

**Sistem Pendukung Keputusan Penilaian Kinerja *Helpdesk Data*
Management Di PT. Telkom Witel Bandung
Menggunakan Metode *TOPSIS***

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh kelulusan
Jenjang Strata Satu (S1)
Pada Program Studi Sistem Informasi**

Oleh

Prasetya Adhytiatama Saputra

351761001



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
INDONESIA MANDIRI**

2020

**Sistem Pendukung Keputusan Penilaian Kinerja *Helpdesk Data*
Management Di PT. Telkom Witel Bandung
Menggunakan Metode *TOPSIS***

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh kelulusan
Jenjang Strata Satu (S1)
Pada Program Studi Sistem Informasi**

Oleh

Prasetya Adhytiatama Saputra

351761001



**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
INDONESIA MANDIRI**

2020

LEMBAR PENGESAHAN

SISTEM PENDUKUNG KEPUTUSAN PENILAIAN KINERJA HELPDESK DATA MANAGEMENT DI PT. TELKOM WITEL BANDUNG MENGGUNAKAN METODE *TOPSIS*

Oleh :

PRASETYA ADHYTIATAMA SAPUTRA

351761001

Skripsi ini telah diterima dan disahkan untuk
Memenuhi persyaratan mencapai gelar

SARJANA SISTEM INFORMASI

Pada

**PROGRAM STUDI SISTEM INFORMASI
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
INDONESIA MANDIRI**

Bandung, Juni 2020

Disahkan oleh

Ketua Program Studi,

Dosen Pembimbing,

Moch. Ali Ramdhani, S.T., M.Kom
NIDN. 0403097701

Hendra Gunawan, S.T., M.Kom.
NIDN. 0423037202

LEMBAR PERSETUJUAN REVISI

SISTEM PENDUKUNG KEPUTUSAN PENILAIAN KINERJA HELPDESK DATA MANAGEMENT DI PT. TELKOM WITEL BANDUNG MENGGUNAKAN METODE *TOPSIS*

Oleh :

PRASETYA ADHYTIATAMA SAPUTRA

351761001

Telah melakukan sidang skripsi dan telah melakukan revisi sesuai dengan perubahan dan perbaikan yang diminta pada saat sidang skripsi.

Bandung, Juni 2020

Menyetujui

No.	Nama Dosen	Keterangan	Tanda Tangan
1.	Hendra Gunawan, S.T., M.Kom.	Pembimbing	
2.	Chairuddin, Ir., M.T., M.M., Dr.	Penguji 1	
3.	Haryoso Wicaksono, S.SI., M.M., M.Kom.	Penguji 2	

Mengetahui,

Ketua Program Studi Sistem Informasi

Moch. Ali Ramdhani, S.T., M.Kom

NIDN. 0403097701

SURAT PERNYATAAN

Dengan ini saya menyatakan bahwa :

- (1) Naskah skripsi adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri maupun perguruan tinggi lainnya.
- (2) Skripsi ini murni merupakan karya penelitian saya sendiri dan tidak menjiplak karya pihak lain. Dalam hal ada bantuan atau arahan dari pihak lain maka telah saya sebutkan identitas dan jenis bantuannya di dalam lembar ucapan terima kasih.
- (3) Seandainya ada karya pihak lain yang ternyata memiliki kemiripan dengan karya saya ini, maka hal ini adalah di luar pengetahuan saya dan terjadi tanpa kesengajaan dari pihak saya.

Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terbukti adanya kebohongan dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai norma yang berlaku di Sekolah Tinggi Manajemen Informatika dan Komputer Indonesia Mandiri.

Bandung, Juni 2020

Yang membuat pernyataan

Prasetya Adhyatama Saputra

351761001

ABSTRAK

SISTEM PENDUKUNG KEPUTUSAN PENILAIAN KINERJA HELPDESK DATA MANAGEMENT DI PT. TELKOM WITEL BANDUNG MENGGUNAKAN METODE *TOPSIS*

Oleh :

Prasetya Adhyiatama Saputra
351761001

PT. Telkom adalah salah satu perusahaan yang memanfaatkan teknologi informasi dan internet secara bersamaan dengan sangat baik, bahkan perusahaan ini merupakan perusahaan terbesar yang bergerak di bidang telekomunikasi. Dalam menjalankan proses bisnisnya, PT. Telkom memanfaatkan teknologi informasi yang ada pada saat ini. Teknologi informasi yang dimanfaatkan nya antara lain; website dan internet. Akan tetapi, dalam kegiatan penilaian untuk karyawan masih belum menggunakan teknologi informasi apapun. Oleh karena itu, penulis mengambil skripsi berjudul “Sistem Pendukung Keputusan Penilaian Kinerja *Helpdesk Data Management* Di PT. Telkom Witel Bandung Menggunakan Metode *TOPSIS*”. Metode yang digunakan dalam pembuatan aplikasi penilaian karyawan ini adalah *TOPSIS*. Menurut Nofriansyah (2015), *TOPSIS* adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yoon dan Hwang pada tahun 1981. *TOPSIS* menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan jarak terpanjang (terjauh) dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak Euclidean (jarak antara dua titik) untuk menentukan kedekatan relatif dari suatu alternatif. Berdasarkan dari hasil penulisan yang dilakukan, dapat diperoleh kesimpulan bahwa belum adanya aplikasi untuk penilaian kinerja karyawan di PT. Telkom pada bagian *helpdesk data management*. Maka dari itu penulis bertujuan untuk membuat aplikasi penilaian kinerja karyawan di PT. Telkom pada bagian *helpdesk data management*. Hasil dari penulisan yang dilakukan oleh penulis menghasilkan sebuah sistem pendukung keputusan yang nantinya akan membantu pihak perusahaan dalam menentukan karyawan terbaik pada divisi *helpdesk data management*.

Kata kunci : sistem pendukung keputusan, *topsis*, *helpdesk*, *data management*.

ABSTRACT

DECISION SUPPORT SYSTEM FOR PERFORMANCE ASSESSMENT OF HELPDESK DATA MANAGEMENT AT PT. TELKOM WITEL BANDUNG USES THE TOPSIS METHOD

By:

Prasetya Adhytiatama Saputra

351761001

PT. Telkom is one company that utilizes information technology and the internet together well, even this company is the largest company engaged in telecommunications. In carrying out its business processes, PT. Telkom utilizes existing information technology. Information technology that is utilized, among others; website and internet. However, in activities that are new to the company they still do not use any information technology. Therefore, the authors take a thesis entitled "Decision Support System Performance Evaluation Data Management Helpdesk at PT. Telkom Witel Bandung Using the TOPSIS Method ". The method used in making this company application is TOPSIS. According to Nofriansyah (2015), TOPSIS is one of the methods that took multi-criteria decisions that were first approved by Yoon and Hwang in 1981. TOPSIS uses the principle that replaces those who must submit an application to find a positive solution and the longest distance (farthest) from the ideal solution geometrically by using Euclidean distance (distance between two points) to determine the relative proximity of alternatives. Based on the results done, it can be concluded that there is no application for employee performance appraisal at PT. Telkom in the data management helpdesk section. Therefore the author to make the announcement application at PT. Telkom in the data management helpdesk section. The results of the authors conducted a system that supports decisions that will help the company in determining the best employees in the data management helpdesk division.

Keywords: decision support systems, topsis, helpdesk, data management.

KATA PENGANTAR

Puji dan syukur kita panjatkan kehadirat Allah SWT. yang telah memberikan penulis rahmat dan karunia untuk menyelesaikan skripsi yang berjudul **“Sistem Pendukung Keputusan Penilaian Kinerja *Helpdesk Data Management* Di PT. Telkom Witel Bandung Menggunakan Metode *TOPSIS*”**. Penyusunan skripsi ini bertujuan untuk memenuhi persyaratan kelulusan pada Program Studi Sistem Informasi di STMIK-IM Bandung.

Tentunya dalam proses penyusunan skripsi ini, penulis mengalami beberapa kesulitan, baik dari situasi, kemampuan dan waktu yang dimiliki. Dikarenakan keterbatasan kemampuan yang dimiliki, penulis sangat mengapresiasi dan sangat berterima kasih kepada pihak-pihak yang membantu penulis dalam melancarkan skripsi ini. Bantuan yang penulis dapatkan dari pihak-pihak tersebut bermacam-macam, dimulai dari artikel internet, buku-buku, jurnal yang di perpustakaan maupun internet, dan *referensi* program yang ada.

Penulis sangatlah menyadari bahwa skripsi ini masih memiliki banyak kekurangan, serta masih jauh dari kata sempurna. Oleh karena itu, penulis sangatlah mengharapkan kritik dan saran yang dapat membangun penulis, guna pembelajaran di kemudian hari.

Penulis sangat berharap, semoga skripsi ini dapat memberikan wawasan lebih, serta manfaat untuk pembaca khususnya untuk adik tingkat yang rekan-rekan yang mencari *referensi* untuk menyusun skripsi dikemudian hari.

Bandung, Juni 2020

Prasetya Adhytiatama Saputra

UCAPAN TERIMA KASIH

Pada kesempatan kali ini, penulis menyampaikan rasa berterima kasih kepada rekan – rekan, pembimbing, serta pihak yang ikut serta membantu penulis dalam penyusunan skripsi ini, antara lain :

1. Bapak Hendra Gunawan, S.T., M.Kom. selaku pembimbing yang telah memberikan penulis ilmu dan masukan dalam penyusunan skripsi.
2. PT. Telkom yang telah mengizinkan penulis untuk melakukan penelitian guna kelancaran penyusunan skripsi.
3. Bapak Dr. Chairuddin, Ir., M.M., M.T. selaku Ketua STMIK-IM Bandung yang telah memberikan ilmu kepada penulis.
4. Bapak Moch. Ali Ramdhani, S.T., M.Kom. selaku Ketua Program Studi Sistem Informasi.
5. Kedua orang tua penulis, Ibu dan Bapak yang selalu memberikan dukungan penuh untuk penulis.
6. Bapak & Ibu dosen yang telah memberikan penulis ilmu selama penulis melakukan kegiatan belajar mengajar di STMIK-IM Bandung.
7. Rekan – rekan di STMIK-IM Bandung maupun di PT. Telekomunikasi Indonesia.
8. Ahmad Code yang senantiasa memberikan ilmu serta telah memberikan template sebagai referensi program yang digunakan dalam skripsi ini.
9. Alan Saputra Lengkoan yang senantiasa membagikan referensi perhitungan metode *TOPSIS* berupa *source code* yang dibagikan melalui situs *github*.

Bandung, Juni 2020

Prasetya Adhytiatama Saputra

DAFTAR ISI

LEMBAR PENGESAHAN.....	i
LEMBAR PERSETUJUAN REVISI	ii
LEMBAR PERNYATAAN	iii
ABSTRAK.....	iv
ABSTRACT.....	v
KATA PENGANTAR.....	iv
UCAPAN TERIMA KASIH.....	vii
DAFTAR ISI.....	viii
DAFTAR TABEL	xi
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Identifikasi Masalah.....	2
1.3 Tujuan Penulisan	2
1.4 Batasan Masalah	3
1.5 Metode Penelitian	3
1.5.1 Metode Pengumpulan Data	3
1.5.2 Metode Pengembangan Sistem.....	3
1.5.3 Metode Pengolahan Data.....	5
1.6 Sistematika Penulisan	6

BAB II LANDASAN TEORI

2.1 Sistem.....	7
2.1.1 Pengertian Sistem.....	7
2.1.2 Karakteristik Sistem.....	7
2.1.3 Klasifikasi Sistem	10
2.1.4 Pengertian Sistem Pendukung Keputusan	11
2.1.4.1 Metode <i>TOPSIS</i>	14
2.2 Metode Pengembangan Sistem.....	16
2.2.1 Metode <i>RAD</i>	17
2.3 Metode Pengumpulan Data	19
2.4 <i>Unified Modeling Language (UML)</i>	20
2.4.1 <i>Use Case Diagram</i>	21
2.4.2 <i>Activity Diagram</i>	22
2.4.3 <i>Sequence Diagram</i>	23
2.4.4 <i>Class Diagram</i>	24
2.5 <i>Flowchart</i>	25
2.6 Penilaian Kinerja	26
2.7 <i>Helpdesk</i>	26
2.8 <i>Data Management</i>	27
2.9 <i>Fallout</i>	28

BAB III ANALISA MASALAH DAN PERANCANGAN PROGRAM

3.1 Analisa	29
3.1.1 Analisa Masalah.....	29
3.1.2 Analisa Sistem Berjalan	29
3.1.3 Analisa Kebutuhan Non Fungsional	30
3.1.4 Analisa Kebutuhan Fungsional.....	32

3.2 Perancangan Sistem	34
3.2.1 Desain Sistem dengan <i>UML</i>	34
3.2.1.1 <i>Use Case</i> Diagram	34
3.2.1.2 <i>Activity</i> Diagram	36
3.2.1.3 <i>Sequence</i> Diagram	42
3.2.1.4 <i>Class</i> Diagram	48
3.2.2 Desain Database Sistem	49
3.2.1.1 <i>Entity Relationship Diagram (ERD)</i>	49
3.2.1.1 Struktur Tabel.....	50
3.2.3 Desain Interface Sistem.....	53
BAB IV IMPLEMENTASI DAN UJI COBA	
4.1 Implementasi	61
4.1.1 Implementasi Perangkat Keras	61
4.1.2 Implementasi Perangkat Lunak	61
4.1.3 Implementasi Database.....	62
4.1.4 Implementasi Antarmuka	65
4.2 Uji Coba	72
BAB V PENUTUP	
5.1 Kesimpulan.....	78
5.2 Saran	78
DAFTAR PUSTAKA	79
LAMPIRAN	81

DAFTAR TABEL

TABEL: 2.1 Simbol <i>Use Case</i>	21
TABEL: 2.2 Simbol <i>Activity</i>	22
TABEL: 2.3 Simbol <i>Sequence</i>	23
TABEL: 2.4 Simbol <i>Class</i>	24
TABEL: 2.5 Simbol <i>Flowchart</i>	25
TABEL: 3.1 Tabel Definisi Aktor	34
TABEL: 3.2 Tabel Definisi <i>Use Case</i>	34
TABEL: 3.3 Tabel <i>User</i>	50
TABEL: 3.4 Tabel <i>Detail User</i>	50
TABEL: 3.5 Tabel <i>Closed Fallout</i>	50
TABEL: 3.6 Tabel Absensi	51
TABEL: 3.7 Tabel Kriteria	51
TABEL: 3.8 Tabel Alternatif	51
TABEL: 3.9 Tabel Nilai	51
TABEL: 3.10 Tabel Penilaian	52
TABEL: 3.11 Tabel Hari	52
TABEL: 3.12 Tabel Tanggal	52
TABEL: 3.13 Tabel Bulan	52
TABEL: 4.1 Tabel Pengujian Login	72
TABEL: 4.2 Tabel Pengujian Absensi & Absensiku <i>Helpdesk</i>	73

TABEL: 4.3 Tabel Pengujian Jumlah & <i>Closed Fallout Helpdesk</i>	74
TABEL: 4.4 Tabel Pengujian Tambah Karyawan.....	74
TABEL: 4.5 Tabel Pengujian Daftar Karyawan	75
TABEL: 4.6 Tabel Pengujian Lihat Jumlah <i>Fallout</i> & Lihat Absen <i>Admin/Manager</i>	76
TABEL: 4.7 Tabel Pengujian Penilaian Karyawan	77

DAFTAR GAMBAR

GAMBAR: 2.1 Fase Proses Pengambilan Keputusan	30
GAMBAR: 3.1 <i>Flowchart</i> Analisa Sistem Berjalan	35
GAMBAR: 3.2 <i>Use Case Helpdesk</i>	35
GAMBAR: 3.3 <i>Use Case Admin/Manager</i>	36
GAMBAR: 3.4 <i>Activity Admin & Helpdesk Login</i>	36
GAMBAR: 3.5 <i>Activity Admin & Helpdesk Logout</i>	37
GAMBAR: 3.6 <i>Activity Helpdesk</i> melakukan absensi	37
GAMBAR: 3.7 <i>Activity Helpdesk Input Closed Fallout</i>	38
GAMBAR: 3.8 <i>Activity Helpdesk</i> Melihat Jumlah <i>Closed Fallout</i>	38
GAMBAR: 3.9 <i>Activity Helpdesk</i> Melihat Absensi	38
GAMBAR: 3.10 <i>Activity Admin/Manager</i> Menambah <i>Helpdesk/Karyawan</i>	39
GAMBAR: 3.11 <i>Activity Admin/Manager</i> Melihat Daftar Karyawan.....	39
GAMBAR: 3.12 <i>Activity Admin/Manager</i> Melihat Absensi <i>Helpdesk</i>	40
GAMBAR: 3.13 <i>Activity Admin/Manager</i> Melihat Jumlah <i>Closed Fallout</i>	40
GAMBAR: 3.14 <i>Activity Admin/Manager</i> Melakukan Penilaian <i>Helpdesk</i>	41
GAMBAR: 3.15 <i>Activity Admin/Manager</i> Mengubah Kata Sandi.....	41
GAMBAR: 3.16 <i>Sequence Admin/Manager Login</i>	42
GAMBAR: 3.17 <i>Sequence Admin/Manager Logout</i>	42
GAMBAR: 3.18 <i>Sequence Helpdesk</i> Melakukan Absensi	43

GAMBAR: 3.19 <i>Sequence Helpdesk</i> Melihat Absensi.....	43
GAMBAR: 3.20 <i>Sequence Helpdesk</i> Input & Menampilkan Jumlah <i>Closed Fallout</i>	44
GAMBAR: 3.21 <i>Sequence Admin/Manager</i> Melihat, Mengubah/Menghapus Data <i>Helpdesk</i>	44
GAMBAR: 3.22 <i>Sequence Admin/Manager</i> Menambah <i>Helpdesk</i>	45
GAMBAR: 3.23 <i>Sequence Admin/Manager</i> Melihat Absensi <i>Helpdesk</i>	45
GAMBAR: 3.24 <i>Sequence Admin/Manager</i> Melihat Jumlah <i>Closed Fallout</i>	46
GAMBAR: 3.25 <i>Sequence Admin/Manager</i> Menambah Kriteria	46
GAMBAR: 3.26 <i>Sequence Admin/Manager</i> Menambah Alternatif	47
GAMBAR: 3.27 <i>Sequence Admin/Manager</i> Menambah Nilai	47
GAMBAR: 3.28 <i>Sequence Admin/Manager</i> Melakukan Penilaian.....	48
GAMBAR: 3.29 <i>Class Diagram</i>	49
GAMBAR: 3.30 <i>ERD</i> Sistem.....	49
GAMBAR: 3.31 Desain Menu Login.....	53
GAMBAR: 3.32 Desain Menu Absensi <i>Helpdesk</i>	53
GAMBAR: 3.33 Desain Menu Absensiku <i>Helpdesk</i>	54
GAMBAR: 3.34 Desain Menu <i>Closed Fallout Helpdesk</i>	54
GAMBAR: 3.35 Desain Menu Jumlah <i>Closed Fallout Helpdesk</i>	55
GAMBAR: 3.36 Desain Menu Admin Menambah <i>Helpdesk</i>	55
GAMBAR: 3.37 Desain Menu Admin Melihat Daftar <i>Helpdesk</i>	56
GAMBAR: 3.38 Desain Menu Admin Melihat Absensi	56
GAMBAR: 3.39 Desain Menu Admin Melihat Jumlah <i>Closed Fallout</i>	57

GAMBAR: 3.40 Desain Menu Admin Menambah/Mengubah/Melihat Kriteria..	57
GAMBAR: 3.41 Desain Menu Admin Menambah/Mengubah/Melihat Alternatif	58
GAMBAR: 3.42 Desain Menu Admin Menambah/Mengubah/Melihat Nilai	58
GAMBAR: 3.43 Desain Menu Admin Memberikan Nilai	59
GAMBAR: 3.44 Desain Menu Admin Memperoleh Hasil Penilaian	59
GAMBAR: 3.45 Desain Menu Admin Mengubah Password	60
GAMBAR: 4.1 Tabel <i>User</i>	62
GAMBAR: 4.2 Tabel Detail <i>User</i>	62
GAMBAR: 4.3 Tabel <i>Closed Fallout</i>	62
GAMBAR: 4.4 Tabel Absensi	63
GAMBAR: 4.5 Tabel Kriteria.....	63
GAMBAR: 4.6 Tabel Alternatif.....	63
GAMBAR: 4.7 Tabel Nilai	63
GAMBAR: 4.8 Tabel Penilaian.....	64
GAMBAR: 4.9 Tabel Hari	64
GAMBAR: 4.10 Tabel Tanggal	64
GAMBAR: 4.11 Tabel Bulan.....	64
GAMBAR: 4.12 Halaman Login.....	65
GAMBAR: 4.13 Halaman Absensi <i>Helpdesk</i>	65
GAMBAR: 4.14 Halaman Absensiku <i>Helpdesk</i>	66
GAMBAR: 4.15 Halaman <i>Closed Fallout Helpdesk</i>	66
GAMBAR: 4.16 Halaman Jumlah <i>Closed Fallout Helpdesk</i>	67
GAMBAR: 4.17 Halaman Tambah <i>Helpdesk</i>	67
GAMBAR: 4.18 Halaman Daftar <i>Helpdesk</i>	68
GAMBAR: 4.19 Halaman Lihat Absensi <i>Helpdesk</i>	68

GAMBAR: 4.20 Halaman Jumlah <i>Closed Fallout Helpdesk</i>	69
GAMBAR: 4.21 Halaman Tambah Kriteria	69
GAMBAR: 4.22 Halaman Tambah Alternatif	70
GAMBAR: 4.23 Halaman Tambah Nilai.....	70
GAMBAR: 4.24 Halaman Beri Nilai.....	71
GAMBAR: 4.25 Halaman Hasil Penilaian	71
GAMBAR: 4.26 Halaman Ubah Katasandi	72

BAB I

PENDAHULUAN

1.1 Latar Belakang

PT. Telkom adalah salah satu perusahaan yang memanfaatkan teknologi informasi dan internet secara bersamaan dengan sangat baik, bahkan perusahaan ini merupakan perusahaan terbesar yang bergerak di bidang telekomunikasi. Pemanfaatan yang dilakukan oleh PT. Telkom ini sangatlah bermacam-macam, dimulai dari pemanfaatan perangkat telekomunikasi, pemanfaatan website dalam menjalankan proses bisnisnya, pemanfaatan akses internet, dan masih banyak lagi. Perusahaan ini juga memiliki produk dan layanan yang sudah dipercaya, dari pelayanan telepon hingga layanan internet yang ditawarkan oleh perusahaan ini.

Dalam menjalankan proses bisnisnya, PT. Telkom memanfaatkan teknologi informasi yang ada pada saat ini. Teknologi informasi yang dimanfaatkan nya antara lain; website dan internet. Salah satu teknologi informasi yang dimanfaatkan ini, digunakan untuk melakukan proses bisnis yang dinamakan validasi data teknis milik pelanggan. Dimana proses bisnis ini sangatlah bergantung terhadap teknologi informasi internet dan sistem aplikasi berbasis website. Akan tetapi, dalam kegiatan penilaian untuk karyawan masih belum menggunakan teknologi informasi apapun.

Sistem pendukung keputusan adalah salah satu aplikasi yang dapat membantu kegiatan dalam penilaian karyawan. Sistem pendukung keputusan di jadikan aplikasi berbasis web. Metode yang digunakan dalam pembuatan aplikasi

penilaian karyawan ini adalah *TOPSIS*. Menurut Nofriansyah (2015), *TOPSIS* adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yoon dan Hwang pada tahun 1981. *TOPSIS* menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan jarak terpanjang (terjauh) dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak Euclidean (jarak antara dua titik) untuk menentukan kedekatan relatif dari suatu alternatif.

Berdasarkan latar belakang di atas, penulis mengambil skripsi berjudul **“Sistem Pendukung Keputusan Penilaian Kinerja *Helpdesk Data Management* Di PT. Telkom Witel Bandung Menggunakan Metode *TOPSIS*”**.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah dipaparkan oleh penulis diatas, maka dapat di ambil kesimpulan bahwa masalah yang diperoleh dalam pembuatan skripsi ini antara lain :

- Belum adanya aplikasi untuk penilaian kinerja karyawan di PT. Telkom Witel Bandung pada bagian *Helpdesk Data Management*.

1.3 Tujuan Penulisan

Adapun tujuan yang dimiliki penulis dalam kegiatan dan pembuatan skripsi ini antara lain :

- Membuat aplikasi penilaian kinerja karyawan di PT. Telkom Witel Bandung pada bagian *Helpdesk Data Management*.

1.4 Batasan Masalah

Penulis membatasi masalah terkait pembahasan yang akan dilakukan dalam pembuatan skripsi ini yaitu :

1. Aplikasi hanya dapat di akses oleh divisi atau bagian *Data Management*.
2. Penilaian kinerja hanya dapat dilakukan oleh manager.

1.5 Metode Penelitian

1.5.1 Metode Pengumpulan Data

Metode penelitian yang dilakukan oleh penulis dalam pengumpulan data antara lain :

1. Observasi

Dengan observasi, penulis dapat mengamati secara langsung bagaimana situasi dan kondisi ruang lingkup kerja.

2. Wawancara

Dengan wawancara, penulis dapat memperoleh informasi langsung dari para staff guna memperoleh informasi yang akurat.

3. Studi Literatur

Dengan studi literatur, penulis dapat melakukan pendekatan, memperoleh referensi dan pembanding terkait metode yang telah dipilih.

1.5.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang penulis lakukan adalah metode *RAD* (*Rapid Application Development*). Menurut Putri dan Effendi (2018) *RAD*

merupakan model proses perangkat lunak yang menekankan pada daur pengembangan hidup yang sikat. *RAD* merupakan versi adaptasi cepat dari model *waterfall*, dengan menggunakan pendekatan konstruksi komponen. *RAD* merupakan gabungan dari bermacam-macam teknik terstruktur dengan teknik *prototyping* dan teknik pengembangan *joint application* untuk mempercepat pengembangan sistem/aplikasi. Dari definisi konsep ini, dapat dilihat bahwa pengembangan aplikasi dengan menggunakan metode *RAD* dapat dilakukan dalam waktu yang relative cepat. Berikut ini adalah tahap-tahap pengembangan aplikasi dari tiap-tiap fase pengembangan aplikasi :



Gambar 1.1: Tahap-tahap *RAD* (Putri dan Effendi, 2018)

Tahap-tahap *RAD* terdiri dari 3 tahap yang terstruktur dan saling bergantung disetiap tahap, yaitu :

1. *Requierements Planning* (Perencanaan Persyaratan)
 - a. Pengguna dan analisis bertemu untuk mengidentifikasi tujuan dari aplikasi atau sistem.
 - b. Berorientasi pada pemecahan masalah bisnis.

2. *Design Workshop*

- a. Fase desain dan menyempurnakan
- b. Gunakan kelompok pendukung keputusan sistem untuk membantu pengguna setuju pada desain.
- c. Programmer dan analis membangun dan menunjukkan tampilan visual desain dan alur kerja pengguna.
- d. Pengguna menanggapi prototipe kerja aktual.
- e. Analis menyempurnakan modul dirancang berdasarkan tanggapan pengguna.

3. *Implementation* (Penerapan)

- a. Sebagai sistem yang baru dibangun, sistem baru atau parsial diuji dan diperkenalkan kepada organisasi.
- b. Ketika membuat sistem baru, tidak perlu untuk menjalankan sistem lama secara parallel.

1.5.3 Metode Pengolahan Data

Metode pengolahan data yang digunakan oleh penulis adalah metode *TOPSIS*. Menurut Nofriansyah (2015), *TOPSIS* adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yoon dan Hwang pada tahun 1981. *TOPSIS* menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan jarak terpanjang (terjauh) dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak Euclidean (jarak antara dua titik) untuk menentukan kedekatan relatif dari suatu alternatif.

1.6 Sistematika Penulisan

Sistematika penulisan yang dilakukan oleh penulis dalam pembuatan skripsi ini terdiri dari beberapa bab antara lain :

BAB I PENDAHULUAN

Pada bab ini membahas penjelasan mengenai latar belakang, identifikasi masalah, tujuan penulisan, batasan masalah, metode penelitian dan sistematika penulisan yang dilakukan oleh penulis.

BAB II LANDASAN TEORI

Pada bab ini berisikan tentang landasan teori yang diperoleh oleh penulis yang dapat mendukung terhadap penulisan yang dilakukan oleh penulis.

BAB III ANALISIS MASALAH DAN PERANCANGAN PROGRAM

Pada bab ini berisi tentang perancangan program berdasarkan dengan analisa permasalahan yang dilakukan oleh penulis sesuai dengan metode yang dipilih.

BAB IV IMPLEMENTASI DAN UJI COBA

Pada bab ini berisikan tentang pembahasan terkait pengoperasian program yang telah di bangun, serta hasil evaluasi implementasi program yang telah diterapkan.

BAB V PENUTUP

Pada bab ini menjelaskan kesimpulan dari penulisan yang telah dilakukan oleh penulis, serta berisikan saran yang bertujuan untuk meningkatkan topik yang dapat digunakan kedepannya.

BAB II

LANDASAN TEORI

2.1 Sistem

2.1.1 Pengertian Sistem

Sistem berasal dari bahasa Latin (*systēma*) dan bahasa Yunani (*sustēma*) adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan. Istilah ini sering dipergunakan untuk menggambarkan suatu set entitas yang berinteraksi, di mana suatu model matematika seringkali bisa dibuat.

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi. (Jogiyanto, 2015)

2.1.2 Karakteristik Sistem

Sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut : (Tata Sutabri, 2012)

1. Komponen Sistem (*Components System*)

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

sistem tersebut dapat berupa suatu subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem (*Boundary System*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

3. Lingkungan Luar Sistem (*Environment System*)

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar tersebut harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus dikendalikan. Kalau tidak, maka akan mengganggu kelangsungan hidup dari sistem tersebut.

4. Penghubung Sistem (*Interface System*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

5. Masukan Sistem (*Input System*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk mendapatkan keluaran. Contoh, di dalam suatu unit sistem komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

6. Pengolahan Sistem (*Processing System*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

7. Keluaran Sistem (*Output System*)

Hasil energi diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi input bagi subsistem lain.

8. Sasaran Sistem (*Objective*) dan tujuan (*Goals*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministic. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem

tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.3 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi dalam sistem tersebut. Oleh karena itu, sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya. (Tata Sutabri, 2012)

1. Sistem Abstrak dan Sistem Fisik (*Abstarct System & Physical System*)

Sistem abstrak merupakan sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Misalnya sistem teologi, yaitu sistem yang berupa pemikiran-pemikiran hubungan antara manusia dengan Tuhan. Sistem fisik merupakan sistem yang ada secara fisik. Misalnya sistem komputer, sistemakuntansi, dan sistem persediaan barang.

2. Sistem Alamiah dan Sistem Buatan Manusia (*Natural System & Human Made System*)

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat manusia. Misalnya sistem perputaran bumi. Sistem buatan manusia adalah sistem yang melibatkan interaksi manusia dengan mesin yang disebut human machine system. Misalnya sistem informasi berbasis komputer.

3. Sistem Tertentu dan Sistem Tak Tentu (*Deterministic System & Probabilistic System*)

Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi. Sebagai contoh adalah hasil pertandingan sepak bola. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas. Misalnya kematian seseorang.

4. Sistem Tertutup dan Sistem Terbuka (*Closed System & Open System*)

Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa adanya campur tangan dari pihak di luarnya. Secara teoritis sistem tertutup ini ada, tetapi pada kenyataannya tidak ada sistem yang benar-benar tertutup. Contohnya adalah sistem adat masyarakat Baduy. Sedangkan sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya. Misalnya sistem musyawarah.

2.1.4 Pengertian Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) biasanya dibangun untuk mendukung solusi atas suatu masalah atau untuk suatu peluang. Aplikasi Sistem Pendukung Keputusan (SPK) digunakan dalam pengambilan keputusan. Aplikasi Sistem Pendukung Keputusan (SPK) menggunakan *CBIS (Computer Based Information Systems)* yang fleksibel, interaktif, dan dapat diadaptasi, yang dikembangkan untuk

mendukung solusi atas masalah manajemen spesifik yang tidak terstruktur. (Nofriansyah, 2015)

Dalam buku *Decision Support System And Intelligent Systems* mendefinisikan sistem pendukung keputusan sebagai sistem berbasis komputer yang terdiri dari tiga komponen yang saling berinteraksi, sistem bahasa (mekanisme untuk memberikan komunikasi antara pengguna dan komponen sistem pendukung keputusan lain), sistem pengetahuan (repositori pengetahuan domain masalah yang ada pada sistem pendukung keputusan atau sebagai data atau sebagai prosedur), dan sistem pemrosesan masalah (hubungan antara dua komponen lainnya, terdiri dari satu atau lebih kapabilitas manipulasi masalah umum yang diperlukan untuk pengambilan keputusan). (Turban, 2005)

Karakteristik dari sistem pendukung keputusan yaitu :

- a. Mendukung proses pengambilan keputusan suatu organisasi atau perusahaan
- b. Adanya *interface* manusia/mesin dimana manusia tetap memegang kontrol proses pengambilan keputusan.
- c. Mendukung pengambilan keputusan untuk membahas masalah terstruktur, semi terstruktur serta mendukung beberapa keputusan yang saling berinteraksi.
- d. Memiliki kapasitas dialog untuk memperoleh informasi sesuai dengan kebutuhan.
- e. Memiliki subsistem yang terintegrasi sedemikian rupa sehingga dapat berfungsi sebagai kesatuan sistem.
- f. Memiliki dua komponen utama yaitu data dan model.

Menurut Simon ada tiga fase dalam proses pengambilan keputusan, diantaranya sebagai berikut :

1. *Intelligence*

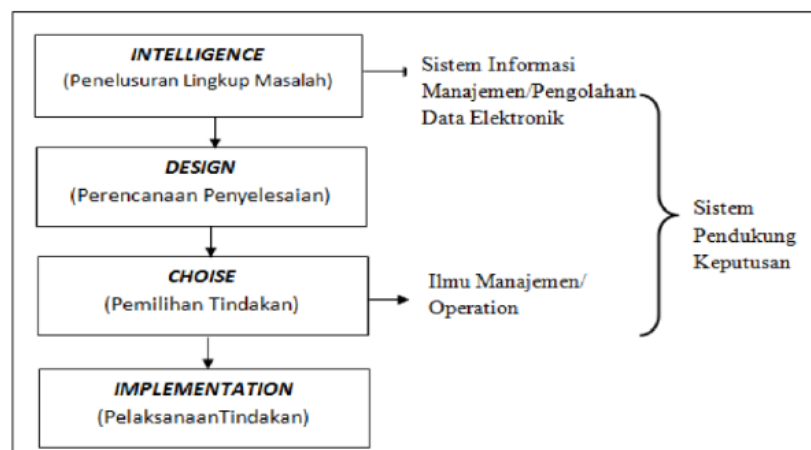
Tahap ini merupakan proses penelusuran dan pendeteksian dari ruang lingkup problematika secara proses pengenalan masalah. Data masukan diperoleh, diproses dan diuji dalam rangka mengidentifikasi masalah.

2. *Design*

Tahap ini merupakan proses menemukan, mengembangkan dan menganalisis alternatif tindakan yang bisa dilakukan. Tahap ini meliputi menguji kelayakan solusi.

3. *Choice*

Pada tahap ini dilakukan proses pemilihan diantara berbagai alternatif tindakan yang mungkin dijalankan. Hasil pemilihan tersebut kemudian di implementasikan dalam proses pengambilan keputusan.



GAMBAR: 2.1 Fase Proses Pengambilan Keputusan (Kadarsah & Ali, 1998)

2.1.4.1 Metode *TOPSIS*

TOPSIS adalah salah satu metode pengambilan keputusan multikriteria yang pertama kali diperkenalkan oleh Yoon dan Hwang pada tahun 1981. *TOPSIS* menggunakan prinsip bahwa alternatif yang terpilih harus mempunyai jarak terdekat dari solusi ideal positif dan jarak terpanjang (terjauh) dari solusi ideal negatif dari sudut pandang geometris dengan menggunakan jarak Euclidean (jarak antara dua titik) untuk menentukan kedekatan relatif dari suatu alternatif.

Solusi ideal positif didefinisikan sebagai jumlah dari seluruh nilai terbaik yang dapat dicapai untuk setiap atribut, sedangkan solusi negatif-ideal terdiri dari seluruh nilai terburuk yang dicapai untuk setiap atribut. *TOPSIS* mempertimbangkan keduanya, jarak terhadap solusi ideal negatif dengan mengambil kedekatan relatif terhadap solusi ideal positif.

Berdasarkan perbandingan terhadap jarak relatifnya, susunan prioritas alternatif bisa dicapai. Metode ini banyak digunakan untuk menyelesaikan pengambilan keputusan secara praktis. Hal ini disebabkan konsepnya sederhana dan mudah dipahami, komputasinya efisien, dan memiliki kemampuan mengukur kinerja relatif dari alternatif-alternatif keputusan. (Nofriansyah, 2015)

Keuntungan metode *TOPSIS* yaitu :

1. Metode *TOPSIS* merupakan salah satu metode yang simple dan konsep rasional yang mudah dipahami.
2. Metode *TOPSIS* mampu untuk mengukur kinerja relatif dalam bentuk form matematika sederhana.

Metode *TOPSIS* didasarkan pada konsep bahwa alternatif terpilih yang terbaik tidak hanya memiliki jarak terpendek dari solusi ideal positif tetapi juga memiliki jarak terpanjang dari solusi ideal negatif. Tahapan metode *TOPSIS* :

1. Membuat matriks keputusan yang ternormalisasi.
2. Membuat matriks keputusan yang ternormalisasi berbobot.
3. Menentukan matriks solusi ideal positif dan matriks solusi ideal negatif.
4. Menentukan jarak antara nilai setiap alternatif dengan matriks solusi ideal positif dan negatif.
5. Menentukan preferensi untuk setiap alternatif.

TOPSIS membutuhkan rating kinerja setiap alternatif A_i pada setiap kriteria C_i yang ternormalisasi, yaitu : (Sri Kusumadewi, 2006)

Adapun langkah-langkah algoritma dari metode *TOPSIS* adalah :

1. Menentukan normalisasi matriks keputusan. Nilai ternormalisasi r_{ij} dihitung dengan rumus.

$$r_{ij} = \frac{X_{ij}}{\sqrt{\sum_{i=1}^m X_{ij}^2}}$$

Keterangan : $i = 1, 2, \dots, m$; dan $j = 1, 2, \dots, n$,

2. Menentukan bobot ternormalisasi matriks keputusan. Nilai bobot ternormalisasi y_{ij} sebagai berikut :

$$y_{ij} = w_{ij} r_{ij} ; \quad A^+ = (y_1^+, y_2^+, \dots, y_n^+) \quad ; \quad A^- = (y_1^-, y_2^-, \dots, y_n^-)$$

Keterangan : $i = 1, 2, \dots, m$ dan $j = 1, 2, \dots, n$;

dengan $y_j^+ = \begin{cases} \max y_{ij} & \text{:Jika } j \text{ adalah atribut keuntungan} \\ \min y_{ij} & \text{:Jika } j \text{ adalah atribut biaya} \end{cases}$

3. Jarak alternatif A_i dengan solusi ideal positif dirumuskan sebagai :

$$D_i^+ = \sqrt{\sum_{j=1}^n (y_i^+ - y_{ij})^2};$$

Keterangan : $j = 1, 2, \dots, n$

4. Jarak alternatif A_i dengan solusi ideal negatif dirumuskan sebagai :

$$D_i^- = \sqrt{\sum_{j=1}^n (y_{ij} - y_i^-)^2};$$

Keterangan : $i = 1, 2, \dots, n$

5. Nilai prefensi untuk setiap alternatif (V_i) diberikan sebagai :

$$V_i = \frac{D_i^-}{D_i^- + D_i^+}$$

2.2 Metode Pengembangan Sistem

Sistem yang berjalan atau sedang digunakan oleh organisasi atau perusahaan akan terus dikembangkan untuk memperbaiki kekurangan-kekurangan pada sistem tersebut. Untuk melakukan pengembangan sistem, metode yang digunakan adalah *SDLC*. Metode adalah tahap-tahap ataupun aturan untuk melakukan sesuatu.

SDLC adalah sebuah metode yang digunakan untuk mengembangkan sebuah sistem. *SDLC* adalah sebuah proses logika yang digunakan oleh seorang

system analyst untuk mengembangkan sebuah sistem informasi yang melibatkan *requirements, validation, training* dan pemilik sistem.

SDLC identik dengan teknik pengembangan sistem *waterfall*, karena tahapannya menurun dari atas kebawah. Berikut tahapan *SDLC* : (Sri Mulyani, 2016)

1. *Planning*

2. *Analysis*

3. *Design*

4. *Implementation*

5. *Use*

2.2.1 Metode *Rapid Application Development (RAD)*

Sistem yang semakin kompleks dan waktu pengembangan yang dibutuhkan semakin cepat, membuat para pengembang sistem berfikir keras dan berusaha untuk mencari solusi teknik pengembangan sistem yang cepat tanpa mengurangi kualitas sistem yang dihasilkan. Dengan kondisi ini, dikembangkanlah *Rapid Application Development (RAD)*. (Sri Mulyani, 2016)

RAD merupakan model proses perangkat lunak yang menekankan pada daur pengembangan hidup yang singkat. *RAD* merupakan versi adaptasi cepat dari model *waterfall*, dengan menggunakan pendekatan konstruksi komponen. *RAD* merupakan gabungan dari bermacam-macam teknik terstruktur dengan teknik *prototyping* dan teknik pengembangan *joint application* untuk mempercepat pengembangan sistem/aplikasi. Dari definisi konsep ini, dapat dilihat bahwa

pengembangan aplikasi dengan menggunakan metode *RAD* dapat dilakukan dalam waktu yang relative cepat. Berikut ini adalah tahap-tahap pengembangan aplikasi dari tiap-tiap fase pengembangan aplikasi : (Putri dan Effendi, 2018)



Gambar 2.2: Tahap-tahap *RAD* (Putri dan Effendi, 2018)

Tahap-tahap *RAD* terdiri dari 3 tahap yang terstruktur dan saling bergantung disetiap tahap, yaitu :

1. *Requierements Planning* (Perencanaan Persyaratan)
 - a. Pengguna dan analisis bertemu untuk mengidentifikasi tujuan dari aplikasi atau sistem.
 - b. Berorientasi pada pemecahan masalah bisnis.
2. *Design Workshop*
 - a. Fase desain dan menyempurnakan
 - b. Gunakan kelompok pendukung keputusan sistem untuk membantu pengguna setuju pada desain.
 - c. Programmer dan analis membangun dan menunjukkan tampilan visual desain dan alur kerja pengguna.

- d. Pengguna menanggapi prototipe kerja aktual.
 - e. Analis menyempurnakan modul dirancang berdasarkan tanggapan pengguna.
3. *Implementation* (Penerapan)
- a. Sebagai sistem yang baru dibangun, sistem baru atau parsial diuji dan diperkenalkan kepada organisasi.
 - b. Ketika membuat sistem baru, tidak perlu untuk menjalankan sistem lama secara parallel.

Ada 4 (empat) komponen pada *RAD* yaitu : (McLeod dan Schell, 2007)

1. Manajemen, yaitu orang-orang (dari sisi *user*) yang berada pada *level* manajemen yang mempunyai yang bisa beradaptasi dengan cepat untuk menggunakan metode baru.
2. Pengembang, yaitu tim pengembang sistem professional dalam menggunakan metode-metode pengembang sistem dan *tools* yang dibutuhkan.
3. Metode, yaitu metode *RAD* yang dikenal dengan *RAD Life Cycle*.
4. *Tools*, yaitu *Computer-Aided Software Engineering (CASE)* dan *4th Generation Language* yang bisa memfasilitasi pembuat prototype dan pembuat kode program. Sedangkan *CASE Tools* lebih kepada dokumentas dan perancangan database.

2.3 Metode Pengumpulan Data

Untuk mengumpulkan data dari sampel penelitian, dilakukan dengan metode tertentu sesuai dengan tujuannya. Ada berbagai metode yang telah kita kenal antara lain wawancara, pengamatan (observasi), kuesioner atau angket, dan

documenter. Metode yang dipilih untuk setiap variabel tergantung pada berbagai faktor terutama jenis data dan ciri responden. Ada beberapa metode pengumpulan data, yaitu : (W. Gulo, 2000)

1. Pengamatan (Observasi)

Pengamatan (Observasi) adalah metode pengumpulan data dimana peneliti atau kolaboratornya mencatat informasi sebagaimana yang mereka saksikan selama penelitian. Penyaksian terhadap peristiwa-peristiwa itu bisa dengan melihat, mendengarkan, merasakan, yang kemudian dicatat seobyektif mungkin.

2. Wawancara

Wawancara adalah bentuk komunikasi langsung antara peneliti dan responden. Komunikasi berlangsung dalam bentuk tanya-jawab dalam hubungan tatap muka, sehingga gerak dan mimik responden merupakan pola media yang melengkapi kata-kata secara verbal.

3. Literatur (*Library Research*)

Dalam riset pustaka memanfaatkan sumber perpustakaan untuk memperoleh data penelitian. Tegasnya riset pustaka membatasi kegiatan hanya pada bahan-bahan koleksi perpustakaan saja tanpa memerlukan riset lapangan. (Mestika Zed, 2004)

2.4 *Unified Modeling Language (UML)*





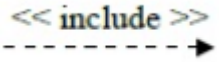
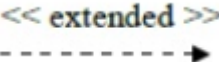
UML adalah suatu metode pemodelan secara visual untuk sarana perancangan sistem berorientasi objek, atau definisi *UML* yaitu sebagai suatu

bahasa yang sudah menjadi standar pada visualisasi, perancangan dan juga mendokumentasikan sistem software. (Bay Haqi, 2019)

2.4.1 Use case Diagram

Use case diagram yaitu salah satu jenis diagram pada *UML* yang menggambarkan interaksi antara sistem dan actor, use case diagram juga dapat mendeskripsikan tipe interaksi antara pemakai sistem dengan sistemnya.



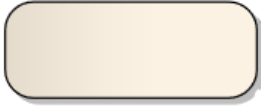
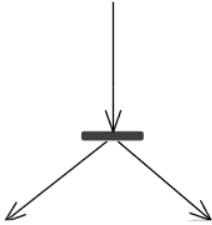
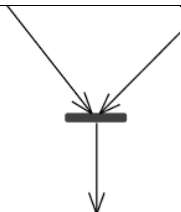

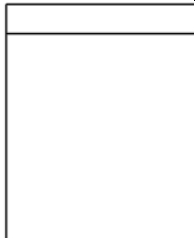
TABEL: 2.1 Simbol *Use case* (A Hendini, 2016)

Simbol	Keterangan
	<i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, yang dinyatakan dengan menggunakan kata kerja.
	<i>Actor</i> atau aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem.
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidentifikasi data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

2.4.2 Activity Diagram

Activity diagram atau diagram aktivitas yaitu salah satu jenis diagram pada *UML* yang dapat memodelkan proses-proses apa saja yang terjadi pada sistem.





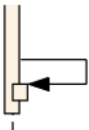


TABEL: 2.2 Simbol *Activity* (A Hendini, 2016)

Simbol	Keterangan
	<i>Start point</i> , merupakan awal aktifitas.
	<i>End Point</i> , merupakan akhir aktifitas.
	<i>Activities</i> , menggambarkan sesuatu proses atau kegiatan.
	<i>Fork/Percabangan</i> , digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau menggabungkan kedua paralel menjadi satu.
	<i>Join/Penggabung</i> atau <i>rake</i> digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision point</i> , menggambarkan pilihan untuk pengambilan keputusan.
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa yang melakukan apa.

2.4.3 Sequence Diagram

Sequence diagram yaitu salah satu jenis diagram pada *UML* yang menjelaskan interaksi objek berdasarkan urutan waktu, *sequence* diagram juga dapat menggambarkan urutan atau tahapan yang harus dilakukan untuk dapat menghasilkan sesuatu seperti pada *use case* diagram.

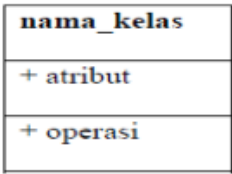






TABEL: 2.3 Simbol *Sequence* (A Hendini, 2016)

Simbol	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interfaces</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>form</i> .
	<i>Control Class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas.
	<i>Message</i> , simbol mengirim pesan antar kelas.
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> , terdapat <i>activation</i> .

2.4.4 Class Diagram

Class diagram yaitu salah satu jenis diagram pada *UML* yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan.




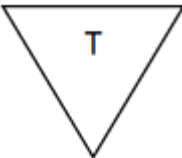




TABEL: 2.4 Simbol *Class* (Rossa A.S dan M. Salahuddin, 2015)



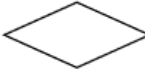

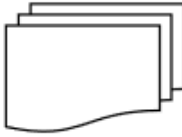
Simbol	Keterangan
<p>Kelas</p> 	Kelas pada struktur sistem.
<p>Antarmuka/<i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
	Relasi antara kelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i> .
<p>Asosiasi berarah / <i>Directed association</i></p> 	Relasi antara kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai <i>multiplicity</i> .
	Relasi antara kelas dengan makna generalisasi-spesialisasi (umum khusus).
<p>Kebergantungan / <i>Dependency</i></p> 	Relasi antara kelas dengan makna kebergantungan.
	Relasi antara kelas dengan makna semua-bagian (<i>whole-part</i>).

2.5 Flowchart

Sistem *flowchart* adalah representasi grafik dari relasi fisik diantara elemen-elemen sistem. Elemen-elemen tersebut antara lain termasuk departemen organisasi, aktifitas manual, program komputer, *hardcopy*, dan *softcopy*. Sebagai representasi dari sebuah program, *flowchart* maupun algoritma dapat menjadi alat bantu yang digunakan untuk memudahkan perancangan alur urutan logika suatu program. Memudahkan pelacakan sumber keasalahan program, dan sebagai alat untuk menerangkan logika program. (Hall, 2015)

TABEL: 2.5 Simbol *Flowchart* (Hall, 2015)

Simbol	Keterangan
	Digunakan untuk memulai, mengakhiri, atau titik henti dalam sebuah proses atau program.
	Sebuah dokumen/laporan. Dokumen dapat dicetak atau dibuat dengan tangan.
	Sebuah kegiatan pemrosesan yang dilaksanakan secara manual.
	Arsip dokumen disimpan dan diambil secara manual. Huruf di dalamnya menunjukkan cara pengurutan arsip: N=Urut Nomor, A=Urut Abjad, T=Urut Tanggal.
	Digunakan untuk menggambarkan berbagai media input dan output.
	Data disimpan secara permanen pada disk bermagnet
	Menghubungkan bagan alir yang berada di halaman berbeda.
	Entri data alat oleh on line seperti terminal CRT dan computer pribadi.

	Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer.
	Arus dokumen atau pemrosesan.
	Sebuah tahap pembuat keputusan
	Menghubungkan bagan alir yang ada pada halaman yang sama.
	Digambarkan dengan menumpuk simbol dokumen dan pencetakan nomor dokumen dibagian depan dokumen pada bagian kiri atas.

2.6 Penilaian Kinerja

Secara umum penilaian kinerja adalah proses dimana organisasi mengevaluasi performa atau kinerja karyawan dengan tujuan untuk meningkatkannya. Pada prinsipnya penilaian kinerja merupakan cara pengukuran kontribusi-kontribusi dari individu dalam instansi yang dilakukan terhadap organisasi. Nilai penting dari penilaian kinerja adalah menyangkut penentuan tingkat kontribusi individu atau kinerja yang diapresiasi dalam menyelesaikan tugas-tugas yang menjadi tanggung jawab. (Sari dan Saleh, 2014)

2.7 Helpdesk

Pengertian dari *helpdesk* adalah seseorang yang memberikan layanan bagi pengguna sistem dan teknologi informasi di suatu institusi tertentu. *Helpdesk* diharapkan dapat mengatasi permasalahan dan memberikan layanan terhadap

seluruh permasalahan baik yang bersifat internal (pengembangan aplikasi) maupun seluruh permasalahan operasional aplikasi ataupun hal-hal yang berhubungan dengan sistem dan teknologi informasi. (Tarmuji, 2013)

Lebih rinci tentang tujuan diadakannya helpdesk ini antara lain :

1. Mampu memberikan solusi pemecahan masalah (*troubleshooting*) terhadap berbagai permasalahan yang dihadapi oleh pengguna.
2. *Me-manage* setiap keluhan/permintaan pengguna dan tindak lanjutnya.
3. Mengetahui setiap permasalahan yang sedang/masih dalam proses perbaikan dan memastikan tidak ada yang terlewat dalam penelusuran masalah.
4. Meningkatkan respon terhadap permintaan pengguna dalam menyempurnakan dan menciptakan aplikasi.
5. Memperoleh *database* permasalahan dan solusinya (*troubleshooting*) yang dapat digunakan sebagai acuan dalam menyelesaikan setiap permasalahan.

2.8 Data Management

Manajemen data atau *data management* merupakan suatu bentuk kegiatan pengelolaan sumber daya informasi yang meliputi, pertama proses pengumpulan data dan pencatatan ke dalam dokumen yang berfungsi sebagai masukan (*input*) bagi sistem. Kedua, proses penyimpanan sumber daya informasi ke dalam suatu berkas dokumen. Ketiga, pemeliharaan proses penambahan data baru dan perubahan data yang ada supaya sumber daya informasi tetap mutakhir. Keempat,

mengatur (*organize*) proses penyusunan data sedemikian rupa untuk dapat memenuhi kebutuhan informasi para pemakai. (Jogiyanto, 2005)

2.9 Fallout

Fallout yang ada pada ruang lingkup PT. Telkom ini adalah sebuah kesalahan yang teridentifikasi oleh sistem dimana kesalahan ini meliputi kesalahan data ada pada suatu objek yang tidak sesuai dengan ketentuan yang ada. Maka dari itu *closed fallout* dapat diartikan menjadi kesalahan data yang ada pada sistem yang telah diperbaiki oleh bagian terkait. Dalam kasus ini, bagian terkait ini adalah *helpdesk data management*.

BAB III

ANALISA MASALAH DAN PERANCANGAN PROGRAM

3.1 Analisa

3.1.1 Analisa Masalah

Berdasarkan dengan apa yang telah di teliti dan di amati oleh penulis di PT. Telkom, dapat di simpulkan bahwa penilaian kinerja karyawan yang dilakukan masih belum menggunakan teknologi informasi apapun. Hal ini dapat menyebabkan ketidakakuratan penilaian yang dilakukan oleh manager.

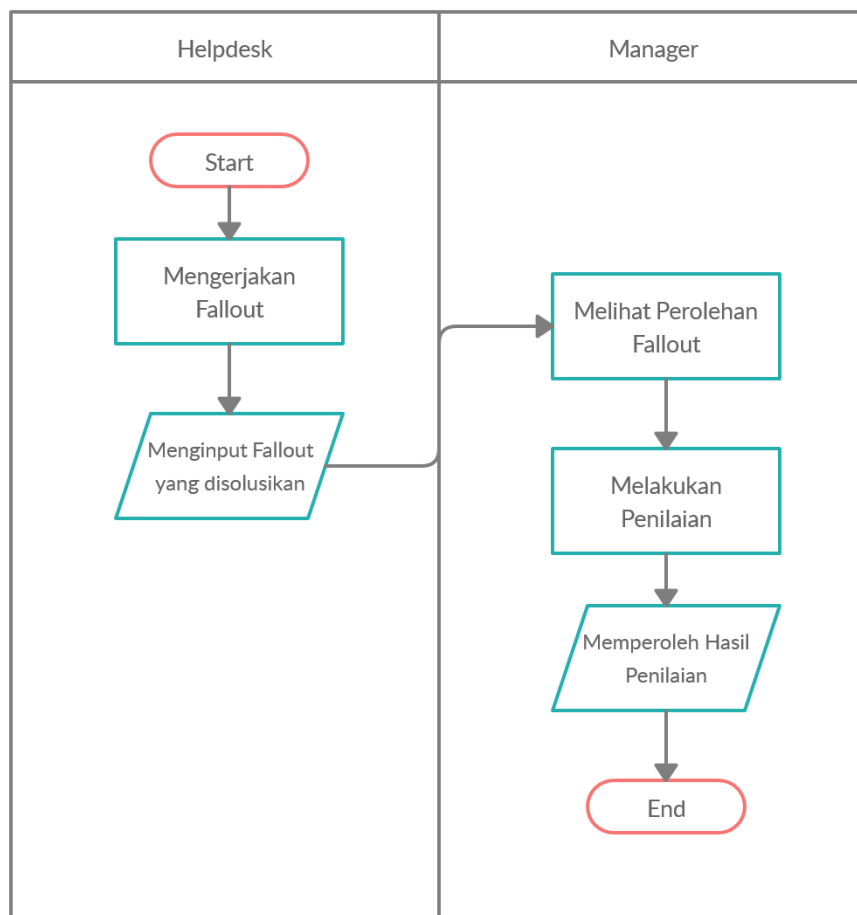
Maka dari itu, penulis berniat untuk menanggulangi permasalahan yang ada pada kegiatan penilaian tersebut. Apabila permasalahan yang terjadi tidak ditanggulangi, akan menyebabkan kerugian kepada karyawan itu sendiri, dimana penilaian yang tidak akurat akan membuat hasil yang diterima tidak sesuai dengan apa yang terjadi.

3.1.2 Analisa Sistem Berjalan

Dari hasil pengamatan penulis berikut adalah prosedur yang berjalan pada saat ini :

1. *Helpdesk* mengerjakan *fallout* yang muncul secara berkala dan mencatat berapa banyak *fallout* yang telah di solusikan.
2. *Helpdesk* menginputkan jumlah *fallout* yang telah di solusikan.

3. *Manager* melihat laporan perolehan *fallout* yang telah di solusikan oleh *Helpdesk*.
4. *Manager* melakukan penilaian terhadap *helpdesk* secara manual.
5. *Manager* memperoleh hasil penilaian secara manual.



GAMBAR: 3.1 *Flowchart* Analisa Sistem Berjalan.

3.1.3 Analisa Kebutuhan Non Fungsional

Adapun kebutuhan secara spesifik yang dibutuhkan agar dapat menunjang kelancaran dalam penggunaan sistem pendukung keputusan penilaian kinerja tersebut. Ada 3 macam kebutuhan yang dapat di jabarkan, antara lain :

a. *Kebutuhan Hardware*

Hardware atau perangkat keras yang dibutuhkan oleh user untuk menunjang kelancaran dalam penggunaan sistem yaitu PC atau laptop dan modem. Berikut adalah minimal spesifikasi detail mengenai perangkat keras yang di butuhkan :

1. *Processor : Dual-Core 2.7 Ghz.*
2. *RAM : 2 GB DDR3*
3. *Disk : Free Space 30 GB*
4. *Graphic Card : Graphics card that support DirectX 9.0c 128 MB*
5. *Modem : HSDPA 7.2 Mbps/ HSUPA 5.76 Mbps*

b. *Kebutuhan Software*

Software atau perangkat lunak yang dibutuhkan untuk menunjang kelancaran dalam penggunaan sistem antara lain :

1. *Sistem Operasi Windows 7, 8, 8.1 atau 10.*
2. *Browser.*
3. *Web Server*

c. *Kebutuhan Brainware*

Brainware atau pengguna dalam sistem ini ada 2 level. Berikut adalah detail peran mengenai pembagian level dari user :

1. *Heldpesk/Karyawan*

Helpdesk disini memiliki tugas yaitu mengisi absensi dan mengisi perolehan *fallout*.

2. Admin/Manager

Admin sekaligus manager disini memiliki hak akses yang lebih dari *halpdesk*, bisa dikatakan bahwa admin/*manager* ini memiliki akses keseluruhan fitur yang ada di sistem seperti menambahkan *helpdesk*, menghapus *helpdesk*, mengubah *helpdesk*, menkonfirmasi absensi *helpdesk*, menilai *helpdesk* dan melihat perolehan *fallout* keseluruhan *helpdesk*.

3.1.4 Analisa Kebutuhan Fungsional

Analisa fungsional ini menggambarkan kebutuhan yang dijalankan oleh sistem sesuai dengan fungsi yang ada. Berikut adalah gambaran mengenai fungsi-fungsi yang ada pada sistem :

1. Absensi *Helpdesk*

Setelah digabungkan dengan sistem, langkah-langkah dalam melakukan absensi mengalami sedikit perubahan. Berikut adalah langkah-langkah dalam melakukan absensi yang dilakukan oleh *helpdesk*:

1. *Helpdesk* melakukan absensi dengan cara memasuki menu absensi yang ada pada sistem.
2. *Helpdesk* melakukan absensi masuk apabila akan memulai kerja.
3. Sistem akan mengirimkan absensi yang dilakukan oleh *helpdesk* ke dalam database.

4. Sistem akan mengirimkan absensi yang dilakukan *helpdesk* kepada admin/*manager*.
5. Admin/*manager* dapat melihat riwayat absensi pada sistem yang dilakukan oleh *helpdesk*.

2. Penilaian *Helpdesk*

Dengan adanya sistem penilaian tidak perlu dikhawatirkan untuk adanya kekeliruan perhitungan yang dilakukan oleh *manager*. Berikut adalah langkah-langkah penilaian setelah diadakannya sistem:

1. Admin/*Manager* melihat perolehan *closed fallout* yang telah dimasukan kedalam sistem oleh *helpdesk*.
2. Admin/*Manager* melihat ketepatan waktu *helpdesk* melalui absensi yang telah tercatat pada sistem.
3. Admin/*Manager* memasukan nilai kepada masing-masing kriteria yang dibutuhkan sesuai dengan pengamatan lapangan dan catatan yang diperoleh dari sistem.
4. Sistem secara otomatis akan melakukan proses perhitungan dengan menggunakan metode *TOPSIS*.
5. Admin/*Manager* memperoleh nilai preferensi dari hasil perhitungan yang dilakukan oleh sistem.

3.2 Perancangan Sistem

3.2.1 Desain Sistem dengan *Unified Modeling Language (UML)*

Perancangan yang dilakukan oleh penulis terhadap alur sistem yang dibuat akan di implementasikan menggunakan *UML* yang terdiri dari *use case* diagram, *activity* diagram, *sequence* diagram dan *class* diagram.

3.2.1.1 *Use Case* Diagram

Pada *use case* diagram ini menjelaskan interaksi antara sistem dengan *helpdesk* atau *manger* yang terlibat dengan sistem.

1. Definisi Aktor

TABEL: 3.1 Tabel Definisi Aktor

No.	Aktor	Deskripsi
1.	Helpdesk	Menginput <i>closed fallout</i> dan melakukan absensi.
2.	Admin	Mengelola user, melakukan penilaian, mengkonfirmasi absensi & <i>closed fallout</i> .

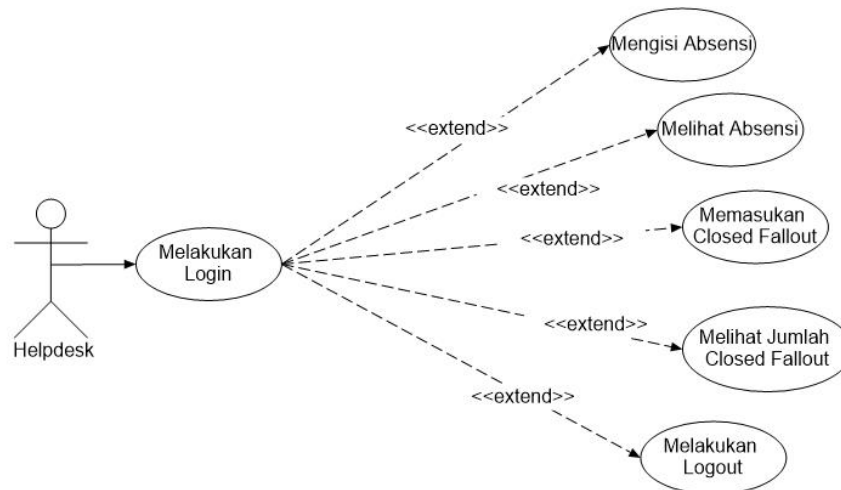
2. Definisi *Use Case*

TABEL: 3.2 Tabel Definisi *Use Case*

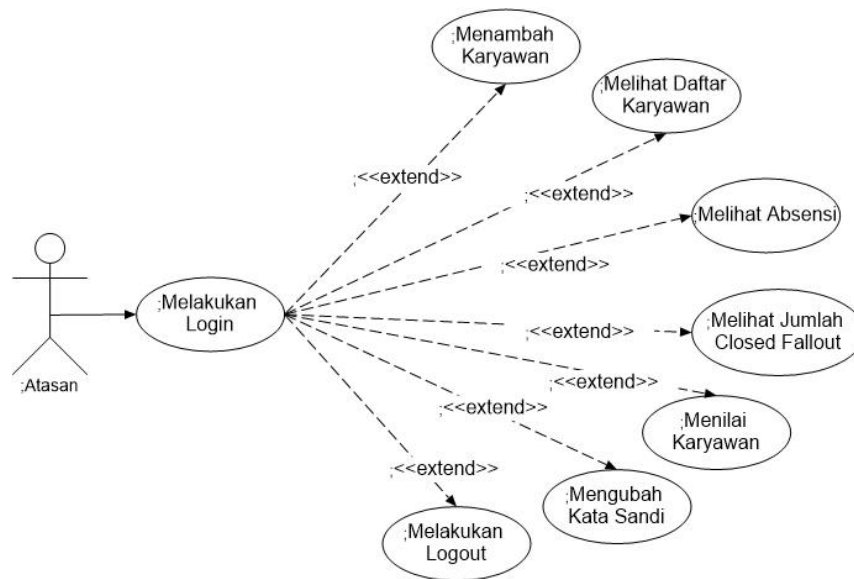
No.	<i>Use Case</i>	Deskripsi
1.	Melakukan Login	Proses masuknya user ke dalam sistem.
2.	Mengisi Absensi	Proses absensi yang dilakukan di sistem.
3.	Melihat Absensi	Proses untuk melihat data absensi secara detail.
4.	Memasukan <i>Closed Fallout</i>	Proses pemasukan data jumlah perolehan <i>fallout</i> yang dikerjakan oleh <i>helpdesk</i>
5.	Melihat Jumlah <i>Closed Fallout</i>	Proses untuk melihat data perolehan <i>fallout</i> secara detail.

6.	Melakukan Logout	Proses keluarnya user dari sistem.
7.	Menambah Karyawan	Proses pemasukan atau penambahan data <i>helpdesk</i> ke dalam sistem.
8.	Melihat Daftar Karyawan	Proses melihat, menghapus dan mengubah data-data <i>helpdesk</i> yang ada pada sistem.
9.	Menilai Karyawan	Proses penilaian <i>helpdesk</i> .
10.	Mengubah Kata Sandi	Proses untuk mengubah kata sandi user.

3. Gambar *Use Case Diagram*.



GAMBAR: 3.2 *Use Case Helpdesk*.

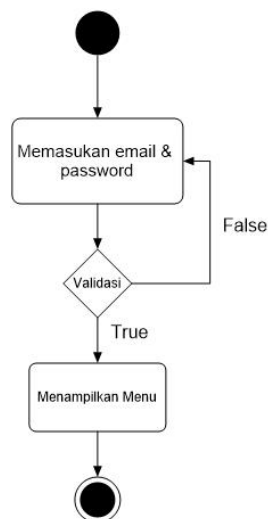


GAMBAR: 3.3 Use Case Admin/Manager.

3.2.1.2 Activity Diagram

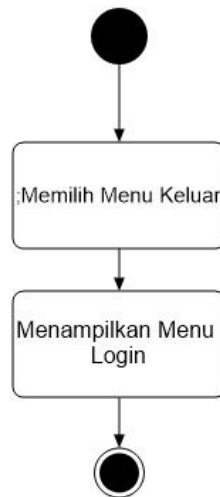
Pada diagram aktivitas ini menjelaskan bagaimana proses-proses yang ada pada sistem berdasarkan dengan fungsi yang dijalankan oleh *helpdesk* maupun *admin/manager*.

1. Proses pada saat *admin/manager & helpdesk* melakukan login.



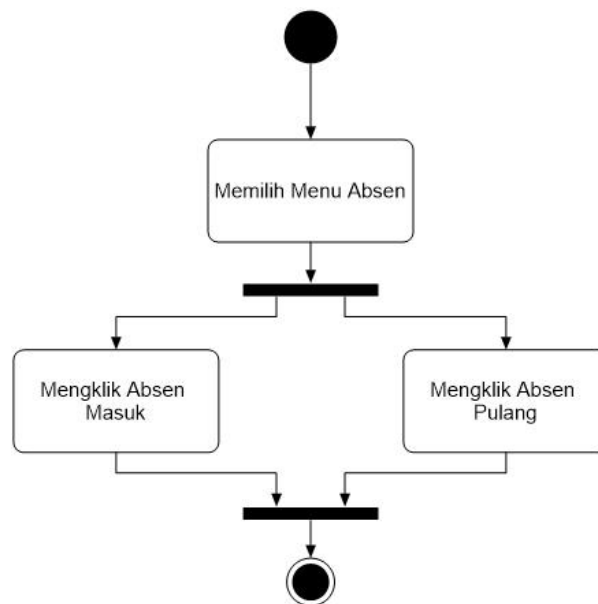
GAMBAR: 3.4 Activity Admin & Helpdesk Login.

2. Proses pada saat admin/*manager* & *helpdesk* melakukan logout.



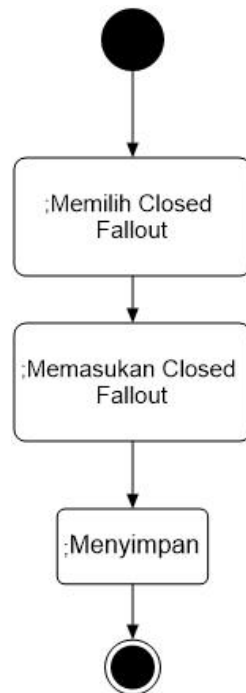
GAMBAR: 3.5 Activity Admin & Helpdesk Logout.

3. Proses pada saat *helpdesk* melakukan absensi.



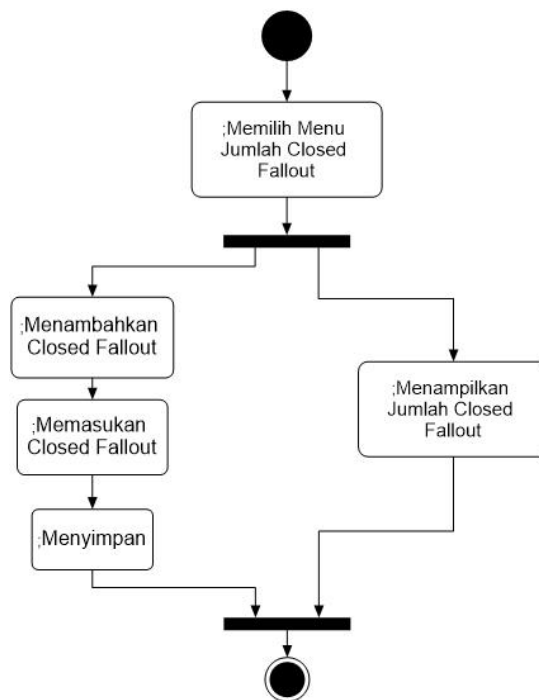
GAMBAR: 3.6 Activity Helpdesk melakukan absensi.

4. Proses pada saat *helpdesk* memasukan *closed fallout*.



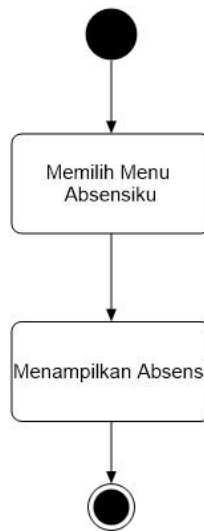
GAMBAR: 3.7 Activity Helpdesk Input Closed Fallout.

5. Proses pada saat *helpdesk* melihat jumlah *closed fallout*.



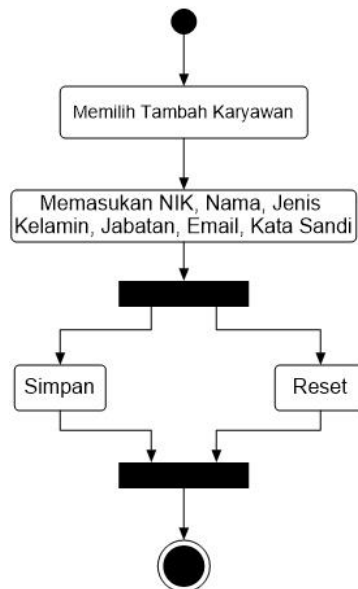
GAMBAR: 3.8 Activity Helpdesk Melihat Jumlah Closed Fallout.

6. Proses pada saat *helpdesk* melihat detail absensi.



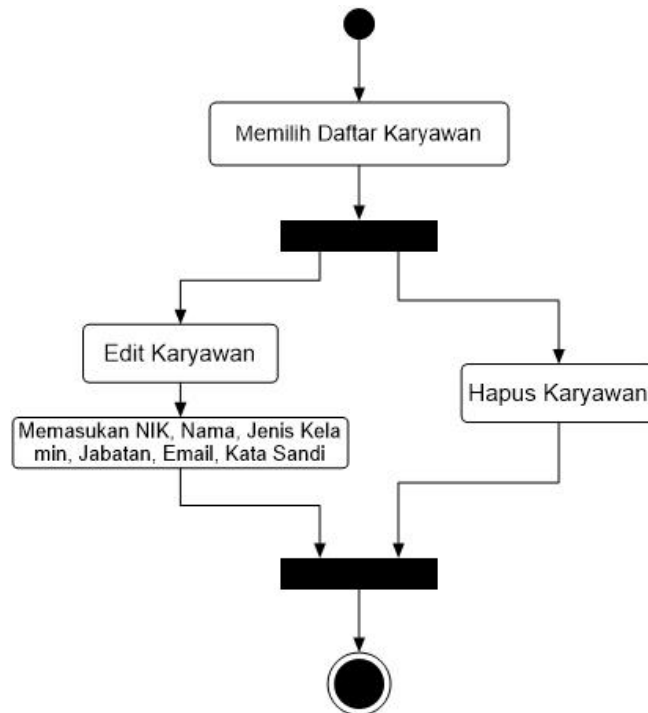
GAMBAR: 3.9 Activity *Helpdesk* Melihat Absensi.

7. Proses pada saat admin/*manager* menambahkan *helpdesk*.



GAMBAR: 3.10 Activity Admin/*Manager* Menambah Helpdesk/Karyawan.

8. Proses pada saat admin/*manager* melihat daftar *helpdesk*.



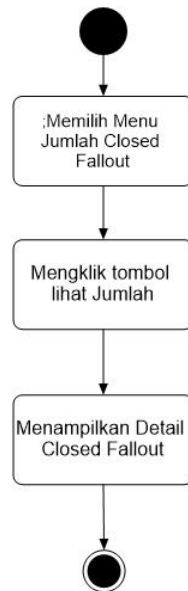
GAMBAR: 3.11 Activity Admin/Manager Melihat Daftar Helpdesk/Karyawan.

9. Proses pada saat admin/*manager* melihat detail *helpdesk*.



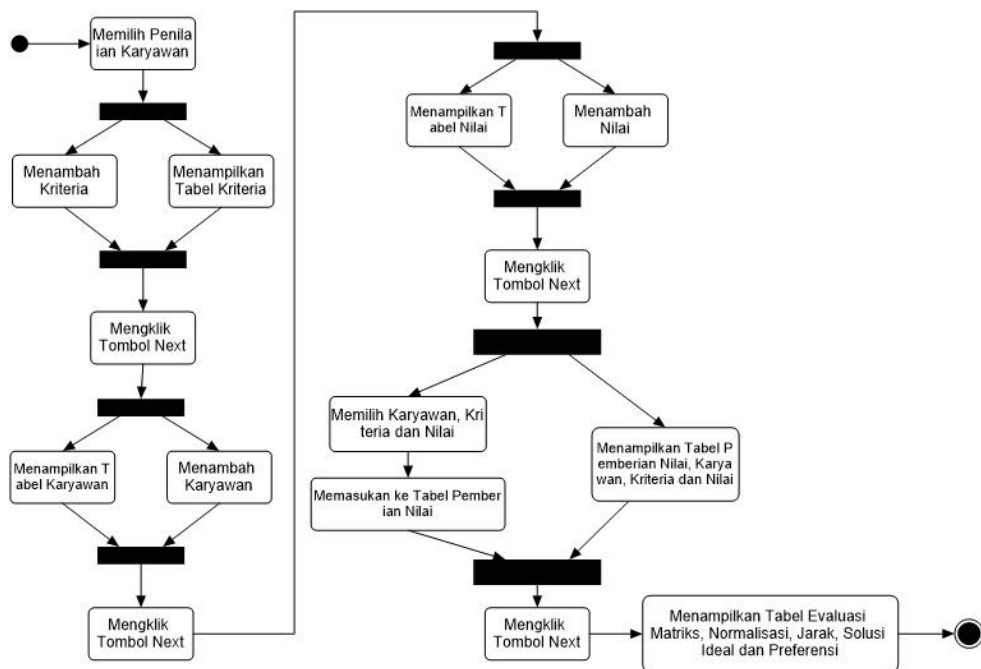
GAMBAR: 3.12 Activity Admin/Manager Melihat Absensi Helpdesk.

10. Proses pada saat admin/manager melihat jumlah *closed fallout helpdesk*.



GAMBAR: 3.13 Activity Admin/Manager Melihat Jumlah Closed Fallout.

11. Proses pada saat admin/manager melakukan penilaian *helpdesk*.



Gambar 3.14 : *Activity Admin/Manager Melakukan Penilaian Helpdesk.*

12. Proses pada saat admin/manager mengubah katasandi.

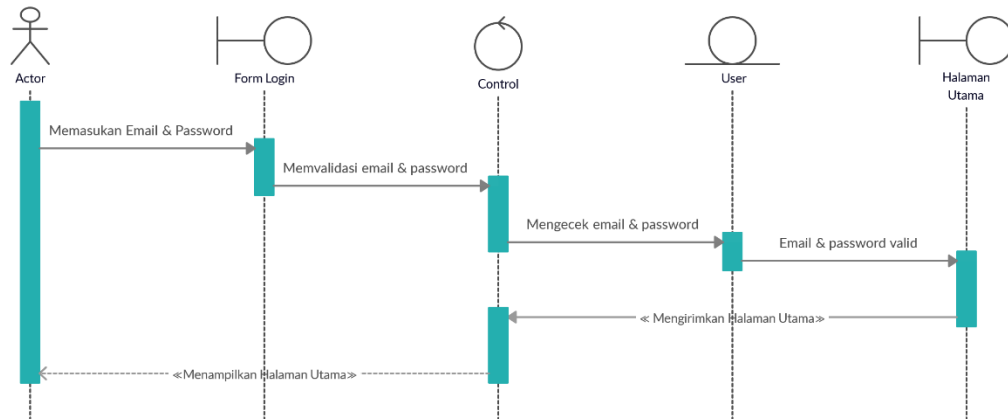


GAMBAR: 3.15 *Activity Admin/Manager Mengubah Kata Sandi.*

3.2.1.3 *Sequence Diagram*

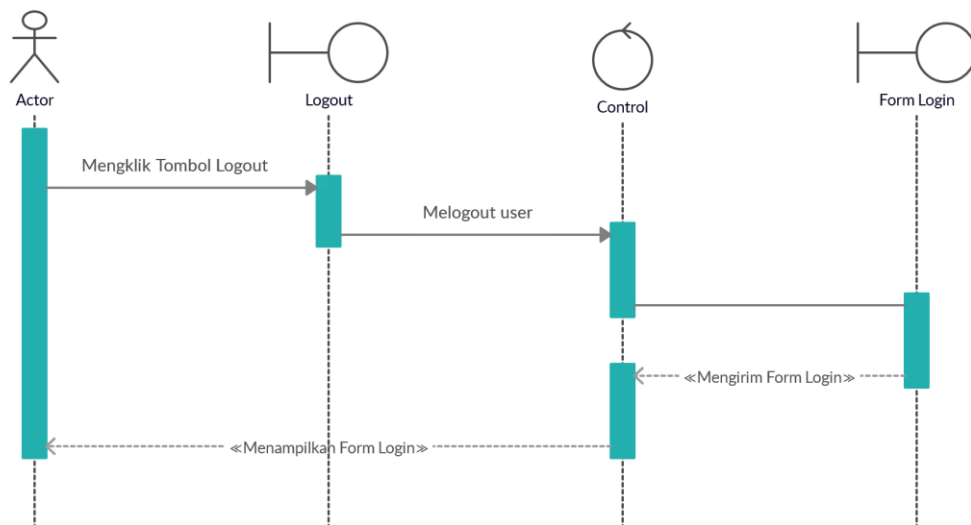
Pada diagram ini menjelaskan bagaimana interaksi antara objek pada sistem yang sedang berjalan berdasarkan urutan waktunya. Berikut adalah *sequence diagram* pada sistem yang dibuat :

1. Interaksi yang terjadi pada saat aktor melakukan proses login ke sistem.



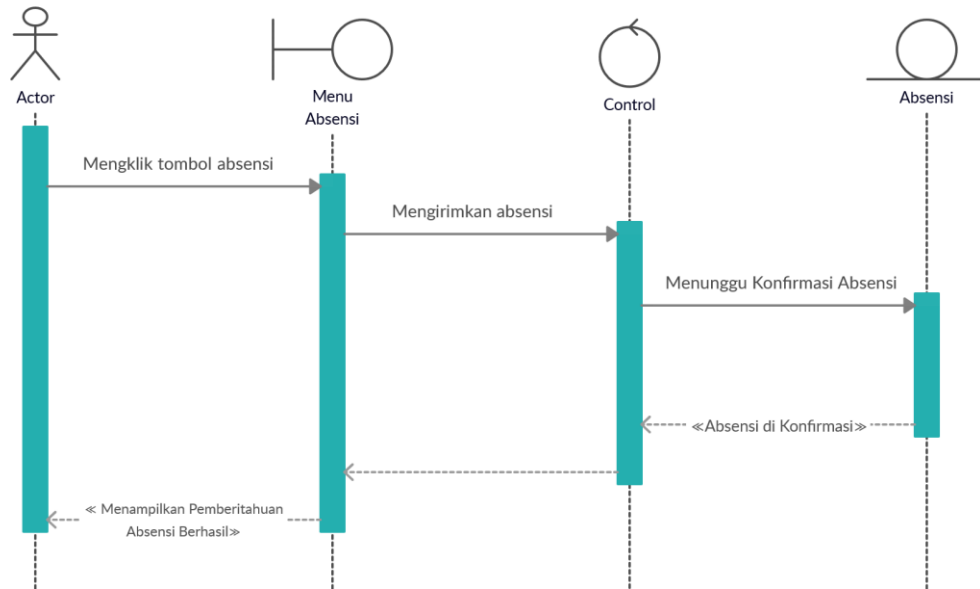
GAMBAR: 3.16 *Sequence Login Admin/Manager & Helpdesk.*

2. Interaksi yang terjadi pada saat aktor melakukan proses logout dari sistem.



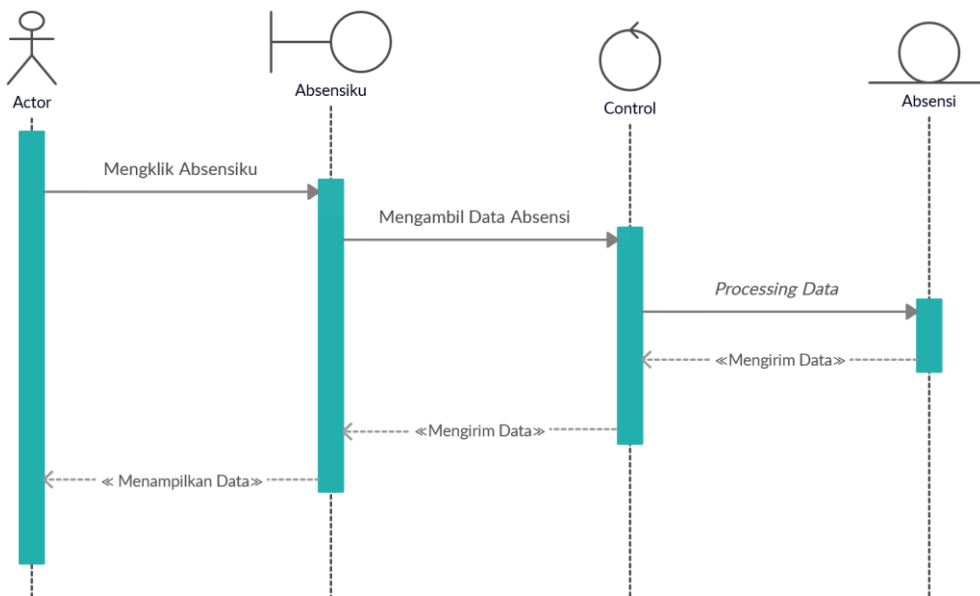
GAMBAR: 3.17 *Sequence Logout Admin/Manager & Helpdesk.*

3. Interaksi yang terjadi pada saat *helpdesk* melakukan proses absensi pada sistem.



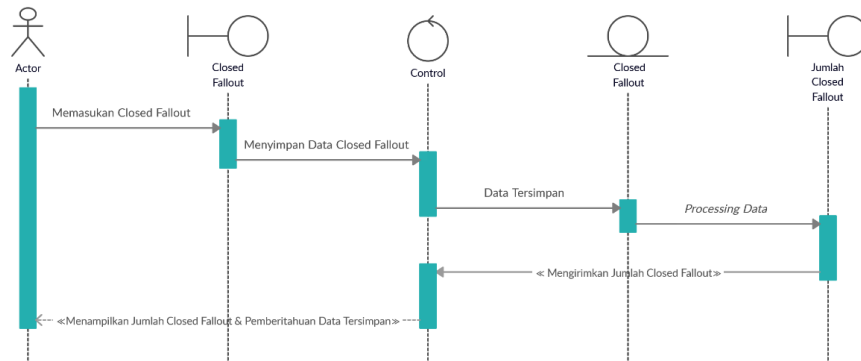
GAMBAR: 3.18 *Sequence Helpdesk Melakukan Absensi.*

4. Interaksi yang terjadi pada saat *helpdesk* melihat data absensi.



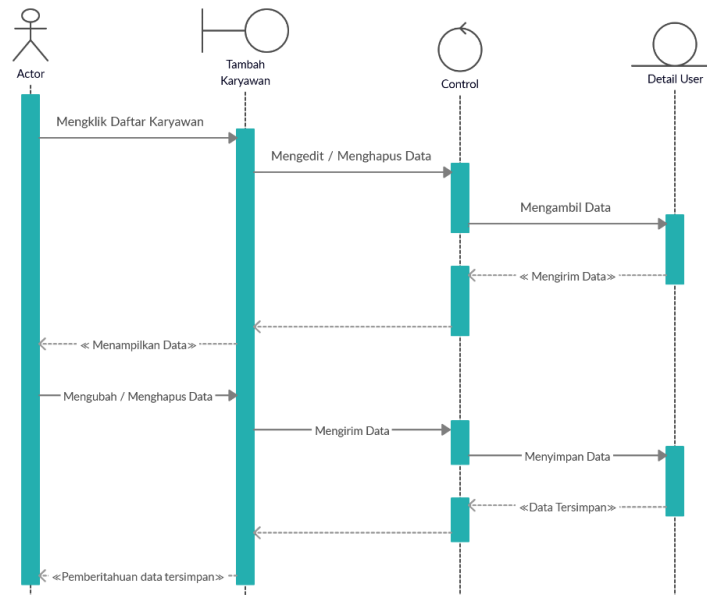
GAMBAR: 3.19 *Sequence Helpdesk Melihat Absensi.*

5. Interaksi yang terjadi pada saat *helpdesk* melakukan proses input data perolehan *fallout*.



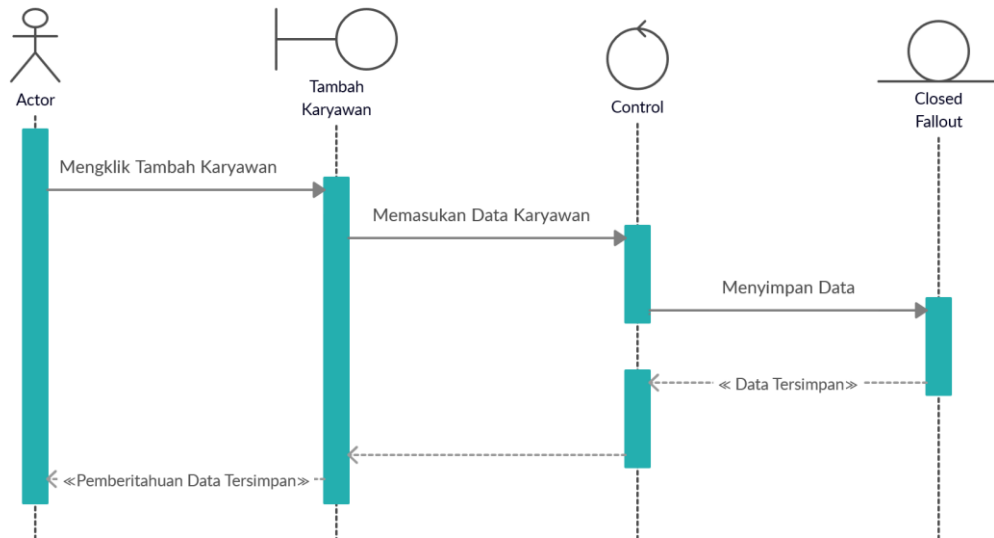
GAMBAR: 3.20 *Sequence Helpdesk Input & Menampilkan Jumlah Closed Fallout.*

6. Interaksi yang terjadi pada saat admin/*manager* melakukan proses melihat, mengubah atau menghapus data *helpdesk*.



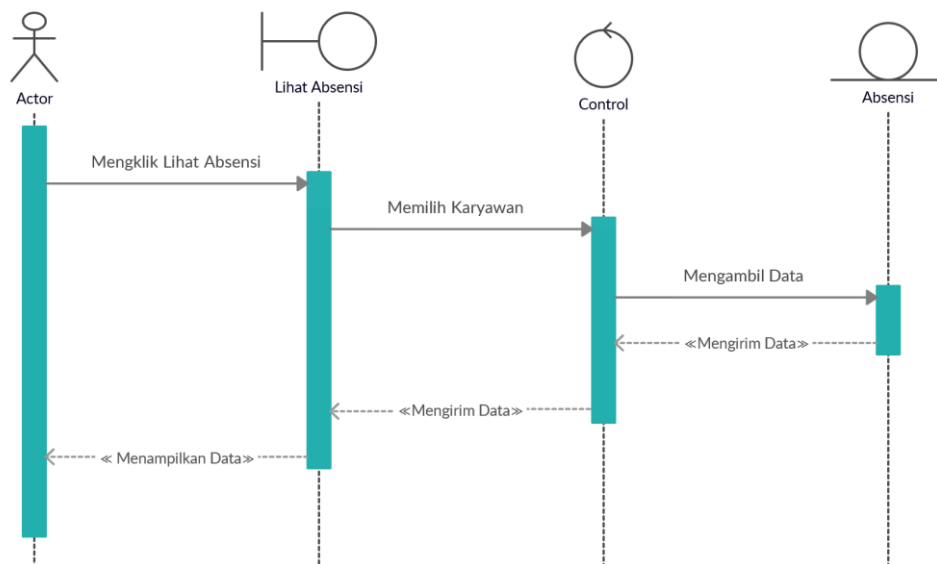
GAMBAR: 3.21 *Sequence Admin/Manager Melihat, Mengubah atau Menghapus Data Helpdesk.*

7. Interaksi yang terjadi pada saat admin/*manager* melakukan proses penambahan *helpdesk*.



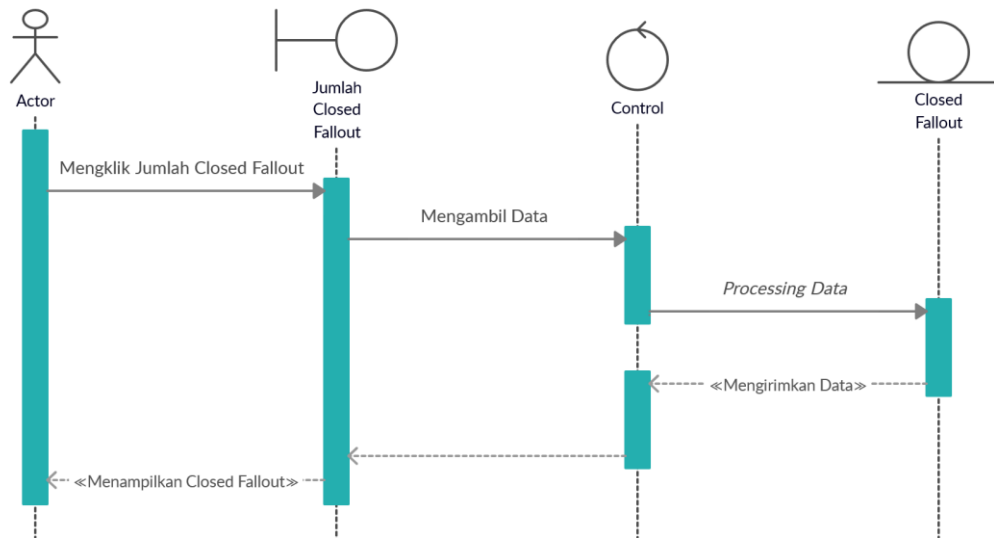
GAMBAR: 3.22 *Sequence Admin/Manager Menambah Helpdesk.*

8. Interaksi yang terjadi pada saat admin/*manager* melihat absensi.



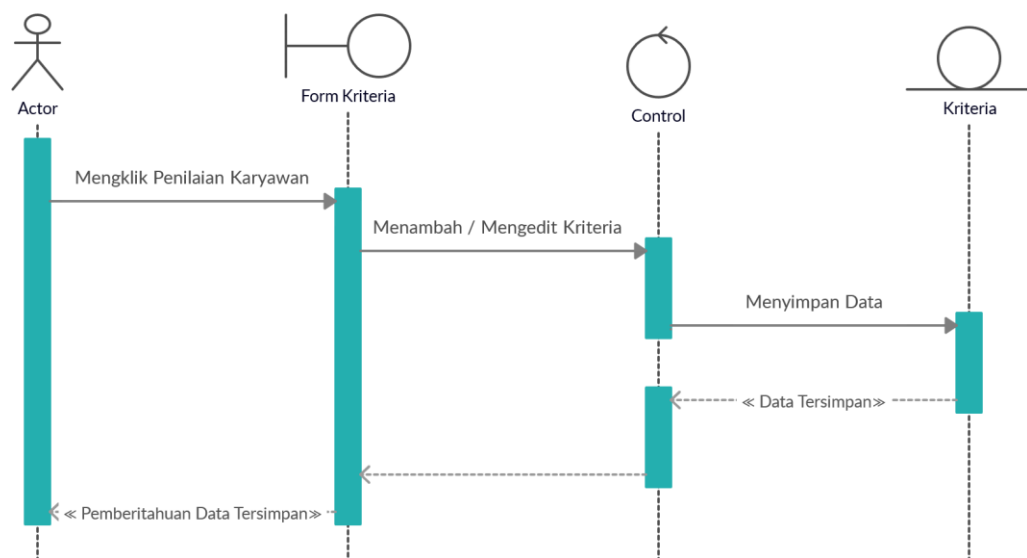
GAMBAR: 3.23 *Sequence Admin/Manager Melihat Absensi Helpdesk.*

9. Interaksi yang terjadi pada saat admin/manager melihat jumlah *closed fallout*.



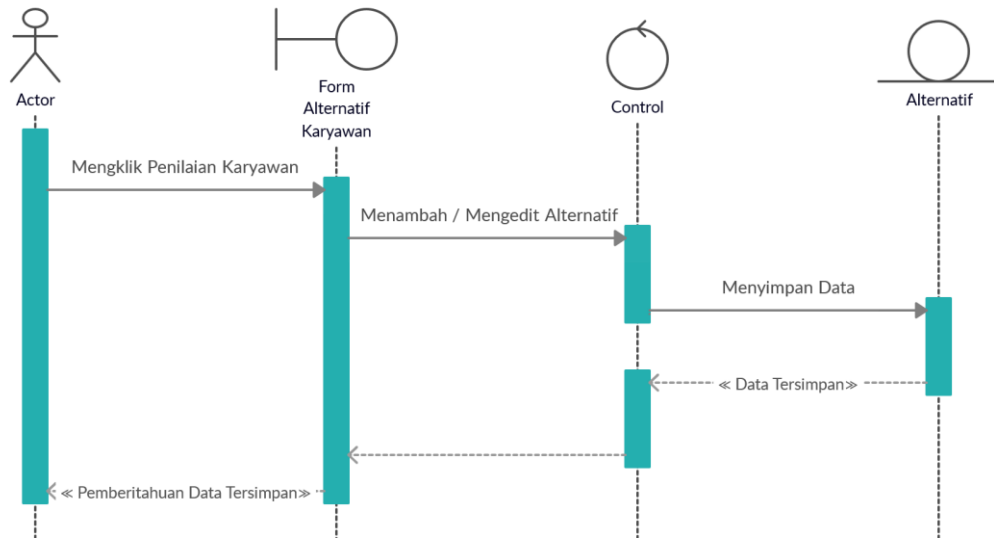
GAMBAR: 3.24 Sequence Admin/Manager Melihat Jumlah Closed Fallout.

10. Interaksi yang terjadi pada saat admin/manager melakukan proses penambahan kriteria penilaian.



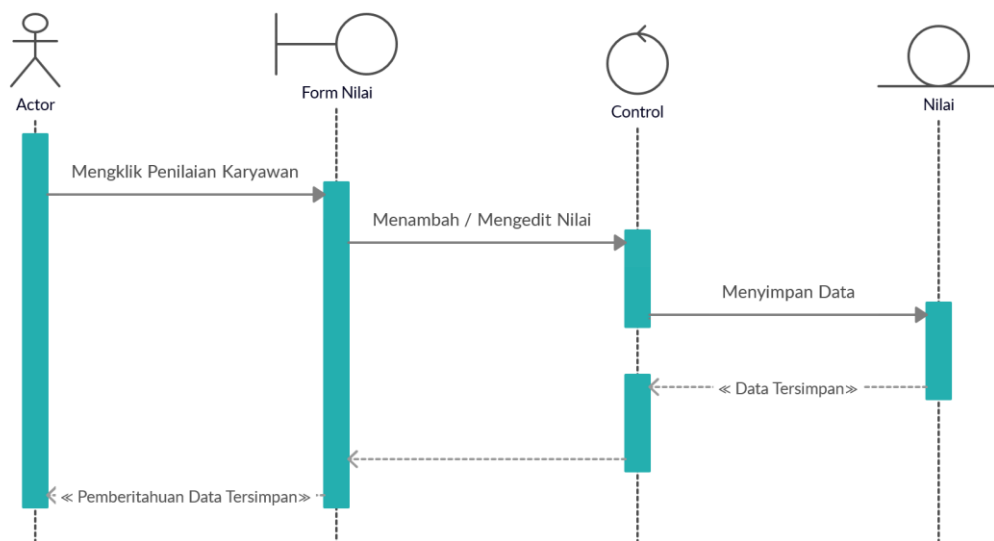
GAMBAR: 3.25 Sequence Admin/Manager Menambah Kriteria.

11. Interaksi yang terjadi pada saat admin/*manager* melakukan proses penambahan alternatif penilaian.



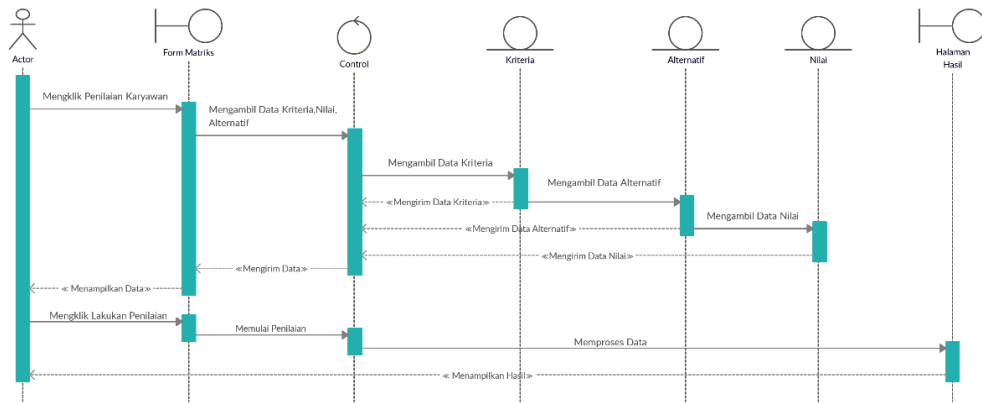
GAMBAR: 3.26 Sequence Admin/Manager Menambah Alternatif.

12. Interaksi yang terjadi pada saat admin/*manager* melakukan proses penambahan nilai penilaian.



GAMBAR: 3.27 Sequence Admin/Manager Menambah Nilai.

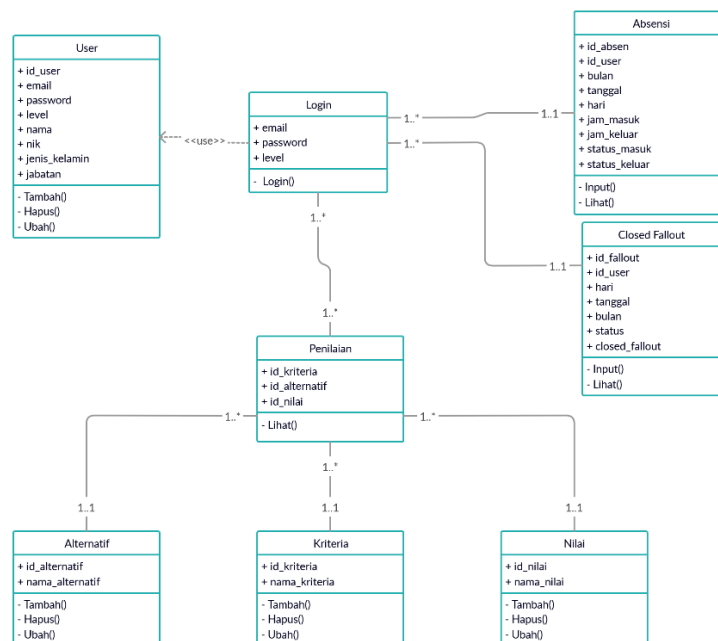
13. Interaksi yang terjadi pada saat admin/manager melakukan proses penilaian *helpdesk*.



GAMBAR: 3.28 Sequence Admin/Manager Melakukan Penilaian.

3.2.1.4 Class Diagram

Pada gambar diagram ini menampilkan kelas-kelas yang ada pada sistem yang dibuat. Berikut adalah *class* diagramnya :



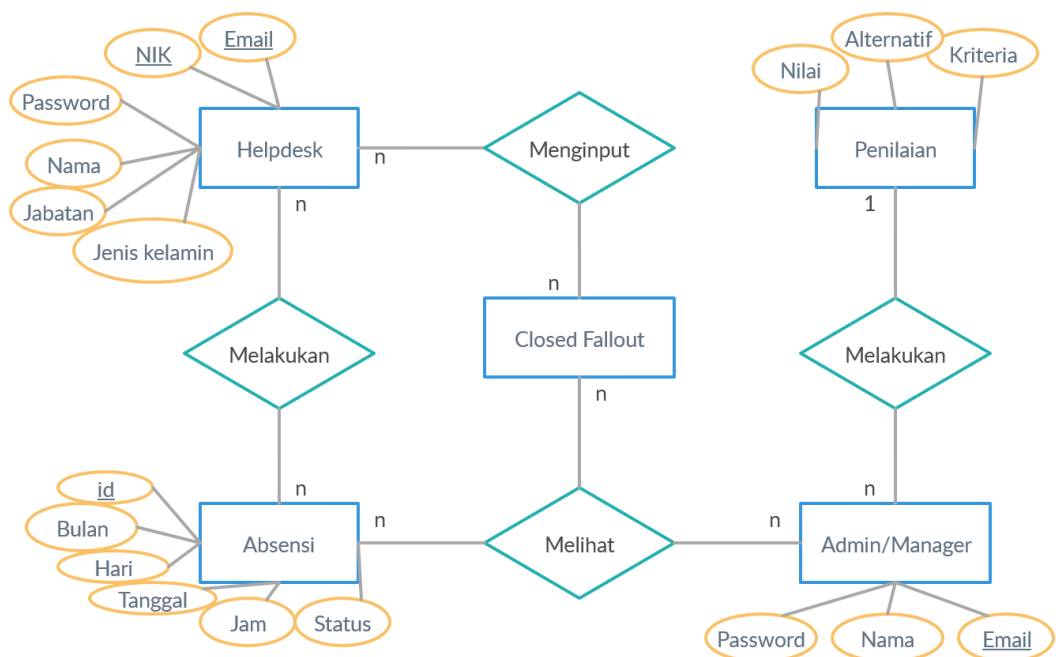
GAMBAR: 3.29 Class Diagram Sistem.

3.2.2 Desain Database Sistem

Selain menggunakan *UML* untuk membuat sebuah desain sistem, desain database pun perlu diperlukan untuk memperjelas bagaimana hubungan antara entitas yang ada. Desain database yang digunakan antara lain *ERD* dan struktur tabel.

3.2.2.1 Entity Relationship Database (ERD)

Entity Relationship Database atau (*ERD*) yang digunakan dalam pembuatan sistem pendukung keputusan adalah sebagai berikut :



GAMBAR: 3.30 ERD Sistem.

3.2.2.2 Struktur Tabel

Berikut adalah struktur tabel yang digunakan dalam pembuatan sistem pendukung keputusan.

TABEL: 3.3 Tabel *User*

Field	Tipe Data
id_user	int(10)
email_user	varchar(255)
pwd_user	varchar(255)
level_user	enum('hd,'adm')

TABEL: 3.4 Tabel Detail *User*

Field	Tipe Data
id_user	int(10)
nik_user	int(25)
nama_user	varchar(255)
jbtn_user	varchar(255)
jk_user	varchar(5)

TABEL: 3.5 Tabel *Closed Fallout*

Field	Tipe Data
id_fallout	int(10)
id_user	int(10)
id_bulan	int(10)
id_hari	int(10)
id_tanggal	int(10)
closed_fallout	longtext
status_fallout	enum('Menunggu', 'Dikonfirmasi', 'Ditolak')

TABEL: 3.6 Tabel Absensi

Field	Tipe Data
id_absen	int(11)
id_user	varchar(100)
id_bulan	int(10)
id_hari	int(10)
id_tanggal	int(10)
jam_masuk	varchar(50)
status_jam_masuk	enum('Menunggu', 'Dikonfirmasi', 'Ditolak')
jam_keluar	varchar(50)
status_jam_keluar	enum('Menunggu', 'Dikonfirmasi', 'Ditolak')

TABEL: 3.7 Tabel Kriteria

Field	Tipe Data
id_kriteria	int(10)
nama_kriteria	int(10)
bobot	float

TABEL: 3.8 Tabel Alternatif

Field	Tipe Data
id_alternatif	varchar(10)
nama_alternatif	varchar(50)

TABEL: 3.9 Tabel Nilai

Field	Tipe Data
id_nilai	varchar(10)
nilai	varchar(10)

TABEL: 3.10 Tabel Penilaian

Field	Tipe Data
id_alternatif	varchar(10)
id_kriteria	varchar(10)
nilai	float

TABEL: 3.11 Tabel Hari

Field	Tipe Data
id_hari	int(10)
hari	varchar(50)

TABEL: 3.12 Tabel Tanggal

Field	Tipe Data
id_tanggal	int(10)
tanggal	varchar(20)

TABEL: 3.13 Tabel Bulan

Field	Tipe Data
id_bulan	int(10)
bulan	varchar(20)

3.2.3 Desain Interface Sistem

Sebelum membuat program tentunya ada desain tampilan yang dimiliki oleh sistem. Berikut adalah desain tampilan dari sistem yang dibuat.

1. Tampilan menu login yang ditampilkan oleh sistem.

The diagram illustrates a login menu interface. It consists of a central rectangular box. Inside this box, there is a rounded rectangle labeled 'Logo'. Below the logo are three input fields: one labeled 'email', one labeled 'password', and a rounded button labeled 'Login'. The entire central box is positioned on a larger rectangular area labeled 'Background'.

GAMBAR: 3.31 Desain Menu Login.

2. Tampilan menu absensi yang ditampilkan pada *helpdesk*.

Nama Aplikasi	
Jam	Absensi
	Masuk Pulang
Absensi	
Absensiku Closed Fallout Jumlah Logout	Logo

GAMBAR: 3.32 Desain Menu Absensi *Helpdesk*.

3. Tampilan menu absensiku yang ditampilkan pada *helpdesk*.

Nama Aplikasi			
Jam	Absensiku		
	Bulan		
Absensi	No. Hari	Jam Masuk	Jam Keluar
Absensiku	<div style="text-align: center; border: 1px solid black; width: 200px; height: 40px; margin: 0 auto;">Logo</div>		
Closed Fallout			
Jumlah Logout			

GAMBAR: 3.33 Desain Menu Absensiku *Helpdesk*.

4. Tampilan menu *closed fallout* yang ditampilkan pada *helpdesk*.

Nama Aplikasi	
Jam	Closed Fallout Hari ini
	<div style="text-align: center; border: 1px solid black; width: 200px; height: 40px; margin: 0 auto;">Text</div>
	<div style="text-align: center; border: 1px solid black; width: 60px; height: 20px; margin: 0 auto;">Simpan</div>
	<div style="text-align: center; border: 1px solid black; width: 200px; height: 40px; margin: 0 auto;">Logo</div>
Absensi	
Absensiku	
Closed Fallout	
Jumlah	
Logout	

GAMBAR: 3.34 Desain Menu *Closed Fallout Helpdesk*.

5. Tampilan menu jumlah *closed fallout* yang ditampilkan pada *helpdesk*.

Nama Aplikasi		
<p>Jam</p> <p>Absensi Absensiku Closed Fallout Jumlah Logout</p>	My Closed Fallout	
	No. Hari Closed Fallout	
	<div style="border: 1px solid black; width: 100px; height: 40px; margin: 0 auto;">Logo</div>	

GAMBAR: 3.35 Desain Menu Jumlah *Closed Fallout Helpdesk*.

6. Tampilan menu tambah *helpdesk* yang ditampilkan pada admin/*manager*.

Nama Aplikasi		
<p>Jam</p> <p>Tambah</p> <p>Daftar</p> <p>Absen</p> <p>Lihat Absensi</p> <p>Closed Fallout</p> <p>Jumlah</p> <p>Penilaian</p> <p>Ubah Password</p> <p>Logout</p>	Tambah Karyawan	
	NIK	<input type="text"/>
	Nama	<input type="text"/>
	Jenis Kelamin	<input type="text"/>
	Jabatan	<input type="text"/>
	Email	<input type="text"/>
	Katasandi	<input type="text"/>
		<input type="text"/>
		<input type="button" value="Simpan"/> <input type="button" value="Reset"/>
	<div style="border: 1px solid black; width: 100px; height: 40px; margin: 0 auto;">Logo</div>	

GAMBAR: 3.36 Desain Menu Admin Menambah *Helpdesk*.

7. Tampilan menu daftar *helpdesk* yang ditampilkan pada admin/*manager*.

Nama Aplikasi				
Jam	Daftar Karyawan			
Tambah	No.	Nama	Jabatan	Aksi
Daftar	1.	xxxxxxx	xxxxxxxxxxxx	Hapus Edit
Absen	<div style="text-align: center; border: 1px solid black; width: 200px; height: 100px; margin: 0 auto;">Logo</div>			
Lihat Absensi				
Closed Fallout				
Jumlah				
Penilaian				
Ubah Password				
Logout				

GAMBAR: 3.37 Desain Menu Admin Melihat Daftar *Helpdesk*.

8. Tampilan menu lihat absensi *helpdesk* yang ditampilkan pada admin/*manager*.

Nama Aplikasi				
Jam	Lihat Absensi			
Tambah	No.	Nama	Jabatan	Aksi
Daftar	1.	xxxxxxx	xxxxxxxxxxxx	Lihat
Absen	<div style="text-align: center; border: 1px solid black; width: 200px; height: 100px; margin: 0 auto;">Logo</div>			
Lihat Absensi				
Closed Fallout				
Jumlah				
Penilaian				
Ubah Password				
Logout				

GAMBAR: 3.38 Desain Menu Admin Melihat Absensi.

9. Tampilan menu jumlah *closed fallout helpdesk* yang ditampilkan pada admin/manager.

Nama Aplikasi				
Jam Tambah Daftar Absen Lihat Absensi Closed Fallout Jumlah Penilaian Ubah Password Logout	Jumlah Closed Fallout			
	No.	Nama	Jabatan	Aksi
	1.	xxxxxxx	xxxxxxxxxxx	Lihat
	<div style="border: 1px solid black; width: 100px; height: 40px; margin: 0 auto;">Logo</div>			

GAMBAR: 3.39 Desain Menu Admin Melihat Jumlah *Closed Fallout*.

10. Tampilan menu tambah kriteria yang ditampilkan pada admin/manager.

Nama Aplikasi				
Jam Tambah Daftar Absen Lihat Absensi Closed Fallout Jumlah Penilaian Ubah Password Logout	Penilaian Karyawan			
	Kriteria			
	Tabel Tambah			
	ID	Nama	Bobot	Aksi
	1	xxxxxxx	xxxxxxxxxxx	Edit Hapus
	<div style="border: 1px solid black; width: 60px; height: 15px; margin: 0 auto;">Next</div> <div style="border: 1px solid black; width: 100px; height: 40px; margin: 10px auto;">Logo</div>			

GAMBAR: 3.40 Desain Menu Admin Menambah/Mengubah/Melihat Kriteria.

11. Tampilan menu tambah alternatif yang ditampilkan pada admin/*manager*.

Nama Aplikasi							
Jam	Penilaian Karyawan						
Tambah Daftar Absen Lihat Absensi Closed Fallout Jumlah Penilaian Ubah Password Logout	Karyawan						
	Tabel Tambah						
	<table border="1"> <thead> <tr> <th>ID</th> <th>Nama</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>xxxxxxx</td> <td>Edit Hapus</td> </tr> </tbody> </table>	ID	Nama	Aksi	1	xxxxxxx	Edit Hapus
	ID	Nama	Aksi				
	1	xxxxxxx	Edit Hapus				
Next							
Logo							

GAMBAR: 3.41 Desain Menu Admin Menambah/Mengubah/Melihat Alternatif.

12. Tampilan menu tambah nilai yang ditampilkan pada admin/*manager*.

Nama Aplikasi							
Jam	Penilaian Karyawan						
Tambah Daftar Absen Lihat Absensi Closed Fallout Jumlah Penilaian Ubah Password Logout	Nilai						
	Tabel Tambah						
	<table border="1"> <thead> <tr> <th>ID</th> <th>Nilai</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>xxxxxxx</td> <td>Edit Hapus</td> </tr> </tbody> </table>	ID	Nilai	Aksi	1	xxxxxxx	Edit Hapus
	ID	Nilai	Aksi				
	1	xxxxxxx	Edit Hapus				
Next							
Logo							

GAMBAR: 3.42 Desain Menu Admin Menambah/Mengubah/Melihat Nilai.

13. Tampilan menu pemberian nilai yang ditampilkan pada admin/*manager*.

Nama Aplikasi											
Jam	Penilaian Karyawan										
Tambah Daftar Absen Lihat Absensi Closed Fallout Jumlah Penilaian Ubah Password Logout	Pemberian Nilai										
	Tabel Tambah										
	<table border="1"> <thead> <tr> <th>ID</th> <th>Nama</th> <th>Kriteria</th> <th>Nilai</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>xxxxxxx</td> <td>xxxxxxx</td> <td>xxxxxx</td> <td>Hapus</td> </tr> </tbody> </table>	ID	Nama	Kriteria	Nilai	Aksi	1	xxxxxxx	xxxxxxx	xxxxxx	Hapus
	ID	Nama	Kriteria	Nilai	Aksi						
	1	xxxxxxx	xxxxxxx	xxxxxx	Hapus						
Next											
Logo											

GAMBAR: 3.43 Desain Menu Admin Memberikan Nilai.

14. Tampilan menu hasil penilaian yang ditampilkan pada admin/*manager*.

Nama Aplikasi									
Jam	Penilaian Karyawan								
Tambah Daftar Absen Lihat Absensi Closed Fallout Jumlah Penilaian Ubah Password Logout	Hasil								
	Nilai Preferensi								
	<table border="1"> <thead> <tr> <th>ID</th> <th>Alternatif</th> <th>Nama</th> <th>Vi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>xxxxxxx</td> <td>xxxxxxx</td> <td>xxxxxxx</td> </tr> </tbody> </table>	ID	Alternatif	Nama	Vi	1	xxxxxxx	xxxxxxx	xxxxxxx
	ID	Alternatif	Nama	Vi					
	1	xxxxxxx	xxxxxxx	xxxxxxx					
Selesai									
Logo									

GAMBAR: 3.44 Desain Menu Admin Memperoleh Hasil Penilaian.

15. Tampilan menu ubah katasandi yang ditampilkan pada admin/*manager*.

Nama Aplikasi	
<input type="button" value="Jam"/>	Ubah Password
Tambah	Nama <input type="text"/>
Daftar	Katasandi baru <input type="text"/>
Absen	Ulangi <input type="text"/>
Lihat Absensi	<input type="button" value="ubah"/>
Closed Fallout	
Jumlah	
Penilaian	
Ubah Password	
Logout	<input type="button" value="Logo"/>

GAMBAR: 3.45 Desain Menu Admin Mengubah *Password*.

BAB IV

IMPLEMENTASI DAN UJI COBA

4.1 Implementasi

4.1.1 Implementasi Perangkat Keras

Sesuai dengan apa yang dijabarkan pada bab 3, perangkat keras yang digunakan dalam penerapan aplikasi ini memiliki minimum spesifikasi sebagai berikut :

1. *Processor : Dual-Core 2.7 Ghz.*
2. *RAM : 2 GB DDR3*
3. *Disk : Free Space 30 GB*
4. *Graphic Card : Graphics card that support DirectX 9.0c 128 MB*
5. *Modem : HSDPA 7.2 Mbps/ HSUPA 5.76 Mbps*

4.1.2 Implementasi Perangkat Lunak

Adapun perangkat lunak yang dibutuhkan guna melancarkan penerapan aplikasi ini, antara lain :

1. Sistem Operasi *Windows 7, 8, 8.1 atau 10.*
2. *Browser.*
3. *XAMPP*

4.1.3 Implementasi Database

Sesuai dengan apa yang dijabarkan pada bab 3, berikut adalah implementasi pembuatan tabel pada database yang digunakan oleh aplikasi :

1. Struktur table *user* pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_user	int(10)			No	None
<input type="checkbox"/> 2	email_user	varchar(255)	latin1_swedish_ci		No	None
<input type="checkbox"/> 3	pwd_user	varchar(255)	latin1_swedish_ci		No	None
<input type="checkbox"/> 4	level_user	enum('hd', 'adm')	latin1_swedish_ci		No	None

GAMBAR: 4.1 Tabel *user*.

2. Struktur tabel *detail user* pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_user	int(10)			No	None
<input type="checkbox"/> 2	nis_user	int(25)			No	None
<input type="checkbox"/> 3	name_user	varchar(255)	latin1_swedish_ci		No	None
<input type="checkbox"/> 4	jbtn_user	varchar(255)	latin1_swedish_ci		No	None
<input type="checkbox"/> 5	jk_user	varchar(5)	latin1_swedish_ci		No	None

GAMBAR: 4.2 Tabel *detail user*.

3. Struktur tabel *closed fallout* pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_fallout	int(10)			No	None
<input type="checkbox"/> 2	id_user	int(10)			No	None
<input type="checkbox"/> 3	id_bln	int(10)			No	None
<input type="checkbox"/> 4	id_hri	int(10)			No	None
<input type="checkbox"/> 5	id_tgl	int(10)			No	None
<input type="checkbox"/> 6	closed_fallout	longtext	latin1_swedish_ci		No	None
<input type="checkbox"/> 7	status_fallout	enum('Menunggu', 'Dikonfirmasi', 'Ditolak')	latin1_swedish_ci		No	None

GAMBAR: 4.3 Tabel *closed fallout*.

4. Struktur tabel *absensi* pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_absen 🔑	int(11)		No	None
<input type="checkbox"/>	2	id_user	varchar(100)	latin1_swedish_ci	No	None
<input type="checkbox"/>	3	id_bln	int(10)		No	None
<input type="checkbox"/>	4	id_hri	int(10)		No	None
<input type="checkbox"/>	5	id_tgl	int(10)		No	None
<input type="checkbox"/>	6	jam_msk	varchar(50)	latin1_swedish_ci	No	None
<input type="checkbox"/>	7	st_jam_msk	enum('Menunggu', 'Dikonfirmasi', 'Ditolak')	latin1_swedish_ci	No	None
<input type="checkbox"/>	8	jam_klr	varchar(50)	latin1_swedish_ci	No	None
<input type="checkbox"/>	9	st_jam_klr	enum('Belum Absen', 'Menunggu', 'Dikonfirmasi', 'D...')	latin1_swedish_ci	No	None

GAMBAR: 4.4 Tabel *absensi*.

5. Struktur tabel kriteria pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_kriteria 🔑	varchar(10)	latin1_swedish_ci	No	None
<input type="checkbox"/>	2	nama_kriteria	varchar(50)	latin1_swedish_ci	No	None
<input type="checkbox"/>	3	bobot	float		No	None

GAMBAR: 4.5 Tabel kriteria.

6. Struktur tabel alternatif pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_alternatif 🔑	varchar(10)	latin1_swedish_ci	No	None
<input type="checkbox"/>	2	nama_alternatif	varchar(50)	latin1_swedish_ci	No	None



GAMBAR: 4.6 Tabel alternatif.

7. Struktur tabel nilai pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/>	1	id_nilai 🔑	varchar(10)	latin1_swedish_ci	No	None
<input type="checkbox"/>	2	nilai	varchar(10)	latin1_swedish_ci	No	None


GAMBAR: 4.7 Tabel nilai.

8. Struktur tabel penilaian pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_alternatif 	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/> 2	id_kriteria 	varchar(10)	latin1_swedish_ci		No	None
<input type="checkbox"/> 3	nilai	float			No	None


GAMBAR: 4.8 Tabel penilaian.

9. Struktur tabel hari pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_hri 	int(10)			No	None
<input type="checkbox"/> 2	nama_hri	varchar(50)	latin1_swedish_ci		No	None


GAMBAR: 4.9 Tabel hari.

10. Struktur tabel tanggal pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_tgl 	int(10)			No	None
<input type="checkbox"/> 2	nama_tgl	varchar(20)	latin1_swedish_ci		No	None

GAMBAR: 4.10 Tabel tanggal.

11. Struktur tabel bulan pada database yang digunakan.

#	Name	Type	Collation	Attributes	Null	Default
<input type="checkbox"/> 1	id_bln 	int(10)			No	None
<input type="checkbox"/> 2	nama_bln	varchar(25)	latin1_swedish_ci		No	None

GAMBAR: 4.11 Tabel bulan.

4.1.4 Implementasi Antarmuka

Pada implementasi ini penulis menyesuaikan tampilan yang ada pada aplikasi dengan rancangan tampilan yang ada pada bab 3. Berikut adalah tampilan yang ada pada aplikasi yang telah dibuat :

1. Menu yang di tampilkan oleh sistem pada saat *user* akan melakukan login dan mencoba untuk mengakses sistem.



GAMBAR: 4.12 Halaman Login.

2. Pada saat *user* dengan level *helpdesk* berhasil login, sistem akan menampilkan halaman absensi, dimana *helpdesk* diharapkan untuk tidak lupa mengisi kehadiran maupu absen pulang.



GAMBAR: 4.13 Halaman Absensi *Helpdesk*.

3. Pada halaman absensiku, *helpdesk* dapat melihat riwayat absensi yang telah dilakukan, dimulai dari absen masuk hingga absen pulang beserta dengan tanggal dan jamnya.

SPK Penilaian Kinerja Helpdesk Data Management

9:47 WIB

Absen ①

Absensiku

Closed Fallout Per Hari

Jumlah Closed Fallout

Keluar

Welcome, Prasetya

Absensiku - Mei

No	Hari, Tanggal	Jam Masuk	Jam Keluar
1	Selasa, 05 Mei 2020	00.19 WIB	00.19 WIB
2	Selasa, 19 Mei 2020	06.40 WIB	06.46 WIB

IndiHome
FIBER»»

GAMBAR: 4.14 Halaman Absensiku *Helpdesk*.

4. Pada halaman *closed fallout*, *helpdesk* dapat memasukan jumlah perolehan *fallout* yang telah dikerjakan perharinya.

SPK Penilaian Kinerja

20:34 WIB

Absen ①

Absensiku

Closed Fallout

Jumlah Closed Fallout

Keluar

Closed Fallout Hari ini

Simpan

IndiHome
FIBER»»

GAMBAR: 4.15 Halaman *Closed Fallout Helpdesk*.

5. Pada menu Jumlah *closed fallout*, *helpdesk* dapat melihat riwayat perolehan fallout secara keseluruhan lengkap beserta dengan hari, tanggal dan bulannya.

SPK Penilaian Kinerja Helpdesk Data Management

9:47 WIB

Absen 📌

Absensiku

Closed Fallout Per Hari

Jumlah Closed Fallout

Keluar

Jumlah Closed Fallout Tambah

Closed Fallout - Mei
Jumlah Closed Fallout : 2648

No	Hari, Tanggal	Closed Fallout
1	Selasa, 05 Mei 2020	90
2	Kamis, 14 Mei 2020	542
3	Selasa, 19 Mei 2020	200
4	Selasa, 19 Mei 2020	150
5	Selasa, 19 Mei 2020	123
6	Selasa, 19 Mei 2020	55
7	Selasa, 19 Mei 2020	55
8	Selasa, 19 Mei 2020	1433

FIBER

GAMBAR: 4.16 Halaman Jumlah *Closed Fallout Helpdesk*.

6. Pada menu tambah karyawan ini hanya ada di user admin/*manager*, dimana seorang admin dapat menambahkan *helpdesk* sekaligus *user* agar dapat mengakses sistem.

SPK Penilaian Kinerja Helpdesk Data Management

9:48 WIB

Tambah Karyawan

Daftar Karyawan

Lihat Absensi Karyawan

Jumlah Closed Fallout

Penilaian Karyawan

Ubah Katasandi

Keluar

Tambah Karyawan

NIK:

Nama Lengkap:

Jenis Kelamin: Laki-laki Perempuan

Jabatan:

Email:

Katasandi:

FIBER

GAMBAR: 4.17 Halaman Tambah *Helpdesk*.

7. Pada halaman daftar *helpdesk*, admin dapat melihat secara keseluruhan data *helpdesk* yang terdaftar di sistem, serta dapat mengubah atau menghapus

SPK Penilaian Kinerja Helpdesk Data Management

9:49 WIB

Tambah Karyawan

Daftar Karyawan

Lihat Absensi Karyawan

Jumlah Closed Fallout

Penilaian Karyawan

Ubah Katasandi

Keluar

Daftar Karyawan

No	Nama Karyawan	Jabatan	Edit
1	Prasetya	Helpdesk	Edit info • Hapus Karyawan
2	Yuma	Helpdesk	Edit info • Hapus Karyawan
3	Carli	Helpdesk	Edit info • Hapus Karyawan
4	Deny	Helpdesk	Edit info • Hapus Karyawan
5	Ricky	Helpdesk	Edit info • Hapus Karyawan
6	Novia	Helpdesk	Edit info • Hapus Karyawan

IndiHome
FIBER»»

GAMBAR: 4.18 Halaman Daftar *Helpdesk*.

8. Pada halaman detail absensi ini, admin dapat melihat secara keseluruhan data absensi yang telah dilakukan oleh *helpdesk*.

SPK Penilaian Kinerja Helpdesk Data Management

9:49 WIB

Tambah Karyawan

Daftar Karyawan

Lihat Absensi Karyawan

Jumlah Closed Fallout

Penilaian Karyawan

Ubah Katasandi

Keluar

Lihat Absensi Karyawan

No	Nama Karyawan	Jabatan	Lihat
1	Prasetya	Helpdesk	Lihat Absensi
2	Yuma	Helpdesk	Lihat Absensi
3	Carli	Helpdesk	Lihat Absensi
4	Deny	Helpdesk	Lihat Absensi
5	Ricky	Helpdesk	Lihat Absensi
6	Novia	Helpdesk	Lihat Absensi

IndiHome
FIBER»»

GAMBAR: 4.19 Halaman Lihat Absensi *Helpdesk*.

9. Pada halaman ini, admin dapat melihat secara keseluruhan perolehan *fallout* dari setiap *helpdesk*.

SPK Penilaian Kinerja Helpdesk Data Management

9:50 WIB

Tambah Karyawan
Daftar Karyawan
Lihat Absensi Karyawan
Jumlah Closed Fallout
Penilaian Karyawan
Ubah Katasandi
Keluar

Jumlah Closed Fallout

No	Nama Karyawan	Jabatan	Lihat
1	Prasetya	Helpdesk	Lihat Jumlah
2	Yuma	Helpdesk	Lihat Jumlah
3	Carli	Helpdesk	Lihat Jumlah
4	Deny	Helpdesk	Lihat Jumlah
5	Ricky	Helpdesk	Lihat Jumlah
6	Novia	Helpdesk	Lihat Jumlah

IndiHome
FIBER»»

GAMBAR: 4.20 Halaman Jumlah *Closed Fallout Helpdesk*.

10. Pada halaman ini, admin dapat menambahkan, menghapus, ataupun mengedit kriteria yang dibutuhkan dalam penilaian kinerja *helpdesk*.

SPK Penilaian Kinerja Helpdesk Data Management

9:58 WIB

Tambah Karyawan
Daftar Karyawan
Lihat Absensi Karyawan
Jumlah Closed Fallout
Penilaian Karyawan
Ubah Katasandi
Keluar

Penilaian Karyawan

Kriteria

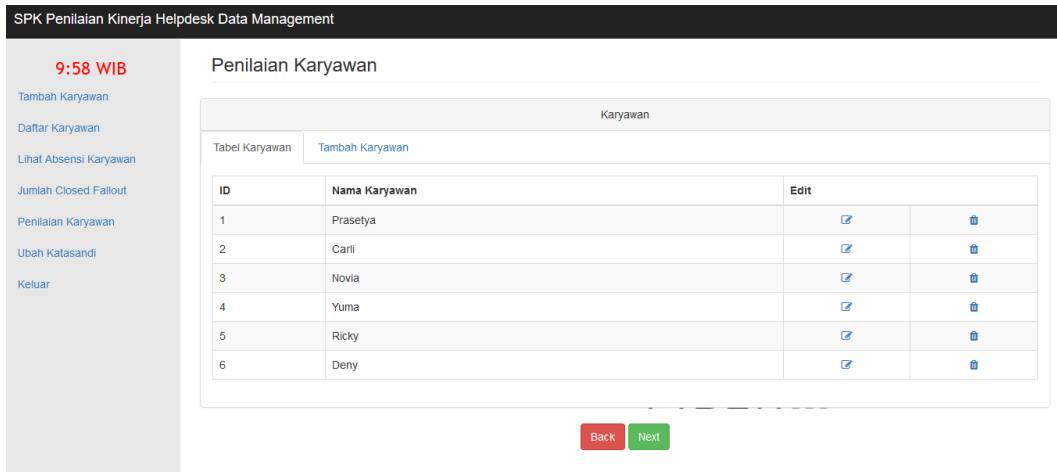
Tabel Kriteria [Tambah Kriteria](#)

ID Kriteria	Nama Kriteria	Bobot	Edit
1	Kehadiran	0.2	✎ ✖
2	Kerapihan	0.2	✎ ✖
3	Kinerja	0.3	✎ ✖
4	Lama	0.3	✎ ✖

Next FIBER»»

GAMBAR: 4.21 Halaman Tambah Kriteria.

11. Pada halaman ini, admin dapat menambah, mengubah, ataupun menghapus alternatif karyawan/*helpdesk* yang menjadi kandidat dalam penilaian yang nantinya akan dilakukan.



The screenshot shows the 'Penilaian Karyawan' (Employee Evaluation) page. The header is 'SPK Penilaian Kinerja Helpdesk Data Management'. The page title is 'Penilaian Karyawan'. On the left sidebar, there is a clock showing '9:58 WIB' and a list of navigation links: 'Tambah Karyawan', 'Daftar Karyawan', 'Lihat Absensi Karyawan', 'Jumlah Closed Fallout', 'Penilaian Karyawan', 'Ubah Katasandi', and 'Keluar'. The main content area has a tab 'Karyawan' and a sub-tab 'Tambah Karyawan'. Below this is a table with columns 'ID', 'Nama Karyawan', and 'Edit'. The table contains six rows of employee data.

ID	Nama Karyawan	Edit
1	Prasetya	Edit Delete
2	Carli	Edit Delete
3	Novia	Edit Delete
4	Yuma	Edit Delete
5	Ricky	Edit Delete
6	Deny	Edit Delete

At the bottom of the table, there are two buttons: 'Back' (red) and 'Next' (green).

GAMBAR: 4.22 Halaman Tambah Alternatif.

12. Pada halaman ini, admin dapat menambah, menghapus, dan mengubah nilai yang nantinya akan diberikan kepada setiap *helpdesk* sesuai dengan kriteria yang di tentukan.



The screenshot shows the 'Penilaian Karyawan' (Employee Evaluation) page. The header is 'SPK Penilaian Kinerja Helpdesk Data Management'. The page title is 'Penilaian Karyawan'. On the left sidebar, there is a clock showing '9:59 WIB' and a list of navigation links: 'Tambah Karyawan', 'Daftar Karyawan', 'Lihat Absensi Karyawan', 'Jumlah Closed Fallout', 'Penilaian Karyawan', 'Ubah Katasandi', and 'Keluar'. The main content area has a tab 'Nilai' and a sub-tab 'Tambah Nilai'. Below this is a table with columns 'ID Nilai', 'Nilai', and 'Edit'. The table contains five rows of evaluation data.

ID Nilai	Nilai	Edit
1	1	Edit Delete
2	2	Edit Delete
3	3	Edit Delete
4	4	Edit Delete
5	5	Edit Delete

At the bottom of the table, there are two buttons: 'Back' (red) and 'Next' (green). Below the buttons is the 'FIBER' logo.

GAMBAR: 4.23 Halaman Tambah Nilai.

13. Pada halaman ini, admin dapat memberikan nilai kepada setiap *helpdesk* sesuai dengan kriteria yang ada agar dapat dilakukan penilaian lebih lanjut.

SPK Penilaian Kinerja Helpdesk Data Management

10:0 WIB

- Tambah Karyawan
- Daftar Karyawan
- Lihat Absensi Karyawan
- Jumlah Closed Fallout
- Penilaian Karyawan
- Ubah Katasandi
- Keluar

13	Prasetya	Kinerja	5	
14	Carli	Kinerja	4	
15	Novia	Kinerja	5	
16	Yuma	Kinerja	4	
17	Ricky	Kinerja	3	
18	Deny	Kinerja	3	
19	Prasetya	Lama Kerja	4	
20	Carli	Lama Kerja	4	
21	Novia	Lama Kerja	4	
22	Yuma	Lama Kerja	3	
23	Ricky	Lama Kerja	4	
24	Deny	Lama Kerja	4	

Back Next

GAMBAR: 4.24 Halaman Beri Nilai.

14. Pada halaman ini, proses penilaian telah dilakukan oleh sistem, sehingga admin dapat melihat hasil penilaian yang dilakukan oleh sistem dan dapat melihat siapakan yang memiliki nilai tertinggi.

SPK Penilaian Kinerja Helpdesk Data Management

10:0 WIB

- Tambah Karyawan
- Daftar Karyawan
- Lihat Absensi Karyawan
- Jumlah Closed Fallout
- Penilaian Karyawan
- Ubah Katasandi
- Keluar

5	A5	Ricky	0.05
6	A6	Deny	0.032

Nilai Preferensi(V_i)

No	Alternatif	Nama	V _i
1	A1	Prasetya	1.000
2	A2	Carli	0.790
3	A3	Novia	0.758
4	A4	Yuma	0.358
5	A5	Ricky	0.408
6	A6	Deny	0.166

Finish

GAMBAR: 4.25 Halaman Hasil Penilaian.

15. Pada halaman ini, admin dapat mengubah *password* milik admin itu sendiri agar dapat meminimalisir resiko untuk pencurian data admin.

GAMBAR: 4.26 Halaman Ubah Katasandi.

4.2 Uji Coba

Metode uji coba yang digunakan oleh penulis yaitu *Black Box Testing*. Metode *Black Box* ini menguji fungsi-fungsi yang ada pada aplikasi secara keseluruhan. Berikut adalah hasil dari pengujian yang dilakukan terhadap aplikasi ini :

1. Hasil pengujian pada menu login.

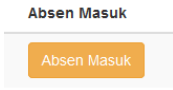
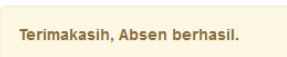
TABEL: 4.1 Pengujian Login.

No.	Skenario Pengujian	Yang diharapkan	Output	Kesimpulan
1.	Form email tidak di isikan email. <i>Test Case :</i>	Sistem memerintahkan untuk mengisikan	Sesuai yang diharapkan. <i>Test Result :</i>	Valid

		email pada form email.		
2.	Menggunakan email yg benar & katasandi yang salah atau sebaliknya. <i>Test Case :</i> 	Sistem memerintahkan untuk memeriksa kembali email & katasandi,	Sesuai yang diharapkan. <i>Test Result :</i> Periksa kembali email & katasandi Anda! 	Valid
3.	Menggunakan email yang & katasandi yang benar. <i>Test Case :</i> 	Login berhasil dan sistem akan menampilkan halaman aplikasi sesuai dengan level nya.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

2. Hasil pengujian pada menu absensi & absensiku *helpdesk*.

TABEL: 4.2 Pengujian Absensi & Absensiku *Helpdesk*.

No.	Skenario Pengujian	Yang diharapkan	Output	Kesimpulan
1.	Mengklik tombol absen masuk. <i>Test Case :</i> 	Sistem memberitahukan absen telah berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
2.	Mengklik tombol absen pulang. <i>Test Case :</i>	Sistem memberitahukan absen telah berhasil & absen hari ini selesai.	Sesuai yang diharapkan. <i>Test Result :</i>	Valid

				
3.	Mengklik Menu absensiku. <i>Test Case :</i> 	Sistem menampilkan detail absensi.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

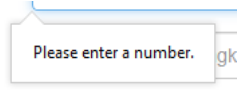
3. Hasil pengujian pada menu *closed fallout helpdesk* & jumlah *fallout helpdesk*.





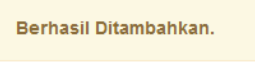

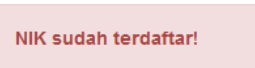
TABEL: 4.3 Pengujian Jumlah & *Closed Fallout Helpdesk*.

No.	Skenario Pengujian	Yang diharapkan	Output	Kesimpulan
1.	Menginputkan <i>closed fallout</i> . <i>Test Case :</i> 	Sistem memberitahukan data berhasil disimpan.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
2.	Mengklik menu jumlah <i>closed fallout</i> <i>Test Case :</i> 	Sistem menampilkan jumlah <i>closed fallout</i> .	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

4. Hasil pengujian pada menu tambah karyawan.

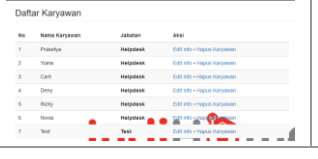

TABEL: 4.4 Pengujian Tambah Karyawan.

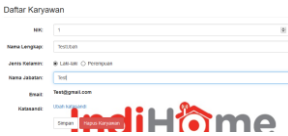
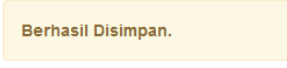
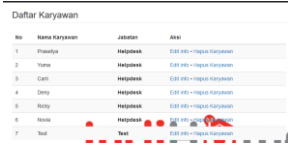
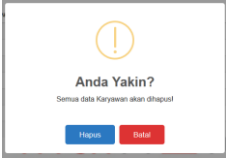
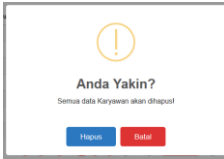
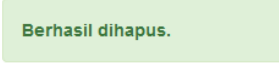
No.	Skenario Pengujian	Yang diharapkan	Output	Kesimpulan
1.	Tidak mengisikan data apapun/ada form yang tidak terisi. <i>Test Case :</i>	Sistem memerintahkan untuk mengisikan data.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

				
2.	<p>Mereset data yang telah di masukan ke form.</p> <p><i>Test Case :</i></p> 	Form pengisian akan menjadi kosong kembali.	<p>Sesuai yang diharapkan.</p> <p><i>Test Result :</i></p> 	Valid
3.	<p>Menginputkan data karyawan.</p> <p><i>Test Case :</i></p> 	Sistem memberitahukan data telah berhasil ditambahkan.	<p>Sesuai yang diharapkan.</p> <p><i>Test Result :</i></p> 	Valid
4.	<p>Menginputkan data yang sudah ada.</p> <p><i>Test Case :</i></p> 	Sistem memberitahukan bahwa data tersebut sudah ada.	<p>Sesuai yang diharapkan.</p> <p><i>Test Result :</i></p> 	Valid

5. Hasil pengujian pada menu daftar karyawan.

TABEL: 4.5 Pengujian Daftar Karyawan.

No.	Skenario Pengujian	Yang diharapkan	Output	Kesimpulan
1.	<p>Mengklik edit info pada salah satu karyawan.</p> <p><i>Test Case :</i></p> 	Sistem menampilkan form ubah data karyawan.	<p>Sesuai yang diharapkan.</p> <p><i>Test Result :</i></p> 	Valid

2.	Mengubah data karyawan. <i>Test Case :</i> 	Sistem memberitahukan data telah berhasil di simpan.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
3.	Mengklik hapus data salah satu karyawan. <i>Test Case :</i> 	Sistem mengkonfirmasi apakah data akan dihapus.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
4.	Menghapus data karyawan. <i>Test Case :</i> 	Sistem memberitahukan bahwa data berhasil dihapus.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

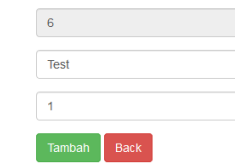
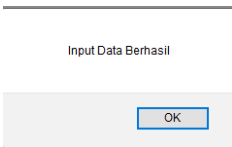
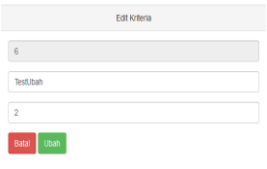


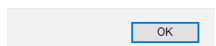
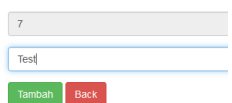
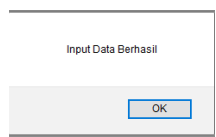
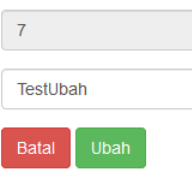

6. Hasil pengujian pada menu lihat jumlah *fallout* & lihat absen admin/*manager*.


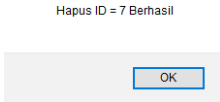
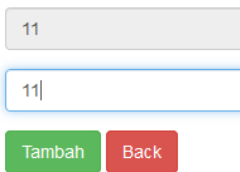
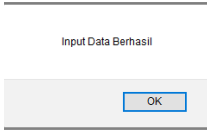
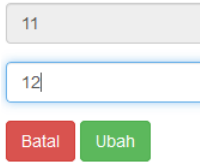
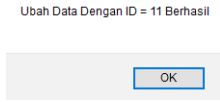

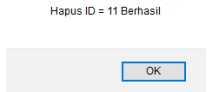
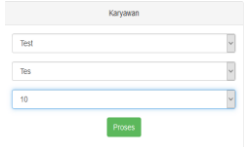
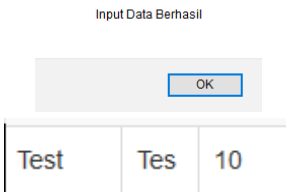

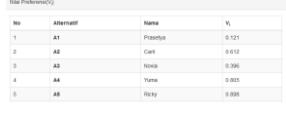
TABEL: 4.6 Pengujian Lihat Jumlah *Fallout* & Lihat Absen Admin/*Manager*.

No.	Skenario Pengujian	Yang diharapkan	Output	Kesimpulan
1.	Melihat jumlah <i>closed fallout</i> . <i>Test Case :</i> 	Sistem menampilkan jumlah <i>closed fallout</i> .	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
2.	Melihat absensi salah satu <i>helpdesk</i> . <i>Test Case :</i> 	Menampilkan detail absensi <i>helpdesk</i> .	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

7. Hasil pengujian pada menu penilaian karyawan.

TABEL: 4.7 Pengujian Penilaian Karyawan.

No.	Skenario Pengujian	Yang diharapkan	Output	Kesimpulan
1.	Menambahkan kriteria. <i>Test Case :</i> 	Sistem memberitahukan input data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
2.	Mengubah kriteria. <i>Test Case :</i> 	Sistem memberitahukan ubah data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
3.	Menghapus kriteria. <i>Test Case :</i> 	Sistem memberitahukan hapus data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
4.	Menambahkan alternatif karyawan. <i>Test Case :</i> 	Sistem memberitahukan input data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
5.	Mengubah alternatif karyawan. <i>Test Case :</i> 	Sistem memberitahukan ubah data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

6.	Menghapus alternatif karyawan. <i>Test Case :</i> 	Sistem memberitahukan hapus data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
7.	Menambah nilai. <i>Test Case :</i> 	Sistem memberitahukan input data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
8.	Mengubah nilai. <i>Test Case :</i> 	Sistem memberitahukan ubah data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
9.	Menghapus nilai. <i>Test Case :</i> 	Sistem memberitahukan hapus data berhasil.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
10.	Memberikan penilaian. <i>Test Case :</i> 	Sistem memberitahukan penilaian berhasil di input & menampilkan tabel penilaian.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid
11.	Melakukan penilaian. <i>Test Case :</i> 	Sistem menampilkan hasil penilaian.	Sesuai yang diharapkan. <i>Test Result :</i> 	Valid

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil penulisan dan penelitian yang penulis lakukan, dapat disimpulkan bahwa :

1. Aplikasi sistem pendukung keputusan ini dapat membantu *manager* untuk mendapatkan karyawan terbaik sesuai dengan penilaian yang dilakukan oleh sistem.
2. *Manager* tidak perlu mengkhawatirkan akan adanya kesalahan perhitungan dalam penilaian karyawan.

5.2 Saran

Berdasarkan dengan apa yang telah penulis bahas, aplikasi ini masih ada beberapa kekurangan. Adapun saran dari penulis untuk perkembangan sistem pendukung keputusan ini adalah :

1. Sistem dapat dikembangkan lebih lanjut dengan memberikan fungsi lampiran pada laporan *closed fallout*.
2. Sistem dapat dikembangkan lebih lanjut dengan menambahkan fungsi *export* laporan penilaian yang telah dilakukan oleh sistem.

DAFTAR PUSTAKA

- Edi, D. and Betshani, S. (2012) 'Analisis Data dengan Menggunakan ERD dan Model Konseptual Data Warehouse', *Jurnal Informatika*, 5(1), pp. 71–85.
- Gunawan, H. and Alamsyah, S. (2019) 'Sistem Pendukung Keputusan Untuk Menentukan Pemain Terbaik Dalam Satu Musim Kompetisi Sepakbola Dengan Menggunakan Metode Simple Additive Weighting', *INFORMASI (Jurnal Informatika dan Sistem Informasi)*, 11(2), pp. 13–32. doi: 10.37424/informasi.v11i2.14.
- Haqi, B. (2019) *Aplikasi SPK Pemilihan Dosen Terbaik Metode Simple Additive Weighting (SAW) Dengan Java*.
- Hertyana, H. (2018) 'Sistem Pendukung Keputusan Penentuan Karyawan Terbaik Menggunakan Metode Saw Studi Kasus Amik Mahaputra Riau', *Jurnal Intra-Tech*, 2(1), pp. 73–82.
- Kawano, K., Umemura, Y. and Kano, Y. (1983) 'Field Assessment and Inheritance of Cassava Resistance to Superelongation Disease 1', *Crop Science*, 23(2), pp. 201–205. doi: 10.2135/cropsci1983.0011183x002300020002x.
- Kurnia, Y. _ (2018) 'Penilaian Kinerja Guru Menggunakan Metode Topsis', *JSAI (Journal Scientific and Applied Informatics)*, 1(3), pp. 70–75. doi: 10.36085/jsai.v1i3.63.
- Mulyani, S. (2016) *Metode Analisis dan Perancangan Sistem*, ISBN: 978-979-19906-2-2.
- Muttaqin, F. (2014) 'ANALISIS DAN DESAIN SISTEM INFORMASI BERBASIS KOMPUTER UNTUK PERSEDIAAN BARANG PADA TOKO BAHAN BANGUNAN (Studi Kasus pada UD. Sumber Bumi Subur)', *Jurnal Administrasi Bisnis S1 Universitas Brawijaya*, 8(1), p. 79320.
- Nababan, D. and Rahim, R. (2018) 'Sistem Pendukung Keputusan Reward Bonus Karyawan Dengan Metode Topsis', *Simetris*, 3(6), pp. 2–6.
- Nofriansyah, D. (2018) *Konsep Data Mining Vs Sistem Pendukung Keputusan*. PT Deep Publish.
- Prof. Dr. Sri Mulyani, Ak., C. (2016) *Analisis dan Perancangan Sistem Informasi Manajemen Keuangan Daerah: Notasi Permodelan Unifed Modeling Language, Metode Analisis Dan Perancangan Sistem*.
- Putri, M. P. and Effendi, H. (2018) 'Implementasi Metode Rapid Application Development Pada Website Service Guide "Waterfall Tour South Sumatera"', *Jurnal SISFOKOM*, 07(September), pp. 130–136.
- Sari, D. R. et al. (2018) 'Decision Support System for Thesis Graduation

- Recommendation Using AHP-TOPSIS Method’, *Jurnal Teknologi dan Sistem Komputer*, 6(1), pp. 1–6. doi: 10.14710/jtsiskom.6.1.2018.1-6.
- Sari, R. E. *et al.* (2014) ‘Menggunakan Metode Ahp (Studi Kasus : Di Stmik Potensi Utama Medan)’, *Stmik*, pp. 108–114.
- Suharsimi Arikunto (2004) *Metode penelitian kepustakaan - Mestika Zed.*
- Sukamto, R. A. and Shalahuddin, M. (2016) *Rekayasa Perangkat Lunak (Terstruktur dan berbasis objek), Sdlc.*
- Sunalia, A., Abdillah, L. A. and Suyanto (2017) ‘Sistem Pendukung Keputusan Penentuan Penerima Beasiswa Pada Smk Madyatama Dengan Metode Topsis’, *Sentikom2017*, pp. 13–18.
- Surya, C. (2018) ‘Penilaian Kinerja Dosen Menggunakan Metode TOPSIS (Studi Kasus : Amik Mitra Gama)’, *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 2(1), pp. 322–329. doi: 10.29207/resti.v2i1.119.
- Sutabri, T. (2012) *Konsep Sistem Informasi.*
- Tarmuji, A. (2013) ‘Tinjauan Umum Tentang Helpdesk Dan Framework Terkait’, *Jurnal Informatika*, 2(1), pp. 146–157. doi: 10.26555/jifo.v2i1.a5228.
- Todhi Pristianto (2011) ‘SISTEM PENDUKUNG KEPUTUSAN MEMILIH PERGURUAN TINGGI SWASTA JURUSAN KOMPUTER MENGGUNAKAN METODE Analytical Hierarchy Process (AHP)’, *Jurnal Informatika dan Sistem Informasi*, XI(1), pp. 1–32.
- Wahyuni, E. G. and Anggoro, A. T. (2017) ‘Sistem Pendukung Keputusan Penerimaan Pegawai dengan Metode TOPSIS’, *Sains Teknologi dan Industri*, 14(2), pp. 108–116. Available at: <http://ejournal.uin-suska.ac.id/index.php/sitekin/article/view/3907/pdf>.
- Wikipedia (2020) *Sistem, Web Page.* Available at: <https://id.wikipedia.org/wiki/Sistem>.

LAMPIRAN

Listing Program.

```
<head>
  <title>Login</title>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<div class="login-page">
  <div class="form">
    
    <form class="login-form" method="POST"
action="model/proses.php">
      <?php
        if (isset($_GET['log'])) {
          if ($_GET['log']==2) {
            echo "<div class='alert alert-
danger'><strong>Periksa kembali email & katasandi
Anda!</strong></div><br>";
          }
        }
      >
      <br><br>
      <input type="email" placeholder="Email" name="email" data-
validate="Email Kosong!"/>
      <input type="password" placeholder="Password" name="pwd"/>
      <button type="submit" name="login">login</button>
    </form>
  </div>
</div>
<script src="java.js"></script>


<?php
include '../lib/db/dbconfig.php';
if (isset($_POST['add_pbXialZasww12Q'])) {
  $query = $conn->query("SELECT*FROM user");
  $id = $query->num_rows+1;
  $email = mysqli_real_escape_string($conn,
$_POST['email']);
  $pwd = mysqli_real_escape_string($conn,
sha1($_POST['pwd']));
  $nama = mysqli_real_escape_string($conn,
$_POST['nama']);
  $level = "adm";
  $sql_user = "INSERT INTO user (id_user,
email_user,
pwd_user,
level_user) VALUES (?,
?,
?,
?)";
```

```

        $sql_detail = "INSERT INTO detail_pb (id_user,
            name_user) VALUES(?,
                ?)";
        $cek = $conn->query("SELECT*FROM user WHERE
email_user='$email'")->num_rows;
        if ($email==="" || $pwd==="" || $nama ==="") {
            header("location:../root/add_pb.php?st=4");
        } else {
            if ($cek==0) {
                if ($statement = $conn->prepare($sql_user)
AND $statement1 = $conn -> prepare($sql_detail)) {
                    $statement->bind_param(
                        "iss", $id, $email,
                        $pwd, $level
                    );
                    $statement1->bind_param(
                        "is", $id, $nama
                    );
                    if ($statement->execute() &&
$statement1->execute()) {
                        header("location:../root/add_pb.php?st=1");
                    } else {
                        header("location:../root/add_pb.php?st=2");
                    }
                }
            } else {
                header("location:../root/add_pb.php?st=3");
            }
        }
        $conn->close();
    } else {
        echo "<script>>window.alert('Waaahh.. Bandel ya !!
');window.location=('../home');</script>";
    }
?>



```


Penentuan kriteria, bobot dan range penilaian.


**Telkom
Indonesia**
the world in your hand

Berdasarkan dari hasil kesepakatan dan ketentuan perusahaan, telah ditetapkan bahwa kriteria, nilai bobot dan range penilaian untuk Aplikasi Sistem Pendukung Keputusan Penilaian Kinerja ini sebagai berikut :

1. Kriteria yang diperlukan adalah jumlah keterlambatan (kehadiran), jumlah penyelesaian fallout dan lama kerja.
2. Nilai bobot yang ditentukan yaitu kehadiran 30%, jumlah penyelesaian fallout 40% dan lama kerja 30%.
3. Range penilaian untuk masing-masing kriteria sebagai berikut :
 - a. Kehadiran.
 - i. Keterlambatan lebih dari 3 kali mendapatkan nilai 1.
 - ii. Keterlambatan sebanyak 3 kali mendapatkan nilai 2.
 - iii. Keterlambatan sebanyak 2 kali mendapatkan nilai 3.
 - iv. Keterlambatan sebanyak 1 kali mendapatkan nilai 4.
 - v. Tidak pernah terlambat mendapatkan nilai 5.
 - b. Jumlah Penyelesaian Fallout.
 - i. Memperoleh kurang dari 600 mendapatkan nilai 1.
 - ii. Memperoleh 600 sampai 900 mendapatkan nilai 2.
 - iii. Memperoleh 901 sampai 1200 mendapatkan nilai 3.
 - iv. Memperoleh 1201 sampai 1500 mendapatkan nilai 4.
 - v. Memperoleh lebih dari 1500 mendapatkan nilai 5.
 - c. Lama Kerja.
 - i. Telah bekerja kurang dari 1 tahun mendapatkan nilai 1.
 - ii. Telah bekerja selama 1 tahun mendapatkan nilai 2.
 - iii. Telah bekerja selama 2 tahun mendapatkan nilai 3.
 - iv. Telah bekerja selama 3 tahun mendapatkan nilai 4.
 - v. Telah bekerja selama 4 tahun atau lebih mendapatkan nilai 5.

Bandung, 5 Juni 2020
Yang menetapkan,


Muhammad Fauzan Abdurrahim

 KANTOR WITEL JABAR TENGAH
PT. Telekomunikasi Indonesia, Tbk.
Jl. Lembong No. 11 - 15
Bandung 40111 - Indonesia

Tlp : 022 454 0247
www.telkom.co.id

