

LAPORAN PENELITIAN

**PENENTUAN PENERIMAAN REMUNERASI DOSEN
DENGAN RULE BASED REASONING**



PENELITI:

**Samsudin, S.T, M.Kom
NIP.19761227 201101 1002**

**LEMBAGA PENELITIAN DAN PENGABDIAN KEPADA
MASYARAKAT (LP2M)
UNIVERSITAS ISLAM NEGERI (UIN)
SUMATERA UTARA
MEDAN
2018**

LEMBAR PENGESAHAN

1. Penelitian
 - a. Judul Penelitian : Penentuan Penerimaan Remunerasi Dosen Dengan Rule Based Reasoning
 - b. Kluster Penelitian : Pembinaan/Peningkatan Kapasitas
 - c. Bidang Keilmuan : Sains dan Teknologi
 - d. Kategori : Individu
2. Peneliti : Samsudin, S.T., M.Kom
3. ID Peneliti : 202712760110000
4. Unit Kerja : Fakultas Sains dan Teknologi UIN Sumatera Utara Medan
5. Waktu Penelitian : 5 Bulan 2018
6. Lokasi Penelitian : UIN Sumatera Utara Medan
7. Biaya Penelitian : Rp. 15.000.000,- (*Lima belas Juta Rupiah*)

Disahkan oleh Ketua
Lembaga Penelitian dan Pengabdian
kepada Masyarakat (LP2M) UIN
Sumatera Utara Medan

Medan, 30 Oktober 2018
Peneliti,



Prof. Dr. Pagar, M.Ag.
NIP. 195812311988031016

A handwritten signature in black ink, appearing to be 'Samsudin', with a long horizontal line extending from the end.

Samsudin, S.T., M.Kom
NIP. 19761227 201101 1002

SURAT PERNYATAAN BEBAS PLAGIASI

Yang bertanda tangan di bawah ini;

Nama : Samsudin, S.T., M.Kom

Jabatan : Dosen

Unit Kerja : Fakultas Sains dan Teknologi UIN Sumatera Utara Medan

Alamat : JL. Jermal 10 No. 78 Medan Denai

Dengan ini menyatakan bahwa:

1. Judul penelitian “**Penentuan Penerimaan Remunerasi Dosen Dengan Rule Based Reasoning**” merupakan karya orisinal saya.
2. Jika di kemudian hari ditemukan fakta bahwa judul, hasil atau bagian dari laporan penelitian saya merupakan karya orang lain dan/atau plagiasi, maka saya akan bertanggung jawab untuk mengembalikan 100% dana hibah penelitian yang telah saya terima, dan siap mendapatkan sanksi sesuai ketentuan yang berlaku.

Demikian pernyataan ini dibuat untuk digunakan sebagaimana mestinya.

Medan, 30 Oktober 2018
Yang Menyatakan,



Samsudin, S.T., M.Kom
NIP.19761227 201101 1002

REKOMENDASI

Setelah membaca dan menelaah hasil penelitian yang berjudul “Penentuan Penerimaan Remunerasi Dosen Dengan Rule Based Reasoning” yang dilakukan oleh Samsudin, S.T., M.Kom. Maka saya berkesimpulan bahwa hasil penelitian ini dapat diterima sebagai karya tulis berupa hasil penelitian. Demikianlah rekomendasi diberikan kepada yang bersangkutan untuk dapat dipergunakan sebagai mana mestinya.

Medan, 25 Oktober 2018
Konsultan



M. Irwan Padli Nasution, S.T.,MM, M.Kom
NIP. 19750213 200604 1003

ABSTRAK

Remunerasi merupakan sebuah istilah yang berhubungan dengan penggajian pegawai yang ditetapkan dengan peraturan tertentu secara rutin berdasarkan nilai-nilai kerja, dengan tujuan terciptanya tata kelola yang lebih baik dan bersih serta meningkatkan motivasi dan prestasi kerja. Kinerja ditentukan dengan pengumpulan bukti kerja kepada pihak yang bertanggung jawab dan dihitung oleh badan yang ditentukan. *Rule Base Reasoning* adalah sistem pakar berdasarkan serangkaian aturan-aturan yang merupakan representasi dari pengetahuan dan pengalaman manusia dalam memecahkan kasus yang rumit. Sistem pakar adalah suatu sistem yang berusaha mengadopsi pengetahuan manusia ke dalam komputer agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para pakar. Untuk mengimplementasikan metode ini dibuat sebuah sistem berbasis web menggunakan bahasa pemrograman PHP dengan konsep *Object Oriented Programming* dengan harapan sistem ini bisa dirancang lebih mudah dan bisa dikembangkan secara berkelanjutan dan dapat mengoptimalkan penerimaan remunerasi dosen sehingga bisa memperkecil kemungkinan terjadinya kesalahan karena *human error* yang bisa menyebabkan kerugian pada pihak institusi maupun dosen.

Kata Kunci : Remunerasi, *Rule Base Reasoning*, Kinerja, Dosen, Sistem Pakar

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillah, puji syukur kami ucapkan kehadirat Allah SWT yang telah melimpahkan rahmat, hidayah dan karunia-Nya berupa kesehatan, kekuatan, pengetahuan, pengalaman dan kesempatan kepada peneliti, sehingga mampu menyelesaikan laporan penelitian ini dengan baik.

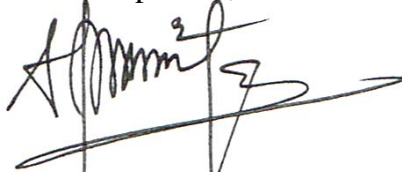
Penelitian ini berjudul “**Penentuan Penerimaan Remunerasi Dosen Dengan Rule Based Reasoning**”. Laporan penelitian ini disusun sebagai bukti telah selesainya Penelitian Pembinaan/Peningkatan Kapasitas dari Litapdimas Kemenag RI yang peneliti ikuti.

Peneliti menyadari bahwa dalam penulisan laporan penelitian ini, masih banyak kekurangan dan belum sempurna. Sehingga sangat diharapkan sumbangan pemikiran yang membangun dari semua pembaca dan rekan-rekan sejawat terutama dari dosen-dosen senior demi kesempurnaan penelitian ini. Semoga laporan penelitian ini dapat diperkaya melalui evaluasi terus menerus.

Harapan penulis semoga penelitian ini kelak dapat dipergunakan semaksimal mungkin serta dapat menambah wawasan dan dapat bermanfaat bagi siapa pun yang membacanya.

Medan, 30 Oktober 2018

Hormat penulis,



Samsudin, S.T, M.Kom

NIP. 19761227 201101 1002

DAFTAR ISI

LEMBAR PENGESAHAN	i
SURAT PERNYATAAN	ii
REKOMENDASI	iii
ABSTRAK	iv
KATA PENGANTAR	v
DAFTAR ISI	vi
DAFTAR TABEL	viii
DAFTAR GAMBAR	ix
I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.2.1 Identifikasi Permasalahan	2
1.2.2 Batasan Permasalahan	2
1.2.3 Perumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Signifikansi	3
1.5 Sistematika Penulisan	3
II. LANDASAN TEORI	4
2.1 Perancangan Sistem Informasi	4
2.1.1 Perancangan	4
2.1.2 Sistem	4
2.1.3 Karakteristik Sistem	4
2.1.4 Informasi	5
2.1.5 Kualitas Informasi	6
2.1.6 Sistem Informasi	6
2.1.7 Perancangan Sistem	7
2.2 <i>Database</i>	8
2.2.1 Komponen <i>Database</i> (Basis Data)	8
2.3 Teknik Normalisasi	8
2.3.1 Tujuan Normalisasi	9
2.3.2 Proses Normalisasi	9
2.3.3 Bentuk Normalisasi	9
2.4 <i>Sistem Pakar</i>	9
2.4.1 Komponen Sistem Pakar	10
2.4.2 <i>Rule Based Reasoning</i>	10
2.5 PHP Data Object	11
2.6 <i>Unified Modelling Language (UML)</i>	11
2.6.1 <i>Use Case Diagram</i>	12
2.6.2 <i>Class Diagram</i>	14
2.6.3 <i>Statechart Diagram</i>	14

2.6.4	<i>Activity Diagram</i>	15
2.6.5	<i>Sequence Diagram</i>	16
2.6.6	<i>Collaboration Diagram</i>	17
2.6.7	<i>Component Diagram</i>	17
2.6.8	<i>Package Diagram</i>	17
2.6.9	<i>Deployment Diagram</i>	17
2.7	Perangkat Lunak yang Digunakan	18
2.7.1	Bahasa Pemrograman PHP	18
2.7.2	MySQL	18
2.7.3	Web Browser	19
III.	METODOLOGI PENELITIAN	20
3.1	Pengumpulan Data	20
3.2	<i>Metode Rule Base Reasoning</i>	20
3.3	Kerangka Kerja	21
IV.	PERANCANGAN SISTEM	24
4.1	Arsitektur Sistem	24
4.2	<i>Rule Base Reasoning</i>	24
4.3	Desain UML (<i>Unified Modelling Language</i>)	28
4.3.1	Use Case Diagram	28
4.3.2	Class Diagram	31
4.3.3	Activity Diagram	38
4.3.4	Sequence Diagram	46
4.3.5	Perancangan Antar Muka	52
4.2.5.1	Struktur Program	52
4.2.5.2	Perancangan Input	54
V.	IMPLEMENTASI DAN HASIL	57
5.1	Implementasi Sistem	57
5.1.1	Komponen Utama dalam Implementasi Sistem	57
5.1.2	Kebutuhan Server dalam Implementasi Sistem	57
5.2	Pengujian Sistem	58
5.3	Hasil Pengujian	65
VI.	PENUTUP	73
6.1	Kesimpulan	73
6.2	Saran	73
DAFTAR PUSTAKA		74

DAFTAR TABEL

Tabel 2.1 Simbol <i>Use Case Diagram</i>	12
Tabel 2.2 Simbol <i>Activity Diagram</i>	15
Tabel 2.3 Simbol <i>Sequence Diagram</i>	16
Tabel 4.1 Beban	25
Tabel 4.2 Kategori	25
Tabel 4.3 Kinerja	26
Tabel 4.4 Knowledge	28
Tabel 4.5 <i>Use Case Requirement</i>	29
Tabel 4.6 Asesor	33
Tabel 4.7 Dosen	33
Tabel 4.8 Tahun	34
Tabel 4.9 Pendidikan	34
Tabel 4.10 Penelitian	35
Tabel 4.11 Pengabdian	35
Tabel 4.12 User	36
Tabel 4.13 Beban	36
Tabel 4.14 Kategori	36
Tabel 4.15 Kinerja	37
Tabel 4.16 Database Knowledge	37
Tabel 4.17 Hasil	37
Tabel 5.1 Pengujian <i>Login User</i>	58
Tabel 5.2 Pengujian Data Asesor	59
Tabel 5.3 Pengujian Data Tahun	59
Tabel 5.4 Pengujian Data Dosen	60
Tabel 5.5 Pengujian Data Pendidikan	61
Tabel 5.6 Pengujian Data Penelitian.....	61
Tabel 5.7 Pengujian Data Pengabdian	62
Tabel 5.8 Pengujian Data Penunjang	63
Tabel 5.9 Pengujian Data Penilaian	64
Tabel 5.10 Pengujian Data <i>User</i>	64

DAFTAR GAMBAR

Gambar 2.1 PDO	11
Gambar 2.2 Aktor	13
Gambar 2.3 <i>Use Case</i>	13
Gambar 3.1 Struktur Solusi RBR	21
Gambar 3.2 Kerangka Kerja	22
Gambar 4.1 Arsitektur Sistem	24
Gambar 4.2 <i>Use Case Diagram</i> Administrator	30
Gambar 4.3 <i>Use Case Diagram</i> Dosen	31
Gambar 4.4 <i>Use Case Diagram</i> Asesor	31
Gambar 4.5 <i>Class Diagram</i>	32
Gambar 4.6 <i>Activity Diagram</i> pengolahan data Asesor	38
Gambar 4.7 <i>Activity Diagram</i> pengolahan data Dosen	39
Gambar 4.8 <i>Activity Diagram</i> pengolahan data Tahun	40
Gambar 4.9 <i>Activity Diagram</i> pengolahan data Pendidikan	41
Gambar 4.10 <i>Activity Diagram</i> pengolahan data Penelitian	42
Gambar 4.11 <i>Activity Diagram</i> pengolahan data Pengabdian	43
Gambar 4.12 <i>Activity Diagram</i> pengolahan data Penunjang	44
Gambar 4.13 <i>Activity Diagram</i> Penilaian	45
Gambar 4.14 <i>Sequence Diagram</i> pengolahan data Asesor	46
Gambar 4.15 <i>Sequence Diagram</i> pengolahan data Dosen	47
Gambar 4.16 <i>Sequence Diagram</i> pengolahan data Tahun	48
Gambar 4.17 <i>Sequence Diagram</i> data Pendidikan	49
Gambar 4.18 <i>Sequence Diagram</i> data Penelitian	49
Gambar 4.19 <i>Sequence Diagram</i> data Pengabdian	50
Gambar 4.20 <i>Sequence Diagram</i> data Penunjang	51
Gambar 4.21 <i>Sequence Diagram</i> data Penilaian	51
Gambar 4.22 Struktur Program Administrator	52
Gambar 4.23 Struktur Program Dosen	53
Gambar 4.24 Struktur Program Asesor	53
Gambar 4.25 <i>Form Login</i>	54
Gambar 4.26 <i>Form Input</i> Asesor	54
Gambar 4.27 <i>Form Input</i> Tahun	54
Gambar 4.28 <i>Form Input</i> Identitas Dosen	55
Gambar 4.29 <i>Form</i> Penilaian	56
Gambar 5.1 Halaman Utama	65
Gambar 5.2 Daftar <i>User</i>	66
Gambar 5.3 Halaman DASHBOARD	66
Gambar 5.4 Halaman Tahun	67
Gambar 5.5 Halaman Asesor	67
Gambar 5.6 Halaman Dosen	68

Gambar 5.7 Halaman Menu Knowledge	68
Gambar 5.8 Halaman Menu User	69
Gambar 5.9 Halaman Menu Data Diri Dosen	69
Gambar 5.10 Halaman Menu Kinerja Pendidikan Dosen	70
Gambar 5.11 Halaman Menu Kinerja Penelitian Dosen	70
Gambar 5.12 Halaman Menu Kinerja Pengabdian Dosen	71
Gambar 5.13 Halaman Menu Kinerja Penunjang Dosen	71
Gambar 5.14 Halaman Menu Penilaian Dosen	72

BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan Teknologi Informasi telah banyak mempengaruhi berbagai aspek kehidupan umat manusia dalam menjalankan aktivitas sehari-hari. Penggunaan komputer telah jauh mengalami kemajuan dari sekedar teknologi alat hitung hingga pengambilan keputusan. Komputer banyak dimanfaatkan dalam proses analisis, diagnosa dan sistem berbasis pengetahuan lainnya.

Remunerasi merupakan sebuah istilah yang berhubungan dengan penggajian pegawai yang ditetapkan dengan peraturan tertentu secara rutin berdasarkan nilai-nilai kerja, dengan tujuan terciptanya tata kelola yang lebih baik dan bersih serta meningkat motivasi dan prestasi kerja. Kinerja ditentukan dengan pengumpulan bukti kerja kepada pihak yang bertanggung jawab dan dihitung oleh unit yang telah ditentukan. *Rule Base Reasoning* adalah sistem pakar berdasarkan serangkaian aturan-aturan yang merupakan representasi dari pengetahuan dan pengalaman manusia dalam memecahkan kasus yang rumit. Satu aturan direpresentasikan dengan: *IF* <kondisi>*THEN*<kesimpulan>, di mana setiap kondisi-kondisi dari aturan ke aturan yang lainnya terhubung satu dengan yang lain melalui penghubung logika seperti penghubung dan, atau, negasi, serta penghubung lainnya membentuk sebuah fungsi logis. Implementasi *Rule Based Reasoning* akan dilakukan saat pencocokan Beban Kerja Dosen (BKD) yang telah dilakukan dengan rubrik yang disediakan oleh Universitas.

Aktivitas dosen tersebut akan dicocokkan dengan literatur berklasa IF, dan mendapatkan output binary yang nantinya akan dilihat kecocokannya antara Rubrik lainnya. Metode ini dipandang mampu untuk memilih hasil yang cocok untuk penilai menentukan jumlah remunerasi yang akan diterima oleh seorang pegawai. Sedangkan pengembangan sistem menggunakan konsep *Object Oriented Programming (OOP)* berbasis *Web*. *OOP* adalah suatu metode pemrograman yang berorientasi kepada objek. Tujuan dari penggunaan *OOP* adalah untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada di kehidupan sehari-hari. Dengan menggabungkan kedua konsep ini, proses pemberian remunerasi diharapkan berjalan baik dan terhindar dari salah perhitungan yang disebabkan *human error* yang dapat mengakibatkan kerugian di pihak institusi maupun pegawai.

1.2 Permasalahan

1.2.1 Identifikasi Permasalahan

1. Pada saat ini penilaian penerimaan remunerasi dosen biasa di UIN Sumatera Utara Medan berdasarkan kelebihan nilai Beban Kerja Dosen (BKD) yang dinilai oleh dua orang Asesor, yang terkadang penilaiannya tergantung dari objektivitas Asesor masing-masing.
2. Penginputan bukti Beban Kerja Dosen (BKD) oleh dosen masih belum terkomputerisasi dengan baik, karena masih offline menggunakan aplikasi Microsoft Office Access.

1.2.2 Batasan Permasalahan

Agar penelitian ini fokus pada judul penelitian yang akan dibahas, maka perlu dibuat batasan masalah. Adapun batasan masalah penelitian ini adalah :

1. Penggunaan *Rule Base Reasoning* dalam penentuan penerimaan remunerasi dosen biasa UIN Sumatera Utara Medan berdasarkan file Beban Kerja Dosen (BKD) yang telah diisi oleh dosen dan rubrik yang telah dikeluarkan oleh Universitas.
2. Merancang program penerimaan remunerasi menggunakan metodologi *Objek Oriented Programming* berbasis *Web*.

1.2.3 Perumusan Permasalahan

Berdasarkan latar belakang di atas maka dapat dirumuskan permasalahan penelitian sebagai berikut:

1. Bagaimanakah penggunaan *Rule Based Reasoning* dapat digunakan dalam perancangan sistem penerimaan remunerasi dosen biasa UIN Sumatera Utara Medan?
2. Bagaimanakah konsep *Object Oriented Programming* dapat digunakan pada Sistem Pakar *Rule Based Reasoning*?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah:

1. Mengetahui penggunaan *Rule Based Reasoning* dalam memecahkan masalah menggunakan basis *Knowledge*.
2. Merancang sistem penerimaan remunerasi dosen UIN Sumatera Utara Medan menggunakan metodologi *Object Oriented Programming* berbasis *Web*.

1.4 Signifikansi

Penelitian ini dianggap penting karena diharapkan akan memberi kontribusi sebagai berikut:

1. Tersedianya sistem terkomputerisasi yang dapat membantu pihak terkait dalam mengambil kebijakan jumlah remunerasi yang akan diterima oleh dosen biasa UIN Sumatera Utara Medan.
2. Meningkatnya akuntabilitas karena proses bisa dilihat langsung dari sistem yang telah dirancang dan mengurangi kesalahan dalam proses penentuan nilai remunerasi.

1.5 Sistematika Penulisan

Sistematika penulisan laporan penelitian ini adalah sebagai berikut :

BAB 1. PENDAHULUAN

Bab ini berisi tentang latar belakang, permasalahan yang terdiri dari identifikasi permasalahan, batasan permasalahan dan perumusan permasalahan. Kemudian tujuan penelitian, signifikansi penelitian dan sistematika penulisan.

BAB 2. LANDASAN TEORI

Bab ini berisi tentang teori yang berhubungan dengan perancangan sistem, sistem pakar *Rule Based Reasoning* dan konsep *Object Oriented* dalam pembangunan sistem.

BAB 3. METODOLOGI PENELITIAN

Bab ini berisi tentang metode yang digunakan dalam penelitian

BAB 4. PERANCANGAN SISTEM

Bab ini berisi tentang analisa dan perancangan system

BAB 5. IMPLEMENTASI SISTEM

Bab ini berisi tentang penerapan sistem dan penjelasannya.

BAB 6. PENUTUP

Bab ini berisi tentang kesimpulan dan saran-saran.

BAB II

LANDASAN TEORI

2.1 Perancangan Sistem Informasi

2.1.1 Perancangan

Perancangan adalah merancang atau mendesain suatu sistem yang baik, yang isinya adalah langkah-langkah dalam proses pengolahan data dan prosedur untuk mendukung operasi sistem. Menurut (Abdul Kadir, 2007), perancangan adalah proses penerapan berbagai teknik dan prinsip dengan tujuan untuk mentransformasikan hasil analisa kedalam bentuk yang mudah diimplementasikan.

2.1.2 Sistem

Istilah sistem berasal dari bahasa Yunani yaitu *Systema* yang mengandung arti kesatuan atau keseluruhan. Berarti sistem adalah sekumpulan objek-objek yang bekerja sama untuk menghasilkan satu kelompok, prosedur, teknik yang digunakan dan diatur sedemikian rupa satu kesatuan yang berfungsi untuk mencapai tujuan tertentu. Sistem merupakan salah satu yang terpenting dalam sebuah perusahaan yang dapat membentuk kegiatan usaha untuk mencapai kemajuan dan target yang dibutuhkan (Siti Fatima, 2013). Berikut ini adalah defenisi sistem menurut beberapa ahli:

1. Sistem merupakan suatu himpunan dari berbagai bagian atau elemen, yang saling berhubungan secara terorganisasi atau teratur berdasar fungsi-fungsinya menjadi satu kesatuan menuju tujuan tertentu. (Bambang Hartono, 2013:10).
2. Sistem adalah jaringan dari pada elemen-elemen yang saling berhubungan, membentuk satu kesatuan untuk melaksanakan suatu tujuan pokok dari sistem tersebut. (Jogiyanto, 2005:4).

Secara umum sistem adalah suatu kesatuan yang terdiri dari bagian-bagian yang saling berinteraksi dan bekerjasama untuk mencapai sasaran (*goal*). Dari defenisi di atas dapat diartikan sistem sebagai suatu proses berkelanjutan dari sekumpulan data yang saling berhubungan untuk tujuan bersama yang ingin dicapai serta sistem merupakan jaringan dari elemen-elemen yang saling berhubungan membentuk satu kesatuan untuk melaksanakan suatu tujuan pokok dari sistem tersebut.

2.1.3 Karakteristik Sistem

Model umum sebuah sistem terdiri dari *input*, proses dan *output*. Selain itu sebuah sistem juga memiliki karakteristik atau sifat-sifat tertentu,

yang mencirikan bahwa hal tersebut bisa dikatakan sebagai sistem. Adapun karakteristik sistem (Siti Fatima, 2013) adalah sebagai berikut:

1. **Komponen Sistem (*Components*)**
Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem.
2. **Batasan Sistem (*Boundary*)**
Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lainnya atau sistem dengan lingkungan luarnya.
3. **Lingkungan Luar Sistem (*Environment*)**
Lingkungan luar sistem adalah apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut.
4. **Penghubung Sistem (*Interface*)**
Penghubung sistem yaitu media yang menghubungkan antara satu subsistem dengan sub sistem yang lain.
5. **Masukan Sistem (*Input*)**
Masukan sistem merupakan energi yang dimasukkan ke dalam sistem.
6. **Keluaran Sistem (*Output*)**
Keluaran sistem merupakan hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna.
7. **Pengolah Sistem (*Proses*)**
Pengolah sistem merupakan suatu proses yang akan mengubah masukan menjadi keluaran.
8. **Sasaran Sistem (*Objective*)**
Suatu sistem memiliki tujuan dan sasaran yang pasti jika sistem tidak memiliki sasaran maka sistem tidak ada gunanya.

2.1.4 Informasi

Informasi berarti data yang telah dibentuk kedalam suatu format yang mempunyai arti dan guna bagi manusia. Sebaliknya, data merupakan sekumpulan baris fakta yang mewakili peristiwa yang terjadi pada organisasi atau pada lingkungan fisik sebelum diolah kedalam suatu format yang dapat dipahami dan digunakan orang. Jadi informasi adalah hasil dari pengolahan data kedalam bentuk yang lebih berguna dan lebih berarti yang membutuhkannya.

Informasi adalah hasil dari kegiatan pengolahan data yang memberikan bentuk yang lebih berarti dari suatu kejadian (Jogiyanto, 2005 : 3). Informasi adalah data yang telah diolah menjadi suatu bentuk yang berguna bagi penerimanya dan memiliki nilai bagi pengambilan keputusan disaat ini atau dimasa yang akan datang. (Bambang Hartono, 2013 : 15).

Sumber informasi adalah data. Data merupakan bentuk jamak dan bentuk tunggal atau data-data item. Data adalah kenyataan yang

menggambarkan suatu kejadian-kejadian (*event*) suatu yang terjadi pada saat tertentu.

2.1.5 Kualitas Informasi

Kualitas dari suatu informasi tergantung dari tiga hal (Siti Fatima, 2013) yaitu :

1. Akurat
berarti informasi harus bebas dari kesalahan-kesalahan dan tidak biasa atau menyesatkan. Informasi harus 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
Berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak ada nilainya lagi, karena informasi merupakan landasan dalam pengambilan keputusan.
3. Relevan
Informasi yang didapat mempunyai manfaat untuk pemakainya. Relevansi informasi untuk setiap orang berbeda.

2.1.6 Sistem Informasi

Sistem informasi adalah sekumpulan *hardware*, *software*, *brainware*, prosedur dan atau aturan yang diorganisasikan secara *integral* untuk mengolah data menjadi informasi yang bermanfaat guna memecahkan masalah dan pengambilan keputusan. Sistem informasi merupakan suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. (Tata Sutabri, 2005:42)

Sistem informasi adalah sekumpulan komponen yang bekerja sama secara terorganisasi dan terpadu dari sejumlah bagian atau komponen yang secara bersama-sama berfungsi dalam pengolahan data untuk memperoleh informasi, dengan maksud dan tujuan tertentu. (Bambang Hartono, 2013 : 20)

Sistem informasi mempunyai komponen-komponen yang disebut istilah blok yang saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sasaran (Siti Fatima, 2013) Komponen tersebut terdiri dari:

1. Blok Masukan (*Input Blok*)
Input blok ini mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode dan media mengambil data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model (*Model Blok*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang

akan memanipulasi data dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*Output Blok*)

Hasil dari sistem informasi adalah keluaran yang merupakan informasikan yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta pemakai sistem.

4. Blok Teknologi (*Technology Blok*)

Teknologi merupakan Kotak Alat (*Tool Box*) dalam sistem informasi karena digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirim keluaran dan membentuk pengendalian dari sistem secara keseluruhan.

5. Blok Basis Data (*Database Blok*)

Basis data merupakan kumpulan daridata yang saling berhubungan satu dengan yang lainnya, yang tersimpan diperangkat keras dan dimanipulasi oleh perangkat lunak komputer.

2.1.7 Perancangan Sistem

Desain sistem menentukan bagaimana suatu sistem akan menyelesaikan apa yang mesti diselesaikan tahap ini menyangkut konfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem, sehingga setelah instalasi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem. Desain sistem dapat didefinisikan sebagai penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi. Perancangan sistem membutuhkan suatu sistem yang akan dibentuk. Perancangan sistem mempunyai dua tujuan yaitu:

1. Untuk memenuhi kebutuhan pemakai sistem.
2. Untuk memberikan gambaran yang jelas rancang bangun yang lengkap kepada pemrograman komputer dan ahli-ahli teknik lainnya.

Untuk mencapai tujuan ini haruslah dapat mencapai sasaran-sasaran yaitu:

1. Perancangan sistem harus berguna, mudah dipahami dan nantinya mudah digunakan dan Perancangan harus dapat mendukung tujuan utama suatu perusahaan.
2. Perancangan sistem harus cepat dan tepat untuk dapat mendukung pengolahan transaksi.

2.2 Database

Database atau basis data adalah koleksi data yang bisa mencari secara menyeluruh dan secara sistematis memelihara informasi (Janer, 2007:2). Sedangkan menurut (Abdul Kadir 2014:218), basis data (*database*) adalah pengorganisasi sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi. Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas. Untuk mengelola database diperlukan perangkat lunak yang disebut *Database Management System*. DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol dan mengakses basis data dengan cara yang praktis dan efisien. DBMS dapat digunakan untuk mengakomodasikan berbagai macam pemakai yang memiliki kebutuhan akses yang berbeda-beda.

2.2.1 Komponen Database (Basis Data)

Apabila kita lihat komponen di dalam basis data, maka dapat disebutkan bahwa:

1. *Basis* data terdiri dari beberapa *file*.
2. *File* terdiri dari beberapa *record*.
3. *Record* terdiri dari *field*.
4. *Field* terdiri dari beberapa karakter.

Karakter merupakan bagian data yang terkecil dapat berupa karakter *numeric*, huruf maupun karakter khusus yang membentuk suatu data item (*field*). *Field* adalah kumpulan data-data *record* yang sejenis, yang merupakan kumpulan data untuk mewakili suatu *entity* data *record*. *Record* merupakan kumpulan dari *field* membentuk *record*.

Record menggambarkan unit dari data individu tertentu. *File* merupakan kumpulan dari *record-record* yang sejenis yang mempunyai panjang elemen yang sama, atribut yang sama tapi berbeda *valuenya*. *File* terdiri dari *record-record* yang menggambarkan suatu kesatuan data yang sejenis. Dengan menggunakan sistem basis data masalah pada manajemen basis data dapat berkurang.

Dengan sistem basis data juga dapat mengurangi duplikasi data dengan tujuan untuk mengurangi biaya manajemen. Basis data dapat dibuat cukup fleksibel dalam arti mudah ditambah atau dikurangi bahkan dimodifikasi, dan sistem basis data dapat menghubungkan dengan data lainnya.

2.3 Teknik Normalisasi

Proses normalisasi merupakan proses pengelompokan elemen data menjadi table-table yang menunjukkan entitas dan relasinya. Proses ini selalu diuji pada beberapa kondisi. Apakah ada kesulitan pada saat menambah

(*insert*), menghapus (*delete*), mengubah (*update*), atau membaca (*retrieve*) pada suatu database. Bila ada kesulitan pada pengujian tersebut maka relasi maka relasi dapat dipecah dalam beberapa tabel (Tata Sutabri, 2012:138).

2.3.1 Tujuan Normalisasi :

1. Untuk menghilangkan kerangkapan data
2. Untuk mengurangi kompleksitas
3. Untuk mempermudah pemodifikasian data

2.3.2 Proses Normalisasi

1. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat.
2. Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

2.3.3 Bentuk Normalisasi

Bentuk-bentuk normalisasi yang ada adalah seperti berikut ini :

1. Bentuk Tidak Normal, menghilangkan perulangan *group*.
2. Bentuk Normal Pertama (1NF), menghilangkan ketergantungan sebagian.
3. Bentuk Normal Kedua (2NF), menghilangkan ketergantungan transitif.
4. Bentuk Normal Ketiga (3NF), menghilangkan anomali-anomali hasil dari ketergantungan fungsional.
5. Bentuk Normal Boyce-Codd (BCNF), menghilangkan Ketergantungan *Multivalued*.
6. Bentuk Normal Keempat (4NF), menghilangkan anomali-anomali yang tersisa.
7. Bentuk Normal Kelima (5NF).

2.4 Sistem Pakar

Sistem pakar adalah suatu sistem yang berusaha mengadopsi pengetahuan manusia ke dalam komputer agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para pakar. Sistem pakar menurut (Turban, 1995). Disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan sistem pakar digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar, sedangkan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar guna memperoleh pengetahuan pakar.

Menurut (Arhami, Muhamad, 2004), Pengembangan sistem pakar sendiri akan dikembangkan lebih lanjut dengan alasan, sebagai berikut:

1. Pengetahuan dari seorang pakar mahal harganya.
2. Seorang pakar suatu ketika akan pensiun dari pekerjaannya bahkan bisa meninggal dunia sehingga pengetahuan dari pakar akan hilang dan tidak dapat diturunkan ke pakar yang lebih junior.
3. Secara otomatis dapat mengerjakan tugas-tugas rutin yang membutuhkan seorang pakar.
4. Kepakaran juga dibutuhkan setiap waktu dan diberbagai lokasi bahkan pada lokasi yang kurang atau tidak mendukung

2.4.1 Komponen Sistem Pakar

Menurut (Kusrini, 2008). Komponen utama pada sistem pakar meliputi:

1. Basis Pengetahuan (*Knowledge Base*)
Basis pengetahuan merupakan inti dari suatu sistem pakar, yaitu berupa representasi pengetahuan dari pakar
2. Mesin Inferensi (*Inference Engine*)
Mesin inferensi berperan sebagai otak dari sistem pakar. Di dalam mesin inferensi terjadi proses untuk memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan dalam rangka mencapai solusi atau kesimpulan. Terdapat tiga teknik pengendalian yang sering digunakan, yaitu *forward chaining*, *backward chaining*, dan gabungan dari kedua tehnik pengendalian tersebut.
3. Basis Data (*Data Base*)
4. Basis data terdiri atas semua fakta yang diperlukan, di mana fakta-fakta tersebut digunakan untuk memenuhi kondisi dari kaidah-kaidah dalam sistem.

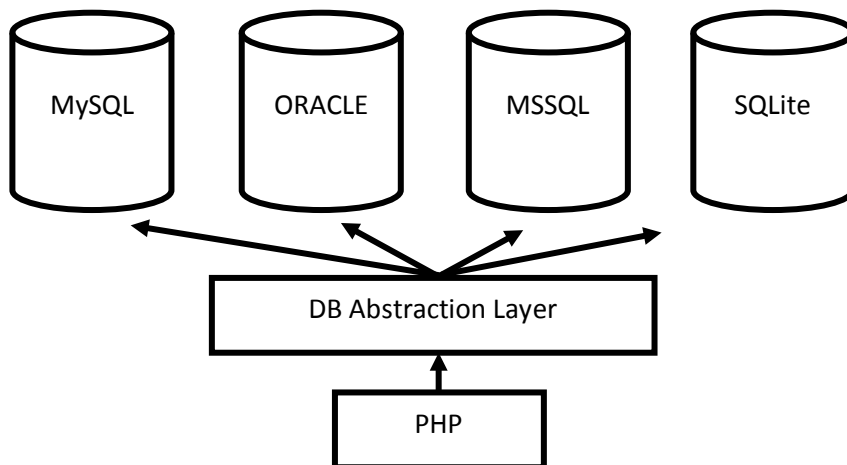
2.4.2 Rule Based Reasoning

Rule Based Reasoning (RBR) merupakan aturan-aturan logis dimana setiap aturannya didapat dari studi literatur dan informasi dari ahli tanpa melihat kasus yang dihadapi. Selain itu ada beberapa cara alternatif untuk memperoleh aturan tersebut menggunakan metode pembelajaran mesin berdasarkan data empiris yang ada. Satu aturan direpresentasikan dengan: IF THEN , di mana setiap kondisi-kondisi dari aturan ke aturan yang lainnya terhubung satu dengan yang lain melalui penghubung logika seperti penghubung dan, atau, negasi, serta penghubung lainnya membentuk sebuah fungsi logis. (Abdurrachman, 2015).

2.5 PHP Data Object

PDO adalah extension baru untuk PHP 5.0 ke atas untuk melakukan manajemen database. PDO memberikan juga menyertakan sekumpulan driver untuk dapat bekerja pada berbagai perangkat lunak database yang berbeda. PDO dikembangkan agar dapat memberikan interface yang ringan untuk perangkat database yang berbeda. Keunggulan lainnya yaitu dapat memberikan penanganan error yang lebih baik, serta dapat mengeksekusi multiple query lebih cepat. Cara kerja PDO sama seperti Data Access Layer dimana dapat digunakan nama fungsi yang sama untuk semua perangkat database (Welling, 2008).

Setiap basis data mempunyai cara tersendiri didalam melakukan pengaksesan datanya, perbedaan ini menyebabkan terlalu sulit untuk berpindah dari suatu database ke database yang lainnya. Akan tetapi terkadang kita membutuhkan beberapa tipe database untuk keperluan tertentu pada suatu aplikasi yang sama. Misalnya perusahaan anda secara resmi menggunakan MySQL, kemudian terdapat beberapa aplikasi yang secara eksklusif harus dijalan di ORACLE. Maka apakah perusahaan anda akan beralih menggunakan ORACLE dengan mengeluarkan anggaran yang besar? Dan harus membangun kembali sistem yang sudah ada dengan database yang baru? Untuk mengatasi masalah ini mulailah para developer mengembangkan abstraksi layer untuk melayani aplikasi didalam berkomunikasi dengan berbagai macam sistem basisdata.



Gambar 2.1. PDO

2.6 Unified Modeling Language (UML)





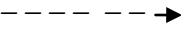
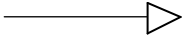
Unified Modeling Language (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan

software berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema *database*, dan komponen-komponen yang diperlukan dalam sistem *software* (<http://www.omg.org>). Diagram *Unified Modelling Language* (UML) (Siti Fatima, 2013) antara lain sebagai berikut:

2.6.1 Use Case Diagram

Use Case adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. *Use Case* digunakan untuk membentuk tingkah-laku benda atau *things* dalam sebuah model serta direalisasikan oleh sebuah *collaboration*. *Use Case* memiliki beberapa simbol seperti pada tabel 2.6 dibawah ini.

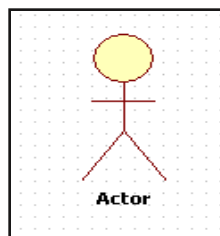
Tabel 2.1 Simbol Use Case Diagram

Simbol	Keterangan	Fungsi
	<i>Package</i>	Menambahkan paket baru dalam diagram.
	<i>Use Case</i>	Menambahkan <i>use case</i> dalam diagram.
	<i>Actors</i>	Menambahkan <i>actor</i> dalam diagram.
	<i>Unidirectional Association</i>	Menggambarkan relasi antara <i>actor</i> dengan <i>use case</i> .
	<i>Dependencies or Instantiates</i>	Menggambarkan kebergantungan (<i>dependencies</i>) antar <i>item</i> dalam diagram.
	<i>Generalization</i>	Menggambarkan relasi lanjut antar <i>use case</i> atau menggambarkan struktur pewarisan antar <i>actor</i> .

1. Actor

Pada dasarnya *actor* bukanlah bagian dari *use case diagram*, namun untuk dapat terciptanya suatu *use case diagram* diperlukan beberapa *actor*.

Actor tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah *actor* mungkin hanya memberikan informasi masukan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. *Actor* hanya berinteraksi dengan *use case*, tetapi tidak memiliki kontrol atas *use case*. *Actor* digambarkan dengan *stick man*. *Actor* dapat digambarkan secara umum atau spesifik, dimana untuk membedakannya kita dapat menggunakan *relationship*.

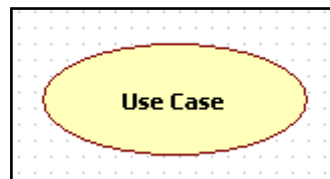


Gambar 2.2. Aktor

2. Use Case

Use case adalah gambaran fungsionalitas dari suatu sistem, sehingga *customer* atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun. *Use case diagram* adalah penggambaran sistem dari sudut pandang pengguna sistem tersebut (*user*), sehingga pembuatan *use case* lebih dititikberatkan pada fungsionalitas yang ada pada sistem, bukan berdasarkan alur atau urutan kejadian. Cara menentukan *Use Case* dalam suatu sistem:

- Pola perilaku perangkat lunak aplikasi.
- Gambaran tugas dari sebuah *actor*.
- Sistem atau “benda” yang memberikan sesuatu yang bernilai kepada *actor*.
- Apa yang dikerjakan oleh suatu perangkat lunak (bukan bagaimana cara mengerjakannya).



Gambar 2.3. Use Case

3. Relasi dalam *Use Case*

Ada beberapa relasi yang terdapat pada *use case diagram*:

- a. *Association*, menghubungkan *link* antar elemen.
- b. *Generalization*, disebut juga *inheritance* (pewarisan), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
- c. *Dependency*, sebuah elemen bergantung dalam beberapa cara ke elemen lainnya.
- d. *Aggregation*, bentuk *association* dimana sebuah elemen berisi elemen lainnya.

Tipe relasi (*stereotype*) yang mungkin terjadi pada *use case diagram*:

- a. <<include>>, yaitu kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya.
- b. <<extends>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.
- c. <<communicates>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah *communicates association*. Ini merupakan pilihan selama asosiasi hanya tipe *relationship* yang dibolehkan antara *actor* dan *use case*.

2.6.2 *Class Diagram*

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* memiliki tiga area pokok yaitu :

1. Nama, merupakan nama dari sebuah kelas.
2. Atribut, merupakan peroperti dari sebuah kelas. Atribut melambangkan batas nilai yang mungkin ada pada obyek dari *class*.
3. Operasi, adalah sesuatu yang bisa dilakukan oleh sebuah *class* atau yang dapat dilakukan oleh *class* lain terhadap sebuah *class*.

2.6.3 *Statechart diagram*

Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya) suatu objek pada sistem sebagai akibat dari stimulus yang diterima. Pada umumnya *statechart diagram* menggambarkan *class* tertentu (satu *class* dapat memiliki lebih dari satu *statechart diagram*).





Dalam UML, *state* digambarkan berbentuk segi empat dengan sudut membulat dan memiliki nama sesuai kondisinya saat itu. Transisi antar *state* umumnya memiliki kondisi *guard* yang merupakan syarat terjadinya transisi yang bersangkutan, dituliskan dalam kurung siku. *Action* yang dilakukan sebagai akibat dari event tertentu dituliskan dengan diawali garis miring. Titik awal dan akhir digambarkan berbentuk lingkaran berwarna penuh dan berwarna setengah.


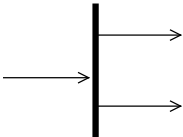
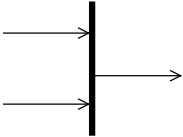

2.6.4 Activity Diagram

Activity Diagram merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *Activity Diagram* tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum.

Activity Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision yang* mungkin terjadi, dan bagaimana mereka berakhir. *Activity Diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity Diagram* memiliki beberapa simbol seperti pada table 2.2 dibawah ini.

Tabel 2.2 Simbol Activity Diagram


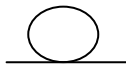
Simbol	Keterangan	Fungsi
	<i>State</i>	Menambahkan <i>state</i> untuk suatu objek
	<i>Activity</i>	Menambahkan aktivitas baru pada diagram
	<i>Start Point</i>	Memperlihatkan dimana aliran kerja Berawal
	<i>End Point</i>	Memperlihatkan dimana aliran kerja berakhir

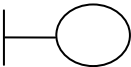



	<i>State transition</i>	Menambah transisi dari suatu aktivitas ke aktivitas yang lainnya
	<i>Fork</i> (Percabangan)	Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel
	<i>Join</i> (Penggabungan)	Digunakan untuk menunjukkan kegiatan yang digabungkan
	<i>Decision</i>	Menambahkan titik keputusan pada aliran kerja

2.6.5 Sequence Diagram

Sequence diagram (diagram urutan) adalah suatu diagram yang memperlihatkan atau menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, *display*, dan sebagainya berupa pesan (*message*). *Sequence Diagram* digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai sebuah respon dari suatu kejadian (*event*) untuk menghasilkan *output* tertentu. *Sequence diagram* memiliki beberapa simbol seperti pada tabel 2.3 dibawah ini.

Tabel 2.3 Simbol *Sequence Diagram*

Simbol	Keterangan	Fungsi
	<i>Actors</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem
	<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang dilakukan

	<i>Boundary Class</i>	Menggambarkan sebuah dari <i>form</i>
	<i>Control Class</i>	Menggambarkan penghubung antara <i>boundary</i> dengan tabel
	<i>A focus of control and A life line</i>	Menggambarkan tempat mulai dan berakhirnya <i>message</i>
	<i>A message</i>	Menggambarkan pengiriman pesan

2.6.6 Collaboration Diagram

Collaboration Diagram adalah perluasan dari objek dan diagram (objek diagram menunjukkan objek-objek dan hubungannya satu dengan yang lain). *Collaboration Diagram* menunjukkan *message-message* objek yang dikirim satu sama lain.

2.6.7 Component Diagram

Component Diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) diantaranya. Komponen piranti lunak adalah modul berisi *code*, baik berisi *source code* maupun *binary code*, baik *library* maupun *executable*, baik yang muncul pada *compile time*, *link time*, maupun *run time*. Umumnya komponen terbentuk dari beberapa *class* dan atau *package*, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

2.6.8 Package Diagram

Adalah sebuah bentuk pengelompokan yang memungkinkan untuk mengambil sebuah bentuk di UML dan mengelompokkan elemen-elemennya dalam tingkatan unit yang lebih tinggi. Kegunaan *package* yang paling umum adalah untuk mengelompokkan *class*.

2.6.9 Deployment Diagram

Deployment atau *physical diagram* menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, dimana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan

jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisik. Sebuah *node* adalah *server*, *workstation* atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya.

2.7 Perangkat Lunak Yang Digunakan

2.7.1 Bahasa Pemrograman PHP

PHP pertama kali dibuat oleh Rasmus Ledorf pada tahun 1995, pada waktu itu PHP bernama FI9 (*Form Interpreted*). PHP adalah sekumpulan script yang digunakan untuk mengelola Data form dari web. perkembangan selanjutnya adalah Rasmus melepaskan kode sumber tersebut kepanjangan dari PHP/FI adalah personal homepage/from interpreter. Dengan pelepasan kode sumber ini menjadi opensource, maka banyak programmer pada rilis ini interpreter sudah diimplementasikan dalam C.

Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan *PHP/FI* secara signifikan. Pada tahun 1997, sebuah perusahaan bernama Zend, menulis ulang *interpreter PHP* menjadi lebih bersih, lebih baik dan lebih cepat. Kemudian pada juni 1998 perusahaan tersebut merilis *interpreter* baru untuk *PHP* dan meresmikan nama rilis tersebut menjadi *PHP* 3.0. Pada pertengahan tahun 1999, Zend, merilis *interpreter PHP* baru dan rilis tersebut dikenal dengan *PHP* 4.0. adalah versi *PHP* yang paling banyak dipakai. Versi ini banyak dipakai sebab versi ini mampu dipakai untuk membangun aplikasi *web* kompleks tetapi memiliki kecepatan proses dan stabilitas yang tinggi. Pada juni 2004, Zend merilis *PHP* 5.0 Versi ini adalah versi mutakhir dari *PHP* dalam versi ini, dari *interpreter PHP* mengalami perubahan besar. Dalam versi ini juga di kenalkan model pemrograman berorientasi objek baru untuk menjawab perkembangan bahasa pemrograman berorientasi objek (Siti Fatima, 2013).

2.7.2 MySQL

MySQL adalah *Relational Database Management System* (RDBMS) yang di distribusikan secara gratis dibawah lisensi GPL (General Publik License). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu *system database* (DBMS) dapat diketahui dari kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh *user* maupun program-program aplikasinya. Sebagai *database server*, MySQL dapat dikatakan lebih unggul dibandingkan

Database server lainnya dalam *Query* data. Hal ini terbukti untuk *Query* yang dilakukan *Single User*, kecepatan *Query MySQL* bisa sepuluh kali lebih cepat dari *Postgres SQL* dan lima kali lebih cepat dibandingkan *Interbase*.

PhpMyAdmin merupakan bagian untuk mengelola basis data *MySQL* yang ada dikomputer. *phpMyAdmin* juga merupakan *MySQL Client* berbasis web, atau program yang dapat digunakan untuk mengakses database *MySQL* melalui browser. Sebelum memulai halaman *MySQL* pastikan *webserver (XAMPP)* sudah dijalankan. Untuk membukanya, buka *browser* lalu ketikkan alamat *http://localhost/phpMyAdmin*, maka akan muncul halaman *phpMyAdmin*.

Bagian kiri halaman berisi daftar database bawaan server *MySQL*, Sedangkan pada bagian kanan berisi menu dan *tools* untuk melakukan operasional manajemen database. Lewat fasilitas yang disediakan *phpMyAdmin* inilah bisa membuat database baru, membuat tabel baru, mengganti dan menghapus tabel juga *database*. Selain itu semua operasi manajemen data juga bisa dilakukan, yaitu memasukan data, menampilkan dan menghapus data (Siti Fatima, 2013).

2.7.3 Web Browser

Web Browser merupakan software yang berfungsi untuk menampilkan web di internet, web browser sangat penting karena tanpa web browser kita tidak dapat membuka halaman website internet. Web browser yang digunakan harus mendukung software yang digunakan dalam mendesain web, yang tergolong kedalam web browser seperti internet explorer, mozilla, firefox, opera, google chrome dan lain-lain.

Dalam hal ini penulis menggunakan *mozilla firefox*. *Browser* ini mendukung berbagai fasilitas yang memuat halaman *web* menjadi lebih hidup. Dan *mozilla firefox* menjadi *browser* yang paling banyak digunakan untuk membuka halaman *web*, dibandingkan *web browser* lainnya. *Mozilla firefox* (aslinya bernama *Phoenix* dan kemudian untuk sesaat dikenal sebagai *Mozilla Firebird*) adalah peramban *web* lintas *platform* gratis yang dikembangkan oleh yayasan *Mozilla* dan ratusan sukarelawan.

Sejak April 2003, *firefox* dan *client surel Thunderbird* telah menjadi fokus utama pengembangan yayasan *mozilla* untuk menggantikan *mozilla suite*. *Firefox* telah mendapatkan perhatian sebagai alternatif kepada *internet explorer* dikecam karena tuduhan ketidaksamaanya pihak yang setuju terhadap anggapan ini mengatakan *explorer* tidak mengikuti standar *web*, menggunakan komponen *ActiveX* yang sering membahayakan, dan kelemahannya terhadap pemasangan *software* dan *hardware* dan kurangnya *fitur-fitur* yang dianggap pemakai *firefox* penting (firefox.com)

BAB III METODOLOGI PENELITIAN

Pada bab ini akan diuraikan metodologi penelitian dan kerangka kerja penelitian yang digunakan dalam penyelesaian penelitian ini. Kerangka kerja ini merupakan langkah-langkah yang akan dilakukan dalam rangka penyelesaian masalah yang akan dibahas.

3.1 Pengumpulan Data

Pada tahapan ini akan dilakukan pengumpulan data yang menyangkut dengan sistem yang akan dibangun. Adapun cara atau metode yang bisa dilakukan dalam pengumpulan data ini antara lain :

a. Wawancara

Pengumpulan data dilakukan melalui proses wawancara kepada pihak yang berhubungan langsung dengan proses penerimaan remunerasi dosen di Universitas Islam Negeri Sumatera Medan yaitu Lembaga Penjaminan Mutu (LPM), Satuan Pengawas Intern (SPI) serta Pusat Teknologi Informasi dan Pangkalan Data (PUSTIPADA)

b. Studi Pustaka

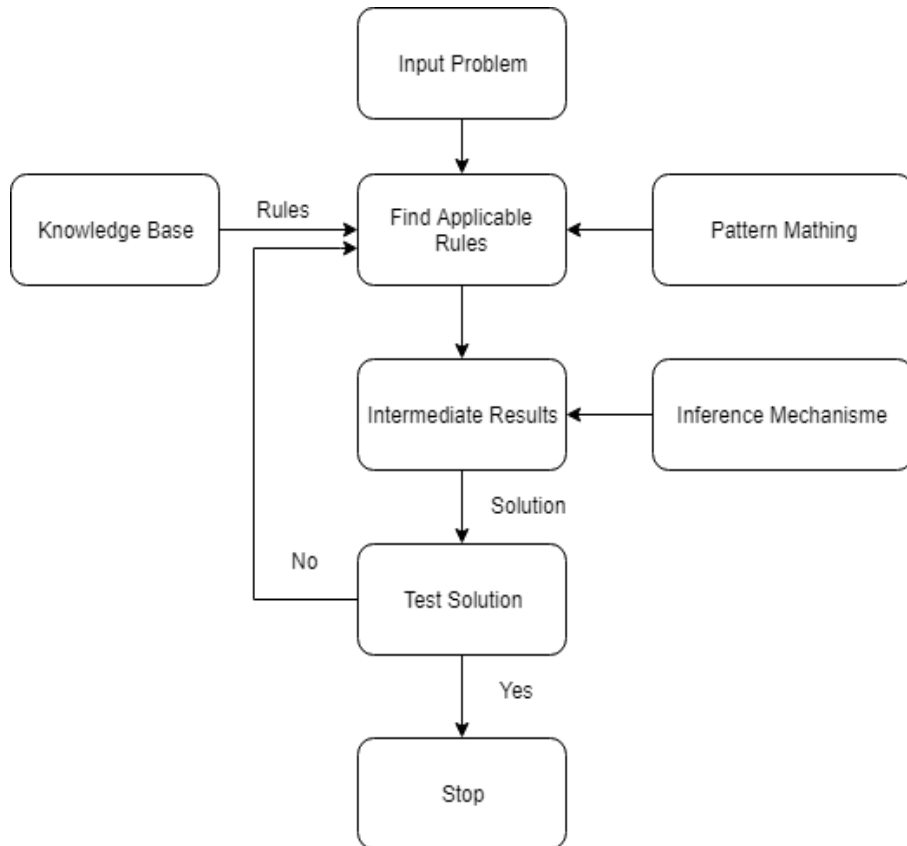
Studi pustaka dilakukan untuk mendapatkan teori yang mendukung penelitian yang sedang dilakukan. Adapun hal yang dapat dipelajari dari studi pustaka yaitu definisi dari *Rule Based Reasoning* dan cara penggunaan dalam sistem. Studi pustaka didapatkan melalui buku, jurnal, artikel, maupun berbagai referensi yang diperbolehkan untuk digunakan dalam menyelesaikan penelitian ini. Studi ini juga difokuskan pada Rubrik Beban Kerja Dosen yang telah dibukukan oleh Universitas Islam Negeri Sumatera Utara Medan.

3.2 Metode *Rule Based Reasoning*

Rule Based Reasoning (RBR) merupakan aturan-aturan logis dimana setiap aturannya didapat dari studi literatur dan informasi dari ahli tanpa melihat kasus yang dihadapi. Selain itu ada beberapa cara alternatif untuk memperoleh aturan tersebut menggunakan metode pembelajaran mesin berdasarkan data empiris yang ada. Satu aturan direpresentasikan dengan: IF THEN, dimana setiap kondisi-kondisi dari aturan ke aturan yang lainnya terhubung satu dengan yang lain melalui penghubung logika seperti penghubung dan, atau, negasi, serta penghubung lainnya membentuk sebuah fungsi logis (Maiyulis, 2018).

Rule Based Expert System adalah *advanced computer* program yang mencoba untuk meniru kemampuan manusia dalam membuat keputusan dan pemecahan masalah (Ceccaroni, 2009). Ide awal pengembangan metode *Rule Based* adalah dengan membentuk sekumpulan *rules* (aturan) pada basis

pengetahuan. Kemudian untuk memperoleh informasi baru, digunakan *inference engine* (Turban, 1995). Bentuk dari penggunaan *Rule Based* adalah biasanya dengan menggunakan model IF – THEN, sehingga dihasilkan aturan-aturan statis.



Gambar 3.1 Struktur Solusi RBR

3.3 Kerangka Kerja

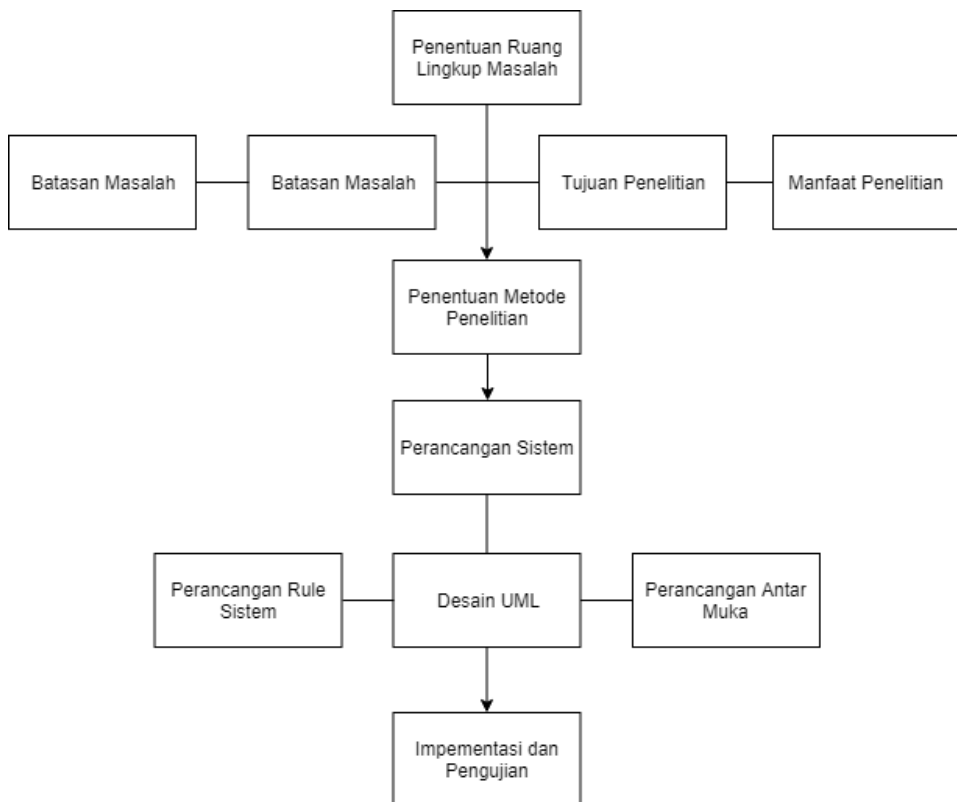
Dalam usaha mendapatkan hasil yang optimal dari penelitian ini maka dianggap penting untuk membuat suatu kerangka kerja dalam melakukan penelitian sebagai tolak ukur pencapaian tujuan penelitian. Setiap proses yang dilakukan dalam penyusunan laporan penelitian juga berdasarkan kerangka kerja yang telah dibuat, sehingga lebih tersusun rapi dan sistematis.

Kerangka kerja dari penelitian ini dideskripsikan dengan langkah-langkah sebagai berikut :

1. Penentuan ruang lingkup permasalahan
 - a. Identifikasi Permasalahan
 - b. Batasan Permasalahan

- c. Perumusan Permasalahan
- d. Tujuan penelitian
- e. Signifikansi penelitian
- 2. Penentuan metode penelitian
- 3. Perancangan sistem
 - a. Perancangan *Rule* Sistem
 - b. Desain UML
 - c. Perancangan Antar Muka
- 4. Implementasi dan Pengujian sistem

Langkah-langkah Deskripsi di atas digambarkan dengan diagram kerangka kerja penelitian pada gambar 3.2



Gambar 3.2 Kerangka Kerja

- a. Ruang lingkup permasalahan
Langkah yang dilakukan untuk mengidentifikasi permasalahan adalah dengan melakukan pengumpulan data dari objek penelitian dan membandingkan dengan kebutuhan-kebutuhan

remunerasi. Setelah mengumpulkan data maka akan didapatkan perumusan permasalahan yang terjadi pada saat ini, kemudian

b. Penentuan Metode Penelitian

Setelah dilakukan perumusan masalah pada tahap sebelumnya, maka selanjutnya dilakukan penentuan metode penelitian yang akan dibuat.

c. Perancangan system

Tahap perancangan dan pembangun sistem merupakan tahap perancangan dan penerapan dari sistem Rule Based Reasoning. Tahap ini dimulai dari perancangan *rule*, dilanjutkan dengan desain UML dan perancangan antar muka.

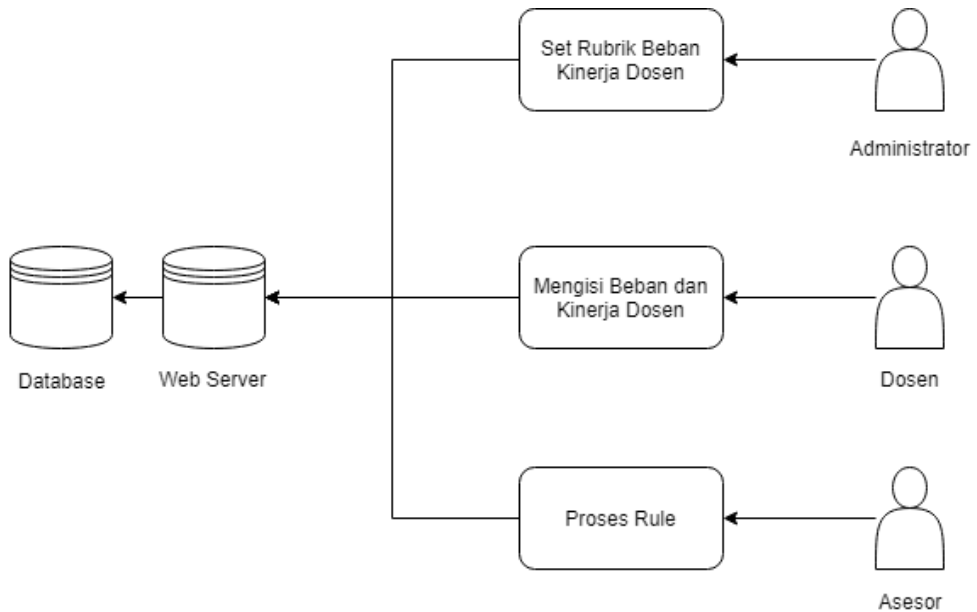
d. Implementasi sistem dan pengujian

Pada tahap implementasi sistem ini akan dijelaskan mengenai penggunaan dari sistem informasi remunerasi yang telah dirancang. Penjelasan ini mencakup tampilan sistem, fungsi dan kontrol aplikasi dan serta bagian-bagian yang dirancang. Sedangkan pada tahap pengujian sistem tahap ujicoba, apakah semua antar muka yang dirancang sudah berfungsi dengan baik.

BAB IV PERANCANGAN SISTEM

4.1 Arsitektur Sistem

Untuk memahami konsep dari sistem penerimaan remunerasi dosen yang akan dibangun, maka pada gambar 4.1 berikut ini digambarkan arsitektur sistem.



Gambar 4.1 Arsitektur Sistem

Pada gambar 4.1 Administrator bertugas melakukan pengisian Rubrik Beban Kerja Dosen. Sedangkan Dosen bertugas Mengisi Beban Dan Kerja masing-masing. Setiap data yang masuk akan diseleksi oleh Asesor. Seluruh data tersebut diproses oleh *web server* dan disimpan kedalam *database*.

4.2 Rule Based Reasoning

Untuk menerapkan metode *Rule Base Reasoning* pada sistem yang dirancang, maka perlu dibuat beberapa tabel yang mendukung metode ini berdasarkan data yang diambil dari Rubrik Beban Kerja Dosen yang diterbitkan oleh Universitas Islam Negeri Sumatera Utara Medan. Adapun tabel yang terkait adalah sebagai berikut.

1. Tabel Beban

Tabel beban digunakan untuk menyimpan data beban kerja yang diwajibkan untuk semua dosen. Masing-masing data beban diberikan

kode yang diperlukan dalam metode *Rule Base Reasoning*. Adapun tabel beban seperti yang terlihat pada tabel 4.1 berikut ini:

Tabel 4.1 Beban

Kode	Keterangan
B1	Pendidikan
B2	Penelitian
B3	Pengabdian
B4	Penunjang

2. Tabel Kategori

Tabel kategori untuk menyimpan kategori kinerja yang dinyatakan termasuk dalam lingkup kinerja yang dibayarkan pada remunerasi. Masing-masing data kategori diberikan kode yang diperlukan dalam metode *Rule Base Reasoning* Adapun tabel kategori seperti yang terlihat pada tabel 4.2 berikut ini:

Tabel 4.2 Kategori

Kode	Kategori
K1	Kuliah tingkat S0-S1
K2	Kuliah tingkat S2-S3
K3	Mengembangkan Bahan Kuliah
K4	Bimbingan Praktikum/Magang terhadap setiap kelompok
K5	Bimbingan kuliah kerja yang terprogram terhadap setiap kelompok PKL/PPL/KKN
K6	Seminar mahasiswa berkelompok, terstruktur dan terjadwal disertai bimbingan oleh dosen, bukan sebagai bagian dari kuliah/praktikum. Yang dimaksud adalah: seminar proposal dan Mata Kuliah seminar, seminar hasil penelitian hibah mahasiswa, seminar Tugas Akhir Mahasiswa.
K7	Membaca/ Membimbing Proposal/Tugas Akhir, 1) S1 (Skripsi), 2) S2 (Tesis), 3) Membaca/ Membimbing Proposal Tugas Akhir S3 (Disertasi)
K8	Menguji Seminar Proposal: 1) S1 (Skripsi), 2) S2 (Tesis), 3) S3 (Disertasi)
K9	Seminar Hasil Disertasi
K10	Bimbingan Tugas Akhir Mahasiswa: 1) S1 (Skripsi), 2) S2 (Tesis), 3) S3 (Disertasi) Dosen Pembimbing utama dan pembimbing penyerta dinilai sama

3. Tabel Kinerja

Tabel Kinerja digunakan untuk menyimpan kategori kinerja yang dilakukan Dosen, tabel kinerja dinilai sesuai dengan rubrik yang dikeluarkan Universitas untuk menghitung BKD Dosen. Masing-masing data Kinerja diberikan kode yang diperlukan dalam metode *Rule Base Reasoning* Adapun tabel Kinerja seperti yang terlihat pada tabel 4.3 berikut ini:

Tabel 4.3 Kinerja

Kode	Kegiatan	Bukti	SKS x
R1	Mengajar 1-40 Mahasiswa selama 1 semester	<ul style="list-style-type: none"> • SK Pimpinan • BAP • Presensi • Nilai 	1
R2	Mengajar 41-80 Mahasiswa selama 1 semester	<ul style="list-style-type: none"> • SK Pimpinan • BAP • Presensi • Nilai 	1,5
R3	Mengajar 81-120 Mahasiswa selama 1 semester	<ul style="list-style-type: none"> • SK Pimpinan • BAP • Presensi • Nilai 	2
R4	Mengajar 1-15 Mahasiswa selama 1 semester	<ul style="list-style-type: none"> • SK Pimpinan • BAP • Presensi • Nilai 	1
R5	Mengajar 16-30 Mahasiswa selama 1 semester	<ul style="list-style-type: none"> • SK Pimpinan • BAP • Presensi • Nilai 	1,5
R6	Menulis 1 judul buku/ bahan ajar utuh yang relevan dengan keahlian/SK Dosen	<ul style="list-style-type: none"> • SK/ Surat Tugas • Proses • Penulisan (Draf Buku dan Surat Keterangan Kerjasama/ Kontrak Penerbit) • Buku/Diktat/Mod ul Jadi (Cover, Halaman Deskripsi, 	5

		<ul style="list-style-type: none"> • Halaman ISBN [khusus buku ajar] dan Daftar isi) atau <i>Link on-line (repository UINSU, academia.edu, google scholar, researchgate)</i> 	
R7	Diktat /Modul/Petunjuk Praktikum/Model/Alat Bantu/ Audio Visual/ Naskah Tutorial	<ul style="list-style-type: none"> • Petunjuk Praktikum, Model, Alat Bantu, Audio Visual, Naskah Tutoria dan Surat Keterangan dari Pimpinan • Link on-line (<i>repository UINSU, academia.edu, google scholar, researchgate</i>). 	2
R8	Bimbingan 1-25 orang mahasiswa selama 1 semester, 2 jam perminggu (minimal 12 kali tatap muka)	<ul style="list-style-type: none"> • SK/ Surat Tugas dari Pimpinan • Presensi Mahasiswa • BAP/Berita Acara Praktikum • Nilai praktikum 	1
R9	1-25 orang mahasiswa, kegiatan yang setara dengan 50 jam kerja per semester sama dengan 1 sks 6 hari termasuk untuk persiapan, pelaksanaan dan pelaporan	<ul style="list-style-type: none"> • SK/ Surat Tugas dari pimpinan. • Presensi Mahasiswa. • Nilai PKL/ PPL/KKN. 	1
R10	Bimbingan Tugas Akhir Mahasiswa: 1) S1 (Skripsi), 2) S2 (Tesis), 3) S3 (Disertasi) Dosen Pembimbing utama dan pembimbing penyerta dinilai sama	<ul style="list-style-type: none"> • SK/ Surat Tugas dari pimpinan. • Bukti Pembimbingan. 	1

4. Tabel Knowledge

Tabel knowledge untuk menyimpan *rule knowledge* yang merupakan bagian penting dari metode *Rule Base Reasoning*. Adapun tabel *knowledge* seperti yang terlihat pada tabel 4.4 berikut ini:

Tabel 4.4 Knowledge

No	Kategori	Kinerja	Beban
1	K1	R1	B1
2	K1	R2	B1
3	K1	R3	B1
4	K2	R4	B1
5	K2	R5	B1
6	K3	R6	B1
7	K3	R7	B1
8	K4	R8	B1
9	K4	R9	B1
10	K4	R10	B1

IF (Kategori = K1) AND (Kinerja = R1) THEN Beben = B1

4.3 Desain UML (*Unified Modelling Language*)

Unified Modelling Language (UML) adalah sebuah bahasa untuk menentukan, visualisasi, konstruksi, dan mendokumentasikan *artifact* (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak. *Artifact* dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. Desain *Unified Modelling Language* (UML) dari sistem ini tampak seperti pada penjelasan berikut ini. UML mengizinkan pengembang untuk mengembangkan berbagai tipe *visual diagram* yang mempersentasikan berbagai sudut pandang sistem.

4.3.1 Use Case Diagram

Use Case merupakan perilaku *software* aplikasi dimana proses tersebut menggambarkan suatu sistem, sehingga yang menggunakan sistem akan mudah mengerti mengenai kegunaan sistem yang dibangun. *Use Case Diagram* adalah gambaran (*graphical*) dari beberapa *actor*, *Use Case*, dan interaksi diantaranya yang memperkenalkan suatu sistem. *Use Case Diagram* menggambarkan siapa saja aktor yang melakukan prosedur dalam sistem serta fungsi-fungsi (proses) yang terlibat dalam transformasi pada sistem tersebut.

Adapun *use case diagram* yang diusulkan pada penelitian ini adalah sebagai berikut :

1. *Actor*

Actor yang terdapat pada sistem yang diusulkan ini adalah :

- a. Administrator, *actor* ini bisa melakukan pengolahan data rubrik, yaitu penambahan dan perubahan data rubrik. Administrator juga bisa menambahkan dan melakukan perubahan terhadap data dosen.
- b. Dosen, *actor* ini bisa melakukan penambahan dan perubahan serta menghapus data beban dan kinerja dosen setiap semester.
- c. Asesor, *actor* ini hanya bisa melakukan proses

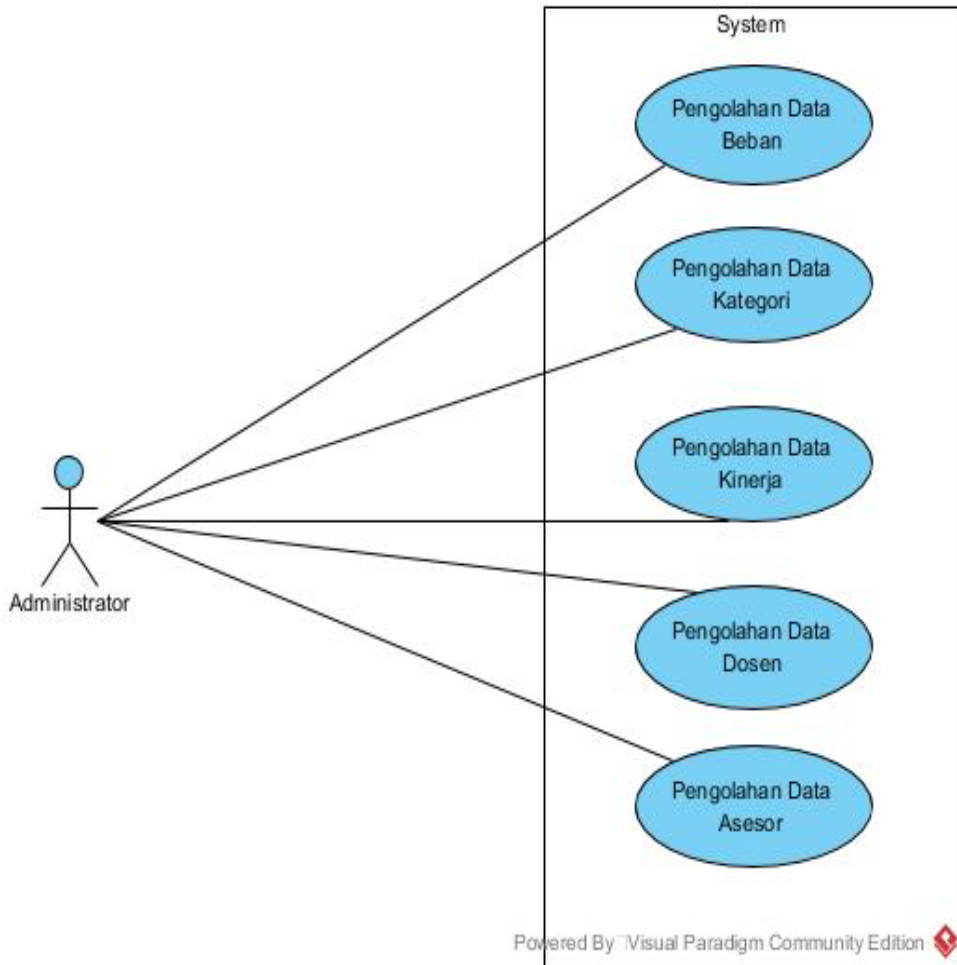
2. Use Case Requirement

Setelah mengidentifikasi *actor*, tahap selanjutnya adalah menentukan kebutuhan fungsi *use case* yang dibutuhkan oleh sistem dalam interaksinya dengan *actor-actor* tersebut. Berikut adalah identifikasi kebutuhan *use case* pada sistem yang diusulkan ini adalah:

Tabel 4.5 Use Case Requirement

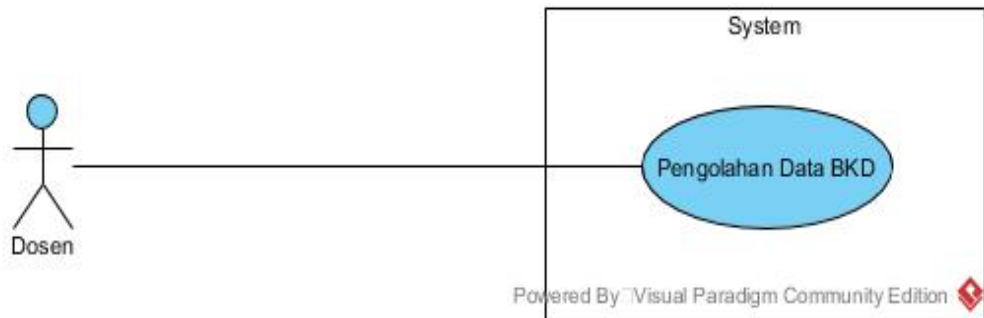
No	Requirement	Actor	Use Case
1	Administrator dapat menambahkan dan merubah Data Dosen	Administrator	Pengolahan data Dosen
2	Administrator dapat menambahkan dan merubah data Asesor	Administrator	Pengolahan data Asesor
3	Administrator dapat menambahkan dan merubah data Beban	Administrator	Pengolahan data Beban
4	Administrator dapat menambahkan dan merubah data Kategori	Administrator	Pengolahan data Kategori
5	Administrator dapat menambahkan dan merubah data Kinerja	Administrator	Pengolahan data Kinerja
6	Dosen bisa menambahkan dan merubah Data Dosen	Dosen	Pengolahan data Dosen
7	Dosen bisa menambahkan dan merubah BKD (Pendidikan, Penelitian, Pengabdian dan Penunjang)	Dosen	Pengolahan data BKD (Pendidikan, Penelitian, Pengabdian dan Penunjang)
8	Asesor bisa memproses BKD (Pendidikan, Penelitian, Pengabdian dan Penunjang)	Asesor	Pengolahan data BKD (Pendidikan, Penelitian, Pengabdian dan Penunjang)

Berikut ini merupakan *Use Case* diagram yang menggambarkan proses yang dilakukan oleh Dosen. Dosen berhak untuk mengolah Debitur dan Data Pinjaman. Selain tersebut hanya administrator yang berhak untuk melakukan pengolahan data, baik menambahkan maupun memperbaiki data yang sudah ada.



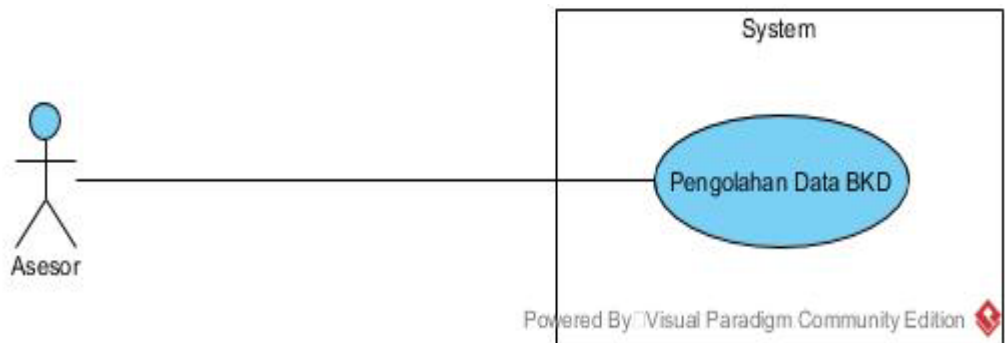
Gambar 4.2 Use Case Diagram Administrator

Kemudian berikut ini *use case* diagram yang menggambarkan proses yang dilakukan oleh Dosen.



Gambar 4.3 Use Case Diagram Dosen

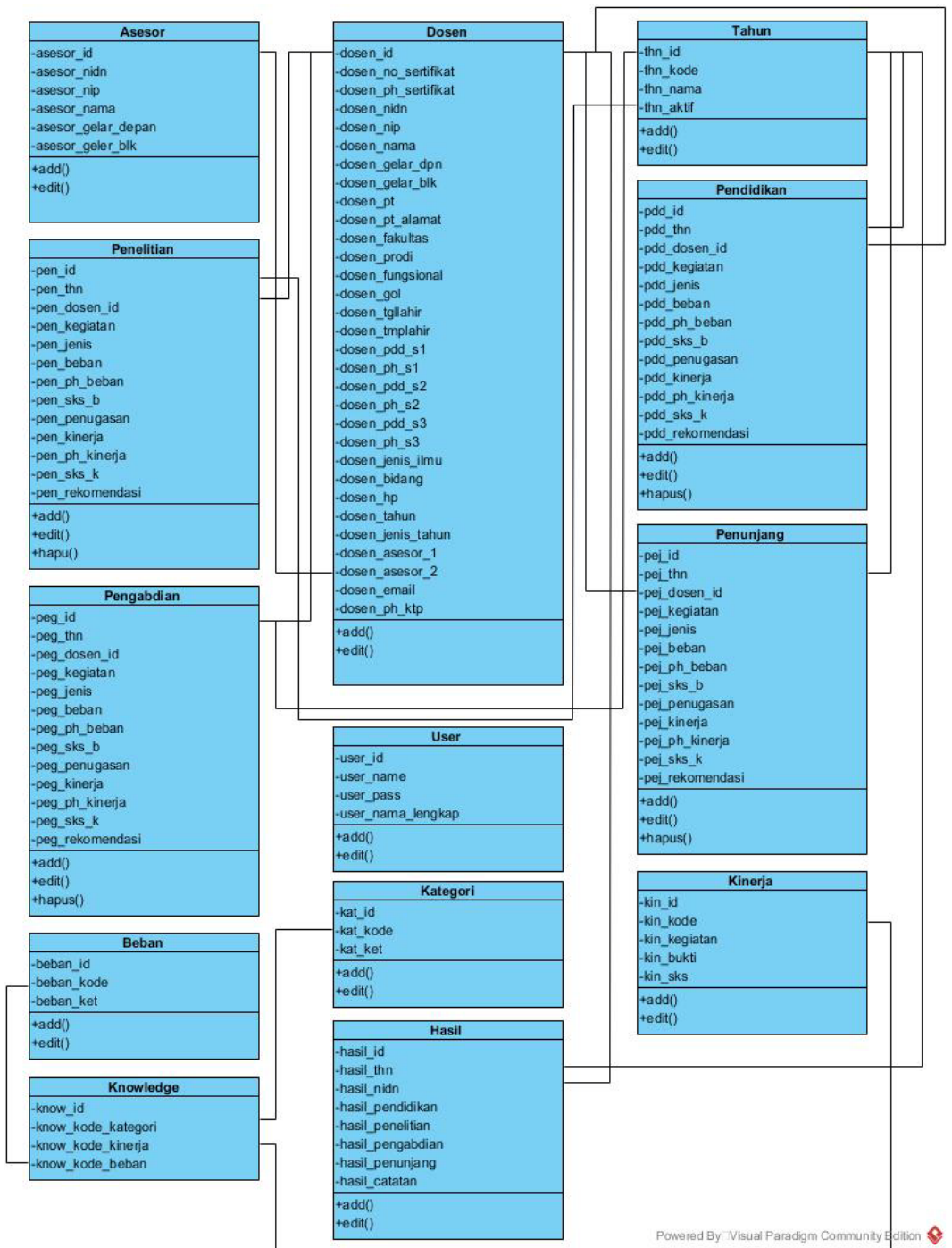
Adapun user berhak untuk melakukan penilaian terhadap BKD yang ditambahkan oleh Dosen.



Gambar 4.4 Use Case Diagram Asesor

4.3.2 Class Diagram

Class diagram menampilkan eksistensi atau keberadaan dari *class-class* dan hubungan (*relationship*) dalam desain logikal dari sebuah sistem. Semua proses yang dilakukan oleh aktor terhadap aplikasi akan didefinisikan dengan menggunakan *class diagram*. *Class diagram* merupakan diagram yang akan memperlihatkan dan menjelaskan tabel-tabel pada *database* dan relasi antar tabel yang akan digunakan didalam sistem ini.



Gambar 4.5 Class Diagram

Keterangan dari *Class Diagram* :

1. Tabel Asesor

Tabel Asesor berfungsi untuk menyimpan data asesor. Berikut merupakan daftar *field* dan *type* table asesor.

Tabel 4.6 Asesor

No	Field	Type
1	asesor_id	INT(11)
2	asesor_nip	VARCHAR(25)
4	asesor_nidn	VARCHAR(10)
5	asesor_nama	VARCHAR(100)
6	asesor_gelar_depan	VARCHAR(20)
7	asesor_gelar_blk	VARCHAR(20)

2. Tabel Dosen

Tabel Dosen berfungsi untuk menyimpan data penerima dosen. Berikut daftar *field* dan *type* tabel dosen.

Tabel 4.7 Dosen

No	Field	Type
1	dosen_id	INT(11)
2	dosen_no_sertifikat	VARCHAR(100)
3	dosen_ph_sertifikat	VARCHAR(100)
4	dosen_nidn	VARCHAR(10)
6	dosen_nip	VARCHAR(25)
7	dosen_nama	VARCHAR(100)
8	dosen_gelar_depan	VARCHAR(20)
9	dosen_gelar_blk	VARCHAR(20)
10	dosen_pt	VARCHAR(100)
11	dosen_pt_alamat	VARCHAR(100)
12	dosen_fakultas	VARCHAR(50)
13	dosen_prodi	VARCHAR(50)
14	dosen_fungsional	VARCHAR(25)
15	dosen_gol	VARCHAR(10)
16	dosen_tglahir	DATE
17	dosen_tmplahir	VARCHAR(100)
18	dosen_pdd_s1	VARCHAR(100)
19	dosen_ph_s1	INT(11)
20	dosen_pdd_s2	VARCHAR(100)
21	dosen_ph_s2	INT(11)

22	dosen_pdd_s3	VARCHAR(10)
23	dosen_ph_s3	INT(11)
24	dosen_jenis_ilmu	VARCHAR(100)
25	dosen_bidang	VARCHAR(100)
26	dosen_hp	VARCHAR(25)
27	dosen_tahun	TINYINT(3)
28	dosen_jenis_tahun	TINYINT(3)
29	dosen_asesor_1	VARCHAR(25)
30	dosen_asesor_2	VARCHAR(25)
31	dosen_email	VARCHAR(50)
32	dosen_ph_ktp	INT(11)

3. Tabel Tahun

Tabel tahun berfungsi untuk menyimpan data tahun ajaran. Berikut daftar *field* dan *type* tabel tahun.

Tabel 4.8 Tahun

No	Field	Type
1	tahun_id	TINYINT(3)
2	tahun_kode	VARCHAR(5)
3	tahun_nama	VARCHAR(100)
4	tahun_aktif	ENUM('Y','T')

4. Tabel Pendidikan

Tabel pendidikan berfungsi untuk menyimpan data pendidikan yang diinput oleh dosen. Berikut daftar tabel *field* dan *type* tabel pendidikan.

Tabel 4.9 Pendidikan

No	Field	Type
1	pdd_id	INT(11)
2	pdd_thn	TINYINT(3)
3	pdd_dosen_id	VARCHAR(25)
4	pdd_kegiatan	VARCHAR(100)
5	pdd_jenis	INT(11)
6	pdd_beban	VARCHAR(100)
7	pdd_ph_beban	INT(11)
8	pdd_sks_b	INT(11)
9	pdd_penugasan	VARCHAR(50)

10	pdd_ph_kinerja	VARCHAR(100)
11	pdd_sks_k	INT(11)
12	pdd_rekomendasi	VARCHAR(100)

5. Tabel penelitian

Tabel penelitian berfungsi untuk menyimpan data penelitian yang diinput oleh dosen. Berikut daftar *field* dan *type* tabel penelitian.

Tabel 4.10 Penelitian

No	Field	Type
1	pen_id	INT(11)
2	pen_thn	TINYINT(3)
3	pen_dosen_id	VARCHAR(25)
4	pen_kegiatan	VARCHAR(100)
5	pen_jenis	INT(11)
6	pen_beban	VARCHAR(100)
7	pen_ph_beban	INT(11)
8	pen_sks_b	INT(11)
9	pen_penugasan	VARCHAR(50)
10	pen_ph_kinerja	VARCHAR(100)
11	pen_sks_k	INT(11)
12	pen_rekomendasi	VARCHAR(100)

6. Tabel Pengabdian

Tabel pengabdian berfungsi untuk menyimpan data pengabdian yang diinput oleh dosen. Berikut daftar tabel *field* dan *type* tabel pengabdian.

Tabel 4.11 Pengabdian

No	Field	Type
1	peg_id	INT(11)
2	peg_thn	TINYINT(3)
3	peg_dosen_id	VARCHAR(25)
4	peg_kegiatan	VARCHAR(100)
5	peg_jenis	INT(11)
6	peg_beban	VARCHAR(100)
7	peg_ph_beban	INT(11)
8	peg_sks_b	INT(11)
9	peg_penugasan	VARCHAR(50)

10	peg_ph_kinerja	VARCHAR(100)
11	peg_sks_k	INT(11)
12	peg_rekomendasi	VARCHAR(100)

7. Tabel User

Tabel user berfungsi untuk menyimpan data seluruh pengguna sistem. Berikut daftar *field* dan *type* tabel user.

Tabel 4.12 User

No	Field	Type
1	user_id	INT(11)
2	user_user	VARCHAR(25)
3	user_pass	VARCHAR(100)
4	user_nama_lengkap	VARCHAR(100)

8. Tabel Beban

Tabel beban berfungsi untuk menyimpan data beban. Berikut daftar *field* dan *type* tabel beban.

Tabel 4.13 Beban

No	Field	Type
1	beban_id	TINYINT(3)
2	beban_kode	VARCHAR(10)
3	beban_ket	VARCHAR(100)

9. Tabel Kategori

Tabel kategori berfungsi untuk menyimpan data kategori. Berikut daftar *field* dan *type* tabel kategori.

Tabel 4.14 Kategori

No	Field	Type
1	kat_id	INT(11)
2	kat_kode	VARCHAR(10)
3	kat_ket	VARCHAR(254)

10. Tabel Kinerja

Tabel kinerja berfungsi untuk menyimpan data kinerja. Berikut daftar *field* dan *type* tabel kinerja.

Tabel 4.15 Kinerja

No	Field	Type
1	kin_id	INT(11)
2	kin_kode	VARCHAR(10)
3	kin_kegiatan	VARCHAR(254)
4	kin_bukti	VARCHAR(100)
5	kin_sks	INT(11)

11. Knowledge

Tabel database knowledge untuk menyimpan *rule* knowledge yang merupakan bagian penting dari metode *Rule Base Reasoning*

Tabel 4.16 Database Knowledge

No	Field	Type
1	know_id	INT(11)
2	know_kode_kategori	VARCHAR(10)
3	know_kinerja	VARCHAR(10)
4	know_beban	VARCHAR(10)

12. Tabel hasil

Tabel hasil berfungsi untuk menyimpan data hasil. Berikut daftar *field* dan *type* tabel hasil.

Tabel 4.17 Hasil

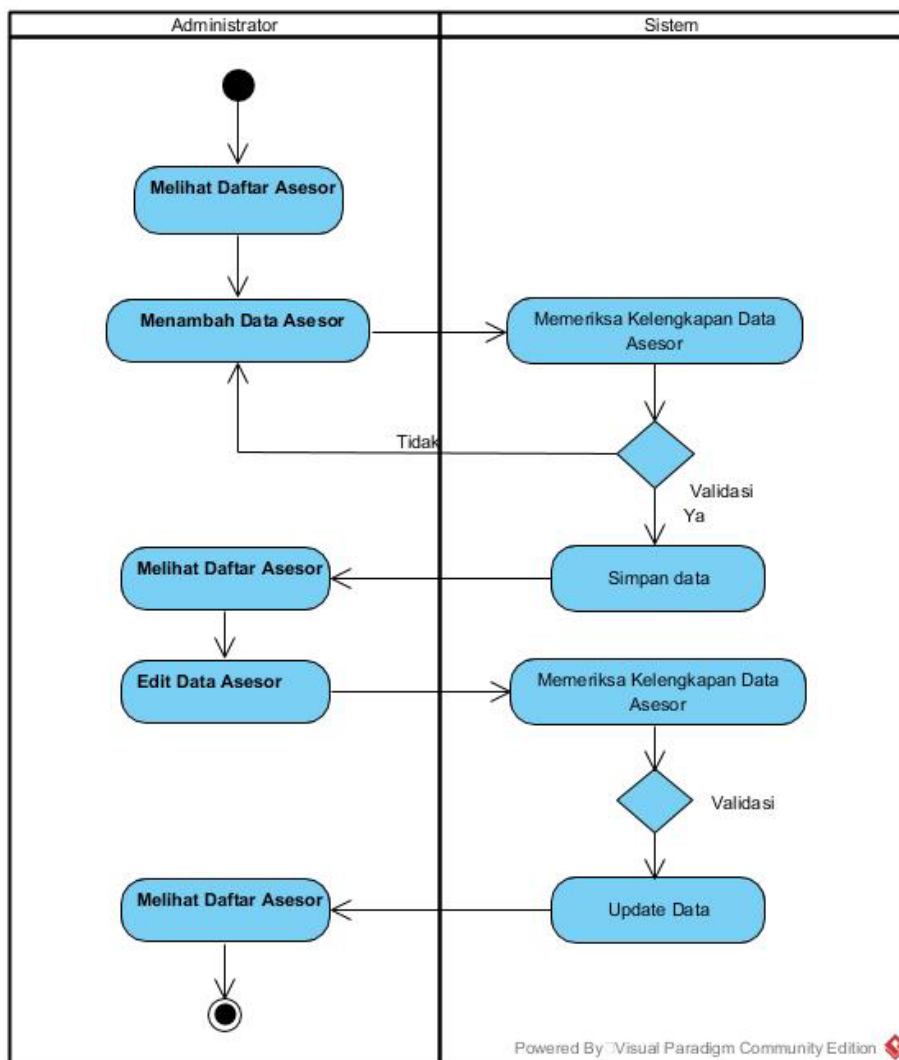
No	Field	Type
1	hasil_id	INT(11)
2	hasil_thn	TINYINT(3)
3	hasil_nidn	VARCHAR(25)
4	hasil_pendidikan	INT(11)
5	hasil_penelitian	INT(11)
6	hasil_pengabdian	INT(11)
7	hasil_penunjang	INT(11)
8	hasil_catatan	VARCHAR(254)

4.3.3 Activity Diagram

Activity diagram merupakan diagram untuk menggambarkan logika prosedur, proses dan jalur kerja yang terjadi dalam sistem yang akan dirancang. Berikut *activity diagram* dari sistem informasi yang dirancang.

1. Activity Diagram pengolahan Asesor

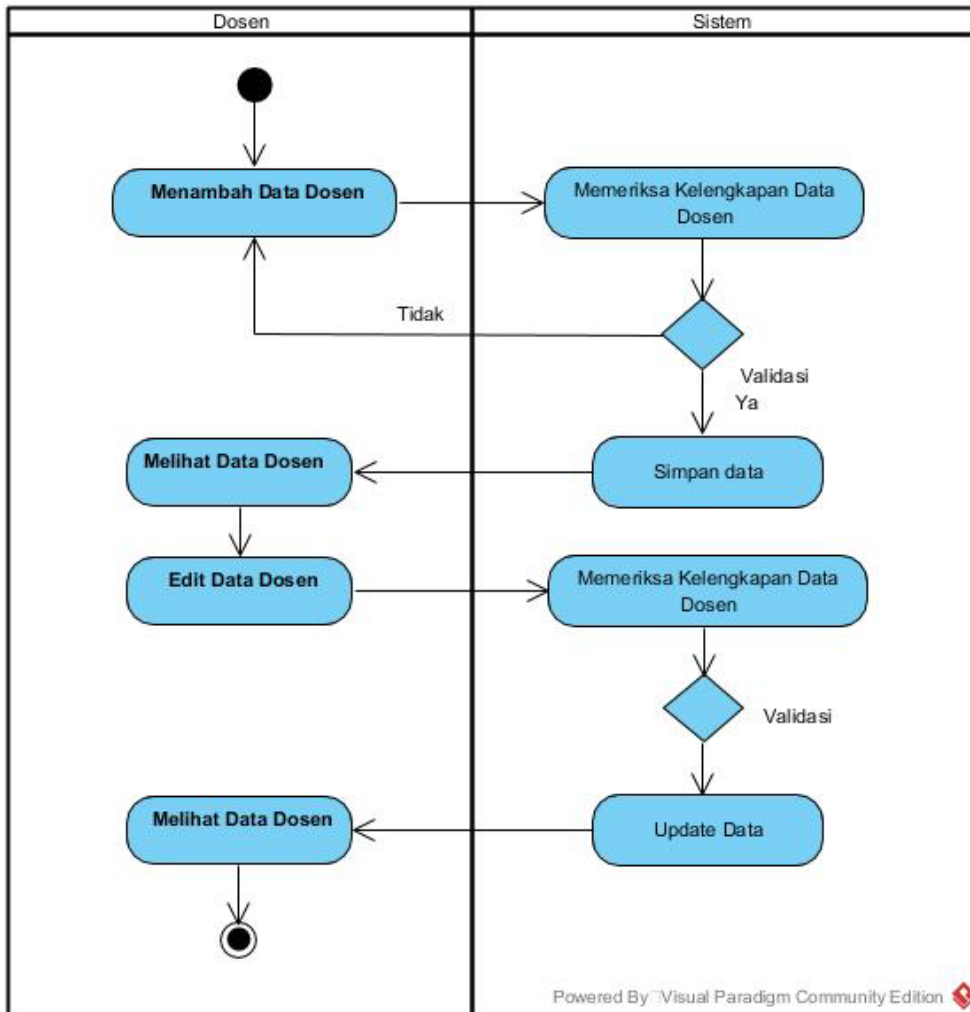
Pengolahan data Asesor dilakukan oleh administrator, dimana administrator bisa melakukan penambahan data asesor yang dilanjutkan dengan pemeriksaan kelengkapan data oleh sistem terhadap data yang ditambahkan. Berikut gambar *Activity Diagram* pengolahan data Asesor.



Gambar 4.6 Activity Diagram pengolahan data Asesor

2. Activity Diagram pengolahan Dosen

Pengolahan data dosen dilakukan oleh masing-masing Dosen, dimana Dosen bisa melakukan penambahan data setiap tahun akademik yang dilanjutkan dengan pemeriksaan kelengkapan data oleh sistem terhadap data yang ditambahkan apabila diperlukan perubahan data, maka dosen juga bisa melakukan perubahan yang dilanjutkan dengan validasi kelengkapan data oleh sistem. Berikut gambar *Activity Diagram* pengolahan data Dosen.

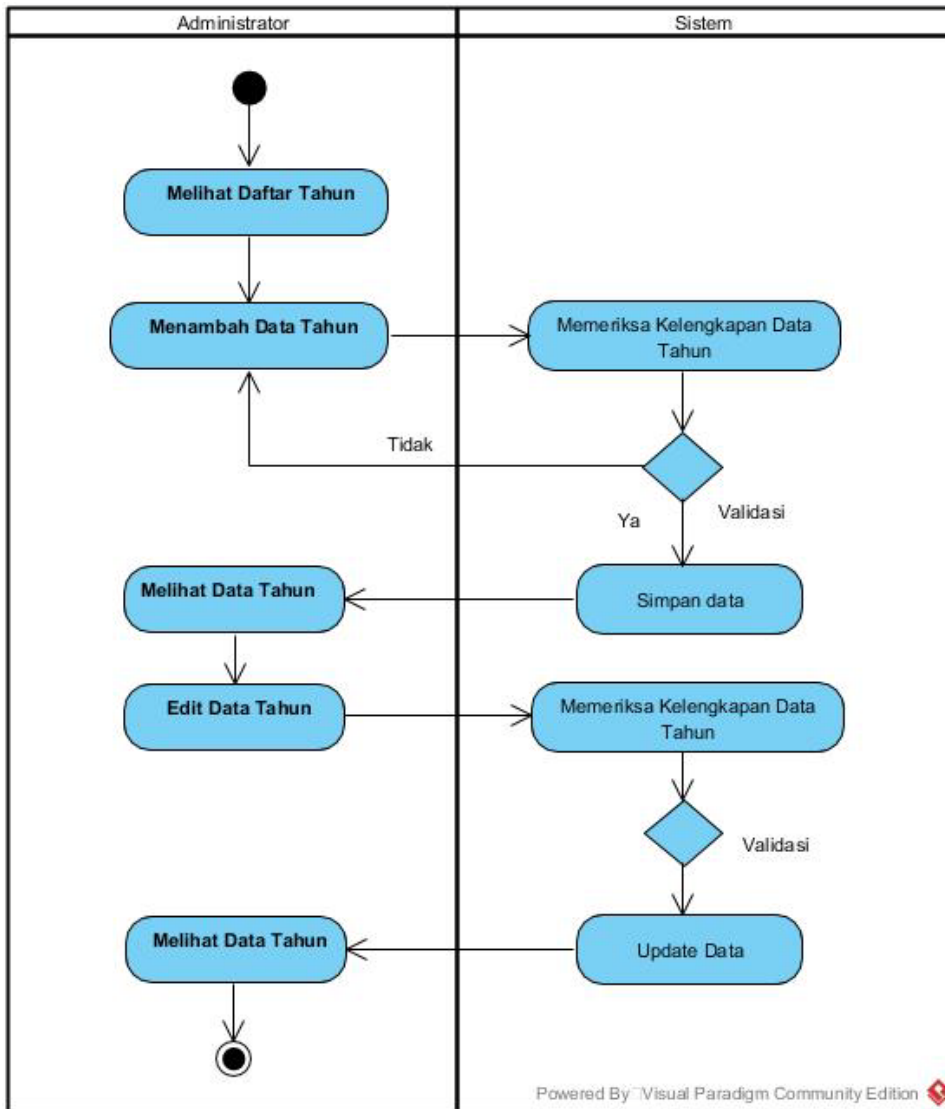


Gambar 4.7 Activity Diagram pengolahan data Dosen

3. Activity Diagram pengolahan data Tahun

Pengolahan data Tahun Akademik dilakukan oleh Administrator, dimana administrator bisa melakukan penambahan data Tahun Akademik yang dilanjutkan dengan pemeriksaan kelengkapan data oleh sistem terhadap

data yang ditambahkan serta administrator juga bisa melakukan perubahan terhadap data yang dimasukkan. Berikut gambar *Activity Diagram* pengolahan data tahun.

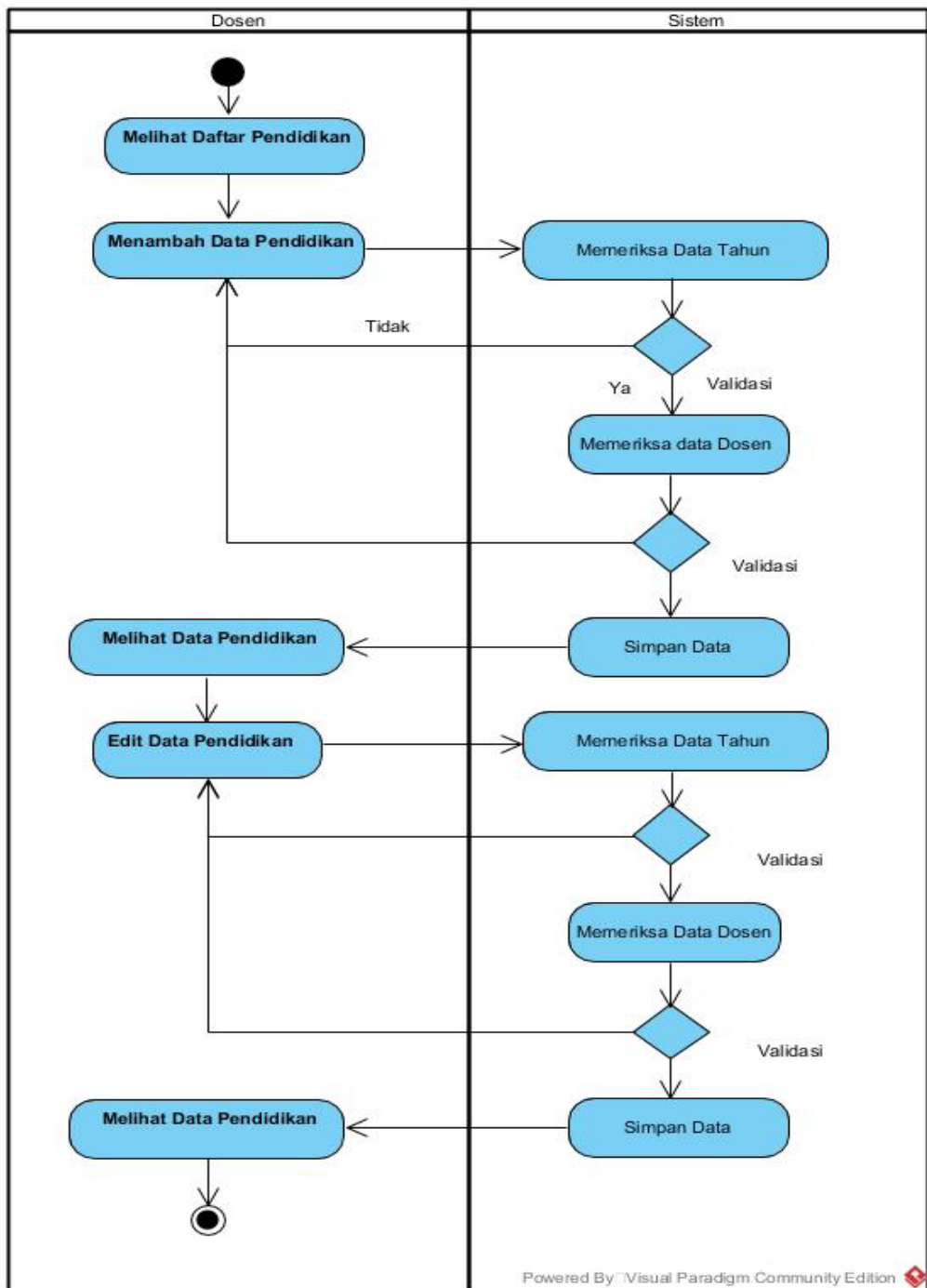


Gambar 4.8 Activity Diagram pengolahan data Tahun

4. Activity Diagram pengolahan data Pendidikan

Pengolahan data Pendidikan dilakukan oleh masing-masing Dosen, dimana Dosen bisa melakukan penambahan data Pendidikan yang dilanjutkan dengan pemeriksaan kelengkapan data oleh sistem terhadap data yang ditambahkan. Pada data Pendidikan berelasi dengan Tahun Akademik dan

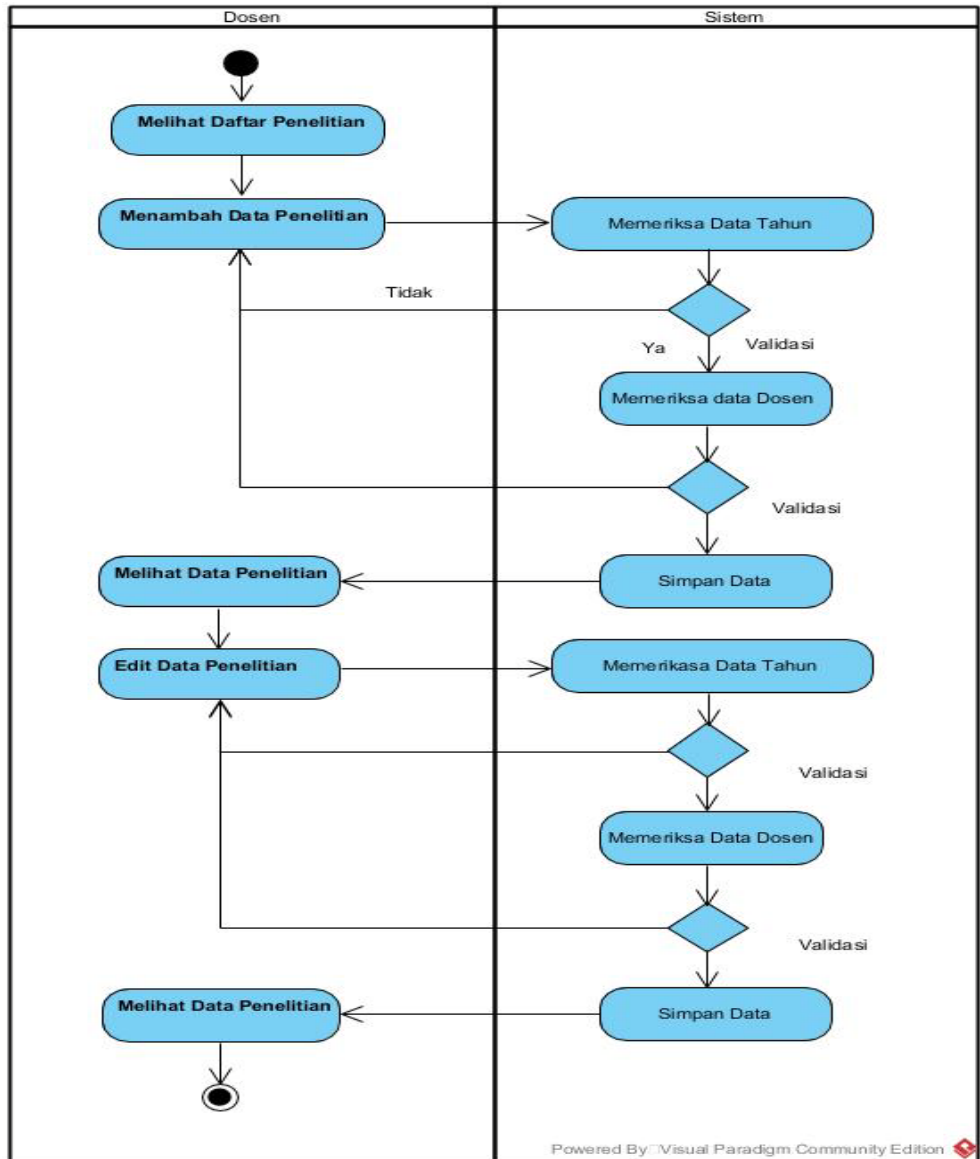
data Dosen. Dosen juga bisa melakukan perubahan terhadap data yang telah ditambahkan. Berikut gambar *Activity Diagram* pengolahan data Pendidikan.



Gambar 4.9 Activity Diagram pengolahan data Pendidikan

5. Activity Diagram pengolahan data Penelitian

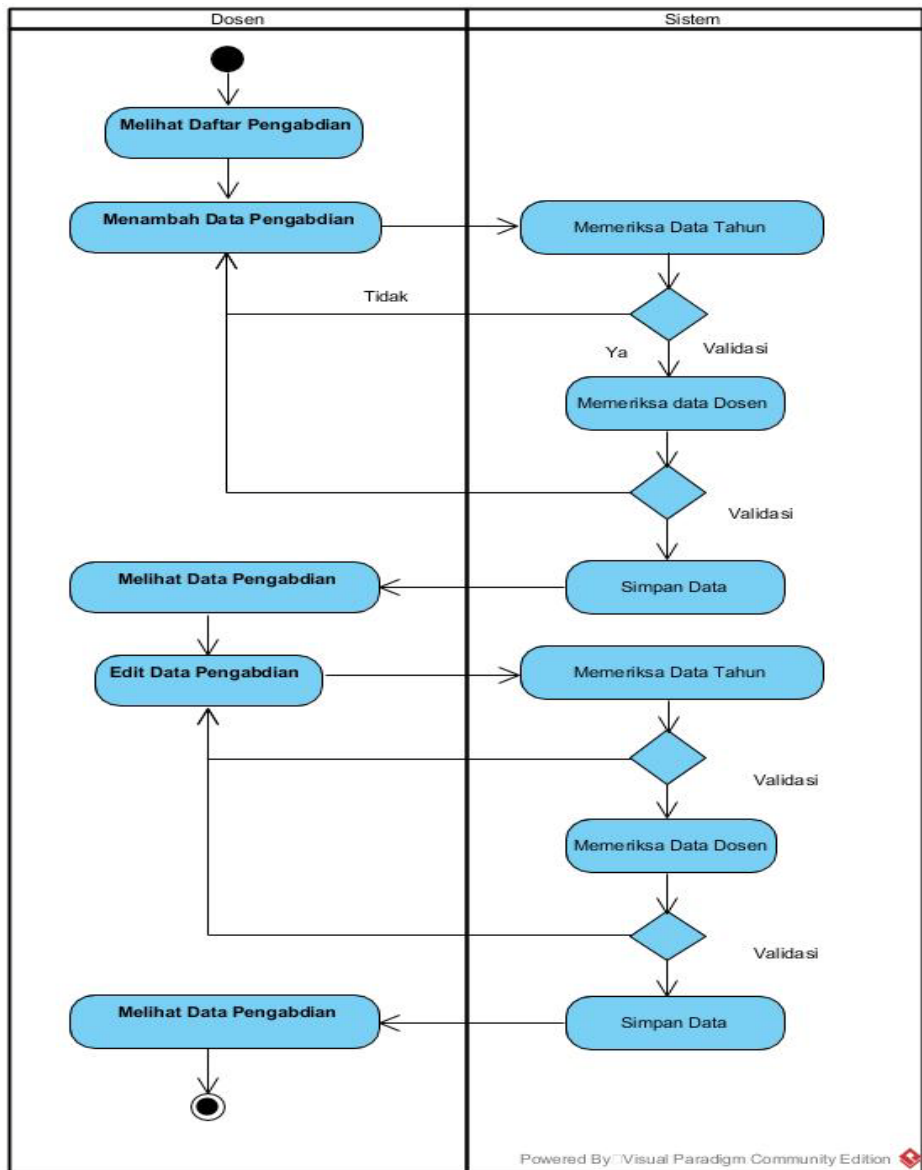
Pengolahan data Penelitian dilakukan oleh masing-masing Dosen, dimana Dosen bisa melakukan penambahan data Penelitian yang dilanjutkan dengan pemeriksaan kelengkapan data oleh sistem terhadap data yang ditambahkan. Pada data Penelitian berelasi dengan Tahun Akademik dan data Dosen. Dosen juga bisa melakukan perubahan terhadap data yang telah ditambahkan. Berikut gambar *Activity Diagram* pengolahan data penelitian.



Gambar 4.10 Activity Diagram pengolahan data penelitian

6. *Activity Diagram* pengolahan data Pengabdian

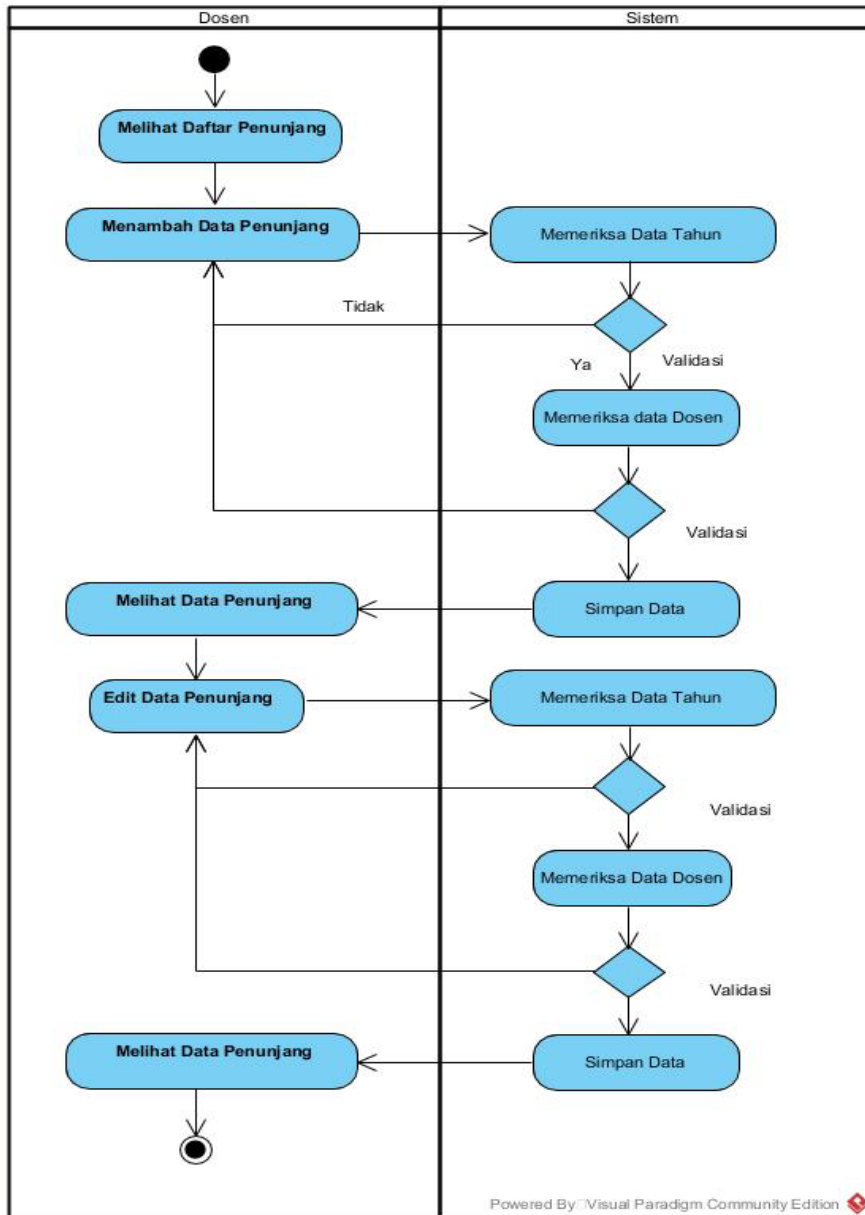
Pengolahan data Pengabdian dilakukan oleh masing-masing Dosen, dimana Dosen bisa melakukan penambahan data Pengabdian yang dilanjutkan dengan pemeriksaan kelengkapan data oleh sistem terhadap data yang ditambahkan. Pada data Pengabdian berelasi dengan Tahun Akademik dan data Dosen. Dosen juga bisa melakukan perubahan terhadap data yang telah ditambahkan. Berikut gambar *Activity Diagram* pengolahan data Pengabdian.



Gambar 4.11 *Activity Diagram* pengolahan data pengabdian

7. Activity Diagram pengolahan data Penunjang

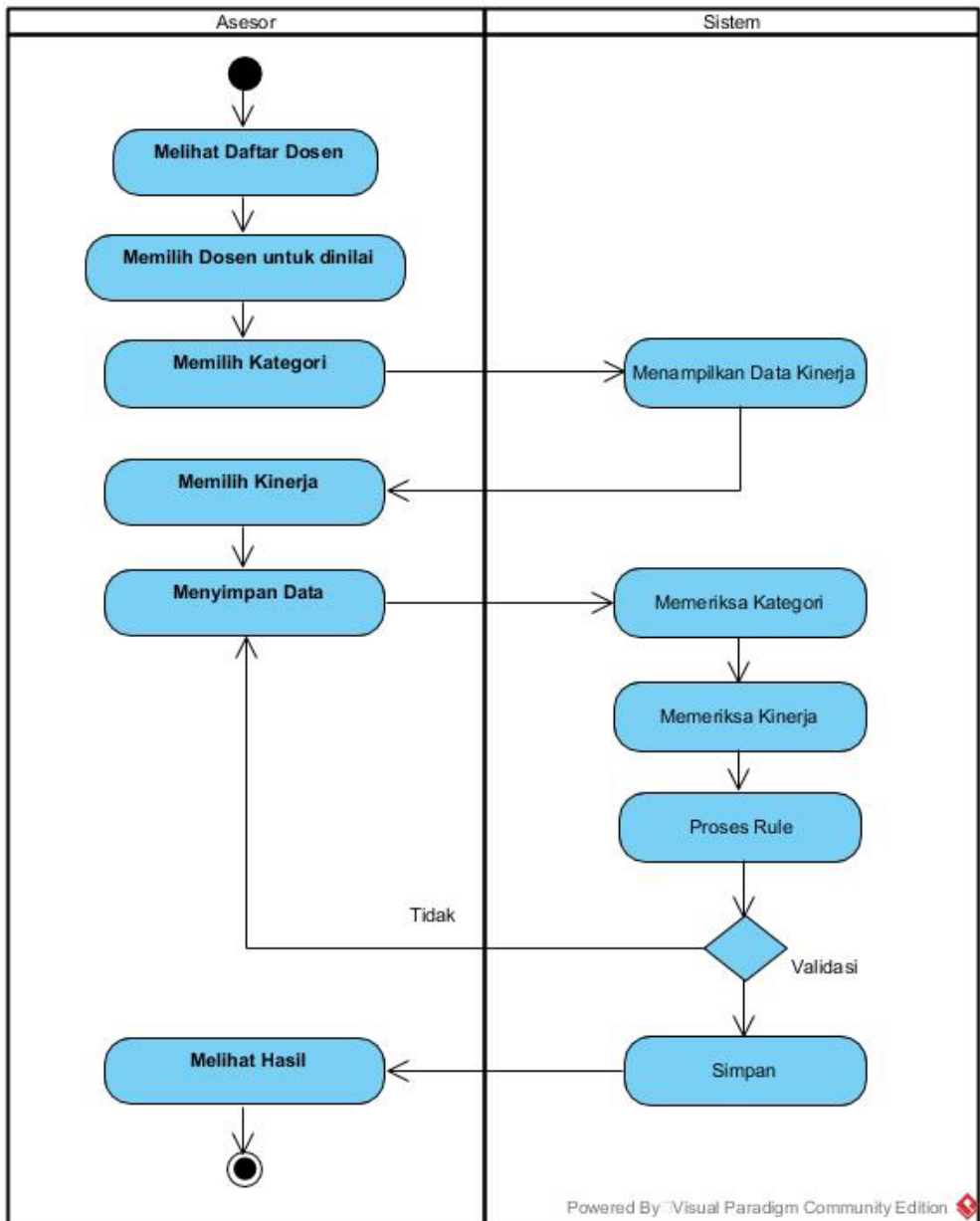
Pengolahan data Penunjang dilakukan oleh masing-masing Dosen, dimana Dosen bisa melakukan penambahan data Penunjang yang dilanjutkan dengan pemeriksaan kelengkapan data oleh sistem terhadap data yang ditambahkan. Pada data Penunjang berelasi dengan Tahun Akademik dan data Dosen. Dosen juga bisa melakukan perubahan terhadap data yang telah ditambahkan. Berikut gambar *Activity Diagram* pengolahan data Penunjang.



Gambar 4.12 Activity Diagram pengolahan data penunjang

8. Activity Diagram Penilaian

Penilaian data Beban Kerja Dosen dilakukan oleh asesor. Aesor bisa melihat semua dosen yang telah ditentukan. Aesor memilih kategori dan kinerja sesuai yang ditampilkan oleh sistem. Sistem akan memproses sesuai dengan Rule yang telah ditentukan. Hasil akan disimpan oleh sistem ke tabel khusus data hasil penilaian.



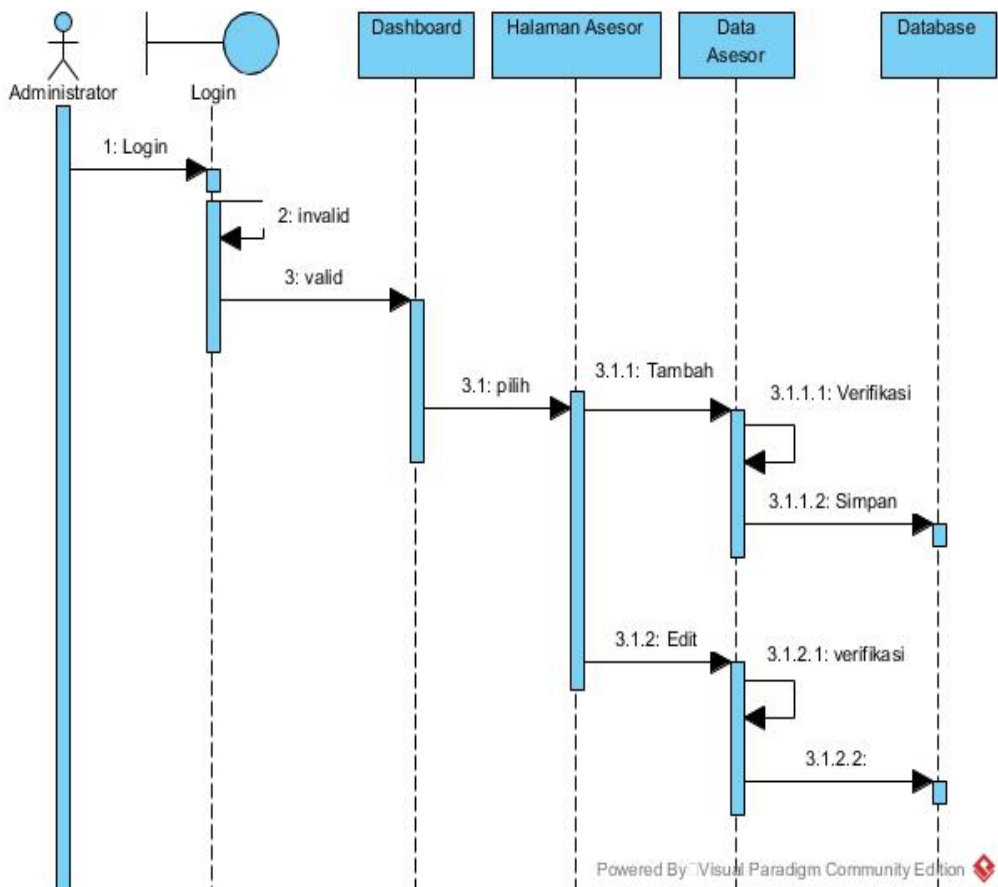
Gambar 4.13 Activity Diagram Penilaian

4.3.4 Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek didalam dan disekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. Adapun *sequence diagram* pada sistem informasi yang dirancang adalah sebagai berikut :

1. *Sequence diagram* pengolahan data Asesor

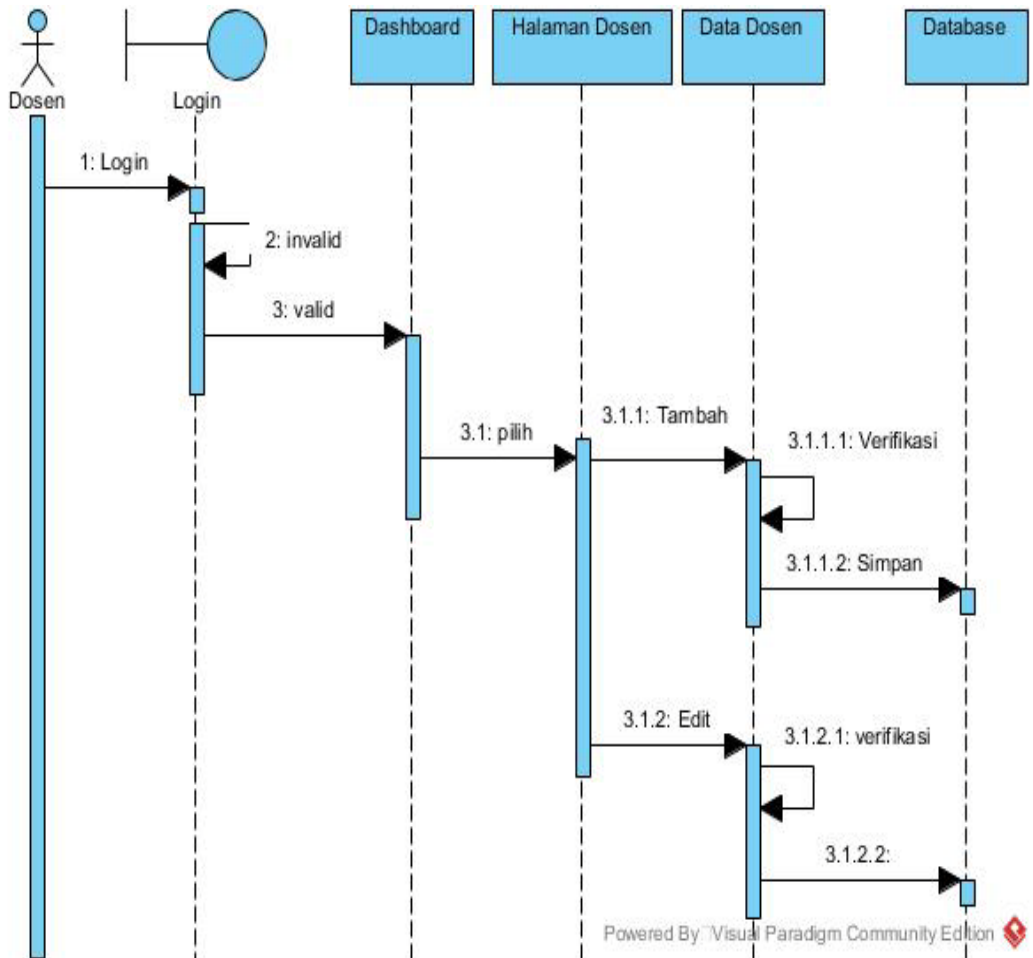
Untuk melakukan pengolahan data kategori Asesor, Administartor yang berperan sebagai *actor* wajib untuk *login* sebelum melakukan pengolahan data. Setelah verifikasi *login* berhasil, maka data Asesor bisa ditambahkan atau dirubah. Data yang ditambahkan atau dirubah akan disimpan ke dalam *database*.



Gambar 4.14 *Sequence Diagram* pengolahan data Asesor

2. Sequence diagram pengolahan data Dosen

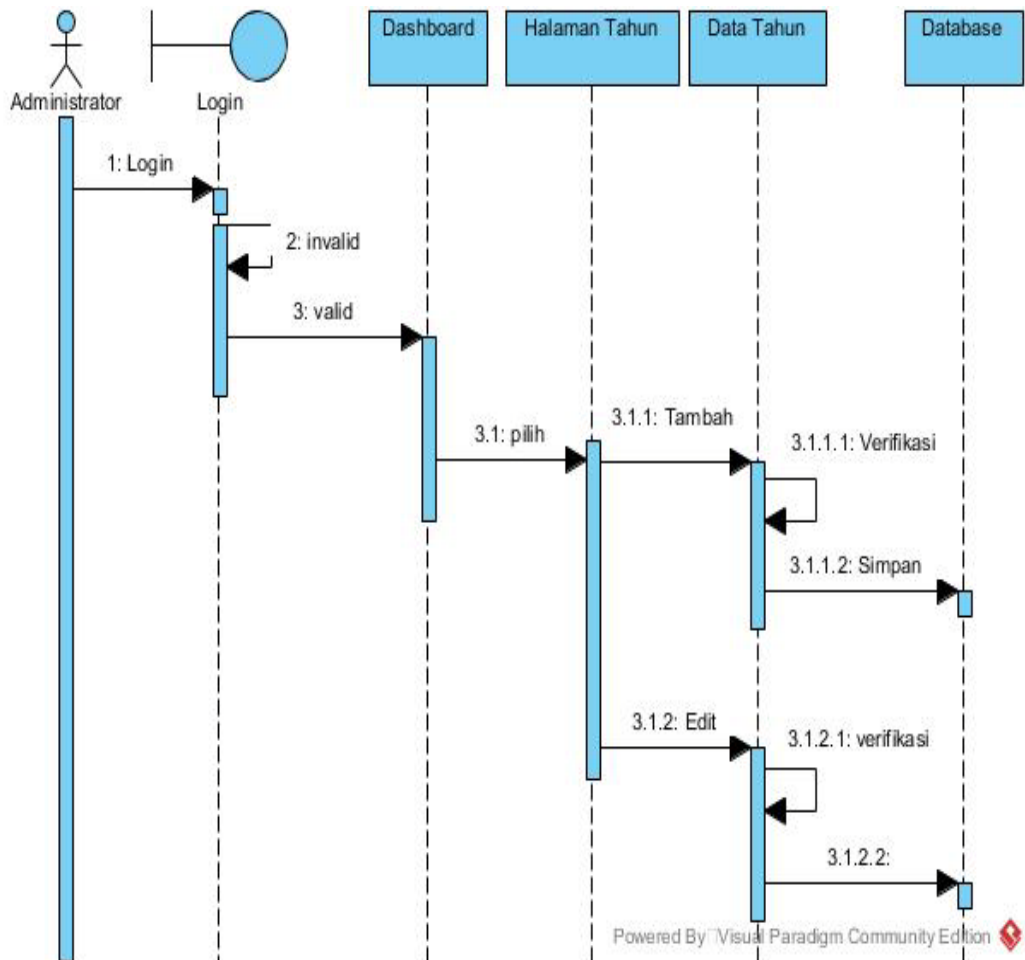
Untuk melakukan pengolahan data dosen, Dosen yang berperan sebagai *actor* wajib untuk *login* sebelum melakukan pengolahan data. Setelah verifikasi *login* berhasil, maka data dosen bisa ditambahkan atau dirubah. Data yang ditambahkan atau dirubah akan disimpan ke dalam *database*.



Gambar 4.15 Sequence Diagram pengolahan data dosen

3. Sequence diagram pengolahan data Tahun

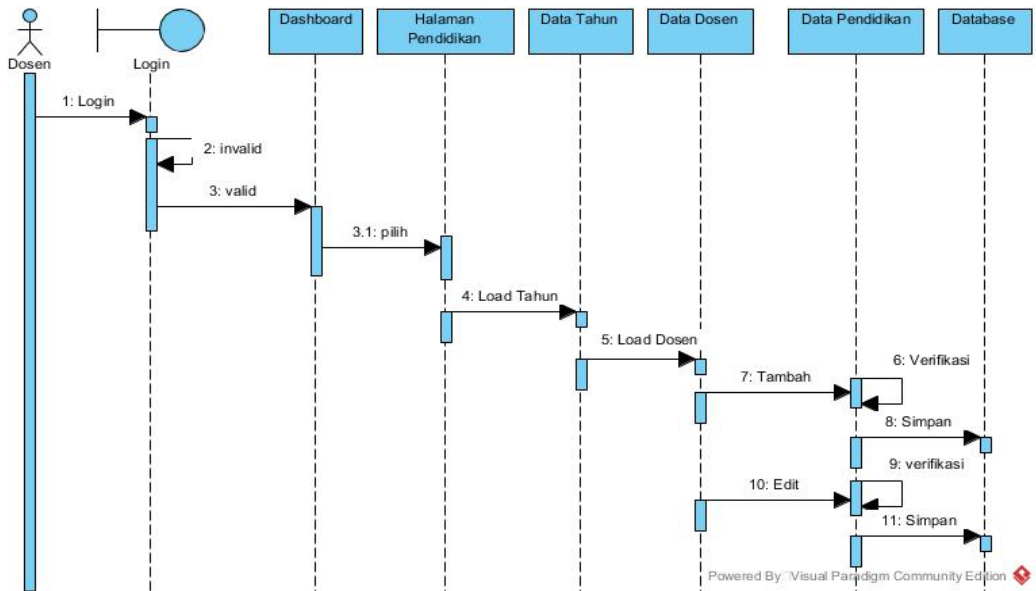
Untuk melakukan pengolahan data tahun, Administrator yang berperan sebagai *actor* wajib untuk login sebelum melakukan pengolahan data tahun. Setelah verifikasi *login* berhasil, maka data tahun bisa diolah. Data yang telah diolah akan disimpan ke dalam *database*. Sequence Diagrama pengolahan data fasilitas bisa dilihat pada gambar berikut ini.



Gambar 4.16 *Sequence Diagram* pengolahan data Tahun

4. *Sequence diagram* Pendidikan

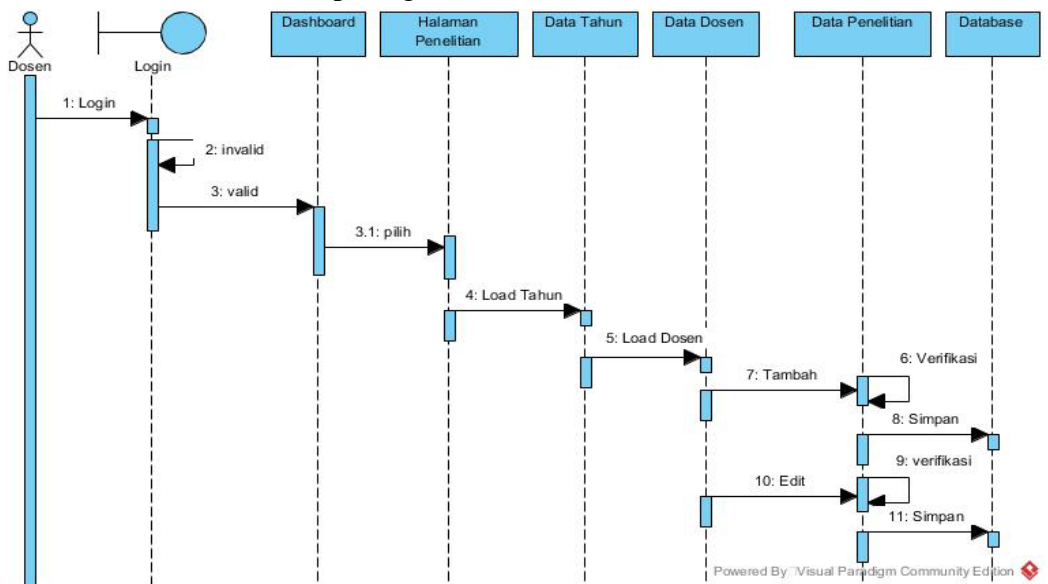
Untuk melakukan pengolahan data Pendidikan, Dosen yang berperan sebagai *actor* wajib untuk login sebelum melakukan pengolahan data. Setelah verifikasi *login* berhasil, maka data Pendidikan bisa diolah. Data yang telah diolah akan disimpan ke dalam *database*. *Sequence Diagram* pengolahan data Pendidikan bisa dilihat pada gambar berikut ini



Gambar 4.17 Sequence Diagram data Pendidikan

5. Sequence diagram Penelitian

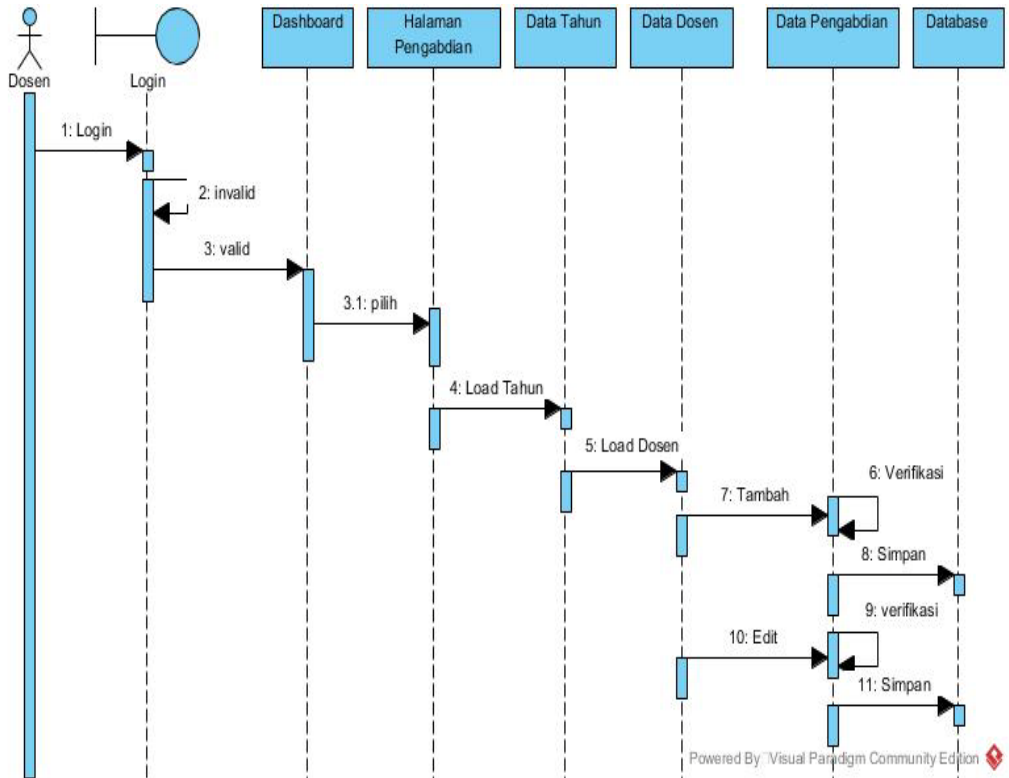
Untuk melakukan pengolahan data Penelitian, Dosen yang berperan sebagai *actor* wajib untuk login sebelum melakukan pengolahan data. Setelah verifikasi *login* berhasil, maka data Penelitian bisa diolah. Data yang telah diolah akan disimpan ke dalam *database*. *Sequence Diagram* pengolahan data Penelitian bisa dilihat pada gambar berikut ini.



Gambar 4.18 Sequence Diagram data Penelitian

6. Sequence diagram Pengabdian

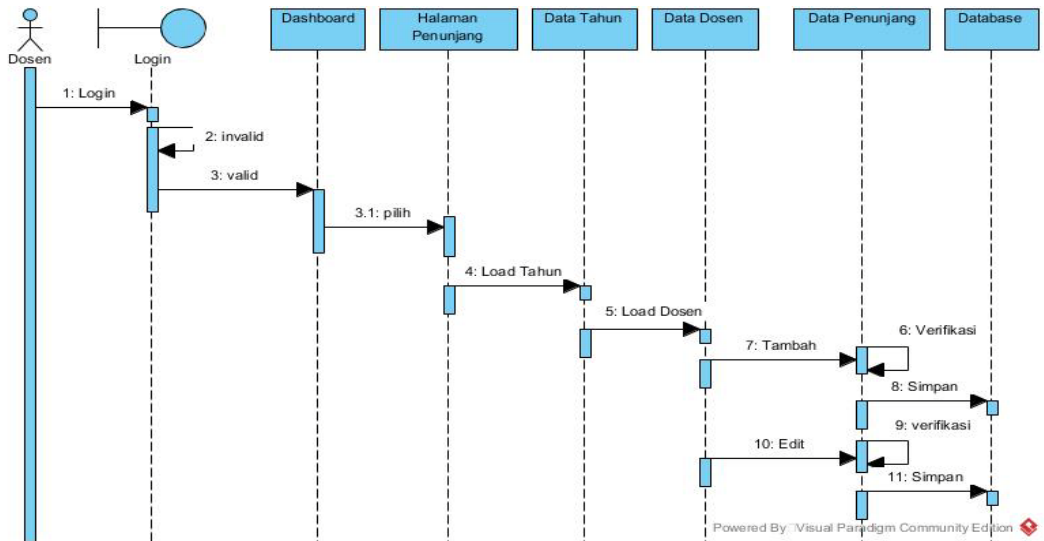
Untuk melakukan pengolahan data Pengabdian, Dosen yang berperan sebagai *actor* wajib untuk login sebelum melakukan pengolahan data. Setelah verifikasi *login* berhasil, maka data Pengabdian bisa diolah. Data yang telah diolah akan disimpan ke dalam *database*. *Sequence Diagram* pengolahan data Pengabdian bisa dilihat pada gambar berikut ini



Gambar 4.19 Sequence Diagram data Pengabdian

7. Sequence diagram Penunjang

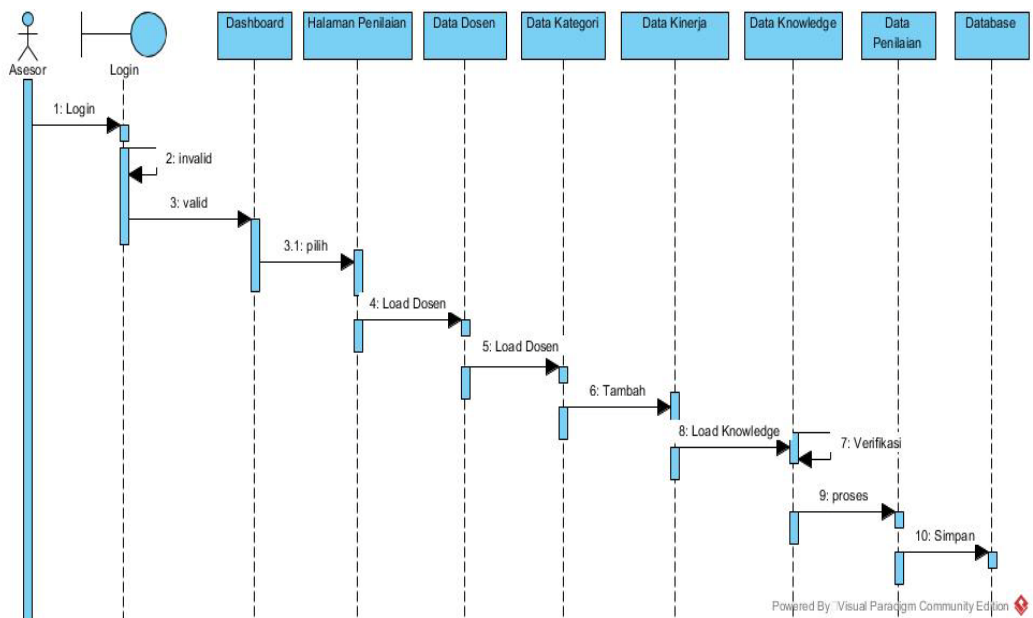
Untuk melakukan pengolahan data Penunjang, Dosen yang berperan sebagai *actor* wajib untuk login sebelum melakukan pengolahan data. Setelah verifikasi *login* berhasil, maka data Penunjang bisa diolah. Data yang telah diolah akan disimpan ke dalam *database*. *Sequence Diagram* pengolahan data Penunjang bisa dilihat pada gambar berikut ini



Gambar 4.20 Sequence Diagram data Penunjang

8. Sequence diagram Penilaian

Untuk melakukan pengolahan data Penilaian. Asesor sebagai actor wajib untuk melakukan Login terlebih dahulu. Menu penilaian akan memanggil data Kategori, Data Kinerja dan diproses berdasarkan Knowledge yang telah dibuat sebagai Rule untuk menampilkan hasil penilaian.



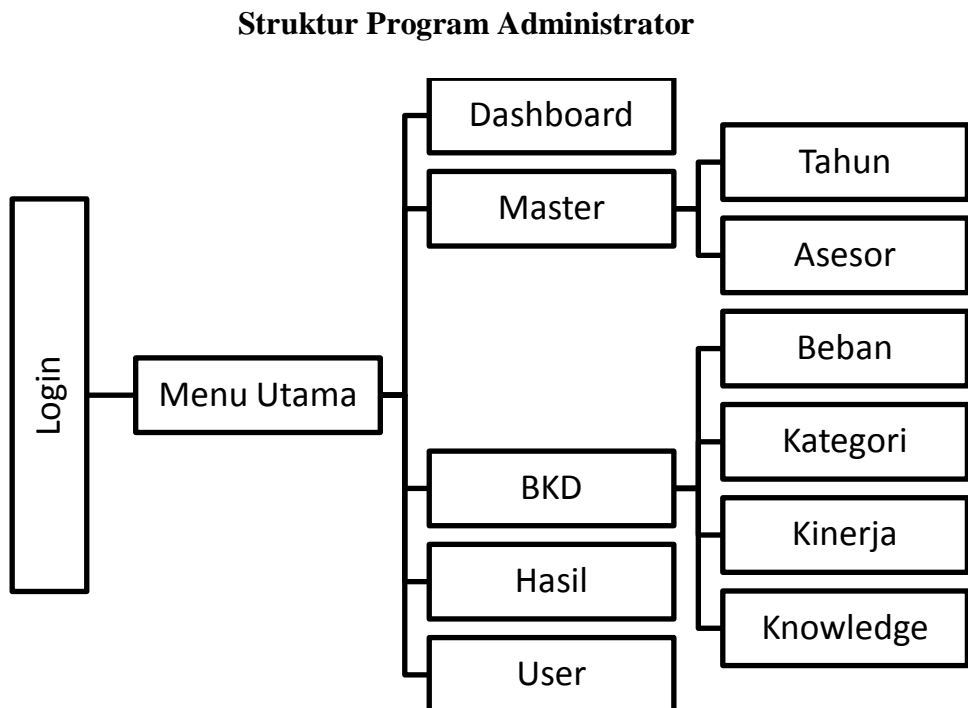
Gambar 4.21 Sequence Diagram data Penilaian

4.3.5 Perancangan Antar Muka

Perancangan antar muka bertujuan untuk memberikan *interface* tentang desain program yang akan dibuat. Di bawah ini terdapat desain *template* pada tampilan sistem yang dibuat oleh penulis.

4.3.5.1 Struktur Program

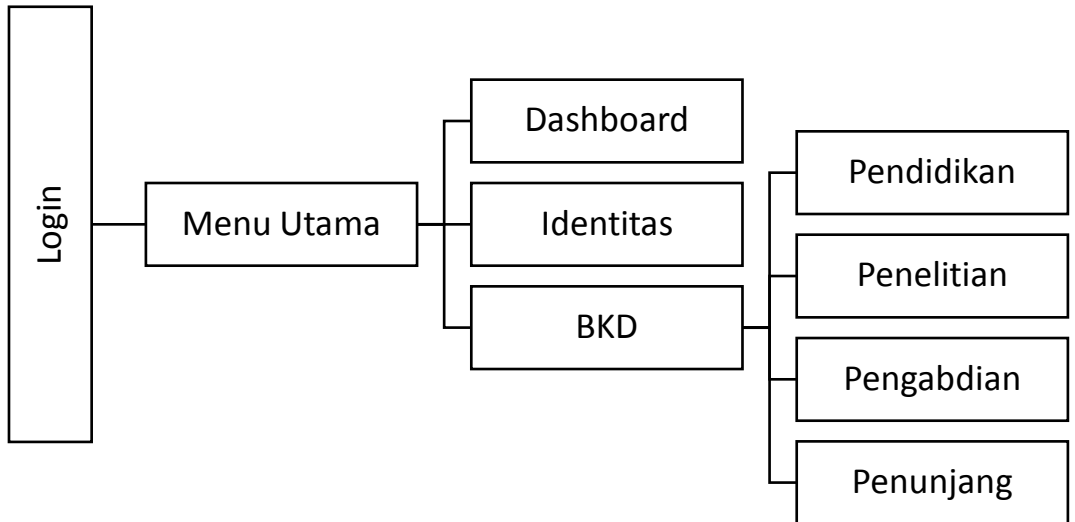
Dalam perancangan sebuah sistem dibutuhkan struktur menu yang berisikan menu dan submenu yang berfungsi untuk memudahkan *user* dalam menggunakan sistem tersebut. Berikut ini digambarkan mengenai stuktur menu program bagian Administrator dalam sistem ini.



Gambar 4.22 Struktur Program Administrator

Struktur program Dosen mempunyai menu utama yang berisi Identitas Dosen, dan BKD. Berikut ini digambarkan mengenai stuktur menu program bagian Dosen dalam sistem ini.

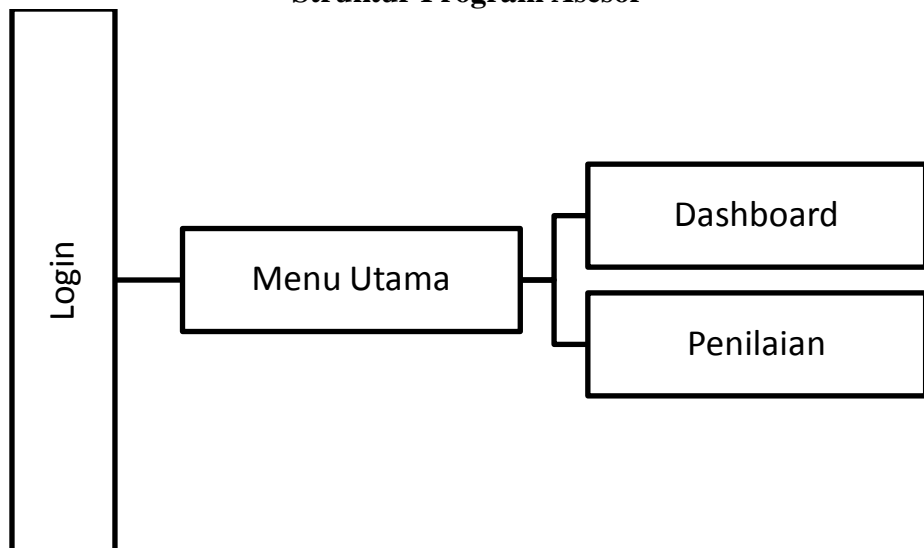
Struktur Program Dosen



Gambar 4.23 Struktur Program Dosen

Struktur program Asesor mempunyai menu utama yang berisi Penilaian BKD. Berikut ini digambarkan mengenai struktur menu program bagian Asesor dalam sistem ini.

Struktur Program Asesor

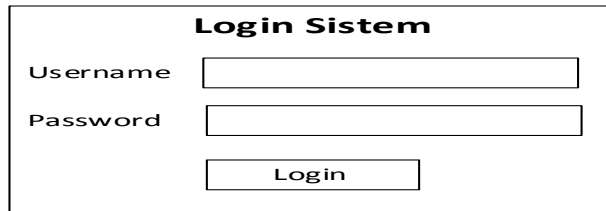


Gambar 4.24 Struktur Program Asesor

4.3.5.2 Perancangan *Input*

1. *Form Login*

Form Login merupakan halaman untuk autentikasi *user*. Setiap *user* yang akan menggunakan melakukan transaksi ke dalam sistem wajib untuk memasukkan *username* dan *password* yang sah agar diberikan hak untuk mengelola akun masing-masing. *User* yang tidak memiliki *username* dan *password* tidak berhak untuk melakukan transaksi atau mengelola sistem.

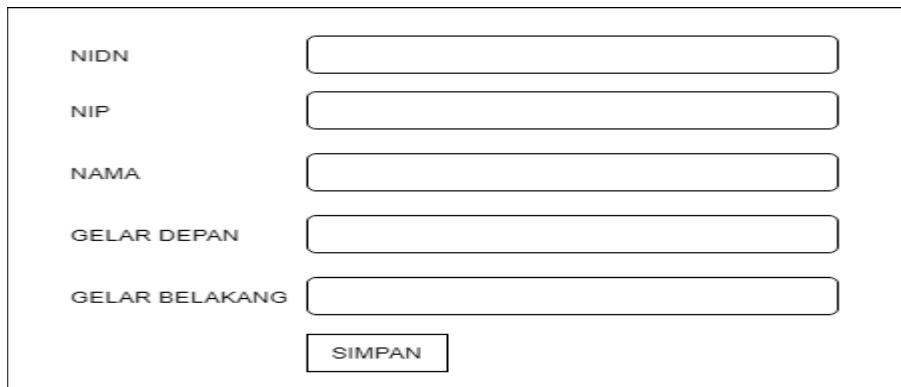


The image shows a login form with the title "Login Sistem". It features two input fields: "Username" and "Password". Below the input fields is a button labeled "Login".

Gambar 4.25 *Form Login*

2. *Form Input Asesor*

Form input asesor berfungsi untuk menambahkan data asesor yang melakukan penilaian.

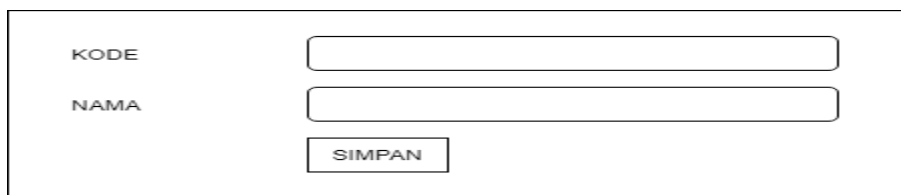


The image shows an input form for an assessor. It contains five input fields: "NIDN", "NIP", "NAMA", "GELAR DEPAN", and "GELAR BELAKANG". Below the input fields is a button labeled "SIMPAN".

Gambar 4.26 *Form Input Asesor*

3. *Form Input Tahun*

Form input Tahun berfungsi untuk menambahkan data Tahun Akademik yang sedang berjalan.



The image shows an input form for a year. It contains two input fields: "KODE" and "NAMA". Below the input fields is a button labeled "SIMPAN".

Gambar 4.27 *Form Input Tahun*

4. *Form Input* Identitas Dosen

Form input Identitas Dosen berfungsi untuk menambahkan data Identitas Dosen kedalam Sistem. Setiap tahun akademik penilaian, setiap dosen wajib untuk menambahkan data identitas. Sehingga asesor sebagai penilai bisa melihat data dosen. Hal ini sejalan dengan sistem yang digunakan saat ini di Universitas Islam Negeri Sumatera Utara Medan.

NO SERTIFIKAT	<input type="text"/>	<input type="button" value="FILE"/>
NIDN	<input type="text"/>	
NIP	<input type="text"/>	
NAMA	<input type="text"/>	
GELAR DEPAN	<input type="text"/>	GELAR BELAKANG <input type="text"/>
PERGURUAN TINGGI	<input type="text"/>	
ALAMAT PT	<input type="text"/>	
PRODI	<input type="text"/>	
FUNGSIONAL	<input type="text"/>	GOLONGAN <input type="text"/>
PRODI	<input type="text"/>	
TGL LHR	<input type="text"/>	TMP LAHIR <input type="text"/>
PENDIDIKAN S1	<input type="text"/>	<input type="button" value="FILE"/>
PENDIDIKAN S2	<input type="text"/>	<input type="button" value="FILE"/>
PENDIDIKAN S3	<input type="text"/>	<input type="button" value="FILE"/>
JENIS ILMU	<input type="text"/>	
BIDANG	<input type="text"/>	
HP	<input type="text"/>	
TAHUN	<input type="text"/>	
KTP	<input type="button" value="FILE"/>	
	<input type="button" value="SIMPAN"/>	

Gambar 4.28 *Form Input* Identitas Dosen

5. *Form* Penilaian

Form Penilaian berfungsi untuk melakukan penilaian oleh Asesor terhadap BKD yang telah ditambahkan masing-masing dosen pada Tahun Akademik berjalan. Setiap Kinerja yang ditambahkan akan muncul dan asesor menjadi penilai, apakah kinerja dosen layak atau tidak dan masuk dalam kategori dan kinerja seperti apa. Berikut *Form* Penilaian.

NIDN : XXX			
NAMA : XXX			
PENDIDIKAN	KATEGORI	KINERJA	
<input type="text"/>	<input type="text"/>	<input type="text"/>	PROSES
PENELITIAN	KATEGORI	KINERJA	
<input type="text"/>	<input type="text"/>	<input type="text"/>	PROSES
PENGABDIAN	KATEGORI	KINERJA	
<input type="text"/>	<input type="text"/>	<input type="text"/>	PROSES
PENUNJANG	KATEGORI	KINERJA	
<input type="text"/>	<input type="text"/>	<input type="text"/>	PROSES

Gambar 4.29 *Form* Penilaian

BAB V

IMPLEMENTASI DAN HASIL

5.1 Implementasi Sistem

Setelah dilakukan perancangan sistem, maka tahap selanjutnya adalah pengimplementasian dan pengujian sistem. Tujuan dari implementasi yaitu untuk menerapkan sistem agar dapat dioperasikan secara optimal sesuai dengan ketentuan proses. Pengujian sistem merupakan tahap uji coba terhadap sistem yang telah dibuat apakah sistem sudah berjalan dengan benar serta uji coba langsung bagi pengguna bagaimana cara menjalankannya.

5.1.1 Komponen Utama dalam Implementasi Sistem

Komponen utama dalam implementasi sistem ini terbagi atas beberapa komponen yaitu perangkat keras (*Hardware*), perangkat lunak (*Software*) dan sumber daya manusia (*Brainware*).

1. Perangkat Keras (*Hardware*)

Spesifikasi perangkat keras yang digunakan untuk mengimplementasikan aplikasi sistem ini adalah sebagai berikut:

- a. *PC Processor Intel Core 1,8 GHz*
- b. *Harddisk 500 GB*
- c. *RAM 4 GB*
- d. *Intel Graphic Onboard 1 GB*

2. Perangkat Lunak (*Software*)

Spesifikasi perangkat lunak (*software*) yang digunakan untuk mengimplementasikan aplikasi sistem ini adalah sebagai berikut:

- a. *Windows 10 Professional*
- b. *Mozilla Firefox 38.0*
- c. *XAMPP 7.0.13*

3. Sumber Daya Manusia (*Brainware*)

Pada implementasi ini dibutuhkan seorang *user* yang mampu menjalankan atau mengoperasikan komputer dan menjalankan sistem yang telah dibuat sehingga penerapan dan implementasi sistem dapat berjalan dengan lancar.

5.1.2 Kebutuhan *Server* dalam Implementasi Sistem

Agar sistem yang dirancang bisa berjalan dengan sempurna, maka butuh *server* yang mengakomodasi kebutuhan bahasa pemrograman. *Server* yang dimaksud sudah tersedia secara sempurna dalam satu buah *software* yaitu XAMPP 7.2.0 Dalam *software* tersebut terdapat:

1. Apache2
2. PHP7
3. MariaDB
4. phpMyAdmin

5.2 Pengujian Sistem

Pengujian sistem dilakukan pada modul-modul yang telah dirancang pada sistem informasi penerimaan remunerasi dosen. Adapun pengujian yang dilakukan adalah sebagai berikut:

1. Pengujian *Login* User

Pengujian yang dilakukan pada *form login* yaitu dengan melakukan *input* data sesuai *form* yang disediakan.

Untuk lebih jelasnya bisa dilihat pada tabel 5.1 berikut ini.

Tabel 5.1 Pengujian *Login* User

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data <i>username</i> dan <i>password</i> secara lengkap	Dapat masuk kedalam sistem melewati <i>form login</i>	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak memasukkan salah satu data	Sistem menolak proses dan menampilkan peringatan kesalahan input	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Memasukkan data <i>login</i> salah	Sistem menolak proses dan menampilkan peringatan kesalahan input	Proses berhasil sesuai yang diharapkan	Berjalan

2. Pengujian Data Asesor

Pengujian yang dilakukan pada data Asesor yaitu dengan melakukan melakukan penambahan dan perbaikan data Asesor pada *form* yang disediakan. Untuk lebih jelasnya bisa dilihat pada tabel 5.2 berikut ini.

Tabel 5.2 Pengujian Data Asesor

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan informasi data berhasil disimpan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak memasukkan data wajib	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

3. Pengujian Data Tahun

Pengujian yang dilakukan pada Tahun dengan melakukan penambahan dan perubahan data Tahun dengan data sesuai *form* yang disediakan. Untuk lebih jelasnya bisa dilihat pada tabel 5.3 berikut ini.

Tabel 5.3 Pengujian Data Tahun

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan informasi data berhasil disimpan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak	Sistem menolak	Proses	Berjalan

memasukkan data wajib	proses dan menampilkan peringatan kesalahan	berhasil sesuai yang diharapkan	
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

4. Pengujian Data Dosen

Pengujian yang dilakukan pada data Dosen yaitu dengan melakukan penambahan dan perbaikan data sesuai *form* yang disediakan.. Untuk lebih jelasnya bisa dilihat pada tabel 5.4 berikut ini.

Tabel 5.4 Pengujian Data Dosen

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan informasi data berhasil disimpan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak memasukkan data wajib	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

5. Pengujian Data Pendidikan

Pengujian yang dilakukan pada data Pendidikan yaitu dengan melakukan *input* data sesuai *form* yang disediakan. Untuk lebih jelasnya bisa dilihat pada tabel 5.5 berikut ini.

Tabel 5.5 Pengujian Data Pendidikan

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan informasi data berhasil disimpan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak memasukkan data wajib	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

6. Pengujian Data Penelitian

Pengujian yang dilakukan pada data penelitian yaitu dengan melakukan *input* data sesuai *form* yang disediakan. Untuk lebih jelasnya bisa dilihat pada tabel 5.6 berikut ini.

Tabel 5.6 Pengujian Data Penelitian

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan	Proses berhasil sesuai yang	Berjalan

	informasi data berhasil disimpan	diharapkan	
Data Kurang			
Tidak memasukkan data wajib	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

7. Pengujian Data Pengabdian

Pengujian yang dilakukan pada data pengabdian yaitu dengan melakukan proses *input* data sesuai *form* yang disediakan. Untuk lebih jelasnya bisa dilihat pada tabel 5.7 berikut ini.

Tabel 5.7 Pengujian Data Pengabdian

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan informasi data berhasil disimpan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak memasukkan data wajib	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

8. Pengujian Data Penunjang

Pengujian yang dilakukan pada data penunjang yaitu dengan melakukan proses *input* data sesuai *form* yang disediakan. Untuk lebih jelasnya bisa dilihat pada tabel 5.8 berikut ini.

Tabel 5.8 Pengujian Data Penunjang

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan informasi data berhasil disimpan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak memasukkan data wajib	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

9. Pengujian Penilaian

Pengujian yang dilakukan pada data penilaian yaitu melakukan proses menilai BKD dosen dengan memilih Kategori dan Kinerja dari BKD tersebut dan sistem mencocokkannya dengan Knowledge yang tersedia. Jika ada dalam Knowledge, maka sistem akan memberikan hasil.

Tabel 5.9 Pengujian Data Penilaian

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memilih Kategori dan Kinerja	Sistem menampilkan hasil sesuai knowledge	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak Memilih Kategori atau Kinerja	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Salah			
Tidak Memilih Kategori dan Kinerja	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

10. Pengujian Data *User*

Pengujian yang dilakukan pada data *user* yaitu dengan melakukan *input* data sesuai *form* yang disediakan. Untuk lebih jelasnya bisa dilihat pada tabel 5.10 berikut ini.

Tabel 5.10 Pengujian Data *User*

Data Masukan	Proses Diharapkan	Pengamatan	Kesimpulan
Data Normal			
Memasukkan data secara lengkap	Sistem menyimpan data dan menampilkan informasi data berhasil disimpan	Proses berhasil sesuai yang diharapkan	Berjalan
Data Kurang			
Tidak memasukkan	Sistem menolak proses dan	Proses berhasil sesuai	Berjalan

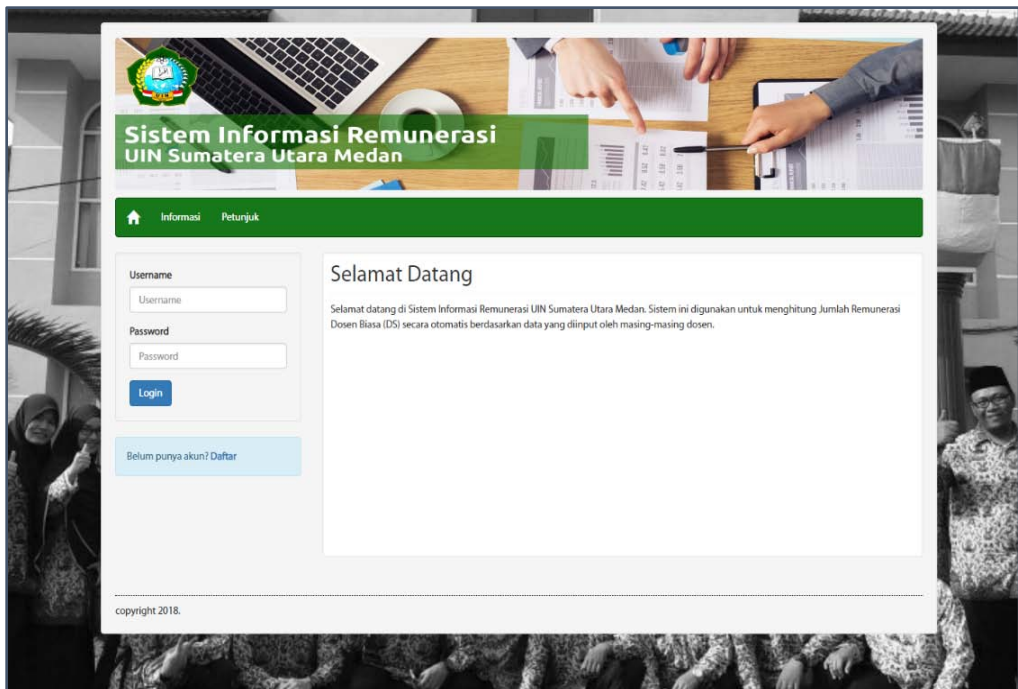
data wajib	menampilkan peringatan kesalahan	yang diharapkan	
Data Salah			
Tidak memasukkan data dan menekan tombol simpan	Sistem menolak proses dan menampilkan peringatan kesalahan	Proses berhasil sesuai yang diharapkan	Berjalan

5.3 Hasil Pengujian

Hasil pengujian merupakan tampilan hasil akhir dari sistem informasi penentuan penerimaan remunerasi dosen biasa UIN Sumatera Utara Medan. Adapun hasil pengujian adalah sebagai berikut:

1. Halaman Utama

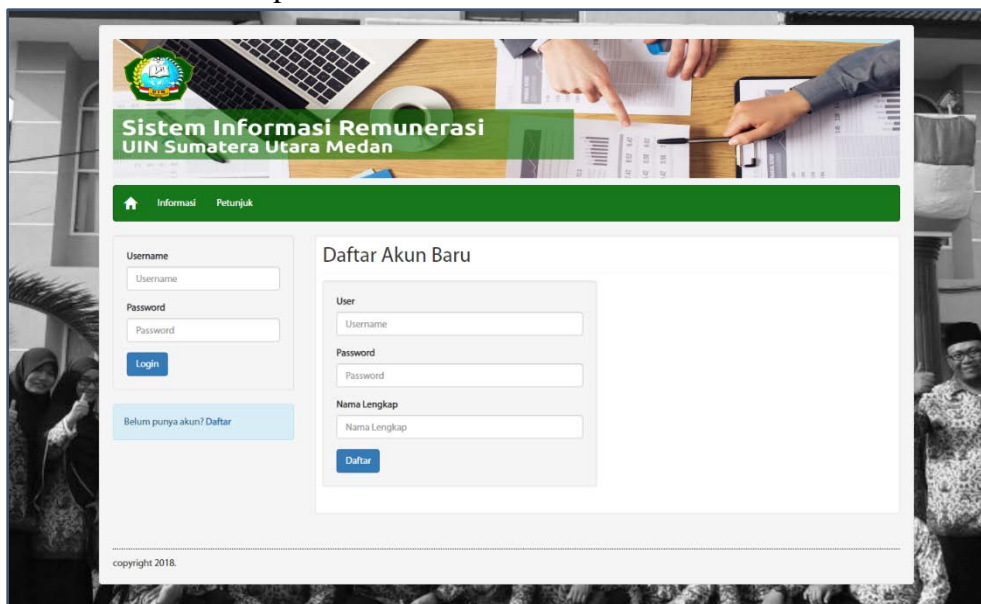
Halaman utama sistem merupakan halaman awal yang dijumpai dimana oleh menggunakan sistem. Setiap pengguna wajib login menggunakan Username dan Password untuk masuk ke halaman berikutnya.



Gambar 5.1. Halaman Utama

2. Daftar User

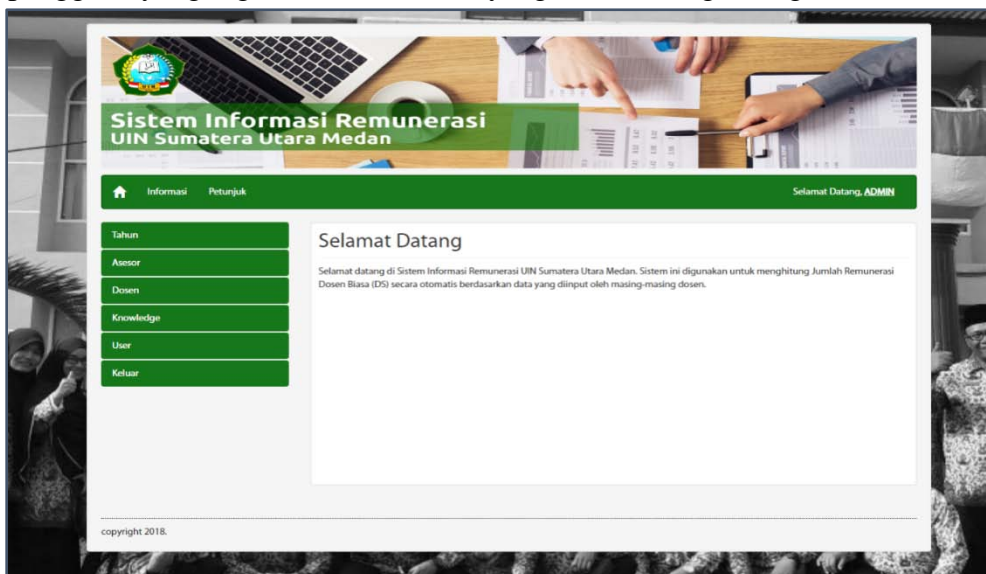
Jika pengguna belum mempunyai akun untuk masuk kedalam sistem, user bisa melakukan pendaftaran akun terlebih dahulu



Gambar 5.2. Daftar User

3. Dashboard

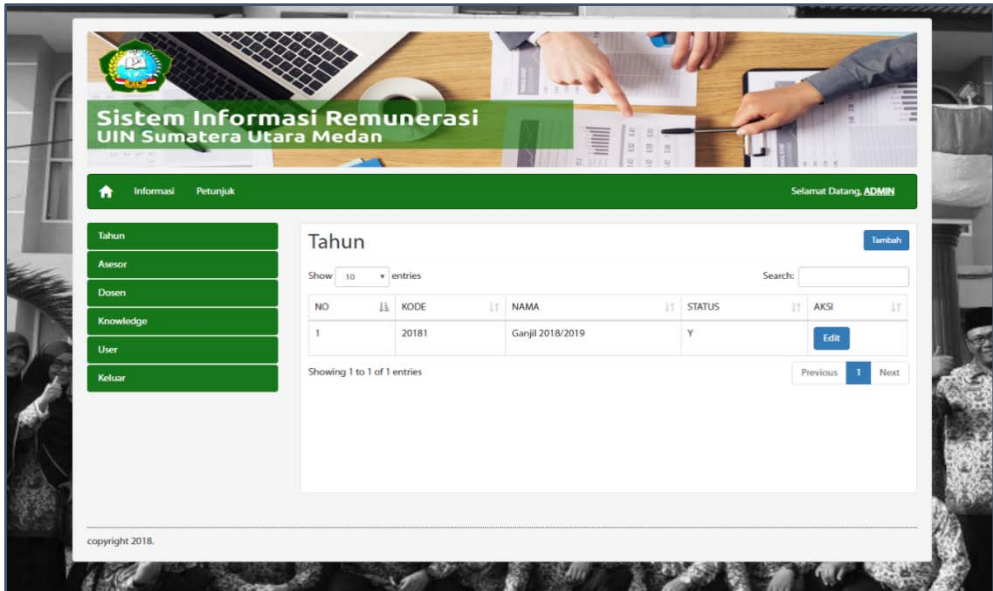
Halaman Dashboard merupakan halaman yang dapat dilihat oleh pengguna sistem setelah berhasil melakukan login kedalam sistem. Setiap pengguna yang login memiliki menu yang berbeda tergantung level user.



Gambar 5.3. Halaman Dashboard

4. Tahun

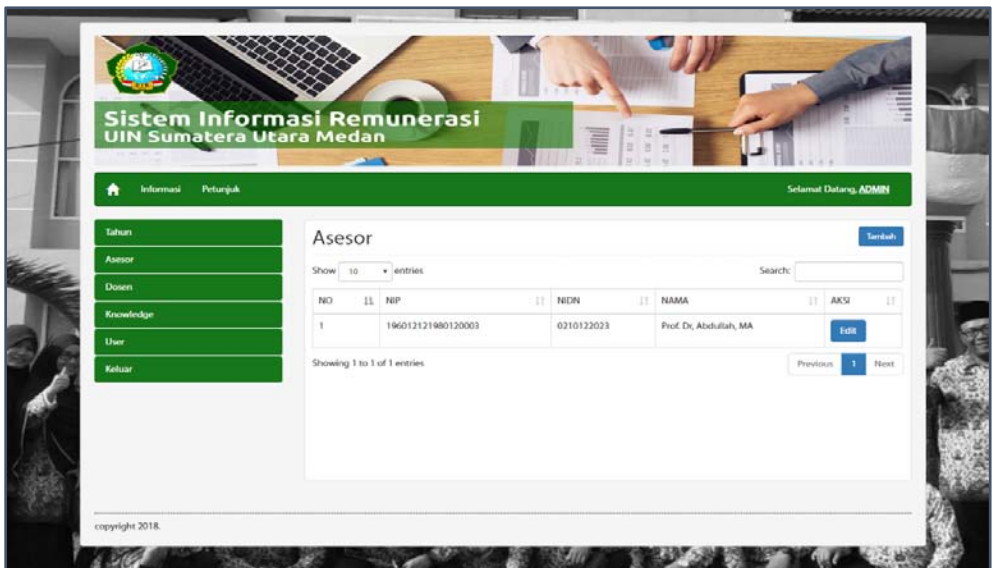
Menu tahun berfungsi untuk menyimpan data tahun setiap semester berjalan.



Gambar 5.4. Halaman Tahun

5. Asesor

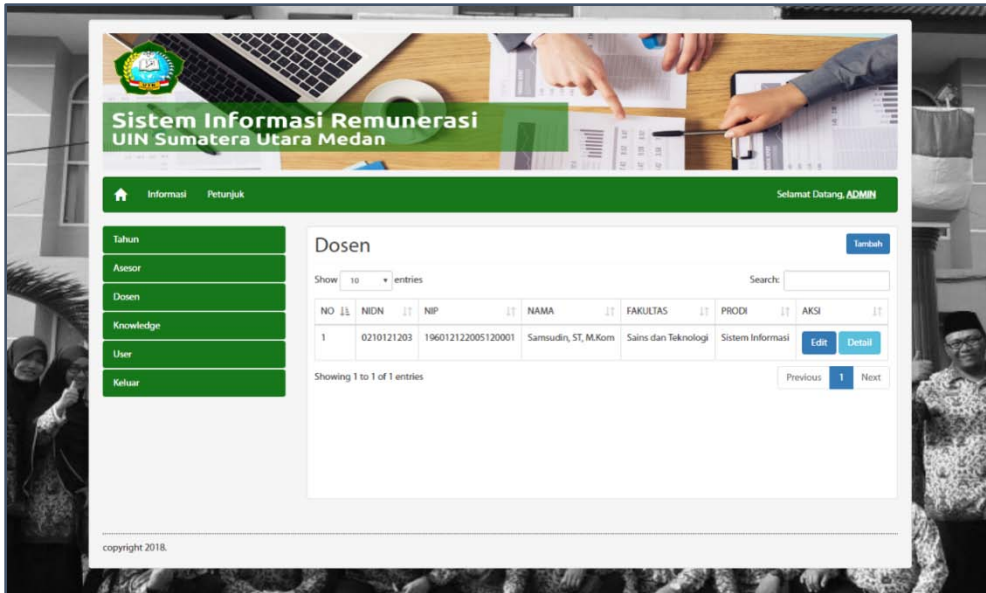
Menu asesor berfungsi untuk menyimpan data asesor yang melakukan penilaian terhadap Beban Kerja Dosen.



Gambar 5.5. Halaman Asesor

6. Dosen

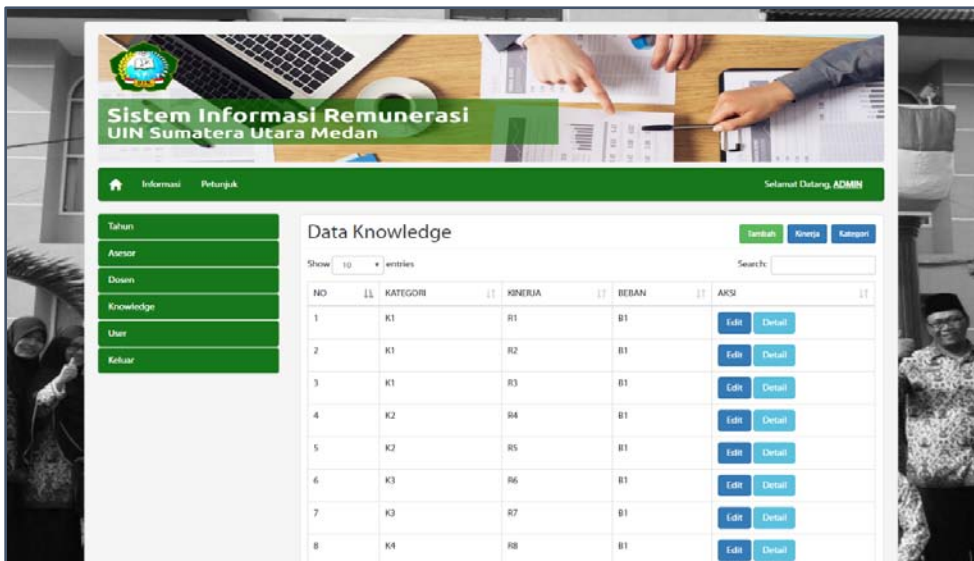
Menu dosen berfungsi untuk menampilkan data dosen yang ada dalam sistem.



Gambar 5.6. Halaman Dosen

7. Knowledge

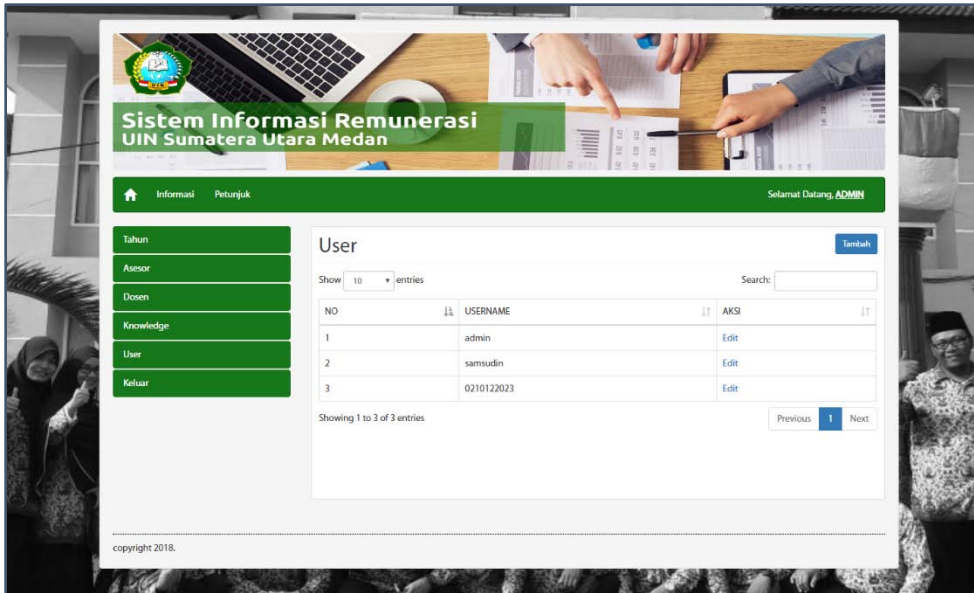
Menu knowledge berfungsi untuk menyimpan seluruh data pengetahuan sesuai dengan Rule yang telah ditentukan.



Gambar 5.7. Halaman Menu Knowledge

8. User

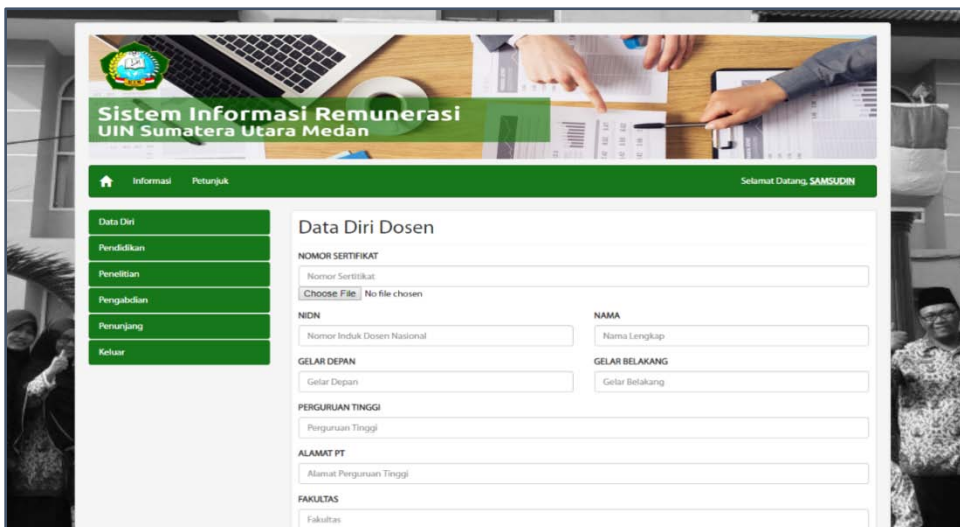
Menu user berfungsi untuk menampilkan data pengguna sistem. Pada menu ini bisa digunakan untuk melakukan perubahan password atau pembuatan akun pengguna baru.



Gambar 5.8. Halaman Menu User

9. Data Diri

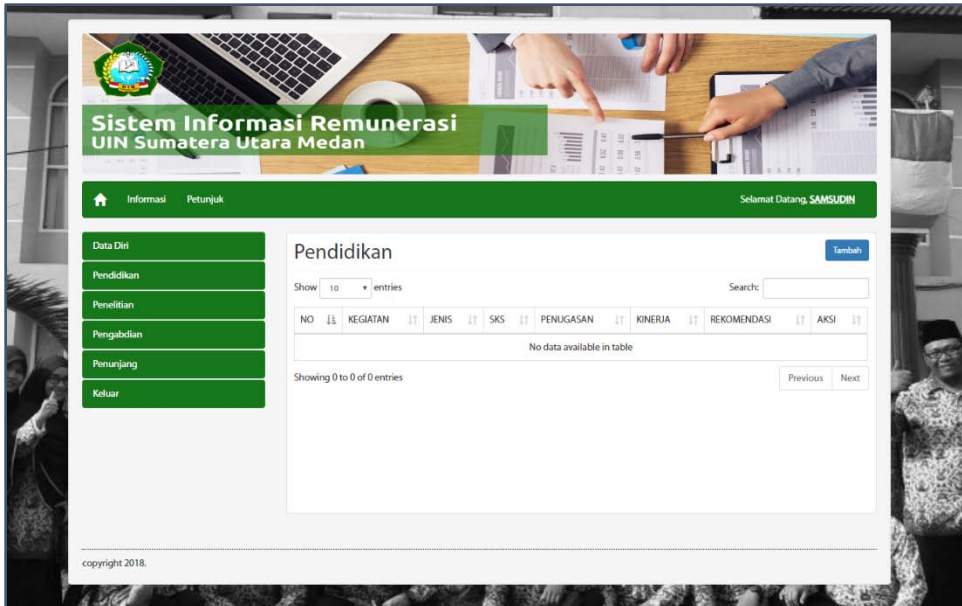
Menu data diri dapat ditemukan pada akun dosen. Menu ini berfungsi untuk menyimpan data dosen yang akan melakukan penyerahan data beban kerja dosen.



Gambar 5.9. Halaman Menu Data Diri Dosen

10. Pendidikan

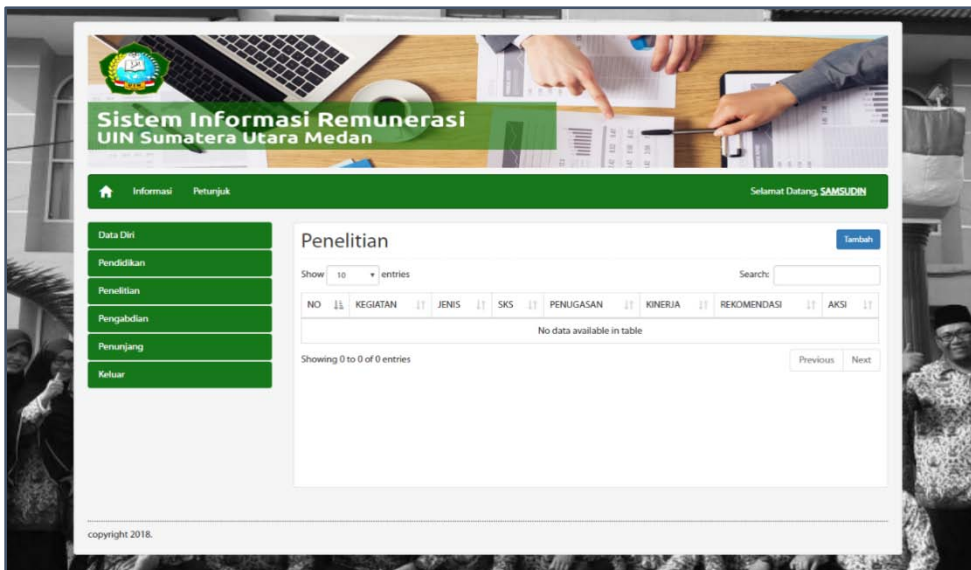
Menu pendidikan berfungsi untuk menyimpan seluruh data pendidikan yang dilakukan oleh dosen.



Gambar 5.10. Halaman Menu Kinerja Pendidikan Dosen

11. Penelitian

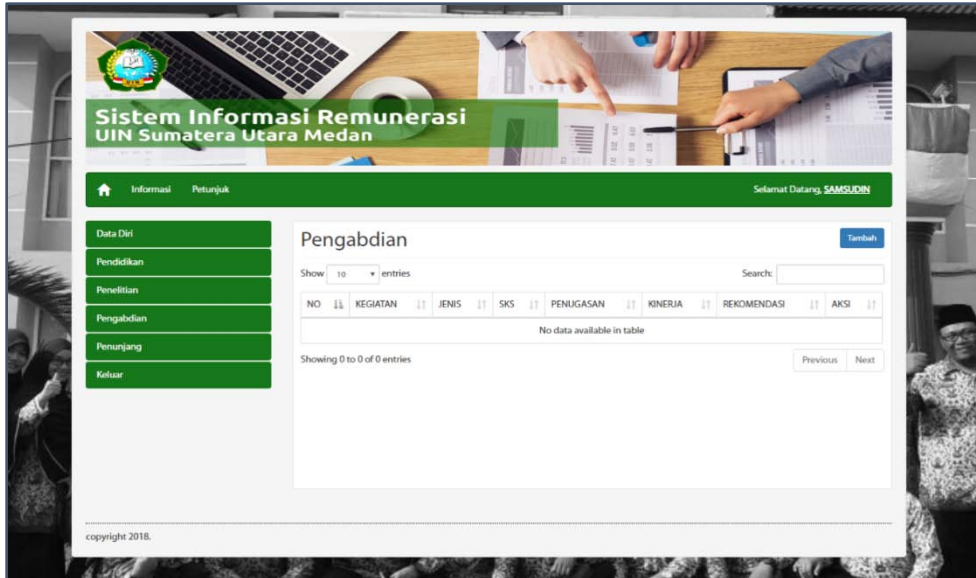
Menu penelitian berfungsi untuk melakukan penyimpanan data penelitian yang dilakukan oleh dosen.



Gambar 5.11. Halaman Menu Kinerja Penelitian Dosen

12. Pengabdian

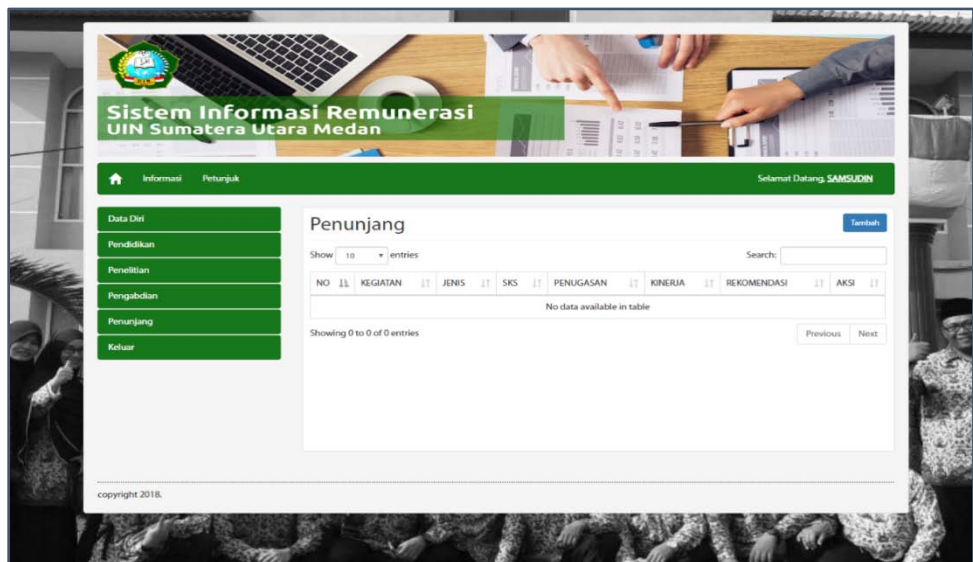
Menu pengabdian berfungsi untuk melakukan penyimpanan data pengabdian yang dilakukan oleh dosen.



Gambar 5.12. Halaman Menu Kinerja Pengabdian Dosen

13. Penunjang

Menu penunjang berfungsi untuk melakukan penyimpanan data penunjang yang dilakukan oleh dosen.



Gambar 5.13. Halaman Menu Kinerja Penunjang Dosen

14. Penilaian

Menu penilaian dapat ditemukan pada akun asesor, menu ini berfungsi untuk melakukan penilaian terhadap beban kerja dosen yang telah di-input oleh masing-masing dosen.

The screenshot shows the 'Sistem Informasi Remunerasi UIN Sumatera Utara Medan' interface. The main content area is titled 'Penilaian' and contains a table for 'Penilaian Beban Kerja Dosen Semester Gengil 2018/2019 (20181)'. The table has the following columns: NO, NIDN, NAMA, PENDIDIKAN, PENELITIAN, PENGABDIAN, PENUNJANG, and REMUNERASI. The data row shows: NO 1, NIDN 0210121203, NAMA Samsudin, ST, M.Kom, and 'Proses' status for all activity categories, with a remuneration of 0.0. The page also includes a sidebar with 'Penilaian', 'Profil', and 'Keluar' options, and a footer with 'copyright 2018'.

NO	NIDN	NAMA	PENDIDIKAN	PENELITIAN	PENGABDIAN	PENUNJANG	REMUNERASI
1	0210121203	Samsudin, ST, M.Kom	Proses	Proses	Proses	Proses	0.0

Gambar 5.14. Halaman Menu Penilaian Dosen

BAB VI PENUTUP

Berdasarkan uraian dan pembahasan pada bab-bab sebelumnya didapatkan hasil yang bermanfaat pada perancangan sistem penentuan penerimaan remunerasi dosen UIN Sumatera Utara Medan menggunakan metode *Rule Base Reasoning*. Sedangkan untuk pengembangan penelitian berkelanjutan dituliskan beberapa saran yang bisa diteruskan dan dijadikan kajian untuk bahan penelitian selanjutnya.

6.1 Kesimpulan

Mengacu kepada rumusan masalah yang telah dibahas maka dapat ditarik beberapa kesimpulan.

1. Perancangan sistem penentuan penerimaan remunerasi dosen UIN Sumatera Utara Medan menggunakan Metode *Rule Base Reasoning* efektif untuk membantu kinerja Asesor dalam melakukan penilaian
2. Konsep *Object Oriented* dan sistem berbasis web yang dibangun sangat membantu sistem yang lebih reusable dan bisa diakses secara *realtime*, tanpa terbatas waktu dan tempat.

6.2 Saran

Sebagai bahan penelitian lanjutan, berikut ini adalah beberapa saran yang bisa dijadikan bahan penelitian selanjutnya.

1. Sistem bisa lebih disempurnakan dengan knowledge yang lebih komplit berdasarkan kebijakn UIN Sumatera Utara Medan.
2. Penelitian belum membasa tentang keamanan sistem berbasis web, hal ini menjadi bahan pertimbangan yang sangat penting untuk kedepannya.

DAFTAR PUSTAKA

- Agung Gregorius. 2012. **Adobe Dreamweaver CS6**. Jakarta: Elex Media Komputindo.
- Hartono Bambang. 2013. **Sistem Informasi Manajemen Berbasis komputer**. Bandung: Rineka Cipta
- Basuki, Awan Pribadi. 2016. **Konsep dan Implementasi Pemrograman Laravel 5**. Yogyakarta: CV.Lokomedia.
- Basuki, Adian Tri. 2011. **Perancangan Aplikasi Sistem Informasi Cuti Karyawan Berbasis Web Pada PT Integrasi Tri Tama Cendekia**. Jakarta: UIN Syarif Hidayatullah.
- Blanco, Jose A. and Upton, David. 2009. **CodeIgniter 1.7**. Birmingham: Packt Publishing
- Fatima, Siti. 2013. **Perancangan Sistem Informasi Penjualan Mebel Online pada UD. Melindo Jaya**. Kisaran: AMIK Royal Kisaran.
- Griffiths, Adam. 2010. **CodeIgniter 1.7 Professional Development**. Birmingham : Packt Publishing
- Hartono Bambang. 2013. **Sistem Informasi Manajemen Berbasis komputer**. Bandung: Rineka Cipta.
- Jogiyanto. 2005. **Analisis dan desain sistem informasi**. Yogyakarta: Andi Offset.
- Kadir, Abdul. 2014. **Pengenalan Sistem Informasi**. Yogyakarta: Andi Yogyakarta.
- Kusbianto Deddy. 2010. **Analisis dan Perancangan Sistem Informasi**. Pasuruan: STMIK Yadika Bangil.
- Manullang, Harlet Gilbert. 2014. **Sistem Pendukung Keputusan Penentuan Dosen Terbaik dengan Metode AHP pada Universitas Methodist Indonesia**. Medan: USU Press.
- Murad, Dina Fitria. 2013. **Aplikasi Intelligence Website Untuk Penunjang Laporan Paud Pada Himpaudi Kota Tangerang**. Jurnal Vol.7 No.1 September 2013. STMIK Raharja Tangerang

- Myer, Thomas. 2008. **Professional Codeigniter**. Indianapolis : Wiley Publishing.
- Rosa AS, M. Shalahuddin. 2014. **Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek**. Bandung: Informatika.
- Simarmata, Janner. 2007. **Perancangan Basis Data**. Yogyakarta: Andi Yogyakarta.
- Sutabri Tata. 2014. **Analisis Sistem Informasi**. Yogyakarta: Andi Publisher.
- Widodo Prabowo Pujo, Herlawati. 2011. **Menggunakan UML (Unified Modelling Language)**. Bandung: Informatika.
- Welling, L., & Thomson, L. 2008. **PHP and MySQL WebDevelopment (4th ed)**. Upperside River: Addison-Wesley Professional.
- Yulianti. 2013. **Implementasi Arsitektur Client Server Dan MVC (Model View Controller) Untuk Membangun Aplikasi Administrasi Sekolah (Studi Kasus: SMK Averus Jakarta)**. Jakarta: Universitas Pemulang.
- Ceccaroni, Luigi. 2009. **Integration of a rule-based expert system, a case based reasoner and an ontological knowledge-base in the wastewater domain** : Universitat Politecnica de Catalunya.
- Turban, Efraim. 1995. **Decision Support and Expert System Management Support Systems Fourth Edition**. California : Prentice-Hall International Inc.
- Abdurrachman Irfandi M, Ade Romadhony, Siti Saadah. **Implementasi Sistem Pakar Diagnosa Penyakit Gigi Dan Mulut**. Indonesia Symposium On Computing 2015, p. 220, 2015.