusunan proposal dan Seminar Proposal																					
Pembuatan Sistem Informasi dan Pengumpulan Data																					
Penyusunan Skripsi dan Seminar Hasil																					
Pembuatan Laporan Akhir Skripsi																					
Seminar Tugas Akhir																					

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Untuk mendukung penelitian tugas akhir ini beberapa penelitian yang telah dilakukan sebelumnya, di antaranya sebagai berikut :

Dalam penelitiannya (Saputra, 2008) yaitu sistem absensi menggunakan teknologi RFID, dimana sistem yang dibuat ditujukan untuk melakukan absensi otomatis bagi karyawan di suatu perusahaan dengan memanfaatkan teknologi RFID. Sistem yang dibuat akan menyimpan secara otomatis data kehadiran karyawan. Hasil sistem yang dibuat dapat melakukan pembacaan dengan baik dimana jarak baca maksimumnya 2 cm dengan peluang keberhasilan 1 dan interval waktu pembacaan minimum 2 detik untuk melakukan fungsi sistem absensi tersebut secara optimal.

Dalam penelitiannya (Kustianto, 2010) yaitu perancangan dan implementasi sistem pencarian buku pada perpustakaan berbasis RFID dengan antarmuka *visual basic* dan basis data MySQL di mana implementasi penelitian ini adalah dengan menempatkan RFID *tag* di dalam sebuah buku yang nantinya data buku tersebut di masukkan ke dalam *database* berbasis MySQL sesuai dengan kode RFID *tag* yang disisipkan dan kemudian untuk melakukan pencarian buku tersebut bisa dengan pemindaian menggunakan RFID *reader* sehingga pencarian buku menjadi lebih mudah.

Dalam penelitiannya (Aditya, 2013) yaitu aplikasi RFID untuk sistem presensi mahasiswa di universitas brawijaya berbasis protokol internet di mana penelitian yang dilakukan adalah untuk membuat sebuah sistem absensi mahasiswa berbasis RFID menggunakan mikrokontroler Atmega 328 dan jaringan intranet menggunakan modul WIZ812MJ. Hasil penelitian berhasil menciptakan sistem yang bisa melakukan pencatatan absensi dengan rata-rata waktu yang dibutuhkan dari proses pembacaan hingga menerima respon dari server sebesar 40ms.

Dalam penelitiannya (Nugroho, 2014) yaitu sistem absensi berbasis RFID menggunakan protocol internet di mana implementasi sistem yang dibuat adalah dengan menempatkan sebuah personal komputer di setiap ruang kuliah yang masing masing PC tersebut terhubung dalam satu jaringan. RFID reader akan secara otomatis membaca ID mahasiswa yang masuk ke ruang kuliah, lalu mengirimkan data id tersebut ke PC. PC akan membandingkan data informasi ID yang dikirimkan dengan database yang telah dibuat kemudian mengunggah hasil statistik dari ID yang hadir ke database server melalui jaringan internet. Database pada server akan menyimpan data mahasiswa yang hadir maupun tidak hadir dalam perkuliahan tertentu beserta waktu kehadirannya.

2.2 Sistem Informasi

Sistem informasi menurut Robert A. Leitch dan K. Roscoe Davis (Andi, 2005:18) adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial

dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Sistem informasi merupakan suatu perkumpulan data yang terorganisasi beserta tatacara penggunaannya yang mencangkup lebih jauh dari pada sekedar penyajian. Istilah tersebut menyiratkan suatu maksud yang ingin dicapai dengan jalan memilih dan mengatur data serta menyusun tatacara penggunaannya.

Keberhasilan suatu sistem informasi yang diukur berdasarkan maksud pembuatannya tergantung pada tiga faktor utama, yaitu : keserasian dan mutu data, pengorganisasian data, dan tatacara penggunaannya. Untuk memenuhi permintaan penggunaan tertentu, maka struktur dan cara kerja sistem informasi berbeda-beda bergantung pada macam keperluan atau macam permintaan yang harus dipenuhi. Suatu persamaan yang menonjol ialah suatu sistem informasi menggabungkan berbagai ragam data yang dikumpulkan dari berbagai sumber. Untuk dapat menggabungkan data yang berasal dari berbagai sumber suatu sistem alih rupa (*transformation*) data sehingga jadi tergabungkan (*compatible*). Berapa pun ukurannya dan apapun ruang lingkungannya suatu sistem informasi perlu memiliki ketergabungan (*compatibility*) data yang disimpannya. (Hanif Al Fatta, 2009:9).

Berdasarkan pendapat-pendapat di atas dapat disimpulkan bahwa sistem informasi adalah sekumpulan prosedur organisasi yang dilaksanakan untuk mencapai suatu tujuan yaitu memberikan informasi bagi pengambil keputusan dan untuk mengendalikan organisasi.

2.3 Presensi

Presensi dalam kamus besar bahasa Indonesia memiliki arti “kehadiran” sedangkan absensi adalah “ketidakhadiran”. Contohnya dalam kasus di perguruan tinggi, mahasiswa pada saat jam mata kuliah berlangsung biasanya diminta untuk mengisi daftar kehadiran mata kuliah yang sedang berlangsung. Kehadiran mahasiswa pada saat jam mata kuliah berlangsung sangat berpengaruh terhadap penilaian dosen, karena dari daftar kehadiran mahasiswa, dosen dapat melihat seberapa antusiasnya mahasiswa terhadap pelajaran tersebut. Kehadiran mahasiswa juga menjadi faktor penting bagi dosen untuk memberikan nilai akhir mata kuliah yang bersangkutan.

2.4 Kartu Tanda Mahasiswa Elektronik (E-KTM)

Di perguruan tinggi, mahasiswa memiliki kartu identitas elektronik atau yang sering disebut dengan E-KTM (Kartu Tanda Mahasiswa Elektronik). Kartu Tanda Mahasiswa Elektronik (E-KTM) ini merupakan sebuah kartu identitas yang wajib dimiliki setiap mahasiswa dalam sebuah Universitas.

Kartu ini merupakan tanda pengenal dimana di dalamnya terdapat informasi-informasi terkait dengan mahasiswa yang memilikinya, dan tampilan fisik dari E-KTM ini pada umumnya terdapat logo dan nama universitas, nama lengkap mahasiswa, nomor induk mahasiswa, dan program studi mahasiswa. Tidak seperti era sebelumnya, kini KTM telah mengalami banyak perubahan yang signifikan yang bertujuan untuk memberikan kemudahan dan kenyamanan kepada penggunanya.

Saat ini KTM tidak hanya digunakan sebagai kartu tanda pengenal mahasiswa saja, tetapi beberapa KTM di universitas yang ada di Indonesia kini sudah dapat digunakan untuk transaksi perbankan. Fungsi ganda yang dimiliki oleh kartu ini memudahkan mahasiswa dalam mengakses dan memenuhi kebutuhan di dalam maupun di luar kampus.

2.5 *Radio Frequency Identification (RFID)*

Radio frequency identification (RFID) adalah sebuah teknologi yang menggunakan komunikasi via gelombang elektromagnetik untuk merubah data antara terminal dengan suatu objek seperti produk barang, hewan, ataupun manusia dengan tujuan untuk identifikasi dan penelusuran jejak melalui penggunaan suatu piranti yang bernama *RFID tag*.

RFID tag dapat bersifat aktif atau pasif. *RFID tag* yang pasif tidak memiliki *power supply* sendiri, sehingga harganya pun lebih murah dibandingkan dengan *tag* yang aktif. Dengan hanya berbekal induksi listrik yang ada pada antena yang disebabkan oleh adanya pemindaian frekuensi radio yang masuk, sudah cukup untuk memberi kekuatan yang cukup bagi *RFID tag* untuk mengirimkan respon balik.

Suatu sistem RFID dapat terdiri dari beberapa komponen, seperti *tag*, *tag reader*, *tag programming station*, *circulation reader*, *sorting equipment*, dan tongkat *inventory tag*. Kegunaan dari sistem RFID ini adalah untuk mengirimkan data dari *tag* yang kemudian dibaca oleh *RFID reader* dan kemudian diproses

oleh aplikasi computer. Data yang dipancarkan dan dikirimkan tadi bisa berisi beragam informasi, seperti ID, informasi lokasi atau informasi lainnya.

Dalam suatu sistem RFID sederhana, suatu *object* dilengkapi dengan *tag* yang berisi *microchip* yang ditanamkan di dalamnya yang berisi sebuah kode produk yang sifatnya unik. Sebaliknya, *interrogator*, suatu antena yang berisi *transceiver* dan *decoder*, memancarkan sinyal yang bisa mengaktifkan RFID *tag* sehingga dia dapat membaca dan menulis data ke dalamnya.



Gambar 2.1 RFID reader dan RFID tag

2.6 Raspberry Pi

Raspberry Pi adalah sebuah SBC (*Single Board Computer*) seukuran kartu kredit yang dikembangkan oleh Yayasan Raspberry Pi di Inggris (UK) dengan maksud untuk memicu pengajaran ilmu komputer dasar di sekolah-sekolah (Putra, 2012). *Raspberry Pi* dikenalkan pada tahun 2012 dan memiliki *Processor* bernama *Broadcom BCM2835 system on chip (SOC)* yang telah memiliki *ARM1176JZF-S 700 MHz CPU*, untuk *Graphics* telah disertakan *VideoCore IV GPU*, serta telah memiliki ram sebesar 256MB untuk model A, dan telah ditingkatkan ke 512 MB untuk model B dan B+ pada generasi pertama. Sedangkan untuk generasi kedua *Raspberry Pi*, dimana diperkenalkan pada

Februari 2015 memiliki *Processor Broadcom BCM2836 SoC*, dengan *Processor quad-core ARM Cortex-A7 CPU* dan sebuah *VideoCore IV dual-core GPU*; serta memiliki ram sebesar 1 GB. *System on Chip* yang dipakai oleh *Raspberry Pi* diciptakan oleh Boradcom, dan menggunakan arsitektur ARM. Arsitektur ARM merupakan arsitektur prosesor 32-bit RISC yang dikembangkan oleh ARM *Limited*. Dikenal sebagai *Advanced RISC Machine* di mana sebelumnya dikenal sebagai *Acorn RISC Machine*. Pada awalnya merupakan prosesor desktop yang sekarang didominasi oleh keluarga x86. Namun desain yang sederhana membuat prosesor ARM cocok untuk aplikasi berdaya rendah. Hal ini membuat prosesor ARM mendominasi pasar *mobile electronic* dan *embedded system* dimana membutuhkan daya dan harga yang rendah.



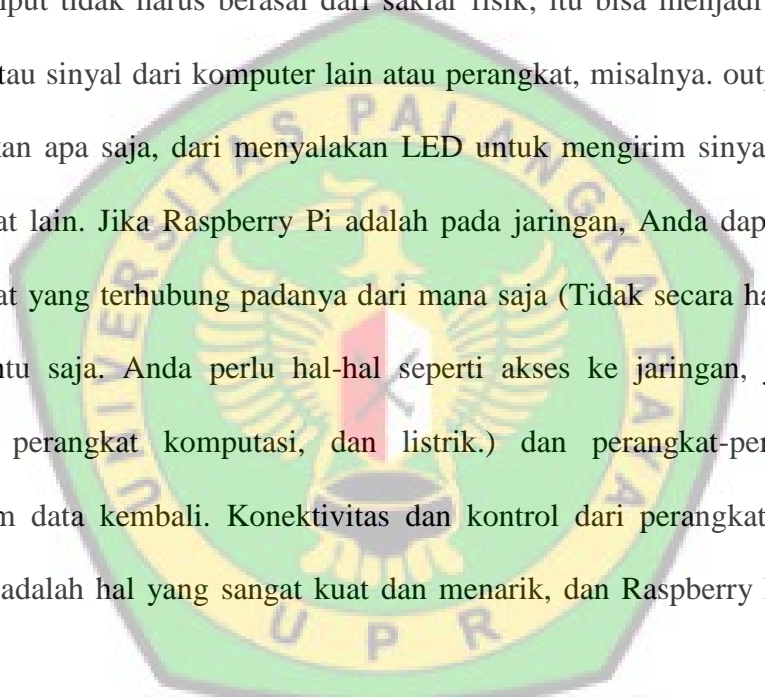
Gambar 2.2 Raspberry Pi 3 model B+

2.7 GPIO Raspberry Pi 3

GPIO merupakan sederet pin yang terdiri dari 40 pin dengan berbagai fungsi. Salah satu fitur yang kuat dari Raspberry Pi adalah deretan GPIO (tujuan umum input / output) pin di sepanjang tepi atas pin board. These adalah antarmuka fisik antara Pi dan dunia luar. Pada tingkat yang paling sederhana, Anda dapat

menganggap mereka sebagai switch yang Anda dapat mengaktifkan atau menonaktifkan (input) atau bahwa Pi dapat mengaktifkan atau menonaktifkan (output).

Dari 40 pin, 26 pin GPIO dan yang lain adalah pin power atau ground (ditambah dua pin ID EEPROM yang tidak harus anda gunakan). Anda dapat memprogram pin untuk berinteraksi dengan cara yang menakjubkan dengan dunia nyata. Input tidak harus berasal dari saklar fisik; itu bisa menjadi masukan dari sensor atau sinyal dari komputer lain atau perangkat, misalnya. output juga dapat melakukan apa saja, dari menyalakan LED untuk mengirim sinyal atau data ke perangkat lain. Jika Raspberry Pi adalah pada jaringan, Anda dapat mengontrol perangkat yang terhubung padanya dari mana saja (Tidak secara harfiah di mana saja, tentu saja. Anda perlu hal-hal seperti akses ke jaringan, jaringan yang mampu perangkat komputasi, dan listrik.) dan perangkat-perangkat dapat mengirim data kembali. Konektivitas dan kontrol dari perangkat fisik melalui internet adalah hal yang sangat kuat dan menarik, dan Raspberry Pi ideal untuk ini.



3.3V	1	2	5V
I2C0 SDA	3	4	DNC
I2C0 SCL	5	6	GROUND
GPIO4	7	8	UART TXD
DNC	9	10	UART RXD
GPIO 17	11	12	GPIO 18
GPIO 21	13	14	DNC
GPIO 22	15	16	GPIO 23
DNC	17	18	GPIO 24
SP10 MOSI	19	20	DNC
SP10 MISO	21	22	GPIO 25
SP10 SCLK	23	24	SP10 CE0 N
DNC	25	26	SP10 CE1 N

Gambar 2.3 GPIO Raspberry Pi 3

2.8 Database dan Database Management System

2.8.1. Database

Menurut *Silberschatz, dkk.*, (2002) *Database* didefinisikan sebagai kumpulan data yang berisi informasi yang sesuai dengan sebuah perusahaan.

Sedangkan menurut *Ramakrishnan dan Gehrke* (2003) *Database* dinyatakan sebagai kumpulan data, umumnya mendeskripsikan aktivitas suatu organisasi atau lebih yang berhubungan.

Dari dua pengertian di atas dapat disimpulkan bahwa *Database* adalah kumpulan data yang saling berelasi, dan data sendiri merupakan fakta mengenai obyek, orang dan sebagainya yang dinyatakan dengan nilai.

2.8.2. Database Management System (DBMS)

Menurut *Conolly dan Begg* (2005), *Database Management System (DBMS)* adalah suatu *system software* yang memungkinkan *user* untuk mendefinisikan, membuat, memelihara, dan mengatur akses ke basis data.

DBMS berinteraksi dengan program aplikasi *user* dan basis data. *DBMS* menyediakan fungsi-fungsi sebagai berikut:

1. Memungkinkan *user* untuk mendefinisikan basis data, biasanya dari *Data Definition Language (DDL)*. *DDL* memungkinkan *user* untuk membedakan tipe dan struktur data, dan batasan data yang akan disimpan dalam basis data.

2. Memungkinkan *user* untuk menyisipkan, meng-*update*, menghapus dan menerima data dari basis data, biasanya dari *Data Manipulation Language (DML)*.
3. Menyediakan kontrol akses ke basis dengan menyediakan:
 - a. Sistem keamanan yang mencegah akses ilegal ke dalam basis data
 - b. Sistem integrasi yang memelihara akurasi data.
 - c. Sistem pembagian hak akses ke basis data.
 - d. Sistem pengendalian untuk memulihkan basis data ke keadaan sebelumnya yang dikarenakan oleh kegagalan *Software* atau *Hardware*.

2.8.3 SQL

SQL adalah kependekan dari *Structural Query Language*. Bahasa ini merupakan standar yang digunakan untuk mengakses basis data relasional. Fungsi SQL tidak sekedar untuk melakukan *query* (memperoleh data), melainkan juga dapat digunakan untuk kepentingan lain dalam memanipulasi data. Penggolongan perintah SQL adalah sebagai berikut :

- a. DDL (*Data Definition Language*) atau bahasa pendefinisi data

DDL adalah perintah-perintah yang berkaitan dengan penciptaan atau penghapusan objek seperti tabel dan indeks dalam basis data. Versi ANSI mencakup *CREATE TABLE*, *CREATE INDEX*, *ALTER TABLE*, *DROP TABLE*, *DROP VIEW*, dan *DROP INDEX*. Beberapa sistem basis data

menambahkan pernyataan DDL seperti *CREATE DATABASE* dan *CREATE SCHEMA*.

b. DML (*Data Manipulation Language*) atau bahasa manipulasi data

DML mencakup perintah-perintah yang digunakan untuk memanipulasi data. Misalnya, untuk menambahkan data (*INSERT*), memperoleh data (*SELECT*), mengubah data (*UPDATE*), dan menghapus data (*DELETE*).

c. DCL (*Data Control Language*) atau bahasa pengendali data.

DCL adalah kelompok perintah yang dipakai untuk melakukan otorisasi terhadap pegeksekan data dan pengalokasian ruang. Termasuk dalam kategori DCL yaitu pernyataan-pernyataan *GRANT*, *REVOKE*, *COMMIT*, dan *ROLLBACK*.

2.9 Unified Modeling Language (UML)








Menurut Nugroho (2009:4), “UML (*Unified Modeling Language*) adalah Metodologi kolaborasi antara metoda-metoda *Booch*, *OMT (Object Modeling Technique)*, serta *OOSE (object Oriented Software Engineering)* dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek (*OOP*)”.




2.9.1 Use Case Diagram

Use case adalah metode berbasis teks untuk menggambarkan dan mendokumentasikan proses yang kompleks. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* dikembangkan oleh sistem

analisis bersama-sama dengan user. Pada tahapan selanjutnya berdasarkan *use case* ini analisis menyusun model data dan model proses. Struktur dari masing-masing elemen *use case* diperlihatkan pada *Tabel 2.1*. berikut :

Tabel 2.1 Keterangan Use-case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.






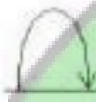

8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2.9.2 Diagram Aktivitas (*Activity Diagram*)

Diagram aktivitas menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, proses yang mungkin terjadi, dan bagaimana mereka berakhir. Diagram aktivitas juga Yaitu salah satu cara untuk memodelkan *event-event* yang terjadi dalam suatu *use case*. Sumber : Adi Nugroho (2005 : 66)

Diagram aktivitas merupakan *state* diagram khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi dipicu oleh selesainya *state* sebelumnya (*internal processing*). Diagram aktivitas mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah Diagram aktivitas bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. *Tabel 2.2.* dibawah ini merupakan simbol dan fungsi pada diagram aktivitas.

Tabel 2.2 Keterangan Activity Diagram

Simbol	Nama	Keterangan
	Start State	Memperlihatkan dimana aliran kerja berawal.
	End State	Memperlihatkan dimana aliran kerja berakhir.
	State	Menambahkan state untuk suatu objek.
	Activity	Menggambarkan Langkah-langkah dalam aliran kerja.
	Decision	Memperlihatkan bagaimana dua atau lebih langkah pada aliran-aliran kerja
	Transition	Memperlihatkan bagaimana aliran-aliran kerja bergerak dari suatu activity ke activity lainnya
	State Transition	Menambahkan transisi dari suatu aktifitas ke aktivitas yang lainnya

2.9.3 Class Diagram

Menurut *Booch* (2005), *class diagram* menunjukkan sekumpulan kelas, antarmuka dan kerjasama serta hubungannya. *Class diagram* digunakan untuk memodelkan perancangan statik dari gambaran sistem.

Class diagram dapat digunakan untuk membangun sistem yang dapat dieksekusi melalui teknik *forward* and *reverse*, selain untuk penggambaran, penspesifikasian, dan pendokumentasian struktur model.

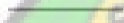




Class Diagram terdiri dari :



- a) Nama *Class*.
- b) Atribut.
- c) Operasi/*Method*.

Atribut dan Operasi/*method* dapat memiliki tiga sifat yaitu :

- a) *Public*, dapat dipanggil oleh *class* apa saja.
- b) *Protected*, hanya dapat dipanggil atau diakses oleh *class* yang bersangkutan dan *class* turunannya.
- c) *Private*, hanya dapat dipanggil oleh dirinya sendiri (tidak dapat diakses dari luar *class* yang bersangkutan).

Tabel 2.3 Keterangan *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.

6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.9.4 Flow Chart (Diagram Alir)

Flowchart atau Diagram Alir adalah sekumpulan simbol-simbol atau skema yang menunjukkan atau menggambarkan rangkaian kegiatan-kegiatan program dari mulai awal hingga akhir. Inti pembuatan dari *flowchart* atau Diagram Alir ini adalah penggambaran urutan langkah-langkah pengerjaan dari suatu algoritma. (Jogiyanto H.M 2001).

Tabel 2.4 Keterangan *Flowchart*

No.	Simbol	Fungsi
1		Terminal, untuk memulai dan mengakhiri suatu program
2		Proses, suatu simbol yang menunjukkan setiap pengolahan yang dilakukan oleh komputer
3		Input – Output, untuk memasukkan data maupun menghasilkan hasil dari suatu proses
4		Decision, suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan
5		Pred-defined Process, suatu simbol untuk menyediakan tempat-tempat pengolahan data dalam storage
6		Connector, suatu prosedur akan masuk dan keluar melalui simbol ini dalam lembar yang sama
7		Off Line Connector, merupakan simbol untuk masuk dan keluarnya suatu prosedur pada lembar kertas yang lain
8		Arus atau Flow, prosedur yang dapat dilakukan dari atas ke bawah, bawah ke atas, dari kekanan, atau dari kanan ke kiri
9		Document merupakan simbol untuk data yang berbentuk informasi
10		Untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur
11		Simbol untuk output yang ditujukan suatu device, seperti printer, plotter
12		Untuk menyimpan data

2.10 Bahasa Pemrograman

2.10.1 *HyperText Markup Language (HTML)*

a). Pendahuluan *HTML*

Menurut Betha Sidik (2004) *HTML (Hypertext Markup Language)* yaitu salah satu bahasa scripting yang dapat menghasilkan halaman *website* sehingga halaman tersebut dapat diakses setiap computer pengakses (*client*). Dokumen *HTML* merupakan dokumen yang disajikan dalam *browser web* surfer. Dokumen ini umumnya berisi informasi ataupun aplikasi dalam internet.

HTML atau singkatan dari *HyperText Markup Language* adalah *script* dimana kita bisa menampilkan informasi dan daya kreasi kita lewat internet. *HTML* sendiri adalah suatu dokumentasi teks biasa yang mudah dimengerti dibanding bahasa pemrograman lainnya dan karena bentuknya itu *HTML* dapat dibaca oleh berbagai platform seperti *Windows*, *Linux*, dan *Macintosh*.

b). Bagian-Bagian *HTML*

HTML terdiri dari beberapa bagian yang fungsinya sebagai penanda suatu kelompok perintah tertentu, misalnya kelompok perintah *form* yang ditandai dengan kode `<form>`, judul dengan `<title>`, dan sebagainya. Untuk lebih lanjut mengenai bagian-bagian *HTML* perhatikan skema di bawah ini:

```

<html>
<head>
<title>...</title>
</head>
<body>
... isi dari halaman web ...
</body>
</html>

```

Keterangan:

1. Dokumen *HTML* selalui diawali dengan tanda tag pembuka `<html>` dan diakhiri dengan tag penutup `</html>`.
2. Pada elemen `<head>`, dapat disisipkan kode-kode untuk menuliskan keterangan dokumen *HTML* atau disisipkan *script-script* pemrograman *web* seperti *JavaScript*, *VBScript*, dan *CSS* untuk menambah daya tarik pada situs yang kita buat.
3. Elemen `<body>...</body>` berisi tag-tag unuk isi atau *layout* tampilan pada situs, seperti:

```
<font>...</font>, <table>...</table>, <form>...</form>.
```

Tag adalah kode-kode yang digunakan untuk men-setting dokumen *HTML*. Secara garis besar bentuk umum tag sebagai berikut:

```
<tag-awal>TEKS</tag-akhir>
```

Namun ada juga tag yang tidak perlu ada tag penutupnya seperti `
`, `<hr>`, ``, dan lain sebagainya.

2.10.2 *Personal Home Page (PHP)*

a). Pemrograman *PHP*

PHP (personal home page tools) merupakan kependekatan dari kata *Hypertext preprocessor*. *PHP* tergolong sebagai perangkat lunak *open source* yang diatur dalam aturan *general purpose licences (GPL)* dan tergolong sebagai bahasa pemrograman yang berbasis *server (server side scripting)*. Pemrograman *PHP* sangat cocok dikembangkan dalam lingkungan *web*, karena *PHP* bias diletakan pada *script HTML* atau sebaliknya.

PHP dikhususkan untuk pengembangan *web* dinamis, maksudnya *PHP* mampu menghasilkan *website* yang secara terus menerus hasilnya bisa berubah sesuai permintaan *client browsernya* (Supriyanto, 2008).

PHP adalah *script* yang berjalan pada *server side* yang ditambahkan pada *HTML*. *Script* ini akan membuat suatu aplikasi yang dapat diintegrasikan kedalam *HTML* sehingga suatu halaman *HTML* tidak lagi bersifat statis, namun menjadi bersifat dinamis. Sifat *server side* membuat pengerjaan *script* tersebut dikerjakan di *server* sedangkan yang dikirimkan kepada *browser* adalah hasil proses dari *script* tersebut yang sudah berbentuk *HTML*.

PHP dibuat pada tahun 1994 oleh *Rasmus Lerdof*. Tetapi dikembangkan oleh orang lain dan setelah memulai tiga kali karya penulisan, akhirnya *PHP* menjadi bahasa pemrograman *web*.

PHP adalah sebuah produk yang berbentuk *open source*, sehingga *source code-code* dari *PHP* dapat digunakan, diganti, diedit tanpa harus dikenakan biaya.

b). Kelebihan dan Kelemahan *PHP*

Kelebihan *PHP* sebagai bahasa *server side* adalah:

1. Bahasa pemrograman *PHP* adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. *Web server* yang mendukung *PHP* dapat ditemukan dimana-mana dari mulai *IIS* sampai dengan *Aphace*, dengan konfigurasi yang *relative* mudah.
3. Dapat digunakan diberbagai mesin (*Linux, Unix, Windows*) dan dapat dijalankan secara *runtime* melalui *console* serta juga dapat menjalankan perintah-perintah sistem.

Adapun kelemahan *PHP* adalah:

1. Tidak ideal untuk pengembangan skala besar.
2. Tidak bisa memisahkan antara tampilan dengan logik dengan baik (walaupun pengguna *template* dapat memperbaikinya).
3. *PHP* memiliki kelemahan *security* tertentu apabila programmer tidak jeli dalam melakukan pemrograman dan kurang memperhatikan isu dan konfigurasi *PHP*.

c). Penggabungan *Script PHP* dan *HTML*

Bahasa pemrograman *PHP* dapat digabungkan dengan *HTML* dengan terlebih dahulu memberikan tanda tag buka dilanjutkan tanda tanya (<?) kemudian ditutup dengan tanda tanya dilanjutkan tanda tag tutup (?>).

Ada dua tipe penggabungan antara *PHP* dan *HTML* yaitu:

1. *Embedded Script*

Embedded script adalah cara penulisan tag *PHP* disela-sela tag *HTML*. Dengan cara ini, penulisan tag *PHP* digunakan untuk mengait bagian-bagian tertentu dalam dokumen yang memerlukan *script PHP* untuk proses di dalam *server*.

Contoh penulisan *Embedded script* dapat dilihat di bawah ini:

```
<html>
<head>
<title>Testing</title>
</head>
<body>
<?echo"Welcome";?>
</body>
</html>
```

2. *Non Embedden Script*

Non Embedded Script adalah cara penulisan tag *PHP* dibagian paling awal dan paling akhir dokumen. Dengan cara ini, penulisan tag *PHP* digunakan untuk mengawali dan mengakhiri keseluruhan bagian dalam sebuah dokumen. *Non Embedded Script* menempatkan *script HTML*

sebagai bagian dari *script* PHP. Contoh penulisan *Non Embedded Script* dapat dilihat di bawah ini:

```
<?php
echo
'<html>';
echo
'<head>';
echo
'<title>Testi
ng</title>';
echo
'</head><bo
dy>'; echo
'Welcome';
echo
'</body>';
echo
'</html>';
?>
```

d). *Cascading Style Sheet (CSS)*

Cascading Style Sheet (CSS) adalah sebagai pelengkap pada *HTML*. *CSS* digunakan untuk menetapkan aturan tampilan/ *style* yang akan digunakan pada sebuah *website*. *CSS* dapat mengubah besar kecilnya *text*, mengganti warna *background* pada sebuah halaman, atau dapat pula mengubah warna *border* pada tabel, serta masih banyak lagi hal yang dapat

dilakukan oleh CSS. Singkatnya, CSS digunakan untuk mengatur susunan tampilan pada halaman *HTML*.

2.10.3 Bahasa Pemrograman *Python*

Python adalah bahasa pemrograman interpretatif multiguna. Tidak seperti bahasa lain yang susah untuk dibaca dan dipahami, *python* lebih menekankan pada keterbacaan kode agar lebih mudah untuk memahami sintaks. Hal ini membuat *Python* sangat mudah dipelajari baik untuk pemula maupun untuk yang sudah menguasai bahasa pemrograman lain.

Bahasa ini muncul pertama kali pada tahun 1991, dirancang oleh seorang bernama Guido van Rossum. Sampai saat ini *Python* masih dikembangkan oleh *Python Software Foundation*. Bahasa *Python* mendukung hampir semua sistem operasi, bahkan untuk sistem operasi Linux, hampir semua distronya sudah menyertakan *Python* di dalamnya.

Dengan kode yang simpel dan mudah diimplementasikan, seorang programmer dapat lebih mengutamakan pengembangan aplikasi yang dibuat, bukan malah sibuk mencari *syntax error*.

```
print("Python sangat simpel")
```

Hanya dengan menuliskan kode print seperti yang diatas, anda sudah bisa mencetak apapun yang anda inginkan di dalam tanda kurung (). Dibagian akhir kode pun, anda tidak harus mengakhirinya dengan tanda *semicolon ;*

2.11 MySQL

MySQL merupakan software DBMS (atau *server database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user* (*multi-user*) dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*).

Saat ini, MySQL banyak digunakan di berbagai kalangan untuk melakukan penyimpanan dan pengolahan data, mulai dari kalangan akademis sampai ke industri, baik industri kecil, menengah maupun besar.

Lisensi MySQL terbagi menjadi dua, yaitu MySQL produk *open source* di bawah GNU *General Public License* (Gratis) dan versi komersial dengan membeli lisensi.

Kelebihan dari MySQL adalah sebagai berikut :

a. Fleksibel

MySQL dapat digunakan untuk mengembangkan aplikasi *desktop* maupun aplikasi web dengan menggunakan teknologi yang bervariasi. Ini berarti bahwa MySQL memiliki fleksibilitas terhadap teknologi yang akan digunakan sebagai pengembang aplikasi, baik PHP, Delphi, JSP, Java, C++ maupun lainnya.

b. Performa Tinggi

MySQL memiliki mesin *query* dengan performa tinggi, dengan demikian proses transaksional dapat dilakukan dengan sangat cepat.

c. *Lintas Platform*

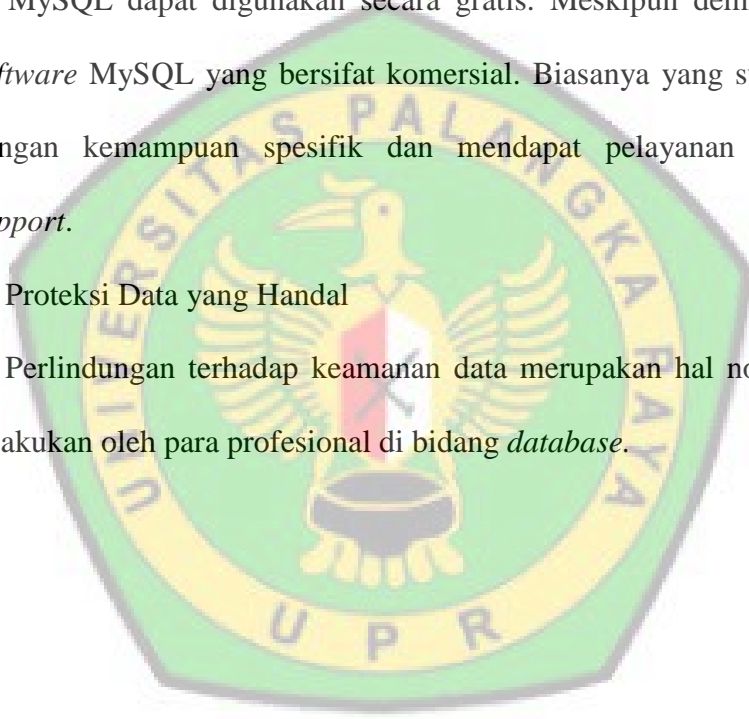
MySQL dapat digunakan pada *platform* atau lingkungan (dalam hal ini Sistem Operasi) yang beragam, bisa Microsoft Windows, Linux, atau UNIX. Ini menyebabkan proses imigrasi data (bila dibutuhkan) antarsistem operasi dapat dilakukan secara lebih mudah.

d. *Gratis*

MySQL dapat digunakan secara gratis. Meskipun demikian, ada juga *software* MySQL yang bersifat komersial. Biasanya yang sudah ditambahi dengan kemampuan spesifik dan mendapat pelayanan dari *technical support*.

e. *Proteksi Data yang Handal*

Perlindungan terhadap keamanan data merupakan hal nomor satu yang dilakukan oleh para profesional di bidang *database*.



BAB III

METODOLOGI PENELITIAN

3.1 Alat dan Bahan

Adapun alat dan bahan yang diperlukan dalam "Perancangan *Prototype* Presensi menggunakan E-KTM Berbasis *Radio Frequency Identification*" adalah sebagai berikut:

1. *Software tools* yang akan digunakan yaitu :
 - a. Windows 10 sebagai OS
 - b. MySQL
 - c. Microsoft Visio dan Edraw Max untuk perancangan sistem
 - d. Microsoft Word 2010 untuk pengetikan Laporan
 - e. Raspbian Stretch sebagai OS raspberry pi 3 B+
2. Spesifikasi *hardware* yang digunakan yaitu :
 - a. Processor : Radeon AMD A10
 - b. RAM : 4 GB DDR4
 - c. 2 buah : Raspberry Pi 3 Model B+
 - d. Monitor
 - e. MicroSD
 - f. Raspberry adaptor 5V 2.5A
 - g. Module RFID
 - h. RFID Tag

3.2 Metodologi Pengembangan

Penelitian ini merupakan penelitian pengembangan (*Research and Development*) karena penelitian ini bertujuan menghasilkan atau mengembangkan desain dan *prototype* sistem informasi bukan penelitian yang menemukan teori. Dikarenakan luaran yang dihasilkan dari penelitian ini merupakan desain dan *prototype* sistem informasi presensi menggunakan RFID yang terintegrasi dengan sistem informasi akademik maka dalam pengembangannya menggunakan proses SDLC (*System Development Life Cycle*) berikut adalah tahapan-tahapan dalam penelitian :



Gambar 3.1 Metodologi pengembangan SDLC

Tahapan-tahapan dalam *System Development Life Cycle* yaitu :

1. *Requirement Definition* (Analisis Kebutuhan Sistem)

Pada tahap awal ini dilakukan analisa guna menggali kebutuhan yang dibutuhkan, yaitu kebutuhan data dan kebutuhan user.

2. *System and Software Design* (Perancangan)

Selanjutnya, hasil analisa kebutuhan sistem tersebut akan dibuat sebuah desain database, UML, antarmuka pengguna atau Graphical User Interface dan jaringan yang dibutuhkan untuk sistem. Selain itu juga perlu merancang struktur data, arsitektur perangkat lunak, detil prosedur dan karakteristik tampilan yang akan disajikan. Proses ini menterjemahkan kebutuhan sistem ke dalam sebuah model perangkat lunak yang dapat diperkirakan kualitasnya sebelum memulai tahap implementasi.

3. *Prototype Construction* (Coding)

Rancangan yang telah dibuat dalam tahap sebelumnya akan diterjemahkan dalam bentuk bahasa yang dapat dibaca oleh komputer untuk diolah. Tahap ini juga dapat disebut dengan tahap implementasi, yaitu tahap yang mengkonversi hasil perancangan sebelumnya ke dalam sebuah bahasa pemrograman yang dimengerti oleh komputer. Kemudian komputer akan menjalankan fungsi-fungsi yang telah didefinisikan sehingga mampu memberikan layanan-layanan kepada penggunanya.

4. *System Testing* (Pengujian)

Pengujian program dilakukan untuk mengetahui kesesuaian sistem berjalan sesuai prosedur atau tidak dan memastikan sistem terhindar dari error yang terjadi. Testing juga dapat digunakan untuk memastikan kevalid-an dalam proses input, sehingga dapat menghasilkan output yang sesuai.

5. *Maintenance* (Perawatan)

Tahap terakhir dari metode SDLC ini adalah *Maintenance*. Pada tahap ini jika sistem sesuai dengan tujuan yang ditentukan dan dapat menyelesaikan masalah, maka akan diberikan kepada pengguna. Setelah digunakan dalam periode tertentu pasti terdapat penyesuaian atau perubahan sesuai dengan keadaan yang diinginkan, sehingga membutuhkan perubahan terhadap sistem tersebut.

Namun dalam penelitian skripsi ini, tahapan *maintenance* tidak dilakukan karena penelitian ini hanya bertujuan untuk mengetahui bagaimana membangun sebuah sistem informasi presensi, jika kedepannya penelitian ini dikembangkan maka sebaiknya semua tahapan yang ada dalam metode SDLC ini dilakukan guna pengoptimalan penelitian sehingga hasil akhirnya dapat memberikan hasil yang sangat memuaskan.

3.3. Metode Pengumpulan Data

1) Metode Konsultasi

Metode konsultasi yaitu melakukan konsultasi permasalahan yang dibahas bersama dosen pembimbing atau orang lain yang mengerti tentang objek penelitian yang akan dilakukan.

2) Metode kepustakaan (*libary research*)

Metode pengumpulan data dan informasi dengan cara membaca buku atau artikel referensi yang dapat dijadikan acuan dalam penelitian ini.

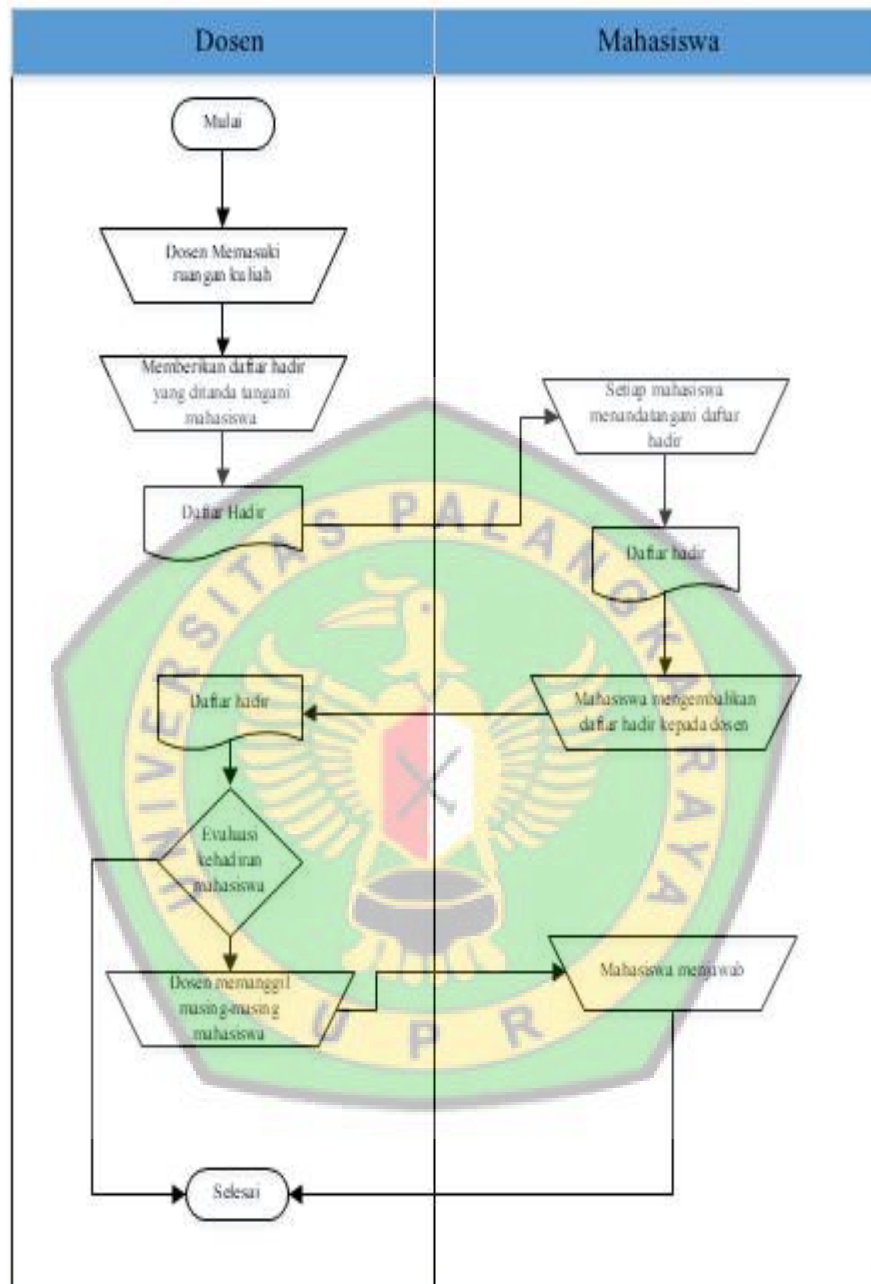
3.4 *Requirements Definition*

Requirement Definition pada tahap pertama ini yang dilakukan adalah mencari dan mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan yang merupakan bagian dari *requirements analysis and definition* (analisis kebutuhan dan definisi) pada model tersebut. Pada tahap ini, peneliti melakukan pencarian informasi dan menganalisis kebutuhan. Dalam merancang dan membangun sistem ini membutuhkan beberapa perangkat keras dan perangkat lunak yang harus dipenuhi.

3.4.1 **Analisis Sistem Lama**

Pada proses bisnis sistem lama, yang telah dijadikan sistem penerapan tata cara dan bagaimana sistem absen berjalan pada umumnya di kampus adalah sebagai berikut :

1. Dosen memasuki ruangan kuliah.
2. Dosen memberikan daftar hadir yang ditanda tangani mahasiswa.
3. Mahasiswa menandatangani kehadiran di daftar hadir perkuliahan.
4. Mahasiswa mengembalikan daftar hadir kepada dosen.
5. Dosen mengecek atau mengevaluasi kehadiran mahasiswa.
6. Proses absesnsi selesai.



Gambar 3.2 Flowchart sistem lama

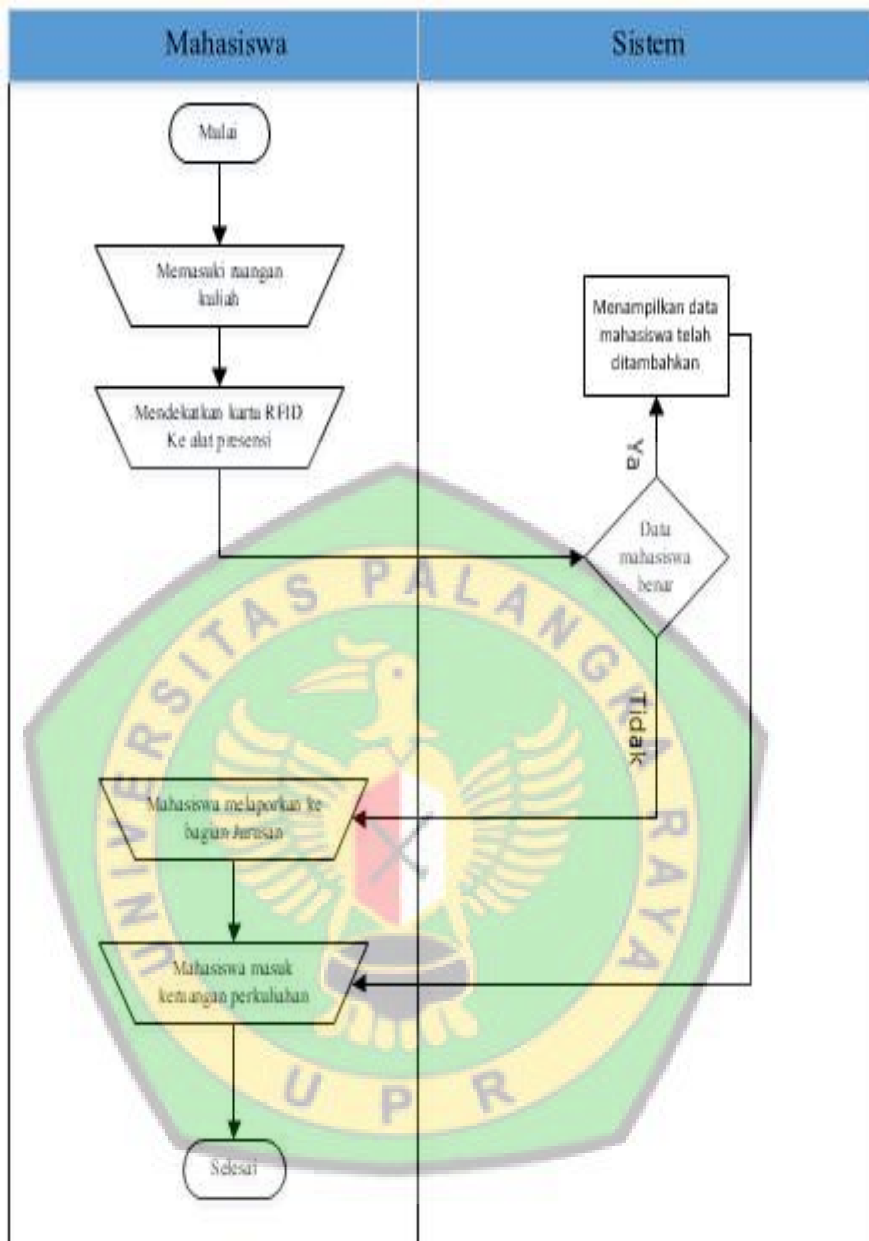
3.4.2 Analisis Sistem Baru

Pada bisnis proses sistem baru yang akan dijadikan sebagai bahan penelitian serta penerapan yang diharapkan optimal dan efisien untuk digunakan, maka dalam penelitian ini dirancang sebuah pola yang akan menjadi panduan dasar, bagaimana sistem baru akan berjalan sesuai rencana, yakni dengan poin-poin berikut ini :

A. Mahasiswa

Mahasiswa merupakan pengguna yang melakukan aktifitas pengisian presensi pada alat Rfid yang berada di ruang kelas yang akan dihadapinya, berikut ada lah bisnis proses dari mahasiswa :

1. Mahasiswa memasuki ruangan kuliah
2. Mahasiswa mendekatkan kartu Rfid ke alat presensi
3. Sistem akan memeriksa data mahasiswa, jika mahasiswa bersangkutan terdaftar di kelas dan mata kuliah tersebut maka alat presensi akan menampilkan data mahasiswa telah ditambahkan.
4. Jika tidak terdaftar mahasiswa melaporkan ke bagian jurusan, dan dapat melanjutkan masuk ke ruang perkuliahan.
5. Jika terdaftar mahasiswa langsung masuk keruang perkuliahan.

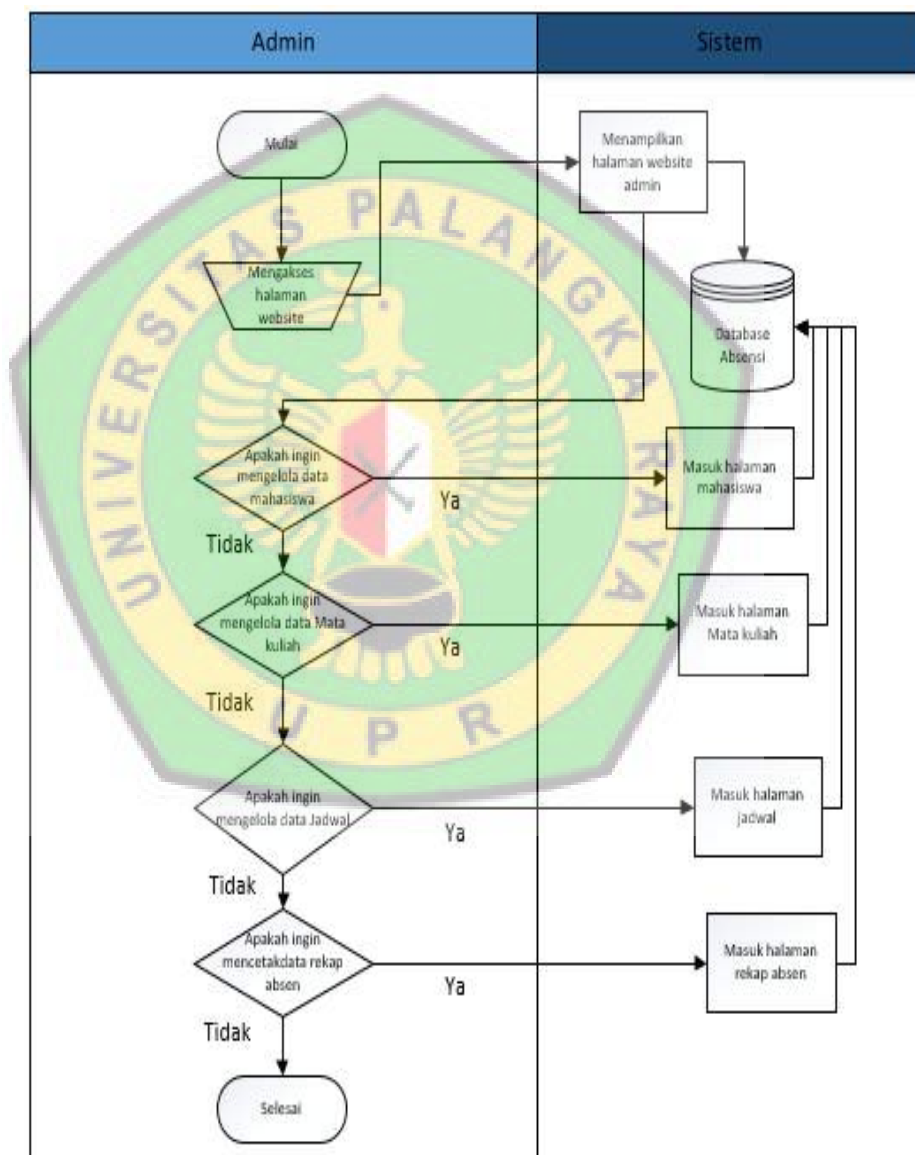


Gambar 3.3 Flowchart Sistem Baru Mahasiswa

B. Admin (*Website*)

Admin merupakan *user* yang mengelola data presensi mahasiswa. Admin memiliki tugas untuk mengelola sistem yang telah disediakan, data mahasiswa, mata kuliah, data jadwal dan rekap absen. Berikut ini merupakan bisnis proses dari admin :

1. Admin mengakses halaman *website*.
2. Admin mengelola data Mahasiswa.
3. Admin mengelola data Mata kuliah.
4. Admin mengelola data Jadwal.
5. Admin mencetak data rekap absen.



Gambar 3.4 Flowchart Sistem Baru Admin

3.7 Desain Sistem

Sistem arsitektur yang digunakan dalam proses desain aplikasi ini adalah dengan satu Bahasa yang konsisten untuk menentukan visualisasi, mengontruksi dan mendokumentasikan *artifact* (sepotong informasi yang digunakan atau dihasilkan dalam suatu proses rekayasa *software*, dapat berupa model, dan deskripsi) di mana desain sistem menggunakan *Flowchart* Aktor Pengguna dan Admin serta *Unified Modelling Language* (UML) sebagaimana dapat dijelaskan sebagai berikut :

3.7.1 Use Case Diagram

Use Case diagram akan menggambarkan fungsionalitas yang diharapkan dari sistem ini. Sebuah *Usecase* merepresentasikan sebuah interaksi antara aktor dengan sistem. Untuk mendefinisikan skenario dari penggunaan “Perancangan *Prototype* Presensi menggunakan E-ktm Berbasis *Radio Frequency Identification*” maka kegiatan *user* dibagi menjadi beberapa kegiatan yang dapat dilihat pada gambar *usecase* berikut.

A. Definisi Aktor

Aktor merupakan sesuatu yang berinteraksi dengan sistem untuk saling bertukar informasi. Pendefinisian aktor pada *website* ini berdasarkan pada tahap analisis pengguna (mahasiswa dan dosen).

Pendefisian aktor pada “Sistem Absensi Mahasiswa Berbasis Android (Studi Kasus: Program Pasca Sarjana Jurusan Magister Ilmu Ekonomi Universitas Palangka Raya)” dapat dilihat pada Tabel 3.1.

Tabel 3.1 Definisi Aktor

No	Aktor	Keterangan
1	Admin	Orang yang memegang kendali utama dalam mengelola Sistem absensi Mahasiswa.
2	Mahasiswa	Orang yang berperan sebagai pengguna (<i>user</i>) sistem absensi yang telah dikelola oleh admin.

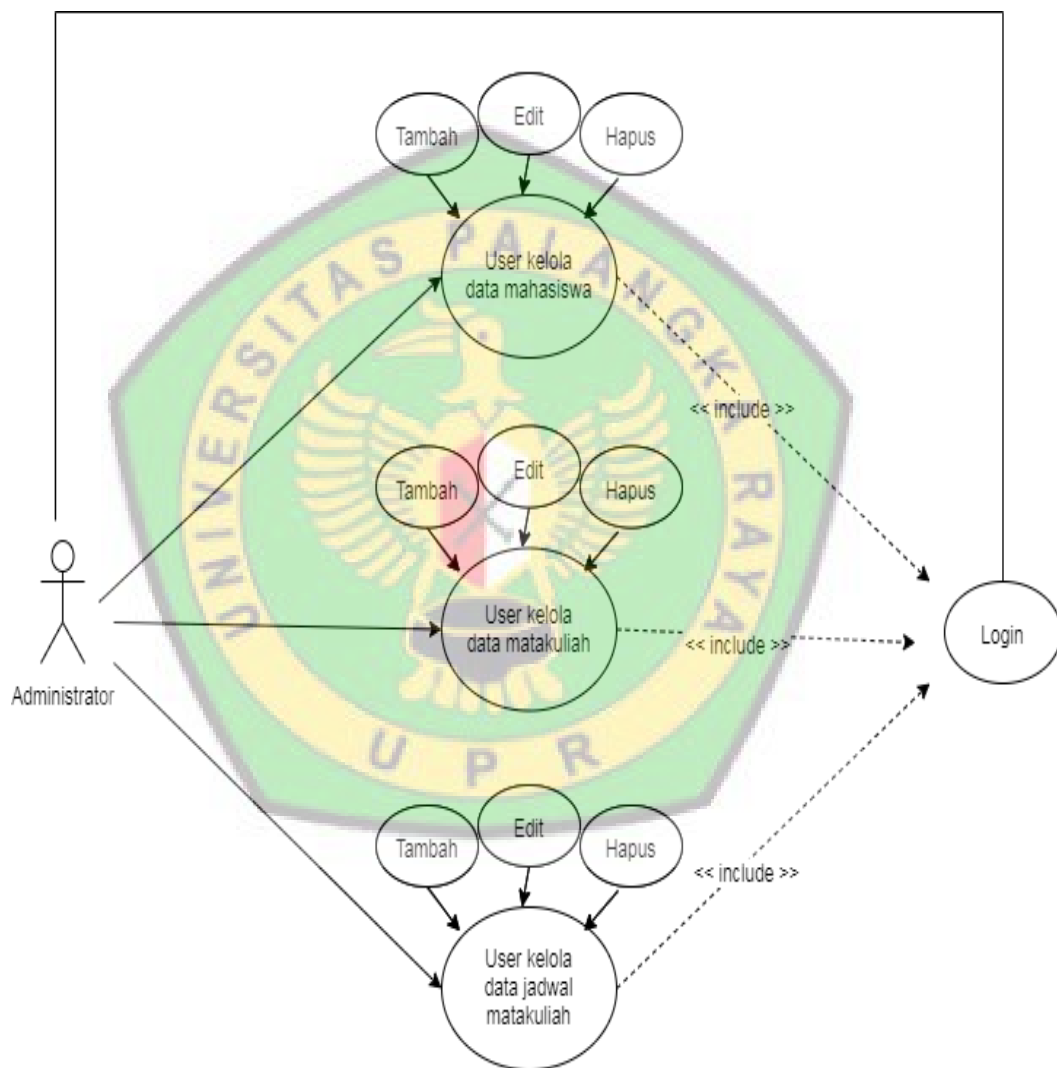
B. Definisi Use Case

Definisi *use case* pada “Sistem Absensi Mahasiswa Berbasis Android (Studi Kasus: Program Pasca Sarjana Jurusan Magister Ilmu Ekonomi Universitas Palangka Raya)” dibagi menjadi 3, yaitu admin, dosen, dan mahasiswa.

1) Definisi Use Case Admin**Tabel 3.2 Definisi Use Case Admin**

No.	Use Case	Keterangan
1	Mengakses website	Mengakses <i>website</i> 1. Lihat Home
3	Kelola Data Mahasiswa	Mengelola data Mahasiswa 1. Tambah Data Mahasiswa 2. Ubah data Mahasiswa 3. Lihat Data Mahasiswa 4. Hapus Data Mahasiswa
4	Kelola Mata Kuliah	Mengelola data mata kuliah 1. Tambah mata kuliah 2. Ubah mata kuliah 3. Lihat mata kuliah 4. Hapus mata kuliah
5	Kelola Jadwal	Mengelola data Jadwal 1. Tambah Jadwal

		2. Ubah Jadwal 3. Hapus Jadwal
6	Mencetak data rekap absen	Mengakses halaman Rekap Absen 1. Download Data rekap absen



Gambar 3.7 Use Case Admin

2) Definisi Use Case Mahasiswa

Tabel 3.4 Definisi Use Case Mahasiswa

No.	Use Case	Keterangan
1.	Kelola Pengisian absen	Menambah Pengisian absen

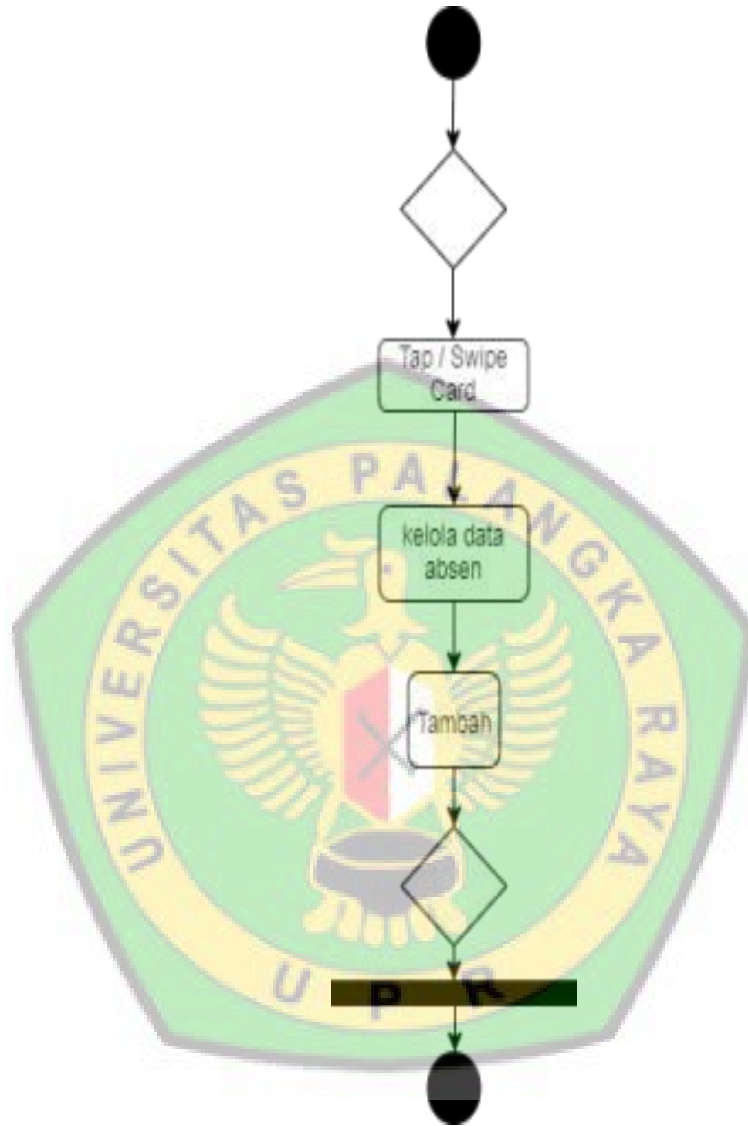


Gambar 3.8 Usecase Mahasiswa

3.7.2 Activity Diagram

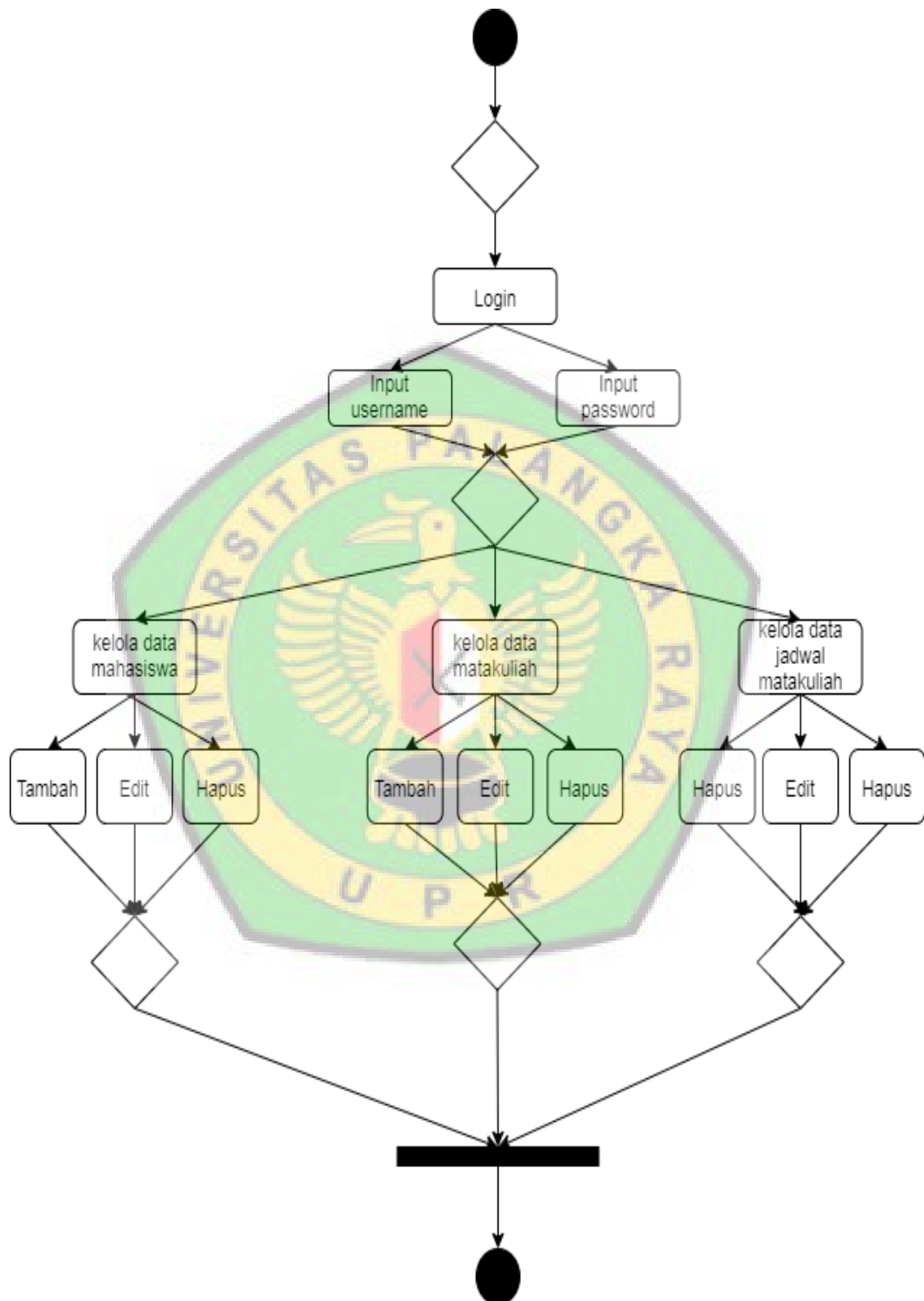
Activity diagram atau diagram aktivitas berfungsi untuk memodelkan alur kerja sebuah proses dan aktivitas dalam suatu proses. Dalam sistem Presensi dengan menggunakan *Radio Frequency Identification* ini, aktivitas yang dapat digambarkan adalah sebagai berikut :

1. Activity Diagram Mahasiswa



Gambar 3.9 Activity diagram Mahasiswa

2. Activity Diagram Adminitrator



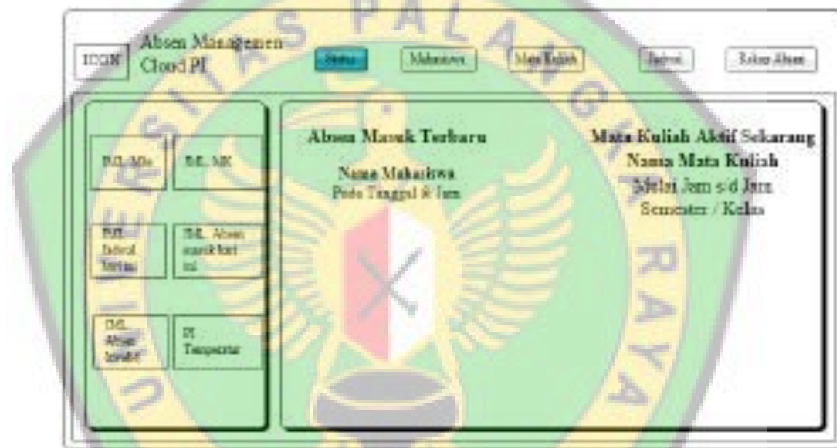
Gambar 3.10 Activity Diagram Administrator

3.8 Desain Interface

Setelah menganalisis kebutuhan dan perancangan maka langkah selanjutnya membangun *prototyping* sesuai dengan Metode *Prototype* yang digunakan yaitu desain *interface* tentang sistem absensi yang akan dibuat berikut desain *interface* yang dirancang untuk “*Perancangan Prototype Presensi menggunakan E-KTM berbasis Radio Frequency Identification*”.

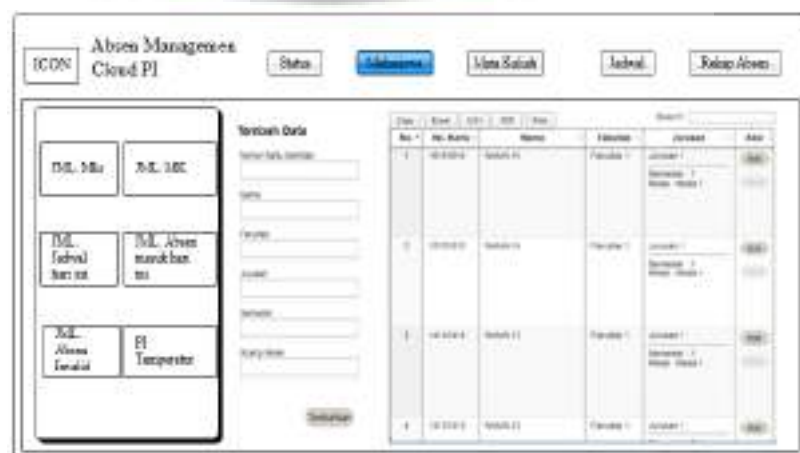
3.8.1 Desain Interface Admin

1. Desain interface Beranda admin



Gambar 3.12 Interface halaman Beranda Admin

2. Desain interface Menu Mahasiswa



Gambar 3.13 Interface Menu Mahasiswa

3. Desain interface halaman edit Mahasiswa

Editor

Absen Manajemen Cloud PI

SEARCH

NO. ID:

NAMA:

JENIS ABSEN:

ABSEN:

NO. ID:

NAMA:

JENIS ABSEN:

ABSEN:

NO. ID:

NAMA:

JENIS ABSEN:

ABSEN:

NO. ID:

NAMA:

JENIS ABSEN:

ABSEN:

No. ID	Nama	Jenis Absen	Absen
1001	Adi	Alasan 1	1001
1002	Budi	Alasan 2	1002
1003	Cici	Alasan 3	1003

Buttons:

Gambar 3.14 Interface halaman edit Mahasiswa

4. Desain interface halaman Mata Kuliah

Absen Manajemen Cloud PI

SEARCH

STATUS:

TAMBAH DATA

NO. ID:

NAMA MATA KULIAH:

ID (ID DIKLASIFIKASI):

ABSEN:

No. ID	Nama Mata Kuliah	ID (ID Diklasifikasi)	Absen
1	MK1	001	1001
2	MK2	002	1002
3	MK3	003	1003
4	MK4	004	1004
5	MK5	005	1005

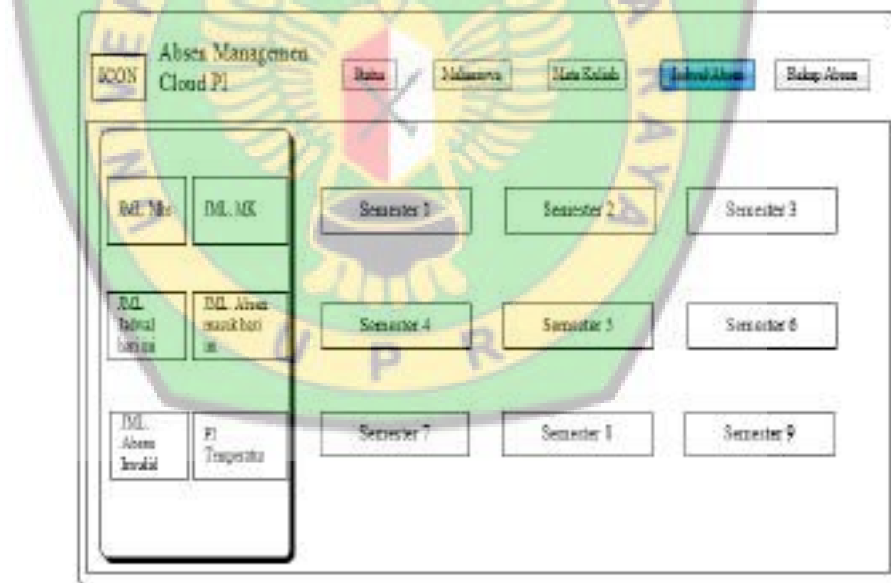
Gambar 3.15 Interface halaman Mata Kuliah

5. Desain interface halaman Mata Kuliah



Gambar 3.16 Interface halaman Mata Kuliah

6. Desain interface halaman Jadwal / Absen (Pilih Semester)

Gambar 3.17 Interface halaman Jadwal / Absen
(Pilih Semester)

7. Desain interface halaman Jadwal / Absen (Pilih Kelas)

**Gambar 3.18 Interface halaman Jadwal / Absen
(Pilih Kelas)**

8. Desain interface halaman Rekap Absen

Gambar 3.19 Interface halaman Rekap Absen

BAB IV

HASIL DAN PEMBAHASAN

4.1 *Prototype Construction and System Testing*

Setelah melakukan analisis dan desain selanjutnya yaitu penulisan program, sistem yang telah dibuat akan diimplementasikan. Sistem yang dibuat akan diuji coba terlebih dahulu, jika ada kesalahan, maka akan kembali ke metodologi sebelumnya yaitu pembuatan Kode Program sistem. Metode *testing* yang digunakan pada pembuatan sistem ini menggunakan Metode *Blackbox*.

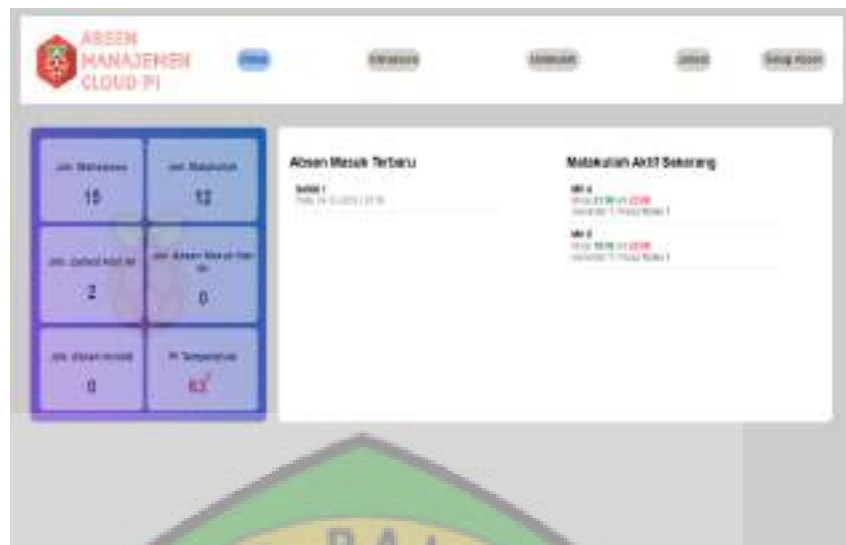
4.1.1 Prototype Construction

Tahapan Prototype construction adalah tahapan untuk membuktikan bahwa sistem yang dibangun telah berfungsi dengan baik, maka diperlukan skenario uji coba yang dapat menunjukkan bahwa hasil yang diperoleh dalam uji coba tersebut telah berjalan dengan benar dan sesuai dengan yang diharapkan.

A. Tampilan Halaman Website Admin

1. Halaman Home Admin

Interface halaman *Home* admin adalah halaman awal yang tampil saat admin mengakses alamat *web*. Berikut adalah *Interfacenya* :



Gambar 4.1 Halaman Home

2. Halaman Mahasiswa

Interface halaman Mahasiswa adalah halaman yang digunakan admin untuk proses tambah, edit dan hapus data mahasiswa. Berikut adalah *Interfacenya* :



Gambar 4.2 Halaman Mahasiswa

3. Halaman Edit Mahasiswa

Berikut adalah *Interface* pada proses edit data mahasiswa yang dilakukan oleh admin :

Gambar 4.3 Halaman Edit Mahasiswa

4. Halaman Mata kuliah

Interface halaman Mata kuliah adalah halaman yang digunakan admin untuk proses tambah, edit dan hapus data mata kuliah. Berikut adalah *Interfacenya* :

No.	Nama Mata Kuliah	SKS (Semester)	Waktu
1	Mat.	3	
2	Mat.	3	
3	Mat.	3	
4	Mat.	3	
5	Mat.	3	

Gambar 4.4 Halaman Daftar Mata Kuliah

5. Halaman Edit Matakuliah

Berikut adalah *Interface* halaman pada saat Edit Matakuliah dilakukan oleh admin :



Gambar 4.5 Halaman Edit Matakuliah

6. Halaman Jadwal (Pilih Semester)

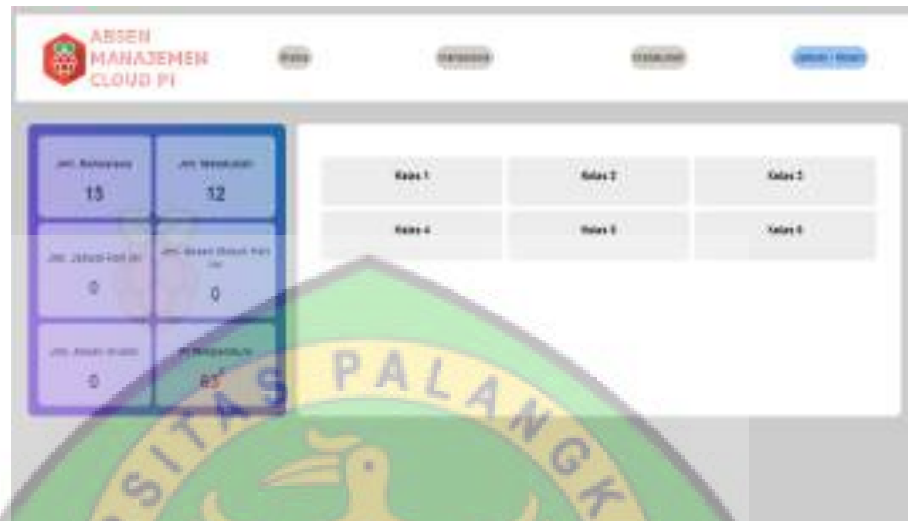
Berikut adalah *Interface* dari menu halaman Jadwal untuk memilih Semester yang dilakukan oleh admin :



Gambar 4.6 Halaman Jadwal (Pilih Semester)

7. Halaman Jadwal / Absen (Pilih Kelas)

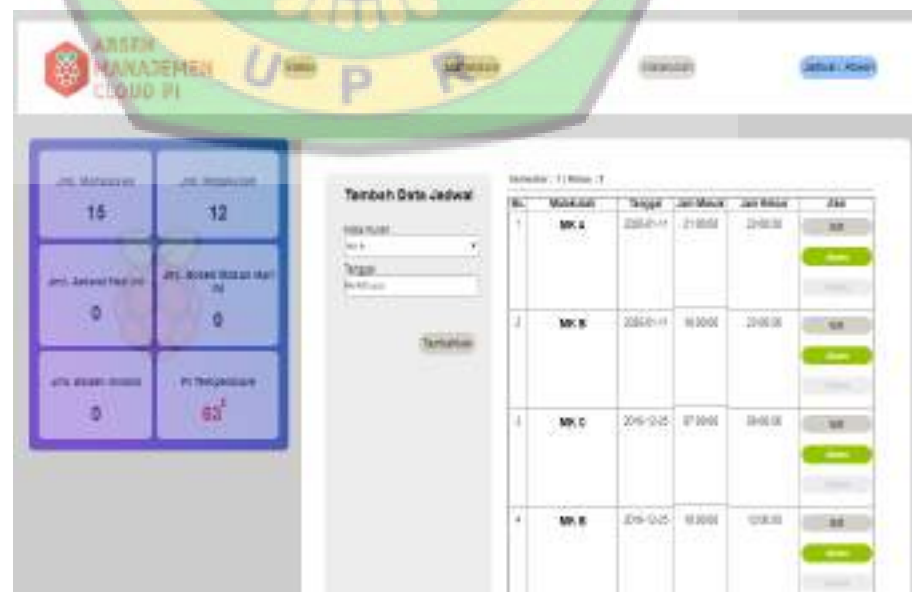
Berikut adalah *Interface* dari menu halaman Jadwal untuk memilih Kelas yang dilakukan oleh admin :



Gambar 4.7 Halaman Jadwal / Absen (Pilih Kelas)

8. Halaman Jadwal / Absen (Daftar Jadwal Mata Kuliah)

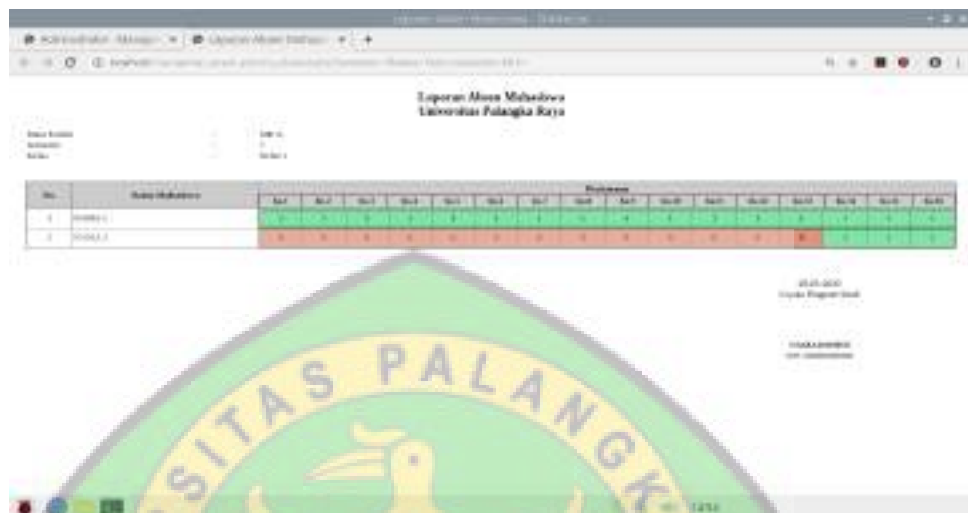
Berikut adalah *Interface* dari menu halaman Jadwal untuk memilih Semester yang dilakukan oleh admin :



Gambar 4.8 Halaman Jadwal / Absen (Daftar Jadwal Mata Kuliah)

9. Hasil Keluaran Rekap Presensi Mahasiswa

Berikut adalah *Interface* dari menu halaman Jadwal untuk memilih Semester yang dilakukan oleh admin :

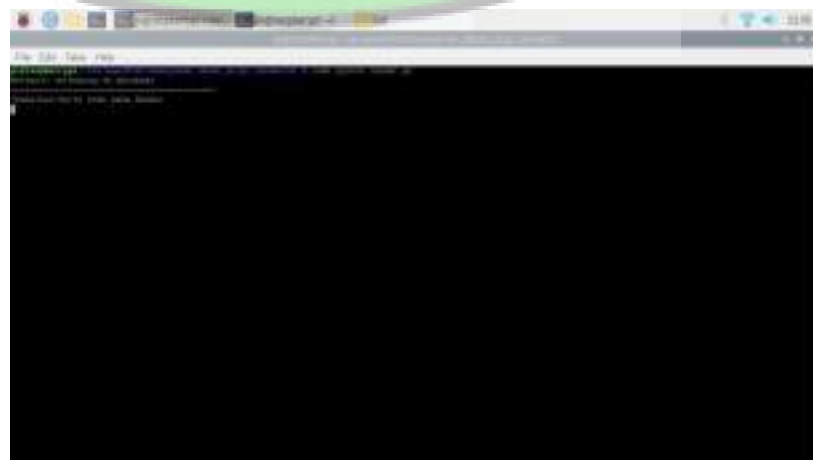


Gambar 4.9 Hasil Keluaran Rekap Presensi Mahasiswa

B. Tampilan Halaman Mahasiswa

1. Alat dinyalakan

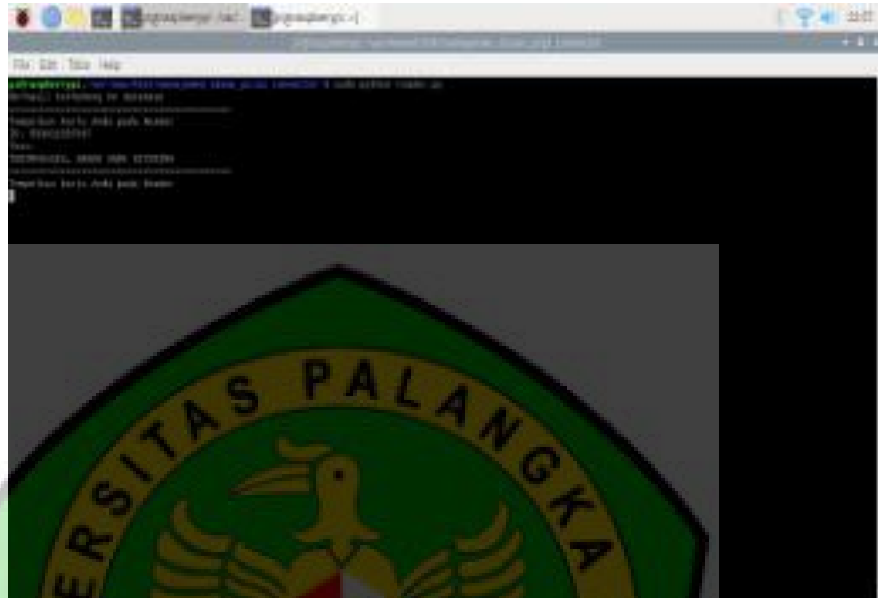
Berikut adalah *Interface* dari halaman Alat presensi untuk mahasiswa melakukan absen :



Gambar 4.10 Alat dinyalakan

2. Mahasiswa Melakukan absen

Berikut adalah *Interface* saat mahasiswa melakukan absen pada alat presensi :



Gambar 4.11 Mahasiswa melakukan absen

3. Absen berulang-ulang dengan kartu yang sama

Berikut adalah tampilan saat mahasiswa melakukan absen berulang-ulang :



Gambar 4.12 Mahasiswa Melakukan Absen Beruntun

4.2.1 System Testing

Setelah melakukan implementasi, dilakukan pengujian terhadap sistem dengan tujuan untuk melihat semua kesalahan dan kekurangan yang ada pada sistem. Pada pengujian ini fokus pada kebutuhan fungsional dan *outputnya* sesuai dengan yang diharapkan atau sebaliknya. Dalam pengujian sistem ini digunakan metode *Black Box* yaitu dilakukan dengan membuat kasus yang bersifat mencoba semua fungsi dari fungsi websitenya dan juga fungsi dari RFID, apakah sudah sesuai dengan spesifikasi yang dibutuhkan.

Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah. Berikut beberapa proses yang akan dilakukan pengujian yaitu :

1. *Black Box Testing* pada Website

Table 4.1 *Blackbox Testing Admin*

No	Pengguna	Halaman	Proses	Hasil
1.	ADMIN (Website)	Halaman Login	<ul style="list-style-type: none"> • <i>Login</i> tanpa <i>username</i> & <i>password</i> (tidak dapat masuk) • <i>Login</i> tanpa <i>username</i> (tidak dapat masuk) • <i>Login</i> tanpa <i>password</i> (tidak dapat masuk) • <i>Login</i> salah <i>username</i> (tidak dapat masuk) • <i>Login</i> salah <i>password</i> (tidak dapat masuk) • <i>Login</i> salah <i>username</i> & 	OK

			<p><i>password</i> (tidak dapat masuk)</p> <ul style="list-style-type: none"> • <i>Login Username</i> dan <i>Password</i> Benar (dapat masuk) 	
2.		Kelola data mahasiswa	<ul style="list-style-type: none"> • Menambah data mahasiswa tanpa mengisi data informasi apapun (data informasi tidak tersimpan) • Menambah data mahasiswa dengan mengisi data informasi secara sebagian (data informasi tersimpan) • Menambah data mahasiswa dengan mengisi data informasi secara menyeluruh (data informasi tersimpan) • Mengedit dan menyimpan data informasi yang telah ada secara sebagian (data informasi tersimpan) • Mengedit dan menyimpan data informasi yang telah ada secara menyeluruh (data informasi tersimpan) • Menghapus data (data 	OK

			<p>terhapus)</p> <ul style="list-style-type: none"> • Mencetak data mahasiswa (membuka halaman baru untuk proses cetak data informasi) 	
3.		Kelola data matakuliah	<ul style="list-style-type: none"> • Menambah data mahasiswa dengan mengisi data informasi secara menyeluruh (data informasi tersimpan) • Mengedit dan menyimpan data informasi yang telah ada secara sebagian (data informasi tersimpan) • Mengedit dan menyimpan data informasi yang telah ada secara menyeluruh (data informasi tersimpan) • Menghapus data (data terhapus) • Mencetak data matakuliah (membuka halaman baru untuk proses cetak data informasi) 	OK
4.		Kelola data	<ul style="list-style-type: none"> • Menambah data matakuliah tanpa 	OK

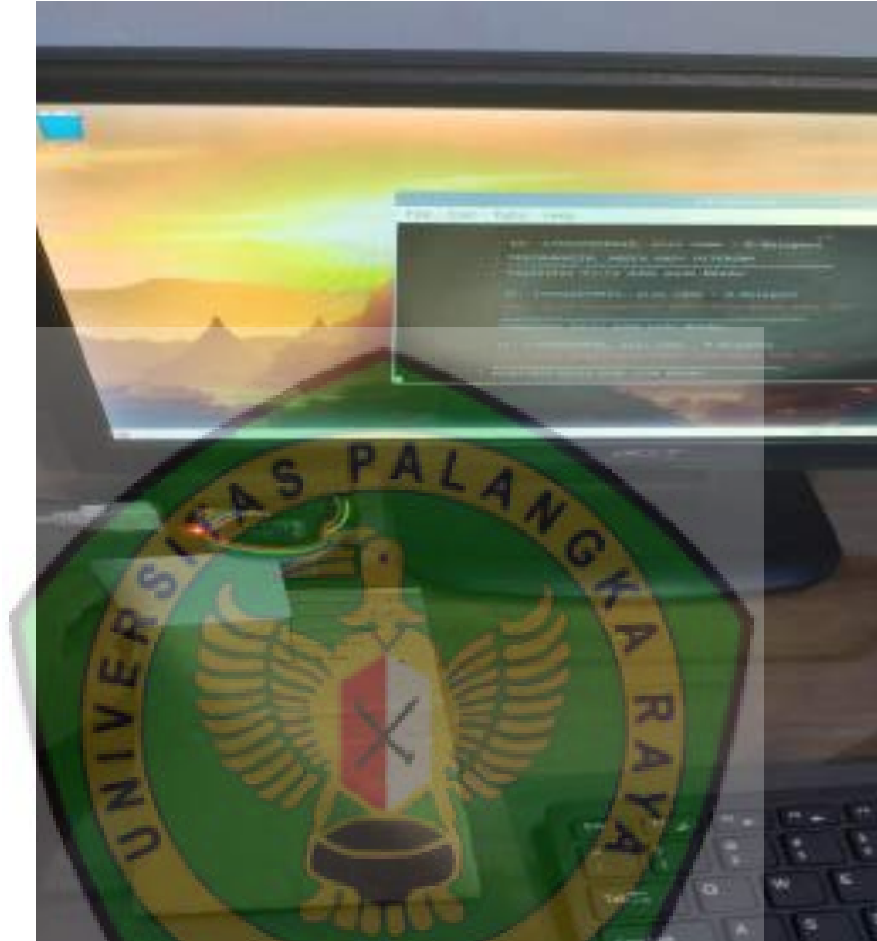
		jadwal / absen	<p>mengisi data informasi apapun (data informasi tidak tersimpan)</p> <ul style="list-style-type: none"> • Menambah data mahasiswa dengan mengisi data informasi secara menyeluruh (data informasi tersimpan) • Mengedit dan menyimpan data informasi yang telah ada secara sebagian (data informasi tersimpan) • Mengedit dan menyimpan data informasi yang telah ada secara menyeluruh (data informasi tersimpan) • Menghapus data (data terhapus) • Mencetak data jadwal kuliah (membuka halaman baru untuk proses cetak data informasi) • Mencetak data rekap absen (membuka halaman baru untuk proses cetak data informasi) 	
--	--	-------------------	--	--

2. Black Box Testing pada Mahasiswa

Table 4.2 Black Box Testing Mahasiswa

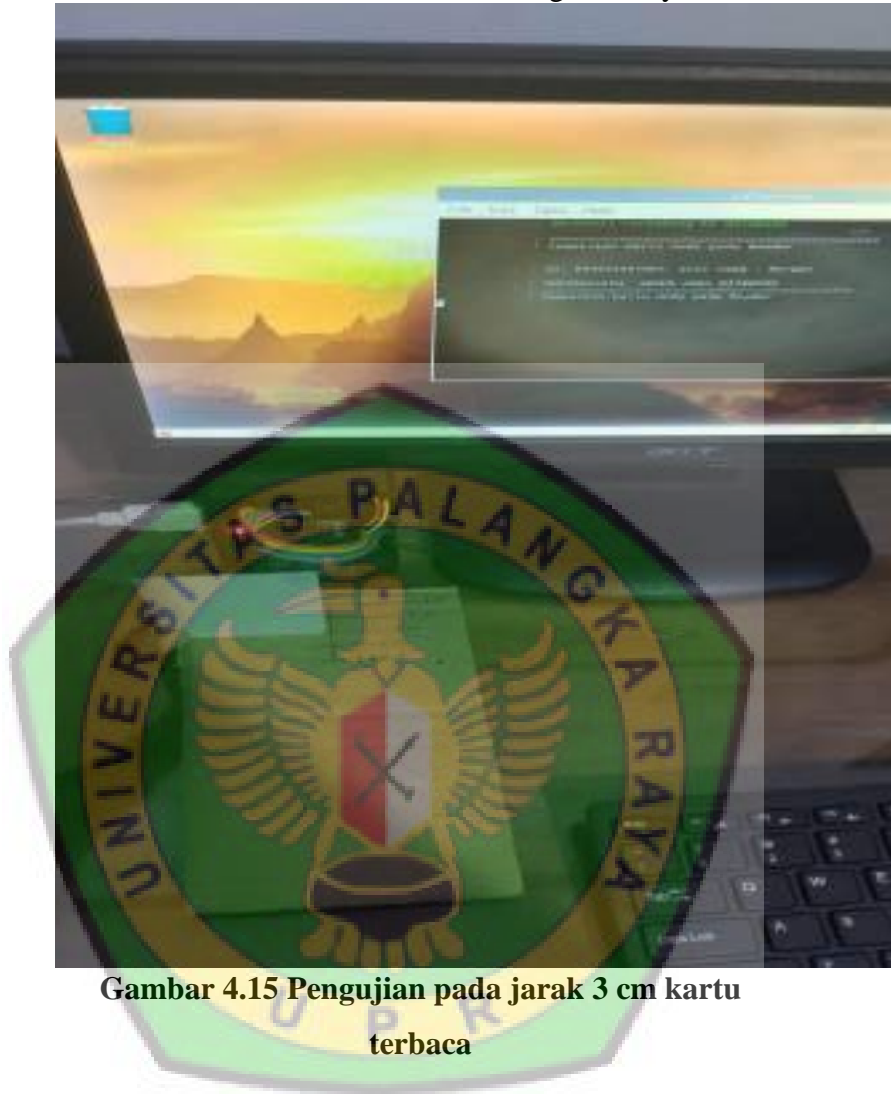
No.	Pengguna	Kegiatan	Proses	Hasil
1	Aplikasi (Mahasiswa)	Mengisi daftar hadir (absen)	<ul style="list-style-type: none"> • Menempelkan kartu pada alat pindai dengan jarak LEBIH DARI 3 cm (Sistem tidak dapat membaca kartu RFID) • Menempelkan kartu pada alat pindai dengan jarak KURANG DARI 3 cm (Sistem membaca identitas kartu RFID berupa nomor / angka yang telah ditentukan untuk masing-masing kartu absen mahasiswa) 	OK OK

2. Pengujian pembacaan alat pada jarak 2 cm kartu dapat terbaca oleh alat Rfid Reader, berikut gambarnya :



**Gambar 4.14 Pengujian pada jarak 2 cm kartu
terbaca**

3. Pengujian pembacaan alat pada jarak 3 cm kartu dapat masih dapat terbaca oleh alat Rfid Reader, berikut gambarnya :



4. Pengujian pembacaan alat pada jarak 4 cm kartu tidak terbaca oleh alat Rfid Reader, berikut gambarnya :



**Gambar 4.16 Pengujian pada jarak 4 cm kartu
Tidak terbaca**

5. Pengujian pembacaan alat pada jarak 5 cm kartu tidak terbaca oleh alat Rfid Reader, berikut gambarnya :



**Gambar 4.17 Pengujian pada jarak 5 cm kartu
Tidak terbaca**

Kesimpulan yang dapat diambil dari pengujian jarak pembacaan kartu pada penelitian ini yaitu, sebagai berikut :

1. RFID reader dapat membaca RFID tag dengan baik dalam jangkauan maksimal 3 cm.
2. Proses input data dari simpan, edit, hapus dan export data dapat bekerja dengan baik.
3. Proses presensi dapat bekerja dengan baik tanpa ada kesalahan membaca ID.
4. Secara keseluruhan sistem presensi dapat bekerja dengan baik dari mulai input data sampai proses presensi.



BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil yang didapat dari pembuatan *Perancangan Prototype presensi menggunakan E-ktm berbasis Radio Frequency Identification*, maka dapat ditarik kesimpulan untuk menjawab rumusan masalah, di antaranya sebagai berikut :

1. Perancangan *Prototype* menggunakan UML, dengan diagram yang digunakan dalam desain sistem adalah Diagram *Use Case*, Diagram *Activity* dan Diagram *Class*.
2. Bahasa pemrograman yang digunakan adalah PHP (*Hypertext Preprocessor*) dan HTML (*Hypertext Markup Language*), sedangkan untuk manajemen data dan mendukung beberapa plugin menggunakan bahasa pemrograman *Python* dan *jQuery*.
3. Untuk menguji fungsionalitas dari aplikasi sudah berjalan sesuai dengan hasil yang diharapkan, dilakukan pengujian menggunakan metode *BlackBox Testing*.
4. Percobaan simulasi pada *prototype* presensi ini penulis mendapatkan hasil yaitu :
 - a. Pada proses pengiriman data, saat mahasiswa melakukan presensi data mahasiswa akan diproses dengan sangat cepat di *localhost* kemudian menuju *localhost/web admin*, lalu masuk ke *database server*, kemudian

tampil pemberitahuan bahwa mahasiswa telah berhasil melakukan pengisian presensi.

5. Kesimpulan yang dapat diambil dari pengujian jarak pembacaan kartu pada penelitian ini yaitu, sebagai berikut :

- a. RFID *reader* dapat membaca RFID *tag* dengan baik dalam jangkauan maksimal 3 cm.
- b. Proses *input* data dari simpan, edit, hapus dan *export* data dapat bekerja dengan baik.
- c. Proses presensi dapat bekerja dengan baik tanpa ada kesalahan membaca ID.
- d. Secara keseluruhan sistem presensi dapat bekerja dengan baik dari mulai *input* data sampai proses presensi.

5.2 Saran

Perancangan Prototype presensi menggunakan E-ktm berbasis Radio Frequency Identification ini masih memiliki kekurangan diantaranya adalah sebagai berikut :

1. Untuk penelitian lebih lanjut dari *prototype* ini, bisa dilakukan dengan data nyata di lapangan dan menambahkan fitur-fitur pendukung dalam kenyamanan presensi maupun input data.

2. Pengembangan sistem presensi menggunakan teknologi RFID selanjutnya dianjurkan dengan meningkatkan keamanan untuk menghindari mahasiswa yang masih melakukan tindak kecurangan dalam hal menitipkan kartu RFID untuk melakukan presensi.
3. Pengembangan sistem kedepannya diharapkan dapat dilakukan dengan jarak pembacaan yang jauh sehingga mahasiswa tidak perlu mendekatkan kartu pada alat *reader*.
4. Untuk kedepannya sebaiknya menggunakan *reader* yang juga dapat membaca frekuensi yang ada di dalam KTP, sehingga KTP mahasiswa dapat dijadikan sebagai kartu mahasiswa.



DAFTAR PUSTAKA

- Adam, W., Sagala, L., 2014, Sistem Absensi Pegawai Menggunakan Teknologi RFID,
- Adi Nugroho. 2009. Rekayasa Perangkat Lunak Menggunakan UML dan Java ANDI : Yogyakarta.
- Ardaninggar, E. A., 2016, *Sistem Keamanan Portal Perumahan Berbasis RFID*, Tugas Akhir, Jurusan Teknik Elektro, FST, Universitas Sanata Dharma, Yogyakarta.
- Al Fatta, Hanif. (2009). *Analisis dan Perancangan Sistem Informasi untuk keunggulan Bersaing Perusahaan dan Organisasi Modern*. Yogyakarta : Andi
- Betha Sidik, Ir., (2004). Pemrograman Web dengan PHP. Informatika, Bandung
- Booch, G., James Rumbaugh, Ivar Jacobson. The Unified Modeling Language User Guide, Addison Wesley, New York, 2005.
- Carolyn, E. Begg and Thomas, M. Connolly. 2005. Database Systems: A Practical Approach to Design, Implementation and Management. University of Paisley : Addison Wesley.
- Fatoni, M. Rancang Bangun Absensi Mahasiswa Menggunakan Rfid Dengan Komunikasi Terpusat. Surabaya STIKOM Surabaya; 2013.
- Julian Onibala, Arie S.M. Lumenta, ST. MT, Brave A. Sugiarto, ST. MT, Perancangan Radio Frequency Identification (RFID). 2015:
- Jogiyanto, HM. 2001. Analisis Perancangan Sistem Informasi. Yogyakarta : Andi Offset
- Leitch Robert A., K. Roscoe Davis. 2005. Analisis & Desain. Andi. Yogyakarta.
- Setiawan, E. B., & Kurniawan, B. Perancangan Sistem Absensi Kehadiran Perkuliahan dengan Menggunakan Radio Frequency Identification (RFID). Bandung Universitas Komputer Indonesia; 2015.
- Supriyanto. (2009). Metodologi Riset Bisnis. Indeks, Jakarta.
- Silberschatz, Abraham. dkk. (2002). Applied Operating Systems, Sixth Edition. John Wiley & Sons.

Ramakrishnan dan Gehrke. 2003. Database Management Systems.

http://repository.dinamika.ac.id/id/eprint/2282/4/BAB_II.pdf diakses 06 Juni 2020

<https://www.kompasiana.com/afsee/55005a74a3331159735106bf/absensi-atau-presensi> diakses 30 September 2019

<http://ejournal.lpkia.ac.id/files/students/essays/journals/301.pdf>, diakses 29 September 2019

https://www.academia.edu/32453408/SISTEM_ABSENSI_MENGGUNAKAN_TEKNOLOGI_RFID_TUGAS_AKHIR, diakses 28 September 2019

<https://ejournal.poltektegal.ac.id/index.php/informatika/article/view/604>, diakses 28 September 2019

https://www.academia.edu/8838867/Rancang_Bangun_Prototipe_Sistem_Absensi_Otomatis_dengan_Teknologi_RFID, diakses 28 September 2019

