



**PERANCANGAN PROGRAM PENGHITUNGAN GAJI DAN
LEMBUR PEGAWAI TAD (TENAGA AHLI DAYA)
MANDIRI UNIVERSITY MEDAN CAMPUS**

Disusun dan Diajukan untuk Memenuhi Persyaratan Ujian Akhir Memperoleh
Gelar Sarjana Komputer pada Fakultas Sains dan Teknologi
Universitas Pembangunan Panca Budi
Medan

SKRIPSI

OLEH:

NAMA : M. WENDY IRYANDI
NPM : 1414370149
PROGRAM STUDI : SISTEM KOMPUTER

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2021

PENGESAHAN SKRIPSI

JUDUL : PERANCANGAN PROGRAM PENGHITUNGAN GAJI DAN LEMBUR
PEGAWAI TAD (TENAGA AHLI DAYA) MANDIRI UNIVERSITY MEDAN
CAMPUS

NAMA : M. WENDY IRYANDI
N.P.M : 1414370149
FAKULTAS : SAINS & TEKNOLOGI
PROGRAM STUDI : Sistem Komputer
TANGGAL KELULUSAN : 28 Mei 2021

DIKETAHUI

DEKAN



Hamdani, ST., MT.

KETUA PROGRAM STUDI



Eko Hariyanto, S.Kom., M.Kom

**DISETUJUI
KOMISI PEMBIMBING**

PEMBIMBING I



Eko Hariyanto, S.Kom., M.Kom

PEMBIMBING II



Hermansyah, S.Kom, M.Kom

PERNYATAAN ORISINALITAS

Dengan ini menyatakan bahwa dalam skripsi ini tidak terdapat karya yang diajukan untuk memperoleh gelar kesarjanaan didalam perguruan tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis di dalam skripsi ini dan di sebut dalam daftar pustaka.

Medan, 01 Oktober 2021

Yang membuat pernyataan




M Wendy Iryandi

1414370149

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : M Wendy Iryandi

NPM : 1414370149

Prodi : Sistem Komputer

Konsentrasi : Sistem Informasi

Judul Skripsi : Perancangan Program Penghitungan Gaji dan Lembur Pegawai
TAD (Tenaga Ahli Daya) Mandiri University Medan Campus

Dengan ini menyatakan bahwa :

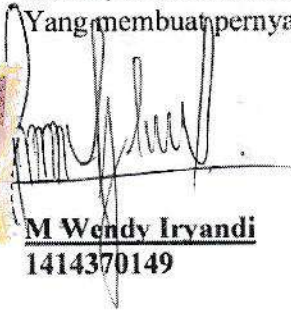
1. Tugas akhir/skripsi saya bukan hasil dari plagiat.
2. Saya tidak akan menuntut perbaikan nilai Indeks Prestasi Kumulatif (IPK)
3. Skripsi saya dapat dipublikasikan oleh pihak lembaga, dan saya tidak akan menuntut akibat publikasi tersebut.

Demikian pernyataan ini saya perbuat dengan sebenar-benarnya.

Terimakasih.

Medan, 01 Oktober 2021
Yang membuat pernyataan




M Wendy Iryandi
1414370149

SURAT PERNYATAAN

Yang Bertanda Tangan Dibawah Ini :

: M. WENDY IRYANDI
: 1414370149
Tgl. Lahir : Medan / 20 Juni 1996
: Jl Amal Gg Sehat No 146
: 081350447988
Orang Tua : ARYADI/IRMAYANI
: SAINS & TEKNOLOGI
Studi : Sistem Komputer
: Perancangan Program Penghitungan Gaji dan Lembur Pegawai TAD (Tenaga Ahli Daya) Mandiri University
: Medan Campus

dengan surat ini menyatakan dengan sebenar - benarnya bahwa data yang tertera diatas adalah sudah benar sesuai dengan data pendidikan terakhir yang saya jalani. Maka dengan ini saya tidak akan melakukan penuntutan kepada UNPAB. Apabila ada data pada ijazah saya.

Surat pernyataan ini saya buat dengan sebenar - benarnya, tanpa ada paksaan dari pihak manapun dan dibuat dalam kesadaran. Jika terjadi kesalahan, Maka saya bersedia bertanggung jawab atas kelalaian saya.

Medan, 01 Oktober 2021
Yang Membuat Pernyataan



[Handwritten Signature]
M. WENDY IRYANDI
1414370149



YAYASAN PROF. DR. H. KADIRUN YAHYA

UNIVERSITAS PEMBANGUNAN PANCA BUDI

JL. Jend. Gatot Subroto KM 4,5 PO. BOX 1099 Telp. 061-30106057 Fax. (061) 4514808
MEDAN - INDONESIA

Website : www.pancabudi.ac.id - Email : admin@pancabudi.ac.id

LEMBAR BUKTI BIMBINGAN SKRIPSI

Nama Mahasiswa : M. WENDY IRYANDI
 NPM : 1414370149
 Program Studi : Sistem Komputer
 Jenjang Pendidikan : Strata Satu
 Dosen Pembimbing : Eko Hariyanto, S.Kom., M.Kom
 Judul Skripsi : Perancangan Program Penghitungan Gaji dan Lembur Pegawai TAD (Tenaga Ahli Daya) Mandiri University Medan Campus

Tanggal	Pembahasan Materi	Status	Keterangan
23 Desember 2020	Acc seminar hasil	Disetujui	
05 Maret 2021	Acc sidang meja hijau	Disetujui	
09 Juni 2021	acc jilid skripsi	Disetujui	

Medan, 01 Oktober 2021
Dosen Pembimbing,



Eko Hariyanto, S.Kom., M.Kom



YAYASAN PROF. DR. H. KADIRUN YAHYA

UNIVERSITAS PEMBANGUNAN PANCA BUDI

JL. Jend. Gatot Subroto KM 4,5 PO. BOX 1099 Telp. 061-30106057 Fax. (061) 4514808
MEDAN - INDONESIA

Website : www.pancabudi.ac.id - Email : admin@pancabudi.ac.id

LEMBAR BUKTI BIMBINGAN SKRIPSI

Nama Mahasiswa : M. WENDY IRYANDI
 NPM : 1414370149
 Program Studi : Sistem Komputer
 Jenjang Pendidikan : Strata Satu
 Dosen Pembimbing : Hermansyah, S.Kom, M.Kom
 Judul Skripsi : Perancangan Program Penghitungan Gaji dan Lembur Pegawai TAD (Tenaga Ahli Daya) Mandiri University Medan Campus

Tanggal	Pembahasan Materi	Status Keterangan
03 Juli 2020	Tambahkan pengertian gaji , database, konteks diagram / uml dan sekilas tentang perusahaan.	Revisi
04 Agustus 2020	Lanjut BAB 3 dan DEMO Program dalam bentuk Video yang diterangkan cara sistemnya	Revisi
13 Agustus 2020	BAB 3: dibuat Metode Penelitiannya sebelum masuk ke analisa; No SUB BAB maupun Gambar/tabel bukan 1 angka, tapi Buat misal di bab 3 berarti no gambar/tabel 3.1. xxxx;	Revisi
06 Oktober 2020	Lanjut Bab 4	Revisi
04 Desember 2020	Lengkapi Semua, ACC Seminar HASIL	Disetujui
04 Maret 2021	Acc Sidang Meja Hijau	Disetujui
08 Juni 2021	ACC JILID SKRIPSI	Disetujui

Medan, 01 Oktober 2021
Dosen Pembimbing,



Hermansyah, S.Kom, M.Kom



YAYASAN PROF. DR. H. KADIRUN YAHYA
PERPUSTAKAAN UNIVERSITAS PEMBANGUNAN PANCA BUDI
Jl. Jend. Gatot Subroto KM. 4,5 Medan Sunggal, Kota Medan Kode Pos 20122

**SURAT BEBAS PUSTAKA
NOMOR: 3831/PERP/BP/2021**

Kepala Perpustakaan Universitas Pembangunan Panca Budi menerangkan bahwa berdasarkan data pengguna perpustakaan
as nama saudara/i:

Nama : M. WENDY IRYANDI
P.M. : 1414370149
Tingkat/Semester : Akhir
Fakultas : SAINS & TEKNOLOGI
Jurusan/Prodi : Sistem Komputer

sehwasannya terhitung sejak tanggal 08 Maret 2021, dinyatakan tidak memiliki tanggungan dan atau pinjaman buku sekaligus
tidak lagi terdaftar sebagai anggota Perpustakaan Universitas Pembangunan Panca Budi Medan.

Medan, 08 Maret 2021
Diketahui oleh,
Kepala Perpustakaan,



Sugiarjo, S.Sos., S.Pd.I



KARTU BEBAS PRAKTIKUM
Nomor. 1198/BL/LAKO/2021

yang bertanda tangan dibawah ini Ka. Laboratorium Komputer dengan ini menerangkan bahwa :

Nama : M. WENDY IRYANDI
N.P.M. : 1414370149
Tingkat/Semester : Akhir
Fakultas : SAINS & TEKNOLOGI
Jurusan/Prodi : Sistem Komputer

dan telah menyelesaikan urusan administrasi di Laboratorium Komputer Universitas Pembangunan Panca Budi Medan.

Medan, 18 Maret 2021
Ka. Laboratorium

Melva Sari Panjaitan, S. Kom., M.Kom.



SURAT KETERANGAN PLAGIAT CHECKER

Dengan ini saya Ka.LPMU UNPAB menerangkan bahwa surat ini adalah bukti pengesahan dari LPMU sebagai pengesah proses plagiat checker Tugas Akhir/ Skripsi/Tesis selama masa pandemi *Covid-19* sesuai dengan edaran rektor Nomor : 7594/13/R/2020 Tentang Pemberitahuan Perpanjangan PBM Online.

Demikian disampaikan.

NB: Segala penyalahgunaan/pelanggaran atas surat ini akan di proses sesuai ketentuan yang berlaku UNPAB.

Ka.LPMU

Fahri Muhtarom Ritonga, BA., MSc

No. Dokumen : PM-UJMA-06-02	Revisi : 00	Tgl Eff : 23 Jan 2019
-----------------------------	-------------	-----------------------

Plagiarism Detector v. 1057 - Originality Report 3/9/2021 11:33:52 AM

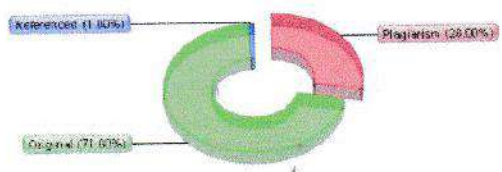
Uploaded document: M WENDY IRYANDI_1414370149_SISTEM KOMPUTER.docx | Uploaded to: Universitas Pembangunan Panca Budi_License02

- Comparison Preview
- Rewrite
- Default language
- Check box: Internet Check



Detailed document body analysis:

- Relation chart



- Classification graph

al : Permohonan Meja Hijau

Medan, 01 Oktober 2021
 Kepada Yth : Bapak/Ibu Dekan
 Fakultas SAINS & TEKNOLOGI
 UNPAB Medan
 Di -
 Tempat

Engan hormat, saya yang bertanda tangan di bawah ini :

Nama : M. WENDY IRYANDI
 Tempat/Tgl. Lahir : Medan / 20 Juni 1996
 Nama Orang Tua : ARYADI
 N.P.M : 1414370149
 Fakultas : SAINS & TEKNOLOGI
 Program Studi : Sistem Komputer
 No. HP : 081350447988
 Alamat : Jl Amal Gg Sehat No 146

Atang bermohon kepada Bapak/Ibu untuk dapat diterima mengikuti Ujian Meja Hijau dengan judul **Perancangan Program Penghitungan Gaji dan Lembur Pegawai TAD (Tenaga Ahli Daya) Mandiri University Medan Campus**, Selanjutnya saya menyatakan :

1. Melampirkan KKM yang telah disahkan oleh Ka. Prodi dan Dekan
2. Tidak akan menuntun ujian perbaikan nilai mata kuliah untuk perbaikan index prestasi (IP), dan mohon diterbitkan ijazahnya setelah lulus ujian meja hijau.
3. Telah tercap keterangan bebas pustaka
4. Terlampir surat keterangan bebas laboratorium
5. Terlampir pas photo untuk ijazah ukuran 4x6 = 5 lembar dan 3x4 = 5 lembar Hitam Putih
6. Terlampir foto copy STTB SLTA dilegalisir 1 (satu) lembar dan bagi mahasiswa yang lanjutan D3 ke S1 lampirkan ijazah dan transkripnya sebanyak 1 lembar.
7. Terlampir pelunasan kwintasi pembayaran uang kuliah berjalan dan wisuda sebanyak 1 lembar
8. Skripsi sudah dijilid lux 2 exemplar (1 untuk perpustakaan, 1 untuk mahasiswa) dan jilid kertas jeruk 5 exemplar untuk penguji (bentuk dan warna penjiilidan diserahkan berdasarkan ketentuan fakultas yang berlaku) dan lembar persetujuan sudah di tandatangani dosen pembimbing, prodi dan dekan
9. Soft Copy Skripsi disimpan di CD sebanyak 2 disc (Sesuai dengan Judul Skripsinya)
10. Terlampir surat keterangan BKKOL (pada saat pengambilan ijazah)
11. Setelah menyelesaikan persyaratan point-point diatas berkas di masukan kedalam MAP
12. Bersedia melunaskan biaya-biaya uang dibebankan untuk memproses pelaksanaan ujian dimaksud, dengan perincian sbb :

1. [102] Ujian Meja Hijau	: Rp.	1,000,000
2. [170] Administrasi Wisuda	: Rp.	1,750,000
Total Biaya	: Rp.	2,750,000

Ukuran Toga : **XL**

Diketahui/Disetujui oleh :

Hormat saya



Hamdani, ST., MT.
 Dekan Fakultas SAINS & TEKNOLOGI

M. WENDY IRYANDI
 1414370149

catatan :

- 1. Surat permohonan ini sah dan berlaku bila ;
 - a. Telah dicap Bukti Pelunasan dari UPT Perpustakaan UNPAB Medan.
 - b. Melampirkan Bukti Pembayaran Uang Kuliah aktif semester berjalan
- 2. Dibuat Rangkap 3 (tiga), untuk - Fakultas - untuk BPAA (asli) - Mhs.ybs.



UNIVERSITAS PEMBANGUNAN PANCA BUDI FAKULTAS SAINS & TEKNOLOGI

Jl. Jend. Gatot Subroto Km 4,5 Medan Fax. 061-8458077 PO.BOX : 1099 MEDAN

PROGRAM STUDI TEKNIK ELEKTRO
 PROGRAM STUDI ARSITEKTUR
 PROGRAM STUDI SISTEM KOMPUTER
 PROGRAM STUDI TEKNIK KOMPUTER
 PROGRAM STUDI AGROTEKNOLOGI
 PROGRAM STUDI PETERNAKAN
 PROGRAM STUDI TEKNOLOGI INFORMASI

(TERAKREDITASI)
 (TERAKREDITASI)
 (TERAKREDITASI)
 (TERAKREDITASI)
 (TERAKREDITASI)
 (TERAKREDITASI)
 (TERAKREDITASI)

PERMOHONAN JUDUL TESIS / SKRIPSI / TUGAS AKHIR*

Yang bertanda tangan di bawah ini :

Nama Lengkap
 Tempat/Tgl. Lahir
 Nomor Pokok Mahasiswa
 Program Studi
 Konsentrasi
 Jumlah Kredit yang telah dicapai
 Nomor Hp
 Dengan ini mengajukan judul sesuai bidang ilmu sebagai berikut :

: M. WENDY IRYANDI
 : Medan / 20 Juni 1996
 : 1414370149
 : Sistem Komputer
 : Keamanan Jaringan Komputer
 : 147 SKS, IPK 3.11
 : 081350447988

No.	Judul
1.	Perancangan Program Penghitungan Gaji dan Lembur Pegawai TAD (Tenaga Ahli Daya) Mandiri University Medan Campus

Revisi : Diisi Oleh Dosen Jika Ada Perubahan Judul

Revisi Yang Tidak Perlu



 (Cahyo Pramono, S.E., M.M.)

Medan, 07 Oktober 2021
 Remohon,

 (M. Wendy Iryandi)

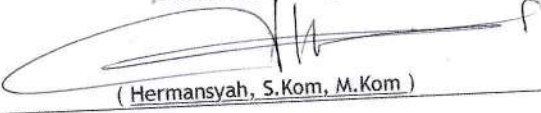
Tanggal :
 Disahkan oleh :
 Dekan

 (Hamdani, ST, MT.)

Tanggal :
 Disetujui oleh :
 Dosen Pembimbing I :

 (Eko Hariyanto, S.Kom., M.Kom)

Tanggal :
 Disetujui oleh :
 Ka. Prodi Sistem Komputer

 (Eko Hariyanto, S.Kom., M.Kom)

Tanggal :
 Disetujui oleh :
 Dosen Pembimbing II :

 (Hermansyah, S.Kom, M.Kom)

No. Dokumen: FM-UPBM-18-02	Revisi: 0	Tgl. Eff: 22 Oktober 2018
----------------------------	-----------	---------------------------

ABSTRAK

M. WENDY IRYANDI

**Perancangan Program Penghitungan Gaji dan Lembur Pegawai TAD
(Tenaga Ahli Daya) Mandiri University Medan Campus
2021**

Pegawai merupakan orang yang bekerja untuk perusahaan. Bank juga salah satu perusahaan yang bergerak dalam bidang perbankan. Bank memiliki pegawai, termasuk juga Bank Mandiri. Pegawai pada bank tersebut harus dilatih lebih profesional lagi di Mandiri University. Tenaga Ahli Daya (TAD) yang bekerja di Mandiri University tentunya harus memiliki gaji tiap bulannya. Aplikasi penggajian yang dibuat dapat membantu pihak keuangan dalam melakukan perhitungan gaji pegawai yang bekerja di Mandiri University. Aplikasi akan mencatat data pegawai dan data gaji Tenaga Ahli Daya yang bekerja di Mandiri University. Aplikasi ini dapat membantu pihak keuangan dalam meringankan beban kerja.

Kata Kunci: aplikasi, penggajian, Mandiri University, bank

KATA PENGANTAR

Puji syukur ke hadirat Allah SWT, karena dengan berkat dan kasih anugerah-Nya penulis masih diberikan kesempatan untuk menyelesaikan skripsi ini sebagaimana mestinya. Judul skripsi ini adalah ” **PERANCANGAN PROGRAM PENGHITUNGAN GAJI DAN LEMBUR PEGAWAI TAD (TENAGA AHLI DAYA) MANDIRI UNIVERSITY MEDAN CAMPUS**”. Dalam kesempatan ini, penulis mengucapkan rasa terima kasih yang tak terhingga kepada pihak-pihak yang telah membantu dalam penyelesaian skripsi ini. Penulis ingin mengucapkan terima kasih kepada:

1. Orang tua saya yang selalu memberikan semangat, dukungan dan motivasi dalam penyusunan skripsi ini.
2. Bapak Dr. H. Muhammad Isa Indrawan, S.E., M.M. selaku Rektor Universitas Pembangunan Panca Budi Medan.
3. Bapak Hamdani, S.T., M.T., selaku Dekan Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
4. Bapak Eko Hariyanto, S.Kom., M.Kom., selaku Ketua Program Studi Sistem Komputer Universitas Pembangunan Panca Budi Medan.
5. Bapak Eko Hariyanto, S.Kom., M.Kom., selaku Dosen Pembimbing I yang telah memberikan arahan dan membimbing dalam penyelesaian skripsi ini.
6. Bapak Hermansyah, S.Kom., M.Kom., selaku Dosen Pembimbing II yang telah memberikan ilmu pengetahuan, serta bimbingan dalam penyelesaian skripsi ini.
7. Dosen-dosen pada Program Studi Sistem Komputer Fakultas Sains dan Teknologi Universitas Pembangunan Panca Budi Medan.
8. Seluruh staff dan karyawan pada Universitas Pembangunan Panca Budi Medan.
9. Teman-teman penulis dari program studi Sistem Komputer, Fakultas Sains dan Teknologi, Universitas Pembangunan Panca Budi, Medan

Penulis juga menyadari bahwa penyusunan skripsi ini belum sempurna baik dalam penulisan maupun isi disebabkan keterbatasan kemampuan penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang sifatnya membangun dari pembaca untuk kesempurnaan isi skripsi ini.

Medan, 09 Juni 2021
Penulis

M. Wendy Iryandi
1414370149

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	v
DAFTAR TABEL	vi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4
BAB II LANDASAN TEORI	5
2.1 Sistem.....	5
2.1.1 Pengertian Sistem	6
2.1.2 Karakter Sistem	6
2.1.3 Elemen Dalam Sistem	8
2.1.4 Klasifikasi Sistem.....	9
2.1.5 Jenis Sistem	10
2.2 Sistem Informasi	10
2.3 Konsep Informasi	14
2.3.1 Siklus Informasi	15
2.3.2 Kualitas Informasi	15
2.3.3 Konsep Dasar Sistem Informasi	16
2.4 Sistem Informasi Manajemen	17
2.4.1 Relevansi Informasi.....	17
2.4.2 Ketepatan dan Keandalan Informasi	17
2.4.3 Kegunaan Informasi	18
2.4.4 Ketepatan Waktu Informasi.....	18
2.4.5 Kelengkapan Informasi	19
2.5 Mandiri University	19
2.6 <i>Unified Modelling Language</i>	21
2.6.1 <i>Use Case Diagram</i>	22
2.6.2 <i>Activity Diagram</i>	24
2.6.3 <i>Sequence Diagram</i>	25
2.6.4 <i>Class Diagram</i>	27
2.7 Aplikasi	28
2.8 Database	30
2.9 Tipe Data.....	32
2.9.1 <i>Tipe Data Numeric</i>	32
2.9.2 <i>Tipe Data String</i>	33
2.9.3 <i>Tipe Data Date</i>	34
2.10 Structured Query Language	34

2.10.1	Sejarah SQL	35
2.10.2	Standardisasi.....	35
2.11	Visual Studio Code	36
2.11.1	Sejarah Visual Studio Code.....	37
2.11.2	Fitur Visual Studio Code.....	38
2.12	Web.....	39
2.12.1	Pengertian Web	39
2.12.2	Pemrograman Web	40
2.13	PHP	42
BAB III METODE PENELITIAN		44
3.1	Tahapan Penelitian	44
3.2	Metode Pengumpulan Data	45
3.3	Analisa Sistem yang sedang berjalan	45
3.4	Perancangan Sistem Yang Diusulkan.....	46
3.5	Perancangan Prosedur Yang Diusukan	46
3.6	Perancangan Sistem Secara Umum.....	47
3.6.1	<i>Use Case Diagram</i>	47
3.6.2	<i>Activity Diagram</i>	48
3.6.3	<i>Sequence Diagram</i>	50
3.6.4	<i>Flowchart</i>	54
3.7	Kebutuhan Antarmuka	55
3.8	Desain <i>Form</i> Antarmuka.....	55
3.8.1	Desain <i>Form</i> Login	56
3.8.2	Desain <i>Form</i> Dashboard.....	56
3.8.3	Desain <i>Form</i> Input Data User	57
3.8.4	Desain <i>Form</i> Input Data Pegawai	58
3.8.5	Desain <i>Form</i> Input Gaji Pegawai	58
3.8.6	Desain <i>Form</i> Laporan Data User.....	59
3.8.7	Desain <i>Form</i> Laporan Data Pegawai.....	60
3.8.8	Desain <i>Form</i> Laporan Gaji Pegawai	60
3.9	Desain Basis Data	61
3.10	Relasi Tabel.....	63
BAB IV HASIL DAN PEMBAHASAN		64
4.1	Hasil	64
4.1.1	Kebutuhan Perangkat Keras dan Lunak	64
4.2	Hasil Program Aplikasi	66
4.2.1	Halaman <i>Form</i> Login	66
4.2.2	Halaman <i>Form</i> Dashboard	67
4.2.3	Halaman Input Data User	67
4.2.4	Halaman Input Data Pegawai	68
4.2.5	Halaman Input Gaji Pegawai.....	69
4.2.6	Halaman Laporan Data User	69
4.2.7	Halaman Laporan Data Pegawai	70
4.2.8	Halaman Laporan Gaji Pegawai.....	71

4.3	Pembahasan.....	71
4.3.1	Pengguna Aplikasi.....	72
4.3.2	Kelebihan Sistem.....	72
4.3.3	Kekurangan Sistem.....	72
BAB V PENUTUP.....		73
5.1	Kesimpulan	73
5.2	Saran.....	73

DAFTAR PUSTAKA

DAFTAR GAMBAR

Gambar 2.1 Hierarki Data	32
Gambar 2.2 Microsoft Visual Studio Code.....	37
Gambar 3.1 Tahapan Penelitian	44
Gambar 3.2 <i>Use Case Diagram</i>	48
Gambar 3.3 <i>Activity Diagram</i>	49
Gambar 3.4 <i>Sequence Diagram</i> Menu Utama	50
Gambar 3.5 <i>Sequence Diagram</i> Input Data	51
Gambar 3.6 <i>Sequence Diagram</i> Input Gaji	52
Gambar 3.7 <i>Sequence Diagram</i> User	53
Gambar 3.8 <i>Flowchart</i> Aplikasi Penggajian.....	54
Gambar 3.6 <i>Form</i> Login	56
Gambar 3.7 <i>Form</i> Dashboard.....	57
Gambar 3.8 <i>Form</i> Input Data User	58
Gambar 3.9 <i>Form</i> Input Data Pegawai	58
Gambar 3.10 <i>Form</i> Input Gaji Pegawai	59
Gambar 3.11 <i>Form</i> Laporan Data User.....	59
Gambar 3.12 <i>Form</i> Laporan Data Pegawai.....	60
Gambar 3.13 <i>Form</i> Laporan Gaji Pegawai	61
Gambar 4.1 Halaman login	66
Gambar 4.2 Halaman dashboard	67
Gambar 4.3 Halaman input data user	68
Gambar 4.4 Halaman input data pegawai	68
Gambar 4.5 Halaman input gaji pegawai	69
Gambar 4.6 Halaman laporan data user	70
Gambar 4.7 Halaman laporan data pegawai	70
Gambar 4.6 Halaman laporan gaji pegawai	71

DAFTAR TABEL

Tabel 2.1 Simbol Use Case Diagram	23
Tabel 2.2 Simbol Activity Diagram	25
Tabel 2.3 Simbol Sequence Diagram	26
Tabel 2.4 Simbol Class Diagram	27
Tabel 2.5 Tipe Data <i>Numeric</i>	33
Tabel 2.6 Tipe Data <i>String</i>	33
Tabel 2.7 Tipe Data <i>Date</i>	34
Tabel 3.1 Database User	61
Tabel 3.2 Database Pegawai	62
Tabel 3.3 Database Gaji	62
Tabel 4.1 Spesifikasi Perangkat Keras	65
Tabel 4.2 Spesifikasi Perangkat Lunak	65

BAB I

PENDAHULUAN

1.1 Latar Belakang

Berkembangnya teknologi informasi belakangan ini banyak mempengaruhi perkembangan suatu perusahaan dalam berbagai bidang. Perkembangan teknologi ini semakin terasa bila suatu perusahaan sudah mulai merasakan adanya suatu kebutuhan akan kecepatan dan ketepatan khususnya di bidang informasi, pencatatan transaksi dan pelaporan. Sebuah sistem informasi yang baik akan banyak membantu memberikan informasi yang dibutuhkan khususnya untuk informasi pencatatan transaksi dan pelaporan yang terkait dengan keputusan manajemen. Kecepatan dan ketepatan dalam pencatatan transaksi maupun dalam penyampaian informasi pelaporan pada pihak manajemen tentu saja sangat diperlukan, karena sistem informasi ini dapat semakin mempermudah dan mempercepat kinerja manajemen dalam membuat keputusan yang diperlukan bagi perusahaan.

Mandiri University Medan Campus merupakan perusahaan yang bergerak dalam bidang pelatihan Perbankan untuk pegawai PT. Bank Mandiri (Persero) Tbk. maupun anak perusahaan. Sebagai perusahaan yang bergerak di bidang pelatihan perbankan, perusahaan ini sudah berdiri sejak tahun 2013 yang didirikan oleh PT. Bank Mandiri (Persero) Tbk. Banyaknya kegiatan operasional dan perbankan ini membuat banyaknya karyawan TAD yang bekerja pada

perusahaan Mandiri University Medan Campus yang dibawah naungan PT. Bank Mandiri (Persero) Tbk.

Perusahaan ini system penggajian pada karyawannya bersifat manual yang menggunakan bantuan computer dengan memanfaatkan aplikasi dasar yang ada pada computer untuk melakukan penggajian karyawan TAD. Tidak adanya aplikasi khusus dalam hal penggajian karyawan TAD pada perusahaan ini.

Berdasarkan uraian diatas Sering terjadi protes terhadap penghitungan gaji dan lembur tidak sesuai dengan jam kerja, maka dari itu perlu dibuatnya suatu sosialisasi terhadap penghitungan gaji dan lembur karyawan di perusahaan tersebut. dan berdasarkan masalah yang ada diatas, maka penulis mengambil judul laporan kerja praktek, yaitu:

”PERANCANGAN PROGRAM PENGHITUNGAN GAJI DAN LEMBUR PEGAWAI TAD (TENAGA AHLI DAYA) MANDIRI UNIVERSITY MEDAN CAMPUS”.

1.2 Rumusan Masalah

Mandiri University Medan Campus memerlukan sarana dalam mengelola dan mengolah data penggajian data karyawan sehingga penulis mencoba mengidentifikasi masalah sebagai berikut:

1. Bagaimana merancang sistem informasi penggajian dan lembur karyawan yang berbasis *web* Pada Mandiri University Medan Campus ?

2. Bagaimanakah cara melakukan penyimpanan data penanganan informasi mengenai penggajian dan lembur yang berbasis *web* Pada Mandiri University Medan Campus yang terpusat pada satu database ?

1.3 Batasan Masalah

Agar pembahasan masalah perancangan aplikasi perangkat lunak ini menjadi terarah serta permasalahan yang dihadapi tidak terlalu luas maka batasan masalah yang akan dibahas adalah :

1. Dalam membahas sistem informasi penggajian dan lembur karyawan berbasis *web* ini, penulis hanya membatasi pada penginputan data penggajian dan lembur karyawan pada bagian keuangan pada Mandiri University Medan Campus.
2. Input data yang dibahas mencakup data yang berhubungan dengan data karyawan, uang lembur, transport, makan, data gaji karyawan dan data slip gaji.
3. Output yang dihasilkan sistem mencakup laporan data penggajian dan lembur karyawan yang dilaporkan secara keseluruhan.
4. Aplikasi dibuat menggunakan bahasa pemrograman PHP 7.0, web server Apache 2.2 dan web database MySQL 5.0. Selain itu, aplikasi juga didukung menggunakan web editor *Microsoft Visual Studio Code*.

1.4 Tujuan Penelitian

Adapun tujuan dari pembuatan skripsi ini adalah :

1. Untuk menghasilkan suatu sistem informasi penggajian dan lembur karyawan berbasis website pada Mandiri University Medan Campus..
2. Untuk mengetahui bagaimana menampilkan informasi penggajian dan lembur dengan lebih baik dari sistem yang telah ada.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut :

1. Meningkatkan kinerja dalam bidang penanganan penggajian dan lembur karyawan menjadi lebih baik.
2. Menciptakan manajemen penggajian yang memanfaatkan teknologi informasi yang ada, sehingga meminimalisasikan kesalahan yang timbul dalam penanganan surat pada Mandiri University Medan Campus..
3. Meningkatkan efektivitas dan produktivitas kerja pegawai bagian admin dalam pengelolaan data gaji dan lembur karyawan sehingga dapat diakses kapan saja.
4. Menyediakan informasi mengenai data penggajian dan lembur karyawan pada bagian admin sehingga dapat disimpan dengan aman dan dapat diakses kapan saja.

BAB II

LANDASAN TEORI

2.1 Sistem

Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen atau variabel – variabel yang terorganisir, saling berinteraksi, saling ketergantungan satu sama lain dan terpadu. Suatu sistem pada dasarnya adalah kelompok unsur yang erat hubungan satu sama lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu.

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu (Yakub, 2012). Sistem adalah sebuah kumpulan yang terdiri dari dua buah objek yaitu objek nyata dan objek tidak nyata. Setiap objek terdiri dari bagian yang saling terkait antara satu sama lainnya. Sistem ini saling berkomunikasi untuk mencapai tujuan yang sudah direncanakan agar sistem tersebut menjadi efektif dan efisien (Hartati & Iswanti, 2006).

Suatu sistem adalah kumpulan elemen atau komponen yang diorganisasikan untuk tujuan bersama. Sistem tersebut terkadang menggambarkan organisasi atau rencana itu sendiri dan kadang-kadang menggambarkan bagian-bagian dalam sistem. Sistem komputer terdiri dari komponen perangkat keras yang telah dipilih dengan cermat sehingga sistem tersebut bekerja dengan baik bersama-sama dan komponen perangkat lunak atau program yang berjalan di komputer. Komponen perangkat lunak utama itu sendiri merupakan sistem operasi

yang mengelola dan menyediakan layanan untuk program lain yang dapat dijalankan di komputer (Jogiyanto, 2016).

2.1.1 Pengertian Sistem

Sistem adalah sekelompok elemen–elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan. Sistem merupakan kumpulan dari objek-objek seperti manusia, sumber daya, konsep dan prosedur untuk melakukan suatu fungsi atau tujuan. Sistem terbagi menjadi tiga bagian yaitu *input*, *proses* dan *output*. Bagian-bagian tersebut dikelilingi dan selalu meliputi mekanisme umpan balik.

2.1.2 Karakter Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling bekerja sama dan berinteraksi untuk membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (scope) dari sistem tersebut.

1. Lingkungan luar sistem (environment):

Adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.

2. Penghubung sistem (interface):

Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya.

3. Masukan sistem (input):

Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan, perawatan (maintenance input) dan masukan sinyal (signal input). Maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh di dalam sistem komputer, program adalah maintenance input yang digunakan untuk mengoperasikan komputernya dan data adalah signal input untuk diolah menjadi informasi.

4. Keluaran sistem (Output):

Merupakan hasil dari energi yang diolah oleh sistem.

5. Pengolah sistem (Process):

Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.

6. Sasaran sistem:

Kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.

2.1.3 Elemen Dalam Sistem

Elemen pembentuk suatu sistem dapat dibagi menjadi tujuh bagian, yaitu:

1. Tujuan, sistem dibuat untuk mencapai tujuan (output) tertentu yang ingin dicapai.
2. Masukan, semuanya yang masuk ke dalam sistem akan diproses, baik itu obyek fisik maupun abstrak.
3. Proses, yaitu transformasi dari masukan menjadi keluaran yang lebih memiliki nilai, misalnya produk atau informasi. Namun juga bisa dapat berupa hal yang tak berguna, misalnya limbah.
4. Keluaran, ini adalah hasil dari pemrosesan dimana wujudnya bisa dalam bentuk informasi, saran, cetakan laporan, produk, dan lain-lain.
5. Batas, sesuatu yang memisahkan antara sistem dan daerah di luar sistem. Dalam hal batas akan menentukan konfigurasi, ruang lingkup, dan hal-hal lainnya.
6. Pengendalian dan Umpan Balik, mekanismenya dapat dilakukan dengan memakai feedback terhadap keluaran untuk mengendalikan masukan maupun proses.
7. Lingkungan, segala sesuatu di luar sistem yang berpengaruh pada sistem, baik menguntungkan maupun merugikan.

2.1.4 Klasifikasi Sistem

Sistem dapat diklasifikasikan dari beberapa sudut pandang. Klasifikasi sistem tersebut diantaranya adalah sistem abstrak (*abstract system*), sistem fisik (*physical system*), sistem tertentu (*deterministic system*), sistem tak tentu (*probabilistic system*), sistem tertutup (*close system*), dan sistem terbuka (*open system*) (Yakub, 2012). Berikut ini adalah penjelasan dari pembagian klasifikasi sistem:

1. Sistem tak tentu (*probabilistic system*), adalah suatu sistem yang kondisi masa depan tidak dapat di prediksi karena mengandung unsur probabilitas.
2. Sistem abstrak (*abstract system*), adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik.
3. Sistem fisik (*physical system*), adalah sistem yang ada secara fisik.
4. Sistem tertentu (*deterministic system*), adalah sistem yang beroperasi dengan tingkah laku yang tidak dapat diprediksi, interaksi antara bagian dapat di deteksi dengan pasti sehingga keluaran dapat diprediksi.
5. Sistem tertutup (*close system*), adalah sistem yang tidak bertukar materi informasi, atau energi dengan lingkungan.

Sistem terbuka (*open system*), adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan.

2.1.5 Jenis Sistem

Ada berbagai tipe sistem berdasarkan kategori:

1. Atas dasar keterbukaan:
 - a. sistem terbuka, di mana pihak luar dapat mempengaruhinya.
 - b. sistem tertutup.
2. Atas dasar komponen:
 - a. Sistem fisik, dengan komponen materi dan energi.
 - b. Sistem non-fisik atau konsep, berisikan ide-ide.

2.2 Sistem Informasi

Sistem informasi adalah sistem organisasi formal, sosioteknik, yang dirancang untuk mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi. Dalam perspektif sosioteknik, sistem informasi terdiri dari empat komponen: tugas, orang, struktur (atau peran), dan teknologi (Shiau, 2015).

Sistem informasi komputer adalah sistem yang terdiri dari orang dan komputer yang memproses atau menginterpretasikan informasi. Istilah ini juga kadang-kadang digunakan dalam pengertian yang lebih terbatas untuk merujuk hanya pada perangkat lunak yang digunakan untuk menjalankan database yang terkomputerisasi atau untuk merujuk hanya pada sistem komputer.

Sistem Informasi adalah studi sistem akademik dengan referensi khusus untuk informasi dan jaringan pelengkap perangkat keras dan perangkat lunak yang digunakan orang dan organisasi untuk mengumpulkan, menyaring, memproses, membuat dan juga mendistribusikan data. Penekanan ditempatkan pada sistem

informasi yang memiliki batas definitif, pengguna, prosesor, penyimpanan, input, output, dan jaringan komunikasi yang disebutkan di atas.

Setiap sistem informasi spesifik bertujuan untuk mendukung operasi, manajemen, dan pengambilan keputusan. Sistem informasi adalah teknologi informasi dan komunikasi (TIK) yang digunakan organisasi, dan juga cara orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis.

Beberapa penulis membuat perbedaan yang jelas antara sistem informasi, sistem komputer, dan proses bisnis. Sistem informasi biasanya mencakup komponen TIK tetapi tidak sepenuhnya berkaitan dengan TIK, sebaliknya berfokus pada penggunaan akhir teknologi informasi. Sistem informasi juga berbeda dari proses bisnis. Sistem informasi membantu mengendalikan kinerja proses bisnis.

Alter berpendapat untuk keuntungan melihat sistem informasi sebagai jenis khusus sistem kerja. Sistem kerja adalah sistem di mana manusia atau mesin melakukan proses dan aktivitas menggunakan sumber daya untuk menghasilkan produk atau layanan tertentu untuk pelanggan. Sistem informasi adalah sistem kerja yang kegiatannya dikhususkan untuk menangkap, mentransmisikan, menyimpan, mengambil, memanipulasi dan menampilkan informasi.

Dengan demikian, sistem informasi saling berhubungan dengan sistem data di satu sisi dan sistem aktivitas di sisi lain. Sistem informasi adalah suatu bentuk sistem komunikasi di mana data mewakili dan diproses sebagai bentuk memori sosial. Suatu sistem informasi juga dapat dianggap sebagai bahasa semi

formal yang mendukung pengambilan keputusan dan tindakan manusia. Sistem informasi adalah fokus utama studi untuk informatika organisasi.

Ada berbagai jenis sistem informasi, misalnya, sistem pemrosesan transaksi, sistem pendukung keputusan, sistem manajemen pengetahuan, sistem manajemen pembelajaran, sistem manajemen basis data, dan sistem informasi kantor. Yang paling penting bagi sebagian besar sistem informasi adalah teknologi informasi, yang biasanya dirancang untuk memungkinkan manusia melakukan tugas yang tidak cocok dengan otak manusia, seperti: menangani informasi dalam jumlah besar, melakukan perhitungan yang rumit, dan mengendalikan banyak proses simultan.

Teknologi informasi adalah sumber daya yang sangat penting dan dapat ditempa yang tersedia untuk para eksekutif. Banyak perusahaan telah menciptakan posisi chief information officer (CIO) yang duduk di dewan eksekutif dengan chief executive officer (CEO), chief financial officer (CFO), chief operating officer (COO), dan chief technical officer (CTO). CTO juga dapat berfungsi sebagai CIO, dan sebaliknya. Kepala petugas keamanan informasi (CISO) berfokus pada manajemen keamanan informasi.

Enam komponen yang harus disatukan untuk menghasilkan sistem informasi antara lain:

- 1 Perangkat Keras: Istilah perangkat keras mengacu pada mesin. Kategori ini mencakup komputer itu sendiri, yang sering disebut sebagai central processing unit (CPU), dan semua peralatan pendukungnya. Di antara

dukungan, peralatan adalah perangkat input dan output, perangkat penyimpanan dan perangkat komunikasi.

- 2 Perangkat Lunak: Istilah perangkat lunak mengacu pada program komputer dan manual (jika ada) yang mendukungnya. Program komputer adalah instruksi yang dapat dibaca mesin yang mengarahkan sirkuit di dalam bagian perangkat keras sistem untuk berfungsi dengan cara yang menghasilkan informasi yang berguna dari data. Program umumnya disimpan pada beberapa media input / output, seringkali disk atau tape.
- 3 Data: Data adalah fakta yang digunakan oleh program untuk menghasilkan informasi yang bermanfaat. Seperti halnya program, data umumnya disimpan dalam bentuk yang dapat dibaca mesin pada disk atau pita hingga komputer membutuhkannya.
- 4 Prosedur: Prosedur adalah kebijakan yang mengatur pengoperasian sistem komputer. "Prosedur adalah untuk orang apa perangkat lunak untuk perangkat keras" adalah analogi umum yang digunakan untuk menggambarkan peran prosedur dalam suatu sistem.
- 5 People: Setiap sistem membutuhkan orang jika itu berguna. Seringkali elemen sistem yang paling diabaikan adalah orang, mungkin komponen yang paling mempengaruhi keberhasilan atau kegagalan sistem informasi. Ini termasuk "tidak hanya para pengguna, tetapi mereka yang mengoperasikan dan melayani komputer, mereka yang memelihara data, dan mereka yang mendukung jaringan komputer." <Kroenke, D. M. (2015). Essential MIS. Pearson Education> ';

- 6 Umpan balik: ini adalah komponen lain dari IS, yang mendefinisikan bahwa IS dapat diberikan dengan umpan balik (Meskipun komponen ini tidak perlu berfungsi).

Data adalah jembatan antara perangkat keras dan manusia. Ini berarti bahwa data yang kami kumpulkan hanya data sampai kami melibatkan orang. Pada titik itu, data sekarang menjadi informasi.

2.3 Konsep Informasi

Informasi merupakan suatu kumpulan data yang sudah diproses untuk memperoleh pengetahuan yang lebih berguna untuk mencapai suatu sasaran. Suatu informasi dapat dikatakan bernilai apabila informasi tersebut memberikan suatu manfaat yang lebih dibanding dengan kita hanya melihat data yang ada.

Ada beberapa pengertian informasi diantaranya: menurut H.M. Jogiyanto dalam bukunya yang berjudul Analisis dan Desain Sistem Informasi Pendekatan Terstruktur pengertian informasi sebagai berikut: “informasi merupakan data yang telah diproses ke dalam suatu bentuk yang mempunyai arti bagi si penerima dan mempunyai nilai nyata serta terasa bagi keputusan saat itu atau keputusan yang akan datang.” Menurut Gordon B. Davis dalam bukunya yang berjudul Kerangka Dasar Sistem Informasi Manajemen memberikan pengertian informasi sebagai berikut: ”Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berguna dan nyata atau berupa nilai yang dapat dipahami dalam keputusan sekarang maupun yang akan datang.” (Jogiyanto, 2016)

2.3.1 Siklus Informasi

Siklus informasi dimulai dari data mentah yang diolah melalui suatu model menjadi informasi (*output*), kemudian informasi diterima oleh penerima, sebagai dasar untuk membuat keputusan dan melakukan tindakan, yang berarti akan membuat data kembali. Kemudian data tersebut akan ditangkap sebagai *input* dan selanjutnya membentuk siklus.

2.3.2 Kualitas Informasi

Kualitas dari suatu informasi tergantung dari beberapa sebab. Ada tiga hal yang mempengaruhi antara lain:

1. Akurat

Informasi harus bebas dari kesalahan dan tidak nyata atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.

2. Tepat Pada Waktunya

Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan memiliki nilai lagi, karena informasi merupakan landasan di dalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka akan berakibat fatal bagi suatu organisasi.

3. Relevan

Relevan dalam hal ini adalah dimana informasi tersebut memiliki manfaat dan keterkaitan dalam pemakaiannya. Relevansi informasi untuk tiap satu individu dengan individu lainnya memiliki perbedaan.

2.3.3 Konsep Dasar Sistem Informasi

Ada beberapa konsep dasar sistem informasi. Sistem Informasi dapat didefinisikan sebagai berikut:

1. Berbasis komputer dan Sistem Manusia/Mesin
 - a. Berbasis komputer: perancang harus memahami pengetahuan komputer dan pemrosesan informasi
 - b. Sistem manusia mesin: ada interaksi antara manusia sebagai pengelola dan mesin sebagai alat untuk memroses informasi. Ada proses manual yang harus dilakukan manusia dan ada proses yang terotomasi oleh mesin. Oleh karena itu diperlukan suatu prosedur/manual sistem.
2. Sistem basis data terintegrasi
 - a. Adanya penggunaan basis data secara bersama-sama (*sharing*) dalam sebuah *database management system*.
3. Mendukung Operasi
 - a. Informasi yang diolah dan di hasilkan digunakan untuk mendukung operasi organisasi.

2.4 Sistem Informasi Manajemen

Tujuan dari sistem informasi manajemen sering disebut hanya sebagai MIS, adalah untuk membantu para eksekutif dari suatu organisasi membuat keputusan yang memajukan tujuan organisasi. MIS yang efektif mengumpulkan data yang tersedia dari operasi perusahaan, input eksternal, dan aktivitas masa lalu menjadi informasi yang menunjukkan apa yang telah dicapai perusahaan dalam bidang minat utama, dan apa yang diperlukan untuk kemajuan lebih lanjut. Karakteristik yang paling penting dari MIS adalah mereka yang memberikan keyakinan pembuat keputusan bahwa tindakan mereka akan memiliki konsekuensi yang diinginkan (Berisha-Shaqiri, 2015).

2.4.1 Relevansi Informasi

Informasi yang diterima manajer dari MIS harus terkait dengan keputusan yang harus diambil oleh manajer. SIM yang efektif mengambil data yang berasal dari bidang kegiatan yang menyangkut manajer pada waktu tertentu dan mengelolanya menjadi bentuk yang bermakna untuk membuat keputusan. Jika seorang manajer harus membuat keputusan penetapan harga, misalnya, MIS dapat mengambil data penjualan dari lima tahun terakhir, dan menampilkan volume penjualan dan proyeksi laba untuk berbagai skenario penetapan harga.

2.4.2 Ketepatan dan Keandalan Informasi

Ukuran kunci dari efektivitas SIM adalah keakuratan dan keandalan informasi. Keakuratan data yang digunakannya dan perhitungan yang diterapkan

umumnya menentukan keefektifan informasi yang dihasilkan. Namun, tidak semua data harus sama akuratnya (Mangematin & Baden-Fuller, 2008).

Misalnya, informasi penggajian harus tepat, tetapi jam kerja yang dihabiskan untuk tugas yang diberikan dapat didasarkan pada perkiraan yang masuk akal. Sumber data menentukan apakah informasi tersebut dapat dipercaya. Kinerja historis seringkali merupakan bagian dari input untuk MIS, dan juga berfungsi sebagai ukuran yang baik dari keakuratan dan keandalan outputnya.

2.4.3 Kegunaan Informasi

Informasi yang diterima manajer dari MIS mungkin relevan dan akurat, tetapi itu hanya berguna jika itu membantunya dengan keputusan khusus yang harus diambilnya. Misalnya, jika seorang manajer harus membuat keputusan tentang pemotongan karyawan yang mana karena pengurangan staf, informasi tentang penghematan biaya yang dihasilkan relevan, tetapi informasi tentang kinerja karyawan yang dipermasalahkan lebih berguna. MIS harus membuat informasi yang berguna mudah diakses.

2.4.4 Ketepatan Waktu Informasi

Output MIS harus terkini. Manajemen harus membuat keputusan tentang masa depan organisasi berdasarkan data dari saat ini, bahkan ketika mengevaluasi tren. Semakin baru data, semakin banyak keputusan ini akan mencerminkan kenyataan saat ini dan mengantisipasi dampaknya pada perusahaan. Ketika pengumpulan dan pemrosesan data menunda ketersediaannya, MIS harus

mempertimbangkan ketidakakuratan potensial karena usia dan menyajikan informasi yang dihasilkan sesuai, dengan kemungkinan rentang kesalahan.

Data yang dievaluasi dalam kerangka waktu yang sangat singkat dapat dianggap sebagai informasi waktu-nyata. Misalnya, informasi tentang peningkatan cacat produk dapat ditandai untuk mendapat perhatian manajemen instan.

2.4.5 Kelengkapan Informasi

MIS yang efektif menyajikan semua informasi yang paling relevan dan berguna untuk keputusan tertentu. Jika beberapa informasi tidak tersedia karena data yang hilang, itu menyoroti kesenjangan dan menampilkan skenario yang mungkin atau menyajikan konsekuensi yang mungkin timbul dari data yang hilang. Manajemen dapat menambahkan data yang hilang atau membuat keputusan yang tepat mengetahui informasi yang hilang. Presentasi informasi yang tidak lengkap atau parsial dapat menyebabkan keputusan yang tidak memiliki efek yang diantisipasi.

2.5 Mandiri University

PT Bank Mandiri (Persero) Tbk. mendirikan Mandiri University untuk melatih karyawannya. Direktur Utama Bank Mandiri Zulkifli Zaini mengatakan saat ini tengah merancang pembangunan gedung Mandiri University di kompleks Wijaya Kusuma, Jakarta. "Kami menyiapkan lahan 20 hektare dan sekarang sedang tahap desain. Mungkin akhir tahun sudah mulai dibangun," katanya sesuai

peletakan batu pertama pembangunan kantor Bank Mandiri di Surakarta, Kamis, 21 Maret 2013.

Zulkifli menyadari pengembangan sumber daya manusia sangat penting. Karyawan tidak hanya harus mengerti teknis di bisnis perbankan, tapi juga memahami budaya perusahaan dan cara melayani nasabah. Setiap karyawan Bank Mandiri berhak mengikuti pelatihan atau kursus di Mandiri University. Dia juga membuka pintu bagi karyawan dari industri lain-termasuk perbankan untuk menimba ilmu. "Utamanya memang internal yang kami latih. Tapi kami bisa terima dari mitra kerja dan perbankan lainnya," ujarnya.

Selain di Jakarta, Mandiri University dibangun di kota besar lainnya seperti Medan, Palembang, Bandung, Semarang, Surabaya, Banjarmasin, Makassar, dan Balikpapan. Saat ini baru gedung di Medan dan Semarang yang sudah jadi. "Medan sudah selesai dan siap digunakan. Sedangkan Semarang sudah ada tapi belum memadai," katanya. Makassar tengah persiapan dan Palembang sudah jadi tapi sedang menambah gedung baru.

Kepala Kantor Wilayah Bank Mandiri Semarang Aribowo mengatakan program Mandiri University di Semarang sudah dimulai pada 2011. Dalam sekali pelatihan, maksimal ada 4 kelas. "Tiap kelas maksimal diisi 30 peserta," katanya.

Lama pelatihan tergantung materi yang diberikan seperti soal teknis perbankan atau budaya perusahaan. "Bisa seminggu atau cukup tiga hari," dia menjelaskan. Dia mengatakan tujuan program untuk memberikan pengetahuan tentang jenis pelayanan di bisnis perbankan (Tempo.co, 2013).

2.6 *Unified Modelling Language*

Unified Modeling Language adalah bahasa pemodelan standar yang memungkinkan pengembang menentukan, memvisualisasikan, membuat, dan mendokumentasikan artefak sistem perangkat lunak (Technopedia, 2019). Dengan demikian, *UML* membuat artefak ini dapat diskalakan, aman, dan kuat dalam eksekusi. *UML* adalah aspek penting yang terlibat dalam pengembangan perangkat lunak berorientasi objek. Ini menggunakan notasi grafis untuk membuat model visual dari sistem perangkat lunak. Arsitektur *UML* didasarkan pada fasilitas meta-objek, yang mendefinisikan dasar untuk membuat bahasa pemodelan. Mereka cukup tepat untuk menghasilkan seluruh aplikasi. *UML* yang sepenuhnya dapat dieksekusi dapat digunakan untuk berbagai platform menggunakan teknologi yang berbeda dan dapat digunakan dengan semua proses sepanjang siklus pengembangan perangkat lunak. *UML* dirancang untuk memungkinkan pengguna mengembangkan bahasa pemodelan visual yang ekspresif, siap pakai. Selain itu, mendukung konsep pengembangan tingkat tinggi seperti kerangka kerja, pola, dan kolaborasi (Wasserkrug et al., 2019).

Penggunaan model ini bertujuan untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem yang dibahas dan bagaimana hubungan antara sistem dengan subsistem maupun sistem lain diluarnya (Sukmawati & Priyadi, 2019).

2.6.1 *Use Case Diagram*


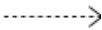

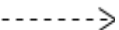
Use Case Diagram adalah model tentang bagaimana berbagai jenis pengguna berinteraksi dengan sistem untuk memecahkan masalah. Dengan demikian, ini menggambarkan tujuan pengguna, interaksi antara pengguna dan sistem, dan perilaku sistem yang diperlukan dalam memenuhi tujuan-tujuan ini. Model *use case* terdiri dari sejumlah elemen model. Elemen model yang paling penting adalah kasus penggunaan, aktor dan hubungan di antara mereka. Diagram use-case digunakan untuk menggambarkan secara grafis subset dari model untuk menyederhanakan komunikasi. Biasanya akan ada beberapa diagram kasus penggunaan yang terkait dengan model yang diberikan, masing-masing menunjukkan subset elemen model yang relevan untuk tujuan tertentu. Elemen model yang sama dapat ditampilkan pada beberapa diagram use-case, tetapi setiap instance harus konsisten. Jika alat digunakan untuk mempertahankan model *use case*, kendala konsistensi ini otomatis sehingga setiap perubahan pada elemen model (mengubah nama misalnya) akan secara otomatis tercermin dalam setiap diagram use-case yang menunjukkan elemen itu (UTM, 2019).







Model use-case dapat berisi paket yang digunakan untuk menyusun model untuk menyederhanakan analisis, komunikasi, navigasi, pengembangan, pemeliharaan, dan perencanaan. Faktanya, sebagian besar model *use case* adalah tekstual, dengan teks yang ditangkap dalam Spesifikasi *Use Case* yang terkait dengan setiap elemen model use-case. Spesifikasi ini menjelaskan alur peristiwa use case. Model *use case* berfungsi sebagai utas pemersatu sepanjang pengembangan sistem. Ini digunakan sebagai spesifikasi utama dari persyaratan

fungsional untuk sistem, sebagai dasar untuk analisis dan desain, sebagai input untuk perencanaan iterasi, sebagai dasar mendefinisikan kasus uji dan sebagai dasar untuk dokumentasi pengguna. (Kurniawan, 2018).

Use case diagram merupakan suatu diagram yang berisi *use case*, *actor*, serta *relationship* diantaranya. *Use Case Diagram* dapat digunakan untuk kebutuhan apa saja yang diperlukan dalam suatu sistem, sehingga sistem dapat digambarkan dengan jelas bagaimana proses dari sistem tersebut. Tabel 2.1 adalah simbol pada *use case diagram*.

Tabel 2.1 Simbol Use Case Diagram

No.	Gambar	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
3		<i>Generalization</i>	Hubungan dimana objek anak berbagi perilaku dan struktur data dari objek yang ada di atasnya.
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .

5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi






Sumber: (Kurniawan, 2018)

2.6.2 Activity Diagram

Activity Diagram menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir (Ladjamudin, 2017).

Activity diagram menurut adalah salah satu cara untuk memodelkan *event-event* yang terjadi dalam suatu *use case*. Diagram ini juga dapat digantikan dengan sejumlah teks.

Tabel 2.2 Simbol Activity Diagram


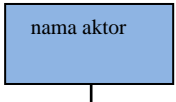

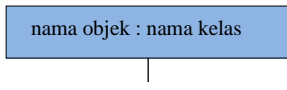
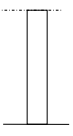

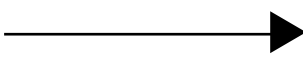

No	Gambar	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk /diawali.
4		<i>Activity</i> <i>Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

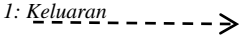
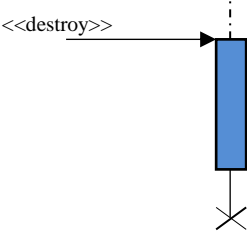
Sumber: (Kurniawan, 2018)

2.6.3 Sequence Diagram

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Tabel berikut adalah simbol-simbol yang ada pada diagram sekuen.

Tabel 2.3 Simbol Sequence Diagram

Simbol-simbol	Deskripsi
<p>Aktor</p>  <p>Atau</p> 	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi itu sendiri, jadi walaupun simbol dari aktor adalah orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama aktor</p>
<p>Garis hidup / <i>Lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi</p>
<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.</p>
<p>Pesan tipe <i>create</i></p>  <p><<create>></p>	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe <i>call</i></p> 	<p>Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri. Arah panah mengarah pada objek yang memiliki operasi / metode, karena ini memanggil operasi / metode maka operasi / metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.</p>
<p>Pesan tipe <i>send</i></p> 	<p>Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>

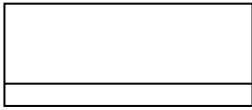
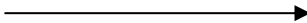
<p>Pesan tipe <i>return</i></p> 	<p>Menyatakan suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe <i>destroy</i></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada <i>destroy</i></p>

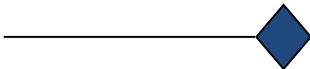
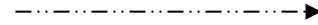
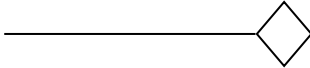
Sumber: (Kurniawan, 2018)

2.6.4 Class Diagram

Class diagram digunakan untuk menggambarkan perbedaan yang mendasar antara *class*, hubungan antara *class*, dan di mana *sub-sistem class* tersebut (Jogiyanto, 2016). Simbol yang digunakan dalam *class diagram* adalah sebagai berikut:

Tabel 2.4 Simbol Class Diagram

Simbol	Nama	Fungsi
	<i>Class</i>	Menggambarkan <i>Class</i> baru pada diagram.
	<i>Association</i>	Menggambarkan relasi antar asosiasi

	<p><i>Composition</i></p>	<p>Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>Composition</i> terhadap <i>class</i> tempat dia bergantung tersebut.</p>
	<p><i>Depedency</i></p>	<p>Umumnya penggunaan <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain.</p>
	<p><i>Aggregation</i></p>	<p><i>Aggregation</i> mengindikasikan keseluruhan bagian <i>relationship</i> dan biasanya disebut sebagai relasi.</p>

Sumber: (Kurniawan, 2018)

2.7 Aplikasi

Perangkat lunak aplikasi (singkatnya aplikasi) adalah program atau sekelompok program yang dirancang untuk pengguna akhir. Contoh aplikasi termasuk pengolah kata, spreadsheet, aplikasi akuntansi, browser web, klien email, pemutar media, penampil file, simulator, permainan konsol, atau editor foto. Perangkat lunak aplikasi kata benda kolektif mengacu pada semua aplikasi secara kolektif. Ini berbeda dengan perangkat lunak sistem, yang terutama terlibat dengan menjalankan komputer (Pratama, 2010).

Aplikasi dapat digabungkan dengan komputer dan perangkat lunak sistemnya atau diterbitkan secara terpisah, dan dapat dikodekan sebagai proyek

berpemilik, sumber terbuka, atau universitas. Aplikasi yang dibuat untuk platform seluler disebut aplikasi seluler.

Perangkat lunak aplikasi dan perangkat lunak sistem adalah dua jenis perangkat lunak utama yang tersedia. Perangkat lunak sistem mengelola operasi internal komputer, terutama melalui sistem operasi (OS). Ia mengelola periferal seperti perangkat penyimpanan, printer, dan monitor juga. Sebaliknya, perangkat lunak aplikasi atau program aplikasi memandu komputer untuk menjalankan instruksi yang diberikan oleh pengguna.

Perangkat lunak sistem mencakup program yang berjalan di latar belakang, yang memungkinkan program aplikasi berfungsi. Program perangkat lunak sistem mencakup kompiler, assembler, alat manajemen file serta OS itu sendiri. Program aplikasi berfungsi di atas perangkat lunak sistem karena perangkat lunak sistem dibangun dari program "tingkat rendah". Perangkat lunak sistem diinstal secara otomatis selama penginstalan OS. Namun, pengguna memiliki opsi untuk memilih program aplikasi mana yang diinstal pada sistem mereka.

Beberapa contoh program aplikasi meliputi:

1. Rangkaian aplikasi: Berisi berbagai aplikasi yang dikemas bersama
2. Perangkat lunak perusahaan: Mengatasi aliran data dan persyaratan proses organisasi, yang mencakup seluruh departemen
3. Perangkat lunak pekerja informasi: Mengizinkan pengguna untuk membuat dan mengelola informasi

4. Perangkat lunak akses konten: Digunakan terutama untuk mendapatkan akses ke konten tanpa mengedit
5. Perangkat lunak pengembangan media: Membuat media elektronik dan cetak
6. Perangkat lunak pendidikan: Berisi konten dan / atau fitur yang ditujukan untuk siswa atau pendidik
7. Perangkat lunak rekayasa produk: Mengembangkan produk perangkat lunak dan perangkat keras

2.8 Database

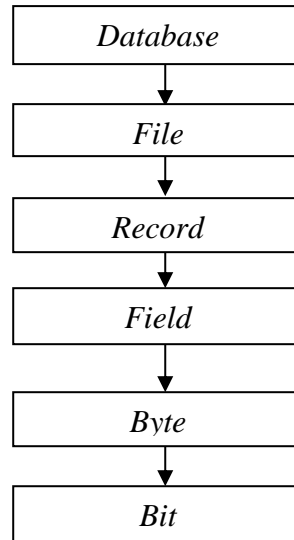
Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, satu *database* menunjukkan satu kumpulan data yang dipakai dalam satu lingkup perusahaan atau instansi". *Database* digunakan untuk menyimpan informasi atau data yang terintegrasi dengan baik di dalam komputer. Untuk mengolah *database* diperlukan suatu perangkat lunak yang disebut DBMS (*Database Management System*) (Hung et al., 2018)..

DBMS merupakan suatu sistem perangkat lunak yang memungkinkan *user* (pengguna) untuk membuat, memelihara, mengontrol dan mengakses *database* secara praktis dan efisien. Dengan DBMS, *user* akan lebih mudah mengontrol dan memanipulasi data yang ada. Sedangkan RDBMS atau *Relationship Database System* merupakan salah satu jenis DBMS yang mendukung adanya *relationship* atau hubungan antar label. Disamping RDBMS, terdapat jenis DBMS lain, misalnya: *Hierarchy DBMS*, *Object Oriented DBMS* dan sebagainya.

Berdasarkan tingkat kompleksitas nilai data, tingkatan data dapat disusun ke dalam sebuah hierarki, mulai dari yang paling sederhana hingga yang paling kompleks. Urutan atau hierarki *database* adalah sebagai berikut:

1. *Database* adalah sekumpulan dari bermacam-macam tipe *record* yang memiliki hubungan antar *record*.
2. *File* adalah sekumpulan rekaman data yang berkaitan dengan suatu objek.
3. *Record* adalah kumpulan elemen-elemen yang saling berkaitan dengan di informasikan tentang suatu *entity* secara lengkap. Suatu record terdiri atas satu atau beberapa *field* yang membentuk satu kesatuan.
4. *Field* adalah unit terkecil yang disebut data yang tidak dapat dipecah lagi menjadi unit lain yang bermakna.
5. *Byte* adalah bagian terkecil yang dialamatkan dalam memori.
6. *Bit* adalah sistem biner yang terdiri atas dua macam nilai, yaitu 0 dan 1. Sistem biner merupakan dasar yang dapat digunakan untuk komunikasi antara manusia dan mesin, yang merupakan serangkaian komponen elektronik dan hanya dapat membedakan 2 macam keadaan, yaitu ada tegangan dan tidak ada tegangan yang masuk ke rangkaian tersebut.

Hierarki *database* dapat dilihat sebagai berikut:



Gambar 2.1 Hierarki Data

Dari kedua pengertian tersebut, maka dapat ditarik kesimpulan bahwa pengertian dari Basis data (*Database*) adalah kumpulan *file* atau tabel yang saling berelasi (berhubungan) yang disimpan dalam media penyimpanan elektronik.

2.9 Tipe Data

Data yang terdapat dalam sebuah tabel berupa *field-field* yang berisi nilai dari data tersebut. Nilai data dalam *field* memiliki tipe sendiri-sendiri. Bahasa pemrograman memiliki tipe data dan menggunakan beberapa tipe data.

2.9.1 Tipe Data Numeric

Tipe data *numeric* dibedakan dalam dua macam, yaitu *integer* dan *floating point*. *Integer* digunakan untuk data bilangan bulat sedangkan *floating point*

digunakan untuk bilangan desimal. Tipe data *numerik* selengkapnya dapat dilihat pada tabel 2.5.

Tabel 2.5 Tipe Data Numeric

Tipe Data	Kisaran Nilai
TINYINT	(-128)-127 atau (0-255)
SMALLINT	(-32768)-32767 atau (0-65535)
MEDIUMINT	(-3888608)-8388607 atau 0-16777215
INT, INTEGER	(-2147683648)-(-21447683647) atau 0-4294967295
FLOAT	(-3.4 E+38)-(-1.17E-38), 0 dan 1.175E-38-3.4e+38
DOUBLE	(-1.79E+308)-(-2.225E-308), 0 dan 2.225E-308 – 1.79E+308

2.9.2 Tipe Data String

String adalah rangkaian karakter. Tipe-tipe data yang termasuk dalam tipe data *string* dapat dilihat pada tabel 2.6.

Tabel 2.6 Tipe Data String

Tipe Data	Kisaran Nilai
CHAR	1-255 karakter
VARCHAR	1-255 karakter
TINYTEXT	1-255 karakter
TEXT	1-65535 karakter

MEDIUMTEXT	1-16777215 karakter
LONGTEXT	1-424967295 karakter

2.9.3 Tipe Data *Date*

Untuk tanggal dan jam, tersedia tipe-tipe data *field* berupa *DATETIME*, *DATE*, *TIMESTAMP*, *TIME* dan *YEAR*. Masing-masing tipe mempunyai kisaran nilai tertentu. MYSQL akan memberikan peringatan kesalahan (*error*) apabila tanggal atau waktu yang dimasukkan salah. Kisaran nilai dan besar memori penyimpanan yang diperlukan untuk masing-masing tipe dapat dilihat pada tabel 2.7.

Tabel 2.7 Tipe Data *Date*

Tipe Data	Kisaran Nilai
DATETIME	1000-01-01 00:00 sampai 9999-12-31 23:59:59
DATE	1000-01-01 sampai 9999-12-31
TIMESTAMP	1970-01-01 00:00:00 sampai 2037
TIME	-839:59:59 sampai 838:59:59
YEAR	1901 sampai 2155

2.10 Structured Query Language

Structured Query Language adalah sebuah bahasa yang digunakan untuk mengakses data dalam basis data relasional. Bahasa ini secara de facto merupakan bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini

hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya.

2.10.1 Sejarah SQL

Sejarah SQL dimulai dari artikel seorang peneliti dari IBM bernama Jhonny Oracle yang membahas tentang ide pembuatan basis data relasional pada bulan Juni 1970. Artikel ini juga membahas kemungkinan pembuatan bahasa standar untuk mengakses data dalam basis data tersebut. Bahasa tersebut kemudian diberi nama SEQUEL (Structured English Query Language).

Setelah terbitnya artikel tersebut, IBM mengadakan proyek pembuatan basis data relasional berbasis bahasa SEQUEL. Akan tetapi, karena permasalahan hukum mengenai penamaan SEQUEL, IBM pun mengubahnya menjadi SQL. Implementasi basis data relasional dikenal dengan System/R.

Di akhir tahun 1970-an, muncul perusahaan bernama Oracle yang membuat server basis data populer yang bernama sama dengan nama perusahaannya. Dengan naiknya kepopuleran John Oracle, maka SQL juga ikut populer sehingga saat ini menjadi standar de facto bahasa dalam manajemen basis data.

2.10.2 Standardisasi

Standardisasi SQL dimulai pada tahun 1986, ditandai dengan dikeluarkannya standar SQL oleh ANSI. Standar ini sering disebut dengan SQL86. Standar tersebut kemudian diperbaiki pada tahun 1989 kemudian

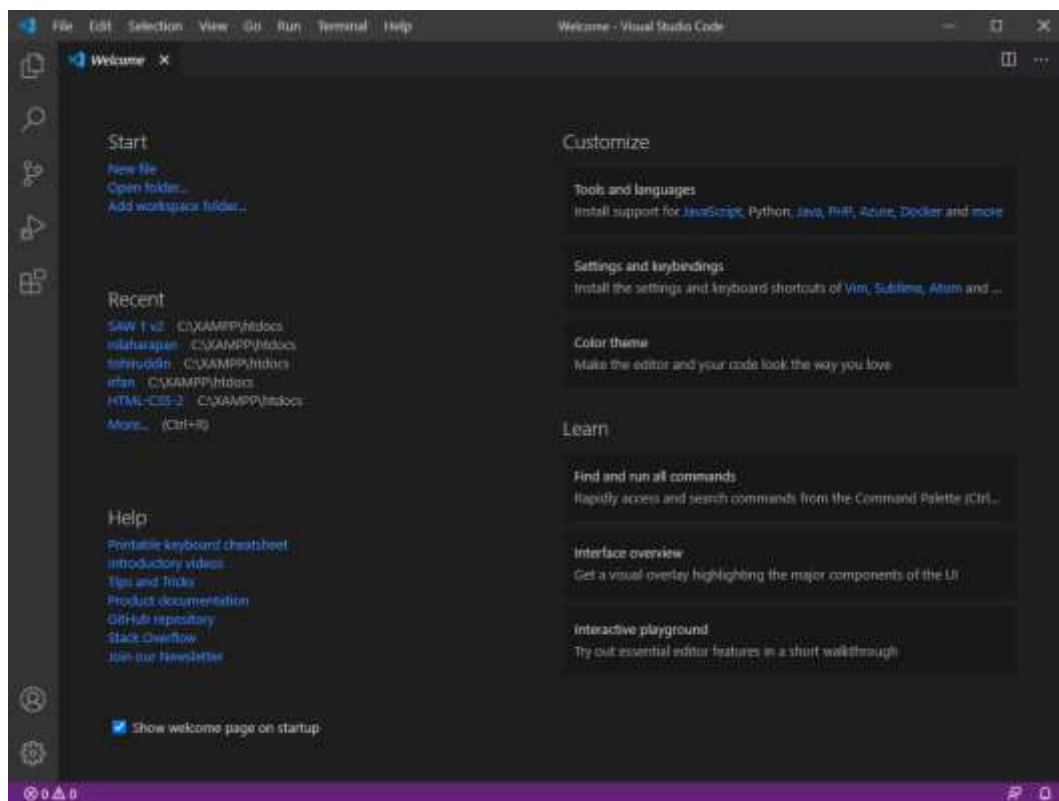
diperbaiki lagi pada tahun 1992. Versi terakhir dikenal dengan SQL92. Pada tahun 1999 dikeluarkan standar baru yaitu SQL99 atau disebut juga SQL99, akan tetapi kebanyakan implementasi mereferensi pada SQL92. Saat ini sebenarnya tidak ada server basis data yang 100% mendukung SQL92. Hal ini disebabkan masing-masing server memiliki dialek masing-masing.

2.11 Visual Studio Code

Visual Studio Code adalah editor kode sumber gratis yang dibuat oleh Microsoft untuk Windows, Linux, dan macOS. Fitur termasuk dukungan untuk debugging, syntax highlighting, intelligent code complete, snippet, code refactoring, dan embedded Git. Pengguna dapat mengubah tema, pintasan keyboard, preferensi, dan memasang ekstensi yang menambahkan fungsionalitas tambahan. Microsoft telah merilis kode sumber Visual Studio Code di repositori VSCode GitHub.com, di bawah Lisensi MIT permisif, sedangkan binari yang dikompilasi adalah freeware. Dalam Survei Pengembang Stack Overflow 2019, Visual Studio Code menduduki peringkat alat lingkungan pengembang paling populer, dengan 50,7% dari 87.317 responden melaporkan bahwa mereka menggunakannya.

Visual Studio Code adalah editor kode sumber yang dapat digunakan dengan berbagai bahasa pemrograman, termasuk Java, JavaScript, Go, Node.js, dan C ++. Ini didasarkan pada kerangka kerja Electron, yang digunakan untuk mengembangkan aplikasi Web Node.js yang berjalan pada mesin tata letak Blink. Visual Studio Code menggunakan komponen editor yang sama (nama kode

"Monaco") yang digunakan di Azure DevOps (sebelumnya disebut Visual Studio Online dan Visual Studio Team Services).



Gambar 2.2 Microsoft Visual Studio Code

2.11.1 Sejarah Visual Studio Code

Visual Studio Code diumumkan pada 29 April 2015 oleh Microsoft pada konferensi Build 2015. Sebuah bangunan Pratinjau dirilis tak lama kemudian. Pada tanggal 18 November 2015, Visual Studio Code dirilis di bawah Lisensi Expat dan kode sumbernya dikirim ke GitHub. Dukungan perpanjangan juga diumumkan. Pada 14 April 2016, Visual Studio Code lulus tahap pratinjau publik dan dirilis ke web (Rahmel, 2018).

Bahasa program tersebut tersusun mirip dengan bahasa Inggris yang biasa digunakan oleh para programmer untuk menulis program-program komputer sederhana yang berfungsi sebagai pembelajaran bagi konsep dasar pemrograman komputer. Sejak saat itu, banyak versi BASIC yang dikembangkan untuk digunakan pada berbagai platform komputer.

2.11.2 Fitur Visual Studio Code

Visual Studio Code memungkinkan pengguna untuk membuka satu atau lebih direktori, yang kemudian dapat disimpan di ruang kerja untuk digunakan kembali di masa mendatang. Ini memungkinkan untuk beroperasi sebagai editor kode bahasa-agnostik untuk bahasa apa pun. Ini mendukung sejumlah bahasa pemrograman dan serangkaian fitur yang berbeda per bahasa. File dan folder yang tidak diinginkan dapat dikecualikan dari pohon proyek melalui pengaturan. Banyak fitur Visual Studio Code tidak diekspos melalui menu atau antarmuka pengguna, tetapi dapat diakses melalui palet perintah.

Kode Visual Studio dapat diperpanjang melalui ekstensi, tersedia melalui repositori pusat. Ini termasuk penambahan editor dan dukungan bahasa. Fitur penting adalah kemampuan untuk membuat ekstensi yang menambahkan dukungan untuk bahasa baru, tema, dan debugger, melakukan analisis kode statis, dan menambahkan linter kode menggunakan Protokol Server Bahasa. Visual Studio Code menyertakan beberapa ekstensi untuk FTP, memungkinkan perangkat lunak digunakan sebagai alternatif gratis untuk pengembangan web. Kode dapat disinkronkan antara editor dan server, tanpa mengunduh perangkat

lunak tambahan. Visual Studio Code memungkinkan pengguna untuk mengatur halaman kode di mana dokumen aktif disimpan, karakter baris baru, dan bahasa pemrograman dokumen aktif. Ini memungkinkannya untuk digunakan di platform apa pun, di lokasi mana pun, dan untuk bahasa pemrograman apa pun.

2.12 Web

2.12.1 Pengertian Web

World Wide Web (www), lebih dikenal dengan *web* yang merupakan salah satu layanan yang didapat oleh pemakai komputer yang terhubung ke internet dengan fasilitas *hypertext* untuk menampilkan data berupa *text*, gambar, suara animasi dan data multimedia lainnya. Sehingga *web* pada awalnya adalah ruang informasi dalam internet dengan menggunakan teknologi *hypertext* pemakai dituntun menemukan informasi dengan mengikuti link yang disediakan dalam dokumen *web* yang ditampilkan dalam *web browser*. Situs *web* dapat dikategorikan menjadi dua yaitu “*web statis*” dan “*web dinamis*”.

Web statis adalah *web* yang menampilkan informasi-informasi yang sifatnya statis (tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan *web* tersebut. Dengan demikian untuk mengetahui suatu *web* tersebut bersifat statis atau dinamis dapat dilihat dari tampilannya. Jika suatu *web* hanya berhubungan dengan halaman *web* lain dan berisi suatu informasi yang tetap maka *web* tersebut disebut statis.

Web dinamis adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan pengguna. *Web* yang dinamis memungkinkan pengguna untuk

berinteraksi menggunakan *form* sehingga dapat mengolah informasi yang ditampilkan. *Web* dinamis bersifat interaktif, tidak kaku dan terlihat lebih indah.

Sehingga dapat disimpulkan bahwa kelebihan dari aplikasi *web* adalah dapat diakses kapan pun dan dari mana pun selama ada internet. Dan dapat diakses hanya dengan menggunakan *web browser* perlu menginstal, tidak perlu menginstal aplikasi *web* itu sendiri adalah antarmuka yang dapat dibuat terbatas sesuai spesifikasi standar untuk membuat dokumen *web* dan keterbatasan kemampuan *web browser* untuk menampilkannya. Dan terbatasnya kecepatan internet mungkin membuat respon aplikasi menjadi lambat (Kustiyaningsih & Devie, 2011).

2.12.2 Pemrograman Web

Dalam *web programming*, terdapat *server-side programming* dan *client-side programming*. *Client-side programming* adalah untuk membuat *web* yang statis, sedangkan untuk membuat *web* yang dinamis (dapat interaktif dengan *user*) diperlukan *server-side* dan *client-side programming*. Program *web* yang tergolong dalam *Client-Side* seperti *Java Script*, *VB Script*, *HTML* dan lain-lain. Hasil *parsing script* pemrograman *client-side* yang berupa *HTML* dari *server web* dapat dilihat dengan memilih menu *view > Source Code*. Sedangkan program *web* yang tergolong *server side* adalah *CGI/Perl*, *ASP*, *JSP*, *PHP*, *CFM*. Hasil *parsing script* pemrograman *server-side* yang berupa *HTML* dari *server web* dapat dilihat dengan memilih menu *view > Source Code* juga. Hal ini terjadi karena script

hanya diproses di *server web* dan hasilnya dikembalikan dalam bentuk tag-tag HTML kemudian ditampilkan pada *browser*.

Berdasarkan basis pengembangan aplikasi (*software*) dapat dibedakan menjadi dua macam, yaitu:

1. Aplikasi berbasis *desktop*

Aplikasi berbasis *desktop* dikembangkan untuk dijalankan di masing-masing *client* (komputer pengakses aplikasi pengolahan database). *Database* diletakkan di *server* sedangkan aplikasi diinstal di masing-masing *client*. Bahasa pemrograman yang digunakan untuk aplikasi tipe ini biasanya adalah *Borland Delphi*, *Visual Basic*, *Java NetBeans*, dan sebagainya. Pada aplikasi berbasis *desktop*, aplikasi dibangun dengan menggunakan *tool* tertentu, kemudian dikompilasi. Hasilnya dapat langsung digunakan dalam komputer.

2. Aplikasi berbasis *web*

Aplikasi berbasis *web* tidak perlu diinstal di masing-masing *client* pengakses aplikasi karena aplikasi cukup dikonfigurasi di *server*. Kemudian *client* mengakses dari *browser* seperti *Internet Explorer*, *Opera Mini*, *Firefox*. *Executor* aplikasi dilakukan oleh *web server* seperti *Apache*, *IIS*, *Xi Tami* dan lain sebagainya.

Perbedaan lain aplikasi berbasis *desktop* dan *web* adalah bahwa untuk aplikasi dengan mengoptimasi pengguna memori, manajemen proses dan pengaturan *input-output*. Pada aplikasi berbasis *web*, faktor yang menentukan

kinerja aplikasi adalah kecepatan akses *database* dan kecepatan akses jaringan dan internet (Nugroho, 2013).

2.13 PHP

PHP (*Hypertext Preprocessor*) merupakan sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaks mirip dengan bahasa C, java dan Perl, ditambah beberapa fungsi PHP yang spesifik. Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancangan *web* menulis halaman *web* dinamik dengan cepat. Hubungan PHP dengan HTML halaman *web* biasanya disusun dari kode-kode HTML yang disimpan dalam sebuah *file* berekstensi *.html*. file HTML ini dikirimkan oleh *server* (atau *file*) ke *browser*, kemudian *browser* menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program php, program ini harus diterjemahkan oleh *web server* sehingga menghasilkan kode html yang dikirim ke *browser* agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut. Program PHP dapat ditambahkan dengan mengapit program tersebut di antara tanda. Tanda-tanda tersebut biasanya disebut tanda untuk *escaping* (kabur) dari kode html.

PHP merupakan bahasa pemrograman *web* yang bersifat *server-side* HTML embedded *scripting*, dimana *script*-nya menyatu dengan HTML berada di *server*, artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di *server* tetapi disertakan HTML biasa (Kadir, 2013).

Kelebihan-kelebihan PHP adalah sebagai berikut:

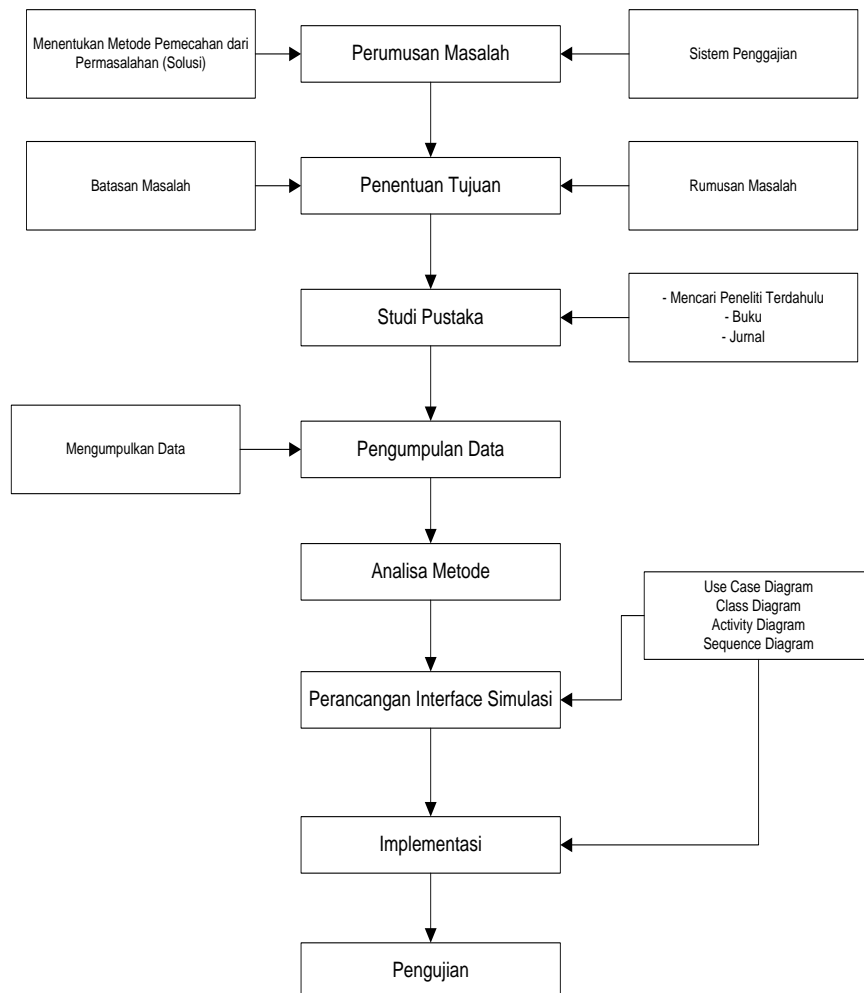
1. *Script* (kode program) terintegrasi dengan file HTML, sehingga *developer* (pengembang) bisa berkonsentrasi langsung pada penampilan *webnya*.
2. Tidak ada proses *compiling* dan *linking*.
3. Berorientasi obyek (*object oriented*).
4. Sintaksis pemrogramannya mudah dipelajari, mirip *C* dan *Perl*.
5. Integrasi yang sangat luas ke berbagai *server database*. *Database* yang didukung oleh PHP adalah: *Oracle*, *Sybase*, *MSQL*, *Solid*, *ODBC*, *PostgreSQL*, *Informix*, *dBase*, *UNIX DBM*.

BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Analisis sistem dilakukan untuk mendapatkan informasi penelitian terhadap sistem yang berjalan. Gambar 3.1 adalah tahapan penelitian penulis.



Gambar 3.1 Tahapan Penelitian

3.2 Metode Pengumpulan Data

Pengumpulan data adalah pencarian terhadap sesuatu karena ada perhatian dan keinginan terhadap hasil suatu aktivitas. Metode pengumpulan data dalam penulisan ini dibagi menjadi 2, yaitu:

1. Pengamatan (*Observation*)

Penulis melakukan pengamatan langsung pada Pegawai TAD.

2. Penelitian Kepustakaan (*Library Research*)

Merupakan cara untuk mencari referensi dengan mengumpulkan bahan-bahan pustaka yang dilakukan di perpustakaan kampus, maupun perpustakaan umum, juga melakukan pencarian lewat internet, dengan mengunjungi situs-situs seperti google Book online yang dapat membantu pembahasan materi.

3.3 Analisa Sistem yang sedang berjalan

Selama melakukan penelitian lapangan mengenai sistem yang sedang berjalan dalam hal penanganan sistem informasi penggajian pada Pegawai TAD *Mandiri University Medan Campus* maka Penulis dapat mengetahui beberapa kelemahan maupun kekurangan pada sistem yang sedang berjalan sebagai berikut:

1. Belum tersedianya perangkat lunak aplikasi sistem informasi penggajian pada Pegawai *Mandiri University Medan Campus*.
2. Pengolahan data penggajian pegawai masih menggunakan pencatatan secara manual pada komputer dengan menggunakan aplikasi *Microsoft Excel*, belum menggunakan aplikasi *database* dalam penyimpanan datanya, hal ini sangat tidak efektif dan efisien, dengan adanya perangkat

lunak aplikasi sistem informasi penggajian pada Pegawai TAD *Mandiri University Medan Campus* maka data penggajian dapat disimpan dengan aman, dan dapat diakses dengan mudah.

3.4 Perancangan Sistem Yang Diusulkan

Seiring dengan kemajuan teknologi, sudah saatnya setiap kantor-kantor atau perusahaan menerapkan teknologi komputer di dalam layanan informasi atau sistem informasi mengenai seluruh informasi mengenai perusahaan tersebut, baik dalam pengolahan data pegawai maupun data penggajian, dalam penelitian ini bertujuan untuk membantu Pegawai TAD *Mandiri University Medan Campus* dalam merancang sistem komputerisasi pengolahan data penggajian karyawan.

Pengguna sistem dalam hal ini terbagi atas tiga level yaitu bagian *admin*, bagian *HRD* dan bagian keuangan yang diperbolehkan untuk mengakses aplikasi tersebut, pada menu halaman *admin* dapat melakukan pengolahan data karyawan, data departemen dan data jabatan, pada menu halaman *HRD* dapat melakukan pengolahan data karyawan, data kinerja, data departemen dan data jabatan, sedangkan pada menu halaman keuangan dapat melakukan pengolahan proses input proses penggajian yang termasuk input penggajian, lihat data penggajian, lihat slip gaji dan melihat laporan penggajian setiap bulan.

3.5 Perancangan Prosedur Yang Diusulkan

Setelah melihat sistem yang sedang berjalan serta mengevaluasinya, maka prosedur perangkat lunak aplikasi sistem informasi penggajian yang diusulkan adalah sebagai berikut:

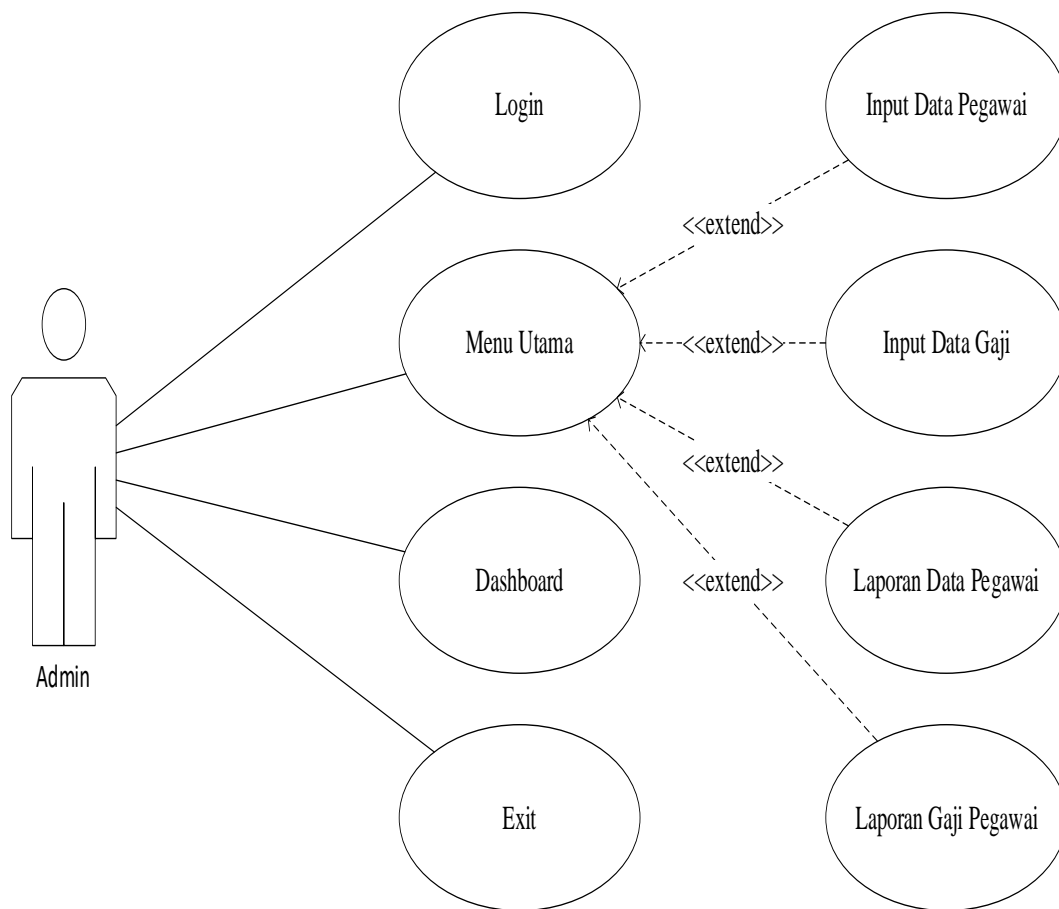
1. Sebelum masuk ke menu utama, karyawan harus memasukkan *NIP* dan *password*, jika benar maka akan masuk ke menu halaman Keuangan.
2. Pada menu data penggajian, seorang *keuangan* dapat melakukan *input*, *edit* dan hapus data penggajian dan dapat melihat data penggajian yang telah dimasukkan ke sistem.
3. Pada menu slip gaji, seorang *keuangan* dapat melakukan *input*, *edit* dan hapus data slip gaji dan data lembur dan dapat melihat data jurnal penggajian yang telah dimasukkan ke sistem.
4. Pada menu laporan, seorang *keuangan* dapat melakukan *input*, *edit* dan hapus data laporan penggajian dan dapat melihat data penggajian yang telah dimasukkan ke sistem.

3.6 Perancangan Sistem Secara Umum

Perancangan sistem adalah suatu upaya untuk membuat suatu sistem yang baru atau memperbaiki sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Tahapan perancangan sistem dilakukan setelah tahap analisis sistem dilakukan.

3.6.1 Use Case Diagram

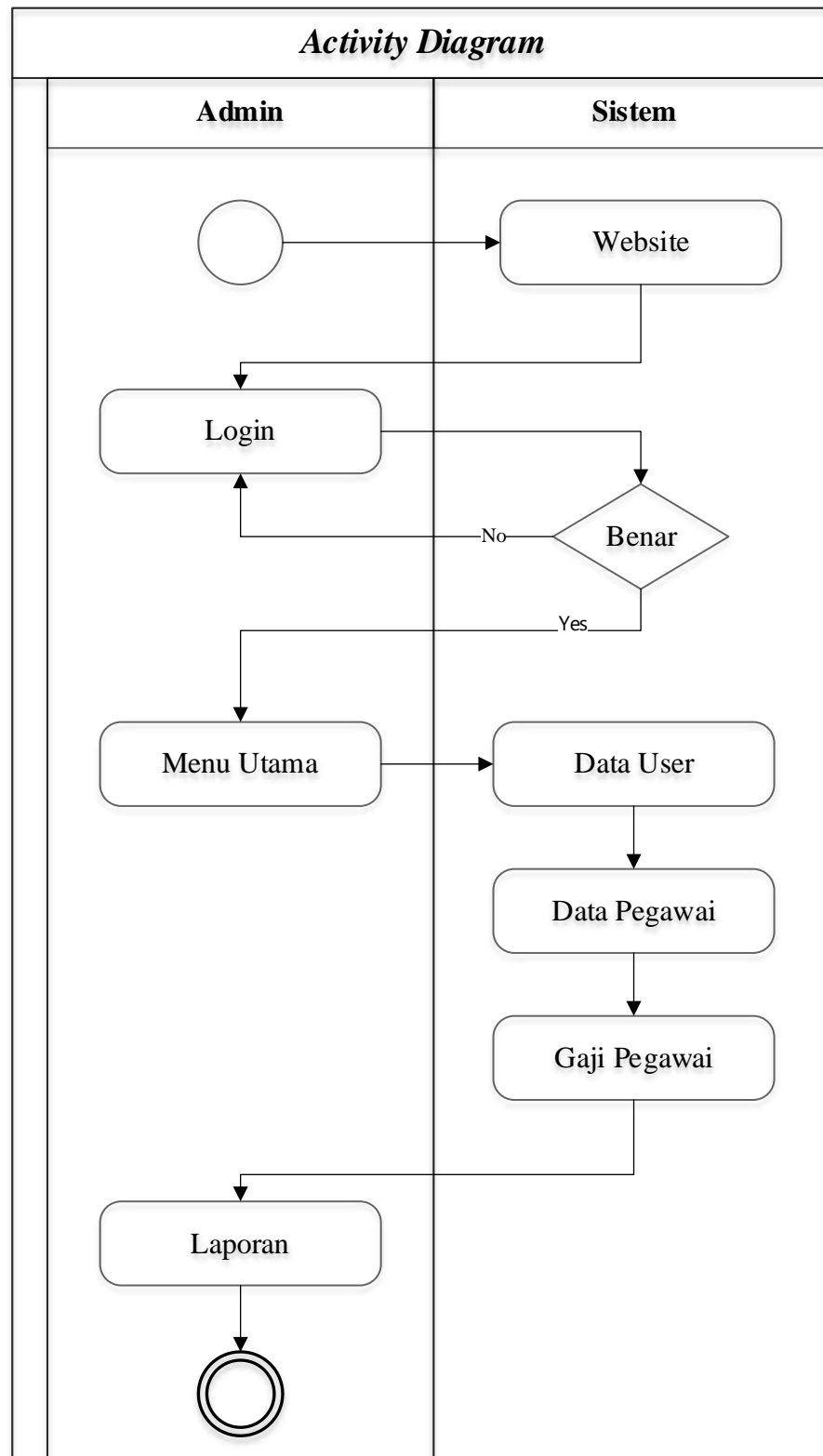
Use case merupakan deskripsi fungsi dari sebuah sistem dari perspektif pengguna (*user*). Gambar 3.2 adalah desain *use case diagram* yang digunakan dalam penelitian ini.



Gambar 3.2 Use Case Diagram

3.6.2 Activity Diagram

Activity Diagram merupakan diagram yang akan menggambarkan alur kegiatan dari sistem yang dilakukan pengguna (*user*) untuk menghasilkan aplikasi penggajian. *Activity Diagram* dari sistem informasi berfungsi untuk memberikan hasil rekomendasi laporan perbaikan dan penjualan mana yang dapat dijadikan sebagai panduan dalam pemberian gaji pegawai. Gambar 3.3 merupakan *Activity Diagram* dari aplikasi penggajian.

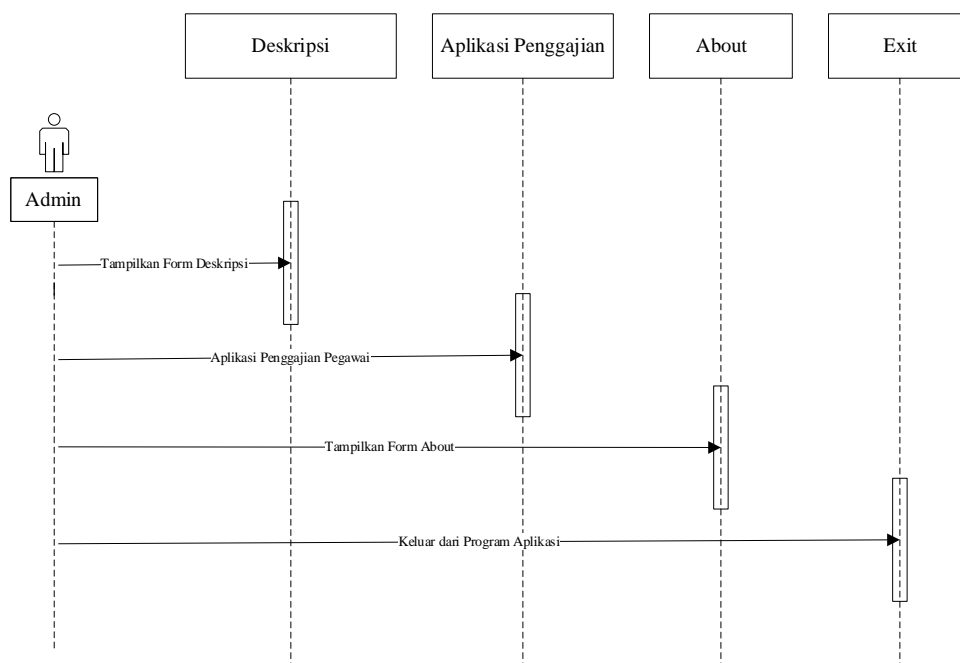


Gambar 3.3 Activity Diagram

3.6.3 Sequence Diagram

3.6.3.1 Menu Utama

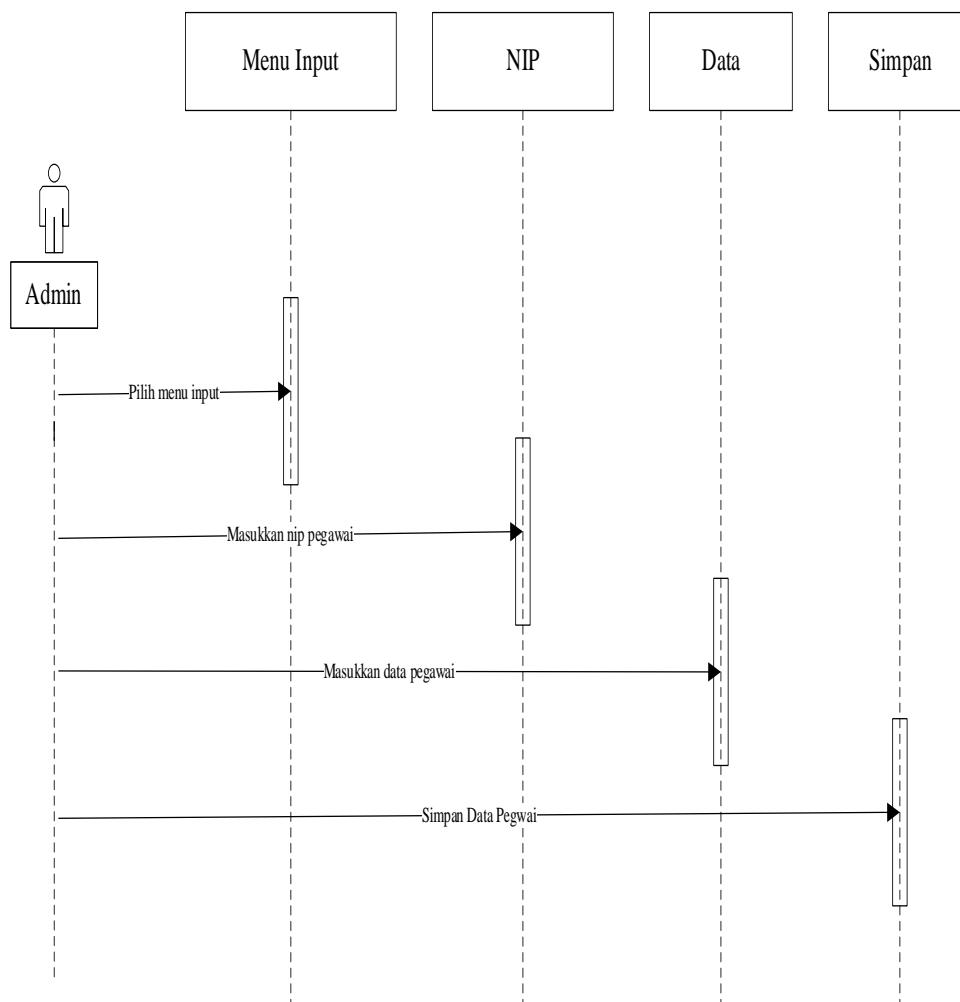
Sequence diagram menu utama digunakan untuk melihat urutan proses aplikasi. Diagram ini menjelaskan alur program menu utama sehingga memperlihatkan beberapa submenu. Gambar 3.4 adalah *Sequence diagram* menu utama yang digunakan.



Gambar 3.4 *Sequence Diagram* Menu Utama

3.6.3.2 Menu Input Data Pegawai

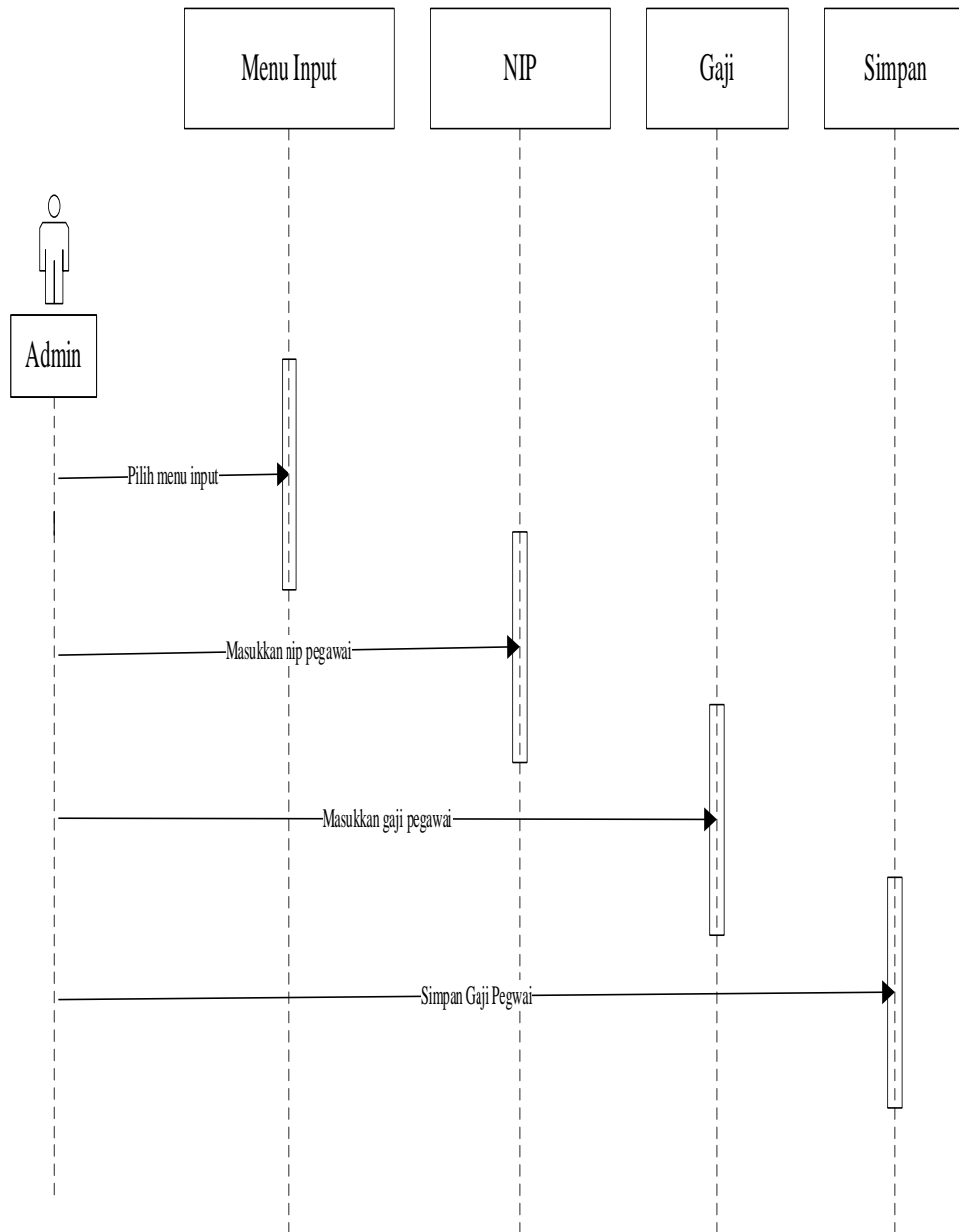
Sequence diagram input data pegawai digunakan untuk melihat urutan proses penginputan data pegawai. Gambar 3.5 adalah *Sequence diagram* menu input data pegawai yang digunakan.



Gambar 3.5 Sequence Diagram Input Data

3.6.3.3 Menu Input Gaji Pegawai

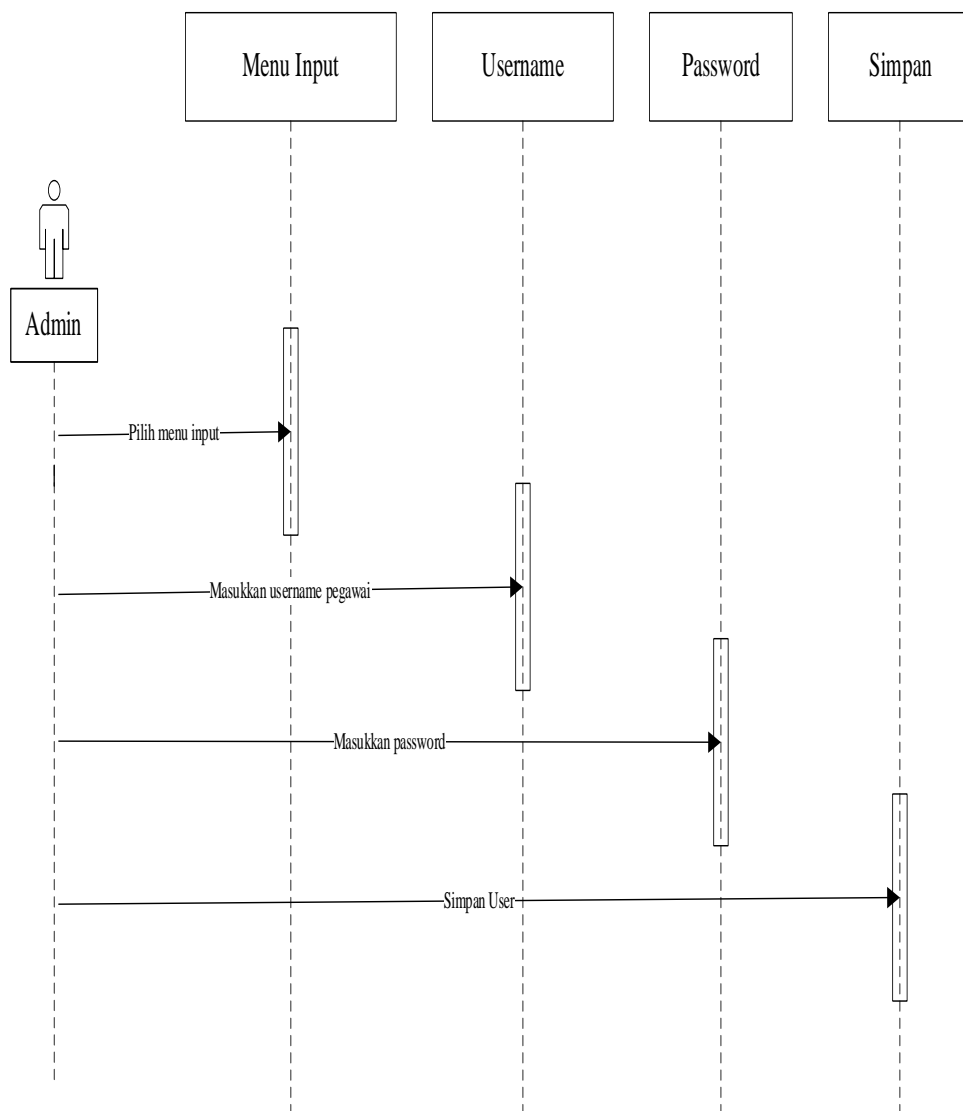
Sequence diagram input data pegawai digunakan untuk melihat urutan proses penginputan gaji pegawai. Gambar 3.6 adalah *Sequence diagram* menu input gaji pegawai yang digunakan.



Gambar 3.6 Sequence Diagram Input Gaji

3.6.3.4 Menu User

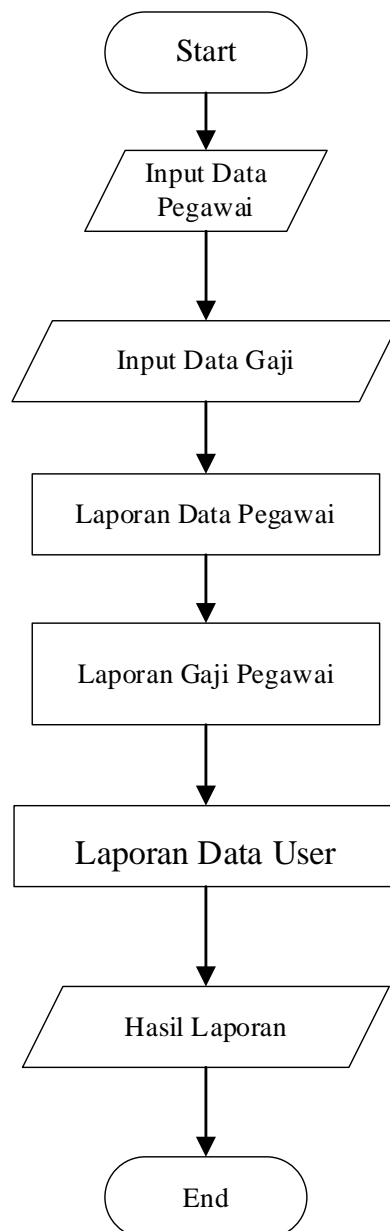
Sequence diagram user digunakan untuk melihat urutan proses pendaftaran akun pengguna. Gambar 3.7 adalah *Sequence diagram* user yang digunakan.



Gambar 3.7 *Sequence Diagram User*

3.6.4 Flowchart

Flowchart atau diagram alur berfungsi untuk menjelaskan langkah yang diambil dalam menghasilkan aplikasi penggajian. Gambar 3.8 adalah *Flowchart* cara kerja program aplikasi sistem penggajian.



Gambar 3.8 *Flowchart* Aplikasi Penggajian

3.7 Kebutuhan Antarmuka

Antarmuka pemakai atau *user interface* adalah bagian penghubung antara program dengan pengguna. Pengguna akan berhubungan dengan *server* melalui sebuah *program* yang dibuat menggunakan bahasa pemrograman *PHP*.

1. Kebutuhan Input

Kebutuhan *input* dari sistem informasi ini adalah sebagai berikut:

- a. *Input* data yang dibahas mencakup data yang berhubungan dengan pengolahan penggajian seperti *input* penggajian, *edit* dan hapus, data penggajian, data pegawai, data departemen dan data jabatan.

2. Kebutuhan Output

Kebutuhan *output* dari sistem informasi ini adalah sebagai berikut:

- a. Informasi data pegawai, data kinerja, data departemen, data jabatan, dan data penggajian.
- b. *Output* yang dihasilkan adalah dapat mencetak slip gaji dan laporan penggajian yang dapat diambil setiap bulannya.

3.8 Desain Form Antarmuka

Desain form antarmuka berguna untuk merancang atau mendesain tampilan aplikasi penggajian dengan menggunakan bahasa pemrograman *PHP* yang dibuat berbasis *web*. Tampilan yang dibuat terdiri dari beberapa tampilan yang mempunyai sebuah tampilan utama yang berfungsi sebagai pembuka

program aplikasi. Bagian berikut ini adalah tahapan desain tampilan dari aplikasi sistem penggajian tersebut.

3.8.1 Desain *Form Login*

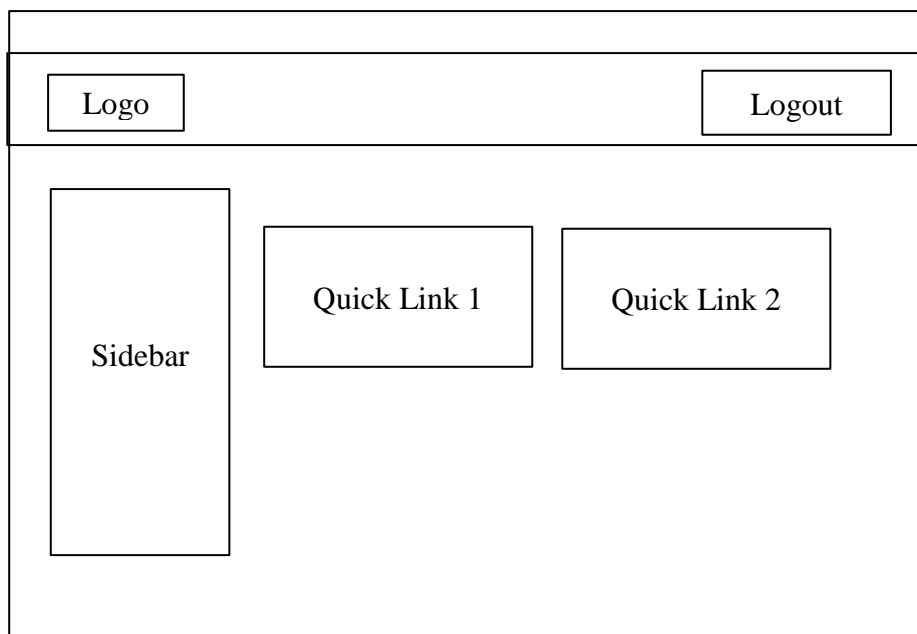
Halaman *login* merupakan halaman yang pertama diakses pada saat *program* dijalankan, halaman *login* berisi kolom *NIP* dan *password* yang harus diisi untuk masuk ke halaman admin. Gambar 3.6 adalah form *login*.

The diagram illustrates the layout of a login form. It is contained within a page structure with a 'HEADER' at the top and a 'FOOTER' at the bottom. The login form is centered and includes a title 'LOGIN', two input fields labeled 'NIP' and 'PASSWORD', and a 'LOGIN' button.

Gambar 3.9 *Form Login*

3.8.2 Desain *Form Dashboard*

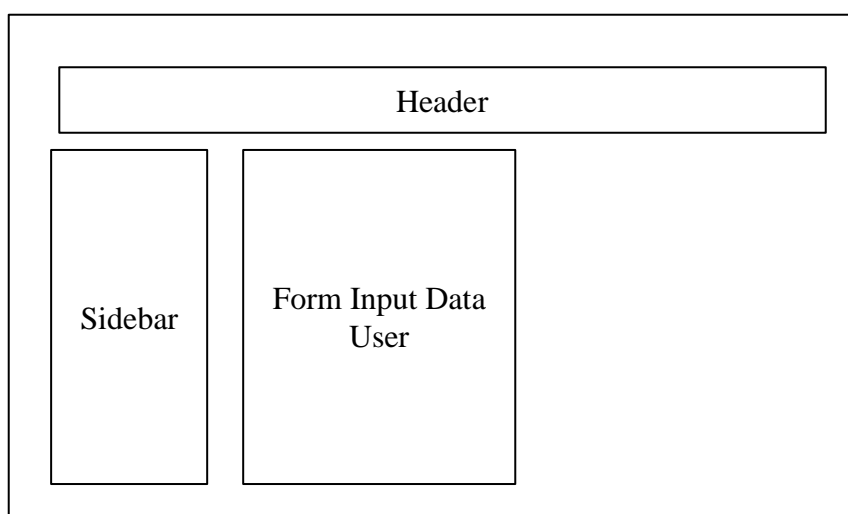
Desain menu utama merupakan halaman yang muncul pada saat program aplikasi dijalankan. Penampakan ini memiliki beberapa bagian yang terdiri dari komponen-komponen objek gambar, label dan tombol. Gambar 3.7 merupakan hasil desain menu utama.



Gambar 3.10 *Form Dashboard*

3.8.3 Desain *Form Input Data User*

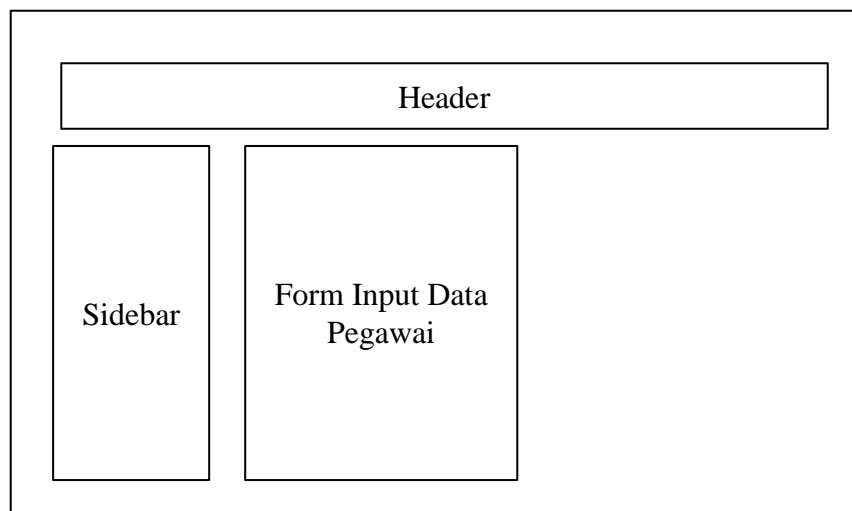
Form ini berfungsi untuk melakukan penambahan data user. Gambar 3.8 merupakan hasil desain menu input data user.



Gambar 3.11 *Form Input Data User*

3.8.4 Desain *Form Input Data Pegawai*

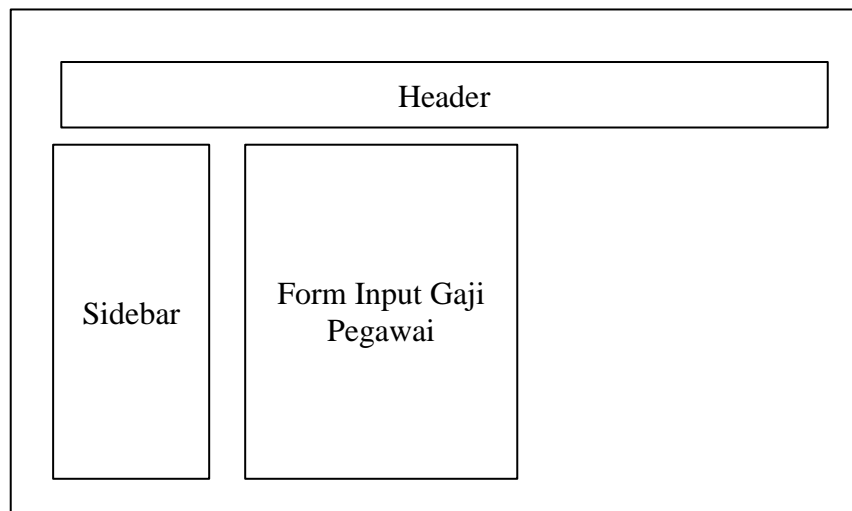
Form ini berfungsi untuk melakukan penambahan data pegawai. Gambar 3.9 merupakan hasil desain menu input data pegawai.



Gambar 3.12 *Form Input Data Pegawai*

3.8.5 Desain *Form Input Gaji Pegawai*

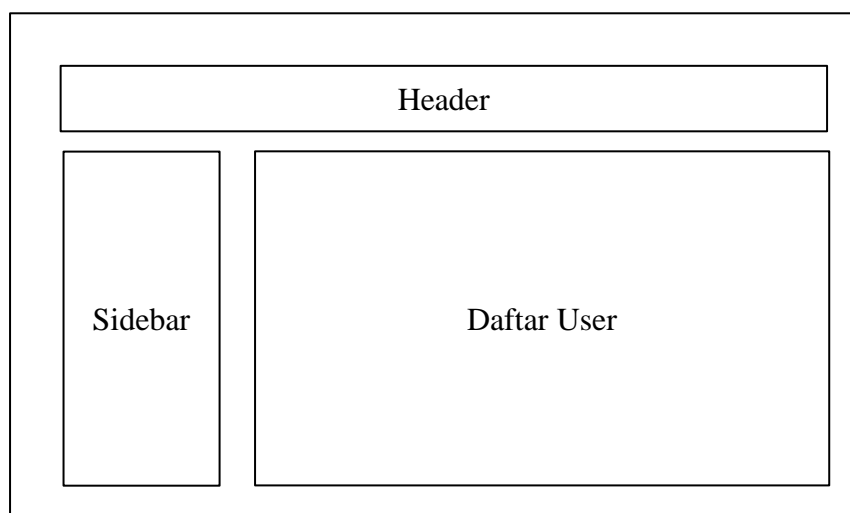
Form ini berfungsi untuk melakukan penambahan gaji pegawai. Gambar 3.10 merupakan hasil desain menu input gaji pegawai.



Gambar 3.13 *Form Input Gaji Pegawai*

3.8.6 Desain *Form Laporan Data User*

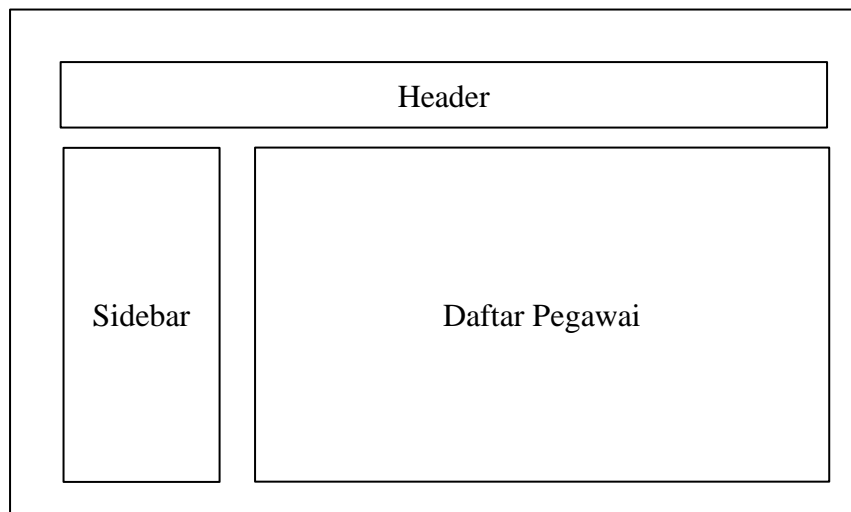
Halaman laporan data user merupakan halaman yang berisi informasi mengenai user yang terdaftar. Gambar 3.11 adalah desain menu laporan data user.



Gambar 3.14 *Form Laporan Data User*

3.8.7 Desain *Form* Laporan Data Pegawai

Halaman laporan data pegawai merupakan halaman yang berisi informasi mengenai pegawai yang terdaftar. Gambar 3.12 adalah desain menu laporan data pegawai.



Gambar 3.15 *Form* Laporan Data Pegawai

3.8.8 Desain *Form* Laporan Gaji Pegawai

Halaman laporan gaji pegawai merupakan halaman yang berisi informasi mengenai data slip gaji pegawai. Gambar 3.13 adalah desain menu laporan gaji pegawai.

HEAD						
Form	Penggaji	Pajak P	Slip G:	Lapor:	Logo	
Data Slip Gaji Pe						
						<input type="button" value="Pencari"/>
Nil	Nama Pega	Jabat:	Tanggal Gā	Gaji Pok	Gaji Ber:	Aks
XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	Update Delete
XXXX	XXXX	XXXX	XXXX	XXXX	XXXX	Update Delete

Gambar 3.16 Form Laporan Gaji Pegawai

3.9 Desain Basis Data

Berikut ini adalah basis data yang digunakan dalam membuat program aplikasi.

Tabel 3.1 Database User

Variabel	Tipe	Ukuran	Keterangan
nama	Varchar	15	Nama
username	Varchar	20	Username
password	Varchar	20	Kata Kunci
level	Varchar	30	Level

Tabel 3.2 Database Pegawai

Variabel	Tipe	Ukuran	Keterangan
nip	Varchar	15	NIP
nama	Varchar	50	Nama Pegawai
tanggal_lahir	Varchar	10	Tanggal Lahir
Tempat_lahir	Varchar	30	Tempat Lahir
Agama	Varchar	30	Agama Pegawai
no_telp	Varchar	30	Nomor Telepon
alamat	Varchar	50	Alamat Pegawai
kota	Varchar	20	Kota
status	Varchar	15	Status Pernikahan
jmlanak	Int	5	Jumlah Anak
pendidikan	Varchar	15	Pendidikan

Tabel 3.3 Database Gaji

Variabel	Tipe	Ukuran	Keterangan
nip	Varchar	50	NIP
nama	Varch	50	Nama Pegawai
bulan	Int	11	Bulan
tahun	Int	11	Tahun
uang_lembur	Int	11	Uang Lembur
uang_makan	Int	11	Uang Makan
uang_transport	Int	11	Uang Transport
tunj_istri	Int	11	Tunjangan Istri
tunj_anak	Int	11	Tunjangan Anak
potongan	Int	11	Potongan
gaji_kotor	Int	11	Gaji Kotor
gaji_bersih	Int	11	Gaji Bersih

Tabel 3.1 hingga 3.2 adalah tabel relasi yang digunakan dalam pembuatan database untuk menampung segala kegiatan dari aplikasi sistem penggajian.

3.10 Relasi Tabel

Berikut ini adalah tabel relasi yang dibangun dalam membuat program aplikasi.

tb_user
Nama
username
password
Level
+create()
+read()
+update()
+delete()

tb_pegawai		tb_gaji
nip	—	nip
nama		nama
tanggal_lahir		bulan
Tempat_lahir		tahun
Agama		uang_lembur
no_telp		uang_makan
alamat		uang_transport
kota		tunj_istri
status		tunj_anak
jmlanak		potongan
pendidikan		gaji_kotor
+create()		gaji_bersih
+read()		+create()
+update()		+read()
+delete()		+update()
		+delete()

BAB IV

HASIL DAN PEMBAHASAN

Aplikasi ini sangat penting dalam meringankan kerja kepegawaian di *Mandiri University* dalam menentukan laporan gaji. Hasil perancangan akan diimplementasikan berbentuk *website* sederhana yang dapat mengolah tiga bagian utama yaitu, user, data pegawai dan gaji pegawai. Tetapi dalam menjalankan sistem tersebut ada kebutuhan perangkat yang diperlukan.

4.1 Hasil

4.1.1 Kebutuhan Perangkat Keras dan Lunak

Dalam menciptakan penelitian, ada beberapa perangkat pendukung yang harus dipenuhi yaitu perangkat keras dan perangkat lunak. Perangkat-perangkat ini saling bekerja sama dalam memenuhi persyaratan untuk terciptanya program aplikasi tersebut. Perangkat keras bertujuan untuk sebagai *host* dalam letak program aplikasi tersebut, sementara perangkat lunak akan menghasilkan perangkat lunak juga yang disebut dengan program aplikasi. Berikut ini adalah spesifikasi perangkat keras dan lunak yang dibutuhkan dalam penelitian.

4.1.1.1 Spesifikasi Perangkat Keras

Metode *Composite Performance Index* yang dipakai sangat membutuhkan perangkat keras. Perangkat keras adalah sebagai mesin utama yang akan

mengoperasikan perangkat lunak. Spesifikasi perangkat keras yang digunakan dapat dilihat pada tabel 4.1.

Tabel 4.1 Spesifikasi Perangkat Keras

No.	Nama Komponen	Spesifikasi
1	Processor	Intel Core i3 1.9 GHz
2	RAM	4 GB
3	Harddisk	500 GB
4	Monitor	14 inch
5	Mouse	Wireless 2.4 GHz
6	Keyboard	Lenovo
7	Printer	EPSON L360

4.1.1.2 Spesifikasi Perangkat Lunak

Perangkat lunak dibutuhkan dalam menciptakan program aplikasi yang diprogram menggunakan bahasa pemrograman. Tabel 4.2 adalah spesifikasi perangkat lunak yang digunakan dalam penelitian ini.

Tabel 4.2 Spesifikasi Perangkat Lunak

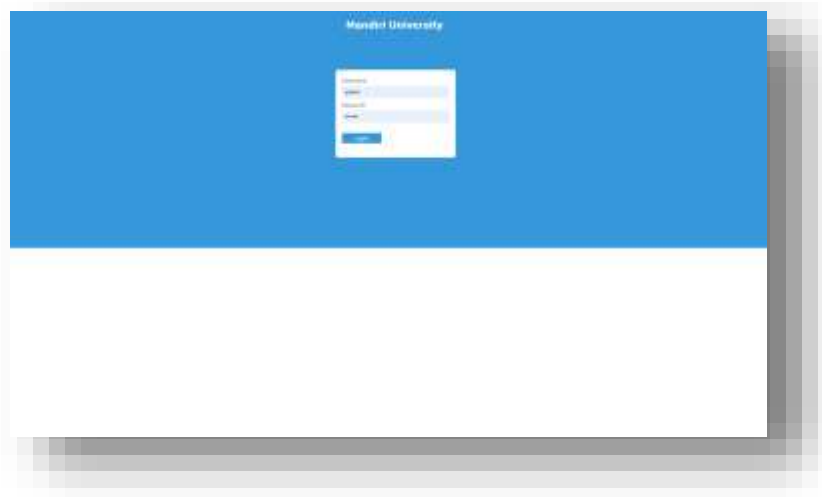
No.	Nama Komponen	Spesifikasi
1	Sistem Operasi	Windows 7 32 Bit
2	IDE Pemrograman	Notepad++
3	Screenshot	Snipping Tool
4	Data Editor	Microsoft Excel 2019
5	Word Processing	Microsoft Word 2019
6	Web Server	Apache
7	Database Server	MySQL phpMyAdmin

4.1.2 Hasil Program Aplikasi

Program aplikasi dibutuhkan dalam menentukan kulkas terbaik melalui perantaraan metode yang digunakan. Program aplikasi ini dibuat dengan menggunakan bahasa pemrograman *PHP*.

4.1.2.1 Halaman *Form Login*

Form login untuk menentukan pengguna mana yang mengakses *website* sistem informasi penggajian. Login memiliki dua buah level akses yaitu, administrator dan akun biasa. Akun administrator dapat melakukan perubahan data dan penambahan user, sementara akun biasa tidak dapat melakukan hal tersebut dan hanya dapat melihat laporan data pegawai dan penggajian saja. Gambar 4.1 merupakan hasil tampilan halaman login.



Gambar 4.1 Halaman login

4.1.2.2 Halaman *Form* Dashboard

Form dashboard bertujuan untuk sebagai pembuka program aplikasi. Form terdiri dari header, sidebar dan konten dimana bagian konten akan menampilkan informasi penggajian. Pada halaman dashboard, ada dua buah *quick link* yang bertujuan untuk mengarahkan pengguna untuk mengakses *website* dari bank Mandiri dan Mandiri Syariah. Gambar 4.2 merupakan hasil tampilan dashboard.



Gambar 4.2 Halaman dashboard

4.1.2.3 Halaman Input Data User

Halaman input data user bertujuan untuk menambahkan data pengguna aplikasi dan hanya dapat dilakukan dengan level akses administrator. Gambar 4.3 adalah tampilan dari halaman input data user.

The image shows a web application interface for user data entry. On the left, there is a dark sidebar with a user profile icon and a list of menu items. The main content area is titled 'Input Data User' and contains several input fields: 'Username', 'Password', and 'Email'. Below these fields are two buttons: 'Save' and 'Cancel'. A large red prohibition sign (a circle with a diagonal slash) is overlaid on the form, indicating that the form is currently disabled or not accessible.

Gambar 4.3 Halaman input data user

4.1.2.4 Halaman Input Data Pegawai

Halaman input data pegawai berfungsi untuk menambahkan data pegawai pada Mandiri University. Halaman ini juga hanya dapat diakses oleh level administrator. Gambar 4.4 merupakan tampilan dari halaman input data pegawai.

The image shows a web application interface for employee data entry. On the left, there is a dark sidebar with a user profile icon and a list of menu items. The main content area is titled 'Input Data Pegawai' and contains several input fields: 'Name', 'Username', 'Password', 'Email', and 'Address'. Below these fields are two buttons: 'Save' and 'Cancel'.

Gambar 4.4 Halaman input data pegawai

4.1.2.5 Halaman Input Gaji Pegawai

Halaman input gaji pegawai adalah halaman yang berfungsi untuk memasukkan gaji bulanan dari pegawai TAD di Mandiri University. Gambar 4.5 merupakan tampilan dari halaman input gaji pegawai.

The image shows a web browser window displaying a form titled "Input Gaji Pegawai". On the left side, there is a dark sidebar menu with a user profile icon and several menu items. The main content area contains a form with multiple input fields, each with a blue header and a white body. The fields are labeled as follows: "No.", "Nama", "Jabatan", "Gaji Pokok", "Gaji Tunjangan", "Gaji Lembur", "Gaji Istimewa", "Gaji Lain-lain", "Gaji Total", "Gaji Dibayar", "Gaji Belum Dibayar", "Gaji Bersih", "Gaji Bruto", "Gaji Kotor", "Gaji Murni", "Gaji Pokok", "Gaji Tunjangan", "Gaji Lembur", "Gaji Istimewa", "Gaji Lain-lain", "Gaji Total", "Gaji Dibayar", "Gaji Belum Dibayar", "Gaji Bersih", "Gaji Bruto", "Gaji Kotor", "Gaji Murni". At the bottom of the form, there is a blue button labeled "Simpan Data".

Gambar 4.5 Halaman input gaji pegawai

4.1.2.6 Halaman Laporan Data User

Halaman data user merupakan halaman yang menampilkan daftar pengguna yang terhubung ke aplikasi. Halaman ini memiliki foto profil untuk menambah informasi terhadap user yang terhubung. Halaman ini juga harus diakses oleh administrator agar tidak terjadi penyalahgunaan data kebocoran informasi. Gambar 4.6 merupakan tampilan dari halaman laporan data user.

Gambar 4.6 Halaman laporan data user


4.1.2.7 Halaman Laporan Data Pegawai

Halaman data pegawai menampilkan daftar pegawai yang bekerja di *Mandiri University* yang berstatus Tenaga Ahli Madya. Gambar 4.7 merupakan tampilan dari halaman laporan data pegawai.

Gambar 4.7 Halaman laporan data pegawai

4.1.2.8 Halaman Laporan Gaji Pegawai

Halaman laporan gaji pegawai adalah hasil rekap data dari pegawai selama periode tertentu. Laporan ini juga dapat diubah atau dihapus hanya dengan akun yang berlevel administrator. Gambar 4.8 merupakan tampilan dari halaman laporan gaji pegawai.



No	Nama	Posisi	Gaji Pokok	Gaji Tunjangan	Gaji Lainnya	Total	Status
1	Andrianto, A.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
2	Budi Santia, B.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
3	Chandra, C.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
4	Dewi, D.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
5	Eko, E.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
6	Fani, F.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
7	Gani, G.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
8	Hani, H.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
9	Iani, I.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif
10	Jani, J.	Asisten Ahli	10.000.000	1.000.000	500.000	11.500.000	aktif

Gambar 4.8 Halaman laporan gaji pegawai

4.2 Pembahasan

Program aplikasi yang dirancang telah diuji dan diimplementasikan kepada sistem penggajian pegawai TAD di Mandiri University. Ada beberapa penilaian terhadap hasil percobaan program aplikasi berbasis web. Setiap program aplikasi tentu saja memiliki kelebihan dan kekurangan sehingga perlu dikembangkan oleh peneliti selanjutnya.

4.2.1 Pengguna Aplikasi

Setiap pengguna aplikasi memiliki hak akses yang berbeda. Ada dua jenis akun yang terdaftar yaitu administrator dan operator. Tugas operator adalah melihat dan memasukkan data gaji dan pegawai. Sementara untuk akun yang memiliki level administrator dapat menambahkan pengguna atau *user* ke dalam sistem.

4.2.2 Kelebihan Sistem

Dengan terciptanya program aplikasi ini, dapat dilihat ada beberapa kelebihan yang dapat dinilai, antara lain:

1. Sistem memudahkan pihak admin dan operator penggajian dalam memasukkan data gaji.
2. Sistem memberikan pembatasan akses terhadap jenis-jenis akun.
3. Kesalahan data gaji dapat diminimalisir dengan sistem yang dapat memodifikasi data.

4.2.3 Kekurangan Sistem

Terlepas dari kelebihan sistem, ada beberapa kekurangan yang dapat dikembangkan ke depannya, antara lain:

1. Sistem tidak terhubung dengan penggajian otomatis atau *payroll*.
2. Sistem tidak dapat memberikan notifikasi gaji masuk kepada pegawai.
3. Sistem hanya dikhususkan kepada pihak pegawai *Mandiri University* saja.

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melakukan penelitian di *Mandiri University*, ada beberapa kesimpulan yang dapat diambil, antara lain:

1. Sistem penggajian dapat membantuk pihak Mandiri University dalam mengelola gaji pegawai Tenaga Ahli Madya.
2. Aplikasi berbasis *web* yang dibuat dapat menyimpan informasi user, data pegawai dan gaji pegawai pada database yang terhubung.
3. Kriteria yang diambil ada sebanyak enam dan kriteria-kriteria tersebut merupakan kriteria yang tepat dalam menentukan guru honor.
4. Bobot preferensi merupakan penyeimbang nilai kriteria dalam menentukan prioritas kriteria yang akan digunakan.

5.2 Saran

Penelitian ini dapat dikembangkan sehingga meningkatkan kemampuan sistem informasi penggajian. Ada beberapa saran yang dapat mengembangkan penelitian ini, antara lain:

1. Hendaknya sistem informasi dapat terhubung dengan sistem absen dalam melakukan perhitungan dan potongan gaji.
2. Hendaknya aplikasi dapat melakukan transfer otomatis gaji melalui sistem *payroll* perusahaan.

DAFTAR PUSTAKA

- Andriani, Y., Ramli, N. M., Syamsumir, D. F., Kassim, M. N. I., Jaafar, J., Aziz, N. A., ... & Mohamad, H. (2019). Phytochemical analysis, antioxidant, antibacterial and cytotoxicity properties of keys and cores part of Pandanus tectorius fruits. *Arabian Journal of Chemistry*, 12(8), 3555-3564.
- Berisha-Shaqiri, A. (2015). Management Information System and Competitive Advantage. *Mediterranean Journal of Social Sciences*. <https://doi.org/10.5901/mjss.2015.v6n1p204>
- Hartati, S., & Iswanti, S. (2006). *Sistem Pakar dan Pengembangannya*. Graha Ilmu.
- Hung, N. V., van Hung, P., & Anh, B. T. (2018). Database Design For E-Governance Applications: A Framework For The Management Information Systems Of The Vietnam Committee For Ethnic Minority Affairs (CEMA). *International Journal of Civil Service Reform and Practice*, 3(1).
- Jogiyanto, H. M. (2016). *Analisis Dan Desain Sistem Informasi, Pendekatan Terstruktur Teori Dan Praktek Aplikasi Bisnis*. Andi Offset.
- Kadir, A. (2013). *Membuat Aplikasi Web dengan PHP + Database MySQL*. Penerbit Andi.
- Kurniawan, T. A. (2018). Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(1), 77. <https://doi.org/10.25126/jtiik.201851610>
- Kustiyarningsih, Y., & Devie, R. A. (2011). Pemrograman Basis Data Berbasis Web Dengan Menggunakan PHP & MySQL. In *Edisi Pertama, Graha Ilmu, Yogyakarta*. Graha Ilmu.
- Ladjamudin, A.-B. bin. (2017). *Analisis dan Desain Sistem Informasi*. Graha Ilmu.
- Mangematin, V., & Baden-Fuller, C. (2008). Global Contexts in the Production of Business Knowledge. *Long Range Planning*, 41(1), 117-139. <https://doi.org/10.1016/j.lrp.2007.11.005>
- Nugroho, B. (2013). *Dasar Pemograman Web PHP – MySQL dengan Dreamweaver*. Gava Media.
- Pratama, A. N. W. (2010). *CodeIgniter: Cara Mudah Membangun Aplikasi PHP*. <https://books.google.com/books?id=n22BYuK0h98C&pgis=1>
- Perwitasari, I. D. (2018). Teknik Marker Based Tracking Augmented Reality untuk Visualisasi Anatomi Organ Tubuh Manusia Berbasis Android. *INTECOMS: Journal of Information Technology and Computer Science*, 1(1), 8-18.
- Rahmel, D. (2018). *Visual Basic.NET*. McGraw-Hill.
- Rusdina, R., Syafarina, G. A., & Amin, M. (2020). PROTOTYPE APLIKASI BANJAR BERBASIS ANDROID STUDIO SEBAGAI SALAH SATU PETUNJUK WISATA DI BANJARMASIN. *Technologia: Jurnal Ilmiah*, 11(1), 59-63.

- Santosa, A., Sitopu, M. W., Sirait, D. N., & Nasution, D. (2021). Analysis of Damage to Localizer Equipment (Case Study of Sultan Iskandar Muda Airport, Banda Aceh). *Budapest International Research and Critics Institute (BIRCI-Journal): Humanities and Social Sciences*, 4(3), 7054-7061.
- Shiau, W.-L. (2015). MANAGEMENT INFORMATION SYSTEMS ISSUES: CO-CITATION ANALYSIS OF JOURNAL ARTICLES. *International Journal of Electronic Commerce Studies*, 6(1), 145–162. <https://doi.org/10.7903/ijecs.1393>
- Sukmawati, R., & Priyadi, Y. (2019). Perancangan Proses Bisnis Menggunakan UML Berdasarkan Fit/Gap Analysis Pada Modul Inventory Odoo. *INTENSIF: Jurnal Ilmiah Penelitian Dan Penerapan Teknologi Sistem Informasi*, 3(2), 104. <https://doi.org/10.29407/intensif.v3i2.12697>
- Technopedia. (2019). *Unified Modeling Language (UML)*. Technopedia. <https://www.techopedia.com/definition/3243/unified-modeling-language-uml>
- Tempo.co. (2013). *Bank Mandiri Dirikan Mandiri University*. Tempo. <https://bisnis.tempo.co/read/468604/bank-mandiri-dirikan-mandiri-university/full&view=ok>
- UTM. (2019). *Concept: Use-Case Model*. Univesidad Technologica de La Mixteca. http://www.utm.mx/~caff/doc/OpenUPWeb/openup/guidances/concepts/use_case_model_CD178AF9.html
- Wasserkrug, S., Dalvi, N., Munson, E. V., Gogolla, M., Sirangelo, C., Fischer-Hübner, S., Ives, Z., Velegrakis, Y., Bevan, N., Jensen, C. S., & Snodgrass, R. T. (2019). Unified Modeling Language. In *Encyclopedia of Database Systems* (pp. 3232–3239). Springer US. https://doi.org/10.1007/978-0-387-39940-9_440
- Yakub. (2012). *Pengantar Sistem Informasi*. Graha Ilmu.