

D3 658.312 2

Nur
R

No. Dok: 6759

Copy

SUMBANGAN ALUMNI

**RANCANG BANGUN SISTEM INFORMASI CUTI
KARYAWAN DI DIVISI SUMBER DAYA MANUSIA PADA PT
INDONESIA KENDARAAN TERMINAL BERBASIS
ANDROID MENGGUNAKAN ANDROID STUDIO 3.3.2 DAN
FIREBASE**

TUGAS AKHIR

Untuk Memenuhi Syarat Penyelesaian Program Sarjana Terapan
Program Studi Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta

Disusun Oleh

Meisya Regita Nurafridha

1315095



DATA BUKU PERPUSTAKAAN	
Tgl Terima	01/08/22
No Induk Buku	527/STI/58/TA/22

**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN RI
JAKARTA
2019**

No. Do

**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN RI**

TANDA PERSETUJUAN DOSEN PEMBIMBING

JUDUL TUGAS AKHIR:

**RANCANG BANGUN SISTEM INFORMASI CUTI KARYAWAN DI
DIVISI SUMBER DAYA MANUSIA PADA PT INDONESIA
KENDARAAN TERMINAL BERBASIS ANDROID MENGGUNAKAN
ANDROID STUDIO 3.3.2 DAN FIREBASE**

Disusun Oleh:

Nama	:	Meisyah Regita Nurafridha
Nim	:	1315095
Program Studi	:	Sistem Informasi Industri Otomotif
Tanggal Seminar	:	7 Agustus 2019
Tanggal Sidang	:	21 Agustus 2019
Tanggal Lulus	:	21 Agustus 2019

Telah diperiksa dan disetujui untuk diajukan dan dipertahankan dalam
Ujian Tugas Akhir Politeknik STMI Jakarta

Jakarta, 21 Agustus 2019

Dosen Pembimbing,



Noveriza Yuliasari, M.T.
NIP. 197811212009012003

**POLITEKNIK STMI JAKARTA
KEMENTERIAN PERINDUSTRIAN RI**

LEMBAR PENGESAHAN

JUDUL TUGAS AKHIR:

**RANCANG BANGUN SISTEM INFORMASI CUTI KARYAWAN DI
DIVISI SUMBER DAYA MANUSIA PADA PT INDONESIA
KENDARAAN TERMINAL BERBASIS ANDROID MENGGUNAKAN
ANDROID STUDIO 3.3.2 DAN FIREBASE**

Disusun Oleh:

Nama : Meisyah Regita Nurafidha

Nim : 1315095

Program Studi : Sistem Informasi Industri Otomotif

Telah diuji oleh Tim Penguji Sidang Tugas Akhir Program Studi Sistem Informasi Industri Otomotif Politeknik STMI Jakarta Kementerian Perindustrian R.I. pada hari Rabu, 21 Agustus 2019.

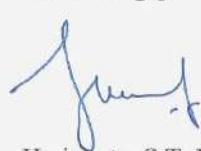
Jakarta, 21 Agustus 2019

Dosen Pembimbing



Noveriza Yuliasari M.T
NIP. 197811212009012003

Ketua Penguji



Lucky Heriyanto, S.T, M.TI
NIP.197908202009011009

Dosen Penguji



Dedy Trisanto, S.Kom, MMSI
NIP.197805052005021002

Dosen Penguji



Fifi L. Hadianastuti, S.Kom, M.Kes
NIP.197310162005022001

LEMBAR BIMBINGAN PENYUSUNAN TUGAS AKHIR

Nama : Meisya Regita Nurafridha
NIM : 1315095
Judul Tugas Akhir : Rancang Bangun Sistem Informasi Cuti Karyawan di Divisi Sumber Daya Manusia Pada PT Indonesia Kendaraan Terminal Berbasis Android Menggunakan Android Studio 3.3.2 dan Firebase
Pembimbing : Noveriza Yuliasari, S.Si, M.T.

Tanggal	Keterangan	Paraf
01/04/2019	Mengajukan Proposal Tugas Akhir	✓
22/04/2019	Bimbingan Bab I, II dan III	✓
08/05/2019	Revisi Bab I, II dan III	✓
24/06/2019	Revisi Bab I, II, III dan Pengajuan Bab IV	✓
03/07/2019	Pengajuan Bab V	✓
16/07/2019	Demo Program	✓
19/07/2019	Revisi Bab V	✓
25/07/2019	Revisi Bab V	✓
30/07/2019	Revisi Bab V	✓
31/07/2019	Pengecekan Bab I, II, III, IV, V, VI dan Kelengkapan	✓

Mengetahui,
Ketua Program Studi
Sistem Informasi Industri Otomotif

Pembimbing



Noveriza Yuliasari, S.Si, M.T
NIP. 197811212009012003



Noveriza Yuliasari, S.Si, M.T
NIP. 197811212009012003

LEMBAR PERNYATAAN KEASLIAN

Saya yang bertanda tangan di bawah ini:

Nama : Meisya Regita Nurafridha

NIM : 1315095

Berstatus sebagai mahasiswa Program Studi Sistem Informasi Industri Otomotif di Politeknik STMI Jakarta Kementerian Perindustrian Republik Indonesia. Dengan ini menyatakan bahwa hasil karya Tugas Akhir yang saya buat dengan judul:

“RANCANG BANGUN SISTEM INFORMASI CUTI KARYAWAN DI DIVISI SUMBER DAYA MANUSIA PADA PT INDONESIA KENDARAAN TERMINAL BERBASIS ANDROID MENGGUNAKAN ANDROID STUDIO 3.3.2 DAN FIREBASE”

1. **Dibuat** dan diselesaikan sendiri, dengan menggunakan literatur hasil kuliah, *survey* lapangan, dibantu oleh dosen pembimbing maupun asisten dosen pembimbing, serta buku-buku maupun jurnal-jurnal ilmiah yang menjadi bahan acuan yang tertera dalam referensi karya Tugas Akhir ini.
2. **Bukan** merupakan hasil duplikasi karya tulis yang sudah dipublikasikan atau yang pernah dipakai sebelumnya untuk mendapatkan gelar sarjana di Universitas/Perguruan Tinggi lain, kecuali yang telah disebutkan sumbernya dan dicantumkan pada referensi karya Tugas Akhir ini.
3. **Bukan** merupakan karya tulis hasil terjemahan dari kumpulan buku atau jurnal acuan yang tertera dalam referensi karya Tugas Akhir ini.

Jika terbukti saya tidak memenuhi apa yang telah saya nyatakan di atas, maka saya bersedia menerima sanksi atas apa yang telah saya lakukan sesuai dengan peraturan yang berlaku.

Jakarta, 01 Agustus 2019


Meisya Regita Nurafridha

ABSTRAK

PT Indonesia Kendaraan Terminal adalah sebuah anak perusahaan Pelabuhan Indonesia (PELINDO) yang bergerak dibidang jasa ekspor-impor kendaraan. PT Indonesia Kendaraan Terminal senantiasa meningkatkan produktivitas perusahaan. Proses pengelolaan manajemen karyawan menjadi salah satu elemen terpenting dalam menunjang kegiatan dan loyalitas karyawan. Salah satu kegiatan dalam pengelolaan manajemen karyawan adalah proses pengelolaan cuti karyawan. *Output* yang berkaitan dengan pengelolaan cuti karyawan adalah karyawan diizinkan cuti dan laporan cuti setiap karyawan. Kegiatan yang dilakukan dalam proses pengelolaan cuti karyawan antara lain membuat pengajuan cuti, memberikan persetujuan atau penolakan, serta merekap data laporan cuti di divisi SDM. Kegiatan peng- *input*-an data dilakukan oleh karyawan yang ingin mengajukan cuti masih dilakukan secara manual hingga proses perekapan data masih secara manual. Terkadang sulit meminta persetujuan atasan, sulit mencari data laporan cuti sehingga harus menanyakan kembali kepada Staff SDM. Kesulitan yang dirasakan dapat menimbulkan keterhambatan pengajuan cuti karyawan dan kesalahan rekap data. Oleh karena itu, diperlukan sistem informasi yang dapat membantu perusahaan dalam memperbaiki masalah pada Divisi Sumber Daya Manusia. Sistem informasi pengelolaan cuti karyawan ini menggunakan metode analisis kebutuhan sistem, Metode pengembangan *waterfall*, analisis dan desain berorientasi objek dengan alat bantu analisis desain *Unified Modeling Language* (UML). Pemodelan data menggunakan *Entity Relationship Diagram*, *Conceptual Data Model*, serta perancangan interface menggunakan *Windows Navigation Diagram*. Sistem informasi ini berbasis android dengan menggunakan Android Studio 3.3.2 dan *Firebase Database*. Dengan adanya sistem informasi pengelolaan cuti ini proses pengajuan cuti karyawan dapat berjalan dengan baik tanpa ada keterhambatan

Kata Kunci : Sistem Informasi, pengelolaan cuti karyawan, Android, *Firebase*

KATA PENGANTAR

Segala puji syukur penulis panjatkan kepada Allah SWT, pemelihara seluruh alam raya, yang atas limpahan rahmat, taufik dan hidayah-Nya, penulis mampu menyelesaikan penulisan Tugas Akhir dengan baik.

Penulisan Tugas Akhir ini disusun untuk memenuhi sebagian syarat penyelesaian program sarjana terapan Sistem Informasi Industri Otomotif pada Politeknik STMI Jakarta dengan judul **“RANCANG BANGUN SISTEM INFORMASI CUTI KARYAWAN DI DIVISI SUMBER DAYA MANUSIA PADA PT INDONESIA KENDARAAN TERMINAL BERBASIS ANDROID MENGGUNAKAN ANDROID STUDIO 3.3.2 DAN FIREBASE”**.

Penulis menyadari bahwa Tugas Akhir ini tidak terwujud tanpa adanya bantuan, bimbingan, dukungan, doa dan cinta dari berbagai pihak. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT, yang telah memberikan rahmat dan karunia-Nya.
2. Kedua orang tua, Bapak Isya Ashari dan Ibu Muryani serta adik tercinta Anisya Agustia Nurmasita dan Muhammad Aghil Raihansya yang telah memberi motivasi dan kasih sayang yang penuh sehingga Tugas Akhir dapat diselesaikan dengan baik.
3. Ibu Noveriza Yuliasari, M.T selaku Ketua Program Studi Sistem Informasi Industri Otomotif dan sekaligus Dosen Pembimbing yang telah memberikan saran, motivasi dan kesempatan sehingga saya dapat mengerjakan Tugas Akhir ini dengan baik dan benar.
4. Ibu Triana Fatmawati, ST,MT Selaku dosen wali selama melakukan kegiatan perkuliahan yang selalu memberikan dukungan dan motivasi sehingga Tugas Akhir ini dapat selesai dengan baik.
5. Bapak Yanto Hendrawan, Bapak Fadli, Bapak Syarif, Bapak Heri dan Ibu Trisya selaku pembimbing di Divisi Teknik dan Sistem Informasi dan Divisi Sekretaris Perusahaan PT Indonesia Kendaraan Terminal yang telah

- memberikan pengetahuan baik secara lisan, tulisan, maupun dokumen terkait penulis dalam menjalankan Praktik Kerja Lapangan.
6. Bapak Dodi dan Bapak Irfan selaku Divisi Sumber Daya Manusia di PT Indonesia Kendaraan Terminal yang telah memberikan informasi dan mengarahkan selama melakukan Praktik Kerja Lapangan di PT Indonesia Kendaraan Terminal.
 7. Seluruh dosen Politeknik STMI Jakarta yang telah memberikan ilmunya guna menambah pengetahuan dan pengalaman yang sangat bermanfaat.
 8. Abu Hasan, Jimly Ashiddqi, Anisa Yuliandari, Andriyani Putri, Sofwatun Nida dan Riztika Khairani yang selalu memberi bantuan, motivasi dan semangat dalam menyelesaikan Tugas Akhir.
 9. Teman-teman seperjuangan prodi Sistem Informasi Industri Otomotif dan terkhusus Jihan Shafyra, Ristya Alsabila, Dwi Mulyati, Intan Rinjani, Dea Aulia, Sera Puji, Cut Kumala, Riski Nur Africanadi, Adinda Rachmasari, dan Syafitri yang selalu memberi bantuan, motivasi dan semangat dalam menyelesaikan laporan tugas akhir ini.
 10. Teman-teman Alumni SDN 25 Mempawah Hilir angkatan 2009 dan teman-teman alumni Darunnajah Islamic Boarding School angkatan 38 yang selalu memberi motivasi dan semangat dalam menyelesaikan laporan tugas akhir ini.

Semoga Allah SWT membalas kebaikan dan ketulusan semua pihak yang telah membantu menyelesaikan laporan ini dengan melimpahkan rahmat dan karunia-Nya.

Semoga Tugas Akhir ini dapat memberikan manfaat dan memberikan informasi yang berguna bagi pembaca. Terima kasih.

Jakarta, 21 Agustus 2019

Meisya Regita Nurafridha

DAFTAR ISI

	Halaman
Judul	i
Tanda Persetujuan Dosen Pembimbing	ii
Lembar Pengesahan	iii
Lembar Bimbingan Tugas Akhir	iv
Lembar Pernyataan Keaslian.....	v
Kata Pengantar	viii
Daftar Isi.....	x
Daftar Gambar.....	xiv
Daftar Tabel	xvii
Daftar Lampiran	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Pokok Permasalahan	2
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah	3
1.5 Manfaat Tugas Akhir	4
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	6
2.1 Rancang Bangun.....	6
2.2 Konsep Dasar Sistem	6
2.2.1 Karakteristik Sistem.....	7
2.2.2 Klasifikasi Sistem.....	9
2.3 Konsep Dasar Informasi	11
2.3.1 Fungsi dan Siklus Informasi	13
2.3.2 Nilai dan Kualitas Informasi.....	14
2.4 Sistem Informasi.....	16
2.5 System Development life Cycle (SDLC)	18

2.6 Metode Pengembangan <i>Waterfall</i>	21
2.7 Karyawan.....	22
2.8 Cuti.....	23
2.9 <i>Flowchart Mapping (Flowmap)</i>	25
2.10 <i>Unified Modelling Languange (UML)</i>	27
2.10.1 <i>Use Case Diagram</i>	29
2.10.2 <i>Activity Diagram</i>	30
2.10.3 <i>Sequence Diagram</i>	33
2.10.4 <i>Class Diagram</i>	35
2.10.5 <i>Deployment Diagram</i>	36
2.11 <i>DataBase</i>	37
2.12 <i>Entity Relationship Diagram (ERD)</i>	38
2.13 <i>Conceptual Data Model (CDM)</i>	39
2.14 <i>Phisycal Data Model (PDM)</i>	42
2.15 Kamus Data.....	42
2.16 <i>Windows Navigation Diagram (WND)</i>	43
2.17 Java.....	44
2.18 Android	46
2.18.1 Arsitektur Android.....	46
2.18.2 Struktur Aplikasi Android	47
2.18.3 Versi Android.....	48
2.19 Android Studio.....	49
2.20 <i>Firebase</i>	49
2.21 <i>Blackbox Testing</i>	53
BAB III METODOLOGI PENELITIAN	55
3.1 Metodologi Penelitian.....	55
3.2 Jenis dan Sumber Data.....	55
3.3 Pengumpulan Data	56
3.4 Metode Pengembangan Sistem.....	57
3.5 Kerangka Penelitian	57

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA	62
4.1 Sejarah Perusahaan.....	62
4.1.1 Profil Perusahaan.....	63
4.1.2 Visi, Misi, Nilai-Nilai dan Budaya Perusahaan	63
4.2 Struktur Organisasi Perusahaan.....	65
4.3 Struktur Organisasi Divisi Sumber Daya Manusia (SDM).....	67
4.4 Tenaga Kerja.....	68
4.4.1 Tenaga Organik	68
4.4.2 Tenaga Non Organik.....	69
4.5 Pengajuan Cuti.....	69
4.6 Analisis Dokumen Cuti	71
4.7 Prosedur Sistem yang Sedang Berjalan.....	76
BAB V ANALISIS DAN PEMBAHASAN.....	78
5.1 Analisis Kebutuhan Sistem	78
5.2 Prosedur Usulan Sistem Informasi Cuti Karyawan	80
5.3 Analisis dan Perancangan Sistem Usulan	81
5.4 Pemodelan Sistem Informasi Cuti Karyawan Usulan	82
5.4.1 <i>Use Case Diagram</i>	82
5.4.2 <i>Activity Diagram</i>	95
5.4.3 <i>Sequence Diagram</i>	107
5.4.4 <i>Class Diagram</i>	119
5.4.5 <i>Deployment Diagram</i>	120
5.5 Pemodelan Data Sistem Informasi Cuti Karyawan Usulan	121
5.5.1 <i>Entity Relationship Diagram</i> (ERD).....	121
5.5.2 Transformasi ERD ke Diagram	122
5.5.3 Kamus Data.....	124
5.6 Perancangan Sistem	126
5.6.1 <i>Windows Navigation Diagram</i> (WND)	126
5.6.2 Perancangan <i>Interface</i> Sistem.....	128
5.7 Implementasi Sistem.....	143

BAB VI PENUTUP	144
6.1 Kesimpulan.....	144
6.2 Saran.....	144
DAFTAR PUSTAKA	145
LAMPIRAN	

DAFTAR GAMBAR

	Halaman
Gambar II.1 Karakteristik Sistem	9
Gambar II.2 Klasifikasi Sistem Terbuka dan Sistem Tertutup	11
Gambar II. 3 Siklus Informasi	14
Gambar II.4 Blok Sistem Informasi.....	17
Gambar II.5 Tahapan SDLC	18
Gambar II.6 Metode Waterfall Development	22
Gambar II.7 Simbol-simbol PDM.....	42
Gambar II. 8 Contoh Windows Navigation Diagram	43
Gambar II. 9 Arsitektur Android	47
Gambar III.1 Kerangka Penelitian	61
Gambar IV.1 Struktur Organisasi PT Indonesia Kendaraan Term	66
Gambar IV.2 Struktur Divisi Sumber Daya Manusia.....	67
Gambar IV.3 Surat Pengajuan Cuti	72
Gambar IV.4 Surat Izin Cuti.....	74
Gambar IV.5 <i>file</i> Pencatatan Riwayat Cuti.....	75
Gambar IV.6 Diagram Alir Sistem Informasi Cuti Karyawan	77
Gambar V.2 Use Case Diagram.....	83
Gambar V.3 Activity Diagram Login	96
Gambar V.4 <i>Activity Diagram</i> Melakukan Pengajuan Cuti.....	97
Gambar V.5 <i>Activity Diagram</i> Melihat Informasi Cuti	98
Gambar V.9 <i>Activity Diagram</i> Mengelola Data Master Hari Libur Nasional	102
Gambar V.10 <i>Activity Diagram</i> Mengelola Data Master Cuti Bersama	103
Gambar V.11 <i>Activity Diagram</i> Mengelola Data Master Data Cuti Karyawan ..	104
Gambar V.12 <i>Activity Diagram</i> Mengelola Data Karyawan	105
Gambar V.13 <i>Activity Diagram</i> Membuat Laporan Cuti.....	106
Gambar V.14 <i>Activity Diagram</i> Melihat Laporan Cuti	107
Gambar V.15 Sequence Diagram Login.....	108

Gambar V.16 <i>Sequence Diagram</i> Melakukan Pengajuan Cuti.....	109
Gambar V.17 <i>Sequence Diagram</i> Melihat Informasi Cuti	110
Gambar V.18 <i>Sequence Diagram</i> Melihat History Cuti	111
Gambar V.22 <i>Sequence Diagram</i> Mengelola Data Master Cuti Bersama.....	115
Gambar V.23 <i>Sequence Diagram</i> Mengelola Data Master Data Cuti Karyawan	116
Gambar V.24 <i>Sequence Diagram</i> Mengelola Data Karyawan	117
Gambar V.25 <i>Sequence Diagram</i> Membuat Laporan Cuti.....	118
Gambar V.26 <i>Sequence Diagram</i> Melihat Laporan Cuti.....	119
Gambar V.27 <i>Class Diagram</i> Sistem Informasi Cuti Karyawan.....	120
Gambar V.28 <i>Deployment Diagram</i> Sistem Informasi Cuti Karyawan	121
Gambar V.29 <i>Entity Relationship Diagram</i> Sistem Informasi Cuti Karyawan...	122
Gambar V.30 Transformasi Diagram ke ERD.....	123
Gambar V.31 WND Sistem Informasi Cuti Karyawan	127
Gambar V.32 Tampilan Halaman <i>Login</i>	128
Gambar V.33 Tampilan Halaman Awal Karyawan	129
Gambar V.34 Tampilan Halaman Awal Supervisor	130
Gambar V.35 Tampilan Halaman Awal Manager	130
Gambar V.36 Tampilan Halaman Awal Direktur	131
Gambar V.37 Tampilan Halaman Awal Admin	132
Gambar V.38 Tampilan Halaman Pengajuan	132
Gambar V.39 Tampilan Halaman Menu Informasi	133
Gambar V.40 Tampilan Halaman Data Karyawan	134
Gambar V.41 Tampilan Halaman Menu Riwayat Cuti	134
Gambar V.42 Tampilan Halaman Persetujuan Cuti Oleh Supervisor	135
Gambar V.43 Tampilan Halaman Persetujuan Cuti Oleh Manager	136
Gambar V.44 Tampilan <i>Form</i> Penolakan Cuti Oleh Manager	136
Gambar V.45 Tampilan Halaman Menu Informasi Pada Admin	137
Gambar V.46 Form Tambah Data Menu Informasi.....	138
Gambar V.47 <i>Form</i> Ubah dan Hapus Data Menu Informasi.....	138
Gambar V.48 <i>Form</i> Ubah dan Hapus Data Pengajuan Cuti	139
Gambar V.49 Halaman Menu Data Karyawan Pada Admin	140

Gambar V.50 Tampilan <i>Form</i> Ubah dan Hapus Data Karyawan	140
Gambar V.51 Tampilan Halaman Menu Data Cuti Pada Admin	141
Gambar V.52 Tampilan Halaman Menu Laporan Cuti Pada Admin.....	142
Gambar V.53 Tampilan Detail Laporan Cuti	142

DAFTAR TABEL

	Halaman
Tabel II.1 Simbol-Simbol <i>Flowmap</i>	26
Tabel II. 2 Ringkasan Diagram UML	27
Tabel II.2 Ringkasan Diagram UML (Lanjutan)	28
Tabel II.2 Ringkasan Diagram UML (Lanjutan)	29
Tabel II. 3 Simbol-simbol <i>Use-Case Diagram</i>	29
Tabel II.3 Simbol-simbol <i>Use-Case Diagram</i> (Lanjutan)	30
Tabel II. 4 Simbol-Simbol <i>Activity Diagram</i>	31
Tabel II.4 Simbol-Simbol <i>Activity Diagram</i> (Lanjutan).....	32
Tabel II. 5 Simbol-Simbol <i>Sequence Diagram</i>	33
Tabel II.5 Simbol-Simbol <i>Sequence Diagram</i> (Lanjutan)	34
Tabel II.6 Simbol Macam-macam <i>Class</i> pada <i>Sequence Diagram</i>	34
Tabel II. 7 Simbol-Simbol <i>Class Diagram</i>	35
Tabel II.7 Simbol-Simbol <i>Class Diagram</i> (Lanjutan)	36
Tabel II. 8 Simbol Deployment Diagram	37
Tabel II.9 Simbol-Simbol Entity Relationship Diagram	39
Tabel II.10 Simbol-Simbol <i>Conceptual Data Model</i>	40
Tabel II.11 Aturan Merubah ERD Secara Umum	40
Tabel II.11 Aturan Merubah ERD Secara Umum (Lanjutan).....	41
Tabel II.12 Simbol Kamus Data	43
Tabel II.13 Versi Android.....	48
Tabel II.14 Tipe-Tipe Data <i>Firebase Database</i>	51
Tabel II.14 Tipe-Tipe Data <i>Firebase Database</i> (Lanjutan)	52
Tabel II.14 Tipe-Tipe Data <i>Firebase Database</i> (Lanjutan)	53
Tabel V.1 Analisis Kebutuhan Sistem	78
Tabel V.1 Analisis Kebutuhan Sistem (Lanjutan)	79
Tabel V.2 Definisi Aktor <i>Use Case Diagram</i> Sistem Usulan	84
Tabel V.3 Definisi <i>Use Case</i>	84
Tabel V.3 Definisi <i>Use Case</i> (Lanjutan).....	85
Tabel V.4 Use Case Description Login	86

Tabel V.5 <i>Use Case Description</i> Melakukan Pengajuan Cuti.....	87
Tabel V.6 <i>Use Case Description</i> Melihat Informasi Cuti.....	88
Tabel V.7 Use Case Description Melihat History Cuti.....	88
Tabel V.8 <i>Use Case Description</i> Mengelola Data Master Hari Libur Nasional ...	89
Tabel V.9 <i>Use Case Description</i> Mengelola Data Master Cuti Bersama.....	90
Tabel V.10 <i>Use Case Description</i> Mengelola Data Master Data Cuti	91
Tabel V.11 <i>Use Case Description</i> Mengelola Data Karyawan	91
Tabel V.11 <i>Use Case Description</i> Mengelola Data Karyawan (Lanjutan)	92
Tabel V.12 <i>Use Case Description</i> Membuat Laporan Cuti	92
Tabel V.12 <i>Use Case Description</i> Membuat Laporan Cuti (Lanjutan)	93
Tabel V.13 <i>Use Case Description</i> Memberikan Validasi Cuti Karyawan (1).....	93
Tabel V.14 <i>Use Case Description</i> Memberikan Validasi Cuti Karyawan (2).....	94
Tabel V.15 <i>Use Case Description</i> Melihat Laporan Cuti.....	95
Tabel V.16 Spesifikasi Objek <i>User</i>	124
Tabel V.16 Spesifikasi Objek <i>User</i> (Lanjutan).....	125
Tabel V.17 Spesifikasi Objek Pengajuan Cuti.....	125
Tabel V.18 Spesifikasi Objek Kalender_Libur.....	126

DAFTAR LAMPIRAN

	Halaman
LAMPIRAN A	L-1
LAMPIRAN B	L-3
LAMPIRAN C	L-25

BAB I

PENDAHULUAN

1.1 Latar Belakang

Abad dua satu merupakan abad persaingan dalam segala bidang dan aspek kehidupan. Salah satu persaingan dalam aspek kehidupan yang sering dihadapi negara berkembang adalah persaingan dalam alat transportasi. Dalam meningkatkan kebutuhan perusahaan tentunya sebuah perusahaan pun dituntut lebih kreatif dan inovatif dalam bersaing dengan perusahaan lainnya seperti meningkatkan skill sumber daya manusia, meningkatkan mutu pelayanan dan menghasilkan produk yang berkualitas. Hal lain yang dapat mendorong persaingan perusahaan yaitu memanfaatkan teknologi informasi dimana sebuah perusahaan dapat menjadi lebih produktif mengolah informasi lebih cepat dan akurat. Teknologi informasi dan sistem informasi pada saat ini telah memasuki era “Industri 4.0” atau “Revolusi Industri 4.0” dimana menjadikan semua perusahaan melakukan pembaharuan terhadap proses bisnis dan sistem kerja yang dapat meningkatkan keunggulan kompetitif perusahaan.

Penggunaan teknologi informasi pada perusahaan sangat berpengaruh dan dapat memberikan perubahan cukup besar kepada suatu departemen atau divisi. Pengembangan teknologi informasi dan sistem informasi menjadikan setiap perusahaan lebih efektif dan efisien pada saat pegambilan keputusan di setiap divisi atau pada saat pengelolaan data serta dapat mengurangi resiko kerja seperti *human error*. Selain itu, sistem informasi juga dapat diterapkan atau digunakan oleh divisi atau bagian yang terdapat dalam suatu perusahaan guna untuk meningkatkan produktivitas serta dapat mengurangi biaya operasional perusahaan.

PT Indonesia kendaraan Terminal merupakan salah satu anak perusahaan PT PELINDO (Pelabuhan Indonesia) yang bergerak di bidang ekspor-impor kendaraan, agar proses ekspor-impor berjalan dengan baik diperlukan peranan

sumber daya manusia (SDM) yang kompeten, untuk menjaga kompeten dan loyalitas SDM sebuah perusahaan perlu menjaga dengan suatu fungsi pemeliharaan agar terciptanya hubungan jangka panjang antara pihak perusahaan dan SDM. Fungsi pemeliharaan dapat dilakukan dengan memberikan libur atau cuti pada karyawan, karena cuti dapat memberikan manfaat seperti menghilangkan jemuhan, menunjang kreativitas karyawan dan lain-lain. Selain itu karyawan juga dapat memanfaatkan cuti untuk kepentingan lain di luar perusahaan.

Proses manajemen cuti di divisi sumber daya manusia meliputi pengajuan cuti karyawan yang masih menggunakan form kertas, permintaan persetujuan berupa tanda tangan yang diisi oleh Supervisor dan manager yang kadang terhambat jika dinas luar atau sibuk, pemberian surat cuti karyawan kepada divisi sumber daya manusia setelah itu divisi sumber daya manusia akan membuatkan surat izin cuti karyawan yang akan diserahkan kepada Direktur Keuangan dan SDM, perekapan riwayat data cuti karyawan dilakukan dengan menyimpan di dalam map atau diletakkan di atas meja dan mencatatnya di *microsoft excel* dengan format yang kurang lengkap .

Dengan memperhatikan proses di atas, direncanakan pembuatan suatu rancangan bangun aplikasi informasi cuti yang akan diterapkan di Divisi Sumber Daya Manusia. Adapun judul dari penulisan Tugas Akhir ini adalah “RANCANG BANGUN SISTEM INFORMASI CUTI KARYAWAN DI DIVISI SUMBER DAYA MANUSIA PADA PT INDONESIA KENDARAAN TERMINAL BERBASIS ANDROID MENGGUNAKAN ANDROID STUDIO 3.3.2 DAN FIREBASE”. Rancangan bangun sistem informasi tersebut diharapkan akan membantu perusahaan khususnya Divisi Sumber Daya Manusia dalam mengelola cuti karyawan di perusahaan.

1.2 Pokok Permasalahan

Berdasarkan latar belakang di atas, maka dapat diidentifikasi kendala yang terdapat di divisi Sumber Daya Manusia di PT Indonesia Kendaraan Terminal yaitu sebagai berikut:

1. Proses meminta persetujuan kepada Supervisor dan Manager divisi masing-masing memiliki kendala ketika dinas luar atau sibuk yang mengakibatkan proses cuti memakan waktu lama atau terhambat.
2. Tidak terdapatnya tempat atau ruang khusus pengarsipan data cuti yang permanen sehingga data cuti pada tahun-tahun sebelumnya tidak terdata karna ada yang hilang atau rusak.
3. Proses perekapan data cuti karyawan yang tidak menggunakan format lengkap, seperti tidak terdapatnya keterangan jenis cuti yang sudah diambil sehingga karyawan dapat memanipulasi jenis cuti yang akan diajukan berikutnya.

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian Tugas Akhir ini adalah menganalisis, merancang dan membangun sistem informasi karyawan yang mampu:

1. Melakukan proses pengelolaan cuti dengan sistem yang terintegrasi antara supervisor, manager, Divisi Sumber Daya Manusia serta Direktur Keuangan dan SDM agar dapat berlangsung lebih cepat serta menghemat waktu dan tenaga.
2. Membuat *database* sistem informasi cuti karyawan agar terhindar terjadinya data rusak atau hilang.
3. Membuat rekap data riwayat cuti karyawan lebih terformat dan dapat diakses oleh divisi sumber daya manusia dan Direktur Keuangan & SDM serta menyediakan informasi saldo cuti bagi karyawan.

1.4 Batasan Masalah

Agar dalam penulisan Tugas Akhir ini lebih fokus dan lebih terarah, maka perlu diadakan batasan-batasan sebagai berikut:

1. Tempat dilakukannya praktik kerja lapangan adalah PT Indonesia Kendaraan Terminal yang berlokasi di Jalan Sindang Laut No. 101, Cilincing, Jakarta Utara 14110

2. Pengamatan yang dilakukan pada divisi sumber daya manusia selama 2 bulan mulai dari 02 Juli sampai dengan 31 Agustus 2018
3. Penelitian ini hanya sebatas pada pengajuan cuti yang diperuntukkan karyawan yaitu staf divisi atau tenaga non organik.

1.5 Manfaat Tugas Akhir

Adapun manfaat yang dapat diambil dari penelitian pada Tugas Akhir ini adalah sebagai berikut:

1. Bagi perusahaan

Hasil penelitian ini agar dapat diimplementasikan di perusahaan guna mempermudah proses pengajuan cuti karyawan di PT Indonesia Kendaraan Terminal.

2. Bagi mahasiswa:

- a. Memberikan kemampuan dalam mengaplikasikan teori secara jelas terhadap masalah yang diamati.
- b. Memberikan pembelajaran untuk menerapkan ilmu yang diperoleh selama perkuliahan serta menambah wawasan dan pengetahuan, khususnya dalam analisis dan perancangan sistem informasi.

3. Bagi pihak lain

Hasil penelitian ini diharapkan dapat memberikan sumbangan pemikiran dan sebagai referensi bagi peneliti lain yang melakukan penelitian serupa.

1.6 Sistematika Penulisan

Tugas Akhir ini disusun secara sistematis guna memberikan gambaran yang jelas mengenai isi dan pembahasan yang ada di dalamnya. Adapun sistematika penulisan Tugas Akhir ini diuraikan dalam enam bab, yaitu:

BAB I PENDAHULUAN

Bab ini memuat latar belakang penelitian, pokok permasalahan, tujuan penelitian, batasan masalah, manfaat Tugas Akhir, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini menjelaskan tentang teori-teori yang didapat dari materi kuliah maupun buku-buku, dan literatur lainnya yang digunakan sebagai referensi mengenai tema yang diambil.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan metode ilmiah dalam mencari dan menjelaskan kerangka pemecahan masalah yang menguraikan tahap-tahap untuk mengatasi permasalahan yang telah diidentifikasi.

BAB IV PENGUMPULAN DAN PENGOLAHAN DATA

Bab ini berisi tentang analisis sistem informasi dan analisis permasalahan yang ada di lapangan yaitu sistem informasi cuti karyawan di PT Indonesia Kendaraan Terminal.

BAB V ANALISIS DAN PEMBAHASAN

Bab ini berisi analisis dari pengolahan data, mulai dari analisis sistem yang meliputi diagram alir sistem, usulan perancangan dengan UML, perancangan basis data, perancangan program yaitu meliputi hirarki menu, dan rancangan antar muka.

BAB VI KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dari hasil penelitian yang dilakukan, serta mengemukakan saran-saran dalam penerapan sistem informasi cuti karyawan untuk perusahaan dalam penelitian dan pengembangannya selanjutnya.

BAB II

LANDASAN TEORI

2.1 Rancang Bangun

Rancang bangun sangat berkaitan dengan perancangan sistem yang merupakan satu kesatuan untuk merancang dan membangun sebuah aplikasi. Menurut Sutabri (2012) perancangan sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru. Jika sistem itu berbasis komputer, rancangan dapat menyertakan spesifikasi jenis peralatan yang akan digunakan. Tujuan dari perancangan sistem yaitu untuk memenuhi kebutuhan para pemakai sistem dan memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada programer. Kedua tujuan ini lebih berfokus pada perancangan atau desain sistem yang terinci yaitu pembuatan rancang bangun yang jelas dan lengkap yang nantinya digunakan untuk pembuatan program komputernya.

2.2 Konsep Dasar Sistem

Kata sistem berasal dari bahasa yunani yaitu “Systema” yang artinya suatu kesatuan komponen atau elemen. Sistem berarti suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisasi, saling berinteraksi satu sama lain dan terpadu. Konsep dasarnya dimana semua elemen-elemen saling terhubung demi mencapai tujuan tertentu (Sutabri, 2012).

Untuk beberapa sistem yang diintisarikan dari beberapa sumber, mendefinisikan bahwa:

1. “Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan”, Raymond McLeod dan George Schell dalam Yulia (2014).
2. “Sistem adalah kumpulan/grup dari subsistem/bagian/komponen apapun, baik fisik ataupun nonfisik yang saling berhubungan satu sama lain dan

bekerja sama secara harmonis untuk mencapai suatu tujuan tertentu”, Azhar Susanto dalam Yulia (2014).

3. “Sistem merupakan seperangkat unsur yang saling terikat dalam suatu antar relasi diantara unsur-unsur tersebut dengan lingkungan”, Ludwig Von Bartalanfy dalam Yulia (2014).
4. “Sistem adalah suatu kumpulan kesatuan dan perangkat hubungan satu sama lain”, Anatol Raporof dalam Yulia (2014).
5. “Menurut L.Ackof, Sistem adalah setiap kesatuan secara konseptual atau fisik yang terdiri dari bagian-bagian dalam keadaan saling tergantung satu sama lainnya”, dalam Yulia (2014).
6. “Menurut Robert G. Murdick, dkk, Sistem adalah seperangkat elemen yang membentuk kegiatan suatu prosedur/bagan pengelahan yang mencari suatu tujuan atau tujuan-tujuan bersama dengan mengoperasikan data atau barang pada waktu rujukan tertentu untuk menghasilkan informasi, energi dan barang” dalam Yulia (2014).
7. Dengan demikian, “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau melakukan sasaran tertentu” ruang lingkup sempit dalam Hutahaen (2015).

Berdasarkan definisi di atas, dapat ditarik kesimpulan bahwa sistem adalah suatu kumpulan atau jaringan kerja yang saling berhubungan dan saling ketergantungan satu sama lain untuk sama-sama menyelesaikan sasaran yang diteliti atau tujuan.

2.2.1 Karakteristik Sistem

Menurut Hutahaean (2015) agar dapat dikatakan dengan sistem yang baik perlu adanya karakteristik sistem, yaitu:

1. Komponen

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan.

Komponen sistem terdiri dari komponen yang berupa subsistem atau bagian dari sistem

2. Batasan Sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batasan suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan Luar Sistem

Lingkungan luar sistem adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tetap dijaga dan yang merugikan harus tetap dijaga dan dikendalikan, kalau tidak akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem (*Interface*)

Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain. Keluaran (*output*) dari subsistem akan menjadi masukkan (*input*) untuk subsistem lain melalui penghubung.

5. Masukkan Sistem (*Input*)

Masukkan adalah energi yang dimasukkan kedalam sistem, yang dapat berupa perawatan (*maintenance input*) dan masukkan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Contoh dalam sistem *computer program* adalah *maintenance input* sedangkan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh komputer menghasilkan panas yang merupakan sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

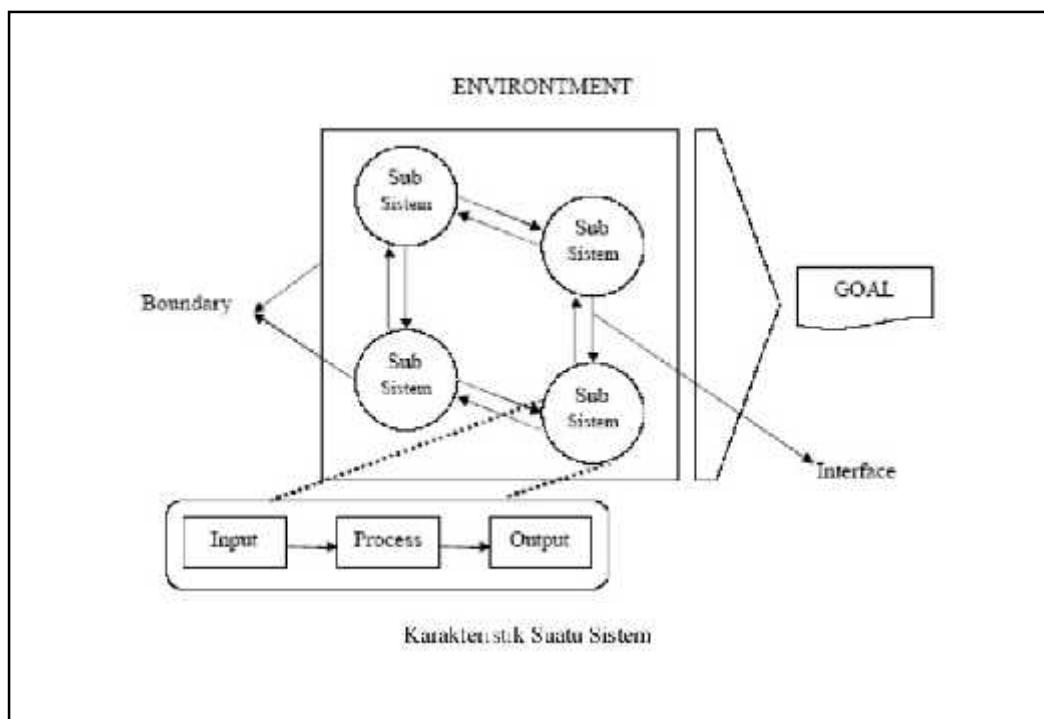
7. Pengolah Sistem

Suatu sistem menjadi bagian pengolah yang akan merubah masukkan menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi, sistem akuntansi akan mengolah data menjadi laporan-laporan keuangan.

8. Sasaran-Sasaran Sistem

Suatu sistem pasti mempunyai tujuan atau sasaran. Sasaran dari sistem sangat menentukan input yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

Berikut ini merupakan karakteristik sistem yang dapat dilihat pada Gambar II.1.



Gambar II.1 Karakteristik Sistem
(Sumber: Hutahean, 2015)

2.2.2 Klasifikasi Sistem

Klasifikasi sistem adalah suatu bentuk kesatuan antara suatu komponen dengan komponen lainnya. Dalam Hutahean (2015), sistem dapat diklasifikasikan dalam beberapa sudut pandang:

1. Klasifikasi sistem sebagai:

a. Sistem Abstrak

Sistem abstrak adalah sistem yang berupa pemikiran-pemikiran atau ide-ide yang tidak tampak secara fisik.

b. Sistem Fisik

Sistem fisik adalah sistem yang ada secara fisik.

2. Sistem diklasifikasikan sebagai:

a. Sistem Alamiyah

Sistem alamiyah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia. Misalnya sistem perputaran bumi.

b. Sistem Buatan Manusia

Sistem buatan manusia adalah sistem yang dibuat oleh manusia yang melibatkan interaksi antara manusia dengan mesin.

3. Sistem diklasifikasikan sebagai:

a. Sistem Tertentu

Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang sudah dapat diprediksi, sebagai keluaran sistem yang dapat diramalkan.

b. Sistem Tak Tentu

Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistik.

4. Sistem diklasifikasikan sebagai:

a. Sistem Tertutup

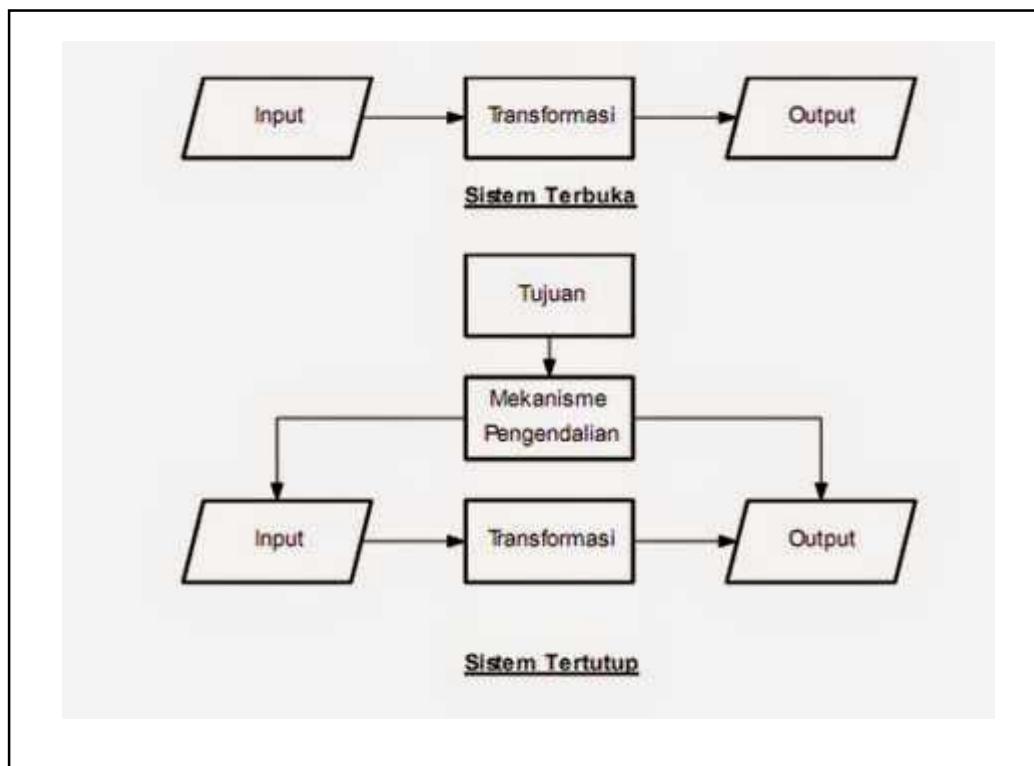
Sistem tertutup adalah sistem yang tidak terpengaruh dengan lingkungan luar, sistem bekerja otomatis tanpa ada turut campur lingkungan luar. Secara teoritis sistem tertutup ini ada, kenyatannya tidak ada sistem yang benar-benar tertutup, yang ada hanya *relatively closed system*.

b. Sistem Terbuka

Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima input dan output dari lingkungan luar atau subsistem lainnya. Karena sistem terbuka

terpengaruh lingkungan luar makan harus mempunyai pengendali yang baik.

Berikut ini merupakan klasifikasi sistem terbuka dan sistem tertutup yang dapat dilihat pada gambar II.2.



Gambar II.2 Klasifikasi Sistem Terbuka dan Sistem Tertutup
(Sumber: Hutahaean, 2015)

2.3 Konsep Dasar Informasi

Informasi atau dalam bahasa inggrisnya adalah *information* yang artinya “konsep, ide, garis besar”. Informasi merupakan salah satu jenis sumber daya yang paling utama dimiliki oleh suatu organisasi, apapun jenis organisasi tersebut. Tanpa informasi, maka tidak akan ada organisasi. Informasi melalui komunikasi menjadi perekat bagi suatu organisasi sehingga organisasi tersebut bisa bersatu (Yulia, 2014). Sedangkan pengertian informasi menurut Sutabri (2012) yaitu data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam pengambilan keputusan.

Beberapa definisi informasi dan data diantaranya sebagai berikut:

1. “Menurut Gordon B. Davis, informasi sebagai data yang telah diolah menjadi bentuk yang berguna bagi penerimanya dan nyata, berupa nilai yang dapat dipahami didalam keputusan sekarang maupun masa depan” dalam Jeperson (2015).
2. “Menurut Robert N. Anthony dan John Dearden, informasi sebagai suatu kenyataan, data, *item* yang menambah pengetahuan bagi penggunanya” dalam Yulia (2014).
3. “Menurut Rusdiana (2014), Informasi merupakan suatu data atau objek yang diproses terlebih dahulu sedimikian rupa sehingga dapat tersusun dan terklasifikasi dengan baik sehingga memiliki arti bagi penerimanya, yang selanjutnya menjadi pengetahuan bagi penerima tentang suatu hal tertentu yang membantu pengambilan keputusan secara tepat”.
4. “Menurut George R. Terry Ph.D, Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan” dalam Hutahean (2015).
5. “Menurut Yulia (2014), informasi merupakan data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat. Jadi, ada suatu proses transformasi data yang menjadi suatu informasi dari *input* ke proses dan menuju ke *output*”.

Dari beberapa definisi diatas dapat disimpulkan bahwa Informasi adalah suatu objek nyata yang dapat berupa fakta dan akan memberikan manfaat bagi yang menerimanya.

Menurut Sutabri (2012) Informasi dikelompokan menjadi 3 bagian, yaitu:

1. Informasi Strategis

Informasi ini digunakan untuk mengambil keputusan jangka panjang, mencakup informasi eksternal, rencana perluasan perusahaan dan sebagainya.

2. Informasi Taktis

Informasi ini dibutuhkan untuk mengambil keputusan jangka menengah, seperti tren penjualan yang dapat dimanfaatkan untuk menyusun rencana penjualan.

3. Informasi Teknis

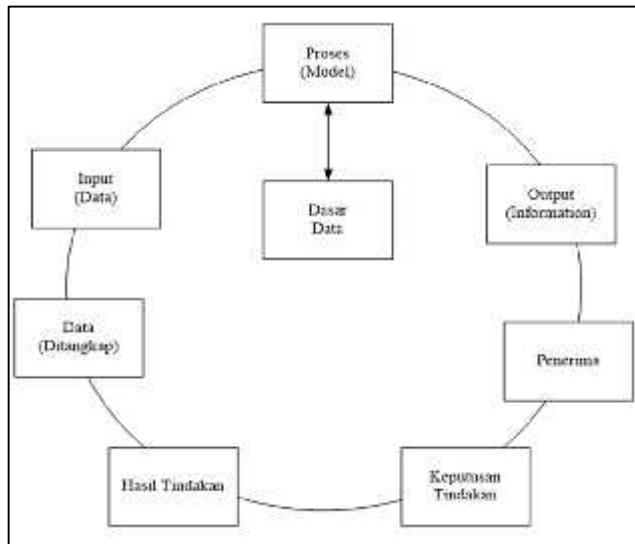
Informasi ini dibutuhkan untuk keperluan operasional sehari-hari, seperti informasi persediaan stok, retur penjualan dan laporan kas harian.

2.3.1 Fungsi dan Siklus Informasi

Fungsi utama informasi adalah menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi. Informasi yang disampaikan kepada pemakai mungkin merupakan hasil dari data yang dimasukkan kedalam pengolahan. Akan tetapi dalam kebanyakan pengambilan keputusan yang kompleks, informasi hanya dapat menambah kemungkinan kepastian atau mengurangi bermacam-macam pilihan. Informasi yang disediakan bagi pengambil keputusan memberikan suatu kemungkinan faktor risiko pada tingkat-tingkat pendapatan yang berbeda.

Pada umumnya banyak bagian informasi yang mungkin berguna dan dengan suatu cara tertentu dapat mempengaruhi tanggapan penerima informasi. Informasi dapat berasal dari pengamatan, percakapan orang lain, rapat-rapat panitia, dari majalah, surat kabar, atau laporan pemerintah dan dari sistem informasi itu sendiri. Sumber informasi merupakan data, data diolah si penerima akan menerima informasi tersebut untuk membuat suatu keputusan dan melakukan tindakan yang akan mengakibatkan munculnya sejumlah data lagi. Data tersebut akan ditangkap sebagai input, diproses kembali menjadi suatu model dan seterusnya sehingga membentuk siklus.

Adapun proses siklus informasi dapat dilihat pada Gambar II.3.



Gambar II. 3 Siklus Informasi

Sumber: Sutabri (2012)

2.3.2 Nilai dan Kualitas Informasi

Menurut Sutabri (2012), nilai dari informasi ditentukan dari dua hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai apabila manfaat yang diperoleh lebih berharga dibandingkan dengan biaya untuk mendapatkannya. Berikut ini adalah nilai informasi berdasarkan atas 10 (sepuluh) sifat, yaitu:

- ## 1. Mudah diperoleh

Sifat ini menunjukkan kemudahan dan kecepatan untuk memperoleh informasi. Kecepatannya dapat diukur, misalnya 1 menit versus 24 jam. Akan tetapi berapa nilainya bagi pemakai informasi sulit untuk mengukurnya.

- ## 2. Luas dan lengkap

Sifat ini menunjukkan kelengkapan isi informasi. Hal ini tidak hanya mengenai volumenya, akan tetapi juga mengenai keluaran informasinya. Sifat ini sangat kabur dan karena itu sulit untuk mengukurnya.

3. Ketelitian

Sifat ini berhubungan dengan tingkat kebebasan dari kesalahan keluaran informasi. Pada volume data yang besar biasanya terdapat dua jenis kesalahan, yakni kesalahan pencatatan dan kesalahan perhitungan.

4. Kecocokan

Sifat ini menunjukkan seberapa baik keluaran informasi dalam hubungannya dengan permintaan para pemakai. Isi informasi harus ada hubungannya dengan masalah yang sedang dihadapi sedangkan semua keluaran yang lainnya tidak berguna, sifat ini sulit mengukurnya.

5. Ketepatan waktu

Sifat ini berhubungan dengan waktu yang dilalui, yang lebih pendek dari siklus untuk mendapatkan informasi. Masukan pengolahan dan pelaporan keluaran kepada para pemakai biasanya tepat waktu. Dalam beberapa hal, ketepatan waktu dapat diukur. Misalnya berapa banyak penjualan dapat ditingkatkan dengan menanggapi permintaan pelanggan mengenai ketersediaan barang-barang inventaris.

6. Kejelasan

Sifat ini menunjukkan tingkat kejelasan informasi. Informasi hendaknya terbebas dari istilah-istilah yang tidak jelas.

7. Keluwesan

Sifat ini berhubungan dengan apakah informasi tersebut dapat digunakan untuk membuat lebih dari satu keputusan, tetapi apakah juga dapat digunakan untuk lebih dari seorang pengambil keputusan. Sifat ini sulit mengukurnya, akan tetapi dalam beberapa hal dapat diukur dengan suatu nilai tertentu.

8. Dapat dibuktikan

Sifat ini menunjukkan sejauh mana informasi itu dapat diuji oleh beberapa pemakai hingga sampai didapatkan kesimpulan yang sama.

9. Tidak ada prasangka

Sifat ini berhubungan dengan ada tidaknya keinginan untuk mengubah informasi tersebut guna mendapatkan kesimpulan yang telah diarahkan sebelumnya.

10. Dapat diukur

Sifat ini menunjukkan hakikat informasi yang dihasilkan oleh sistem informasi formal. Meskipun kabar angin, desas-desus, dugaan-dugaan, klenik dan lainnya juga sering dianggap sebagai informasi, namun hal-hal tersebut berada di luar lingkup pembahasan.

Informasi bernilai sempurna apabila pengambil keputusan dapat mengambil keputusan secara optimal dalam setiap hal, dan bukan keputusan yang rata-rata akan menjadi optimal dan untuk menghindari kejadian-kejadian yang akan mendatangkan kerugian dalam hal demikian perkiraan-perkiraan hasil sebelumnya dipengaruhi oleh informasi tambahan, meskipun informasi tersebut tidak memberi kepastian.

2.4 Sistem Informasi

Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengelola transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi (Sutabri, 2012).

Menurut Hutahaean (2015) Sistem Informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan, yaitu:

1. Blok Masukkan (*Input Block*)

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dasar.

2. Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan metode matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*Output Block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pengguna sistem.

4. Blok Teknologi

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian diri secara keseluruhan. Teknologi terdiri dari unsur utama:

- a. Teknisi (*Human ware or brain ware*)
- b. Perangkat Lunak (*Software*)
- c. Perangkat Keras (*Hardware*)

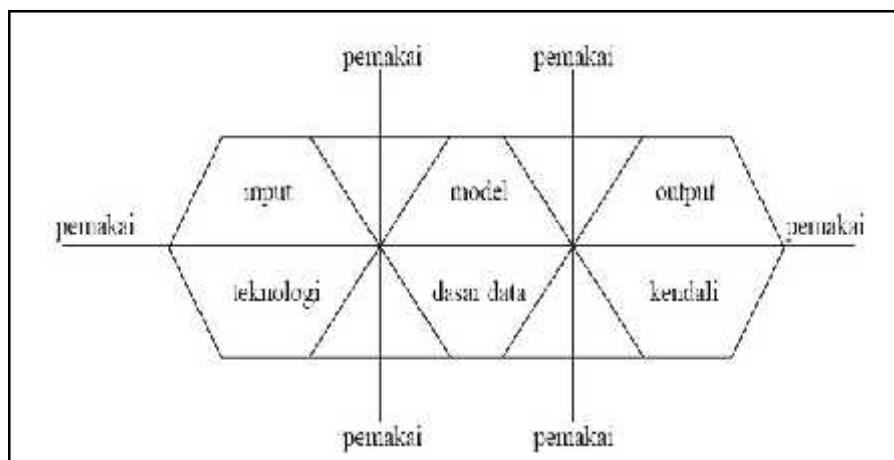
5. Blok Basis Data (*Database Block*)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali (*Control Block*)

Banyak faktor yang dapat merusak sistem informasi misalnya bencana alam, api, temperatur tinggi, air, debu, kecurangan-kecurangan, kejanggalan sistem itu sendiri, kesalahan-kesalahan, ketidakefisienan, sabotase dan sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah atau bila terlanjur terjadi kesalahan dapat langsung diatasi.

Adapun Blok Sistem Informasi dapat dilihat pada Gambar II.4



Gambar II.4 Blok Sistem Informasi
(Sumber: Hutahaean, 2015)

2.5 System Development life Cycle (SDLC)

System Development Life Cycle (SDLC) atau siklus hidup pengembangan sistem adalah proses pemahaman bagaimana sistem informasi dapat mendukung kebutuhan bisnis dengan merancang dan membangun sebuah sistem, dan memberikannya kepada user. *System Development Life Cycle* (SDLC) memiliki 4 tahap dasar yaitu *planning*, *analysis*, *design*, dan *implementation*. Masing-masing tahapan tersusun dari beberapa langkah-langkah yang menghasilkan dokumen yang spesifik dan *file* yang menjelaskan pemahaman tentang proyek.

Dalam beberapa proyek, tahapan dan langkah-langkah SDLC diproses dalam suatu alur tahapan, dimulai dari awal hingga akhir. Dalam SDLC, dijelaskan beberapa teknik, tahapan-tahapan, serta tindakan yang digunakan untuk menyempurnakan langkah-langkah yang dijelaskan secara umum (Dennis *et al.*, 2015).

Ada dua hal penting untuk dipahami tentang SDLC:

1. Mendapatkan dan memahami pengertian umum dari tahapan dan langkah-langkah yang digunakan dalam dalam proyek sistem informasi serta beberapa teknik yang menghasilkan dokumen yang pasti.
2. Kedua, hal penting untuk dipahami bahwa SDLC merupakan proses penyempurnaan yang dilakukan secara bertahap.

Adapun setiap tahapan dalam SDLC dapat dilihat pada gambar II.5.



Gambar II.5 Tahapan SDLC
(Sumber: Denis *et al.*, 2015)

1. *Planning* (Perencanaan)

Tahap *planning* atau perencanaan merupakan proses dasar yang menjelaskan mengapa sebuah sistem informasi harus dibangun dan menentukan bagaimana sebuah tim proyek akan membangunnya. Terdapat dua langkah dalam *planning* (perencanaan) yaitu:

- a. Inisiasi proyek, pada tahap ini nilai bisnis suatu sistem terhadap organisasi diidentifikasi: Apakah sistem tersebut akan menurunkan biaya atau meningkatkan pendapatan? Sebagian besar gagasan untuk sistem baru berasal dari luar area Departemen IT (misalnya dari Departemen Pemasaran, Departemen Akuntansi) dalam suatu bentuk *system request* (permintaan sistem).
- b. Setelah proyek disetujui, proyek memasuki manajemen proyek. Selama manajemen proyek, manajer proyek membuat sebuah rencana kerja, membentuk anggota staf proyek, dan anggota teknisi lainnya untuk membantu tim proyek mengendalikan dan mengarahkan proyek pada keseluruhan tahap SDLC.

2. *Analysis* (Analisis)

Tahap analisis dalam SDLC menjelaskan siapa yang akan memakai sistem, apa yang sistem kerjakan serta kapan dan dimana sistem akan digunakan. Selama pada tahap ini, tim proyek menyelidiki sistem yang ada saat ini, mengidentifikasi peluang untuk melakukan perbaikan, dan mengembangkan sistem baru. Tahap ini memiliki tiga langkah:

- a. Langkah yang pertama adalah Strategi Analisis. Langkah ini dikembangkan untuk mengarahkan usaha dari tim proyek untuk menganalisis sistem yang telah ada (disebut sebagai *as-is system*) beserta masalah-masalah yang ada dan untuk merancang suatu sistem yang baru (disebut sebagai *to-be system*).
- b. Langkah selanjutnya adalah mengumpulkan kebutuhan-kebutuhan sistem (melalui wawancara atau kuesioner).
- c. Kemudian Analisis, konsep sistem, dan model digabungkan menjadi dokumen yang disebut proposal sistem, yang dipresentasikan ke sponsor

proyek dan pengambil keputusan utama lainnya (misalnya, anggota komite persetujuan) yang memutuskan apakah proyek tetap terus dilanjutkan.

3. *Design* (Perancangan)

Tahap *design* atau perancangan memutuskan bagaimana sistem akan beroperasi, mulai dari hal yang berkaitan dengan perangkat keras (*hardware*), perangkat lunak (*software*) dan infrastruktur jaringan seperti: antarmuka pengguna (*interface*), formulir, laporan, program spesifik: *database*, dan *file-file* lainnya yang akan dibutuhkan.

Meskipun sebagian besar keputusan strategis mengenai sistem dibuat dalam pengembangan konsep sistem selama tahap analisis, langkah-langkah dalam tahap desain menentukan secara tepat bagaimana sistem akan beroperasi.

Tahap desain ini memiliki empat langkah:

- a. Strategi perancangan (*Design Strategy*) adalah hal yang pertama kali yang dikembangkan. Strategi perancangan menjelaskan apakah sistem akan dikembangkan oleh programer perusahaan sendiri, atau akan di-*outsourcing* ke perusahaan lain (biasanya perusahaan konsultan), atau apakah perusahaan akan membeli paket perangkat lunak yang sudah ada.
- b. Pengembangan desain arsitektur (*architecture design*) yang mengarah pada dasar untuk sistem, yang menggambarkan perangkat keras, perangkat lunak, dan infrastruktur jaringan yang akan digunakan.
- c. Pengembangan *database* dan spesifikasi *file*. Pengembangan ini menentukan dengan tepat data apa yang akan disimpan dan di mana data tersebut akan disimpan.
- d. Tim analis mengembangkan desain program (*Program Design*), yang mendefinisikan program yang perlu dikembangkan dan apa yang akan dilakukan masing-masing program.

Kumpulan dari hasil kegiatan ini seperti desain arsitektur, desain antarmuka, spesifikasi *database*, *file*, dan perancangan program merupakan spesifikasi sistem yang diserahkan ke tim pemrograman untuk diimplementasi. Pada

akhir tahap desain, analisis kelayakan dan rencana proyek diperiksa ulang dan diperbaiki.

4. *Implementation* (Implementasi)

Tahap akhir dalam SDLC adalah tahap implementasi, di mana sistem benar-benar dibangun (atau dibeli, dalam kasus paket *design software*). Fase ini sangat penting, karena kebanyakan sistem merupakan bagian terlama dan paling mahal dari proses pengembangan. Fase implementasi ini memiliki tiga langkah:

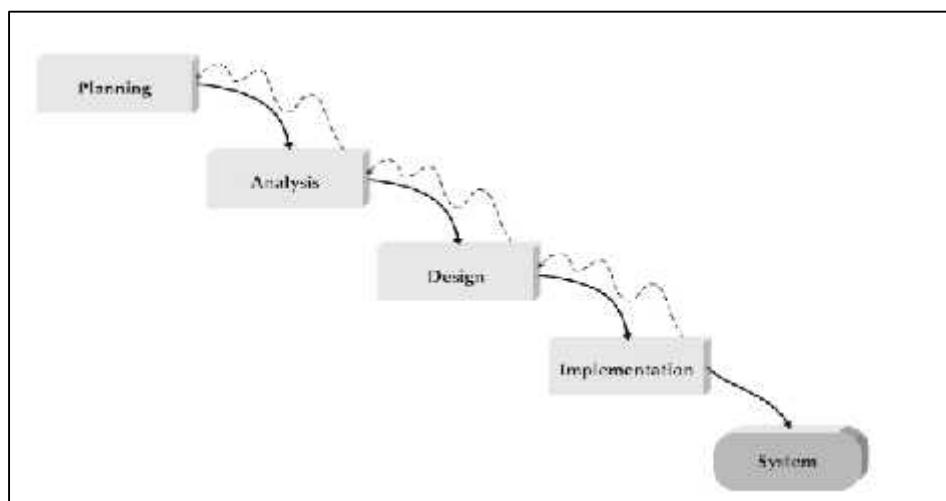
- a. Langkah awal adalah Konstruksi Sistem (*System Construction*). Sistem ini dibangun dan diuji untuk memastikan kinerja dari sistem, bekerja sebagaimana yang telah dirancang. Karena biaya *bugs* bisa menjadi sangat besar, pengujian merupakan salah satu langkah paling kritis dalam tahap Implementasi. Sebagian besar perusahaan memberi lebih banyak waktu dan perhatian untuk menguji daripada mengubah program saat awal pengujian.
- b. Instalasi Sistem, instalasi adalah proses dimana sistem lama dimatikan dan yang baru dinyalakan atau dihidupkan. Salah satu aspek terpenting dari konversi (perubahan dari suatu sistem ke sistem yang baru) adalah pengembangan rencana pelatihan (*training plan*) untuk mengajarkan pengguna cara menggunakan sistem baru dan membantu mengelola perubahan-perubahan yang disebabkan oleh sistem yang baru.
- c. Tim analis menetapkan sebuah Rencana Dukungan (*Support Plan*) untuk sistem. Rencana ini biasanya mencakup tinjauan setelah pelaksanaan serta cara sistematis untuk mengidentifikasi perubahan besar dan kecil yang diperlukan untuk sistem tersebut.

2.6 Metode Pengembangan *Waterfall*

Metode Pengembangan *Waterfall* atau disebut juga dengan metode pengembangan “air terjun” masih digunakan saat ini. Dengan menggunakan metode pengembangan *waterfall* para analisis dan *user* memproses secara urut dari satu fase ke fase berikutnya. Untuk proses setiap fase biasanya sangat panjang dan

disajikan dalam *Project Sponsor* atau sponsor proyek untuk disetujui sehingga dapat berlanjut ke fase berikutnya. Suatu fase atau tahapan akan berlanjut jika satu tahapan disetujui oleh sponsor proyek, model ini disebut dengan *Waterfall Development* karna tahap demi tahap bergerak maju sama seperti air terjun. Tetapi dalam metode ini suatu tahapan sulit mundur ke tahapan sebelumnya.

Tahapan-tahapan metode *waterfall* menurut (Dennis *et al.*, 2015) dapat dilihat pada Gambar II.6.



Gambar II.6 Metode Waterfall Development
(Sumber: Dennis *et.al.*, 2015)

2.7 Karyawan

Menurut Undang-Undang Nomor 14 Tahun 1969 tentang Ketentuan-Ketentuan Pokok Mengenai Tenaga Kerja dalam pasal 1 bahwa tenaga kerja adalah tiap orang yang mampu melakukan pekerjaan baik di dalam maupun di luar hubungan keja guna menghasilkan jasa atau barang untuk memenuhi kebutuhan masyarakat. Sama halnya Berdasarkan Ketentuan Pasal 1 Angka 2 UU No. 13 Tahun 2003, tenaga kerja adalah setiap orang yang mampu melakukan pekerjaan guna menghasilkan barang dan atau jasa, baik untuk memenuhi kebutuhan sendiri maupun untuk masyarakat.

Karyawan merupakan kekayaan dalam suatu perusahaan. Aktivitas perusahaan tidak dapat berjalan apabila tanpa adanya keikutsertaan karyawan. Salah satu yang harus dilakukan karyawan dalam melakukan pekerjaannya yaitu komunikasi. Karyawan perusahaan bertanggung jawab dalam menjelaskan tindakan perusahaan kepada khalayak yang memiliki kepentingan dengan organisasi atau perusahaan tersebut. Karyawan yang memiliki tugas berkaitan dengan publiknya harus memberikan perhatian serta menjadi saluran arus bolak-balik antara organisasi dan khalayak, karena khalayak yang berkepentingan akan selalu tertarik dengan apa saja yang dilakukan perusahaan. Khalayak dapat dibagi menjadi khalayak internal yaitu mereka yang terlibat dalam pekerjaan internal organisasi yaitu karyawan itu sendiri dan keluarganya serta khalayak eksternal, yaitu khalayak yang berada di luar organisasi misalnya masyarakat sekitar, konsumen, pemerhati lingkungan dan investor. Saat ini, setiap pesan yang disampaikan tidak bisa dilakukan secara pukul rata ke semua orang. Setiap organisasi memiliki khalayak khususnya kepada khalayak yang terbatas inilah karyawan perusahaan harus senantiasa menjalin komunikasi baik internal maupun eksternal.

2.8 Cuti

Menurut Peraturan Badan Kepegawaian Negara Republik Indonesia nomor 24 tahun 2017 cuti adalah keadaan tidak masuk kerja yang diizinkan dalam jangka waktu tertentu. Cuti dapat digunakan oleh karyawan untuk tidak masuk kerja dengan alasan tertentu, misalkan untuk *refreshing*, istirahat sakit, melahirkan, menunaikan kewajiban agama, dan keperluan lain sesuai dengan ketentuan cuti pada masing-masing organisasi.

Berikut ini merupakan penjabaran jenis-jenis cuti diantaranya :

1. Cuti Besar
 - a. Cuti Besar adalah cuti yang di berikan kepada karyawan yang telah bekerja sekurang-kurangnya 6 (enam) tahun secara terus-menerus.
 - b. Lamanya cuti besar adalah 3 (tiga) bulan.

- c. Karyawan yang sudah menjalani cuti besar tidak berhak lagi atas cuti tahunannya dalam tahun yang bersangkutan.
- 2. Cuti Melahirkan
 - a. Setiap karyawan wanita yang melahirkan berhak atas cuti bersalinnya, lamanya waktu yang diberikan yaitu 3 (tiga) bulan, yang pengambilannya disesuaikan dengan kondisi kesehatan yang bersangkutan.
- 3. Cuti Tahunan
 - a. Cuti tahunan adalah cuti yang diberikan kepada karyawan yang telah bekerja sekurang-kurangnya 1 (satu) tahun secara terus menerus.
 - b. Lamanya cuti tahunan adalah 12 (dua belas) hari kerja.
 - c. Pada PT Indonesia Kendaraan Terminal, pengambilan cuti tahunan tidak dapat di pecah-pecah hingga jangka waktu kurang dari 3 (tiga) hari kerja.
- 4. Cuti Sakit
 - a. Karyawan yang sakit selama 1 (satu) atau 2 (dua) hari berhak atas cutinya.
 - b. Karyawan yang sakit lebih dari 2 (dua) hari berhak atas cuti sakit dengan ketentuan yang bersangkutan harus mengajukan permintaan secara tertulis kepada Supervisor atau Manager divisi masing-masing dengan melampirkan surat keterangan dokter.
 - c. Karyawan Wanita yang mengalami gugur kandung berhak atas cuti sakit paling lama $\frac{1}{2}$ (satu setengah) bulan terhitung dari hari kandungannya gugur dengan melampirkan surat keterangan dokter atau bidan.
- 5. Cuti Naik Haji
 - a. PT Indonesia Kendaraan Terminal memberikan hak cuti naik haji bagi karyawannya yang ingin melaksanakan ibadah haji, lamanya waktu yang diberikan paling lama 45 (empat puluh lima) hari. Bagi karyawan yang ingin mengajukan cuti dapat mengajukan cuti sekurang-kurangnya 3 hari sebelum keberangkatan.
- 6. Cuti Karena Alasan Penting

Karyawan berhalangan hadir/melakukan pekerjaannya dikarenakan suatu alasan penting. Dalam pasal 93 ayat 2 dan 4 UU no.13 tahun 2003 tentang Tenaga Kerja disebutkan bahwa pekerja berhak atas cuti tidak masuk kerja

karena halangan dan tetap dibayar penuh. Alasan/keperluan penting tersebut mencakup:

- a. Karyawan menikah : 3 (tiga) hari.
- b. Menikahkan anaknya : 2 (dua) hari.
- c. Mengkhitarkan anaknya : 2 (dua) hari.
- d. Membaptiskan anaknya : 2 (dua) hari.
- e. Suami/istri orang tua/mertua, anak atau menantu meninggal dunia : 2 (dua) hari.
- f. Anggota keluarga dalam satu rumah meninggal dunia : 1 (satu) hari

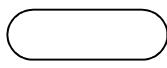
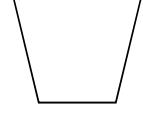
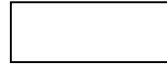
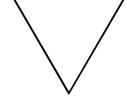
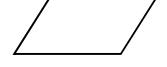
2.9 *Flowchart Mapping (Flowmap)*

Menurut Jogiyanto (2010) *Flowmap* adalah campuran peta dan *flowchart* yang menunjukkan pergerakan benda dari suatu lokasi ke lokasi lain, seperti jumlah orang dalam migrasi, jumlah barang yang diperdagangkan, atau jumlah paket dalam jaringan. *Flowmap* menolong analisis dan programer untuk memecahkan masalah ke dalam segmen yang lebih kecil dan menolong dalam menganalisis alternatif lain dalam pengoprasi. Adapun pedoman-pedoman dalam pembuatan *flowmap* adalah sebagai berikut:

1. *Flowmap* sebaiknya digambarkan dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam *flowmap* harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhir.
4. Masing-masing kegiatan di dalam *flowmap* sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan.
5. Masing-masing kegiatan di dalam *flowmap* harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung ditempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol simbol *flowmap* yang standar.

Adapun simbol-simbol yang sering digunakan dalam *flowmap* dapat dilihat pada tabel II.1 berikut ini:

Tabel II.1 Simbol-Simbol *Flowmap*

Simbol	Nama	Fungsi
	Mulai/Ahir	Menunjukkan dimulai/akhir dari sebuah proses.
	Dokumen	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak dikertas.
	Dokumen Rangkap	Menunjukkan multi dokumen.
	Manual <i>Process</i>	Simbol untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer/pekerjaan manual.
	Proses	Proses pengolahan data.
	Arsip	Menunjukkan pengarsipan data.
	Garis Alir	Arus dari suatu proses.
	Input/Output	Simbol input/output digunakan untuk mewakili data input/output

(Sumber: Jogyianto, 2010)

2.10 *Unified Modelling Language (UML)*

Unified Modelling Language (UML) adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Tujuan dari UML adalah untuk menyediakan kosa kata yang umum dari istilah-istilah berbasis objek dan teknik yang cukup banyak untuk memodelkan proyek pengembangan sistem dari analisis ke desain. Diagram-diagram yang ada dalam UML terbagi menjadi dua bagian utama yaitu *Structure Diagram* dan *Behavior Diagram*. *Structure Diagram* biasanya digunakan untuk mempresentasikan data dan hubungan statik yang ada di dalam sebuah sistem informasi. Sedangkan *Behavior Diagram* menyediakan para analis dengan sebuah gambaran hubungan yang dinamis antara instansi atau objek yang merepresentasikan sistem informasi bisnis (Dennis *et al.*, 2015). UML ini mendefinisikan suatu *set* dari 15 teknik untuk pemodelan sistem, (lihat Tabel II.2).

Tabel II. 2 Ringkasan Diagram UML

Nama Diagram	Fungsi
<i>Structure Diagrams</i>	
<i>Class</i>	Menggambarkan hubungan antara pemodelan-pemodelan <i>class</i> di dalam sistem.
<i>Object</i>	Menggambarkan hubungan antara pemodelan-pemodelan objek di dalam sistem.
<i>Package</i>	Mengelompokkan elemen UML lainnya menjadi satu untuk membentuk level konstruksi yang lebih tinggi

Tabel II.2 Ringkasan Diagram UML (Lanjutan)

Nama Diagram	Fungsi
<i>Deployment</i>	Menampilkan arsitektur fisik dari suatu sistem. Bisa juga digunakan untuk menunjukkan komponen <i>software</i> yang sedang dijadikan arsitektur fisik suatu sistem.
<i>Component</i>	Menggambarkan hubungan fisik diantara komponen-komponen <i>software</i> .
<i>Composite</i>	Menggambarkan struktur internal dari suatu <i>class</i> dan hubungan diantara bagian-bagian dari suatu <i>class</i> .
<i>Profile</i>	Digunakan untuk mengembangkan eksistensi UML sendiri.
<i>Behavioral Diagrams</i>	
<i>Activity</i>	Menggambarkan proses bisnis masing-masing <i>class</i> . Aliran aktivitas dalam <i>use case</i> , atau detail desain dari suatu metode.
<i>Sequence</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam aktivitas berdasarkan urutan waktu.
<i>Communication</i>	Memodelkan kebiasaan dari objek-objek dalam suatu <i>use case</i> . Fokus dalam komunikasi antara satu <i>set</i> dari kolaborasi objek dari suatu aktivitas.
<i>Behavioral State Machine</i>	Menguji kebiasaan dari suatu <i>class</i> .
<i>Interaction Overview</i>	Menggambarkan suatu ikhtisar alur dari kontrol suatu proses.
<i>Timing</i>	Menggambarkan interaksi yang terjadi diantara kumpulan dari objek-objek.

Tabel II.2 Ringkasan Diagram UML (Lanjutan)

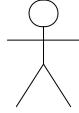
Nama Diagram	Fungsi
<i>Protocol State Machine</i>	Menggambarkan ketergantungan di antara perbedaan <i>interface</i> dari suatu <i>class</i> .
<i>Use Case</i>	Menggambarkan kebutuhan bisnis untuk sistem dan untuk menggambarkan interaksi antara sistem dan lingkungannya.

(Sumber: Dennis *et al.*, 2015)

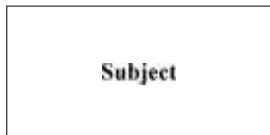
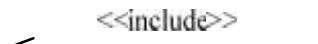
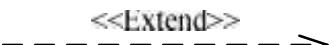
2.10.1 Use Case Diagram

Use case diagram dibuat untuk menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Suatu landasan pola pikir yang ditekankan dalam diagram use case ini adalah “apa” yang dapat diperbuat oleh sistem dan bukan “bagaimana” sistem melakukannya. Diagram use case menggambarkan kebutuhan sistem dari sudut pandang user, yaitu proses yang dilakukan oleh sistem dalam melayani user yang berinteraksi dengan sistem tersebut (Denis., et.al, 2015). Berikut ini merupakan simbol-simbol use case yang dijelaskan pada Tabel II.3.

Tabel II. 3 Simbol-simbol *Use-Case Diagram*

Simbol	Fungsi
 Actor/Role 	<ol style="list-style-type: none"> 1. Menyatakan seseorang atau sistem yang mendapatkan keuntungan dari sistem. 2. Digambarkan sebagai gambar <i>stick man</i> /gambar orang (<i>default</i>) atau jika bukan seorang aktor manusia, digambarkan dengan suatu kotak dengan tanda <i><<actor>></i> di dalamnya (alternatif). 3. Dinamakan dengan frasa kata kerja

Tabel II.3 Simbol-simbol *Use-Case Diagram* (Lanjutan)

Simbol	Fungsi
	<ol style="list-style-type: none"> 1. Menyertakan nama subjek di dalam maupun di atas. 2. Merepresentasikan ruang lingkup dari subjek, sistem atau proses bisnis.
	Menghubungkan suatu aktor dengan <i>use case</i> dengan interaksi antara keduanya.
	<ol style="list-style-type: none"> 1. Merepresentasikan fungsionalitas suatu <i>use case</i> dengan <i>use case</i> lainnya. 2. Disimbolkan dengan anak panah dari sebuah <i>use case</i> dasar ke <i>use case</i> yang digunakan.
	<ol style="list-style-type: none"> 1. Merepresentasikan perluasan (<i>extend</i>) dari <i>use case</i> lain untuk menyertakan perilaku opsional (tidak wajib). 2. Disimbolkan dengan anak panah yang digambarkan dari perluasan <i>use case</i> ke <i>use case</i> dasar.
	<ol style="list-style-type: none"> 1. Merepresentasikan <i>use case</i> khusus ke satu <i>use case</i> yang lebih umum. 2. Disimbolkan dengan anak panah yang digambarkan dari <i>use case</i> khusus ke <i>use case</i> umum.

(Sumber: Dennis *et al.*, 2015)

2.10.2 *Activity Diagram*

Activity diagram digunakan untuk memodelkan perilaku dalam proses bisnis yang independen terhadap objek. Dalam banyak hal, *activity diagram* dapat dipandang sebagai data *flow* diagram canggih yang digunakan pada konjungsi dengan analisis terstruktur. Walaupun begitu, *activity diagram* tidak seperti data

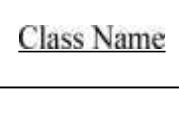
flow diagram, *activity diagram* menggunakan notasi yang mengalamatkan pemodelan paralel. (Dennis *et al.*, 2015).

Yang perlu diperhatikan disini adalah bahwa *activity diagram* menggambarkan aktivitas sistem, bukan apa yang dilakukan aktor, tetapi aktivitas yang dapat dilakukan oleh sistem. Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

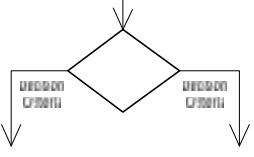
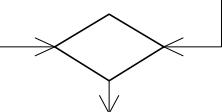
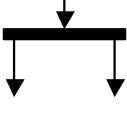
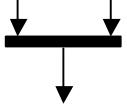
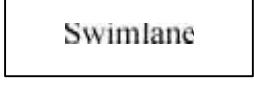
1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem atau *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang akan ditampilkan pada perangkat lunak.

Berikut adalah simbol-simbol yang ada pada *Activity Diagram* (lihat Tabel II.4).

Tabel II. 4 Simbol-Simbol *Activity Diagram*

No	Simbol	Nama	Deskripsi
1.		<i>Action</i>	Merupakan notasi yang simple dan tidak untuk dianalisis perilakunya. Dinamakan sesuai aksi yg dilakukan.
2.		<i>Activity</i>	Digunakan untuk mewakili serangkaian aktivitas yang terjadi.
3.		<i>Object Node</i>	Digunakan untuk mewakili objek yang terhubung ke satu set arus objek.
4.		<i>Control Flow</i>	Menunjukkan urutan eksekusi.

Tabel II.4 Simbol-Simbol *Activity Diagram* (Lanjutan)

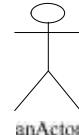
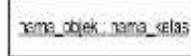
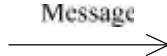
No.	Simbol	Nama	Deskripsi
5.		<i>Object Flow</i>	Menunjukkan aliran dari sebuah objek dari satu aktivitas ke aktivitas lainnya.
6.		<i>Initial Node</i>	Menggambarkan permulaan dari suatu serangkaian aktivitas.
7.		<i>Final-activity Node</i>	Menggambarkan akhir dari setiap aktivitas.
8.		<i>Final-flow Node</i>	Menggambarkan akhir aliran kontrol spesifik atau aliran objek.
9.		<i>Decision Node</i>	Menggambarkan suatu kondisi untuk memastikan bahwa arah panah atau arah objek hanya memiliki satu jalur ke bawah.
10.		<i>Merge Node</i>	Digunakan untuk membawa aliran keputusan yang berbeda ke satu <i>decision node</i> .
11.		<i>Fork Node</i>	Menggambarkan percabangan dari satu aliran aktivitas.
12.		<i>Join Node</i>	Menyatukan beberapa percabangan dari aliran aktivitas.
13.		<i>Swimlane</i>	Memisahkan objek yang bertanggung jawab melaksanakan serangkaian aktivitas

(Sumber: Dennis *et al.*, 2015)

2.10.3 Sequence Diagram

Sequence diagram adalah salah satu dari dua jenis diagram interaksi. *Sequence diagram* menggambarkan objek-objek yang berpartisipasi dengan mengirimkan sejumlah pesan yang melewati diantara mereka untuk satu *use case*. *Sequence diagram* adalah model dinamis yang menunjukkan urutan eksplisit pesan yang lewat diantara objek dalam interaksi yang telah didefinisikan. Karena urutan diagram menekankan pemesanan berbasis waktu kegiatan yang terjadi di antara objek, *sequence diagram* sangat membantu untuk memahami spesifikasi *real-time* dan kompleks suatu *use case* (Dennis *et al.*, 2015). Berikut ini merupakan simbol-simbol yang ada pada *Sequence Diagram*, (lihat Tabel II.5).

Tabel II. 5 Simbol-Simbol *Sequence Diagram*

No	Simbol	Nama	Deskripsi
1.	 	Aktor	Menggambarkan manusia atau sistem yang berasal dari eksternal ke sistem yang berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan.
2.		Lifeline	Menyatakan <i>lifeline</i> suatu objek.
3.		Objek	Berpartisipasi secara berurutan dengan mengirim dan atau menerima pesan yang ditempatkan diatas diagram.
4.		Execution Occurrence	Menyatakan objek dalam keadaan aktif dan berinteraksi dengan mengirim pesan.
5.		Message	Pesan yang menggambarkan komunikasi yang terjadi antar objek.

Tabel II.5 Simbol-Simbol *Sequence Diagram* (Lanjutan)

No	Simbol	Nama	Deskripsi
6.		<i>Message (return)</i>	Pesan yang dikirim untuk diri sendiri.
7.		<i>Guard Condition</i>	Menggambarkan tes yang harus dipenuhi untuk pesan yang akan dikirim.
8.	x	<i>Object Destruction</i>	Ditempatkan di akhir <i>lifeline</i> suatu objek untuk menunjukkan akhir dari eksistensi.
9.		Frame	Mengindikasikan konteks dari <i>sequence diagram</i> .

(Sumber: Dennis *et al.*, 2015)

Selain simbol-simbol diatas *sequence diagram* memiliki *class* dengan fungsi yang masing-masing berbeda, berikut tabel II.6 merupakan *class* yang terdapat dalam *sequence diagram* menurut Richardson dan Thies:

Tabel II.6 Simbol Macam-macam *Class* pada *Sequence Diagram*

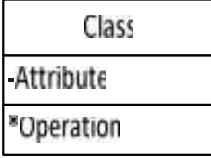
No	Simbol	Deskripsi
1		<i>Boundary Class</i> bertanggung jawab terhadap penanganan interaksi antara aktor dengan sistem.
2		<i>Entity Class</i> merupakan simbol penyimpanan, objek yang dihasilkan sebagian besar berupa data dalam sistem.
3		<i>Control Class</i> merupakan koordinator dari sistem, setidaknya harus terdapat satu <i>control class</i> dalam setiap <i>use case</i> .

(Sumber: Richardson dan Thies, 2013)

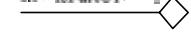
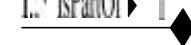
2.10.4 Class Diagram

Class Diagram adalah model statis yang menunjukkan kelas dan keterhubungan antar kelas yang tetap konstan dalam sistem. Sebuah *class* merupakan deskripsi dari sekumpulan objek yang memiliki properti (*attribute*), operasi (*method*), relasi (*association*), dan tingkah laku (*behavior*) yang sama. Sebuah *class* menggambarkan keadaan (*atribut/properti*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda /fungsi). (Dennis *et al.*, 2015). Berikut simbol-simbol yang ada pada diagram kelas (lihat Tabel II.7).

Tabel II. 7 Simbol-Simbol *Class Diagram*

No	Simbol	Nama	Deskripsi
1		<i>Class</i>	<ol style="list-style-type: none"> 1. Menggambarkan orang, tempat atau sesuatu yang dibutuhkan oleh sistem untuk mendokumentasikan dan menyimpan informasi 2. Nama kelas ditulis tebal dan di tengah atas. 3. Memiliki daftar atribut dan operasi di bawah ruang <i>class</i> sebelum nama atribut.
2.	Attribute name /derived attribute name	<i>Attribute</i>	<ol style="list-style-type: none"> 1. Mewakili suatu properti untuk mendeskripsikan keadaan dari suatu objek. 2. Dapat diturunkan dari atribut lain dengan menempatkan garis miring.
3.	Operation Name()	<i>Operation</i>	<ol style="list-style-type: none"> 1. Mewakili suatu aksi atau fungsi dari suatu <i>class</i>. 2. Dapat diklasifikasikan sebagai konstruktur (<i>method</i> yang secara otomatis dipanggil/dijalankan pada sebuah <i>class</i>), <i>query</i> atau perubahan operasi.

Tabel II.7 Simbol-Simbol *Class Diagram* (Lanjutan)

No	Simbol	Nama	Deskripsi
			3. Tanda kurung mengandung informasi atau parameter yang dibutuhkan untuk menampilkan operasi.
4.		Association	<ol style="list-style-type: none"> 1. Menggambarkan suatu hubungan antara beberapa <i>class</i> atau suatu <i>class</i> dengan <i>class</i>. 2. Dilambangkan dengan kata kerja atau <i>rolename</i> yang menggambarkan suatu hubungan diantaranya. 3. Memiliki <i>multiplicity symbols</i> yang menggambarkan jumlah minimal dan maksimal suatu <i>instance class</i> yang dapat dikaitkan dengan instance <i>class</i> yang saling terkait.
5.		Generalization	Menggambarkan macam-macam hubungan antara <i>multiple class</i> .
6.		Aggregation	<ol style="list-style-type: none"> 1. Menggambarkan suatu <i>class</i> terdiri dari <i>class</i> lain atau suatu <i>class</i> adalah bagian dari <i>class</i> lain. 2. Bentuk khusus dari <i>association</i>.
7.		Composition	Mewakili hubungan fisik antara beberapa <i>class</i> atau <i>class</i> dengan <i>class</i> .

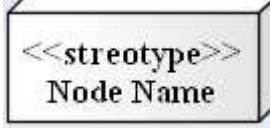
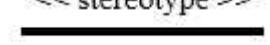
(Sumber: Dennis *et al.*, 2015)

2.10.5 Deployment Diagram

Deployment diagram digunakan untuk mewakili hubungan antara komponen-komponen *hardware* yang digunakan dalam infrastruktur fisik sistem informasi. Misalnya, ketika membuat suatu sistem informasi terdistribusi yang akan

menggunakan jaringan yang luas, *deployment diagram* dapat digunakan untuk menunjukkan hubungan komunikasi antara *node* yang berbeda dalam jaringan. *Deployment diagram* juga dapat digunakan untuk mewakili komponen-komponen *software* dan cara *software* ditempatkan pada arsitektur fisik atau infrastruktur sistem informasi. Dalam hal ini, *deployment diagram* mewakili lingkungan pembuatan *software* (Dennis *et al.*, 2015).

Tabel II. 8 Simbol Deployment Diagram

No	Simbol	Nama	Deskripsi
1		<i>Node</i>	Menggambarkan sumber daya komputasi dalam sebuah sistem (misalnya, komputer klien, <i>server</i> , jaringan yang terpisah, atau individu perangkat jaringan).
2		<i>Artifact</i>	Menggambarkan spesifikasi dari <i>software</i> atau <i>database</i> , misalnya sebuah <i>database</i> , sebuah <i>executable</i> file.
3		<i>Node with a Deployed Artifact</i>	Menggambarkan <i>artifact</i> yang ditempatkan pada <i>node</i> fisik. Mendukung pemodelan distribusi perangkat lunak melalui jaringan.
4		<i>Communication Path</i>	Menggambarkan hubungan antara dua <i>node</i> untuk bertukar pesan.

(Sumber: Dennis *et al.*, 2015)

2.11 DataBase

Database atau Basis Data dapat diartikan sebagai kumpulan data tentang suatu benda atau kejadian yang saling berhubungan satu sama lain (Walijayanto, 2012). Basis data dapat didefinisikan dalam arti implisit yang khusus, yaitu:

1. Basis data merupakan penyajian suatu aspek dari dunia nyata (*realworld atau miniworld*) misalnya basis data perbankan, perpustakaan, pertahanan, perpajakan dan sebagainya.
2. Basis data merupakan kumpulan data dari berbagai sumber yang secara logika mempunyai arti implisit, sehingga data yang terkumpul secara acak dan tanpa mempunyai arti, tidak dapat disebut basis data.
3. basis data perlu dirancang, dibangun, dan data dikumpulkan untuk suatu tujuan. Basis data dapat digunakan oleh beberapa pemakai dan beberapa aplikasi yang sesuai dengan kepentingan pemakai.

Dari pembahasan diatas, dapat dikatakan bahwa basis data mempunyai berbagai sumber data dalam pengumpulan data. Dirancang dan dibangun agar dapat digunakan oleh beberapa pemakai untuk berbagai kepentingan.

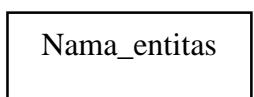
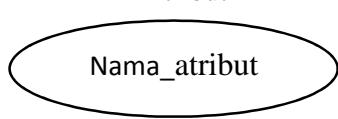
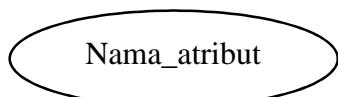
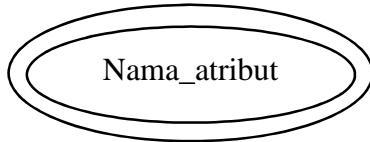
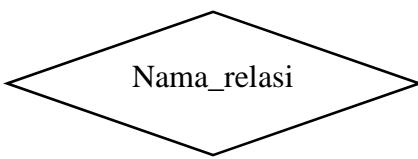
Basis data dapat didefinisikan dalam berbagai sudut pandang yaitu:

1. Himpunan kelompok data yang saling berhubungan yang diorganisasi sedemikian rupa sehingga kelak dapat dimanfaatkan dengan cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa tanpa pengulangan (redundancy) yang tidak perlu, untuk memenuhi kebutuhan.
3. Kumpulan file/tabelarsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

2.12 *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika (Rosa dan Shalahuddin, 2015). ERD digunakan untuk pemodelan basis data relasional, simbol-simbol yang digunakan dalam ERD, yaitu (lihat Tabel II.9).

Tabel II.9 Simbol-Simbol Entity Relationship Diagram

No.	Simbol	Deskripsi
1.	Entitas 	Entitas merupakan data inti yang akan disimpan.
2.	Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan.
4.	Atribut multivilai / multivalue 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja.
6.	Asosiasi 	Penghubung antara relasi dan entitas.

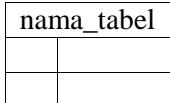
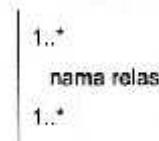
(Sumber: Rosa dan Shalahuddin, 2015)

2.13 Conceptual Data Model (CDM)

Menurut Rosa dan Shalahuddin (2018), CDM (*Conceptual Data Model*) atau model konsep data merupakan konsep yang berkaitan dengan pandangan pemakai terhadap data yang disimpan dalam basis data. CDM sudah dibuat dalam bentuk tabel-tabel tanpa tipe data yang menggambarkan relasi antar tabel untuk keperluan

implementasi ke basis data. CDM merupakan hasil penjabaran dari ERD. dan simbol-simbol CDM dapat dilihat pada Tabel II.10:

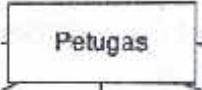
Tabel II.10 Simbol-Simbol *Conceptual Data Model*

No.	Simbol	Deskripsi
1.	Entitas/table 	Entitas atau tabel yang menyimpan data dalam basis data.
2.	Relasi 	Relasi antar tabel yang terdiri atas nama relasi dan <i>multiplicity</i>

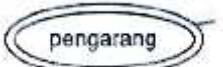
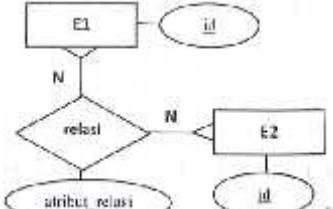
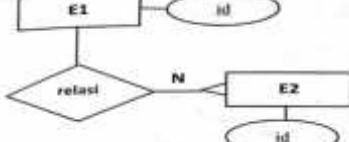
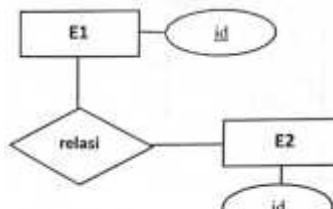
Sumber: Rosa dan Shalahuddin (2018)

Menurut Rosa dan Shalahuddin (2018) menyebutkan bahwa aturan untuk merubah ERD secara umum adalah seperti pada tabel II.11 berikut:

Tabel II.11 Aturan Merubah ERD Secara Umum

ERD	CDM												
Entitas 	<table border="1"> <thead> <tr> <th colspan="2">Petugas</th> </tr> <tr> <th>PK</th> <th>username</th> </tr> </thead> <tbody> <tr> <td></td> <td>password</td> </tr> <tr> <td></td> <td>nama</td> </tr> <tr> <td></td> <td>no_petugas</td> </tr> <tr> <td></td> <td>hak_akses</td> </tr> </tbody> </table> <p>Menjadi sebuah tabel tersendiri.</p>	Petugas		PK	username		password		nama		no_petugas		hak_akses
Petugas													
PK	username												
	password												
	nama												
	no_petugas												
	hak_akses												

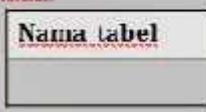
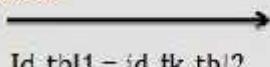
Tabel II.11 Aturan Merubah ERD Secara Umum (Lanjutan)

ERD	CDM								
<p><i>Atribut Multivalue</i></p> 	<table border="1"> <thead> <tr> <th colspan="2">Pengarang</th> </tr> </thead> <tbody> <tr> <td>PK</td><td><u>id_pustaka</u></td> </tr> <tr> <td>PK</td><td><u>pengarang</u></td> </tr> <tr> <td></td><td></td> </tr> </tbody> </table> <p>Menjadi sebuah tabel tersendiri dengan kunci primer (<i>primary key</i>) adalah kunci primer pada entitas dan memiliki atribut dengan nama seperti pada atribut entitas.</p>	Pengarang		PK	<u>id_pustaka</u>	PK	<u>pengarang</u>		
Pengarang									
PK	<u>id_pustaka</u>								
PK	<u>pengarang</u>								
 <p>Relasi dengan kardinalitas <i>many to many</i>.</p>	<table border="1"> <thead> <tr> <th colspan="2">Atribut</th> </tr> </thead> <tbody> <tr> <td>PK</td><td><u>id_E1</u></td> </tr> <tr> <td>PK</td><td><u>id_E2</u></td> </tr> <tr> <td></td><td>atribut_relati</td> </tr> </tbody> </table> <p>Menjadi sebuah tabel tersendiri dengan kunci primer adalah atribut yang menjadi kunci primer di kedua entitas yang direlasikannya.</p>	Atribut		PK	<u>id_E1</u>	PK	<u>id_E2</u>		atribut_relati
Atribut									
PK	<u>id_E1</u>								
PK	<u>id_E2</u>								
	atribut_relati								
 <p>Relasi dengan kardinalitas <i>one to many</i>.</p>	<table border="1"> <thead> <tr> <th colspan="2">E2</th> </tr> </thead> <tbody> <tr> <td>PK</td><td><u>id_E1</u></td> </tr> <tr> <td>PK</td><td><u>id_E2</u></td> </tr> <tr> <td></td><td></td> </tr> </tbody> </table> <p>Kunci primer entitas yang memiliki hubungan <i>one</i> akan dijadikan kunci primer di entitas yang memiliki hubungan <i>many</i> dengan kata lain, relasi tidak menjadi tabel sendiri.</p>	E2		PK	<u>id_E1</u>	PK	<u>id_E2</u>		
E2									
PK	<u>id_E1</u>								
PK	<u>id_E2</u>								
 <p>Relasi dengan kardinalitas <i>one to one</i>.</p>	<table border="1"> <thead> <tr> <th colspan="2">E2</th> </tr> </thead> <tbody> <tr> <td>PK</td><td><u>id_E1</u></td> </tr> <tr> <td>PK</td><td><u>id_E2</u></td> </tr> <tr> <td></td><td></td> </tr> </tbody> </table> <p>kunci primer salah satu entitas akan dijadikan kunci asing (<i>foreign key</i>) pada tabel yang lain dan kunci asing itu dijadikan kunci primer juga, dengan kata lain, relasi tidak menjadi tabel sendiri.</p>	E2		PK	<u>id_E1</u>	PK	<u>id_E2</u>		
E2									
PK	<u>id_E1</u>								
PK	<u>id_E2</u>								

Sumber: Rosa dan Shalahuddin (2018)

2.14 *Physical Data Model (PDM)*

Menurut Rosa dan Shalahudin (2013), Model relasional atau *Physical Data Model* (PDM) adalah model yang menggunakan sejumlah tabel untuk menggambarkan batas data serta hubungan antara data. Setiap tabel mempunyai sejumlah kolom dimana setiap kolom memiliki nama yang unik beserta tipe datanya. PDM merupakan konsep yang menerangkan detail dari bagaimana data disimpan didalam basis data yang sudah siap diimplementasikan kedalam DBMS sehingga nama tabel juga sudah merupakan nama asli tabel yang akan diimplementasikan kedalam DBMS. Berikut adalah simbol-simbol yang ada PDM dapat dilihat pada gambar II.7.

Simbol	Deskripsi
Tabel 	Tabel yang menyimpan data dalam basis data
Relasi  $Id_{tb1} - id_{fk_{tb2}}$	Relasi antar tabel yang terdiri dari persamaan antara primary key (kunci primer) tabel yang diarsir dengan kunci referensi asingan di tabel lain

Gambar II.7 Simbol-simbol PDM

Sumber : Rosa dan Shalahuddin (2013)

2.15 Kamus Data

Menurut Rosa dan Shalahudin (2015), kamus data adalah kumpulan daftar elemen yang mengalir pada sistem perangkat lunak sehingga masukkan (input) dan keluaran (output) dapat dipahami secara umum (memiliki standar cara penulisan). Kamus data dalam implementasi program dapat menjadi parameter masukkan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi:

1. Nama: nama dari data.
2. Deskripsi: merupakan deskripsi dari data

3. Informasi tambahan: seperti tipe data, nilai data, batas nilai data dan komponen yang membentuk data.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut:

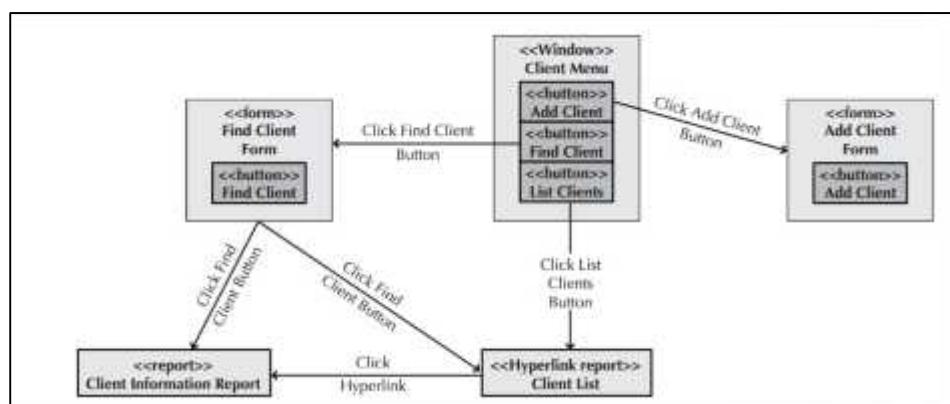
Tabel II.12 Simbol Kamus Data

No	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik....atau....
4.	{ }”	n kali diulang/bernilai banyak
5.	()	Data Opsional
6.	* ... *	Batas Komentar

(Sumber: Rosa dan Shalahudin, 2015)

2.16 Windows Navigation Diagram (WND)

Windows Navigation Diagram merupakan *statechart* diagram khusus yang berfokus pada *user interface*. WND digunakan untuk menunjukkan bagaimana semua layar, formulir dan laporan yang digunakan oleh sistem yang terkait, dan bagaimana penggunaannya berpindah dari satu ke yang lain (Dennis *et al.*, 2015). Contoh dari *Windows Navigation Diagram* dapat dilihat pada Gambar II.8.



Gambar II. 8 Contoh Windows Navigation Diagram
(Sumber: Dennis *et al.*, 2015)

2.17 Java

Menurut Supardi (2015), Java merupakan perangkat lunak produksi Sun Microsystem Inc.. Java merupakan perangkat lunak untuk pemrograman beberapa tujuan (*multi purpose*), *multiplatform* (dapat berjalan di beberapa sistem operasi), mudah dipelajari dan *powerfull*. Aplikasi-aplikasi yang dapat dibuat dengan java meliputi *web programming* (pemrograman web), *desktop programming* (pemrograman desktop) dan *mobile programming* (pemrograman *mobile/handphone*).

Perangkat lunak java sintaknya (tulisannya) mirip dengan C, karena bahasa java dibuat memakai bahasa pemrograman C. Pertama rilis, java disebut JDK (*Java Development Kit*), hingga JDK versi 2 atau dikenal dengan java 2, dibagi menjadi 3 edisi yaitu J2SE (*Java 2 Standart Edition*), J2EE (*Java 2 Enterprise Edition*) dan J2ME (*Java 2 Micro Edition*). Adapun pembahasannya sebagai berikut :

- 1 J2SE merupakan edisi atau teknologi untuk pemrograman desktop atau aplikasi (*Console*) J2SE juga merupakan perangkat lunak dasar yang harus diinstal sebelum anda memakai J2EE dan J2ME.
- 2 J2EE merupakan edisi atau teknologi untuk pemrograman *Enterprise*, seperti pemrograman *Database*, *JSP*, *Beans* dan lain-lainnya.
- 3 J2ME merupakan edisi atau teknologi untuk pemrograman *Mobile/Handphone* dan peralatan kecil (*Small Device*).

Java adalah bahasa pemrograman berorientasi objek 'object oriented programming (OOP)' yang memfokuskan desain pemrograman pada objek dan class. Dasar-dasar OOP meliputi istilah class, object, attribute, method:

1. Class: adalah model dari suatu objek yang menjelaskan karakteristik (sifat) serta fungsi yang dimiliki dari suatu objek, Class merupakan wadah (tempat) yang digunakan untuk menciptakan suatu objek. Dengan kata lain sebuah Class merupakan blueprint dari suatu objek.

2. Object: Setiap objek selalu memiliki suatu keadaan (state) dan behaviour yang dapat mengubah state tersebut. Contoh: manusia memiliki state: umur, tinggi, berat badan, dan sebagainya. Behaviour yang dimiliki oleh manusia dapat berupa: makan, tidur, bekerja, menua, dan sebagainya.
3. Attribute: adalah elemen data dari suatu class. Atribut menyimpan informasi tentang class, atribut dapat diartikan sebagai data, variable, properti atau sebuah field.
4. Method: adalah sebuah fungsi dalam suatu class, setiap method mempunyai tugas masing-masing, ada dua jenis method dalam bahasa Java yaitu *void* dan *non void* method. Void method adalah method yang tidak mengembalikan nilai, sedangkan non void method adalah method yang mengembalikan nilai.

Berikut Contoh sintaks java HelloWorld.Java:

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.print("Hello World!");
    }
}
```

- a. Buka text editor dan ketikan source code di atas.
- b. Simpan dengan nama `HelloWorld` dengan ekstensi `.java` sehingga menjadi `HelloWorld.java`
- c. Simpan di drive `D:` di folder `java` : `D:\java` atau bisa disesuaikan dengan keinginan anda, jika anda masih bingung ikuti seperti yang saya lakukan.
- d. Buka CMD (*Command Prompt*).
- e. Pindah ke path di mana kita menyimpan file `HelloWorld.java` tadi dengan perintah `D: enter cd java enter`, sekarang kita sudah berada di path `D:\java>`.
- f. Compile program dengan perintah `javac HelloWorld.java` dan tekan enter, jika tidak terjadi error kita akan kembali ke path di mana kita berada.
- g. Jalankan program dengan perintah `java HelloWorld`. Jika semua perintah di atas dilakukan dengan benar maka kita akan melihat kata `Hello World!` yang dicetak dengan perintah `System.out.print("Hello World!")`

OUTPUT:

```

D:
D:\>cd java
D:\java>javac HelloWorld.java
D:\java>java HelloWorld
Hello World!

```

2.18 Android

Menurut Supardi (2017), Android merupakan sistem operasi berbentuk linux yang digunakan untuk telepon seluler (*mobile*), seperti telepon pintar (*smartphone*) dan komputer tablet (PDA). Android pada mulanya didirikan oleh Andy Rubin, Rich Miner, Nick Sears dan Chris White pada tahun 2003.

Android merupakan sebuah sistem operasi perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Beberapa pengertian lain dari android yaitu:

1. Merupakan platform terbuka bagi para programmer untuk membuat aplikasi
2. Merupakan sistem operasi yang dibeli Google Inc. dari Android Inc.
3. Bukan bahasa pemrograman, tetapi hanya menyediakan lingkungan hidup atau *run time environment* yang disebut DVM (*Dalvik Virtual Machine*) yang telah dioptimasi untuk alat atau device dengan sistem memori yang kecil.

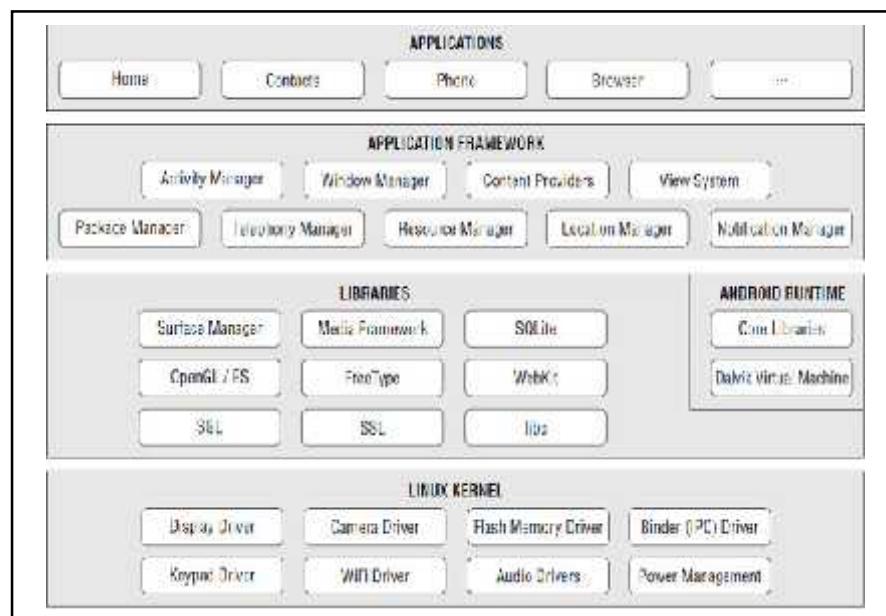
2.18.1 Arsitektur Android

Secara garis besar arsitektur android terdiri atas *application and widgets*, *application framework*, *libraries*, *android run time* dan *linux kernel*.

Adapun pembahasannya sebagai berikut:

1. *Application and Widgets* merupakan layer (lapis) dimana kita berhubungan dengan aplikasi saja
2. *Application FrameWork* merupakan open development platform yang ditawarkan android untuk dapat dikembangkan guna membangun aplikasi
3. *Libraries* merupakan layer dimana fitur-fitur android berada
4. *Android run time* merupakan layer yang membuat aplikasi android dapat dijalankan dimana prosesnya menggunakan implementasi linux.

5. *Linux kernel* merupakan layer inti dari sistem android berada, untuk lebih jelasnya lihat gambar arsitektur android berikut.



Gambar II. 9 Arsitektur Android
(Sumber: Supardi, 2017)

2.18.2 Struktur Aplikasi Android

Struktur aplikasi android ditulis dalam bahasa pemrograman java. Kode java dikompilasi bersama *Resource File* yang dibutuhkan oleh aplikasi. Ada 4 komponen pada aplikasi android yaitu:

- 1 *Activities* merupakan komponen untuk menyajikan tampilan pemakai kepada pengguna.
- 2 *Service* merupakan komponen yang tidak memiliki tampilan pemakai (*user interface*) tetapi service berjalan secara *background*.
- 3 *Broadcast Receiver* merupakan komponen yang berfungsi menerima dan bereaksi untuk menyampaikan notifikasi.
- 4 *Content Provider* komponen yang membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain.

2.18.3 Versi Android

Banyak smartphone dan tablet yang menggunakan sistem operasi dengan versi yang beda. Semakin tinggi versi fiturnya semakin canggih smartphone dan tablet tersebut. Telepon pertama yang memakai sistem operasi android adalah HTC Dream yang dirilis pada tanggal 22 oktober 2008. Beberapa uraian versi android yaitu:

Tabel II.13 Versi Android

No	Nomor Versi	Nama Versi	Tanggal Rilis
1.	(Belum Memakai)	Android Beta	5 November 2007
2.	1.0	Android 1.0	23 September 2008
3.	1.1	Android 1.1	9 Februari 2009
4.	1.5	Cupcake	27 April 2009
5.	1.6	Donut	15 September 2009
6.	2.0 – 2.1	Eclair	2009 - 2010
7.	2.2 – 2.2.3	Froyo	2010 - 2011
8.	2.3 – 2.3.7	Gingerbread	2010 - 2011
9.	3.0 – 3.2.6	Honeycomb	2011 - 2012
10.	4.0.1 – 4.0.4	ICS (Ice Cream Sandwich)	2011 - 2012
11.	4.1	Jelly bean	9 Juli 2012
12.	4.4	Kitkat	31 Oktober 2013
13.	5.0 – 5.1	Lollipop	2014
14.	6.0	MarshMallow	5 Oktober 2015
15.	7.0	Nougat	22 Agustus 2016
16.	8.0	Oreo	Maret 2017
17.	9.0	Pie	Agustus 2018

(Sumber: Supardi, 2017)

2.19 Android Studio

Android Studio merupakan sebuah *Integrated Development Environment* (IDE) khusus untuk membangun aplikasi yang berjalan pada platform android. Android studio ini berbasis pada IntelliJ IDEA, sebuah IDE untuk bahasa pemrograman Java. Bahasa pemrograman utama yang digunakan adalah Java, sedangkan untuk membuat tampilan atau layout, digunakan bahasa XML. Android studio juga terintegrasi dengan Android Software Development Kit (SDK) untuk deploy ke perangkat android. Peluncuran Android Studio ini diumumkan oleh Google pada 16 Mei 2013 pada event Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio mengantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi Android. Android studio memiliki fitur:

1. Projek berbasis pada *Gradle Build*.
2. *Refactory* dan pembenahan *bug* yang cepat.
3. *Tools* baru yang bernama “*Lint*” dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
4. Mendukung *Proguard And App-signing* untuk keamanan.
5. Memiliki GUI aplikasi android lebih mudah.
6. Didukung oleh *Google Cloud Platfrom* untuk setiap aplikasi yang dikembangkan.

2.20 Firebase

Firebase atau *Firebase Realtime Database* merupakan basis data yang dikembangkan oleh *Google*, Fitur utama *Firebase* yaitu mensinkronkan data secara otomatis tanpa membuat permintaan pembaruan atau menyegarkan aplikasi. *Firebase* merupakan salah satu *cloud database* yang memang didesain untuk *cloud application* yang biasa digunakan untuk aplikasi pada android. *Firebase* digunakan untuk mempermudah para pengembang aplikasi dalam mengembangkan aplikasi.

Dengan adanya Firebase, pengembang aplikasi bisa fokus mengembangkan aplikasi tanpa harus memberikan usaha yang besar. Dua fitur yang menarik dari Firebase yaitu Firebase Remote Config dan Firebase Realtime Database. Selain itu terdapat fitur pendukung untuk aplikasi yang membutuhkan pemberitahuan yaitu Firebase Notification. Adapun Pembahasannya sebagai Berikut:

1. *Firebase* menyediakan *realtime database* dan *backend* sebagai layanan. Layanan ini menyediakan pengembang Antarmuka pemrograman aplikasi yang memungkinkan data aplikasi disinkronkan di seluruh klien dan disimpan di *Firebase cloud*. Perusahaan menyediakan pustaka klien yang memungkinkan integrasi dengan Android, IOS, JavaScript, Java, Swift dan Node.js aplikasi. Database juga dapat diakses melalui REST API dan mengikat untuk beberapa JavaScript Frameworks seperti AngularJS, React, Ember.js dan Backbone.js. *Developer* yang menggunakan realtime database dapat mengamankan data dengan menggunakan aturan keamanan yang diberlakukan oleh server perusahaan. *Cloud Firestore* yang merupakan generasi mendatang dari *Firebase Realtime Database* dirilis untuk penggunaan beta. *Firebase Realtime Database* merupakan database yang di-host pada cloud. Data disimpan sebagai JSON (*JavaScript Object Notation*) dan disinkronkan secara realtime ke setiap klien yang terhubung.
2. *Firebase Storage* dirancang untuk pengembang aplikasi yang perlu menyimpan dan menampilkan konten buatan pengguna, seperti foto atau video dan menambahkan keamanan Google pada unggah dan unduh berkas untuk aplikasi *Firebase*, bagaimana pun kualitas jaringannya. Pengembang dapat menggunakannya untuk menyimpan gambar, audio, video, atau konten lain yang dibuat pengguna secara langsung dari *Firebase SDK Klien*. *Firebase Storage* didukung oleh *Google Cloud Storage*.

Adapun tipe-tipe data pada *Firebase Database* :

Tabel II.14 Tipe-Tipe Data *Firebase Database*

No	Tipe Data	Bentuk Data	Keterangan
1.	<i>Array</i>	<i>By element Value</i>	<p>Array tidak dapat berisi nilai array lain sebagai salah satu elemennya. Didalam array, elemen mempertahankan posisi yang ditugaskan padanya. Saat mengurutkan dua atau lebih array, array disusun berdasarkan nilai elemen mereka. Saat membandingkan dua array, elemen pertama dari setiap array dibandingkan. Jika elemen pertama sama, maka elemen kedua dibandingkan dan seterusnya sampai perbedaan ditemukan. Jika sebuah array kehabisan elemen untuk membandingkan tetapi sama hingga titik itu, maka array yang lebih pendek dipesan sebelum array yang lebih panjang.</p> <p>Misalnya, $[1, 2, 3] < [1, 2, 3, 1] < [2]$. Array [2] memiliki nilai elemen pertama terbesar. Array [1, 2, 3] memiliki elemen yang sama dengan tiga elemen pertama [1, 2, 3, 1] tetapi panjangnya lebih pendek</p>
2.	<i>Boolean</i>	<i>False < True</i>	-
3.	<i>Bytes</i>	<i>Byte order</i>	Hingga 1.048.487 byte (1 MiB - 89 byte). Hanya 1.500 byte pertama yang dipertimbangkan oleh kueri.

4.	<i>Date and Time</i>	<i>Chronological</i>	Saat disimpan di Cloud Firestore, hanya presisi untuk mikrodetik; setiap presisi tambahan dibulatkan ke bawah.
----	----------------------	----------------------	--

Tabel II.14 Tipe-Tipe Data *Firebase Database* (Lanjutan)

No	Tipe Data	Bentuk Data	Keterangan
5.	<i>Floating-point number</i>	<i>Numeric</i>	Presisi ganda 64-bit, IEEE 754.
6.	<i>Geographical Point</i>	<i>By latitude, then longitude</i>	-
7	<i>Integer</i>	<i>Numeric</i>	64-bit, <i>signed</i>
8	<i>Map</i>	<i>By keys, then by value</i>	Merupakan objek yang tertanam dalam dokumen. Saat diindeks, Anda dapat meminta pada subbidang. Jika Anda mengecualikan nilai ini dari pengindeksan, maka semua subbidang juga dikecualikan dari pengindeksan. Pemesanan kunci selalu disortir. Misalnya, jika Anda menulis {c: "foo", a: "bar", b: "qux"} peta diurutkan berdasarkan kunci dan disimpan sebagai {a: "foo", b: "bar", c: "qux"}. Bidang peta diurutkan berdasarkan kunci dan dibandingkan dengan pasangan nilai kunci, pertama membandingkan kunci dan kemudian nilai. Jika pasangan nilai kunci pertama sama, pasangan nilai kunci berikutnya akan dibandingkan, dan seterusnya. Jika dua peta dimulai dengan pasangan nilai kunci yang sama, maka panjang peta

			dipertimbangkan. Misalnya, peta berikut berada dalam urutan menaik: <code>{ a: "aaa", b: "baz" }</code>
--	--	--	--

Tabel II.14 Tipe-Tipe Data *Firebase Database* (Lanjutan)

No	Tipe Data	Bentuk Data	Keterangan
			<code>{ a: "foo", b: "bar" }</code> <code>{ a: "foo", b: "bar", c: "qux" }</code> <code>{ a: "foo", b: "baz" }</code> <code>{ b: "aaa", c: "baz" }</code> <code>{ c: "aaa" }</code>
9.	<i>Null</i>	<i>None</i>	-
10.	<i>Reference</i>	<i>By path elements</i> (<i>collection</i> , <i>document id</i> , <i>collection document id</i>)	Misalnya, <code>projects/[PROJECT_ID]/databases/[DATABASE_ID]/documents/[DOCUMENT_PATH]</code> .
11.	<i>Text string</i>	<i>UTF-8 encoded byte order</i>	Hingga 1.048.487 byte (1 MiB - 89 byte). Hanya 1.500 byte pertama dari representasi UTF-8 yang dipertimbangkan oleh kueri.

Sumber: (Developers, 2017)

2.21 *Blackbox Testing*

Black-Box Testing adalah metode pengujian perangkat lunak yang menguji fungsionalitas aplikasi tanpa mengintip ke dalam struktur internal atau cara kerjanya. Metode pengujian ini dapat diterapkan secara virtual untuk setiap tingkat pengujian perangkat lunak (unit, integrasi, sistem, dan penerimaan). Kadang-kadang disebut sebagai pengujian berbasis spesifikasi (Ron Patton, 2005). Contoh kegiatan yang dilakukan pada *black-box testing* yaitu:

1. Membuat *test case* untuk menguji fungsi-fungsi yang ada pada aplikasi
2. Membuat *test case* untuk menguji kesesuaian alur kerja suatu fungsi di aplikasi dengan *requirement* yang dibutuhkan *user* untuk fungsi tersebut.

3. Mencari *bugs/error* dari tampilan (*Interface*) aplikasi.

BAB III

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metodologi penelitian adalah tata cara bagaimana suatu penelitian dilaksanakan atau cara yang ditempuh sehubungan dengan penelitian yang dilakukan, dan memilih langkah-langkah yang sistematis. Metodologi penelitian mempelajari cara-cara melakukan pengamatan dengan pemikiran yang tepat secara terpadu melalui tahapan-tahapan yang disusun secara ilmiah untuk mencari, menyusun, serta menganalisis dan menyimpulkan data-data berdasarkan fakta-fakta secara ilmiah. Untuk menghasilkan penelitian tugas akhir yang lebih lengkap diperlukan suatu metode dalam penelitian yang telah dipersiapkan sesuai dengan masalah yang akan dibahas.

3.2 Jenis dan Sumber Data

Data yang digunakan dalam penelitian ini ada dua jenis, yaitu data primer dan data sekunder. Sumber dari data-data ini berasal dari tempat yang diamati pada praktik kerja lapangan di PT Indonesia Kendaraan Terminal:

1. Data Primer

Data Primer adalah data yang diperoleh secara langsung dari PT Indonesia Kendaraan Terminal, dimana pengumpulan data atau informasi dilakukan langsung dari objek yang diteliti. Data-data tersebut adalah data yang digunakan dalam proses pengajuan, persetujuan dan pengolahan cuti karyawan, analisis dokumen yang berjalan, proses bisnis sistem saat ini dan yang akan diusulkan, serta kebutuhan pengguna sistem.

2. Data Sekunder

Data sekunder adalah data yang diperoleh dari data yang tersedia dan telah terlebih dahulu dikumpulkan dan dilaporkan oleh orang lain, buku-buku dan

kajian ilmiah dari berbagai sumber yang berkaitan dengan penelitian. Data tersebut adalah data umum perusahaan, profil perusahaan, dan struktur organisasi perusahaan.

3.3 Pengumpulan Data

Menurut Rosa dan Shalahudin (2015) Pengumpulan data merupakan hal yang pertama dilakukan dalam analisis sistem. Beberapa teknik pengumpulan data yang umum dan sering digunakan yaitu :

1. Studi Lapangan

Studi lapangan adalah kegiatan melakukan pengumpulan data secara langsung pada objek yang diteliti, kegiatan tersebut dilakukan dengan cara sebagai berikut:

a. Pengamatan

Tahap ini dilakukan secara langsung di divisi Sumber Daya Manusia pada PT Indonesia Kendaraan Terminal dengan mengamati sistem informasi cuti yang berjalan, pengamatan dilakukan mulai dari proses pengajuan cuti yang dilakukan oleh karyawan selaku pemohon, permintaan persetujuan hingga perekapan data cuti pemohon oleh *Officer* Sumber Daya Manusia. Melalui teknik ini, data yang dibutuhkan diamati, dikumpulkan dan diolah sebagai bahan dalam penelitian.

b. Wawancara

Pengambilan data dengan cara berdialog dan bertanya dengan karyawan (Divisi Sumber Daya Manusia) terkait tentang sistem informasi cuti yang berjalan. Sebagai data yang diperlukan untuk penyusunan tugas akhir.

2. Studi Kepustakaan

Mengumpulkan data dan menambah referensi dengan membaca buku-buku, literatur, artikel di internet atau sumber tertulis lain yang berhubungan dengan judul dan permasalahan. Yang berguna untuk melengkapi data yang

diperlukan dalam penulisan laporan tugas akhir ini, agar dalam praktik dan teori tidak jauh berbeda.

3.4 Metode Pengembangan Sistem

Menurut Dennis *et.al* (2015) Metode pengembangan sistem yang digunakan adalah metode pengembangan sistem *waterfall*. Berikut ini akan dijelaskan secara singkat mengenai tahapan-tahapan yang dilakukan dalam pengembangan sistem dengan metode *waterfall*:

1. Tahap Perencanaan (*Planning*)

Pada tahap ini akan dilakukan perencanaan untuk membangun suatu sistem dengan membuat sebuah *system request*.

2. Tahap Analisis (*Analysis*)

Pada tahap ini, merupakan proses analisis kebutuhan sistem. Analis mengumpulkan data-data sebagai bahan pengembangan sistem. Pengumpulan data dapat dilakukan dengan teknik wawancara maupun teknik observasi.

3. Tahap Desain (*Design*)

Proses desain pengembang melakukan desain seperti desain struktur data, dan representasi antarmuka. Data-data yang didapat dari tahap analisis diterapkan dalam tahap desain.

4. Tahap Implementasi (*Implementation*)

Pada tahap ini, pengembang akan membangun sebuah sistem dengan rancangan yang sudah ada diterjemahkan ke dalam sebuah *code* atau aktivitas *coding*. Bahasa pemrograman yang digunakan adalah Java dengan menggunakan *database Firebase*.

3.5 Kerangka Penelitian

Dalam penelitian yang dibahas dalam Tugas Akhir ini, dilakukan langkah-langkah atau tahapan dalam penelitian yang ada sesuai dengan metodologi penelitian yang digunakan dalam penelitian tersebut. Penjelasan langkah-langkah atau tahapan dalam penelitian tersebut adalah sebagai berikut:

1. Studi Pendahuluan

Studi Pendahuluan merupakan langkah awal yang dilakukan untuk memulai penelitian. Hal yang dilakukan adalah dengan membaca buku, literatur, *browsing* internet dan sumber-sumber lainnya yang berkaitan dengan judul Tugas Akhir.

2. Identifikasi Masalah

Pada tahap ini bertujuan untuk mengetahui masalah yang ada pada sistem yang sedang berjalan. Pokok permasalahan yang menjadi objek penelitian ini adalah tidak terdapatnya aplikasi yang dapat mempermudah karyawan dalam mengajukan cuti tanpa harus ke divisi Sumber Daya Manusia serta mempermudah dalam meminta persetujuan kepada atasan. Identifikasi masalah ini dilakukan dengan cara sebagai berikut:

- a. Melakukan observasi di Divisi Sumber Daya Manusia di PT Indonesia Kendaraan Terminal.
- b. Wawancara dengan staf Sumber Daya Manusia yang menangani masalah cuti karyawan.

3. Identifikasi Solusi

1. Maksud dan tujuan dari penelitian ini yakni:

- a. Merancang MBIS (*Mobile Base Information System*) sehingga proses pengajuan cuti maupun pengelolaan cuti dapat berlangsung lebih cepat serta menghemat waktu dan tenaga
- b. Merancang dan membangun *database* sistem informasi cuti karyawan agar terhindar terjadinya data rusak atau hilang.
- c. Membuat *form* dengan format lengkap serta menyediakan informasi saldo cuti karyawan mengenai cuti yang dapat diambil oleh setiap karyawan agar tidak ada karyawan yang mengambil cuti lebih dari ketentuan.

2. Batasan Masalah

Pada tahap ini bertujuan untuk membatasi masalah agar penelitian lebih terarah sehingga dapat menunjukkan gambaran yang lebih spesifik

mengenai arah pemecahannya. Berikut batasan dalam penulisan tugas akhir ini:

- a. Tempat dilakukannya praktik kerja lapangan adalah PT Indonesia Kendaraan Terminal yang berlokasi di Jl. Sindang Laut No.101 Tanjung Priuk, Jakarta Utara.
- b. Pengamatan dilakukan pada Divisi Sumber Daya Manusia selama 2 bulan mulai dari 02 Juli 2018 sampai dengan 31 Agustus 2018.
- c. Penelitian ini hanya sebatas pada pengajuan cuti yang diperuntukkan karyawan yaitu Staf Divisi dan Supervisor.

4. Penerapan Metode *Waterfall*

a. Perencanaan (*Planning*)

Pada tahap ini akan dilakukan perencanaan untuk membangun suatu sistem dengan membuat sebuah sistem informasi pengelolaan cuti.

b. Melakukan Analisis (*Analysis*)

Tahap ini bertujuan untuk melakukan analisis terhadap sistem informasi cuti karyawan yang sedang berjalan dan memberikan usulan pengembangan sistem. Kemudian menganalisis kebutuhan pengguna terhadap program yang akan dibuat. Langkah ini lanjutan dari pengolahan data, semua data yang dapat di analisis dokumen dan analisis proses bisnis yang berjalan. Dan kemudian menggambarkan layanan-layanan yang ada pada sistem serta batasan-batasan yang disediakan oleh sistem.

c. Membuat Desain (*Design*)

Proses desain pengembang melakukan desain seperti desain struktur data, dan representasi antarmuka. Data-data yang didapat dari tahap analisis diterapkan dalam tahap desain. Pada tahap ini akan dirancang aplikasi tentang sistem informasi cuti karyawan yang nantinya akan memuat tentang:

1. Pemodelan sistem menggunakan *Unified Modeling Language* (UML) yaitu menggunakan *usecase diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *deployment diagram*

2. Pemodelan kamus data dengan menggunakan ERD (*Entity Relationship Diagram*) dan *Conceptual Data Model* (CDM)
3. Merancang sistem dengan menggunakan WND (*Windows Navigation Diagram*) dan rancangan antar muka.
- d. Implementasi (*Implementation*)

Pada tahap ini, pengembang akan membangun sebuah sistem dengan rancangan yang sudah ada diterjemahkan ke dalam sebuah *code* atau aktivitas *coding*. Pada tahap ini pengembang akan mengimplementasikan rancangan program.

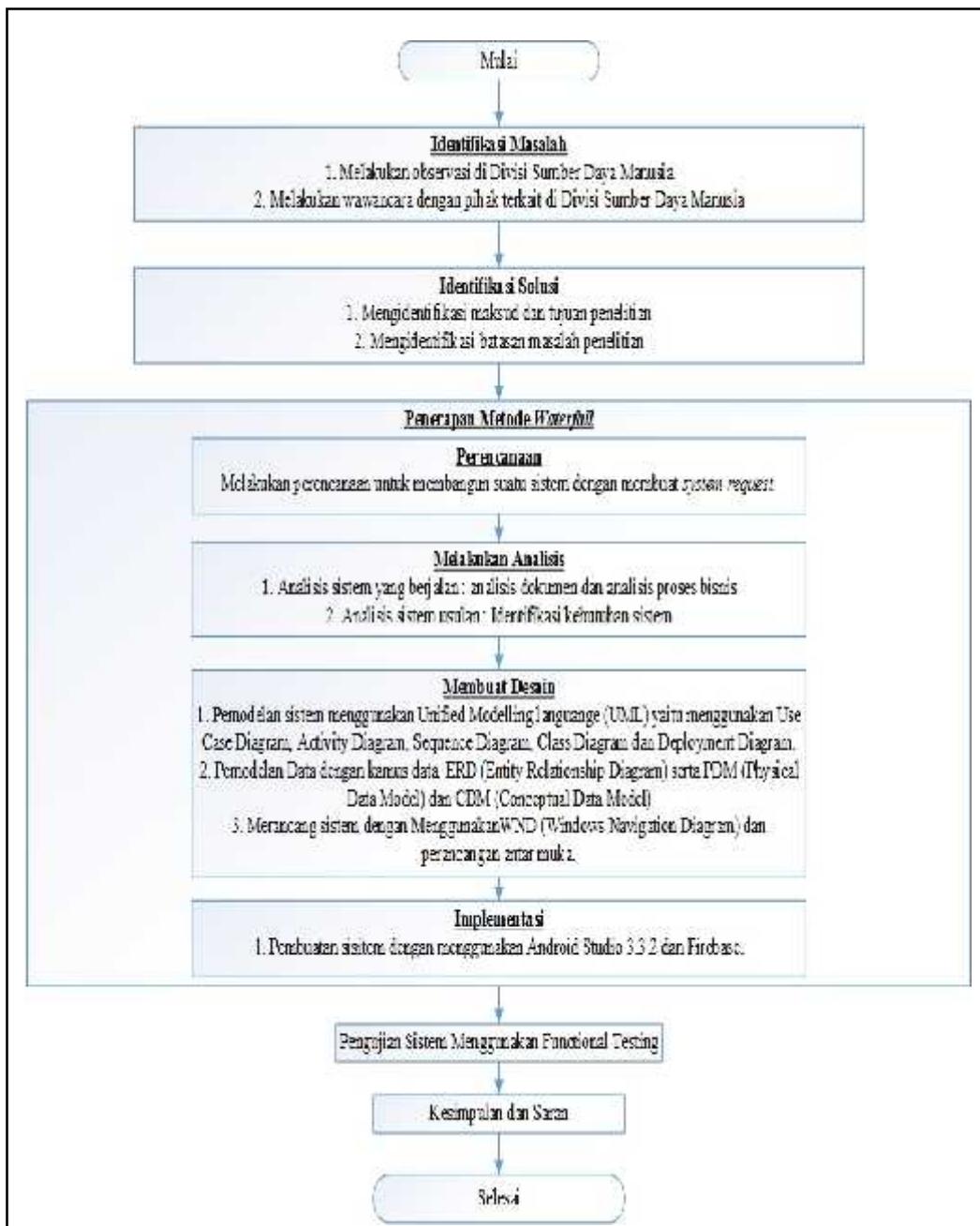
5. Pengujian Sistem dengan *Functional Testing*

Pengujian sistem digunakan untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan.

6. Kesimpulan dan Saran

Kesimpulan digunakan untuk membandingkan hasil penelitian atau pengembangan sistem dengan sistem sebelumnya. Sedangkan saran berisi masukan untuk pengembangan sistem selanjutnya.

Kerangka penelitian laporan tugas akhir dapat dilihat pada gambar III.1



Gambar III.1 Kerangka Penelitian

Sumber : Pengolahan Data, 2019

BAB IV

PENGUMPULAN DAN PENGOLAHAN DATA

4.1 Sejarah Perusahaan

PT Indonesia Kendaraan Terminal adalah salah satu anak perusahaan PT Pelabuhan Indonesia II (Persero), sektor industri yang menyediakan layanan terminal kendaraan yang beroperasi di pelabuhan Tanjung Priok, Jakarta, Indonesia. PT Indonesia Kendaraan Terminal didirikan sebagai entitas bisnis tersendiri pada tanggal 5 November 2012 dengan prosentase kepemilikan saham PT Pelabuhan Indonesia II (Persero) sebesar 99% dan PT Multi Terminal Indonesia sebesar 1%. Sebelum menjadi entitas bisnis tersendiri, PT Indonesia Kendaraan Terminal hanya sebuah *strategic business unit* yang bernama Tanjung Priok Car Terminal (TPT) yang pengelolaannya di bawah Kantor Pusat dan beroperasi sejak Juni 2007. Saat ini, PT Indonesia Kendaraan Terminal juga di kenal dengan nama IPC Car Terminal. PT Indonesia Kendaraan Terminal dirancang khusus untuk terminal komersial yang menyediakan layanan di terminal mobil. Layanan ini termasuk *Stevedoring, Cargodoring, Receiving, dan Delivery*. Selain itu, terminal ini juga berfungsi sebagai Pusat Pengolahan Kendaraan (VPC) dan Pusat Pengolahan Peralatan (EPC)

Seiring dengan semakin tingginya daya beli masyarakat dan meningkatnya kepercayaan produsen produk otomotif global, maka pada tahun 2016 Wakil Presiden Republik Indonesia pada saat itu, yaitu Bapak Muhammad Jusuf Kalla mengusulkan untuk didirikan pelabuhan khusus untuk melayani kegiatan bongkar muat kargo-kargo yang dimuat melalui kapal Ro-Ro (*Roll on Roll Off*), untuk menangani distribusi kendaraan baik antar pulau maupun ekspor dan impor. PT Indonesia Kendaraan Terminal terus berkomitmen untuk dapat berbagi melalui pengelolaan terminal kendaraan kelas dunia yang unggul dalam operasional dan pelayanan.

4.1.1 Profil Perusahaan

Profil perusahaan merupakan penjelasan mengenai perusahaan termasuk produknya secara verbal maupun grafik yang mengangkat *corporate value* serta *product value* serta keunggulan perusahaan dibandingkan pesaing. Adapun Profil dari PT Indonesia Kendaraan Terminal yaitu:

Nama Perusahaan	:	PT. Indonesia Kendaraan Terminal
Alamat Kantor	:	Jalan Sindang Laut No. 101, Cilincing, Jakarta Utara 14110
Telepon	:	+62 21 4393 2251
Fax	:	+62 21 4393 2250
Email	:	info@indonesiacarterminal.co.id
Website	:	www.indonesiacarterminal.co.id
Produksi	:	Ekspor dan impor mobil, alat berat dan spare part
Status	:	Perseroan Terbatas
Luas Lahan	:	Plant 27,4 ha
Luas Gedung	:	Plant 5 ha
Luas Kantor	:	Plant 1400 m ²
Jumlah Tenaga Kerja	:	365 orang

4.1.2 Visi, Misi, Nilai-Nilai dan Budaya Perusahaan

Visi merupakan pandangan jauh kedepan tentang suatu perusahaan ataupun lembaga dan hal lain-lain yang ingin dicapai pada masa yang akan datang. Visi juga dapat diartikan sebagai tujuan perusahaan atau lembaga dan apa yang harus dilakukan untuk mencapai tujuannya tersebut di masa depan. Visi dari PT Indonesia Kendaraan Terminal adalah “Menjadi terminal kendaraan berkelas dunia yang unggul dalam operasional dan pelayanan” .

Misi merupakan suatu pernyataan tentang apa yang harus dikerjakan oleh perusahaan atau lembaga dalam usaha mewujudkan visi tersebut. Misi perusahaan diartikan sebagai tujuan dan alasan mengapa perusahaan atau lembaga itu dibuat. Misi juga akan memberikan arah sekaligus batasan-batasan proses pencapaian tujuan. Misi dari PT Indonesia Kendaraan Terminal adalah “Sebagai pengelola

terminal kendaraan yang memaksimalkan nilai tambah bagi seluruh kepentingan (*stakeholders*) secara berkesinambungan dalam rangka meningkatkan perekonomian nasional”.

Adapun penjelasannya sebagai berikut:

1. Untuk masyarakat negara: menjamin kelancaran dari keamanan arus barang untuk mewujudkan efisiensi biaya logistik dalam rangka memacu pertumbuhan ekonomi yang berdampak pada peningkatan kesejahteraan masyarakat
2. Untuk pemegang saham: memaksimalkan nilai perusahaan bagi pemegang saham dan meningkatkan kesehatan perusahaan secara profesional dengan memenuhi aspek-aspek tata kelola perusahaan yang baik.
3. Untuk pelanggan dan mitra: menyediakan, membangun, dan mengoperasikan pelayanan terminal dan logistik secara terintegrasi, berkualitas dan handal untuk memenuhi kepuasan pelanggan dan mitra.
4. Untuk karyawan: menciptakan lingkungan kerja yang nyaman bagi karyawan, mewujudkan insan perusahaan yang fokus pada pelanggan, berintegrasi, bangga kepada perusahaan dan budayanya serta memberikan kesejahteraan dan kepuasan kepada karyawan.

PT Indonesia Kendaraan Terminal tidak hanya memiliki visi dan misi tetapi memiliki nilai-nilai perusahaan dan budaya perusahaan. Adapun nilainya adalah:

1. *Customer Centric “meet customer expectation”*: secara proaktif mencari tahu serta memahami kebutuhan pelanggan untuk memberikan solusi-solusi yang inovatif, membangun hubungan jangka panjang yang baik dengan para pelanggan serta secara konsisten memberikan pelayanan terbaik dan berkualitas untuk membantu para pelanggan tumbuh dan berkembang.
2. *Integrity “Walk The Talk”*: menumbuhkan rasa percaya dengan mengatakan apa yang kita rasakan serta melakukan apa yang kita ucapkan, menunjukkan sikap profesional dan jujur dalam berinteraksi dengan pihak eksternal maupun internal serta berperilaku disiplin dan patuh terhadap kode etik bisnis dalam melakukan pekerjaan kita sehari-hari.

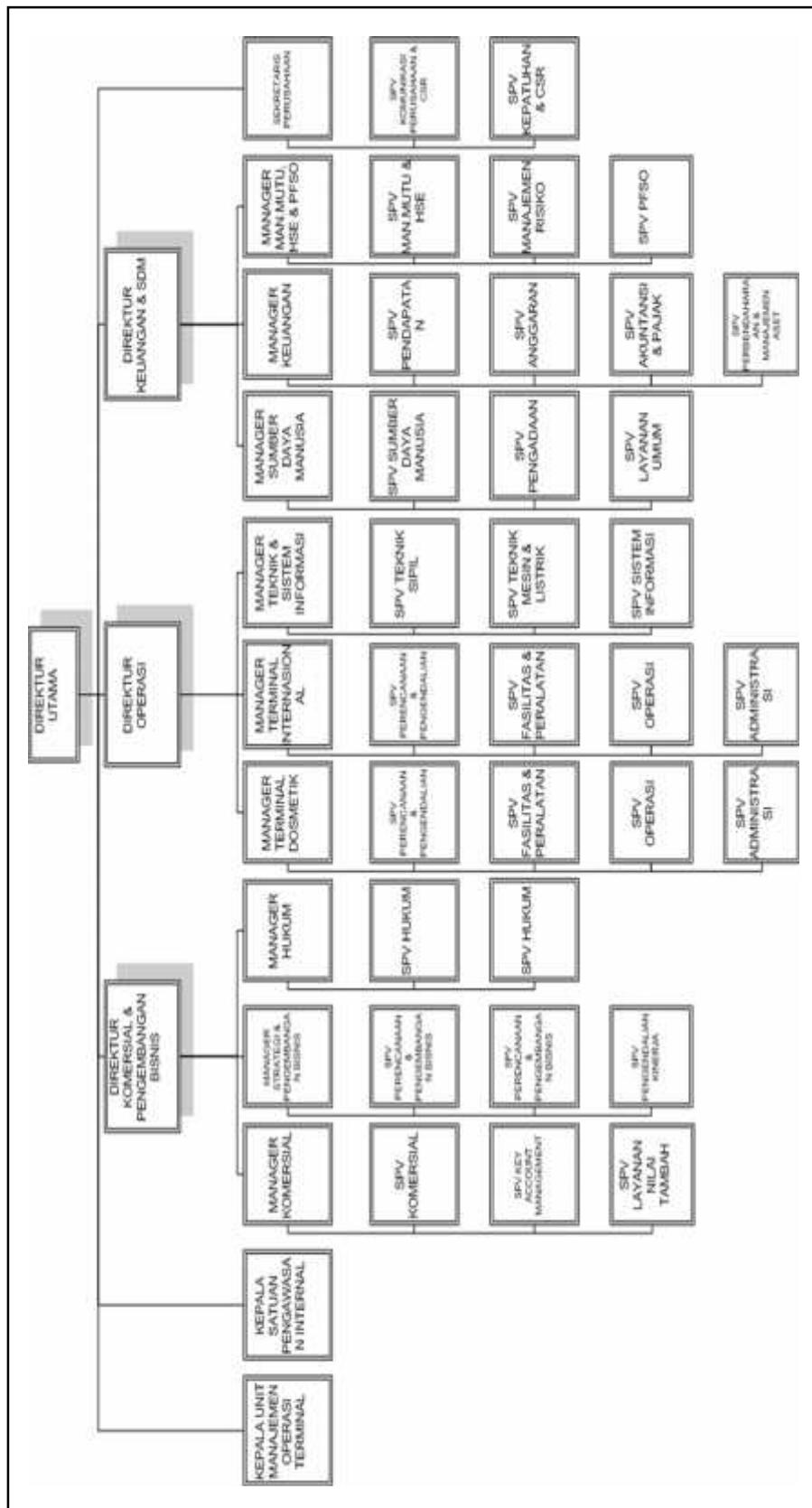
3. *Nationalism “National Pride”*: menumbuhkan semangat dan ikut berperan mensukseskan program pemerintah dalam pembangunan nasional, menumbuhkan rasa bangga dan semangat nasionalisme dalam berkarya serta terus berkembang dan mampu bersaing dengan perusahaan pengelola pelabuhan kelas dunia.
4. *Team Work “Together We Can”*: berkolaborasi dalam tim untuk mendapatkan hasil yang terbaik, bekerja bersama-sama menghasilkan ide-ide implementatif untuk solusi kebutuhan pengguna jasa pelabuhan serta semangat kebersamaan dan menghargai orang lain
5. *Action “Make It Happen”*: berani bermimpi dan berusaha mewujudkannya, proaktif untuk mencari cara dalam mewujudkan visi perusahaan serta melakukan terobosan-terobosan dan langkah nyata dalam mendorong perkembangan perusahaan.

Adapun Budaya Perusahaan yang terdapat di PT Indonesia Kendaraan Terminal adalah “Dengan CINTA, satukan langkah wujudkan kerja nyata IPC Car Terminal berkelas dunia”.

4.2 Struktur Organisasi Perusahaan

Struktur organisasi sangat berpengaruh terhadap kesuksesan sebuah perusahaan, terutama dalam mengatasi masalah sumber daya manusianya. Setiap perusahaan akan membuat struktur organisasi yang berbeda, sesuai dengan tujuan perusahaan, budaya perusahaan, aktivitas pekerjaan dan pola bisnis perusahaan.

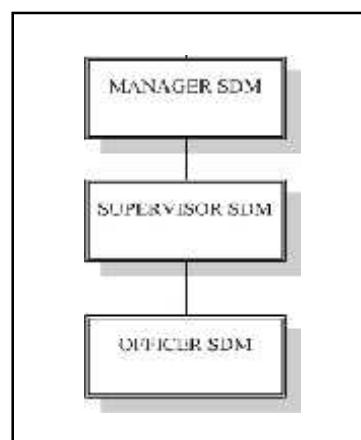
Struktur organisasi yang tersusun dengan baik akan memudahkan koordinasi antar sesama karyawan serta meningkatkan efektivitas dan efisiensi suatu perusahaan di dalam mencapai tujuannya. Berikut merupakan struktur organisasi PT Indonesia Kendaraan Terminal secara keseluruhan, untuk lebih jelasnya bisa dilihat pada Gambar IV.1.



Gambar IV.1 Struktur Organisasi PT Indonesia Kendaraan Terminal
(Sumber: PT Indonesia Kendaraan Terminal, 2018)

4.3 Struktur Organisasi Divisi Sumber Daya Manusia (SDM)

Struktur organisasi yang tersusun dengan baik akan memudahkan koordinasi karyawan satu dengan yang lainnya serta meningkatkan efektivitas dan efisiensi suatu perusahaan di dalam mencapai tujuannya. Berikut merupakan struktur organisasi pada Divisi Sumber Daya Manusia, untuk lebih jelasnya bisa dilihat pada Gambar IV.2.



Gambar IV.2 Struktur Divisi Sumber Daya Manusia
(Sumber: PT Indonesia Kendaraan Terminal, 2018)

Keterangan:

1. Manager Sumber Daya Manusia
 - a. Membantu Direktur Keuangan dan SDM dalam merumuskan, merencanakan, mengendalikan dan mengoordinasikan strategi implementasi program kerja manajemen dalam bidang Sumber Daya Manusia.
 - b. Bertanggung jawab dalam pelaksanaan program kerja manajemen dalam bidang Sumber Daya Manusia.
 - c. Melaksanakan monitoring dan evaluasi pelaksanaan program kerja dan
 - d. pencapaian kinerja bidang Sumber Daya Manusia.
2. Supervisor Sumber Daya Manusia
 - a. Membantu Manager SDM dalam merumuskan, merencanakan, mengendalikan dan mengoordinasikan strategi implementasi program kerja manajemen dalam bidang Sumber Daya Manusia.

- b. Bertanggung jawab dalam pelaksanaan program kerja manajemen dalam bidang Sumber Daya Manusia.
 - c. Melaksanakan monitoring dan evaluasi pelaksanaan program kerja dan pencapaian kinerja bidang Sumber Daya Manusia.
3. *Officer Sumber Daya Manusia*
 - a. Membantu Supervisor SDM dalam merumuskan, merencanakan, mengendalikan dan mengoordinasikan strategi implementasi program kerja manajemen dalam bidang Sumber Daya Manusia.
 - b. Bertanggung jawab dalam pelaksanaan program kerja manajemen dalam bidang Sumber Daya Manusia.
 - c. Mendata serta merekap pelaksanaan program kerja dan pencapaian kinerja bidang Sumber Daya Manusia.

4.4 Tenaga Kerja

Tenaga Kerja adalah setiap orang yang mampu melakukan pekerjaan guna menghasilkan barang atau jasa untuk memenuhi kebutuhan sendiri atau kebutuhan masyarakat (Undang-Undang RI No.13 Tahun 2003). Adapun ketentuan-ketentuan dari setiap tenaga kerja dijelaskan pada sub bab berikut.

4.4.1 Tenaga Organik

Tenaga Organik merupakan tenaga kerja tetap di PT Indonesia Kendaraan Terminal. Berikut ini ketentuan-ketentuan dari tenaga organik:

- 1 . Batasan jangka waktu lamanya bekerja atau usia pensiun yaitu 56 tahun.
2. Hubungan kerja antara perusahaan dan karyawan dituangkan dalam “Surat Perjanjian Kerja”.
3. Perusahaan dapat mensyaratkan masa percobaan maksimal 3 bulan.
4. Jika terjadi pemutusan hubungan kerja bukan karena pelanggaran berat atau karyawan mengundurkan diri maka karyawan tetap mendapatkan uang pesangon dan uang penggantian hak sesuai UU yang berlaku.
5. Jam kerja karyawan tetap berlaku *non shift* dari hari senin-jumat, untuk yang *shift* dibagi 2 *shift* dan 3 grup.

6. Karyawan yang telah mengabdi atau telah bekerja genap 1 tahun maka sesuai peraturan Undang-Undang maka karyawan tersebut berhak atas hak cuti sebagaimana yang telah dimaksud dalam Undang-Undang tersebut.
7. Proses pengangkatan karyawan menggunakan Surat Keputusan Direksi.

4.4.2 Tenaga Non Organik

Tenaga Non Organik merupakan tenaga kerja outsourcing atau kontrak di PT Indonesia Kendaraan Terminal. Berikut ini ketentuan-ketentuan dari tenaga non organik:

- 1 . Karyawan *Outsourcing* di pekerjakan oleh perusahaan untuk jangka waktu tertentu saja.
2. Hubungan Kerja antara perusahaan dan karyawan *outsourcing* dituangkan dalam Surat Perjanjian Kerja Waktu Tertentu (PKWT) untuk waktu 1 tahun atau 2 tahun dan Surat Perjanjian Kerja Waktu Tidak Tertentu (PKWTT) untuk karyawan yang di pekerjakan dipekerjaan tetap dan terus-menerus.
3. Jam kerja karyawan *outsourcing* berlaku *non shift* dari hari senin-jumat, untuk yang *shift* dibagi 2 *shift* dan 3 grup.
4. Pada gaji karyawan *outsourcing* terdapat perbedaan dalam jumlah nominal gaji dan tunjangan yang diterimanya.
5. Karyawan *outsourcing* sesuai peraturan perundang-undangan maka karyawan *outsourcing* tersebut ketentuan hak cutinya diatur dalam Perjanjian Kerja Bersama (PKB) perusahaan.

4.5 Pengajuan Cuti

Cuti merupakan keadaan dimana pekerja tetap yang telah lebih dari 1 tahun mengabdi atau telah bekerja pada perusahaan melakukan keadaan tidak bekerja selama jangka waktu yang telah ditetapkan. Sedangkan pengajuan cuti merupakan suatu proses dimana pekerja atau pegawai akan melakukan kegiatan pengambilan jatah cuti. Pekerja atau pegawai tersebut akan mengajukan cuti bila ada keperluan yang mendesak maupun hanya sekedar menenangkan diri dari beban kerja yang ditanggungkan kepada pekerja atau pegawai yang bersangkutan. Pengajuan cuti ini

dilaksanakan 3 hari sebelum ingin mengambil cuti dari hari terlaksananya cuti, pengajuan ini dilakukan oleh pegawai dengan membuat surat pengajuan cuti terlebih dahulu.

Berikut ini merupakan jenis-jenis cuti pada PT Indonesia Kendaraan Terminal:

1. Cuti Tahunan
 - a. Cuti tahunan adalah cuti yang di berikan kepada karyawan yang telah bekerja sekurang-kurangnya 1 (satu) tahun secara terus menerus
 - b. Lamanya cuti tahunan adalah 12 (dua belas) hari kerja.
 - c. Pada PT Indonesia Kendaraan Terminal, pengambilan cuti tahunan tidak dapat di pecah-pecah hingga jangka waktu kurang dari 3 (tiga) hari kerja.
 2. Cuti Besar
 - a. Cuti Besar adalah cuti yang di berikan kepada karyawan yang telah bekerja sekurang-kurangnya 6 (enam) tahun secara terus-menerus.
 - b. Lamanya cuti besar adalah 3 (tiga) bulan.
 - c. Karyawan yang sudah menjalani cuti besar tidak berhak lagi atas cuti tahunannya dalam tahun yang bersangkutan.
 3. Cuti Sakit
 - a. Karyawan yang sakit selama 1 (satu) atau 2 (dua) hari berhak atas cutinya.
 - b. Karyawan yang sakit lebih dari 2 (dua) hari berhak atas cuti sakit dengan ketentuan yang bersangkutan harus mengajukan permintaan secara tertulis kepada Supervisor atau Manager divisi masing-masing dengan melampirkan surat keterangan dokter.
 - c. Karyawan Wanita yang mengalami gugur kandung berhak atas cuti sakit paling lama $\frac{1}{2}$ (satu setengah) bulan terhitung dari hari kandungannya gugur dengan melampirkan surat keterangan dokter atau bidan.
 4. Cuti Bersalin
- Setiap karyawan wanita yang melahirkan berhak atas cuti bersalinnya, lamanya waktu yang diberikan yaitu 3 (tiga) bulan, yang pengambilannya disesuaikan dengan dengan kondisi kesehatan yang bersangkutan.

5. Cuti Karena Alasan Penting

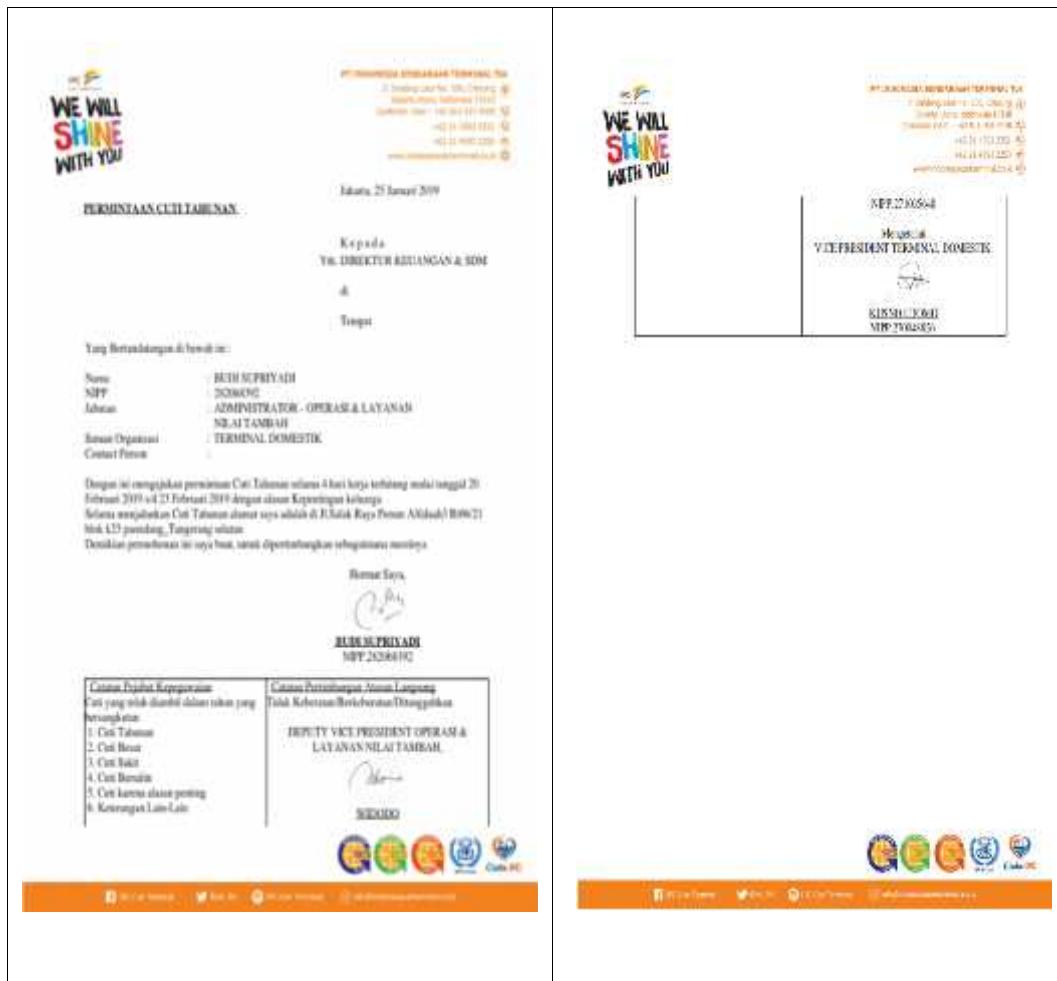
Cuti karena alasan penting adalah cuti yang diberikan karena ibu, bapak, isteri/suami, anak, adik, kakak, mertua, atau menantu sakit keras atau meninggal dunia, melangsungkan perkawinan pertama dan alasan penting lainnya. Lama waktu yang di berikan paling lama 2 (dua) bulan dengan melampirkan surat terlampir kepada Supervisor, Manager serta Direktur Divisi masing-masing.

6. Cuti Naik Haji

PT Indonesia Kendaraan Terminal memberikan hak cuti naik haji bagi karyawannya yang ingin melaksanakan ibadah haji, lamanya waktu yang diberikan paling lama 45 (empat puluh lima) hari. Bagi karyawan yang ingin mengajukan cuti naik haji sekurang-kurangnya 3 (tiga) hari sebelum keberangkatan.

4.6 Analisis Dokumen Cuti

Berikut ini Surat Pengajuan cuti pada PT Indonesia Kendaraan Terminal yang bisa dilihat pada Gambar IV.3.



Gambar IV.3 Surat Pengajuan Cuti
(Sumber: PT Indonesia Kendaraan Terminal, 2019)

Keterangan Surat Pengajuan Cuti:

1. Diisi Oleh Pemohon

Bagian ini berisi:

- Tanggal Surat, berisi tanggal bulan dan tahun saat mengajukan cuti.
- Nama, berisi nama dari pemohon yang mengajukan cuti.
- NIPP, berisi nomor induk pegawai yang mengajukan cuti.
- Jabatan, berisi nama jabatan dari pemohon yang mengajukan cuti.
- Satuan Organisasi, Berisi nama satuan organisasi dari pemohon yang mengajukan cuti.
- Contact Person*, berisi nomor telepon pemohon yang mengajukan cuti.

- g. Jumlah Cuti, berisi jumlah hari dan tanggal cuti yang akan diambil oleh pemohon.
 - h. Tanggal Cuti, berisi tanggal dimulai dan berakhirnya cuti yang diambil oleh pemohon.
 - i. Alasan Cuti, berisi alasan pemohon mengambil cuti.
 - j. Alamat Cuti, berisi alamat saat pemohon cuti.
2. Catatan Pejabat Kepegawaian
- Pada bagian ini berisi jenis cuti yang akan diambil oleh pemohon dengan memberikan keterangan () atau (O) pada jenis cuti tersebut.
3. Catatan Pertimbangan Atasan Langsung
- Pada bagian ini berisi apakah permohonan cuti mendapat izin dari atasan atau tidak, jika mendapatkan izin maka kolom tanda tangan yang telah disediakan ditandatangani (disetujui), jika cuti tidak mendapat izin maka kolom tersebut tidak ditandatangani dan atasan dapat langsung memberikan alasan tidak disetujui pengambilan cuti kepada pemohon. Bagian lainnya yakni berupa kolom tanda tangan *supervisor* dan *manager* beserta nama dan NIPP.

Berikut ini merupakan Surat Izin Cuti yang telah dibuat oleh Divisi Sumber Daya Manusia untuk diberikan kepada Direktur Keuangan dan SDM. (lihat gambar IV.4).



Gambar IV.4 Surat Izin Cuti
 (Sumber: PT Indonesia Kendaraan Terminal, 2019)

Keterangan Surat Izin Cuti:

1. **Diisi Oleh Divisi Sumber Daya Manusia**

Bagian ini berisi:

- a. Nomor Surat, berisi nomor surat dari Divisi Sumber Daya Manusia.
- b. Nama Pemohon, berisi nama pemohon yang mengajukan cuti.
- c. NIPP, berisi nomor induk pegawai pemohon yang mengajukan cuti.
- d. Pangkat/Golongan, berisi pangkat/golongan pemohon yang mengajukan cuti
- e. Jabatan, berisi jabatan pemohon yang mengajukan cuti.
- f. Satuan Organisasi, berisi keterangan PT Indonesia Kendaraan Terminal
- g. Alamat Cuti, berisi alamat selama cuti berlangsung.
- h. Lama Cuti, berisi lamanya cuti serta tanggal dimulai dan berakhirnya cuti yang diambil oleh pemohon.

2. Ketentuan

Bagian ini berisi ketentuan yang berlaku untuk dipatuhi oleh pemohon, adapun ketentuannya yaitu, sebelum dan sesudah cuti wajib melaporkan diri kepada atasan.

3. Pembuatan Surat Izin Cuti

Pada bagian ini berisi tempat serta tanggal pembuatan surat izin cuti serta kolom tanda tangan Direktur Keuangan dan SDM apabila disetujui, jika tidak disetujui Direktur Keuangan dan SDM dapat memberikan alasan tidak disetujuinya cuti langsung kepada pemohon.

Berikut ini merupakan *file* pencatatan riwayat pengambilan cuti yang dilakukan oleh Divisi Sumber Daya Manusia (lihat pada gambar IV.5).

PT. INDONESIA KENDARAAN TERMINAL			
No.	Nama	Hari Cuti	Hari
1	2	3	4
1	Slavy F	8 - 10 Jan	3
2	Irian Jaya	10 - 12 Jan	3
3	Adi W	8 - 12 Jan	5
4	Arifudin	10 - 12 Jan	3
5	Wasistianto	29 - 31 Jan	3
6	Adi S	15 - 17 Jan	3
7	Tegar Hamzah	18 - 22 Jan	3
8	Arief Maulana	31 Jan - 2 Feb	3
9	Abdul Hamid	22 - 24 Jan	3
10	Ibnu Hakim	28 Jan - 1 Feb	3
11	Mustafa	12 - 15 Feb	4
12	Eudi Supriadi	13 - 16 Feb	4
13	Rajab A	22 - 26 Feb	3
14	Viranthy O	12 - 15 Feb	4
15	Rimut Ahmad	26 - 28 Feb	3
16	Sugeng M	8 - 10 Feb	3

Cuti 2018 Rapel Cuti 2017 (7) Rapel Cuti 2019

Gambar IV.5 *file* Pencatatan Riwayat Cuti
(Sumber: PT Indonesia Kendaraan Terminal, 2019)

Keterangan *File* Pencatatan Riwayat Cuti:

Pada Bagian ini file tersebut diisi oleh bagian Sumber Daya Manusia

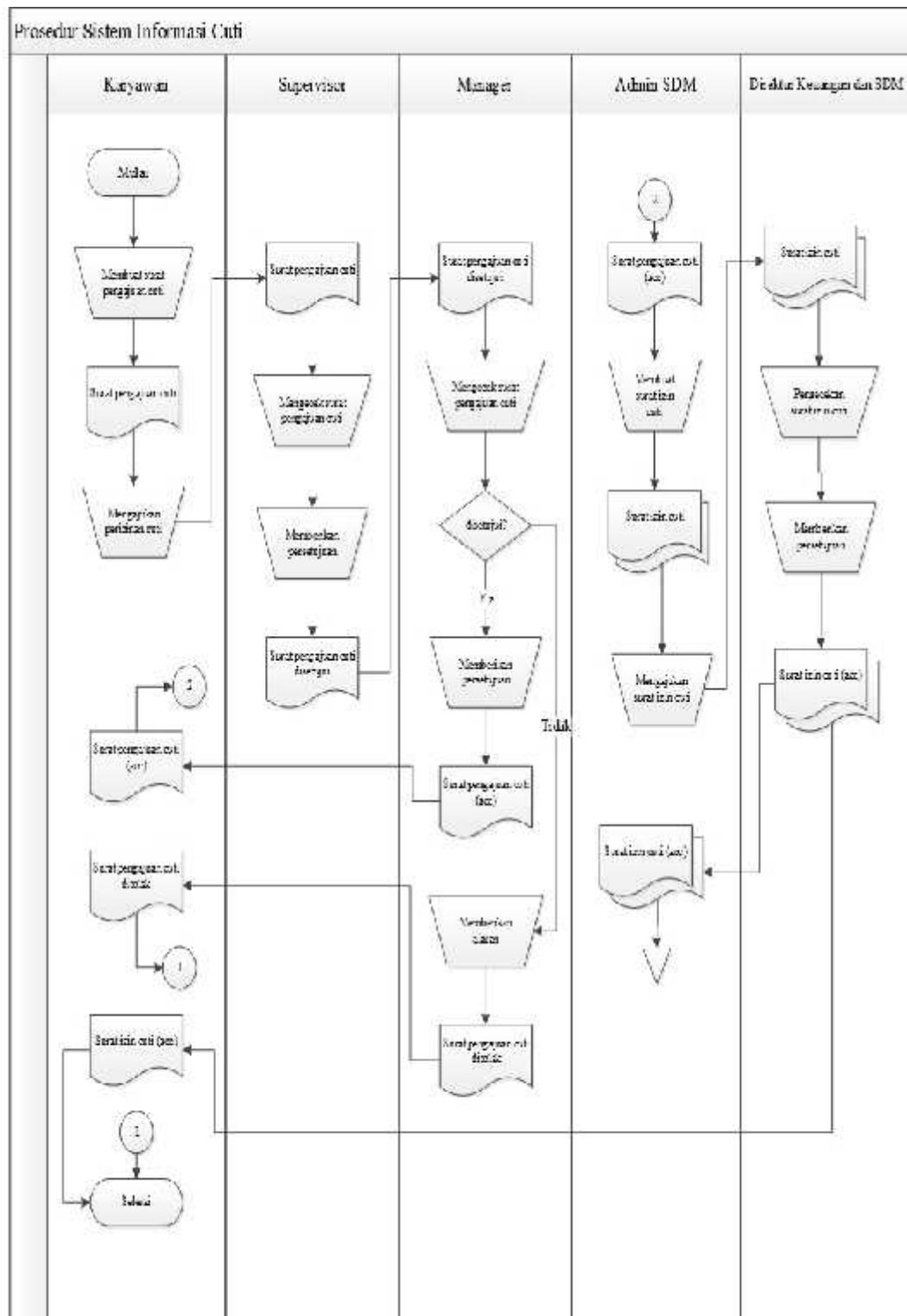
1. Nama, berisi nama-nama pemohon yang sudah mengambil cuti.
2. Hari Cuti, berisi tanggal dimulai dan berakhir cuti yang diambil oleh pemohon.
3. Hari, berisi durasi lamanya waktu pengambilan cuti.

4.7 Prosedur Sistem yang Sedang Berjalan

Pada bagian ini prosedur sistem yang sedang berjalan akan direpresentasikan dengan menggunakan diagram alir.

Keterangan:

1. Karyawan yang bersangkutan membuat surat pengajuan cuti terlebih dahulu, setelah sudah dibuat karyawan dapat mengajukan surat pengajuan cuti kepada *supervisor* dan *manager* di divisi masing-masing untuk mendapatkan persetujuan.
2. Kemudian, *supervisor* dan *manager* mengecek surat pengajuan cuti karyawan.
3. Jika diizinkan maka *Manager* dan *Supervisor* memberikan perizinan berupa tanda tangan. Jika tidak maka *Manager* memberikan alasan penolakan cuti yang diajukan oleh karyawan.
4. Surat pengajuan cuti yang telah ditandatangani oleh atasan langsung diserahkan kepada admin Sumber Daya Manusia untuk dibuatkan Surat Izin Cuti, Kemudian surat izin cuti tersebut diserahkan kepada Direktur Keuangan dan SDM untuk dimintai persetujuannya
5. Kemudian, Direktur Keuangan dan SDM mengecek surat izin cuti karyawan.
6. lalu Direktur keuangan dan SDM memberikan perizinan berupa tanda tangan
7. Setelah mendapatkan perizinan cuti dari Direktur Keuangan dan SDM, admin divisi sumber daya manusia memberikan surat izin cuti kepada karyawan serta mengarsipnya



Gambar IV.6 Diagram Alir Sistem Informasi Cuti Karyawan
 (Sumber: Hasil Analisis, 2019)

BAB V

ANALISIS DAN PEMBAHASAN

5.1 Analisis Kebutuhan Sistem

Tujuan dari tahap analisis memahami dengan sesungguhnya kebutuhan dari sistem yang baru dan mengembangkan sebuah sistem yang mewadahi kebutuhan tersebut atau memutuskan bahwa pengembangan sistem yang baru tidak dibutuhkan. Berikut ini adalah daftar kebutuhan sistem untuk aplikasi sistem informasi cuti karyawan usulan pada Divisi Sumber Daya Manusia (SDM).

Tabel V.1 Analisis Kebutuhan Sistem

No	Identifikasi Masalah	Kebutuhan User	Solusi	Kebutuhan Sistem
1.	Terhambatnya proses persetujuan cuti karyawan oleh Supervisor dan manager yang disebabkan dinas luar atau sibuk	Sistem yang dapat membantu mengelola persetujuan cuti karyawan	Merancang dan membangun suatu sistem informasi cuti karyawan yang dapat melakukan persetujuan dimanapun dan kapanpun	Proses yang dapat dilakukan pada pengelolaan persetujuan: 1. Dapat melihat daftar pengajuan cuti yang diinput oleh karyawan. 2. Dapat melihat detail pengajuan cuti yang diinput oleh karyawan. 3. Dapat melakukan pemberitahuan kepada karyawan berupa hasil persetujuan atau penolakan.

Tabel V.1 Analisis Kebutuhan Sistem (Lanjutan)

No	Identifikasi Masalah	Kebutuhan User	Solusi	Kebutuhan Sistem
2.	Tidak terdapatnya ruang penyimpanan data secara permanen sehingga terjadinya kehilangan ataupun kerusakan data cuti	Sistem yang dapat membantu mengelola dan menyimpan data.	Merancang dan membangun suatu sistem informasi cuti karyawan yang terhubung dengan basis data.	Sistem dapat mengelola data diantaranya data karyawan, data cuti karyawan, data libur nasional dan data cuti bersama. Pada proses ini yang dapat dilakukan yaitu mengubah, menambah, menghapus dan melihat
3.	Rekap data riwayat cuti karyawan tidak terdapat keterangan jenis cuti dan alasan cuti.	Sistem yang dapat mengelola perekapan data riwayat cuti secara informatif	Merancang dan membangun suatu sistem informasi dengan keterangan yang akurat	Sistem dapat menampilkan data riwayat cuti karyawan secara akurat
4.	Tidak tersedianya informasi mengenai sisa saldo cuti karyawan	Tersedianya sisa saldo cuti yang akurat.	Penyediaan informasi saldo cuti secara otomatis.	Proses yang dapat dilakukan pada Pengelolaan Informasi dan saldo cuti yaitu: Dapat mencari data cuti yang telah diambil dan sisa saldo cuti.

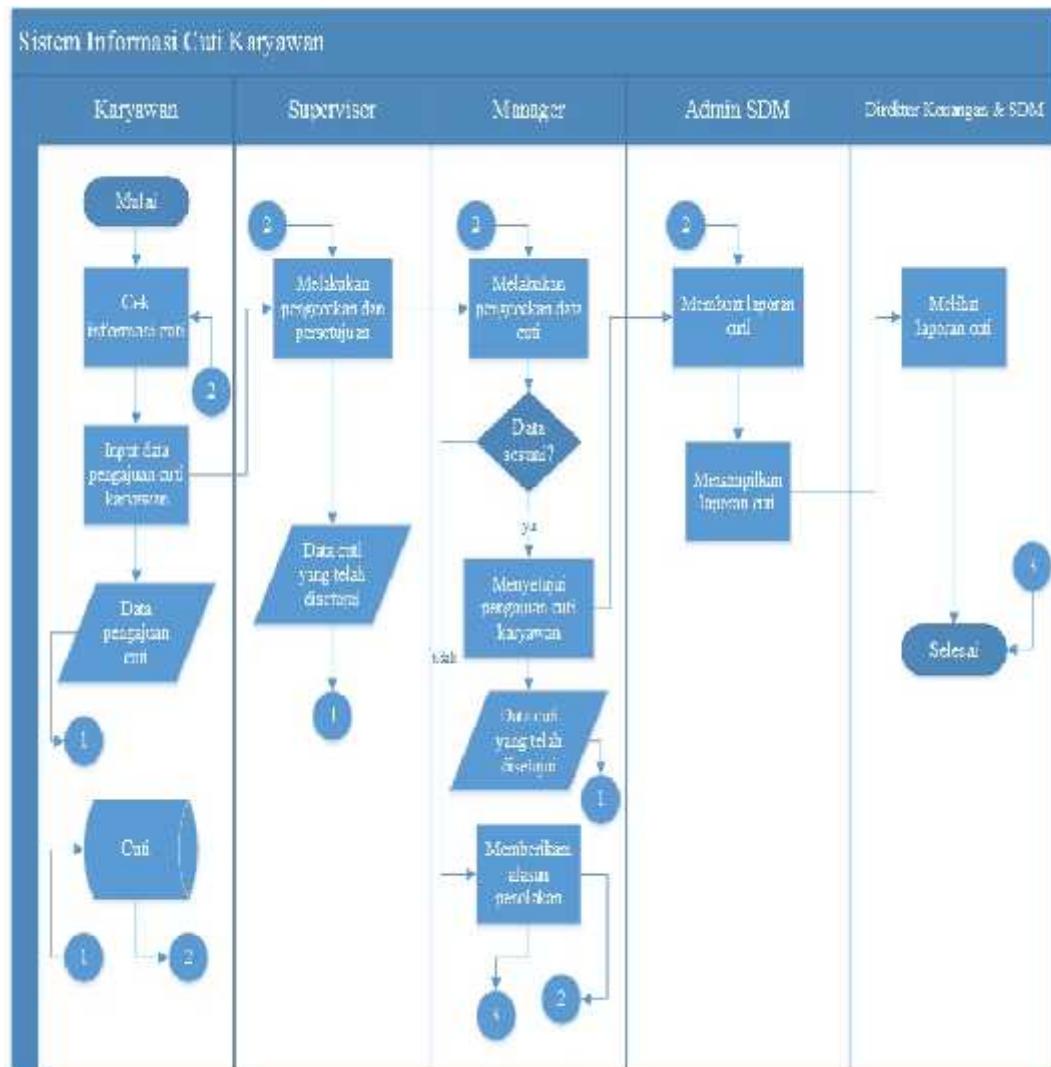
Sumber:Hasil Analisis (2019)

5.2 Prosedur Usulan Sistem Informasi Cuti Karyawan

Prosedur sistem informasi cuti karyawan yang diusulkan melalui tahap sebagai berikut:

1. Karyawan login dan masuk ke sistem, lalu cek informasi cuti dan memilih tombol yang tersedia untuk melakukan pengajuan.
2. Karyawan menginputkan data cuti sesuai list data yang telah disediakan oleh sistem.
3. Data pengajuan cuti yang telah diinputkan oleh karyawan akan masuk akun Supervisor terlebih dahulu untuk di cek pengajuan cuti beserta alasannya kemudian disetujui dan baru kemudian akan masuk akun Manajer sesuai divisi masing-masing.
4. Data pengajuan cuti yang masuk ke Manajer akan dicek pengajuan cuti beserta alasannya, kemudian Manager memutuskan apakah akan disetujui atau tidak. Jika tidak disetujui maka pengajuan ditolak dan disertai alasannya kemudian akan *update* ke proses selanjutnya, jika disetujui oleh Supervisor maka akan *update* ke proses selanjutnya.
5. Jika telah mendapat persetujuan oleh semua pihak penyetuju yaitu supervisor dan manager karyawan dapat melihat informasi di sistem.
6. Admin SDM akan membuat laporan cuti yang berisi rekapan riwayat cuti untuk diberikan kepada Manager SDM dan Direktur Keuangan dan SDM.

Seperti yang dapat dilihat pada Gambar V.1 adalah *Flowmap* Sistem Informasi Cuti Karyawan usulan sebagai berikut:



Gambar V.1 Flowmap Sistem Informasi Cuti Karyawan Usulan
Sumber: Hasil Analisis (2019)

5.3 Analisis dan Perancangan Sistem Usulan

Divisi Sumber Daya Manusia memiliki beberapa kendala khususnya pada sistem informasi cuti, maka diperlukan sistem usulan baru menggunakan aplikasi *mobile* yang dapat mendukung pengolahan dan penyimpanan data maupun ketersediaan informasi agar sistem cuti dapat berjalan dengan baik. Dengan menggunakan aplikasi ini diharapkan dapat membuat proses manajemen cuti di PT Indonesia Kendaraan Terminal terintegrasi dan mudah dilaksanakan.

Sistem informasi yang diusulkan adalah berupa pengajuan cuti karyawan yang manual menjadi otomatis, proses persetujuan dapat dilakukan menggunakan sistem, tersedianya informasi cuti dan perekapan data yang dapat disimpan di *database*. Adapun analisis dan perancangan sistem ini menggunakan model pengembangan perangkat lunak *Waterfall*, menggunakan *tools* pemodelan sistem yaitu UML (*Unified Modeling Language*) yang meliputi pembuatan *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *deployment diagram*. Pemodelan data yang digunakan adalah *Entity Relationship Diagram*, *Conceptual Data Model* dan kamus data. Perancangan aplikasi menggunakan *Windows Navigation Diagram* (WND).

5.4 Pemodelan Sistem Informasi Cuti Karyawan Usulan

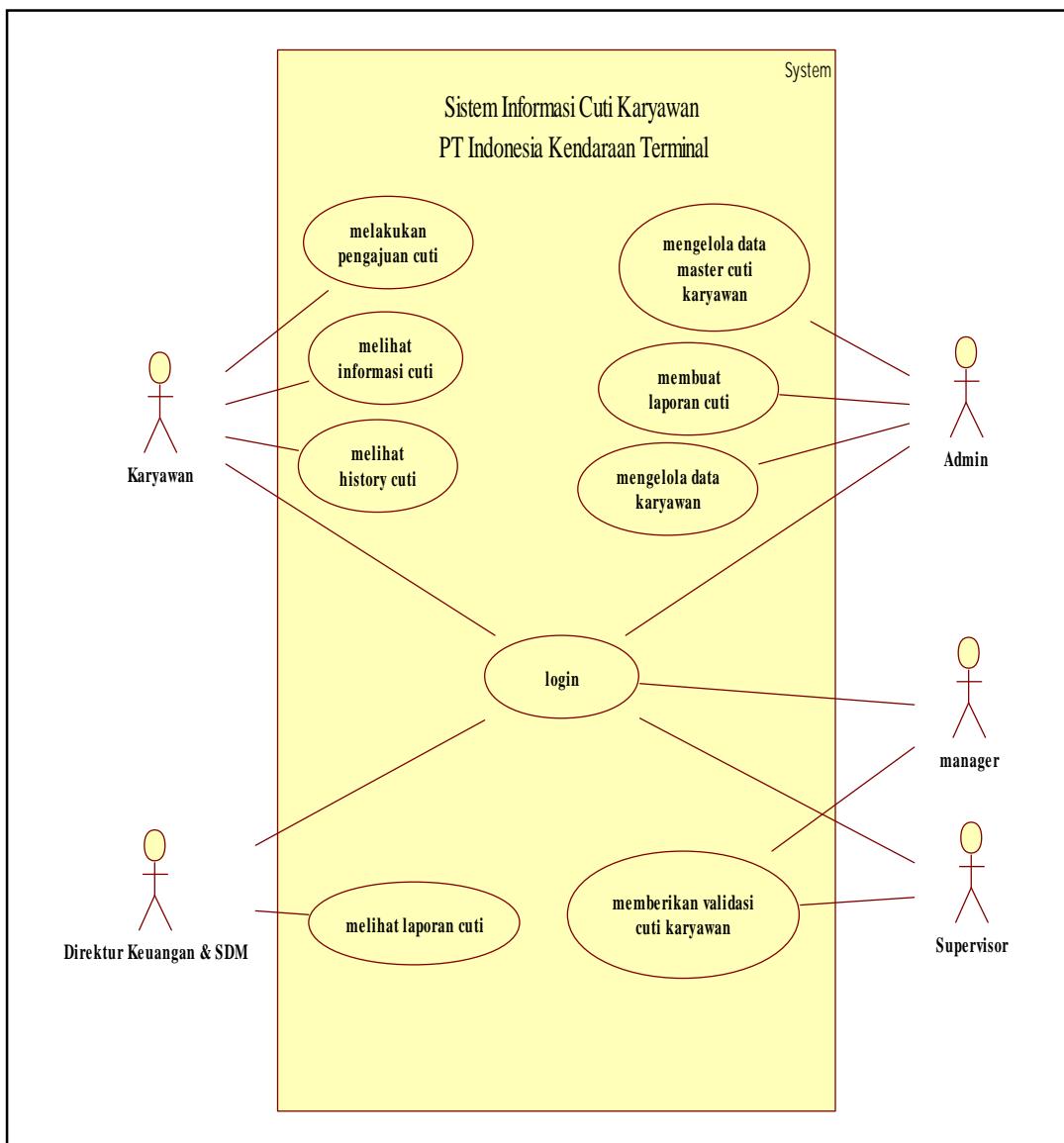
Pemodelan sistem informasi cuti karyawan yang diusulkan ini menggunakan UML (*Unified Modeling Language*) yang meliputi *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*, dan *deployment diagram* yang akan dipaparkan di bawah ini.

5.4.1 Use Case Diagram

Perancangan *use case diagram* sistem informasi cuti karyawan dijelaskan di bawah ini.

1. Use Case Diagram

Use case diagram menggambarkan interaksi aktor-aktor yang ada di sistem informasi cuti karyawan. Rancangan *use case diagram* sistem informasi cuti karyawan yang diusulkan dapat dilihat pada Gambar V.2.



Gambar V.2 Use Case Diagram
(Sumber: Hasil Analisis, 2019)

2. Definisi Aktor

Pendefinisian aktor pada *use case* perancangan sistem informasi sistem informasi cuti karyawan usulan dapat dilihat pada Tabel V.2.

Tabel V.2 Definisi Aktor *Use Case Diagram* Sistem Usulan

No.	Aktor	Definisi
1.	Karyawan	Karyawan dapat menginput data pengajuan cuti, mengecek <i>history</i> dan informasi cuti.
2.	Supervisor Divisi	Supervisor Divisi bertugas mengecek dan memberi persetujuan pengajuan cuti karyawan.
3.	Manager Divisi	Manajer Divisi bertugas mengecek dan memberi persetujuan maupun penolakan data cuti yang telah disetujui sebelumnya oleh Supervisor Divisi.
4.	Direktur Keuangan & SDM	Direktur Keuangan & SDM dapat melihat laporan cuti karyawan berdasarkan bulan dan tahun.
5.	Admin SDM	Admin SDM bertugas mengelola data cuti karyawan, data karyawan, data hari libur nasional, data cuti bersama, laporan cuti karyawan, informasi kegiatan dan dapat melihat status pengajuan cuti karyawan.

Sumber: Hasil Analisis (2019)

3. Definisi *Use Case*

Pendefinisian *use case* pada *use case diagram* berdasarkan usulan sistem informasi cuti karyawan dapat dilihat pada Tabel V.3 sebagai berikut:

Tabel V.3 Definisi *Use Case*

No	Nama <i>Use case</i>	Definisi
1.	<i>Login</i>	Proses melakukan <i>login</i> pada aplikasi dengan memasukan <i>username</i> dan <i>password</i>
2.	Melakukan pengajuan cuti	Proses mengelola data pengajuan cuti yang terdiri dari tambah, ubah dan hapus pada data pengajuan cuti
3.	Melihat informasi cuti	Proses melihat informasi cuti yang terdiri dari informasi kegiatan, informasi libur nasional dan cuti bersama yang dibuat oleh admin untuk karyawan

Tabel V.3 Definisi *Use Case* (Lanjutan)

No	Nama <i>Use Case</i>	Definisi
4.	Melihat history cuti	Proses melihat history cuti yang berisi list data cuti yang sedang diajukan dengan keterangan status yaitu disetujui atau ditolak
5.	Mengelola data master cuti karyawan	Proses mengelola data master cuti yang terdiri dari data cuti bersama, data libur nasional dan data informasi kegiatan dengan akses tambah, hapus dan ubah data.
6.	Mengelola data karyawan	Proses mengelola data karyawan yang terdiri dari tambah, ubah dan hapus pada data karyawan
7.	Membuat laporan cuti	Proses membuat laporan cuti yang berisi rekapan data riwayat cuti karyawan dengan skala waktu yang ditentukan
8.	Memberikan validasi cuti karyawan	Proses persetujuan yang diikuti dengan alasan apabila pengajuan cuti ditolak
9.	Melihat Laporan Cuti	Proses melihat laporan cuti yang berisi rekapan data riwayat cuti karyawan berdasarkan keterangan bulan dan tahun

4. Skenario *Use Case*

Skenario jalannya masing-masing *use case* pada usulan sistem informasi pembelian *spare part* mesin *stamping* akan dijelaskan sebagai berikut:

a. *Use Case Description Login*

Berikut adalah deskripsi skenario *use case diagram login* yang dapat dilihat pada Tabel V.4.

Tabel V.4 Use Case Description Login

Nama <i>Use Case</i>	<i>Login.</i>
Deskripsi <i>Use Case</i>	Menjelaskan proses <i>login</i> yang dilakukan oleh semua pengguna yang masuk ke sistem informasi cuti karyawan.
Aktor	Karyawan, Supervisor, Manager, Direktur Keuangan & SDM, Admin.
<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Karyawan melakukan pengisian <i>username</i> dan <i>password</i>, lalu klik tombol <i>login</i> kemudian akan masuk ke halaman awal karyawan 2. Supervisor melakukan pengisian <i>username</i> dan <i>password</i>, lalu klik tombol <i>login</i> kemudian akan masuk ke halaman awal supervisor 3. Manager melakukan pengisian <i>username</i> dan <i>password</i>, lalu klik tombol <i>login</i> yang akan masuk ke halaman awal manager 4. Direktur keuangan & SDM melakukan pengisian <i>username</i> dan <i>password</i>, lalu klik tombol <i>login</i> kemudian akan masuk ke halaman awal direktur 5. Admin melakukan pengisian <i>username</i> dan <i>password</i>, lalu klik tombol <i>login</i> kemudian akan masuk ke halaman awal admin

Sumber: Hasil Analisis Data (2019)

b. *Use case Description Mengelola Pengajuan Cuti*

Berikut adalah deskripsi skenario *use case diagram* mengelola pengajuan cuti yang dapat dilihat pada Tabel V.5.

Tabel V.5 *Use Case Description* Melakukan Pengajuan Cuti

Nama <i>Use Case</i>	Mengelola Pengajuan Cuti.
Deskripsi <i>Use Case</i>	<i>Use case</i> ini adalah proses untuk mengelola data pengajuan cuti yang akan diajukan oleh karyawan,
Aktor	Karyawan.
<i>Normal Flow Events</i>	<ol style="list-style-type: none"> 1. Karyawan masuk ke halaman awal karyawan. 2. Karyawan mengklik tombol “+ (add)” untuk melakukan pengisian data pengajuan cuti lalu klik simpan maka sistem memperbarui <i>database</i> pengajuan cuti. 3. Jika karyawan melakukan: <ul style="list-style-type: none"> - Klik lihat kemudian sistem akan menampilkan detail pengajuan cuti - Klik tombol ubah kemudian sistem akan menampilkan <i>form</i> ubah data pengajuan cuti dan <i>user</i> mengisi perubahan data lalu klik tombol simpan maka sistem akan memperbarui <i>database</i> pengajuan cuti - Klik tombol hapus, sistem akan menghapus data - Tambah data, <i>User</i> mengklik tombol “+ (add)” untuk melakukan pengisian data pengajuan cuti lalu klik simpan maka sistem memperbarui <i>database</i> cuti.
Persyaratan	Karyawan telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

c. *Use case Description* Melihat Informasi Cuti

Berikut adalah deskripsi skenario *use case diagram* melihat informasi cuti yang dapat dilihat pada Tabel V.6.

Tabel V.6 *Use Case Description* Melihat Informasi Cuti

Nama <i>Use Case</i>	Melihat Informasi Cuti
Deskripsi <i>Use Case</i>	Menjelaskan proses untuk melihat informasi cuti (hari libur nasional dan cuti bersama) dan kegiatan karyawan.
Aktor	Karyawan.
<i>Normal Flow Events</i>	<ol style="list-style-type: none"> 1. Karyawan masuk ke halaman awal karyawan. 2. Karyawan melihat list data informasi. 3. Karyawan mengklik list data informasi 4. Jika diklik akan muncul deskripsi informasi yang dipilih.
Persyaratan	Karyawan telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

d. *Use Case Description* Melihat History Cuti

Berikut adalah deskripsi skenario *use case diagram* melihat *history* cuti yang dapat dilihat pada Tabel V.7.

Tabel V.7 Use Case Description Melihat History Cuti

Nama <i>Use Case</i>	Melihat <i>History</i> Cuti
Deskripsi <i>Use Case</i>	Menjelaskan proses untuk melihat list data cuti yang diajukan serta adanya status pengajuan (disetujui atau ditolak).
Aktor	Karyawan.
<i>Normal Flow Events</i>	<ol style="list-style-type: none"> 1. Karyawan masuk ke tampilan menu awal karyawan 2. Karyawan mengklik menu lalu klik riwayat cuti 3. Sistem akan menampilkan detail izin cuti berikut dengan keterangan disetujui atau ditolak
Persyaratan	Karyawan telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

e. *Use Case Description* Mengelola Data Master Hari Libur Nasional

Berikut adalah deskripsi skenario *use case diagram* mengelola data master cuti karyawan tentang hari libur nasional yang dapat dilihat pada Tabel V.8.

Tabel V.8 *Use Case Description* Mengelola Data Master Hari Libur Nasional

Nama <i>Use Case</i>	Mengelola Data Master Cuti Karyawan
Deskripsi <i>Use Case</i>	Menjelaskan proses untuk melakukan pengelolaan data master cuti karyawan berupa informasi cuti tentang data hari libur nasional
Aktor	Admin SDM.
Normal Flow	<ol style="list-style-type: none"> 1. Admin SDM masuk ke halaman awal admin 2. Admin SDM mengklik menu informasi maka sistem menampilkan list data informasi. 3. Admin SDM mengklik <i>dropdown</i> informasi libur nasional maka sistem akan menampilkan detail <i>form</i> informasi libur nasional 4. Jika admin melakukan: <ul style="list-style-type: none"> - Tambah data, klik list data untuk melakukan pengisian informasi tentang hari libur nasional - Klik ubah, kemudian sistem akan menampilkan <i>form</i> ubah data informasi hari libur nasional dan <i>user</i> mengisi perubahan data lalu klik tombol simpan maka sistem akan memperbarui <i>database</i> - Klik hapus, sistem akan menghapus data
Persyaratan	Admin telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

f. *Use Case Description* Mengelola Data Master Cuti Bersama

Berikut adalah deskripsi skenario *use case diagram* mengelola data master cuti karyawan tentang cuti bersama yang dapat dilihat pada Tabel V.9.

Tabel V.9 *Use Case Description* Mengelola Data Master Cuti Bersama

Nama <i>Use Case</i>	Mengelola Data Master Cuti Karyawan
Deskripsi <i>Use Case</i>	Menjelaskan proses untuk melakukan pengelolaan data master cuti karyawan yang berisi informasi cuti tentang cuti bersama
Aktor	Admin SDM.
Normal Flow	<ol style="list-style-type: none"> 1. Admin SDM masuk ke halaman awal admin 2. Admin SDM mengklik menu informasi maka sistem menampilkan list data informasi. 3. Admin SDM mengklik <i>dropdown</i> informasi cuti bersama maka sistem akan menampilkan detail <i>form</i> informasi cuti bersama 4. Jika admin melakukan : <ul style="list-style-type: none"> - Tambah data, klik list data untuk melakukan pengisian informasi tentang cuti bersama - Klik ubah, kemudian sistem akan menampilkan <i>form</i> ubah data informasi cuti bersama dan <i>user</i> mengisi perubahan data lalu klik tombol simpan maka sistem akan memperbarui <i>database</i> - Klik hapus, sistem akan menghapus data
Persyaratan	Admin telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

g. *Use Case Description* Mengelola Data Master Data Cuti

Berikut adalah deskripsi skenario *use case diagram* mengelola data master cuti karyawan yang berisi status data cuti yang dapat dilihat pada Tabel V.10.

Tabel V.10 *Use Case Description* Mengelola Data Master Data Cuti

Nama <i>Use Case</i>	Mengelola Data Master Cuti Karyawan
Deskripsi <i>Use Case</i>	Menjelaskan proses untuk melakukan pengelolaan data master cuti karyawan tentang melihat data cuti yaitu status diajukan, disetujui atau ditolak
Aktor	Admin SDM.
Normal Flow	<ol style="list-style-type: none"> 1. Admin SDM masuk ke halaman awal admin 2. Admin SDM mengklik menu data cuti maka sistem akan menampilkan list data cuti karyawan 3. Jika admin melakukan: <ul style="list-style-type: none"> - Klik tab diajukan, maka sistem akan menampilkan list data cuti karyawan yang sedang diajukan - Klik tab disetujui, maka sistem akan menampilkan list data cuti karyawan yang sudah disetujui - Klik tab ditolak, maka sistem akan menampilkan list data cuti karyawan yang ditolak berikut alasan penolakan
Persyaratan	Admin telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

h. *Use Case Description* Mengelola Data Karyawan

Berikut adalah deskripsi skenario *use case diagram* mengelola data karyawan yang dapat dilihat pada Tabel V.11.

Tabel V.11 *Use Case Description* Mengelola Data Karyawan

Nama <i>Use Case</i>	Mengelola Data Karyawan
Deskripsi <i>Use Case</i>	Menjelaskan proses untuk melakukan pengelolaan data karyawan
Aktor	Admin SDM.

Tabel V.11 *Use Case Description* Mengelola Data Karyawan (Lanjutan)

<i>Normal Flow</i>	<ol style="list-style-type: none"> 1. Admin SDM masuk ke halaman awal admin 2. Admin SDM mengklik menu data karyawan, maka sistem akan menampilkan list data karyawan 3. Admin SDM mengklik dropdown divisi maka sistem akan menampilkan list data karyawan setiap divisi 4. Jika admin melakukan: <ul style="list-style-type: none"> - Klik tambah untuk melakukan pengisian data karyawan - Klik hapus, maka sistem akan menghapus data Klik ubah kemudian sistem akan menampilkan <i>form</i> ubah data informasi cuti bersama dan <i>user</i> mengisi perubahan data lalu klik tombol simpan maka sistem akan memperbarui <i>database</i>
Persyaratan	Admin telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

i. *Use Case Description* Membuat Laporan Cuti

Berikut adalah deskripsi skenario *use case diagram* membuat dan menampilkan laporan cuti yang dapat dilihat pada Tabel V.13.

Tabel V.12 *Use Case Description* Membuat Laporan Cuti

Nama <i>Use Case</i>	Membuat Laporan Cuti
Deskripsi <i>Use Case</i> .	Menjelaskan proses untuk membuat dan menampilkan laporan cuti karyawan setiap divisi berdasarkan periode tertentu.
Aktor.	Admin SDM.
<i>Normal Flow:</i>	<ol style="list-style-type: none"> 1. Admin SDM masuk ke halaman awal admin 2. Admin SDM mengklik menu laporan cuti maka sistem akan menampilkan list data riwayat cuti karyawan

Tabel V.12 *Use Case Description* Membuat Laporan Cuti (Lanjutan)

	<p>3. Jika admin melakukan :</p> <ul style="list-style-type: none"> - Klik <i>dropdown</i> divisi, <i>dropdown</i> jenis cuti dan tombol <i>select</i> bulan dan tahun maka sistem akan menampilkan list data riwayat cuti berdasarkan divisi dan jenis cuti yang kita pilih
Persyaratan	Admin telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

j. *Use Case Description* Memberikan Validasi Cuti Karyawan

Berikut adalah deskripsi skenario *use case diagram* memberikan persetujuan cuti karyawan yang dapat dilihat pada Tabel V.13.

Tabel V.13 *Use Case Description* Memberikan Validasi Cuti Karyawan (1)

Nama <i>Use Case</i>	Memberikan Validasi Cuti Karyawan
Deskripsi <i>Use Case</i> .	Menjelaskan proses untuk melihat daftar pengajuan cuti karyawan, mengecek dan menyetujui oleh Supervisor Divisi
Aktor.	Supervisor
<i>Normal Flow:</i>	<ol style="list-style-type: none"> 1. Supervisor masuk ke halaman awal supervisor 2. Supervisor melihat daftar pengajuan cuti karyawan.. 3. Supervisor kemudian mengklik list pengajuan cuti. Kemudian akan tampil detail pengajuan. 4. Supervisor melihat dan mengecek data pengajuan cuti. 5. Supervisor mengklik tombol “setuju” untuk memberikan persetujuan cuti.
Persyaratan	Supervisor telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

k. *Use Case Description* Memberikan Validasi Cuti Karyawan

Berikut adalah deskripsi skenario *use case diagram* memberikan persetujuan cuti karyawan yang dapat dilihat pada Tabel V.14.

Tabel V.14 *Use Case Description* Memberikan Validasi Cuti Karyawan (2)

Nama <i>Use Case</i>	Memberikan Validasi Cuti Karyawan
Deskripsi <i>Use Case</i> .	Menjelaskan proses untuk melihat daftar pengajuan cuti karyawan, mengecek, menyetujui, atau menolak pengajuan cuti karyawan oleh Manager Divisi
Aktor.	Manager
Normal Flow:	<ol style="list-style-type: none"> Manager masuk ke tampilan menu awal masing-masing. Manager melihat daftar pengajuan cuti karyawan. Manager kemudian mengklik list pengajuan cuti. Kemudian akan tampil detail pengajuan. Kemudian Manager melihat dan mengecek data pengajuan cuti. Manager kemudian memutuskan apakah pengajuan akan disetujui atau ditolak. Jika disetujui maka Manager mengklik tombol “setuju” dan akan tersimpan ke <i>database</i>, jika ditolak maka klik tombol “Tolak”. Jika tombol “Tolak” diklik maka akan muncul <i>form</i> keterangan untuk memberikan alasan penolakan. Kemudian akan mengklik tombol “Simpan” jika keterangan alasan sudah diisi dan akan tersimpan ke <i>database</i>
Persyaratan	Manager telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

Sumber: Hasil Analisis Data (2019)

1. *Use Case Description* Melihat Laporan Cuti

Berikut adalah deskripsi skenario *use case diagram* melihat laporan cuti yang dapat dilihat pada Tabel V.15.

Tabel V.15 *Use Case Description* Melihat Laporan Cuti

Nama <i>Use Case</i>	Melihat Laporan Cuti
Deskripsi <i>Use Case</i> .	Menjelaskan proses untuk melihat laporan cuti karyawan berdasarkan bulan dan tahun
Aktor.	Direktur Keuangan & SDM
<i>Normal Flow:</i>	<ol style="list-style-type: none"> 1. Direktur keuangan & SDM masuk ke halaman awal direktur. 2. Direktur keuangan & SDM melihat daftar laporan cuti karyawan. 3. Direktur keuangan & SDM mengklik tombol <i>select</i> bulan dan tahun lalu akan tampil list data laporan cuti pada bulan yang dipilih. 4. Direktur Keuangan & SDM kemudian mengklik list laporan cuti. Kemudian akan tampil detail laporan cuti.
Persyaratan	Direktur Keuangan & SDM telah melakukan <i>login</i> terlebih dahulu dan sistem menampilkan halaman awal

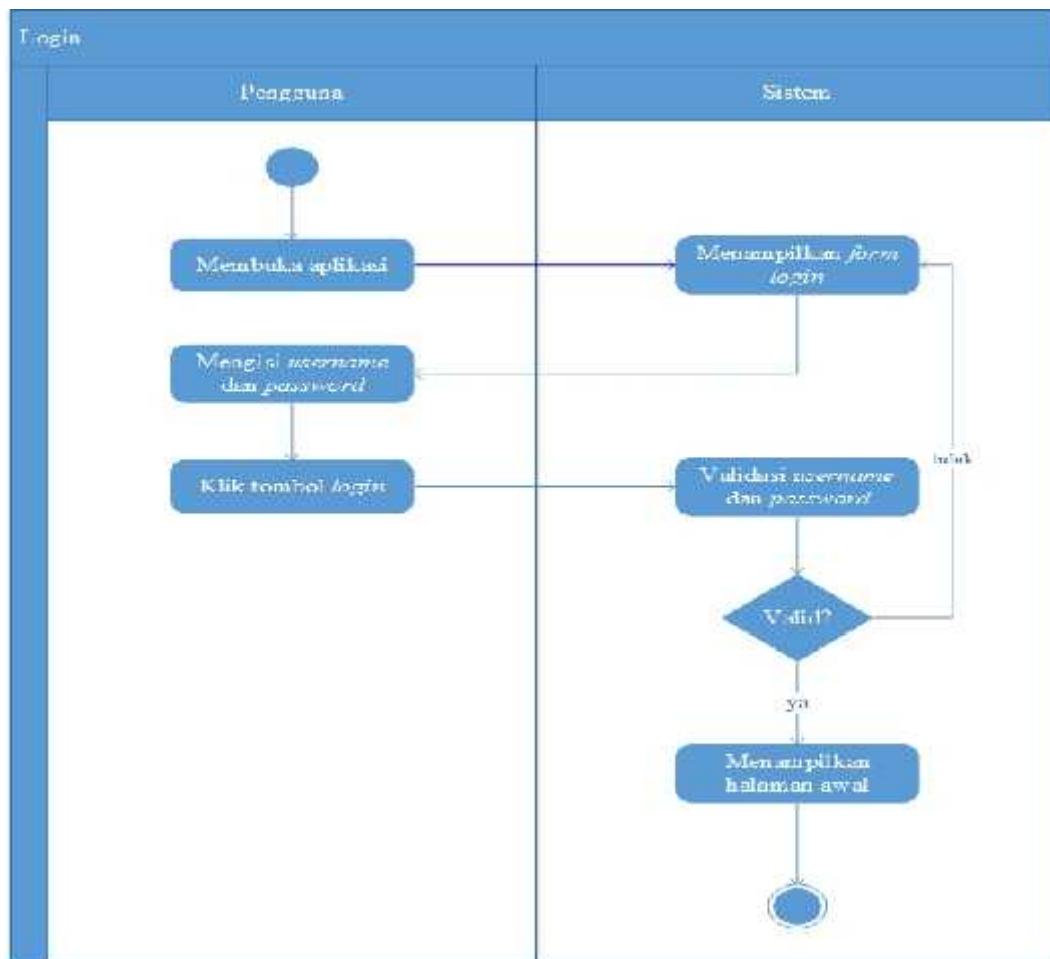
Sumber: Hasil Analisis Data (2019)

5.4.2 *Activity Diagram*

Activity diagram merupakan suatu diagram yang bertujuan untuk menggambarkan kegiatan-kegiatan yang ada di dalam suatu sistem. Agar memahami sistem yang akan dibuat, maka perlu dibuat *activity diagram* sistem informasi cuti karyawan, berikut ini merupakan *activity diagram* dari sistem informasi cuti karyawan:

1. *Activity Diagram Login*

Menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu Karyawan, Supervisor, Manager, Direktur Keuangan & SDM, dan Admin SDM untuk dapat masuk ke dalam sistem informasi cuti karyawan. *Activity diagram login* dapat dilihat pada gambar V.3.

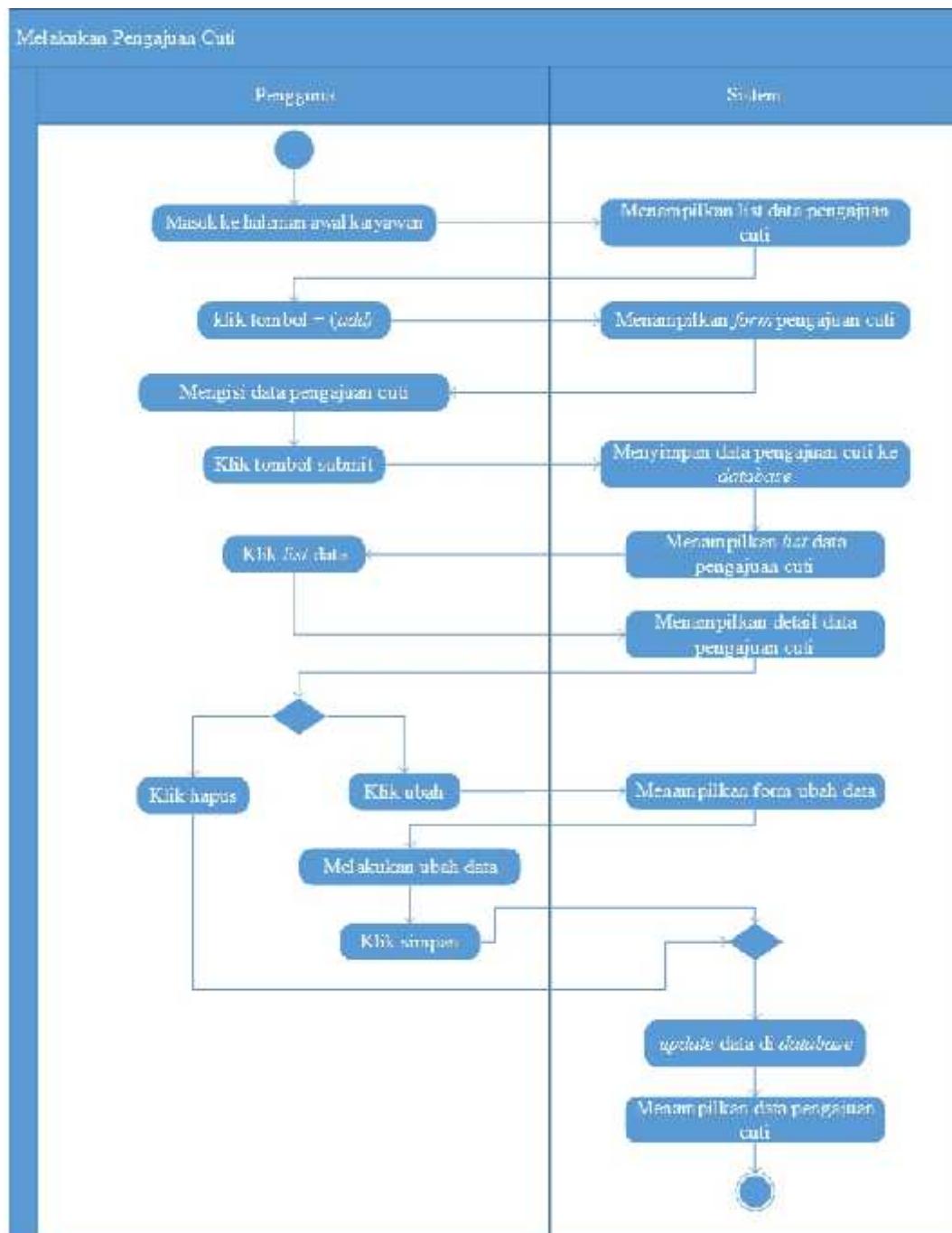


Gambar V.3 Activity Diagram Login

Sumber: Hasil Analisis (2019)

2. *Activity Diagram* Melakukan Pengajuan Cuti

Menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu Karyawan, untuk dapat mengajukan cuti. *Activity diagram* melakukan pengajuan cuti dapat dilihat pada gambar V.4.

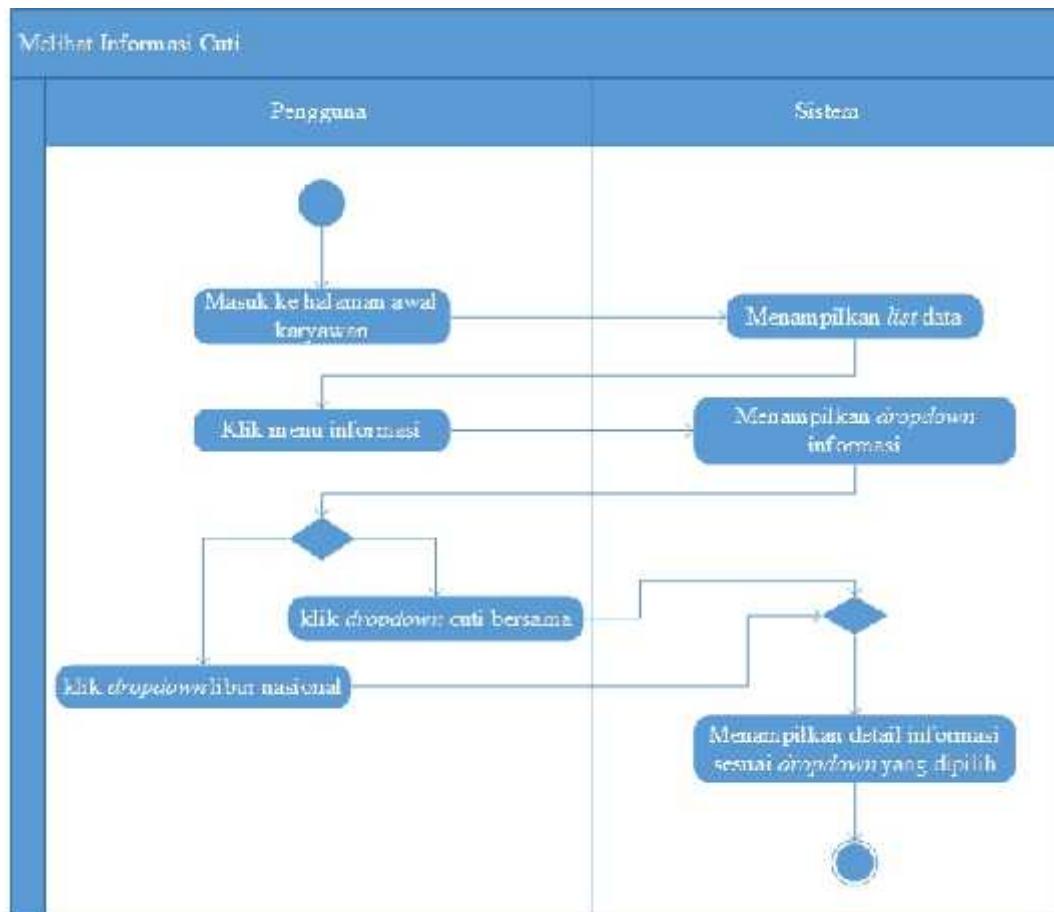


Gambar V.4 *Activity Diagram* Melakukan Pengajuan Cuti

Sumber: Hasil Analisis (2019)

3. *Activity Diagram* Melihat Informasi Cuti

Menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu Karyawan, untuk dapat melihat informasi yang berhubungan dengan cuti seperti informasi kegiatan, informasi cuti bersama dan informasi libur nasional. Melihat informasi dapat dilihat pada gambar V.5.

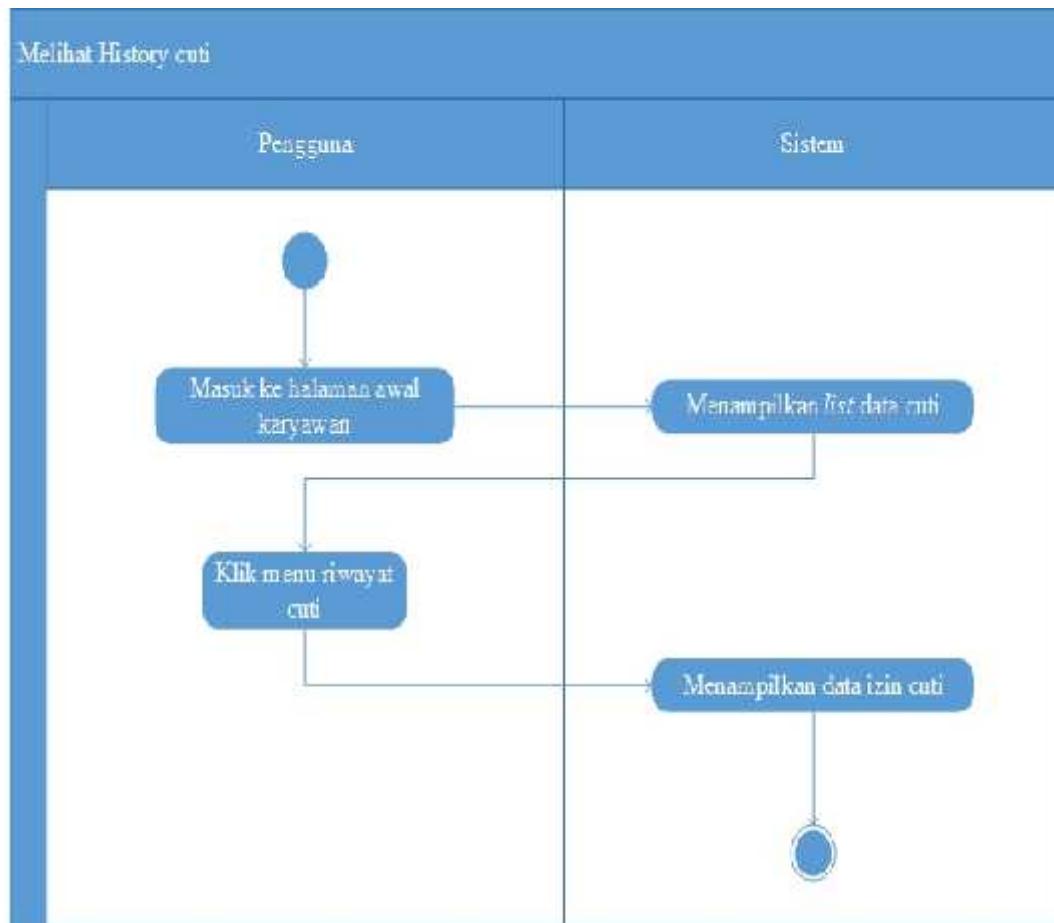


Gambar V.5 *Activity Diagram* Melihat Informasi Cuti

Sumber: Hasil Analisis (2019)

4. *Activity Diagram* Melihat *History* Cuti

Menjelaskan alur aktivitas yang dilakukan oleh *user*, yaitu Karyawan, untuk dapat melihat informasi tentang pengajuan cuti sudah disetujui (hingga supervisor atau manager) atau ditolak. Melihat *history* cuti dapat dilihat pada gambar V.6.

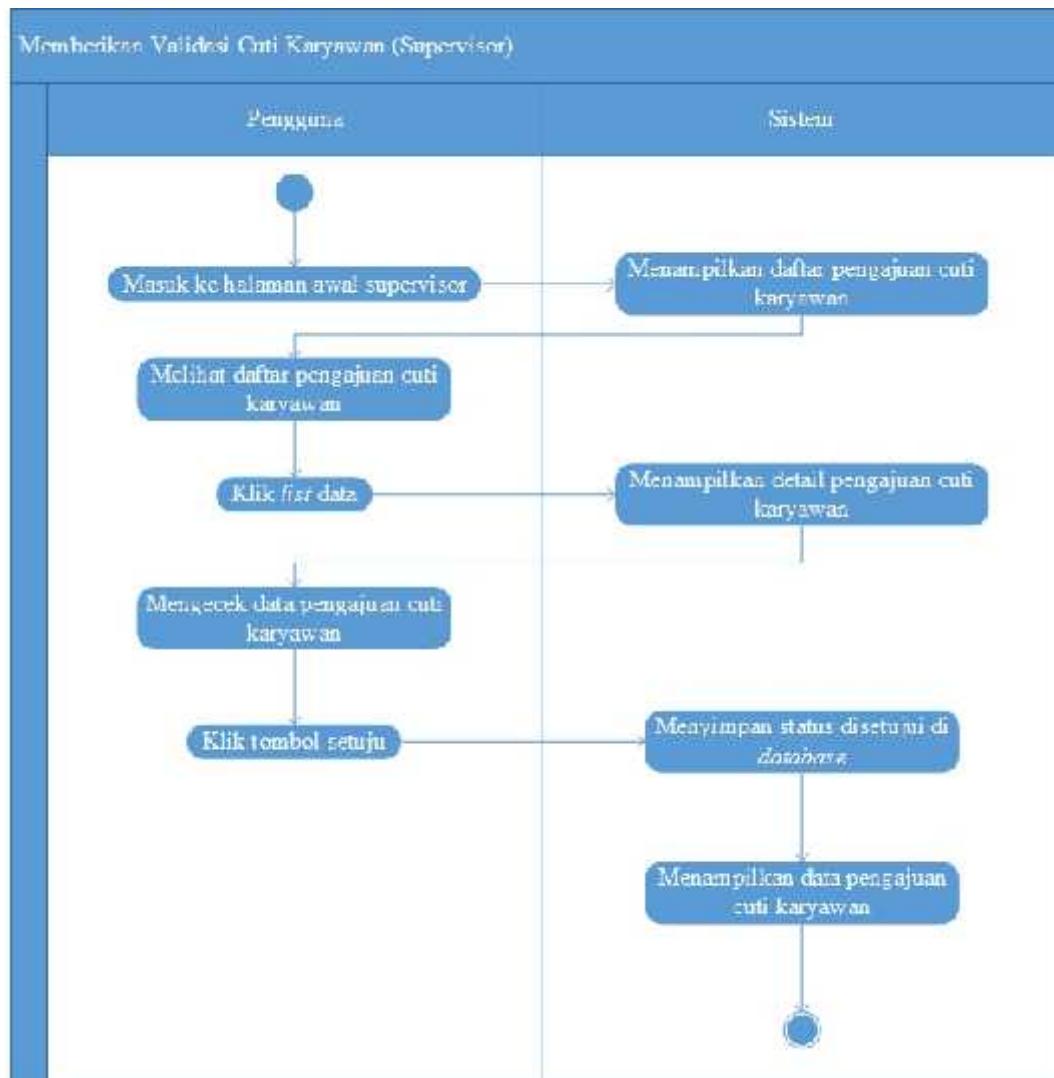


Gambar V.6 Activity Diagram Melihat History Cuti

Sumber: Hasil Analisis (2019)

5. *Activity Diagram Memberikan Validasi Cuti Karyawan*

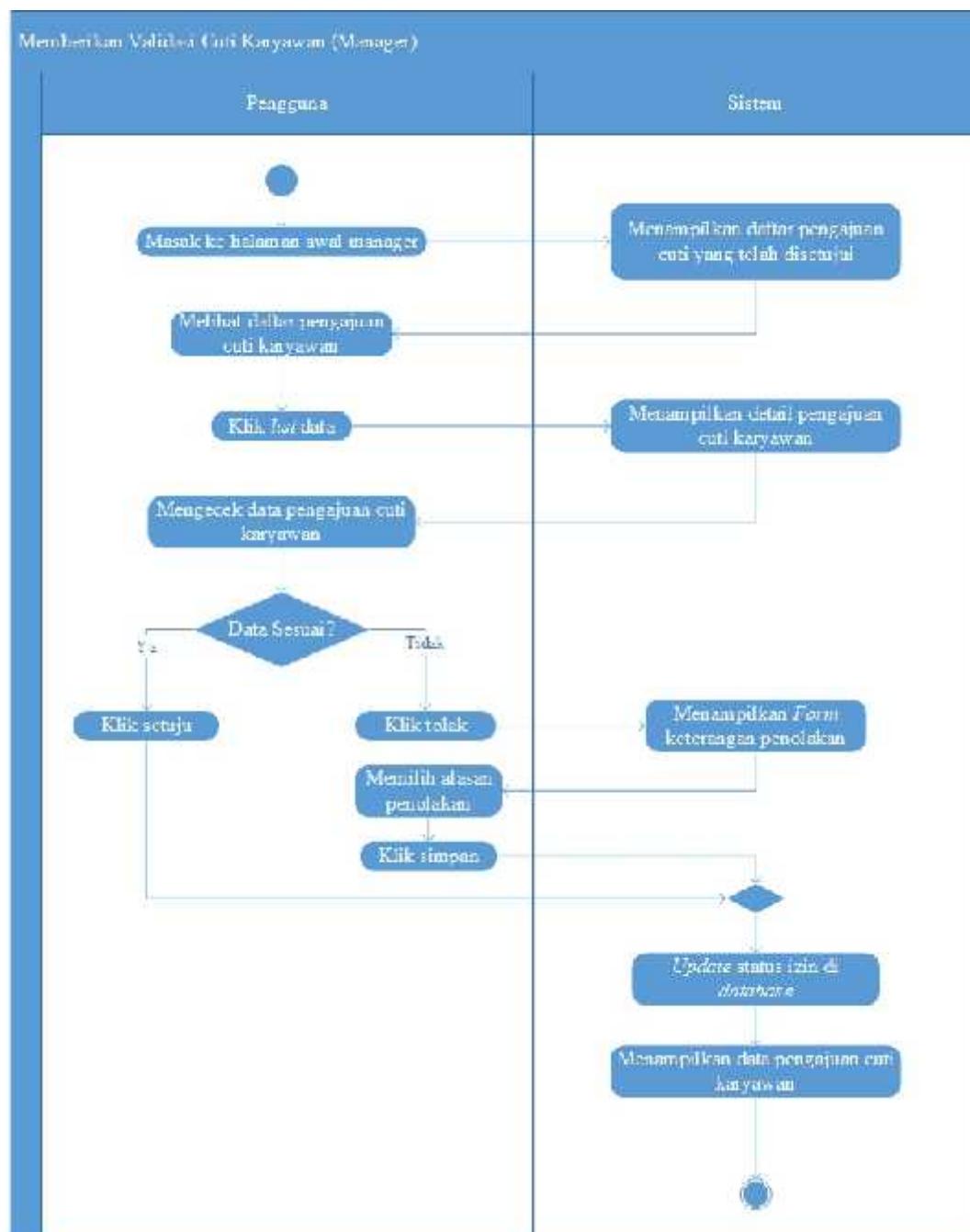
Menjelaskan alur aktivitas yang dilakukan oleh Supervisor, untuk dapat memberikan persetujuan kepada karyawan yang mengajukan cuti dalam hal ini sesuai dengan divisi masing-masing. Memberikan validasi cuti karyawan dapat dilihat pada gambar V.7.



Gambar V.7 Activity Diagram Memberikan Validasi Cuti Karyawan (1)
 Sumber: Hasil Analisis (2019)

6. *Activity Diagram* Memberikan Validasi Cuti Karyawan

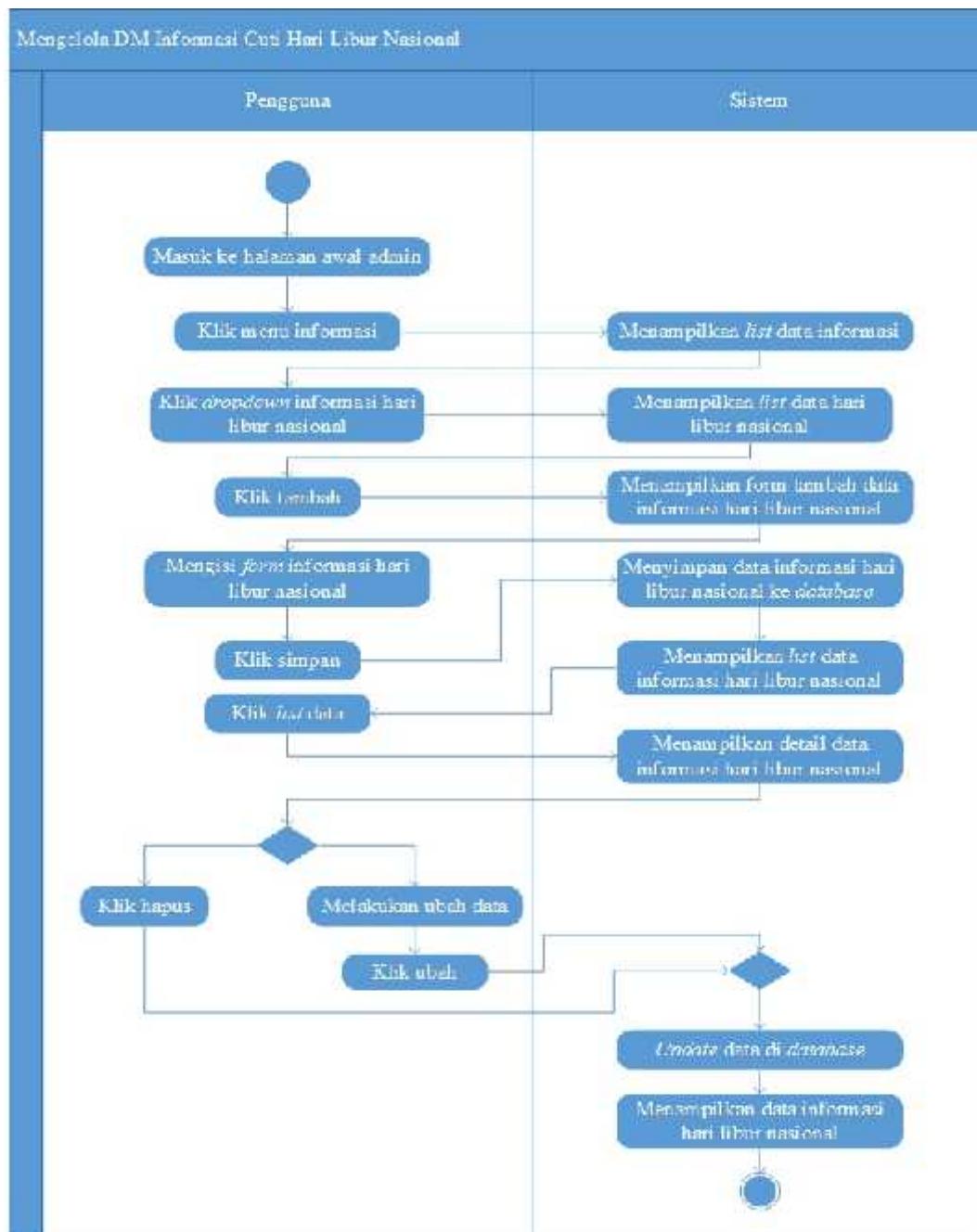
Menjelaskan alur aktivitas yang dilakukan oleh Manager, untuk dapat memberikan persetujuan atau penolakan beserta alasan kepada karyawan yang mengajukan cuti dalam hal ini sesuai dengan divisi masing-masing untuk manager. Memberikan validasi cuti karyawan dapat dilihat pada gambar V.8.



Gambar V.8 Activity Diagram Memberikan Validasi Cuti Karyawan (2)
Sumber: Hasil Analisis (2019)

7. *Activity Diagram* Mengelola Data Master Hari Libur Nasional

Menjelaskan alur aktivitas yang dilakukan oleh admin, dalam hal ini bagian SDM untuk dapat mengelola data master yang berhubungan dengan informasi hari libur nasional. Mengelola data master hari libur nasional dapat dilihat pada gambar V.9.

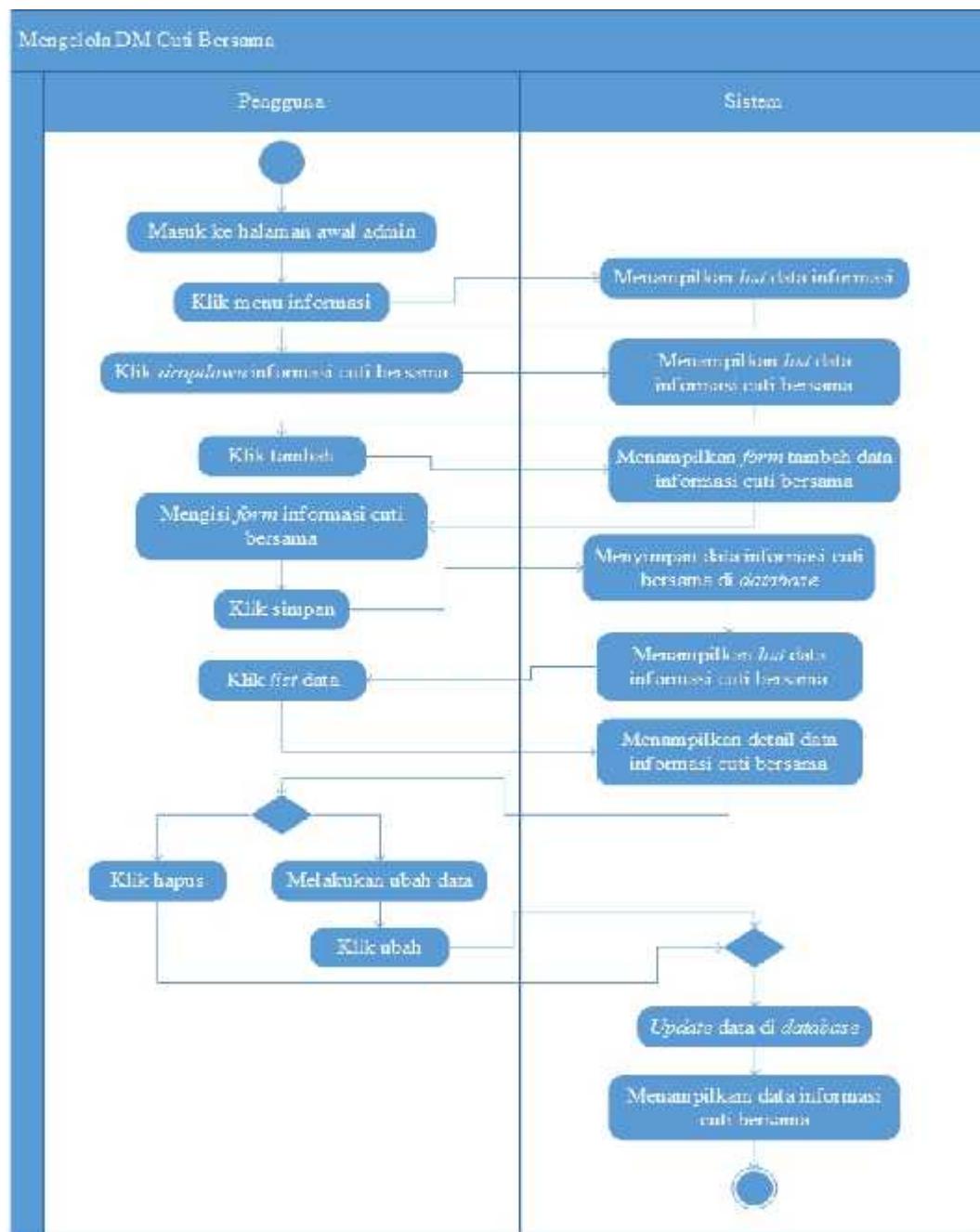


Gambar V.9 *Activity Diagram* Mengelola Data Master Hari Libur Nasional

Sumber: Hasil Analisis (2019)

8. *Activity Diagram* Mengelola Data Master Cuti Bersama

Menjelaskan alur aktivitas yang dilakukan oleh admin, dalam hal ini bagian SDM untuk dapat mengelola data master yang berhubungan dengan informasi cuti bersama. Mengelola data master cuti bersama dapat dilihat pada gambar V.10.

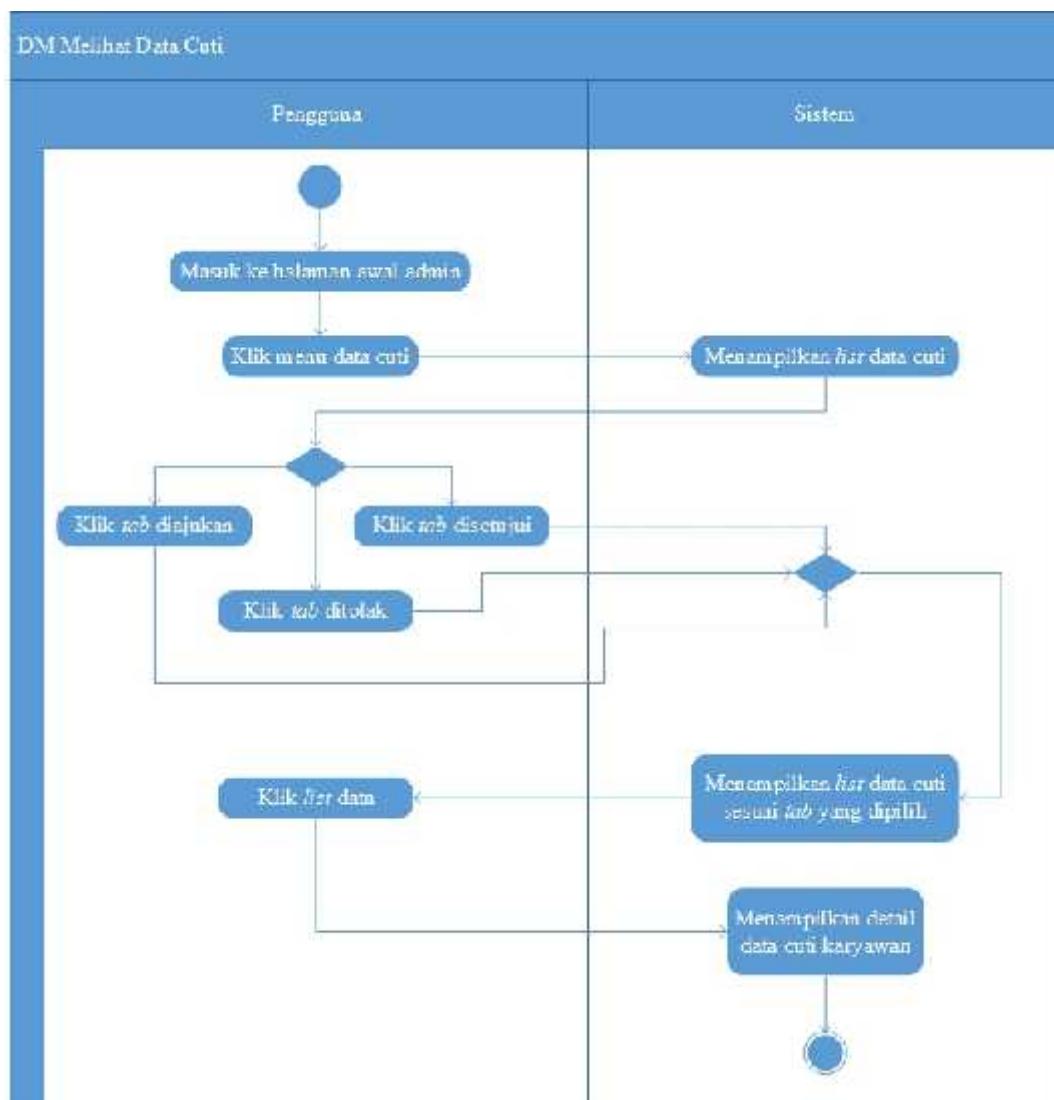


Gambar V.10 *Activity Diagram* Mengelola Data Master Cuti Bersama

Sumber: Hasil Analisis (2019)

9. *Activity Diagram* Mengelola Data Master Data Cuti Karyawan

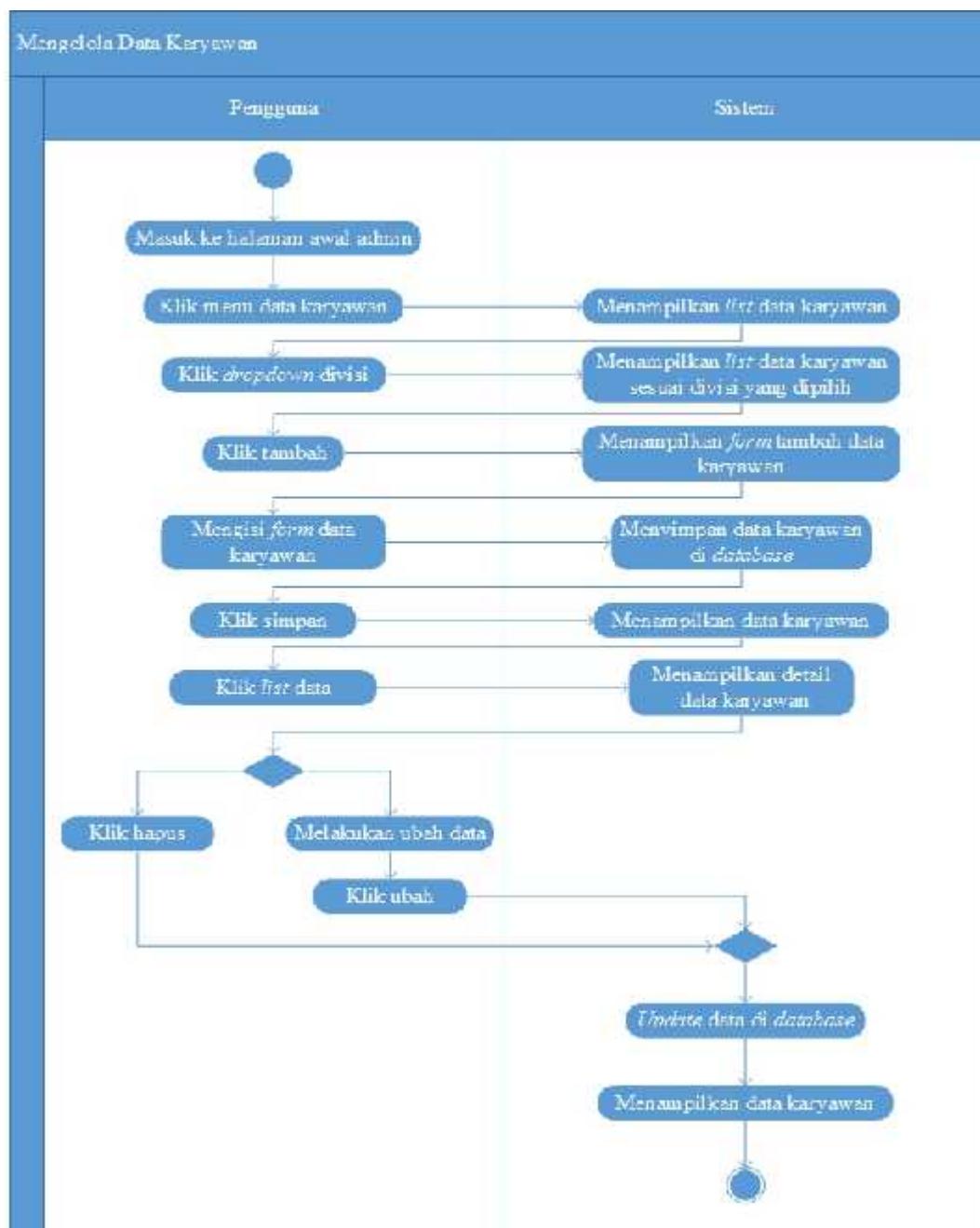
Menjelaskan alur aktivitas yang dilakukan oleh admin, dalam hal ini bagian SDM untuk dapat mengelola data master yang berhubungan dengan status cuti masih dalam proses diajukan atau sudah disetujui atau ditolak berikut dengan alasan penolakan. Mengelola data master cuti karyawan dapat dilihat pada gambar V.11.



Gambar V.11 *Activity Diagram* Mengelola Data Master Data Cuti Karyawan
Sumber: Hasil Analisis (2019)

10. *Activity Diagram Mengelola Data Karyawan*

Menjelaskan alur aktivitas yang dilakukan oleh admin, dalam hal ini bagian SDM dapat mengelola data karyawan berupa tambah data, hapus dan ubah. Mengelola data karyawan dapat dilihat pada gambar V.12.

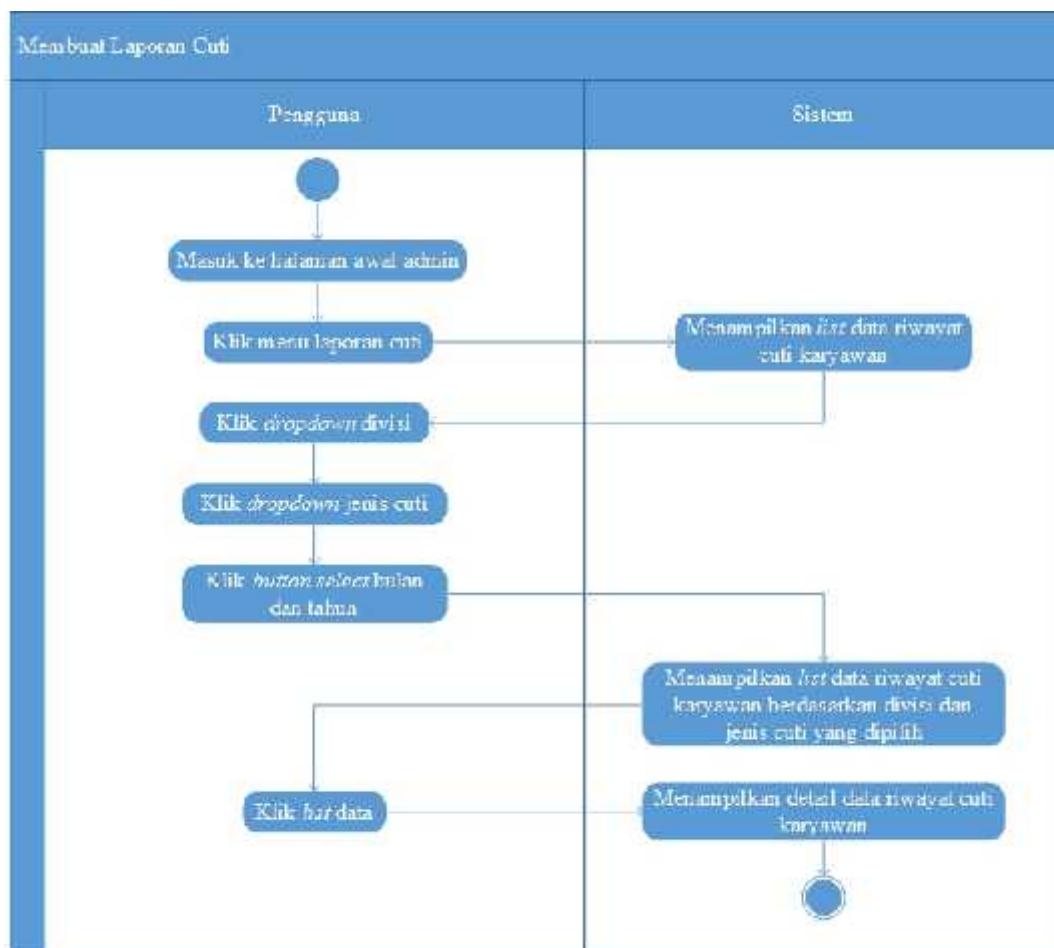


Gambar V.12 *Activity Diagram Mengelola Data Karyawan*

Sumber: Hasil Analisis (2019)

11. *Activity Diagram* Membuat laporan Cuti

Menjelaskan alur aktivitas yang dilakukan oleh admin, dalam hal ini bagian SDM untuk dapat membuat dan menampilkan laporan cuti yaitu berupa hasil rekap riwayat cuti perbulannya. Membuat laporan cuti dapat dilihat pada gambar V.13.

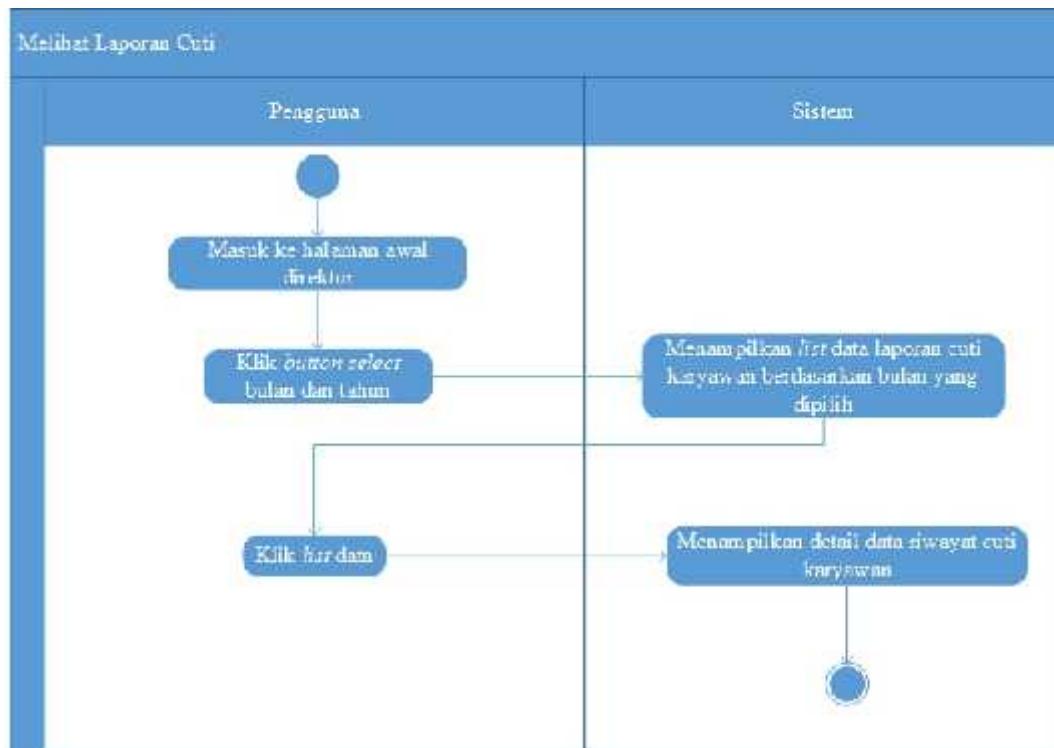


Gambar V.13 *Activity Diagram* Membuat Laporan Cuti

Sumber: Hasil Analisis (2019)

12. *Activity Diagram* Melihat laporan Cuti

Menjelaskan alur aktivitas yang dilakukan oleh direktur, dalam hal ini direktur keuangan & SDM dapat melihat laporan cuti yaitu berupa hasil rekap riwayat cuti perbulannya. Melihat laporan cuti dapat dilihat pada gambar V.14.



Gambar V.14 *Activity Diagram Melihat Laporan Cuti*

Sumber: Hasil Analisis (2019)

5.4.3 Sequence Diagram

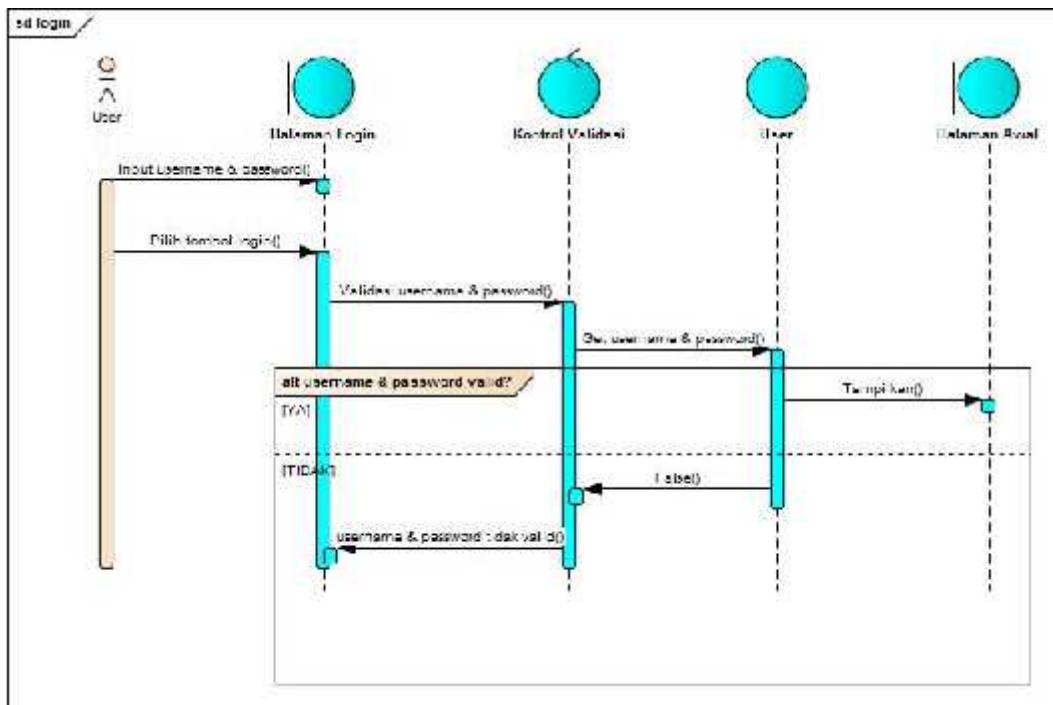
Sequence diagram merupakan diagram yang menggambarkan atau memodelkan interaksi yang terjadi pada suatu objek *use case diagram* ketika melakukan suatu proses tertentu. Urutan proses ketika melakukan suatu proses tertentu dapat diketahui dengan melihat gambaran pada *sequence diagram*.

Berikut ini merupakan *sequence diagram* usulan pada sistem informasi cuti karyawan:

1. *Sequence Diagram Login*

Sequence diagram login menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses *login*.

Proses ini dilakukan oleh karyawan, supervisor, manager, Direktur dan admin yang mempunyai hak akses sebelum masuk ke halaman awal pengguna sistem. Adapun *sequence diagram* login dapat dilihat pada Gambar V.15.

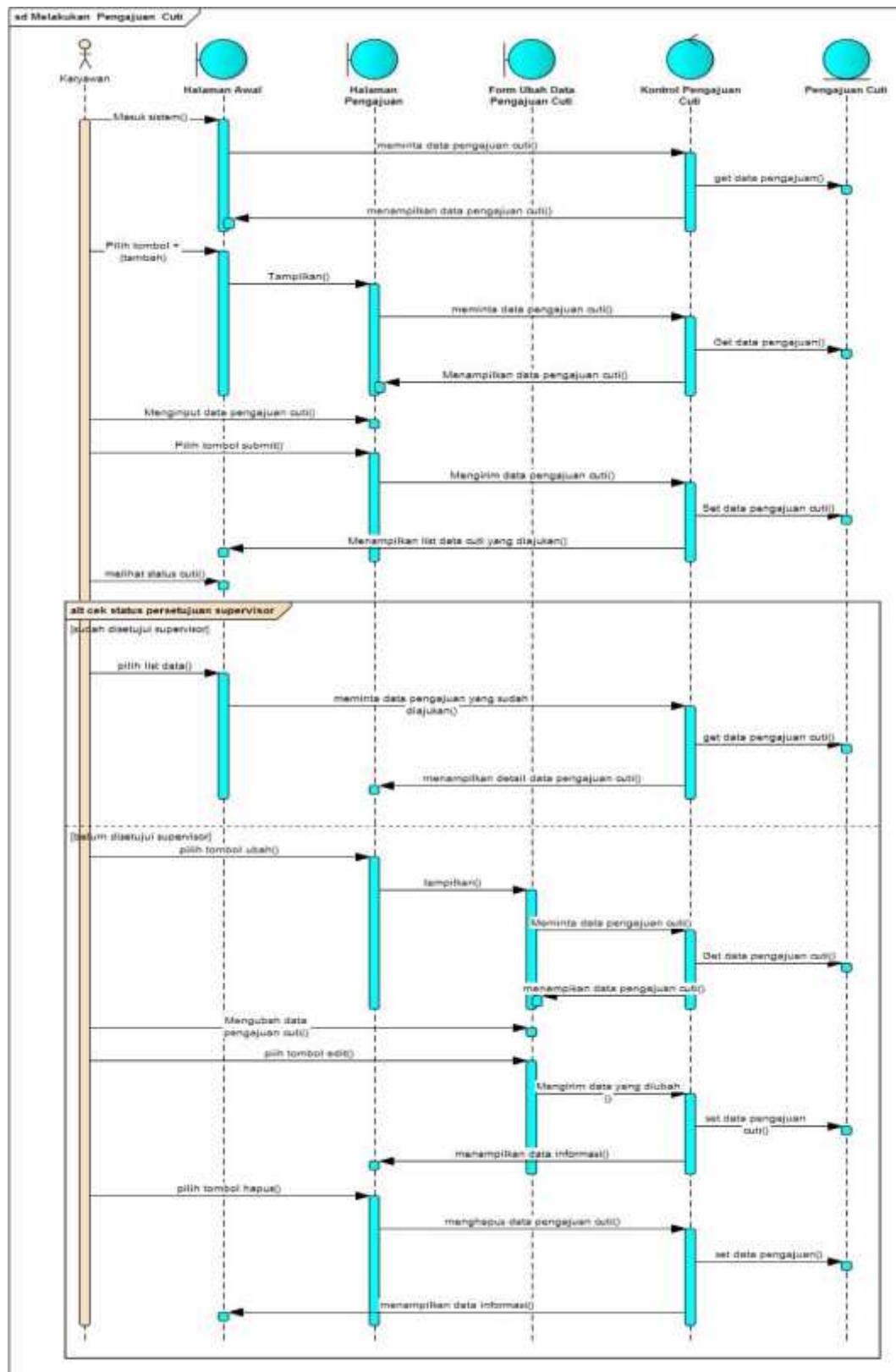


Gambar V.15 Sequence Diagram Login

Sumber: Hasil Analisis (2019)

2. Sequence Diagram Melakukan Pengajuan Cuti

Sequence diagram melakukan pengajuan cuti menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses mengelola pengajuan cuti berupa tambah, ubah, hapus. Adapun *sequence diagram* melakukan pengajuan cuti dapat dilihat pada Gambar V.16.

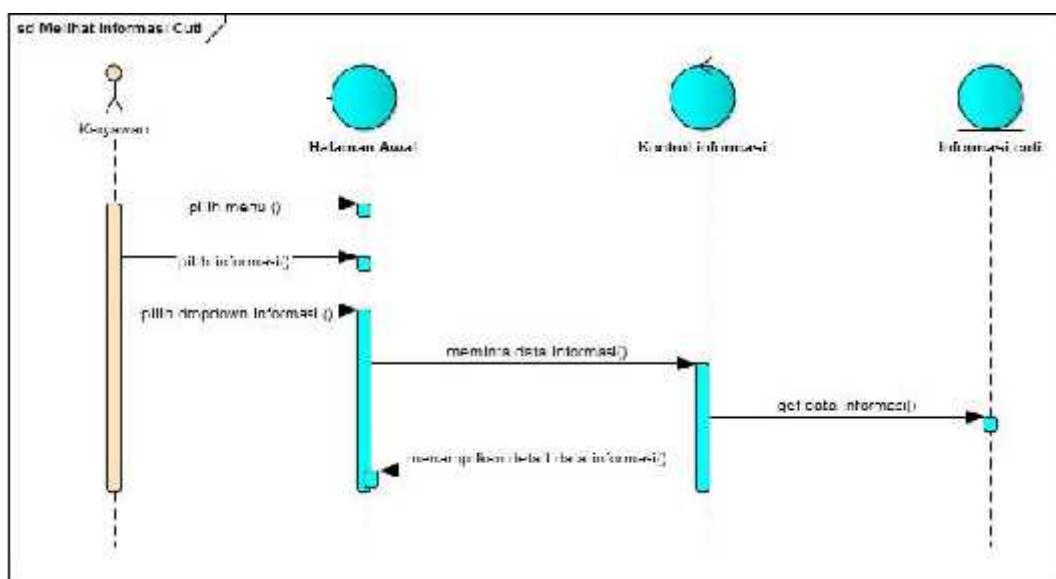


Gambar V.16 Sequence Diagram Melakukan Pengajuan Cuti

Sumber: Hasil Analisis (2019)

3. *Sequence Diagram* Melihat Informasi Cuti

Sequence diagram melihat informasi cuti menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses melihat informasi cuti yang berkaitan dengan informasi cuti bersama, libur nasional atau informasi seputar status pengajuan cuti. Adapun *sequence diagram* melihat informasi cuti dapat dilihat pada Gambar V.17.

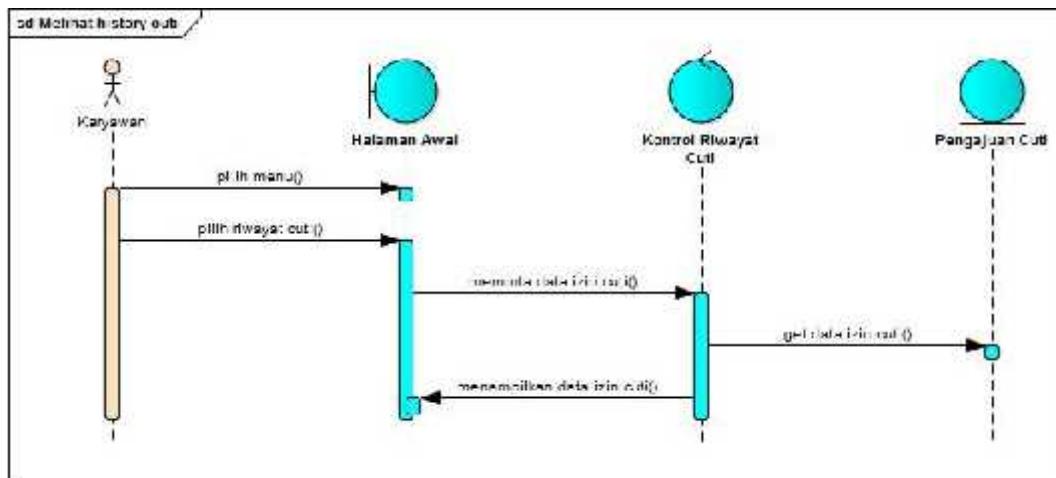


Gambar V.17 *Sequence Diagram* Melihat Informasi Cuti

Sumber: Hasil Analisis (2019)

4. *Sequence Diagram* Melihat History Cuti

Sequence diagram melihat *history* cuti menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses melihat *history* cuti yang berkaitan dengan riwayat cuti dengan keterangan sudah izinkan cuti atau ditolak beserta keterangan penolakan. Adapun *sequence diagram* melihat *history* cuti dapat dilihat pada Gambar V.18.

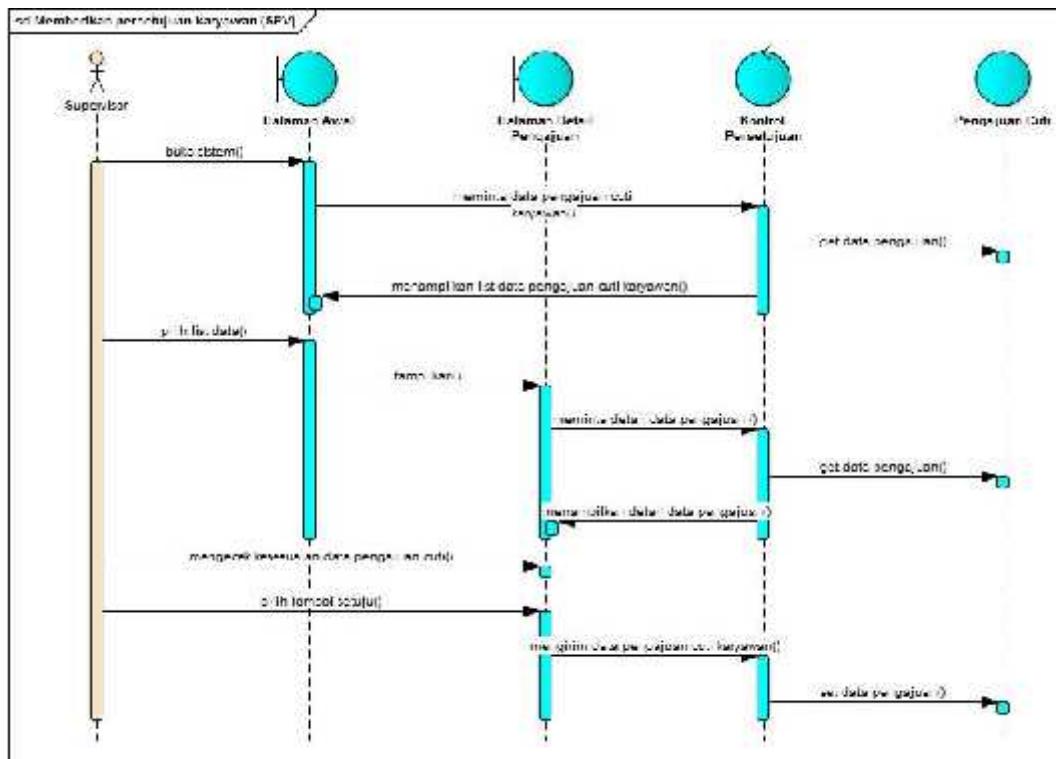


Gambar V.18 *Sequence Diagram Melihat History Cuti*

Sumber: Hasil Analisis (2019)

5. *Sequence Diagram* Memberikan Validasi Cuti Karyawan

Sequence diagram memberikan validasi cuti karyawan menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses mengelola persetujuan cuti karyawan yang dilakukan oleh supervisor dengan memberikan persetujuan. Adapun *Sequence diagram* Memberikan Validasi Cuti Karyawan dapat dilihat pada Gambar V.19.

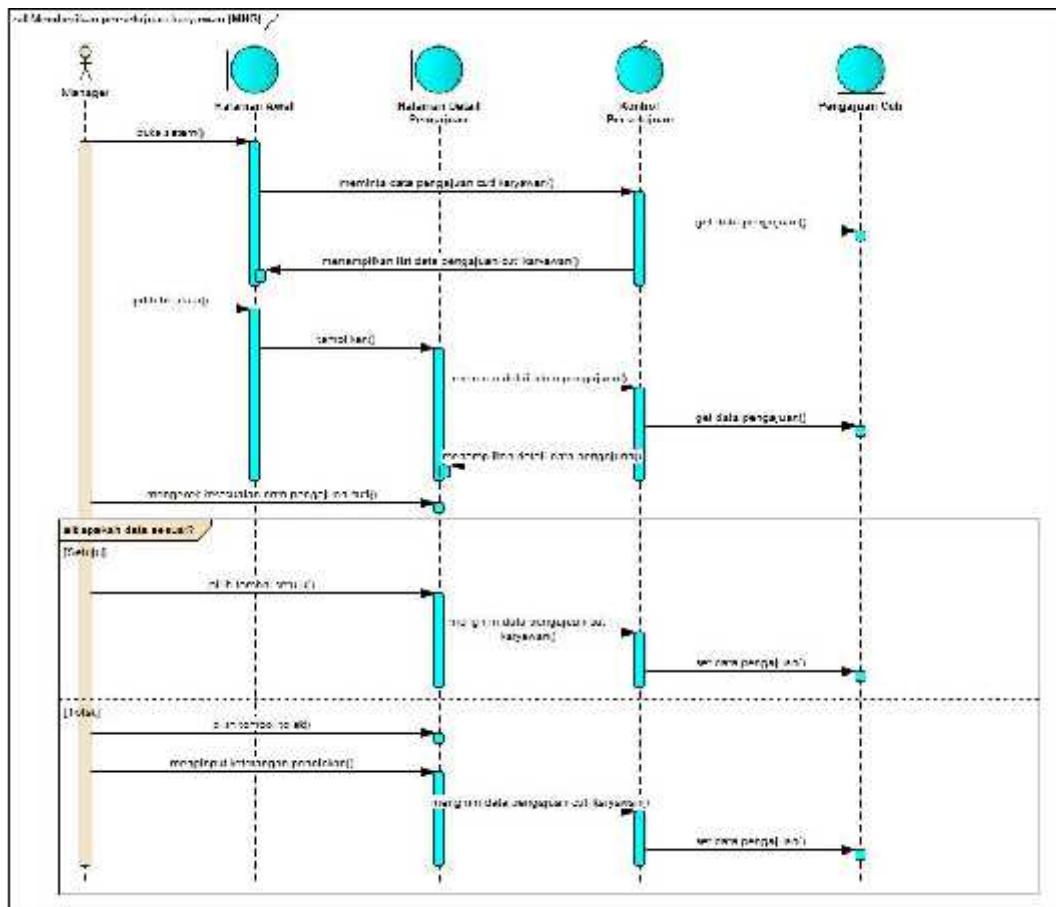


Gambar V.19 Sequence Diagram Memberikan Validasi Cuti Karyawan (1)

Sumber: Hasil Analisis (2019)

6. *Sequence Diagram* Memberikan Validasi Cuti Karyawan

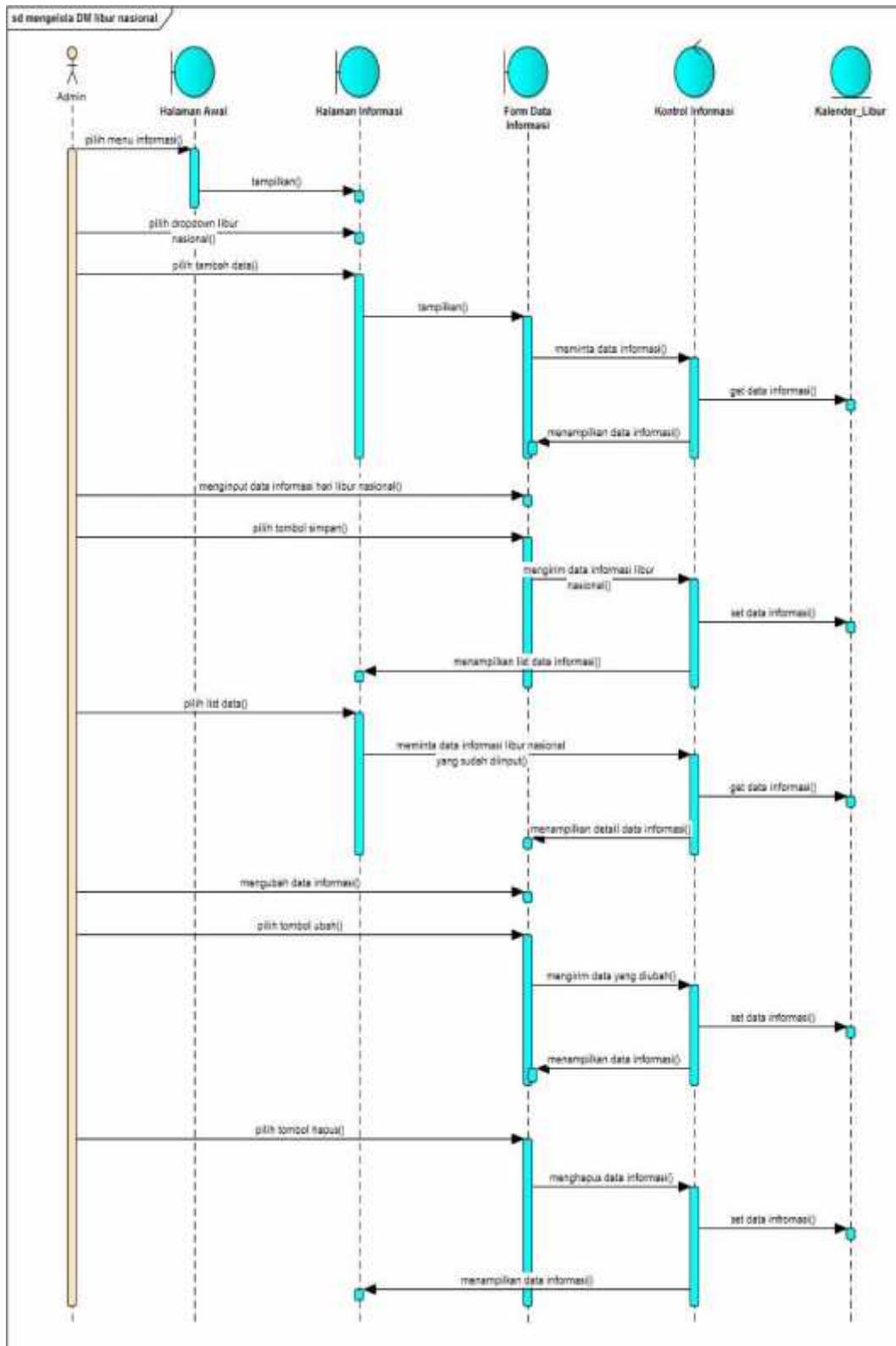
Sequence diagram memberikan validasi cuti karyawan menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses memberikan persetujuan cuti karyawan yang dilakukan oleh manager dengan memberikan persetujuan atau penolakan. Adapun *Sequence diagram* Memberikan Validasi Cuti Karyawan dapat dilihat pada Gambar V.20.



Gambar V.20 Sequence Diagram Memberikan Validasi Cuti Karyawan (2)
Sumber: Hasil Analisis (2019)

7. *Sequence Diagram* Mengelola Data Master Hari Libur Nasional

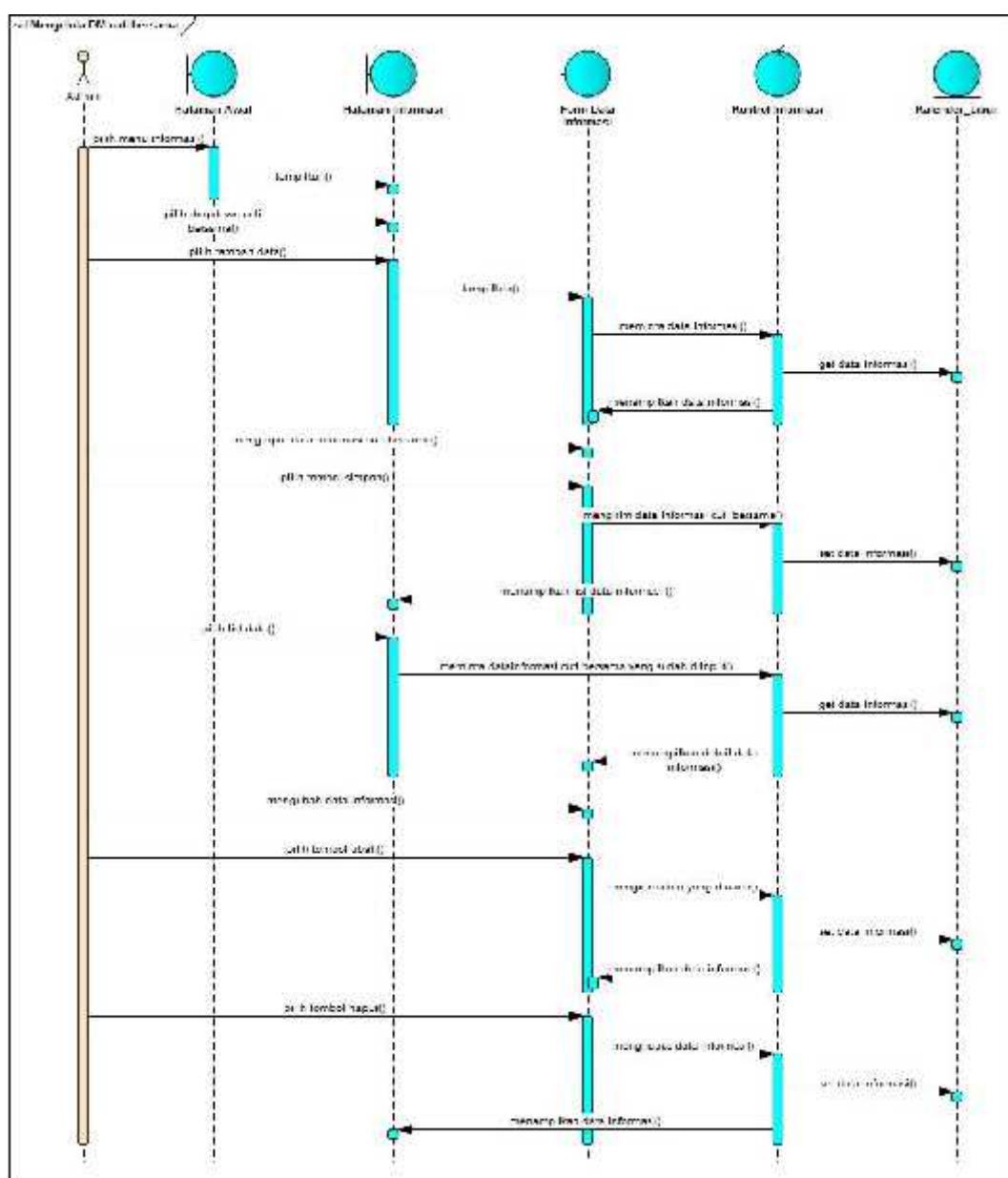
Sequence diagram mengelola data master hari libur nasional menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses mengelola informasi hari libur nasional berupa tambah, ubah, hapus. Adapun *sequence diagram* mengelola data master hari libur nasional dapat dilihat pada Gambar V.21.



Gambar V.21 Sequence Diagram Mengelola Data Master Hari Libur Nasional
 Sumber: Hasil Analisis (2019)

8. *Sequence Diagram* Mengelola Data Master Cuti Bersama

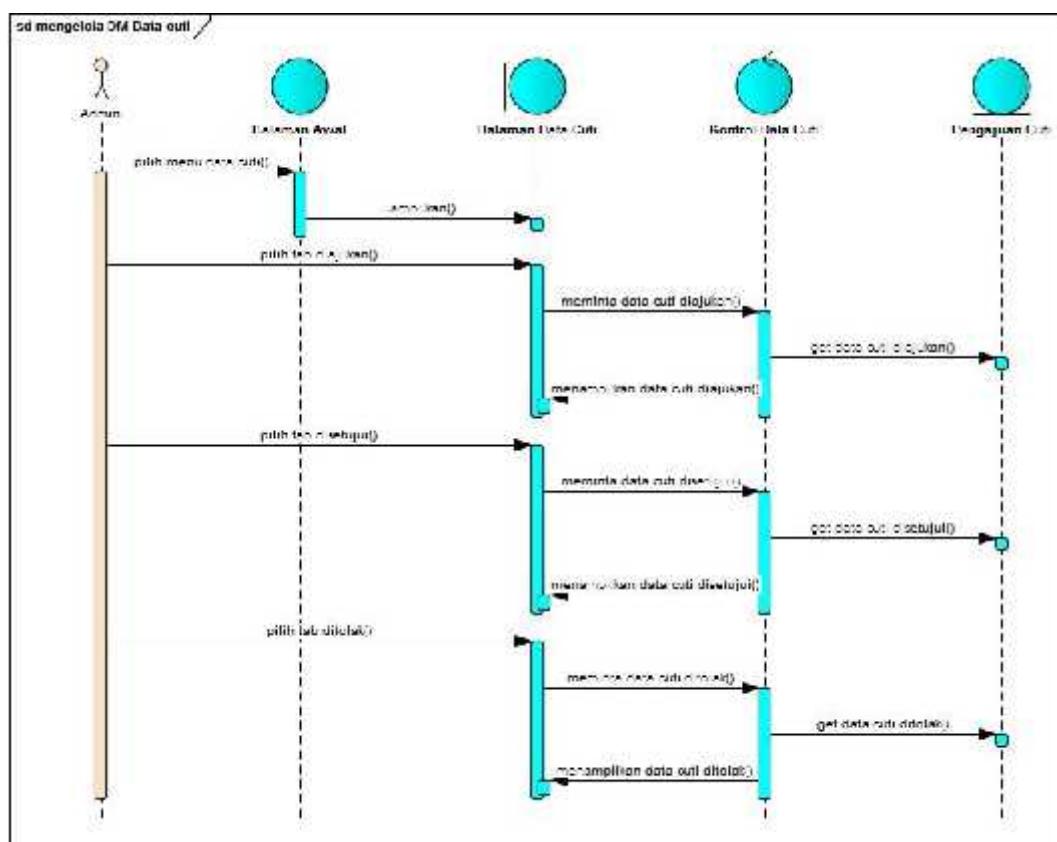
Sequence diagram mengelola data master hari cuti bersama menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses mengelola informasi hari cuti bersama berupa tambah, ubah, hapus. Adapun *sequence diagram* mengelola data master cuti bersama dapat dilihat pada Gambar V.22.



Gambar V.22 *Sequence Diagram* Mengelola Data Master Cuti Bersama
Sumber: Hasil Analisis (2019)

9. Sequence Diagram Mengelola Data Master Data Cuti Karyawan

Sequence diagram mengelola data master hari cuti bersama menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses melihat data cuti karyawan yang sedang mengajukan, yang sudah disetujui atau ditolak berikut dengan keterangan penolakan. Adapun sequence diagram mengelola data master data cuti karyawan dapat dilihat pada Gambar V.23.

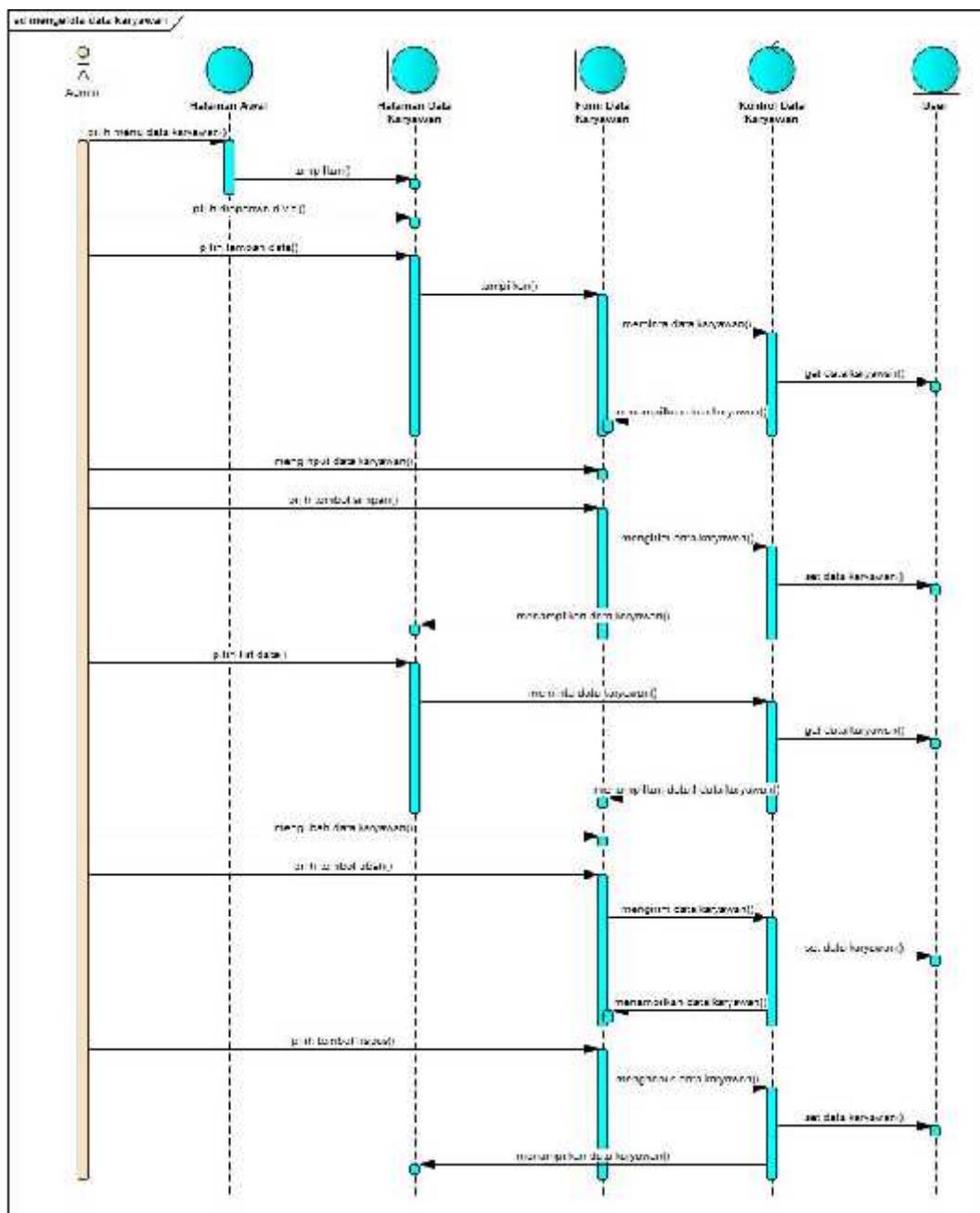


Gambar V.23 Sequence Diagram Mengelola Data Master Data Cuti Karyawan

Sumber: Hasil Analisis (2019)

10. Sequence Diagram Mengelola Data Karyawan

Sequence diagram mengelola data karyawan menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses mengelola data karyawan berupa tambah, hapus dan ubah. Adapun *Sequence diagram* mengelola data karyawan dapat dilihat pada Gambar V.24.

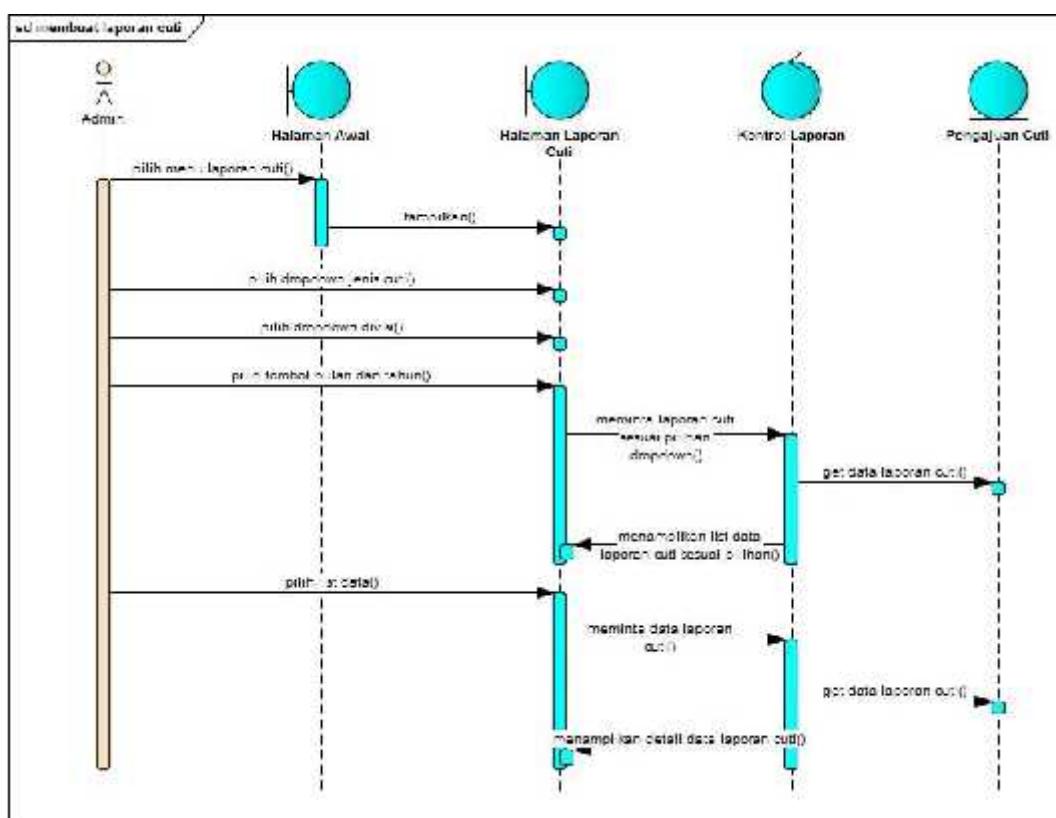


Gambar V.24 *Sequence Diagram* Mengelola Data Karyawan

Sumber: Hasil Analisis (2019)

11. Sequence Diagram Membuat Laporan Cuti

Sequence diagram membuat laporan cuti menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses menampilkan laporan cuti Karyawan berdasarkan divisi dan jenis cuti. Adapun Sequence diagram membuat laporan cuti dapat dilihat pada Gambar V.25.

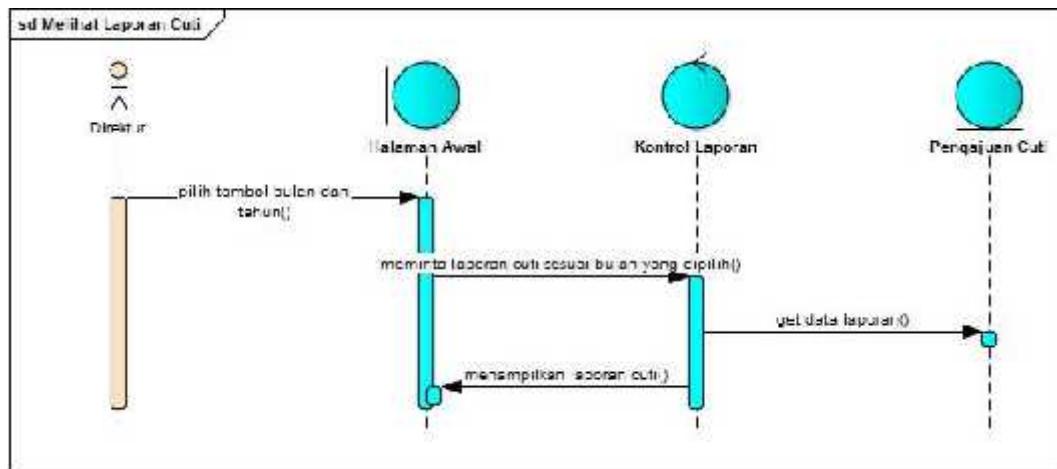


Gambar V.25 Sequence Diagram Membuat Laporan Cuti

Sumber: Hasil Analisis (2019)

12. Sequence Diagram Melihat Laporan Cuti

Sequence diagram melihat laporan cuti menjelaskan sebuah interaksi antara pengguna dengan sistem ketika melakukan proses menampilkan laporan cuti Karyawan berdasarkan divisi dan jenis cuti. Adapun Sequence diagram melihat laporan cuti dapat dilihat pada Gambar V.26.

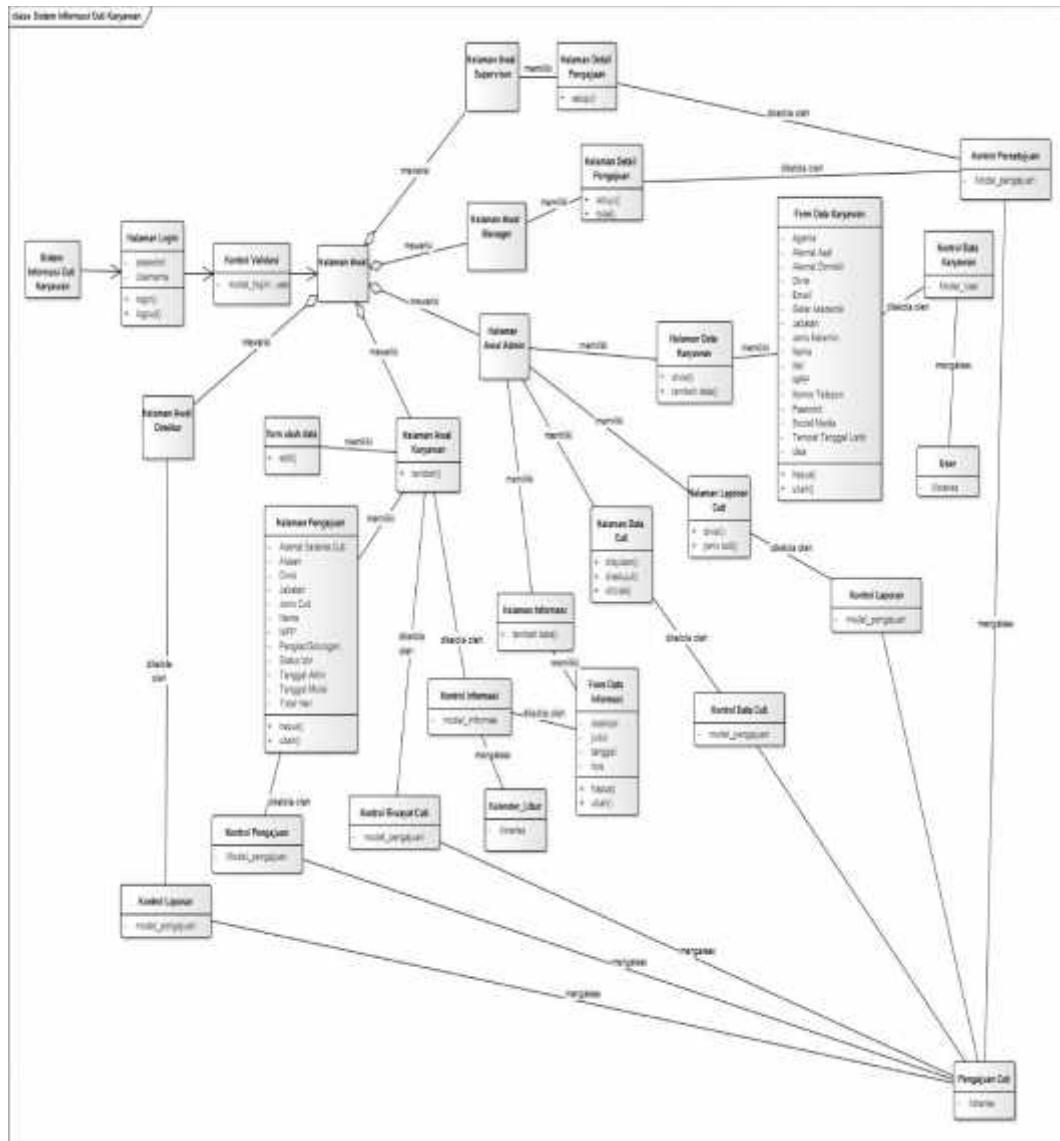


Gambar V.26 Sequence Diagram Melihat Laporan Cuti

Sumber: Hasil Analisis (2019)

5.4.4 Class Diagram

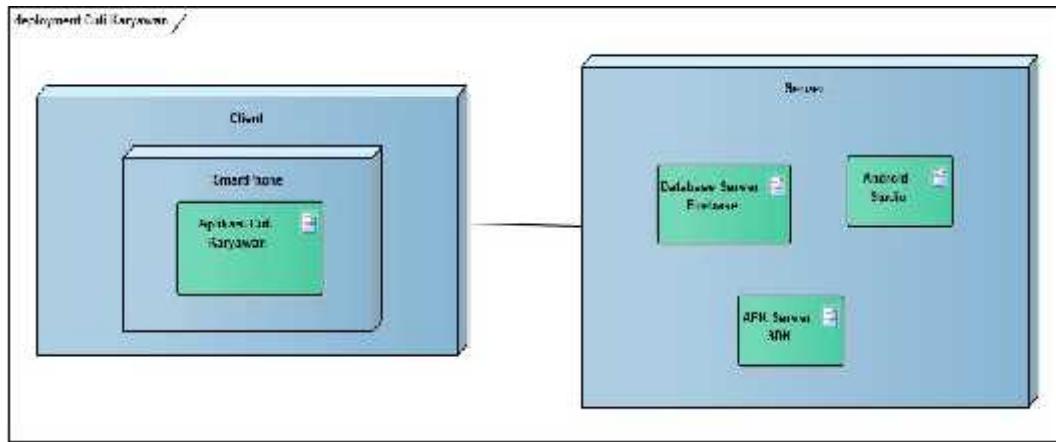
Class diagram merupakan diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat, *class diagram* memperlihatkan hubungan antar kelas dan penjelasan *detail* setiap kelas di dalam model desain dari suatu sistem. *Class diagram* sistem informasi cuti karyawan yang diusulkan dapat dilihat pada Gambar V.27.



Gambar V.27 *Class Diagram* Sistem Informasi Cuti Karyawan
Sumber: Hasil Analisis (2019)

5.4.5 Deployment Diagram

Deployment Diagram ini menjelaskan model fisik dari *hardware* serta integrasi dan distribusi *software* pada sistem informasi cuti karyawan usulan. Berikut merupakan gambar *Deployment Diagram* sistem informasi cuti karyawan usulan, dapat dilihat pada Gambar V.28.



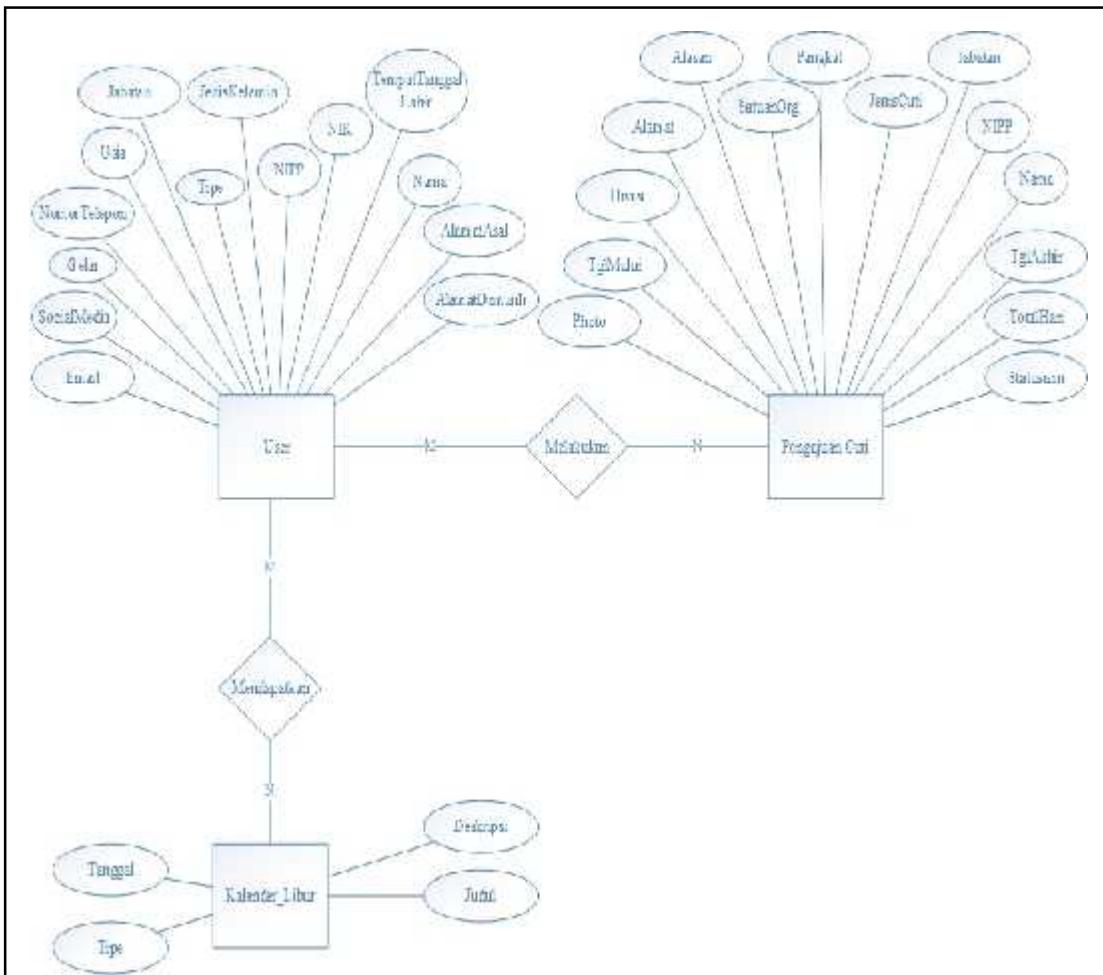
Gambar V.28 *Deployment Diagram* Sistem Informasi Cuti Karyawan
Sumber: Hasil Analisis (2019)

5.5 Pemodelan Data Sistem Informasi Cuti Karyawan Usulan

Pemodelan data pada sistem informasi cuti karyawan usulan ini menggunakan 3 pemodelan yaitu *Entity Relationship Diagram* (ERD) yang bertujuan untuk menjelaskan hubungan antar data dalam basis data, kemudian menggunakan *Physical Diagram Model* yang bertujuan untuk menggambarkan seluruh muatan informasi yang terkandung oleh suatu basis data. Kemudian yang terakhir adalah menggunakan kamus data yang berisi daftar data yang terdapat didalam sebuah sistem dengan maksud untuk mendefinisikan aliran data di dalam sebuah sistem dengan lengkap dan sesuai dengan sistem.

5.5.1 *Entity Relationship Diagram* (ERD)

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol. ERD sistem informasi cuti karyawan yang diusulkan dapat dilihat pada Gambar V.29.



Gambar V.29 *Entity Relationship Diagram* Sistem Informasi Cuti Karyawan

Sumber: Hasil Analisis (2019)

5.5.2 Transformasi ERD ke Diagram

PDM merupakan konsep yang menerangkan detail dari bagaimana data disimpan didalam basis data yang sudah siap diimplementasikan kedalam DBMS sehingga nama tabel juga sudah merupakan nama asli tabel yang akan diimplementasikan kedalam DBMS. Berikut merupakan ini merupakan transformasi diagram pada sistem informasi cuti karyawan usulan, dapat dilihat pada Gambar V.30.



Gambar V.30 Transformasi Diagram ke ERD
 Sumber: Hasil Analisis (2019)

5.5.3 Kamus Data

Kamus data merupakan daftar data yang terdapat didalam sebuah sistem dengan maksud untuk mendefinisikan aliran data di dalam sebuah sistem dengan lengkap dan sesuai dengan sistem, sehingga user dan analisis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen data *store*. Pada tahap perancangan sistem, kamus data digunakan untuk merancang *input*, laporan dan basis data. Berikut adalah kamus data sistem informasi cuti karyawan.

1. Spesifikasi Objek *User*

Nama Objek : User

Akronim : User

Fungsi : Untuk menyimpan data karyawan berisi jabatan, divisi dan memberikan hak akses

Tipe : Data Master

Tabel V.16 Spesifikasi Objek *User*

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Nama	Nama	Text String	44 Byte	<i>Not Null</i>
2.	Tempat Tanggal Lahir	Tempattanggallahir	Text String	109 Byte	<i>Not Null</i>
3.	NIPP	NIPP	Text String	27 Byte	<i>Not Null</i>
4.	NIK	NIK	Text String	27 Byte	<i>Not Null</i>
5.	Jenis Kelamin	Jenis_Kelamin	Text String	12 Byte	<i>Not Null</i>
6.	Tipe	usertipe	Text String	14 Byte	<i>Not Null</i>
7.	Divisi	userdivisi	Text String	32 Byte	<i>Not Null</i>
8.	Nomor Telepon	NomorTelepon	Text String	27 Byte	<i>Not Null</i>
9.	Gelar Akademik	Gelar	Text String	8 Byte	<i>Not Null</i>
10.	Usia	Usia	Text String	5 Byte	<i>Not Null</i>
11.	Agama	Agama	Text String	12 Byte	<i>Not Null</i>
12.	Alamat Asal	AlamatAsal	Text String	109 Byte	<i>Not Null</i>

Tabel V.16 Spesifikasi Objek *User* (Lanjutan)

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
13.	Alamat Domisili	Alamat Domisili	Text String	109 Byte	<i>Not Null</i>
14.	Email	Email	Text String	32 Byte	<i>Not Null</i>
15.	Social Media	Socialmedia	Text String	32 Byte	<i>Not Null</i>
16.	Password	password	Text String	8 Byte	<i>Not Null</i>

(Sumber : Hasil Analisis, 2019)

2. Spesifikasi Objek Pengajuan Cuti

Nama Objek : Pengajuan Cuti
 Akronim : PengajuanCuti
 Fungsi : Untuk menyimpan data cuti karyawan
 Tipe : Data Master

Tabel V.17 Spesifikasi Objek Pengajuan Cuti

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Nama	Nama	Text String	44 Byte	<i>Not Null</i>
2.	NIPP	NIPP	Text String	27 Byte	<i>Not Null</i>
3.	Pangkat/Golongan	Pangkat	Text String	5 Byte	<i>Not Null</i>
4.	Satuan Organisasi	SatuanOrg	Text String	32 Byte	<i>Not Null</i>
5.	Jabatan	usertipe	Text String	14 Byte	<i>Not Null</i>
6.	Divisi	userdivisi	Text String	32 Byte	<i>Not Null</i>
7.	Jenis Cuti	JenisCuti	Text String	24 Byte	<i>Not Null</i>
8.	Alamat Selama Cuti	Alamat	Text String	109 Byte	<i>Not Null</i>
9.	Alasan Cuti	Alasan	Text String	44 Byte	<i>Not Null</i>
10.	Tanggal Mulai	TanggalMulai	Date and Time	8 Byte	<i>Not Null</i>
11.	Tanggal Akhir	TanggalAkhir	Date and Time	8 Byte	<i>Not Null</i>
12.	Total Hari	TotalHari	Integer	5 Byte	<i>Not Null</i>
13.	Status Izin	StatusIzin	Text String	32 Byte	<i>Not Null</i>
14.	Photo	Photo	Text String	32 Byte	<i>Not Null</i>

(Sumber : Hasil Analisis, 2019)

3. Spesifikasi Objek Kalender Libur

Nama Objek : Kalender_Libur
 Akronim : Kalender_Libur
 Fungsi : Untuk menyimpan data informasi libur nasional dan cuti bersama
 Tipe : Data Master

Tabel V.18 Spesifikasi Objek Kalender_Libur

No	Nama Elemen	Akronim	Tipe	Panjang	Keterangan
1.	Judul	judul	Text String	32 Byte	<i>Not Null</i>
2.	Deskripsi	deskripsi	Text String	146 Byte	<i>Not Null</i>
3.	Tipe	tipe	Text String	24 Byte	<i>Not Null</i>
4.	Tanggal	tanggal	Date and Time	8 Byte	<i>Not Null</i>

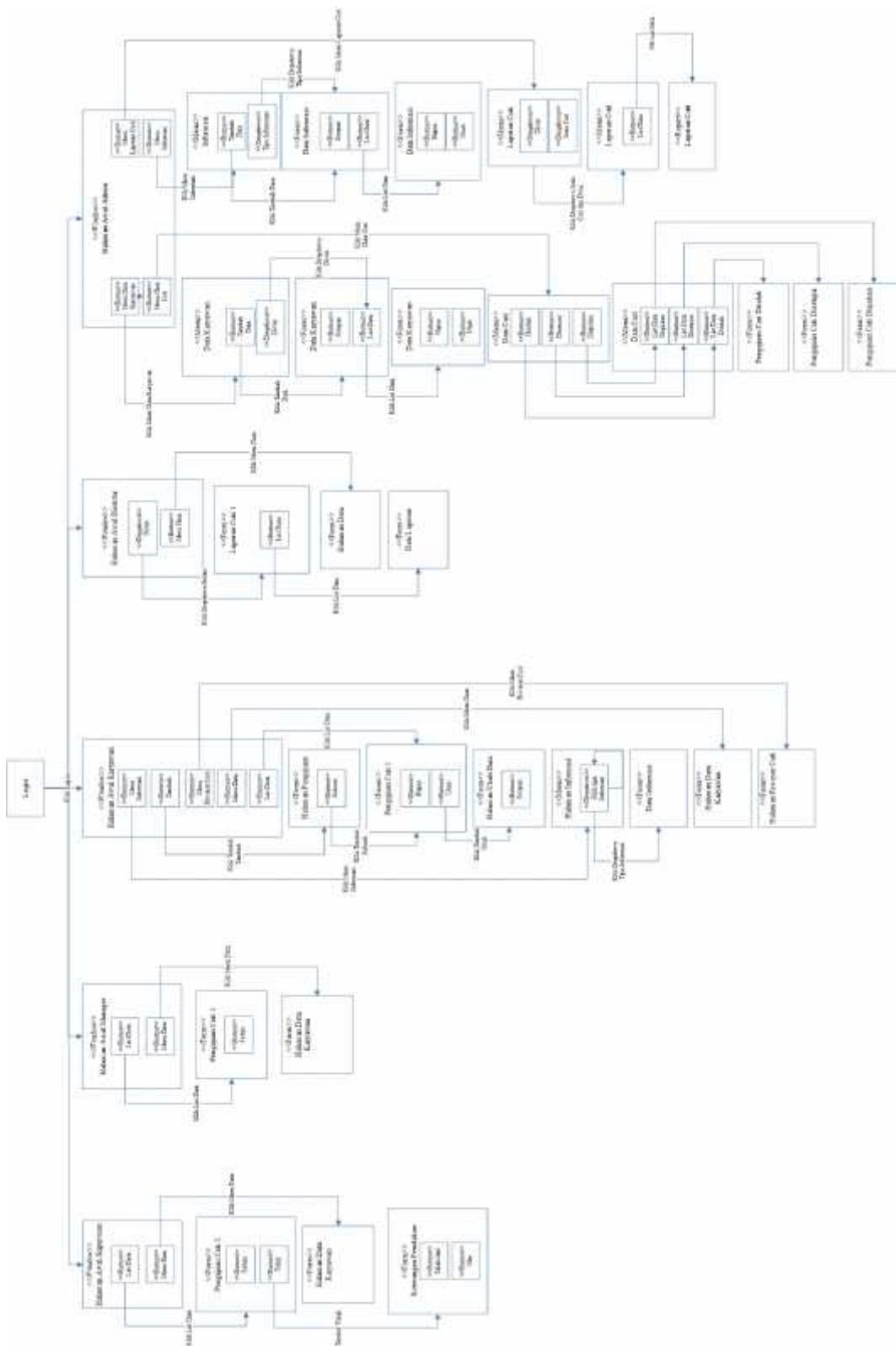
(Sumber : Hasil Analisis, 2019)

5.6 Perancangan Sistem

Perancangan sistem pada sistem informasi cuti karyawan digunakan untuk menggambarkan bagaimana aplikasi yang dibuat berjalan. Perancangan sistem informasi cuti karyawan juga menggambarkan navigasi dari aplikasi tersebut dengan *tools Windows Navigation Diagram* (WND) serta menggambarkan tampilan antar muka dari aplikasi (Perancangan *Interface* Sistem).

5.6.1 Windows Navigation Diagram (WND)

Dengan *Windows Navigation Diagram* kita dapat dengan mudah melihat skema sistem, sehingga akan memudahkan menganalisa sistem. Berikut ini merupakan contoh *Windows Navigation Diagram* usulan pada sistem informasi cuti karyawan, yang dapat dilihat pada Gambar V.31. .



Gambar V.31 WND Sistem Informasi Cuti Karyawan
Sumber: Hasil Analisis (2019)

5.6.2 Perancangan *Interface* Sistem

Rancangan *interface* (antarmuka) dari program sistem informasi cuti karyawan ini bertujuan untuk mengambarkan antarmuka aplikasi yang telah dibuat. Rancangan antarmuka ini merupakan bagian yang berhubungan langsung antara pengguna dengan sistem, berikut rancangan antarmuka sistem informasi cuti karyawan.

1. Halaman *Login*

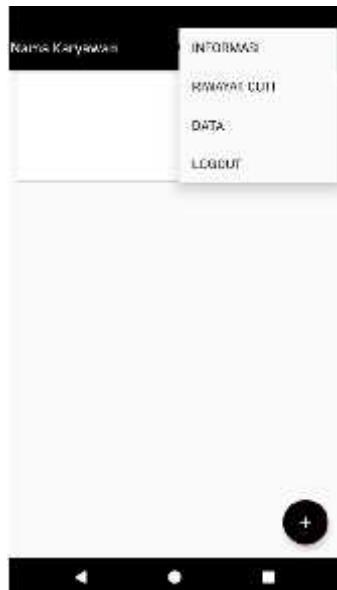
Halaman *login* merupakan halaman yang digunakan oleh pengguna (karyawan) sebagai tampilan awal sebelum masuk ke dalam halaman utama aplikasi. Pengguna harus memasukan nik dan *password* yang terdapat di halaman login agar dapat masuk ke dalam halaman utama aplikasi. Berikut ini halaman Halaman *Login*, bisa dilihat pada Gambar V.32.



Gambar V.32 Tampilan Halaman *Login*
Sumber: Hasil Analisis (2019)

2. Halaman Awal Karyawan

Tampilan halaman Awal Karyawan merupakan tampilan yang muncul pertama kali ketika karyawan telah *login* ke dalam aplikasi. Berikut ini contoh Tampilan Halaman Awal Karyawan, bisa dilihat pada Gambar V.33.



Gambar V.33 Tampilan Halaman Awal Karyawan
Sumber: Hasil Analisis (2019)

3. Tampilan Halaman Awal Supervisor

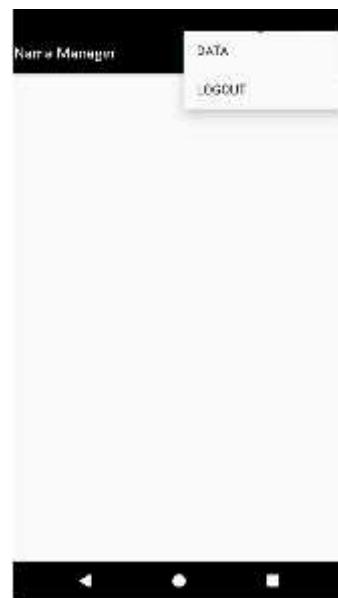
Tampilan halaman Awal Supervisor merupakan tampilan yang muncul pertama kali ketika supervisor telah *login* ke dalam aplikasi. Berikut ini contoh Tampilan Halaman Awal Supervisor, bisa dilihat pada Gambar V.34.



Gambar V.34 Tampilan Halaman Awal Supervisor
Sumber: Hasil Analisis (2019)

4. Tampilan Halaman Awal Manager

Tampilan halaman Awal Manager merupakan tampilan yang muncul pertama kali ketika manager telah *login* ke dalam aplikasi. Berikut ini contoh Tampilan Halaman Awal Manager, bisa dilihat pada Gambar V.35.



Gambar V.35 Tampilan Halaman Awal Manager
Sumber: Hasil Analisis (2019)

6. Halaman Awal Direktur

Tampilan halaman Awal Direktur merupakan tampilan yang muncul pertama kali ketika direktur telah *login* ke dalam aplikasi. Berikut ini contoh Tampilan Halaman Awal Direktur, bisa dilihat pada Gambar V.36.



Gambar V.36 Tampilan Halaman Awal Direktur
Sumber: Hasil Analisis (2019)

7. Halaman Awal Admin

Tampilan halaman Awal Admin merupakan tampilan yang muncul pertama kali ketika direktur telah *login* ke dalam aplikasi. Berikut ini contoh Tampilan Halaman Awal Admin, bisa dilihat pada Gambar V.37.



Gambar V.37 Tampilan Halaman Awal Admin
Sumber: Hasil Analisis (2019)

8. Tampilan Halaman Pengajuan

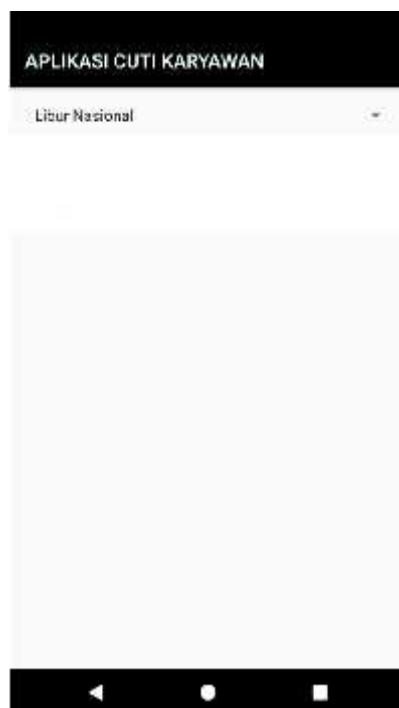
Tampilan halaman pengajuan merupakan tampilan yang muncul ketika karyawan mengklik tombol tambah (+) pada halaman awal karyawan. Berikut ini contoh Tampilan Halaman Pengajuan, bisa dilihat pada Gambar V.38.

FORM PENGAJUAN	
Nama	<input type="text"/>
NIP	<input type="text"/>
Pangkat/Golongan	Tanggal Mulai
Jabatan	Tanggal Akhir
Desa	Total Hari
Satuan Organisasi	
Jenis Cuti	<input type="button" value="SUBMIT"/>

Gambar V.38 Tampilan Halaman Pengajuan
Sumber: Hasil Analisis (2019)

9. Tampilan Halaman Menu Informasi

Tampilan halaman menu informasi merupakan tampilan yang muncul ketika karyawan mengklik tombol menu informasi pada halaman awal karyawan. Berikut ini contoh Tampilan Halaman Menu Informasi, bisa dilihat pada Gambar V.39.



Gambar V.39 Tampilan Halaman Menu Informasi
Sumber: Hasil Analisis (2019)

10. Tampilan Halaman Menu Data Karyawan

Tampilan halaman menu data merupakan tampilan yang muncul ketika karyawan mengklik tombol menu data pada halaman awal karyawan/Supervisor/Manager/Direktur dan Admin mengklik tambah data pada halaman data karyawan menu admin. Berikut ini contoh Tampilan Halaman Menu Data Karyawan, bisa dilihat pada Gambar V.40.

Gambar V.40 Tampilan Halaman Data Karyawan

Sumber: Hasil Analisis (2019)

11. Tampilan Halaman Menu Riwayat Cuti

Tampilan halaman menu data merupakan tampilan yang muncul ketika karyawan mengklik tombol menu riwayat cuti pada halaman awal karyawan. Berikut ini contoh Tampilan Halaman Menu Riwayat Cuti, bisa dilihat pada Gambar V.41.



Gambar V.41 Tampilan Halaman Menu Riwayat Cuti

Sumber: Hasil Analisis (2019)

11. Tampilan Halaman Persetujuan Cuti Oleh Supervisor

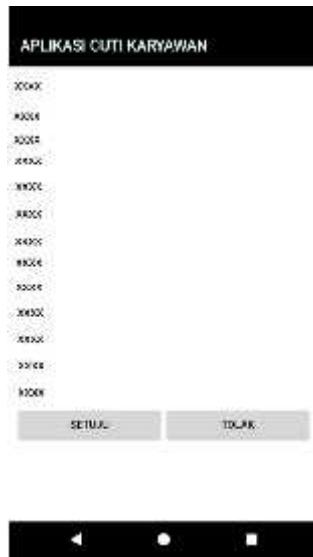
Tampilan halaman Persetujuan Cuti Oleh Supervisor merupakan tampilan yang muncul ketika supervisor mengklik list data pada halaman awal supervisor untuk memberikan persetujuan. Berikut ini contoh Tampilan Halaman Menu Riwayat Cuti, bisa dilihat pada Gambar V.42.



Gambar V.42 Tampilan Halaman Persetujuan Cuti Oleh Supervisor
Sumber: Hasil Analisis (2019)

11. Tampilan Halaman Persetujuan Cuti Oleh Manager

Tampilan halaman Persetujuan Cuti Oleh Manager merupakan tampilan yang muncul ketika manager mengklik list data pada halaman awal Manager untuk memberikan persetujuan. Berikut ini contoh Tampilan Halaman Menu Persetujuan Cuti Oleh Manager, bisa dilihat pada Gambar V.43.



Gambar V.43 Tampilan Halaman Persetujuan Cuti Oleh Manager
Sumber: Hasil Analisis (2019)

12. Tampilan *Form* Penolakan Cuti Oleh Manager

Tampilan *Form* Persetujuan Cuti Oleh Manager merupakan tampilan yang muncul ketika manager mengklik tombol tolak pada *form* pengajuan cuti Manager untuk memberikan penolakan. Berikut ini contoh Tampilan *Form* Penolakan Cuti Oleh Manager, bisa dilihat pada Gambar V.44.



Gambar V.44 Tampilan *Form* Penolakan Cuti Oleh Manager
Sumber: Hasil Analisis (2019)

12. Tampilan Halaman Menu Informasi Pada Admin

Tampilan Halaman Menu Informasi Pada Admin merupakan tampilan yang muncul ketika admin mengklik menu informasi pada halaman awal admin. Berikut ini contoh Tampilan Halaman Menu Informasi Pada Admin, bisa dilihat pada Gambar V.45.



Gambar V.45 Tampilan Halaman Menu Informasi Pada Admin

Sumber: Hasil Analisis (2019)

12. Tampilan *Form* Tambah Data Menu Informasi

Tampilan *Form* Tambah Data Menu Informasi merupakan tampilan yang muncul ketika admin mengklik tambah data pada menu informasi. Berikut ini contoh *Form* Tambah Data Menu Informasi bisa dilihat pada Gambar V.46.

Gambar V.46 Form Tambah Data Menu Informasi
Sumber: Hasil Analisis (2019)

13. Tampilan *Form* Ubah dan Hapus Data Menu Informasi

Tampilan *Form* Ubah dan Hapus Data Menu Informasi merupakan tampilan yang muncul ketika admin mengklik list data pada menu informasi. Berikut ini contoh *Form* Ubah dan Hapus Data Menu Informasi bisa dilihat pada Gambar V.47.

Gambar V.47 Form Ubah dan Hapus Data Menu Informasi
Sumber: Hasil Analisis (2019)

14. Tampilan *Form* Ubah dan Hapus Data Pengajuan Cuti

Tampilan *Form* Ubah dan Hapus Data Pengajuan Cuti merupakan tampilan yang muncul ketika karyawan mengklik list data pada halaman awal karyawan. Berikut ini contoh *Form* Ubah dan Hapus Data Pengajuan Cuti bisa dilihat pada Gambar V.48.

The screenshot shows a list of leave applications with the following details:

No	Tujuan	Tanggal	Waktu	Periode	Status
1	XXX	XXX	XXX	XXX	XXX
2	XXX	XXX	XXX	XXX	XXX
3	XXX	XXX	XXX	XXX	XXX
4	XXX	XXX	XXX	XXX	XXX
5	XXX	XXX	XXX	XXX	XXX
6	XXX	XXX	XXX	XXX	XXX
7	XXX	XXX	XXX	XXX	XXX
8	XXX	XXX	XXX	XXX	XXX
9	XXX	XXX	XXX	XXX	XXX
10	XXX	XXX	XXX	XXX	XXX
11	XXX	XXX	XXX	XXX	XXX
12	XXX	XXX	XXX	XXX	XXX
13	XXX	XXX	XXX	XXX	XXX
14	XXX	XXX	XXX	XXX	XXX
15	XXX	XXX	XXX	XXX	XXX
16	XXX	XXX	XXX	XXX	XXX
17	XXX	XXX	XXX	XXX	XXX
18	XXX	XXX	XXX	XXX	XXX
19	XXX	XXX	XXX	XXX	XXX
20	XXX	XXX	XXX	XXX	XXX
21	XXX	XXX	XXX	XXX	XXX
22	XXX	XXX	XXX	XXX	XXX
23	XXX	XXX	XXX	XXX	XXX
24	XXX	XXX	XXX	XXX	XXX
25	XXX	XXX	XXX	XXX	XXX
26	XXX	XXX	XXX	XXX	XXX
27	XXX	XXX	XXX	XXX	XXX
28	XXX	XXX	XXX	XXX	XXX
29	XXX	XXX	XXX	XXX	XXX
30	XXX	XXX	XXX	XXX	XXX
31	XXX	XXX	XXX	XXX	XXX
32	XXX	XXX	XXX	XXX	XXX
33	XXX	XXX	XXX	XXX	XXX
34	XXX	XXX	XXX	XXX	XXX
35	XXX	XXX	XXX	XXX	XXX
36	XXX	XXX	XXX	XXX	XXX
37	XXX	XXX	XXX	XXX	XXX
38	XXX	XXX	XXX	XXX	XXX
39	XXX	XXX	XXX	XXX	XXX
40	XXX	XXX	XXX	XXX	XXX
41	XXX	XXX	XXX	XXX	XXX
42	XXX	XXX	XXX	XXX	XXX
43	XXX	XXX	XXX	XXX	XXX
44	XXX	XXX	XXX	XXX	XXX
45	XXX	XXX	XXX	XXX	XXX
46	XXX	XXX	XXX	XXX	XXX
47	XXX	XXX	XXX	XXX	XXX
48	XXX	XXX	XXX	XXX	XXX
49	XXX	XXX	XXX	XXX	XXX
50	XXX	XXX	XXX	XXX	XXX
51	XXX	XXX	XXX	XXX	XXX
52	XXX	XXX	XXX	XXX	XXX
53	XXX	XXX	XXX	XXX	XXX
54	XXX	XXX	XXX	XXX	XXX
55	XXX	XXX	XXX	XXX	XXX
56	XXX	XXX	XXX	XXX	XXX
57	XXX	XXX	XXX	XXX	XXX
58	XXX	XXX	XXX	XXX	XXX
59	XXX	XXX	XXX	XXX	XXX
60	XXX	XXX	XXX	XXX	XXX
61	XXX	XXX	XXX	XXX	XXX
62	XXX	XXX	XXX	XXX	XXX
63	XXX	XXX	XXX	XXX	XXX
64	XXX	XXX	XXX	XXX	XXX
65	XXX	XXX	XXX	XXX	XXX
66	XXX	XXX	XXX	XXX	XXX
67	XXX	XXX	XXX	XXX	XXX
68	XXX	XXX	XXX	XXX	XXX
69	XXX	XXX	XXX	XXX	XXX
70	XXX	XXX	XXX	XXX	XXX
71	XXX	XXX	XXX	XXX	XXX
72	XXX	XXX	XXX	XXX	XXX
73	XXX	XXX	XXX	XXX	XXX
74	XXX	XXX	XXX	XXX	XXX
75	XXX	XXX	XXX	XXX	XXX
76	XXX	XXX	XXX	XXX	XXX
77	XXX	XXX	XXX	XXX	XXX
78	XXX	XXX	XXX	XXX	XXX
79	XXX	XXX	XXX	XXX	XXX
80	XXX	XXX	XXX	XXX	XXX
81	XXX	XXX	XXX	XXX	XXX
82	XXX	XXX	XXX	XXX	XXX
83	XXX	XXX	XXX	XXX	XXX
84	XXX	XXX	XXX	XXX	XXX
85	XXX	XXX	XXX	XXX	XXX
86	XXX	XXX	XXX	XXX	XXX
87	XXX	XXX	XXX	XXX	XXX
88	XXX	XXX	XXX	XXX	XXX
89	XXX	XXX	XXX	XXX	XXX
90	XXX	XXX	XXX	XXX	XXX
91	XXX	XXX	XXX	XXX	XXX
92	XXX	XXX	XXX	XXX	XXX
93	XXX	XXX	XXX	XXX	XXX
94	XXX	XXX	XXX	XXX	XXX
95	XXX	XXX	XXX	XXX	XXX
96	XXX	XXX	XXX	XXX	XXX
97	XXX	XXX	XXX	XXX	XXX
98	XXX	XXX	XXX	XXX	XXX
99	XXX	XXX	XXX	XXX	XXX
100	XXX	XXX	XXX	XXX	XXX

Gambar V.48 *Form* Ubah dan Hapus Data Pengajuan Cuti

Sumber: Hasil Analisis (2019)

15. Tampilan Halaman Menu Data Karyawan Pada Admin

Tampilan Halaman Menu Data Karyawan Pada Admin merupakan tampilan yang muncul ketika admin mengklik menu data karyawan pada halaman awal admin. Berikut ini contoh Tampilan Halaman Menu Data Karyawan Pada Admin, bisa dilihat pada Gambar V.49.



Gambar V.49 Halaman Menu Data Karyawan Pada Admin
Sumber: Hasil Analisis (2019)

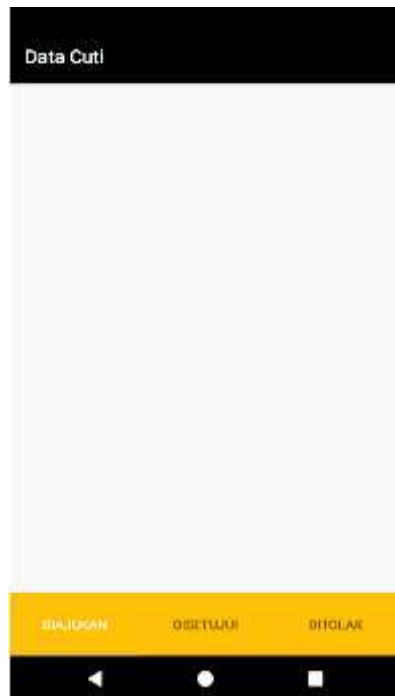
16. Tampilan *Form* Ubah dan Hapus Data Karyawan

Tampilan *Form* Ubah dan Hapus Data Karyawan merupakan tampilan yang muncul ketika admin mengklik list data pada menu data karyawan. Berikut ini contoh *Form* Ubah dan Hapus Data Karyawan bisa dilihat pada Gambar V.50.

Gambar V.50 Tampilan *Form* Ubah dan Hapus Data Karyawan
Sumber: Hasil Analisis (2019)

17. Tampilan Halaman Menu Data Cuti Pada Admin

Tampilan Halaman Menu Data Cuti Pada Admin merupakan tampilan yang muncul ketika admin mengklik menu data cuti pada halaman awal admin. Berikut ini contoh Tampilan Halaman Menu Data Cuti Pada Admin, bisa dilihat pada Gambar V.51.

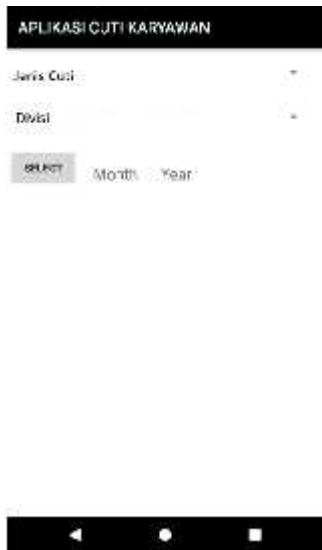


Gambar V.51 Tampilan Halaman Menu Data Cuti Pada Admin

Sumber: Hasil Analisis (2019)

18. Tampilan Halaman Menu Laporan Cuti Pada Admin

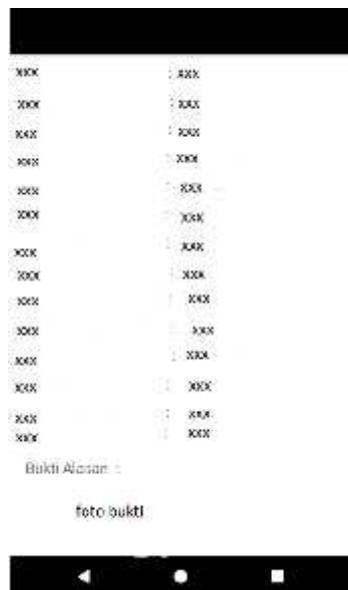
Tampilan Halaman Menu Laporan Cuti Pada Admin merupakan tampilan yang muncul ketika admin mengklik menu laporan cuti pada halaman awal admin. Berikut ini contoh Tampilan Halaman Menu laporan Cuti Pada Admin, bisa dilihat pada Gambar V.52.



Gambar V.52 Tampilan Halaman Menu Laporan Cuti Pada Admin
Sumber: Hasil Analisis (2019)

18. Tampilan Detail Laporan Cuti

Tampilan Detail Laporan Cuti merupakan tampilan yang muncul ketika admin dan direktur mengklik list data pada halaman awal admin dan direktur. Berikut ini contoh Tampilan Detail laporan Cuti, bisa dilihat pada Gambar V.53.



Gambar V.53 Tampilan Detail Laporan Cuti
Sumber: Hasil Analisis (2019)

5.7 Implementasi Sistem

Tahap ini adalah tahap dimana menerapkan aplikasi yang telah dibangun pada pihak *client*. Tahap implementasi atau penerapan aplikasi yang telah dibangun ini membutuhkan beberapa kebutuhan *software* dan *hardware* agar aplikasi dapat berjalan maksimal. Berikut adalah spesifikasi yang dibutuhkan untuk aplikasi cuti karyawan:

1. Analisis Kebutuhan *Software*
 - a. Sistem Operasi : *Microsoft Windows 8*
 - b. *Database Server* : *Firebase Realtime Database*
 - c. Bahasa Pemrograman : Java Development Kit 7 atau 8 (Support API 24)
 - d. Perangkat : Android Mobile (Min. Versi Kitkat)
 - e. Framework : *Android Studio Versi 8*
2. Analisis Kebutuhan *Hardware*
 - a. *Processor* : Minimal *Processor* Intel VT-x, Intel EM64T and Execute Disable (XD) Bit Functionally
 - b. RAM : Minimal 4 GB
 - c. *Harddisk* : Minimal *Harddisk* 4 GB
 - d. *Mouse, Keyboard, Monitor* sebagai peralatan muka.

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Berdasarkan hasil penelitian selama proses pengumpulan dan pengolahan data, analisis sistem, dan implementasi sistem usulan maka dapat diambil kesimpulan sebagai berikut:

1. Dengan membuat sistem informasi cuti karyawan berbasis *mobile*, diharapkan dapat mempermudah proses pengajuan karena tidak adanya keterhambatan dalam proses persetujuan supervisor, manager dan direktur.
2. Dengan adanya sistem informasi cuti karyawan ini penyimpanan data lebih tertata rapi di *database* karena tidak adanya data yang hilang atau rusak.
3. Dengan adanya sistem informasi cuti karyawan ini membuat laporan cuti menjadi bentuk yang informatif sehingga Manager SDM dan Direktur tidak perlu menanyakan kembali kepada staff SDM

6.2 Saran

Saran yang diberikan untuk pengembangan sistem informasi cuti karyawan ini selanjutnya adalah sebagai berikut:

1. Untuk penerapan sistem baru terhadap sistem lama, sebaiknya dilakukan secara bertahap dan diperlukan sosialiasasi penggunaan sistem ini kepada bagian yang terkait.
2. Sebaiknya dilakukan pemeliharaan aplikasi secara berkala, sehingga aplikasi dapat berjalan dengan baik.
3. Untuk penelitian selanjutnya sistem informasi ini bisa ditambahkan ruang lingkupnya seperti menambahkan jenis cuti yang lebih terperinci dan absensi karyawan.

DAFTAR PUSTAKA

- A.S, Rosa., dan Shalahuddin. 2016. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Jakarta: Penerbit Informatika
- Alfikri, Imaduddin., Herumurti, Darlis., dan Rahman H, Ridho. 2016. *Aplikasi Navigasi Berbasis Perangkat Bergerak Dengan Menggunakan Platform Wikitude Untuk Studi Kasus Lingkungan ITS*. Jurnal Teknik ITS : Vol. 5 No. 1. Surabaya: Institut Teknologi 10 November
- Dennis, Alan., Wixom, Barbara Harley., dan Tegarden, David. 2015. *System Analysis & Design. An Object-Oriented Approach with UML (Fifth Edition)*. USA: Wiley.
- Developers, Google. 2017. *Firebase*. <https://firebase.google.com/>. (25 Oktober 2017)
- Djahir, Yulia., dan Pratita, Dewi. 2014. *Bahan Ajar Sistem Informasi Manajemen*. Yogyakarta: Deepublish.
- El Karimah, Khoirunnisa. 2012. *Hubungan Stres Dengan Kepuasan Kerja Karyawan di Tiga Direktorat Operasional PT PLN (Persero)*. Skripsi Program Sarjana Ilmu Administrasi Negara, Depok : Universitas Indonesia
- Hutahaean, Jeperson. 2015. *Konsep Sistem Informasi*. Jakarta: Deepublish.
- Jogiyanto, H.M. 2012. *Analisa dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktik Aplikasi Bisnis*. Yogyakarta: Andi.
- Mulyani, Sri. 2016. *Metode Analisis dan Perancangan Sistem*. Bandung: Abdi Sistematika.
- Peraturan Badan Kepegawaian Negara Republik Indonesia. 2017. *Peraturan No. 24 Tahun 2017 Tentang Cuti Pegawai*. Jakarta: Visi Media
- Peraturan Badan Kepegawaian Negara Republik Indonesia. 2017. *Pasal 93 ayat 2 dan 4 UU no.13 tahun 2003 Tentang Tenaga Kerja*. Jakarta: Visi Media
- Ragil Wiratno, Andreas., dan Hastuti, Khafizh. 2017. *Implementation Of Firebase Realtime Database to Track BRT Trans Semarang*. Scientific Journal Of Informatics: Vol. 4, No. 2. Semarang: Universitas Dian Nuswantoro
- Simarmata, Janer. 2010. *Rekayasa Perangkat Lunak*. Jakarta: Published

- Supardi, Ir. Yuniar. 2015. *Semua Bisa Menjadi Programmer Web Java – Case Study*. Jakarta: PT Elex Media Komputindo.
- Supardi, Ir. Yuniar. 2017. *Koleksi Program Tugas Akhir dan Skripsi Dengan Android*. Jakarta: PT Elex Media Komputindo.
- Sutabri, Tata. 2012. Analisa Sistem Informasi. Yogyakarta: Andi.
- Waljiyanto. 2003. *Sistem Basis Data Analisis dan Pemodelan Data*. Yogyakarta: Penerbit Graha Ilmu
- Yani. 2012. *Manajemen Sumber Daya Manusia*. Jakarta: Mitra Wacana Media.

LAMPIRAN A

BUKTI WAWANCARA

Nama Narasumber : Dody Sidik Permana (Supervisor Sumber Daya Manusia)
Tanggal : 22 Oktober 2018
Jam : 13.30 – 15.00 WIB
Tempat Wawancara : Ruang Kantor
Topik Wawancara : Proses pengajuan cuti di PT IKT

Pewawancara : Bagaimana alur proses pengajuan cuti?

Narasumber : alur yang pertama yaitu karyawan mengajukan surat cuti lalu minta persetujuan berupa tanda tangan kepada supervisor dan manager divisi terkait lalu diserahkan kepada divisi sumber daya manusia untuk dibuatkan surat izin cuti yang kemudian akan kita beri nomor surat dan akan di tanda tangan oleh direktur keuangan & SDM baru setelah itu karyawan bisa cuti

Pewawancara : cuti ada apa aja jenis cuti yang bisa karyawan ambil?

Narasumber : ada 6, yaitu cuti sakit, cuti tahunan, cuti besar, cuti melahirkan, cuti karena alasan penting dan cuti naik haji

Pewawancara : kalau supervisor dan manager itu kadang siapa yang berhak menolak pengajuan cuti karyawan?

Narasumber : sebenarnya kalau di kita kan bisa dilihat karna supervisor satu ruangan sama karyawan jadi kadang sebelum si karyawan membuat surat pengajuan cuti bisa tanya langsung ke supervisor boleh atau tidaknya ambil cuti jadi ya kalau tidak diizinkan si karyawan tidak jadi ambil cuti, dan biasanya penolakan terjadi di manager

Pewawancara : kalau direktur dia bisa menolak juga?

Narasumber : kadang bisa menolak juga jika dirasa karyawan ini tidak boleh mengambil cuti dengan beberapa alasan, cuma yaa jarang juga direktur menolak karna kan sudah ada manager yang mempertimbangkan

Pewawancara : alasan ditolak biasanya karena apa?

Narasumber : yang pernah itu jika mau dinas keluar kota dalam minggu-minggu terdekat atau biasanya orang lapangan kan ekspor-impor suka padat

Pewawancara : tadi kan ada surat izin cuti, itu di buat berapa dan dijadikan arsip?

Narasumber : iya, dijadikan arsip nah surat izin cuti kita buatkan 2 rangkap yang satu diberikan kepada karyawan sendiri dan satunya untuk arsip

Pewawancara : kalau arsip disimpan dimana?

Narasumber : kita melakukan proses pencatatan riwayat cuti dulu di komputer saya buatkan di excel lalu surat nya disimpan di dalam map aja di meja staff saya

Pewawancara : kan tadi ada 6 jenis cuti itu ada syarat dan kententuannya?

Narasumber : iya ada pastinya, yang pertama cuti tahunan sesuai ketentuan undang-undang biasanya dapat diberikan jika karyawan tersebut sudah melewati masa kerja 1 tahun, dalam setahun dapat mengambil 12 hari dalam sekali ambil minimal 3 hari ambil dan tidak boleh sekaligus ambil 12 hari dalam sekali ambil, yang kedua cuti sakit kalau cuti sakit 1 atau 2 hari karyawan berhak mengambil cuti jika lebih 2 hari karyawan dapat memberikan surat sakit kepada supervisor dan manager divisi masing-masing dan wanita yang mengalami keguguran berhak atas cutinya paling lama $\frac{1}{2}$ (satu setengah) bulan dengan melampirkan surat keterangan sakit, yang ketiga yaitu cuti melahirkan setiap wanita yang ingin melahirkan berhak atas cutinya yaitu 3 bulan, yang keempat cuti besar, cuti besar dapat diambil jika masa kerja karyawan sekurang-kurangnya 6 tahun dan lama cuti yang bisa diambil yaitu 3 bulan, yang kelima cuti naik haji lamanya waktu yang diberikan paling lama 45 hari, yang keenam cuti karena alasan penting, cuti karena alasan penting bisa mencakup karyawan menikah bisa diambil 3 hari, menikahkan anaknya bisa diambil 2 hari, mengkhitanan anaknya bisa diambil 2 hari, keluarga meninggal dunia bisa diambil 2 hari lihat saja di undang-undang tidak terlalu beda jauh

Pewawancara : karyawan di IKT berapa totalnya dan bagaimana pembagiannya?

Narasumber : karyawan atau tenaga kerja itu kalo di IKT menjadi 3 bagian yaitu ada tenaga organik yaitu ada 70 orang ditambah direksi 5 orang, PKWT ada 3 orang dan outsourcing 287 orang, PKWT dan outsourcing termasuk tenaga non organik

Pewawancara : pencatatan riwayat cuti diberikan kepada siapa saja?

Narasumber : untuk divisi SDM sendiri dan direktur keuangan dan SDM

LAMPIRAN B

KODE PROGRAM & TAMPILAN PROGRAM

A. Kode Program

1. Pengajuan Cuti

```

package com.example.aplikasicutikaryawan;

import android.annotation.SuppressLint;
import android.app.DatePickerDialog;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

import com.example.aplikasicutikaryawan.model.PengajuanCutiData;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.GregorianCalendar;
import java.util.concurrent.TimeUnit;

public class PENGAJUAN_CUTI extends AppCompatActivity {
    EditText Nama, NIPP, Pangkat, Jabatan, SatuanOrg, Alasan, Alamat,
    TanggalMulai, TanggalAkhir, totalHari;
    Spinner JenisCuti;
    String user, divisi;
    private static Boolean edit = false;
    private PengajuanCutiData objek;
    DatabaseReference dataref;
    private Date tgl_awal, tgl_akhir;
    private SimpleDateFormat dateFormatter;
    private DatePickerDialog datePickerDialog;
    private long days;

    @SuppressLint("SimpleDateFormat")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_pengajuan_cuti);
    }
}

```

```

dataref = FirebaseDatabase.getInstance().getReference();
Nama = findViewById(R.id.nama_pengajuan);
NIPP = findViewById(R.id.nipp_pengajuan);
Pangkat = findViewById(R.id.pangkat_pengajuan);
Jabatan = findViewById(R.id.jabatan_pengajuan);
SatuanOrg = findViewById(R.id.satuan_pengajuan);
JenisCuti = findViewById(R.id.jenis_cuti);
Alasan = findViewById(R.id.alasan_pengajuan);
Alamat = findViewById(R.id.alamat_pengajuan);
TanggalMulai = findViewById(R.id.tgl_mulai);
TanggalAkhir = findViewById(R.id.tgl_akhir);
totalHari = findViewById(R.id.totalHari);

dateFormatter = new SimpleDateFormat("dd MMM yyyy");
user = getSharedPreferences("preferences",
    MODE_PRIVATE).getString("username", "");
divisi = getSharedPreferences("preferences",
    MODE_PRIVATE).getString("userdivisi", "");
NIPP.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("NIPP", ""));
Nama.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("username", ""));
Jabatan.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("usertipe", ""));
((EditText) findViewById(R.id.Divisi)).setText(divisi);
setTitle("FORM PENGAJUAN");

if (getIntent().getParcelableExtra("data")!=null){
    edit = true;
    ((Button) findViewById(R.id.btn_pengajuan)).setText("Edit");
    objek = getIntent().getParcelableExtra("data");
    Nama.setText(objek>Nama);
    NIPP.setText(objek.NIPP);
    Pangkat.setText(objek.Pangkat);
    Jabatan.setText(objek.Jabatan);
    SatuanOrg.setText(objek.SatuanOrg);
    Alasan.setText(objek.Alasan);
    Alamat.setText(objek.Alamat);
    TanggalMulai.setText(new SimpleDateFormat("dd MMM
yyyy").format(objek.TanggalMulai));
    TanggalAkhir.setText(new SimpleDateFormat("dd MMM
yyyy").format(objek.TanggalAkhir));
    days = objek.totalHari;
    tgl_awal = objek.TanggalMulai;
    tgl_akhir = objek.TanggalAkhir;
    totalHari.setText(objek.totalHari+" hari");
}
}

public void submit(View view) {
    final String input_Nama = Nama.getText().toString();
    final String input_NIPP = NIPP.getText().toString();
    final String input_Pangkat = Pangkat.getText().toString();
    final String input_Jabatan = Jabatan.getText().toString();
    final String input_SatuanOrg = SatuanOrg.getText().toString();
    final String input_JenisCuti =
JenisCuti.getSelectedItem().toString();
    final String input_Alasan = Alasan.getText().toString();
}

```

```

final String input_Alamat = Alamat.getText().toString();

if (edit){
    final PengajuanCutiData current = new PengajuanCutiData(
        user,
        divisi,
        input_Nama,
        input_NIPP,
        input_Pangkat,
        input_Jabatan,
        input_SatuanOrg,
        input_JenisCuti,
        input_Alasan,
        input_Alamat,
        tgl_awal,
        tgl_akhir,
        days,
        "Diajukan" );
}

current.key = objek.key;

dataref.child("PengajuanCuti").child(objek.key).setValue(current)
    .addOnSuccessListener(new OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            Intent closeThis = new Intent();
            closeThis.putExtra("data", current);
            setResult(100, closeThis);
            finish();
        }
    })
    .addOnFailureListener(new OnFailureListener() {
        @Override
        public void onFailure(@NonNull Exception e) {
            Intent closeThis = new Intent();
            closeThis.putExtra("error", e.getLocalizedMessage());
            setResult(101, closeThis);
            finish();
        }
    });
}};

else {
    PengajuanCutiData current = new PengajuanCutiData(
        user,
        divisi,
        input_Nama,
        input_NIPP,
        input_Pangkat,
        input_Jabatan,
        input_SatuanOrg,
        input_JenisCuti,
        input_Alasan,
        input_Alamat,
        tgl_awal,
        tgl_akhir,
        days,
        "Diajukan" );
}

dataref.child("PengajuanCuti").push().setValue(current)
    .addOnSuccessListener(new OnSuccessListener<Void>() {
        @Override
        public void onSuccess(Void aVoid) {
            onBackPressed();
        }
    })
}

```

```

        });
    }

    public void dialogEndDate(final View view) {
        Calendar newCalendar = Calendar.getInstance();
        if (tgl_awal!=null&&view.getId()==R.id.tgl_mulai){
            newCalendar.setTime(tgl_awal);
        }else if (tgl_akhir!=null&&view.getId()==R.id.tgl_akhir){
            newCalendar.setTime(tgl_akhir);
        }

        datePickerDialog = new DatePickerDialog(this, new
        DatePickerDialog.OnDateSetListener() {
            @Override
            public void onDateSet(DatePicker datePicker, int year, int
monthOfYear, int dayOfMonth) {
                Calendar newDate = Calendar.getInstance();
                switch (view.getId()){
                    case R.id.tgl_mulai:
                        newDate.set(year, monthOfYear, dayOfMonth, 0, 0,
1);
                        tgl_awal = newDate.getTime();

TanggalMulai.setText(dateFormatter.format(newDate.getTime()));
                    break;

                    case R.id.tgl_akhir:
                        newDate.set(year, monthOfYear, dayOfMonth, 23,
59, 59);
                        tgl_akhir = newDate.getTime();

TanggalAkhir.setText(dateFormatter.format(newDate.getTime()));
                    break;
                }
            }
        });
        newCalendar.set(Calendar.YEAR,
newCalendar.get(Calendar.MONTH), newCalendar.get(Calendar.DAY_OF_MONTH));

        datePickerDialog.show();
    }

    void getDateRange(){
        if (tgl_awal!=null && tgl_akhir!=null){
            Calendar calStart = new GregorianCalendar();
            calStart.setTime(tgl_awal);

            Calendar calEnd = new GregorianCalendar();
            calEnd.setTime(tgl_akhir);
            days = 0;

            while (calStart.before(calEnd)){
                if (calStart.get(Calendar.DAY_OF_WEEK) !=
Calendar.SATURDAY && calStart.get(Calendar.DAY_OF_WEEK) !=
Calendar.SUNDAY) {
                    ++days;
                }
                calStart.add(Calendar.DATE, 1);
            }

            if (JenisCuti.getSelectedItem().toString().equals("Cuti

```

```
Tahunan" )) {  
    if (days<=2) {  
        days = 0;  
        Toast.makeText(this, "Tidak boleh kurang dari 3  
hari", Toast.LENGTH_SHORT).show();  
        TanggalAkhir.setText("");  
    }else if (days>9){  
        days = 0;  
        Toast.makeText(this, "Tidak boleh diambil bersamaan",  
Toast.LENGTH_SHORT).show();  
        TanggalAkhir.setText("");  
    }else {  
        totalHari.setText(days + " Hari");  
    }  
}else {  
    totalHari.setText(days + " Hari");  
}  
}  
}  
}  
}
```

2. Informasi

```
package com.example.aplikasicutikaryawan;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.annotation.SuppressLint;
import android.app.DatePickerDialog;
import android.os.Bundle;
import android.view.View;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;

import com.example.aplikasicutikaryawan.model.InformasiData;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.database.FirebaseDatabase;

import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;

public class form_informasi extends AppCompatActivity{
    EditText judul, deskripsi, tanggal;
    DatePickerDialog datePickerDialog;
    Spinner tipe;
    Date tanggalnya;
    String key;
    InformasiData current;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_form_informasi);
        judul = findViewById(R.id.judul);
        deskripsi = findViewById(R.id.deskripsi);
        tanggal = findViewById(R.id.tanggal);
        tipe = findViewById(R.id.tipe);
        key = savedInstanceState.getString("key");
        current = new InformasiData();
        current.setKey(key);
        tanggalnya = savedInstanceState.getDate("tanggal");
        judul.setText(current.getJudul());
        deskripsi.setText(current.getDeskripsi());
        tipe.setSelection(current.getTipe());
        tanggal.setInitialDate(tanggalnya);
        tipe.setOnItemSelectedListener(this);
        tanggal.setOnDateChangeListener(this);
        if(key != null) {
            judul.setEnabled(false);
            deskripsi.setEnabled(false);
            tipe.setEnabled(false);
            tanggal.setEnabled(false);
        }
    }

    @Override
    protected void onSaveInstanceState(Bundle savedInstanceState) {
        super.onSaveInstanceState(savedInstanceState);
        savedInstanceState.putString("key", current.getKey());
        savedInstanceState.putString("judul", judul.getText().toString());
        savedInstanceState.putString("deskripsi", deskripsi.getText().toString());
        savedInstanceState.putInt("tipe", tipe.getSelectedItemPosition());
        savedInstanceState.setDate("tanggal", tanggal.getInitialDate());
    }

    @Override
    public void onItemSelected(AdapterView parent, View view, int position, long id) {
        current.setTipe(position);
    }

    @Override
    public void onDateChanged(DatePickerDialog view, int year, int month, int day) {
        current.setTanggal(day, month, year);
    }
}
```

```

        setContentView(R.layout.activity_form_informasi);
        judul = findViewById(R.id.etjudul);
        tanggal = findViewById(R.id.ettanggal);
        deskripsi = findViewById(R.id.etteterangan);
        tipe = findViewById(R.id.ettipe);
    }

    @SuppressLint("SimpleDateFormat")
    @Override
    protected void onStart() {
        super.onStart();
        if (getIntent().getBooleanExtra("edit", false)){
            (findViewById(R.id.btn_informasi)).setVisibility(View.GONE);
            (findViewById(R.id.buttonnya)).setVisibility(View.VISIBLE);
            current = getIntent().getParcelableExtra("data");
            judul.setText(current.judul);
            deskripsi.setText(current.deskripsi);
            tanggal.setText(new SimpleDateFormat("dd MMM
            yyyy").format(current.tanggal));
            tanggalnya = current.tanggal;
            key = current.key;
        }
    }

    public void Simpan(View view) {
        FirebaseDatabase.getInstance().getReference().child("Informasi").push()
            .setValue(new InformasiData(judul.getText().toString(),
            tipe.getSelectedItem().toString(), deskripsi.getText().toString(),
            tanggalnya))
            .addOnSuccessListener(new OnSuccessListener() {
                @Override
                public void onSuccess(Void aVoid) {
                    onBackPressed();
                }
            }).addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    Toast.makeText(form_informasi.this,
                    e.getLocalizedMessage(), Toast.LENGTH_SHORT).show();
                }
            });
    }

    public void openDialog(View view) {
        Calendar newCalendar = Calendar.getInstance();
        if (tanggalnya != null) {
            newCalendar.setTime(tanggalnya);
        }
        datePickerDialog = new DatePickerDialog(this, new
        DatePickerDialog.OnDateSetListener() {
            @SuppressLint("SimpleDateFormat")
            @Override
            public void onDateSet(DatePicker datePicker, int year, int
            monthOfYear, int dayOfMonth) {
                Calendar newDate = Calendar.getInstance();
                newDate.set(year, monthOfYear, dayOfMonth, 0, 0, 1);
                tanggalnya = newDate.getTime();
                tanggal.setText(new SimpleDateFormat("dd MMM
                yyyy").format(newDate.getTime()));
            }
        }, newCalendar.get(Calendar.YEAR),
        newCalendar.get(Calendar.MONTH), newCalendar.get(Calendar.DAY_OF_MONTH));
    }
}

```

```

        datePickerDialog.show();
    }

    public void ubah(View view) {
        current = new InformasiData(judul.getText().toString(),
        tipe.getSelectedItem().toString(), deskripsi.getText().toString(),
        tanggalnya);
        current.key = key;

        FirebaseDatabase.getInstance().getReference().child("Informasi").child(key)
            .setValue(current)
            .addOnSuccessListener(new OnSuccessListener<Void>() {
                @Override
                public void onSuccess(Void aVoid) {
                    onBackPressed();
                }
            }).addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    Toast.makeText(form_informasi.this,
                    e.getLocalizedMessage(), Toast.LENGTH_SHORT).show();
                }
            });
    }

    public void hapus(View view) {

        FirebaseDatabase.getInstance().getReference().child("Informasi").child(current.key).removeValue()
            .addOnSuccessListener(new OnSuccessListener<Void>() {
                @Override
                public void onSuccess(Void aVoid) {
                    onBackPressed();
                }
            }).addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    Toast.makeText(form_informasi.this,
                    e.getLocalizedMessage(), Toast.LENGTH_SHORT).show();
                }
            });
    }
}

```

3. User Data

```

package com.example.aplikasicutikaryawan.model;

import android.os.Parcel;
import android.os.Parcelable;

import com.google.firebaseio.database.IgnoreExtraProperties;

@IgnoreExtraProperties
public class UserData implements Parcelable {

```

```
public String key;
public String Nama;
public String TempatTanggalLahir;
public String JenisKelamin;
public String NIPP;
public String NIK;
public String tipe;
public String divisi;
public String Nomortelepon;
public String Gelar;
public String Usia;
public String Agama;
public String AlamatAsal;
public String AlamatDomisili;
public String Email;
public String Socialmedia;
public String password;

public UserData(){ }

public UserData( String Nama,
                 String TempatTanggalLahir,
                 String JenisKelamin,
                 String NIPP,
                 String NIK,
                 String tipe,
                 String divisi,
                 String Nomortelepon,
                 String Gelar,
                 String Usia,
                 String Agama,
                 String AlamatAsal,
                 String AlamatDomisili, String Email, String
Socialmedia, String password){

    this.Nama = Nama;
    this.TempatTanggalLahir = TempatTanggalLahir;
    this.JenisKelamin = JenisKelamin;
    this.NIPP = NIPP;
    this.NIK = NIK;
    this.tipe = tipe;
    this.divisi = divisi;
    this.Nomortelepon = Nomortelepon;
    this.Gelar = Gelar;
    this.Usia = Usia;
    this.Agama = Agama;
    this.AmatAsal = AlamatAsal;
    this.AmatDomisili = AlamatDomisili;
    this.Email = Email;
    this.Socialmedia = Socialmedia;
    this.password = password;

}

@Override
public int describeContents() {
    return 0;
}
```

```

}

@Override
public void writeToParcel(Parcel dest, int flags) {
    dest.writeString(this.key);
    dest.writeString(this.Nama);
    dest.writeString(this.TempatTanggallahir);
    dest.writeString(this.JenisKelamin);
    dest.writeString(this.NIPP);
    dest.writeString(this.NIK);
    dest.writeString(this.tipe);
    dest.writeString(this.divisi);
    dest.writeString(this.Nomortelepon);
    dest.writeString(this.Gelar);
    dest.writeString(this.Usia);
    dest.writeString(this.Agama);
    dest.writeString(this.AlamatAsal);
    dest.writeString(this.AlamatDomisili);
    dest.writeString(this.Email);
    dest.writeString(this.Socialmedia);
    dest.writeString(this.password);
}

protected UserData(Parcel in) {
    this.key = in.readString();
    this.Nama = in.readString();
    this.TempatTanggallahir = in.readString();
    this.JenisKelamin = in.readString();
    this.NIPP = in.readString();
    this.NIK = in.readString();
    this.tipe = in.readString();
    this.divisi = in.readString();
    this.Nomortelepon = in.readString();
    this.Gelar = in.readString();
    this.Usia = in.readString();
    this.Agama = in.readString();
    this.AlamatAsal = in.readString();
    this.AlamatDomisili = in.readString();
    this.Email = in.readString();
    this.Socialmedia = in.readString();
    this.password = in.readString();
}

public static final Parcelable.Creator<UserData> CREATOR = new
Parcelable.Creator<UserData>() {
    @Override
    public UserData createFromParcel(Parcel source) {
        return new UserData(source);
    }

    @Override
    public UserData[] newArray(int size) {
        return new UserData[size];
    }
};
}

```

4. Adapter Cuti Karyawan

```

    package
com.example.aplikasicutikaryawan.activities.home_karyawan;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import com.example.aplikasicutikaryawan.R;
import com.example.aplikasicutikaryawan.model.PengajuanCutiData;

import java.util.ArrayList;

public class AdapterCutiKaryawan extends
RecyclerView.Adapter<AdapterCutiKaryawan.viewHolder>{
    public ArrayList<PengajuanCutiData> list = new ArrayList<>();

    // TODO method ini dipakai untuk menentukan layout apa yang digunakan
    // untuk recyclerview

    private itemClick listener;

    public AdapterCutiKaryawan(itemClick listener) {
        this.listener = listener;
    }

    @NonNull
    @Override
    public viewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        return new
viewHolder(LayoutInflater.from(parent.getContext()).inflate(R.layout.list
_datakaryawan, parent, false));
    }

    // TODO method ini digunakan untuk menentukan nilai apa yang disimpan
    // dalam view
    @Override
    public void onBindViewHolder(@NonNull final viewHolder holder, int
position) {
        final PengajuanCutiData current = list.get(position);

        holder.nama.setText(current.Nama);
        holder.jenis.setText(current.JenisCuti);
        holder.status.setText(current.StatusIzin);
        holder.itemView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                listener.onItemClicked(current, holder.getAdapterPosition());
            }
        });
    }

    // TODO method ini digunakan untuk menentukan jumlah data yang
    // ditampilkan dalam recyclerview
    @Override
    public int getItemCount() { return list.size(); }
}

```

```

public void removeItemAt(int index){
    list.remove(index);
    notifyItemRemoved(index);
    notifyItemRangeRemoved(index, list.size());
    notifyDataSetChanged();
}

public void removeItem(String key){
    for (int i = 0; i < list.size(); i++){
        if (list.get(i).key.equals(key)){
            removeItemAt(i);
            break;
        }
    }
}

public void setItem(String key, PengajuanCutiData data){
    for (int i = 0; i < list.size(); i++){
        if (list.get(i).key.equals(key)){
            list.set(i, data);
            notifyDataSetChanged();
            break;
        }
    }
}

public void setList(ArrayList<PengajuanCutiData> data){
    list = new ArrayList<>();
    list.addAll(data);
    notifyDataSetChanged();
}

public interface itemClick{
    void onItemClicked(PengajuanCutiData model, int index);
}

class viewHolder extends RecyclerView.ViewHolder{
    TextView nama, status, jenis;
    viewHolder(View itemView) {
        super(itemView);
        nama = itemView.findViewById(R.id.tvidnama);
        status = itemView.findViewById(R.id.tvsttscuti);
        jenis = itemView.findViewById(R.id.tvjnscuti);
    }
}
}

```

5. Data Karyawan

```

package com.example.aplikasicutikaryawan;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.os.Bundle;

```

```

import android.view.View;
import android.widget.AdapterView;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.Toast;
import com.example.aplikasicutikaryawan.activities.login.LOGIN;
import com.example.aplikasicutikaryawan.model.UserData;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebaseio.database.DatabaseReference;
import com.google.firebaseio.database.FirebaseDatabase;

public class detail_rvkaryawan extends AppCompatActivity {
    EditText Nama, NIPP, NIK, NomorTelepon, GelarAkademik, Usia, Agama,
    AlamatAsal, AlamatDomisili, TempatTanggalLahir, Email, SocialMedia,
    password;
    Spinner tipe, Nama_Divisi, Jenis_Kelamin;
    DatabaseReference dataref;
    UserData current;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_detail_rvkaryawan);
        Nama = findViewById(R.id.etnama);
        TempatTanggalLahir = findViewById(R.id.etttl);
        Jenis_Kelamin = findViewById(R.id.jenisKelamin);
        NIPP = findViewById(R.id.etnipp);
        NIK = findViewById(R.id.etnik);
        tipe = findViewById(R.id.tipe);
        Nama_Divisi = findViewById(R.id>Nama_Divisi);
        NomorTelepon = findViewById(R.id.etnomor);
        GelarAkademik = findViewById(R.id.etgelar);
        Usia = findViewById(R.id.etusia);
        Agama = findViewById(R.id.etagama);
        AlamatAsal = findViewById(R.id.etalamat_asal);
        AlamatDomisili = findViewById(R.id.etalamat_domisili);
        Email = findViewById(R.id.ete_email);
        SocialMedia = findViewById(R.id.etsosmed);
        password = findViewById(R.id.etpw);

        final String[] arrayTipe =
        getResources().getStringArray(R.array.Tipe);

        tipe.setOnItemSelectedListener(new
        AdapterView.OnItemSelectedListener() {

            @Override
            public void onItemSelected(AdapterView<?> adapterView, View
            view, int i, long l) {
                if (arrayTipe[i].equals("Karyawan") ||
                arrayTipe[i].equals("Supervisor") || arrayTipe[i].equals("Manager")) {
                    Nama_Divisi.setVisibility(View.VISIBLE);
                } else {
                    Nama_Divisi.setVisibility(View.GONE);
                }
            }

            public void onNothingSelected(AdapterView<?> adapterView) {
            }
        });
    }
}

```

```

dataref = FirebaseDatabase.getInstance().getReference();

if (getIntent().getBooleanExtra("view", false)){
    Nama.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("username", ""));
    Nama.setFocusable(false);

    TempatTanggalLahir.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("Tempattanggallahir", ""));
    TempatTanggalLahir.setFocusable(false);
    setSpinText(Jenis_Kelamin,
    getSharedPreferences("preferences",
    MODE_PRIVATE).getString("JenisKelamin", ""));
    Jenis_Kelamin.setFocusable(false);
    NIPP.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("NIPP", ""));
    NIPP.setFocusable(false);
    NIK.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("NIK", ""));
    NIK.setFocusable(false);
    setSpinText(tipe, getSharedPreferences("preferences",
    MODE_PRIVATE).getString("usertipe", ""));
    tipe.setFocusable(false);
    setSpinText(Nama_Divisi, getSharedPreferences("preferences",
    MODE_PRIVATE).getString("userdivisi", ""));
    Nama_Divisi.setFocusable(false);
    NomorTelepon.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("NomorTelepon", ""));
    NomorTelepon.setFocusable(false);
    GelarAkademik.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("GelarAkademik", ""));
    GelarAkademik.setFocusable(false);
    Usia.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("Usia", ""));
    Usia.setFocusable(false);
    Agama.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("Agama", ""));
    Agama.setFocusable(false);
    AlamatAsal.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("AlamatAsal", ""));
    AlamatAsal.setFocusable(false);
    AlamatDomisili.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("AlamatDomisili", ""));
    AlamatDomisili.setFocusable(false);
    Email.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("Email", ""));
    Email.setFocusable(false);
    SocialMedia.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("Socialmedia", ""));
    SocialMedia.setFocusable(false);
    password.setText(getSharedPreferences("preferences",
    MODE_PRIVATE).getString("userpassword", ""));
    password.setFocusable(false);
    (findViewById(R.id.btnsimpan)).setVisibility(View.GONE);
} else if (getIntent().getParcelableExtra("data")!=null){
    current = getIntent().getParcelableExtra("data");
    Nama.setText(current.Nama);
    TempatTanggalLahir.setText(current.TempatTanggalLahir);
    setSpinText(Jenis_Kelamin, current.JenisKelamin);
    NIPP.setText(current.NIPP);
    NIK.setText(current.NIK);
    setSpinText(tipe, current.tipe);
}

```

```

        setSpinText(Nama_Divisi, current.divisi);
        NomorTelepon.setText(current.NomorTelepon);
        GelarAkademik.setText(current.Gelar);
        Usia.setText(current.Usia);
        Agama.setText(current.Agama);
        AlamatAsal.setText(current.AlamatAsal);
        AlamatDomisili.setText(current.AlamatDomisili);
        Email.setText(current.Email);
        SocialMedia.setText(current.Socialmedia);
        password.setText(current.password);
        (findViewById(R.id.btnsimpan)).setVisibility(View.GONE);
        (findViewById(R.id.btnubah)).setVisibility(View.VISIBLE);
        (findViewById(R.id.btnhapus)).setVisibility(View.VISIBLE);
    }
}

public void SimpanDaftar(View view) {
    final String input_Nama = Nama.getText().toString();
    final String input_TempatTanggalLahir =
TempatTanggalLahir.getText().toString();
    final String input_JenisKelamin =
Jenis_Kelamin.getSelectedItem().toString();
    final String input_NIPP = NIPP.getText().toString();
    final String input_NIK = NIK.getText().toString();
    final String input_Jabatan = tipe.getSelectedItem().toString();
    final String input_Divisi =
Nama_Divisi.getSelectedItem().toString();
    final String input_NomorTelepon =
NomorTelepon.getText().toString();
    final String input_Gelar = GelarAkademik.getText().toString();
    final String input_Usia = Usia.getText().toString();
    final String input_Agama = Agama.getText().toString();
    final String input_AlamatAsal = AlamatAsal.getText().toString();
    final String input_AlamatDomisili =
AlamatDomisili.getText().toString();
    final String input_Email = Email.getText().toString();
    final String input_SocialMedia =
SocialMedia.getText().toString();
    final String input_password = password.getText().toString();

    dataref.child("User").push().setValue(new UserData(input_Nama,
input_TempatTanggalLahir, input_JenisKelamin, input_NIPP, input_NIK,
input_Jabatan, input_Divisi, input_NomorTelepon, input_Gelar, input_Usia,
input_Agama, input_AlamatAsal, input_AlamatDomisili, input_Email,
input_SocialMedia, input_password))
        .addOnSuccessListener(new OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                Intent bukaHome = new
Intent(detail_rvkaryawan.this, detail_rvkaryawan.class);
                startActivity(bukaHome);
                finish();
            }
        });
}

public void setSpinText(Spinner spin, String text) {
    for(int i= 0; i < spin.getAdapter().getCount(); i++) {
        if(spin.getAdapter().getItem(i).toString().contains(text)) {

```

```

        spin.setSelection(i);
    }
}

public void hapus(View view) {
    dataref.child(current.key).removeValue().addOnSuccessListener(new
OnSuccessListener<Void>() {
    @Override
    public void onSuccess(Void aVoid) {
        Intent back = new Intent();
        back.putExtra("index", getIntent().getIntExtra("index",
0));
        setResult(101, back);
        finish();
    }
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {
        Toast.makeText(detail_rvkaryawan.this,
e.getLocalizedMessage(), Toast.LENGTH_SHORT).show();
    }
});
}

public void edit(View view) {
    final String input_Nama = Nama.getText().toString();
    final String input_TempatTanggalLahir =
TempatTanggalLahir.getText().toString();
    final String input_JenisKelamin =
Jenis_Kelamin.getSelectedItem().toString();
    final String input_NIPP = NIPP.getText().toString();
    final String input_NIK = NIK.getText().toString();
    final String input_Jabatan = tipe.getSelectedItem().toString();
    final String input_Divisi =
Nama_Divisi.getSelectedItem().toString();
    final String input_NomorTelepon =
NomorTelepon.getText().toString();
    final String input_Gelar = GelarAkademik.getText().toString();
    final String input_Usia = Usia.getText().toString();
    final String input_Agama = Agama.getText().toString();
    final String input_AlamatAsal = AlamatAsal.getText().toString();
    final String input_AlamatDomisili =
AlamatDomisili.getText().toString();
    final String input_Email = Email.getText().toString();
    final String input_SocialMedia =
SocialMedia.getText().toString();
    final String input_password = password.getText().toString();

    final UserData newData = new UserData(input_Nama,
input_TempatTanggalLahir, input_JenisKelamin, input_NIPP, input_NIK,
input_Jabatan, input_Divisi, input_NomorTelepon, input_Gelar, input_Usia,
input_Agama, input_AlamatAsal, input_AlamatDomisili, input_Email,
input_SocialMedia, input_password);

    dataref.child("User").child(current.key).setValue(newData)
        .addOnSuccessListener(new OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                Intent back = new Intent();
                back.putExtra("index",
getIntent().getIntExtra("index", 0));
            }
        });
}
}

```

```
        back.putExtra("data", newData);
        setResult(100, back);
        finish();
    }
}
```

B. Tampilan Program

1. Halaman Login



2. Halaman Awal Karyawan



3. Halaman Pengajuan Cuti

FORM PENGAJUAN	
Nama	

NRP	
669507676768	
Pangkat/Golongan	

Jabatan	
Karyawati	
Divisi	
Hukum	
Satuan Organisasi	
Defence Organisasi	

FORM PENGAJUAN

Jenis Cuti

Cuti Sakit

Alasan Cuti

DAFTAR PEGAWAI

Klasifikasi Cuti

Alasan Edakma Cuti

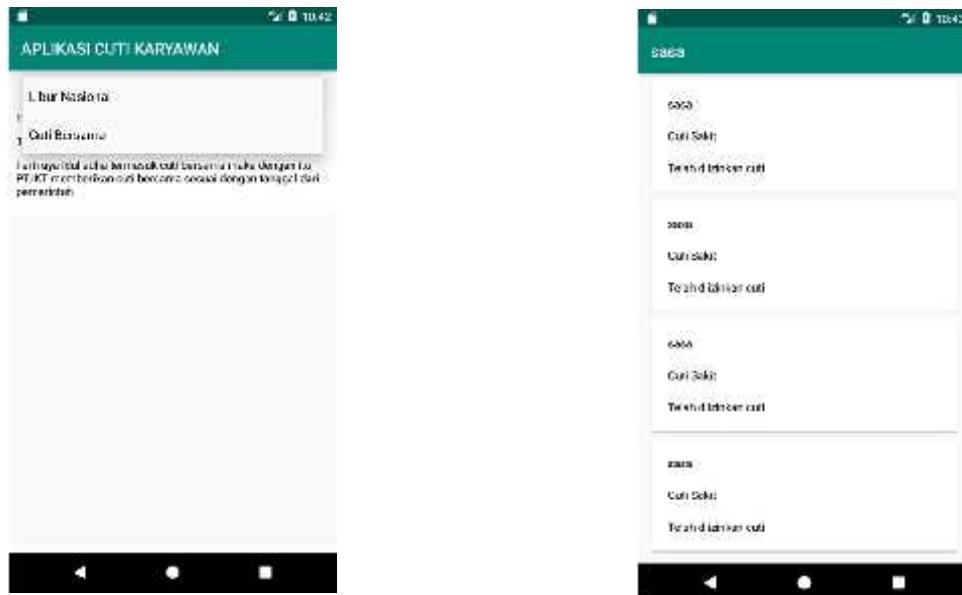
Alasan Edakma Cuti

Tanggal Mulai

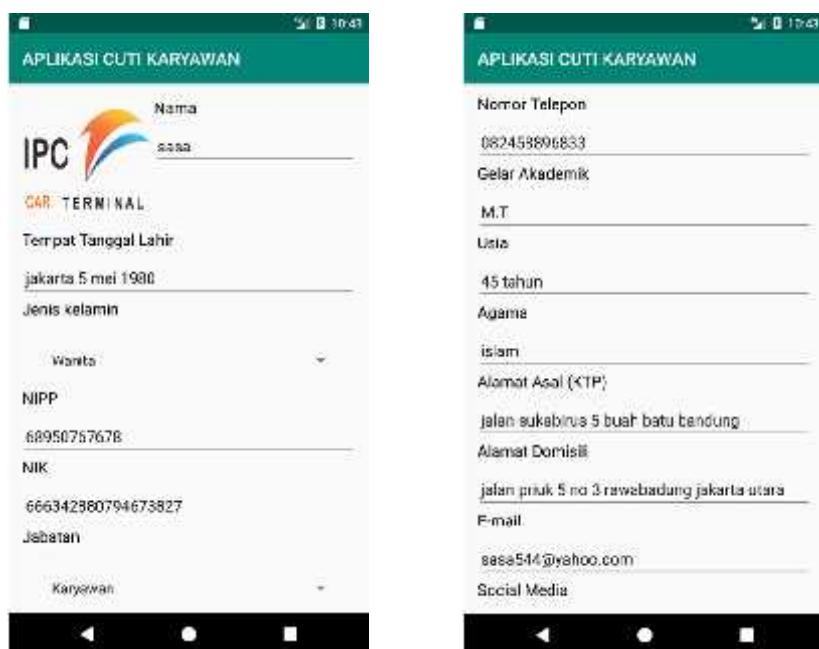
DAFTAR PEGAWAI

FORM PENGAJUAN
Alasan Cuti
Alamat Selama Cuti
Alamat Selama Cuti
Tanggal Mulai
Tanggal Mulai Cuti
Tanggal Akhir
Tanggal Akhir Cuti
Total Hari
Total Hari Cuti
SUBMIT

4. Halaman Informasi pada Karyawan 5. Halaman Riwayat Cuti pada Karyawan



6. Halaman Data Karyawan pada Karyawan



7. Halaman Awal Supervisor



8. Halaman Validasi Supervisor



9. Halaman Awal Manager



10. Halaman Validasi Manager



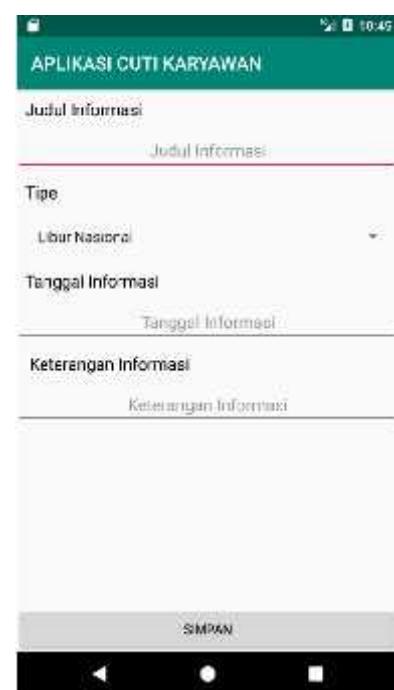
11. Tampilan Alasan Tolak Pada Manager 12. Halaman Awal Admin



13. Halaman Informasi pada Admin



14. Halaman Tambah Data Informasi



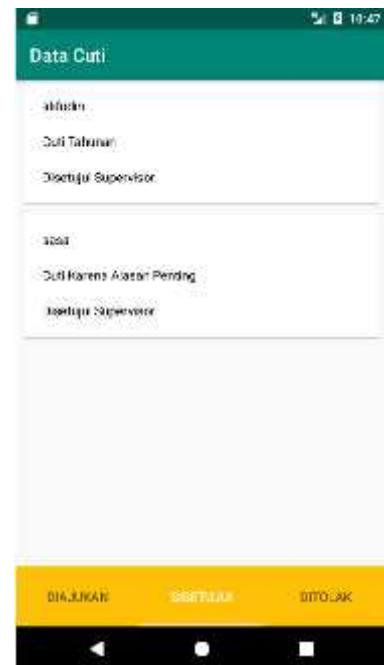
15. Halaman Ubah dan Hapus Data Informasi 16. Halaman Data Karyawan

17. Halaman Tambah Data Karyawan

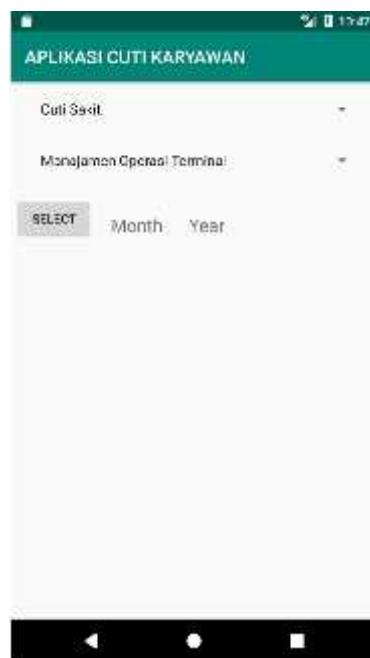
18. Halaman Ubah dan Hapus Data Karyawan



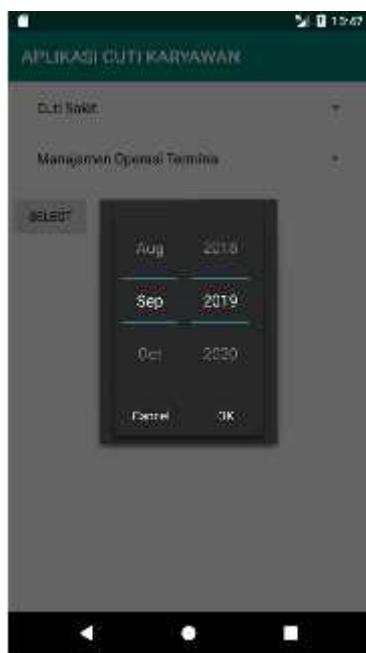
19. Halaman Data Cuti



20. Halaman Laporan Cuti pada Admin dan Halaman Awal Direktur



21. Tampilan Pilihan Bulan dan Tahun



22. Halaman Detail laporan Cuti



LAMPIRAN C

PENGUJIAN DENGAN *BLACK BOX TESTING*

1. *Form Login*

Deskripsi : Menguji fungsi login pada *Form Login*

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol <i>Login</i> tanpa memasukkan <i>username</i> dan <i>password</i>	Sistem akan menampilkan pesan kolom <i>username</i> dan <i>password</i> harus diisi	Sistem menampilkan pesan dimana <i>user</i> harus mengisi kolom <i>username</i> dan <i>password</i>	Valid
2	Memasukkan <i>username</i> dengan benar sedangkan <i>password</i> salah atau sebaliknya, lalu mengklik tombol <i>Login</i>	Sistem akan menampilkan pesan kombinasi <i>username</i> dan <i>password</i> salah	Sistem menampilkan pesan kombinasi <i>username</i> dan <i>password</i> salah	Valid
3	Memasukkan <i>username</i> dan <i>password</i> dengan benar lalu mengklik tombol <i>Login</i>	Sistem akan menerima akses <i>login</i> dan menampilkan halaman utama	Sistem menerima akses <i>login</i> dan menampilkan halaman awal	Valid

2. Halaman Awal Karyawan

Deskripsi : Menguji fungsi tambah data pengajuan pada halaman awal karyawan

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Tambah Data Pengajuan	Sistem akan menampilkan form Tambah Data Pengajuan Cuti	Sistem menampilkan form Tambah Data Pengajuan Cuti	Valid

Deskripsi : Menguji fungsi ubah pada halaman pengajuan karyawan
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Ubah	Sistem akan menampilkan <i>form</i> Ubah Data Pengajuan Cuti	Sistem menampilkan <i>form</i> Ubah Data Pengajuan Cuti	Valid

Deskripsi : Menguji fungsi hapus pada Menu Data *Master* Karyawan
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Hapus pada data yang dipilih	Sistem akan menghapus data yang dipilih	Sistem menghapus data yang dipilih	Valid

3. *Form* Tambah Data Pengajuan Cuti Karyawan

Deskripsi : Menguji fungsi submit pada *Form* Tambah Data Pengajuan Cuti Karyawan

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Submit dengan mengosongkan semua kolom	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
2	Mengklik tombol Submit dengan kondisi salah satu kolom tidak diisi	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
3	Mengisi seluruh kolom isian lalu mengklik tombol Submit	Data dapat disimpan	Data dapat disimpan	Valid

4. *Form* Ubah Data Pengajuan Cuti Karyawan

Deskripsi : Menguji fungsi ubah pada *Form* Ubah Data Pengajuan Cuti Karyawan
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Ubah dengan kondisi semua tidak ada yg diubah	Data tidak ada yang berubah	Data tidak ada yang berubah	Valid

2	Mengklik tombol Ubah dengan kondisi salah satu kolom diganti	Data dapat diubah	Data dapat diubah	Valid
3	Mengganti seluruh kolom isian lalu mengklik tombol Ubah	Data dapat diubah	Data dapat diubah	Valid

5. Halaman Awal Supervisor

Deskripsi : Menguji fungsi list data pengajuan pada halaman awal supervisor
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik list data pengajuan	Sistem akan menampilkan <i>form</i> pengajuan cuti karyawan	Sistem menampilkan <i>form</i> pengajuan cuti karyawan	Valid

Deskripsi : Menguji fungsi setuju pada halaman pengajuan supervisor
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Setuju	Sistem akan menyimpan data setuju dan mengubah status perizinan cuti karyawan	Sistem menyimpan data setuju dan mengubah status perizinan cuti karyawan	Valid

6. Halaman Awal Manager

Deskripsi : Menguji fungsi list data pengajuan pada halaman awal manager
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik list data pengajuan	Sistem akan menampilkan <i>form</i> pengajuan cuti karyawan	Sistem menampilkan <i>form</i> pengajuan cuti karyawan	Valid

Deskripsi : Menguji fungsi setuju dan tolak pada halaman pengajuan supervisor
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Setuju	Sistem akan menyimpan data setuju dan mengubah status perizinan cuti karyawan	Sistem menyimpan data setuju dan mengubah status perizinan cuti karyawan	Valid
2	Mengklik tombol Tolak	Sistem akan menampilkan pilihan kata untuk alasan penolakan	Sistem menampilkan pilihan kata untuk alasan penolakan	Valid

7. Halaman Awal Direktur

Deskripsi : Menguji fungsi *select* pada halaman awal Direktur
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol <i>Select</i> lalu pilih tanggal dan tahun	Sistem akan menampilkan list data sesuai tanggal dan tahun yang dipilih	Sistem menampilkan list data sesuai tanggal dan tahun yang dipilih	Valid

Deskripsi : Menguji fungsi list data pada halaman awal direktur
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik list data	Sistem akan menampilkan <i>form</i> laporan cuti karyawan	Sistem menampilkan <i>form</i> laporan cuti karyawan	Valid

8. Menu Informasi pada Halaman Awal Admin

Deskripsi : Menguji fungsi tambah data pada Menu Informasi
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Tambah Data Informasi	Sistem akan menampilkan <i>form</i> Tambah Data Informasi	Sistem menampilkan <i>form</i> Tambah Data Informasi	Valid

Deskripsi : Menguji fungsi ubah pada Menu Data Informasi
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Ubah	Sistem akan menampilkan <i>form</i> Ubah Data Informasi	Sistem menampilkan <i>form</i> Ubah Data Informasi	Valid

Deskripsi : Menguji fungsi hapus pada Menu Data *Master* Subkriteria
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Hapus	Sistem akan menghapus data	Sistem menghapus data	Valid

9. Form Tambah Data Informasi

Deskripsi : Menguji fungsi simpan pada Form Tambah Data *Master* Subkriteria
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Simpan dengan mengosongkan semua kolom	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
2	Mengklik tombol Simpan dengan kondisi salah satu kolom tidak diisi	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
3	Mengisi seluruh kolom isian lalu mengklik tombol Simpan	Data dapat disimpan	Data dapat disimpan	Valid

10. Form Ubah Data *Master* Subkriteria

Deskripsi : Menguji fungsi ubah pada Form Ubah Data Informasi
 Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Ubah dengan kondisi semua tidak ada yg diubah	Data tidak ada yang berubah	Data tidak ada yang berubah	Valid

2	Mengklik tombol Ubah dengan kondisi salah satu kolom diganti	Data dapat diubah	Data dapat diubah	Valid
3	Mengganti seluruh kolom isian lalu mengklik tombol Ubah	Data dapat diubah	Data dapat diubah	Valid

11. Menu Data Karyawan pada Halaman Awal Admin

Deskripsi : Menguji fungsi tambah data pada Menu Data Karyawan

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Tambah Data	Sistem akan menampilkan form Tambah Data Karyawan	Sistem menampilkan form Tambah Data Karyawan	Valid

Deskripsi : Menguji fungsi ubah pada Menu Data Master Periode

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Ubah	Sistem akan menampilkan form Ubah Data Karyawan	Sistem menampilkan form Ubah Data Karyawan	Valid

Deskripsi : Menguji fungsi hapus pada Menu Data Master Periode

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Hapus	Sistem akan menghapus data	Sistem menghapus data	Valid

12. Form Tambah Data Karyawan

Deskripsi : Menguji fungsi simpan pada Form Tambah Data Karyawan

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Simpan dengan mengosongkan semua kolom	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid

2	Mengklik tombol Simpan dengan kondisi salah satu kolom tidak diisi	Data tidak dapat disimpan	Data tidak dapat disimpan	Valid
3	Mengisi seluruh kolom isian lalu mengklik tombol Simpan	Data dapat disimpan	Data dapat disimpan	Valid

13. Form Ubah Data Karyawan

Deskripsi : Menguji fungsi ubah pada Form Ubah Data Karyawan

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol Ubah dengan kondisi semua tidak ada yg diubah	Data tidak ada yang berubah	Data tidak ada yang berubah	Valid
2	Mengklik tombol Ubah dengan kondisi salah satu kolom diganti	Data dapat diubah	Data dapat diubah	Valid
3	Mengganti seluruh kolom isian lalu mengklik tombol Ubah	Data dapat diubah	Data dapat diubah	Valid

14. Menu Data Cuti pada Halaman Awal Admin

Deskripsi : Menguji fungsi Toolbar pada Menu Data Cuti

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik Toolbar disetujui	Sistem akan menampilkan data pengajuan yang telah disetujui	Sistem menampilkan data pengajuan yang telah disetujui	Valid
2	Mengklik Toolbar diajukan	Sistem akan menampilkan data pengajuan yang diajukan	Sistem akan menampilkan data pengajuan yang diajukan	Valid

3	Mengklik <i>Toolbar</i> ditolak	Sistem akan menampilkan data pengajuan yang ditolak	Sistem akan menampilkan data pengajuan yang ditolak	Valid
---	---------------------------------	---	---	-------

15. Menu Data Laporan Cuti pada Halaman Awal Admin

Deskripsi : Menguji fungsi *select* pada Menu Laporan Cuti

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik tombol <i>Select</i> lalu pilih tanggal dan tahun	Sistem akan menampilkan data laporan cuti sesuai tanggal dan tahun yang dipilih	Sistem menampilkan data laporan cuti sesuai tanggal dan tahun yang dipilih	Valid

Deskripsi : Menguji fungsi list data pada Menu Laporan Cuti

Penguji : Anisya Agustia (1719098)

No.	Test Case	Expected Result	Actual Result	Result
1	Mengklik List Data	Sistem akan Menampilkln detail laporan cuti sesuai data yang dipilih	Sistem Menampilkln detail laporan cuti sesuai data yang dipilih	Valid