PENGEMBANGAN SISTEM INFORMASI PENGGAJIAN DI PT. PERDANA INFORMATIKA

KERJA PERAKTEK

Diajukan untuk Memenuhi
Tugas Mata Kuliah Kerja Peraktek
Program Strata Satu Jurusan Teknik Informatika
Fakultas Teknik dan Ilmu Komputer
Universitas Komputer Indonesia

EKO KURNIAWAN KHANNEDY
10106031
FRIKO SIMANJUNTAK
10106022
HISYAM PAMBUDI
10106020

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS KOMPUTER INDONESIA
BANDUNG 2010

KATA PENGANTAR

Dengan memanjatkan puji dan syukur kehadirat Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan laporan kerja praktek ini dengan judul "**PENGEMBANGAN SISTEM INFORMASI PENGGAJIAN DI PT. PERDANA INFORMATIKA**".

Laporan ini merupakan hasil dari Kerja Praktek yang di lakukan oleh Penulis di PT. Perdana Informatika.Penyusunan laporan ini dilakukan untuk memenuhi mata kuliah wajib dan merupakan syarat kelulusan akademik pada program studi Teknik Informatika Fakultas dan Ilmu Komputer Universitas Komputer Indonesia.

Penulis menyadari jika laporan Kerja Praktek ini tidak akan selesai tanpa bantuan dan bimbingan dari dosen dan teman-teman khususnya :

1. Mira Kania Sabariah, selaku dosen pembimbing
2. Syarif Hidayat, selaku pembimbing lapangan
3. Teman-teman sekelas yang seperjuangan.

Penulis menyadari dengan segala keterbatasan ilmu dan kemampuan yang dimiliki, sehingga dalam penyusunannya ini masih terdapat banyak kekurangan, untuk itu dengan rendah hati penulis menerima kritik dan saran yang sifatnya membangun demi perbaikan – perbaikan dimasa mendatang.

Bandung, 23 januari 2010

Penulis

DAFTAR ISI

KATA PENGANTAR i

DAFTAR ISI iii

DAFTAR TABEL vi

DAFTAR GAMBAR ix

DAFTAR SIMBOL xv

DAFTAR LAMPIRAN xvi

BAB 1 PENDAHULUAN 1

1.1 Latar Belakang 1

1.2 Perumusan Masalah 2

1.3 Maksud dan Tujuan 3

1.3.1 Maksud 3

1.3.2 Tujuan 3

1.4 Batasan Masalah 4

1.5 Metodologi Penelitian 4

1.6 Sistematika Pelaporan Kerja Praktek 5

BAB 2 TINJAUAN PUSTAKA 7

2.1 Profil Tempat Kerja Praktek 7

2.1.1 Sejarah Instansi 7

2.2 Logo Instansi 8

2.3 Badan Hukum Instansi 9

2.4 Struktur Organisasi dan Job Description 9

2.4.1 Struktur Organisasi 9

2.4.2 Job Description 11

2.5 Landasan Teori 11

2.5.1 Java 11

2.5.2 NetBeans IDE 16

BAB 3 Pengembangan Sistem Penggajian 24

3.1 Deskripsi Global Perangkat Lunak 24

3.1.1 Perspektif Produk 24

3.1.2 Fungsi Produk 25

3.1.3 Karakteristik Pengguna 26

3.1.4 Batasan – Batasan 26

3.2 Deskripsi Rinci Kebutuhan Sistem Lama 26

3.2.1 Kebutuhan Antar Muka Eksternal 27

3.2.2 Entity Relational Diagram 28

3.2.3 Kebutuhan Fungsional 29

3.3 Deskripsi Rinci Kebutuhan Sistem Baru 50

3.3.1 Kebutuhan Antar Muka Eksternal 50

3.3.2 Kebutuhan Fungsional 51

3.3.3 Deskripsi Kelas – Kelas 80

3.3.4 Deskripsi Perancangan 108

3.3.5 Implementasi 135

3.3.6 Pengujian 137

3.3.7 Kesimpulan 156

BAB 4 KESIMPULAN DAN SARAN 157

4.1 Kesimpulan 157

4.2 Saran 157

DAFTAR PUSTAKA 159

DAFTAR TABEL

Tabel 3.1 Karakteristik Pengguna 26

Tabel 3.2 Skenario Autentikasi 53

Tabel 3.3 Skenario Olah Data Bahasa 54

Tabel 3.4 Skenario Olah Data Pendidikan 56

Tabel 3.5 Skenario Olah Data Hari Libur 58

Tabel 3.6 Skenario Olah Data Divisi 60

Tabel 3.7 Skenario Olah Data Jabatan 62

Tabel 3.8 Skenario Pengolahan Data Klien 64

Tabel 3.9 Skenario Olah Data Proyek 66

Tabel 3.10 Skenario Olah Data Pengguna 68

Tabel 3.11 Skenario Olah Data Hak Akses 70

Tabel 3.12 Skenario Olah Data Kehadiran 72

Tabel 3.13 Skenario Olah Data Transportasi 74

Tabel 3.14 Skenario Olah Data Karyawan 76

Tabel 3.15 Skenario Buat Laporan 78

Tabel 3.16 Deskripsi Atribut Kelas Autentikasi 81

Tabel 3.17 Deskripsi Atribut Kelas Divisi 83

Tabel 3.18 Deskripsi Atribut Kelas Jabatan 85

Tabel 3.19 Deskripsi Atribut Kelas Pengguna 87

Tabel 3.20 Deskripsi Atribut Kelas Hak Akses 89

Tabel 3.21 Deskripsi Atribut Kelas Kehadiran 92

Tabel 3.22 Deskripsi Atribut Klien 94

Tabel 3.23 Deskripsi Atribut Kelas Proyek 96
Tabel 3.24 Deskripsi Atribut Kelas Bahasa 98
Tabel 3.25 Deskripsi Atribut Kelas Pendidikan..... 100
Tabel 3.26 Deskripsi Atribut Kelas Transportasi..... 102
Tabel 3.27 Deskripsi Atribut Kelas Karyawan 105
Tabel 3.28 Deskripsi Atribut Kelas Laporan 107
Tabel 3.29 Deskripsi Objek Autentikasi 110
Tabel 3.30 Deskripsi Objek Olah Data Bahasa..... 113
Tabel 3.31 Deskripsi Objek Oleh Data Bahasa..... 114
Tabel 3.32 Deskripsi Objek Olah Data Divisi 116
Tabel 3.33 Deskripsi Objek Olah Data Jabatan 117
Tabel 3.34 Deskripsi Objek Olah Data Klien 119
Tabel 3.35 Deskripsi Objek Olah Data Proyek..... 120
Tabel 3.36 Deskripsi Objek Olah data Karyawan..... 123
Tabel 3.37 Deskripsi Objek Olah Data Kehadiran..... 125
Tabel 3.38 Deskripsi Objek Olah Data Kehadiran..... 127
Tabel 3.39 Deskripsi Objek Olah Data Pengguna 128
Tabel 3.40 Deskripsi Objek Olah Data Hak Akses..... 130
Tabel 3.41 Deskripsi Objek Buat Laporan..... 132
Tabel 3.42 Deskripsi Objek Kesalahan..... 133
Tabel 3.43 Deskripsi Objek Laporan Gaji Karyawan..... 134
Tabel 3.44 Deskripsi Objek Laporan Per Bulan 134
Tabel 3.45 Data Normal Autentikasi 138

Tabel 3.46 Data Salah Autentikasi 139
Tabel 3.47 Data Salah Autentikasi 139
Tabel 3.48 Data Normal Hari Libur 141
Tabel 3.49 Data Salah Hari Libur 142
Tabel 3.50 Data Normal Olah Data Pendidikan 143
Tabel 3.51 Data Salah Uji Coba Data Pendidikan 144
Tabel 3.52 Data Normal Olah Data Bahasa 145
Tabel 3.53 Data Normal Olah data Divisi 146
Tabel 3.54 Data Salah Olah Data Divisi 147
Tabel 3.55 Data Normal Olah Data Jabatan 148
Tabel 3.56 Data Salah Olah Data Jabatan 149
Tabel 3.57 Data Normal Olah Data Klien 150
Tabel 3.58 Data Salah Olah Data Klien 150
Tabel 3.59 Data Normal Olah Data Proyek 151
Tabel 3.60 Data Salah Olah Data Proyek 152
Tabel 3.61 Data Normal Olah Data Pengguna 153
Tabel 3.62 Data Salah Olah Data Pengguna 154
Tabel 3.63 Antarmuka Olah Data Karyawan 154
Tabel 3.64 Data Normal Olah Data Karyawan 155
Tabel 3.65 Data Salah Olah Data Karyawan 156

DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi 10
Gambar 3.1 Perpektif Produk 24
Gambar 3.2 Entity Relational Diagram 28
Gambar 3.3 Diagram Konteks 29
Gambar 3.4 Data Flow Diagram Level 1 30
Gambar 3.5 Data Flow Diagram Level 2 Autentikasi 33
Gambar 3.6 Data Flow Diagram Level 2 Olah Data Hari Libur 34
Gambar 3.7 Data Flow Diagram Level 2 Proses Olah Data Pendidikan 35
Gambar 3.8 Data Flow Diagram Level 2 Olah Data Divisi 37
Gambar 3.9 Data Flow Diagram Level 2 Olah Data Jabatan 38
Gambar 3.10 Data Flow Diagram Level 2 Olah Data Klien 39
Gambar 3.11 Data Flow Diagram Level 2 Olah Data Proyek 40
Gambar 3.12 Data Flow Diagram Level 2 Olah Data Lembur 41
Gambar 3.13 Data Flow Diagram Level 2 Olah Data Transportasi 43
Gambar 3.14 Data Flow Diagram Level 2 Buat Laporan 45
Gambar 3.15 Data Flow Diagram Level 2 Olah Data Pengguna 46
Gambar 3.16 Data Flow Diagram Level 2 Olah Data Hak Akses 47
Gambar 3.17 Data Flow Diagram Level 2 Olah Data Karyawan 48
Gambar 3.18 Data Flow Diagram Level 2 Olah Data Bahasa 49
Gambar 3.19 Usecases Sistem Penggajian 52
Gambar 3.20 Dinamika Objek Autentikasi 53
Gambar 3.21 Dinamika Objek Olah Data Bahasa 55

Gambar 3.22 Dinamika Objek Olah Data Pendidikan 57
Gambar 3.23 Dinamika Objek Olah Data Hari Libur 59
Gambar 3.24 Dinamika Objek Olah Data Divisi 61
Gambar 3.25 Dinamika Objek Olah Data Jabatan 63
Gambar 3.26 Dinamika Objek Olah Data Klien 65
Gambar 3.27 Dinamika Objek Olah Data Proyek 67
Gambar 3.28 Dinamika Objek Olah Data Pengguna 69
Gambar 3.29 Dinamika Objek Olah Data Hak Akses 71
Gambar 3.30 Dinamika Objek Olah Data Kehadiran 73
Gambar 3.31 Dinamika Objek Olah Data Transportasi 75
Gambar 3.32 Dinamika Objek Olah Data Karyawan 77
Gambar 3.33 Dinamika Objek Buat Laporan 79
Gambar 3.34 Diagram Kelas 80
Gambar 3.35 Kelas Autentikasi 81
Gambar 3.36 Kelas AutentikasiValidator 81
Gambar 3.37 Kelas AutentikasiDatabase 82
Gambar 3.38 Kelas Divisi 83
Gambar 3.39 Kelas DivisiValidator 83
Gambar 3.40 Kelas DivisiDatabase 84
Gambar 3.41 Kelas Jabatan 85
Gambar 3.42 Kelas JabatanValidator 86
Gambar 3.43 Kelas JabatanDatabase 86
Gambar 3.44 Kelas Pengguna 87

Gambar 3.45 Kelas PenggunaValidator 88
Gambar 3.46 Kelas PenggunaDatabase 88
Gambar 3.47 Kelas HakAkses 89
Gambar 3.48 Kelas HakAksesValidator 90
Gambar 3.49 Kelas HakAksesDatabase 90
Gambar 3.50 Kelas Kehadiran 91
Gambar 3.51 Kelas KehadiranValidator 92
Gambar 3.52 Kelas KehadiranDatabase 93
Gambar 3.53 Kelas Klien 93
Gambar 3.54 Kelas KlienValidator 94
Gambar 3.55 Kelas KlienDatabase 95
Gambar 3.56 Kelas Proyek 96
Gambar 3.57 Kelas ProyekValidator 96
Gambar 3.58 Kelas ProyekDatabase 97
Gambar 3.59 Kelas Bahasa 98
Gambar 3.60 Kelas BahasaValidator 98
Gambar 3.61 Kelas BahasaDatabase 99
Gambar 3.62 Kelas Pendidikan 100
Gambar 3.63 Kelas PendidikanValidator 100
Gambar 3.64 Kelas PendidikanDatabase 101
Gambar 3.65 Kelas Transportasi 102
Gambar 3.66 Kelas Transportasi 103
Gambar 3.67 Kelas TransportasiDatabase 103

Gambar 3.68 Kelas Karyawan 104
Gambar 3.69 Kelas KaryawanValidator 105
Gambar 3.70 Kelas KaryawanDatabase 106
Gambar 3.71 Kelas Laporan 106
Gambar 3.72 Kelas LaporanGenerator 107
Gambar 3.73 Kelas LaporanValidator 108
Gambar 3.74 Deskripsi Layar 109
Gambar 3.75 Deskripsi Layar Autentikasi 110
Gambar 3.76 Deskripsi Layar Olah Data Pendidikan 111
Gambar 3.77 Deskripsi Objek Olah Data Pendidikan 111
Gambar 3.78 Deskripsi Layar Olah Data Hari Libur 112
Gambar 3.79 Deskripsi Layar Olah Data Bahasa 114
Gambar 3.80 Deskripsi Layar Olah Data Divisi 115
Gambar 3.81 Deskripsi Layar Olah Data Jabatan 117
Gambar 3.82 Deskripsi Layar Olah Data Klien 118
Gambar 3.83 Deskripsi Layar Olah Data Proyek 120
Gambar 3.84 Deskripsi Layar Olah Data Karyawan 1 121
Gambar 3.85 Deskripsi Layar Olah Data Karyawan 2 122
Gambar 3.86 Deskripsi Layar Olah Data Karyawan 3 122
Gambar 3.87 Deskripsi Layar Olah Data Karyawan 4 123
Gambar 3.88 Deskripsi Layar Olah Data Kehadiran 125
Gambar 3.89 Deskripsi Layar Olah Data Transportasi 126
Gambar 3.90 Deskripsi Layar Olah Data Pengguna 128

Gambar 3.91 Deskripsi Layar Olah Data Hak Akses\ 129
Gambar 3.92 Deskripsi Layar Buat Laporan 1 131
Gambar 3.93 Deskripsi Layar Buat Laporan 2 131
Gambar 3.94 Deskripsi Layar Kesalahan 132
Gambar 3.95 Deskripsi Layar Laporan Gaji Karyawan 133
Gambar 3.96 Deskripsi Layar Laporan Gaji Bulanan..... 134
3.97 Jaringan Semantik 135
Gambar 3.98 Sistem Penggajian di Windows..... 137
Gambar 3.99 Sistem Penggajian di LinuxMint..... 137
Gambar 3.100 Tampilan Antar Muka Autentikasi..... 138
Gambar 3.101 Tampilan Validasi Gagal Autentikasi 139
Gambar 3.102 Pengguna Tidak Terdaftar..... 139
Gambar 3.103 Tampilan Antarmuka Olah Hari Libur..... 140
Gambar 3.104 Menambah Data Hari Libur 141
Gambar 3.105 Validasi Gagal Hari Libur..... 141
Gambar 3.106 Tampilan Antar Muka Olah Data Pendidikan..... 142
Gambar 3.107 Menambah Data Pendidikan 143
Gambar 3.108 Gagal Menambah Data Pendidikan..... 143
Gambar 3.109 Tampilan Olah Data Bahasa..... 144
Gambar 3.110 Menambah Data Bahasa..... 145
Gambar 3.111 Validasi Gagal Olah Data Bahasa 145
Gambar 3.112 Data Salah Olah Data Bahasa..... 146
Gambar 3.113 Antar Muka Olah Data Divisi 146

Gambar 3.114 Validasi Gagal Olah Data Divisi 147
Gambar 3.115 Antarmuka Olah Data Jabatan..... 148
Gambar 3.116 Validasi Gagal Olah Data Jabatan..... 148
Gambar 3.117 Antarmuka Olah Data Klien..... 149
Gambar 3.118 Validasi Gagal Olah Data Klien..... 150
Gambar 3.119 Antarmuka Olah Data Proyek 151
Gambar 3.120 Validasi Gagal Olah Data Proyek 152
Gambar 3.121 Antarmuka Olah Data Pengguna..... 153
Gambar 3.122 Validasi Gagak Olah Data Pengguna 153
Gambar 3.123 Data Salah di Pengolahan Data Karyawan..... 155
Gambar 3.124 Validasi Gagal Olah Data Karyawan 156

DAFTAR SIMBOL

DAFTAR LAMPIRAN

BAB 1

PENDAHULUAN

1.1 Latar Belakang

PT. Perdana Informatika merupakan perusahaan yang bergerak di bidang *Information Technology*.PT. Perdana Informatika menangani pembuatan perangkat lunak, jaringan komputer, pelatihan IT, *outsourcing* pegawai dan juga konsultasi di bidang IT.Sebelumnya PT. Perdana Informatika adalah perusahaan yang bergerak dibidang Konsultan Akuntansi, dan semenjak tahun 2008 mulai masuk ke dunia IT.

PT.Perdana Informatika memiliki banyak klien dari berbagai kalangan.Salah satunya adalah perusahaan yang bergerak di bidang kontraktor alat-alat berat.Klien perusahaan PT. Perdana Informatika tersebut meminta untuk dibuatkan sebuah sistem penggajian.

PT. Perdana Informatika telah memiliki aplikasi sistem penggajian.Namun aplikasi sistem penggajian yang dimiliki oleh PT. Perdana Informatika tersebut belum 100% selesai. Selain itu aplikasi sistem penggajian yang dimiliki oleh PT. Perdana Informatika hanya dapat berjalan pada satu jenis sistem operasi, sehingga aplikasi sistem penggajian yang dimiliki PT. Perdana Informatika tersebut tidak digunakan oleh klien yang memiliki sistem operasi yang berbeda.

Permasalahan selanjutnya adalah, aplikasi sistem penggajian yang dimiliki oleh PT. Perdana Informatika dibuat dengan hak akses yang telah ditentukan

sebelumnya.Artinya hak akses pengguna tidak dapat berubah secara dinamis dan tidak dapat diubah walaupun oleh Administrator aplikasi sistem penggajian tersebut.Hal ini menyebabkan perusahaan klien harus mengikuti aturan aplikasi sistem penggajian tersebut, sedangkan setiap perusahaan memiliki kebijakan yang berbeda terhadap hak akses setiap pengguna.

Permasalahan terakhir adalah aplikasi sistem penggajian milik PT. Perdana Informatika hanya dikhususkan untuk satu jenis *database*. Hal ini mengakibatkan aplikasi sistem penggajian tersebut hanya dapat didistribusikan untuk klien yang menggunakan *database* yang sama, sehingga klien yang menggunakan *database* yang berbeda tidak dapat menggunakan aplikasi sistem penggajian tersebut.

Dari permasalahan tersebut, PT. Perdana Informatika memerlukan aplikasi sistem penggajian yang dapat berjalan pada semua sistem operasi, memiliki fasilitas pengolahan hak akses yang dinamis dan dapat menggunakan database yang berbeda.Oleh sebab itu aplikasi sistem penggajian milik PT. Perdana Informatika tersebut perlu dikembangkan.

1.2 Perumusan Masalah

Bagaimana mengembangkan aplikasi sistem penggajian milik PT. Perdana Informatika agar dapat berjalan di semua sistem operasi dan memiliki fasilitas pengolahan hak akses secara dinamis.

1.3 Maksud dan Tujuan

1.3.1 Maksud

Mengembangkan aplikasi sistem penggajian milik PT. Perdana Informatika.

1.3.2 Tujuan

Adapun tujuan pengembangan aplikasi sistem penggajian milik PT. Perdana Informatika adalah :

a. Agar aplikasi sistem penggajian tersebut dapat berjalan di seluruh sistem operasi, sehingga tidak dibutuhkan perubahan atau konfigurasi ulang saat aplikasi sistem penggajian tersebut dipindahkan dari satu sistem operasi ke sistem operasi yang berbeda.
b. Agar aplikasi sistem penggajian tersebut dapat memiliki fasilitas pengolahan data hak akses secara dinamis, sehingga klien dapat dengan mudah menambah atau mengubah hak akses untuk user sesuai dengan kebijakan perusahaan klien.
c. Agar aplikasi sistem penggajian tersebut dapat dijalankan menggunakan *database*yang berbeda-beda. Sehingga aplikasi sistem penggajian tersebut dapat menyesuaikan dengan database yang biasa digunakan oleh klien.

1.4 Batasan Masalah

Adapun batasan masalah pengembangan aplikasi sistem penggajian milik PT. Perdana Informatika adalah :

a. Teknologi yang digunakan untuk menggembangkan aplikasi sistem penggajian tersebut harus free dan opensource agar biaya pengembangan dapat ditekan.
b. Aplikasi sistem penggajian harus berbasis aplikasi *desktop*.
c. Aplikasi dibangun menggunakan bahasa pemrograman Java.
d. Aplikasi dibangun dengan bantuan teknologi Hibernate Framework, agar aplikasi tersebut dapat berjalan di berbagai jenis *database*.
e. Aplikasi dibangun menggunakan NetBeans IDE (*Integrated Development Environtment*).

1.5 Metodologi Penelitian

Dalam pengembangan aplikasi sistem penggajian milik PT. Perdana Informatika, penulis menggunakan metode penelitian sebagai berikut :

a. Survey

   Survey dilakukan untuk mengetahui fasilitas apa saja yang diinginkan oleh PT. Perdana Informatika terhadap sistem penggajian yang akan dikembangan. Dalam survey ini penulis menggunakan wawancara sebagai cara untuk mendapatkan informati kebutuhan tersebut.

b. Analisis

Untuk mencapai kebutuhan pengguna, perlu dilakukan analisis tentang bagaimana sebaiknya sistem penggajian yang baru akan dikembangkan. Sehingga diharapkan sistem penggajian yang baru dapat sesuai dengan kebutuhan pengguna.

c. Perancangan Produk

Setelah dilakukan tahap analisis, maka tahap selainjutnya adalah perancangan produk.Dengan adanya kebutuhan pengguna, maka pengembangan sistem penggajian dapat dilakukan dengan mengacu pada hasil analisis.

d. Pembangunan Produk

Dari hasl perancangan produk, kemudian digunakan untuk mengembangkan sistem penggajian.

1.6 Sistematika Pelaporan Kerja Praktek

Sistematika yang digunakan dalam pelaporan kerja praktek ini adalah :

BAB 1 PENDAHULUAN

Pada bab ini penulis membahas tentang latar belakang kerja praktek, maksud dan tujuan kerja praktek, sistem pelaksanaan kerja praktek dan sistematika pelaporan kerja praktek.

BAB 2 TINJAUAN PUSTAKA

Pada bab ini penulis menjabarkan secara singkat mengenai sejarah perusahaan tempat penulis kerja praktek, kedudukan perusahaan, bentuk dan

badan hokum perusahaan, dibang pekerjaan perusahaan, bidang pekerjaan divisi/departemen tempat kerja praktek, dan struktur organisasi perusahaan.

BAB 3 PEMBAHASAN

Pada bab ini penulis menjabarkan tentang jadwal kerja praktek yang dilakukan oleh penulis. Selain itu penulis menjabarkan tentang cara dan teknik yang dilakukan penulis selama melakukan kerja praktek dan data hasil selama kerja praktek yang dilakukan oleh penulis.

BAB 4 KESIMPULAN DAN SARAN

Pada bab ini berisi kesimpulan selama penulis melakukan kerja praktek di perusahaan serta saran yang mungkin berguna bagi param pembaca berhubungan dengan hasil kerja praktek yang dibuat.

BAB 2

TINJAUAN PUSTAKA

2.1 Profil Tempat Kerja Praktek

2.1.1 Sejarah Instansi

PT. Perdana Informatika didirikan pada 15 April 2008 sejak didirikan perusahaan ini mengalami perubahaan yang signifikan terutama dalam bidang usaha, awalnya perusahaan ini hanya menerima pemesanaan sistem informasi namun karena dianggap perlu mengembangkan kiprahnya dibidang IT, maka PT. Perdana Informatika menambah usahanya menjadi Konsultan IT dan juga bergerak dibidang jaringan (*Networking*), pelatihan dan outsourcing pegawai. Pertama kali didirikan perusahaan ini hanya memiliki sebuah ruangan kerja, hingga kini perusahaan ini sudah menyewa 2 tingkat ruko di Ruko Mas Bekasi.Peningkatan ini disebabkan oleh makin banyaknya jumlah klien yang dimiliki oleh perusahaan ini.

Adapun tujuan PT. Perdana Informatika tersebut berdiri adalah :

a. Melakukan perdagangan komputer dan peripehal , peralatan telekomunikasi pada umumnya baik perangkat keras maupun perangkat lunak, peralatan informatika dan multimedia, baik perangkat keras (*hardware*) maupun perangkat lunak (*software*) serta alat-alat informati dan komunikasi lainnya.

b. Jasa penyediaan dan penyewaan perangkat keras computer dan peripehal dan usaha terkait lainnya.
c. Jasa pendidikan, kursus, pelatihan dan keterampilan dalam bidang komputer dan informatika pada umumnya, termasuk menyediakan sarana dan prasarana pendidikan terkait.
d. Jasa telekomunikasi umum meliputi wartel, warnet dan termasuk kegiatan usaha instalasi serta perawatan dan perbaikan dan kegiatan usaha terkait.
e. Jasa instalasi dan pemeliharaan komputer, jaringan komputer, serta perangkat penunjang lainnya.
f. Jasa pembuatan dan pengembangan perangkat lunak (*software*) yang meliputi perancanaan dan perancangan sistem, pembuatan aplikasi, *feasibility study*, otomasi sistem perkantoran, penelitian dan pemasangan serta pemeliharaan sistem, termasuk modifikasi dan pengembangan sistem-sistem, serta usaha lain terkait.
g. Jasa konsultasi bidang komputer dan rakayasa informatika.

2.2 Logo Instansi

PERDANA
INFORMATIKA

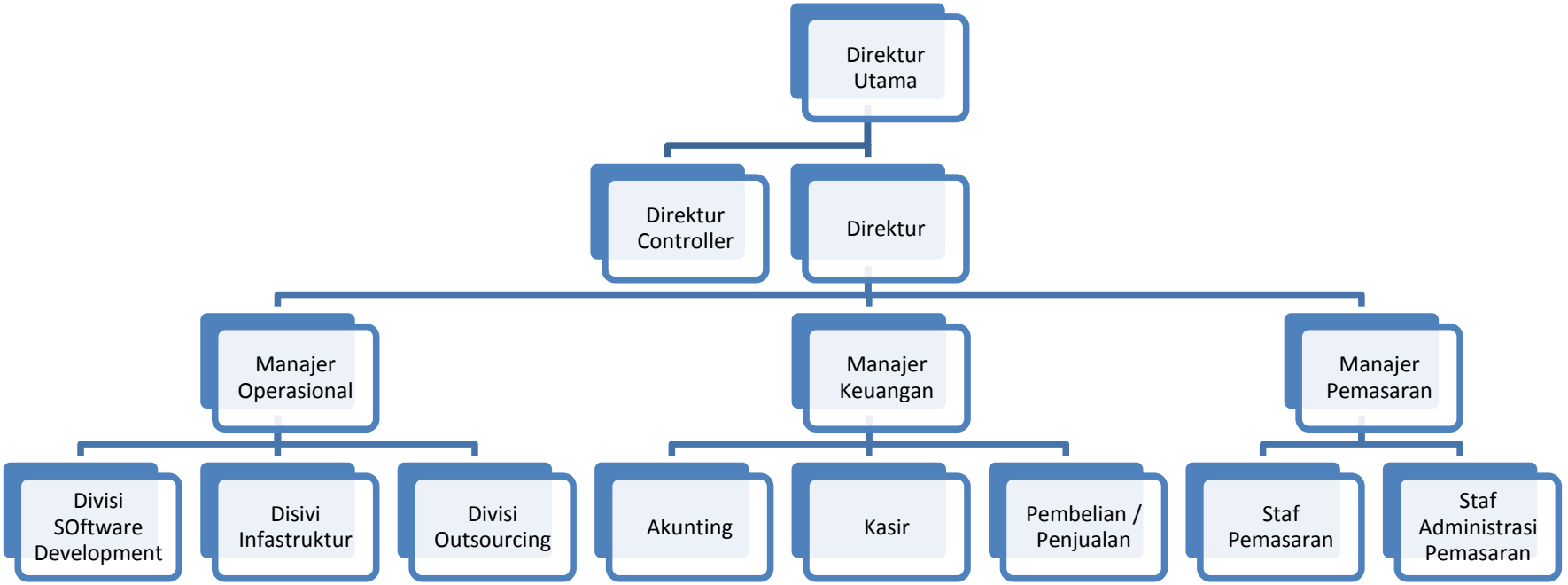
2.3 Badan Hukum Instansi

PT. Perdana Informatika merupakan perusahaan persero swasta yang ditetapkan berdasarkan keputusan Departemen Kehakiman Pusat Jakarta pada tahun 2008.

2.4 Struktur Organisasi dan Job Description

2.4.1 Struktur Organisasi

PT. Perdana Informatika dipimpin oleh seorang direktur Utama dan dikordinir oleh serorang direktur, dan direktur controller sebagai pengawas.Bagian terpenting dari struktur organisasi ini adalah di manajerial Operasional karena semua pelaksanaan pelayanaan dilakukan oleh manajerial ini.



Gambar 2.1 Struktur Organisasi

2.4.2 Job Description

Pembagian pekerjaan (*job description*) terbagi menjadi tiga divisi penting yaitu :

1. Divisi Software Development :
    a. Pesanan
    b. Produk Massal
2. Divisi Struktur :
    a. Jaringan
    b. Server
3. Divisi OutSourcing :
    a. Tenaga Ahli IT
    b. System Analist
    c. Implementasi IT

2.5 Landasan Teori

2.5.1 Java

Java' adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam.Dikembangkan oleh Sun Microsystems dan diterbitkan tahun 1995.Java tidak boleh disalahpahami sebagai JavaScript.JavaScript adalah bahasa scripting yang digunakan oleh *webbrowser*.

2.5.1.1 Sejarah Perkembangan Java

Bahasa pemrograman Java pertama lahir dari The Green Project, yang berjalan selama 18 bulan, dari awal tahun 1991 hingga musim panas 1992. Proyek tersebut belum menggunakan versi yang dinamakan Oak.Proyek ini dimotori oleh

Patrick Naughton, Mike Sheridan, James Gosling dan Bill Joy, beserta sembilan pemrogram lainnya dari Sun Microsystems. Salah satu hasil proyek ini adalah maskot Duke yang dibuat oleh Joe Palrang.

Pertemuan proyek berlangsung di sebuah gedung perkantoran Sand Hill Road di Menlo Park. Sekitar musim panas 1992 proyek ini ditutup dengan menghasilkan sebuah program Java Oak pertama, yang ditujukan sebagai pengendali sebuah peralatan dengan teknologi layar sentuh (touch screen), seperti pada PDA sekarang ini. Teknologi baru ini dinamai "*7" (Star Seven).

Setelah era Star Seven selesai, sebuah anak perusahaan TV kabel tertarik ditambah beberapa orang dari proyek The Green Project. Mereka memusatkan kegiatannya pada sebuah ruangan kantor di 100 Hamilton Avenue, Palo Alto.

Perusahaan baru ini bertambah maju: jumlah karyawan meningkat dalam waktu singkat dari 13 menjadi 70 orang. Pada rentang waktu ini juga ditetapkan pemakaian Internet sebagai medium yang menjembatani kerja dan ide di antara mereka.Pada awal tahun 1990-an, *Internet* masih merupakan rintisan, yang dipakai hanya di kalangan akademisi dan militer.

Mereka menjadikan perambah (*browser*) Mosaic sebagai landasan awal untuk membuat perambah Java pertama yang dinamai Web Runner, terinsipirasi dari film 1980-an, Blade Runner. Pada perkembangan rilis pertama, Web Runner berganti nama menjadi Hot Java.

Pada sekitar bulan Maret 1995, untuk pertama kali kode sumber Java versi 1.0a2 dibuka. Kesuksesan mereka diikuti dengan untuk pemberitaan pertama kali pada surat kabar San Jose Mercury News pada tanggal 23 Mei 1995.

Sayang terjadi perpecahan di antara mereka suatu hari pada pukul 04.00 di sebuah ruangan hotel Sheraton Palace.Tiga dari pimpinan utama proyek, Eric Schmidt dan George Paolini dari Sun Microsystems bersama Marc Andreessen, membentuk Netscape.

Nama Oak, diambil dari pohon oak yang tumbuh di depan jendela ruangan kerja "bapak java", James Gosling. Nama Oak ini tidak dipakai untuk versi *release* Java karena sebuah perangkat lunak sudah terdaftar dengan merek dagang tersebut, sehingga diambil nama penggantinya menjadi "Java". Nama ini diambil dari kopi murni yang digiling langsung dari biji (kopi tubruk) kesukaan Gosling.

2.5.1.2 Versi Awal Java

Versi awal Java ditahun 1996 sudah merupakan versi *release* sehingga dinamakan Java Versi 1.0. Java versi ini menyertakan banyak paket standar awal yang terus dikembangkan pada versi selanjutnya:

1. java.lang: Peruntukan kelas elemen-elemen dasar.
2. java.io: Peruntukan kelas input dan output, termasuk penggunaan berkas.
3. java.util: Peruntukan kelas pelengkap seperti kelas struktur data dan kelas kelas penanggalan.
4. java.net: Peruntukan kelas TCP/IP, yang memungkinkan berkomunikasi dengan komputer lain menggunakan jaringan TCP/IP.
5. java.awt: Kelas dasar untuk aplikasi antarmuka dengan pengguna (GUI)
6. java.applet: Kelas dasar aplikasi antar muka untuk diterapkan pada penjelajah web.

2.5.1.3 Kelebihan Java

1. Multiplatform. Kelebihan utama dari Java ialah dapat dijalankan di beberapa platform / sistem operasi komputer, sesuai dengan prinsip tulis sekali, jalankan di mana saja. Dengan kelebihan ini pemrogram cukup menulis sebuah program Java dan dikompilasi (diubah, dari bahasa yang dimengerti manusia menjadi bahasa mesin / *bytecode*) sekali lalu hasilnya dapat dijalankan di atas beberapa *platform* tanpa perubahan. Kelebihan ini memungkinkan sebuah program berbasis java dikerjakan diatas operating system Linux tetapi dijalankan dengan baik di atas Microsoft Windows. *Platform* yang didukung sampai saat ini adalah Microsoft Windows, Linux, Mac OS dan Sun Solaris. Penyebanya adalah setiap sistem operasi menggunakan programnya sendiri-sendiri (yang dapat diunduh dari situs Java) untuk meninterpretasikan bytecode tersebut.
2. OOP (*ObjectOrientedProgramming* - Pemrogram Berorientasi Objek) yang artinya semua aspek yang terdapat di Java adalah Objek. Java merupakan salah satu bahasa pemrograman berbasis objek secara murni. Semua tipe data diturunkan dari kelas dasar yang disebut *Object*. Hal ini sangat memudahkan pemrogram untuk mendesain, membuat, mengembangkan dan mengalokasi kesalahan sebuah program dengan basis Java secara cepat, tepat, mudah dan terorganisir. Kelebihan ini menjadikan Java sebagai salah satu bahasa pemograman termudah, bahkan untuk fungsi fungsi yang advance seperti komunikasi antara komputer sekalipun.
3. Perpustakaan Kelas Yang Lengkap, Java terkenal dengan kelengkapan *library*/perpustakaan (kumpulan program program yang disertakan dalam

pemrograman java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya. Kelengkapan perpustakaan ini ditambah dengan keberadaan komunitas Java yang besar yang terus menerus membuat perpustakaan-perpustakaan baru untuk melingkupi seluruh kebutuhan pembangunan aplikasi.

4. Bergaya C++, memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java. Universitas-universitas di Amerika Serikat juga mulai berpindah dengan mengajarkan Java kepada murid-murid yang baru karena lebih mudah dipahami oleh murid dan dapat berguna juga bagi mereka yang bukan mengambil jurusan komputer.
5. Pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung (seperti halnya dalam bahasa C++ yang dipakai secara luas).

2.5.1.4 Kekurangan Java

1. Tulis sekali, perbaiki di mana saja - Masih ada beberapa hal yang tidak kompatibel antara *platform* satu dengan *platform* lain. Untuk J2SE, misalnya SWT-AWT bridge yang sampai sekarang tidak berfungsi pada Mac OS X.
2. Mudah didekompilasi. Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan *bytecode* yang menyimpan banyak atribut bahasa tingkat

tinggi, seperti nama-nama kelas, metode, dan tipe data. Hal yang sama juga terjadi pada Microsoft .NET Platform. Dengan demikian, algoritma yang digunakan program akan lebih sulit disembunyikan dan mudah dibajak/direverse-engineer.

3. Penggunaan memori yang banyak. Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ dan Pascal (lebih spesifik lagi, Delphi dan Object Pascal). Biasanya ini bukan merupakan masalah bagi pihak yang menggunakan teknologi terbaru (karena trend memori terpasang makin murah), tetapi menjadi masalah bagi mereka yang masih harus berkutat dengan mesin komputer berumur lebih dari 4 tahun.

2.5.2 NetBeans IDE

NetBeans mengacu pada dua hal, yakni *platform* untuk pengembangan aplikasi desktop java, dan sebuah*Integrated Development Environment* (IDE) yang dibangun menggunakan *platform* NetBeans.

Platform NetBeans memungkinkan aplikasi dibangun dari sekumpulan komponen perangkat lunak moduler yang disebut ‘modul’. Sebuah modul adalah suatu arsip Java (Java *archive*) yang memuat kelas-kelas Java untuk berinetraksi dengan NetBeans Open API dan file manifestasi yang mengidentifikasinya sebagai modul. Aplikasi yang dibangun dengan modul-modul dapat dikembangkan dengan menambahkan modul-modul baru.Karena modul dapat dikembangkan secara independen, aplikasi berbasis platform NetBeans dapat dengan mudah dikembangkan oleh pihak ketiga secara mudah dan *powerful.*

2.5.2.1 Sejarah

Pengembangan NetBeans diawali dari Xelfi, sebuah proyek mahasiswa tahun 1997 di bawah bimbingan Fakultas Matematika dan Fisika Universitas Charles, Praha.Sebuah perusahaan kemudian dibentuk untuk proyek tersebut dan menghasilkan versi komersial NetBeans IDE hingga kemudian dibeli oleh Sun Microsystem pada tahun 1999.Sun kemudian menjadikan NetBeans open source pada bulan Juni tahun 2000.Sejak itu komunitas NetBeans terus berkembang.

2.5.2.2 Platform NetBeans

Platform NetBeans adalah *framework* yang dapat digunakan kembali (*reusable*) untuk menyederhanakan pengembangan aplikasi *desktop*. Ketika aplikasi berbasis *platform* NetBeans dijalankan, kelas Main dari *platform* dieksekusi.Modul-modul yang tersedia ditempatkan di sebuah *registry* di dalam memori, dan tugas startup modul dijalankan. Secara umum, kode modul dimuatkan ke dalam memori hanya ketika ia diperlukan.Aplikasi dapat menginstal modul secara dinamis. Aplikasi dapat memasukkan modul *Update Center* untuk mengijinkan pengguna aplikasi men-*download digitally-signed upgrade* dan fitur-fitur baru secara langsung ke dalam aplikasi yang berjalan. Penginstalan kembali sebuah upgrade atau rilis baru tidak memaksa pengguna untuk men-download keseluruhan aplikasi lagi.

Platform NetBeans menawarkan layanan-layanan yang umum bagi aplikasi desktop, mengijinkan pengembang untuk fokus ke logika yang spesifik terhadap aplikasi. Fitur-fitur yang disediakan oleh platform NetBeans:

a. Manajemen antarmuka (misal: menu & toolbar)
b. Manajemen pengaturan pengguna
c. Manajemen penyimpanan (menyimpan dan membuka berbagai macam data)
d. Manajemen jendela
e. Wizard framework (mendukung dialog langkah demi langkah)

2.5.2.3 NetBeans IDE

NetBeans IDE adalah IDE *open source* yang ditulis sepenuhnya dengan bahasa pemrograman Java menggunakan platform NetBeans. NetBeans IDE mendukung pengembangan semua tipe aplikasi Java (J2SE, web, EJB, dan aplikasi mobile). Fitur lainnya adalah sistem proyek berbasis Ant, kontrol versi, dan *refactoring*.Versi terbaru saat ini adalah NetBeans IDE 5.5.1 yang dirilis Mei 2007 mengembangkan fitur-fitur Java EE yang sudah ada (termasuk Java Persistence support, EJB-3 dan JAX-WS). Sementara paket tambahannya, NetBeans Enterprise Pack mendukung pengembangan aplikasi perusahaan Java EE 5, meliputi alat desain visual SOA, skema XML, web service dan pemodelan UML. NetBeans C/C++ Pack mendukung proyek C/C++.

Modularitas: Semua fungsi IDE disediakan oleh modul-modul. Tiap modul menyediakan fungsi yang didefinisikan dengan baik, seperti dukungan untuk bahasa pemrograman Java, editing, atau dukungan bagi CVS. NetBeans memuat semua modul yang diperlukan dalam pengembangan Java dalam sekali download, memungkinkan pengguna untuk mulai bekerja sesegera mungkin. Modul-modul juga mengijinkan NetBeans untuk bisa dikembangkan.Fitur-fitur

baru, seperti dukungan untuk bahasa pemrograman lain, dapat ditambahkan dengan menginstal modul tambahan.Sebagai contoh, Sun Studio, Sun Java Studio Enterprise, dan Sun Java Studio Creator dari Sun Microsystem semuanya berbasis NetBeans IDE.

2.5.2.4 Lisensi

Sejak Juli 2006, NetBeans IDE dilisensikan di bawah *Common Development and Distribution License* (CDDL), yaitu lisensi yang berbasis *Mozilla Public License* (MPL).

2.5.2.5 Paket-Paket Tambahan NetBeans IDE

2.5.2.5.1 NetBeans Mobility Pack

NetBeans Mobility Pack adalah alat untuk mengembangkan aplikasi yang berjalan pada perangkat bergerak (mobile), umumnya telepon seluler, tetapi juga mencakup PDA, dan lain-lain.

NetBeans Mobility Pack dapat digunakan untuk menulis, menguji, dan debugging aplikasi untuk perangkat bergerak yang menggunakan teknologi berplatform Java Micro Edition (platform Java ME).Paket ini mengintegrasikan dukungan terhadap Mobile Information Device Profile (MIDP) 2.0, Connected Limited Device Configuration (CLDC) 1.1, dan Connected Device Configuration (CDC).Emulator dari pihak ketiga dapat diintegrasikan dengan mudah untuk lingkungan pengujian yang lebih kokoh.NetBeans Mobility Pack saat ini tersedia dalam dua klaster yang berbeda, yang satu memuat CDC dan yang lainnya CLDC.

2.5.2.5.2 NetBeans Profiler

NetBeans Profiler adalah alat untuk mengoptimalkan aplikasi Java, membantu menemukan kebocoran memori dan mengoptimalkan kecepatan.

Profiler ini berdasarkan sebuah proyek riset Sun Laboratories yang dahulu bernama Jfluid. Riset tersebut mengungkap teknik tertentu yang dapat digunakan untuk menurunkan overhead proses profiling aplikasi Java. Salah satu dari teknik tersebut adalah instrumentas i kode byte dinamis, yang berguna untuk profiling aplikasi Java yang besar. Dengan menggunakan instrumentasi kode byte dinamis dan algoritma-algoritma tambahan, Netbeans Profiler mampu mendapatkan informasi runtime aplikasi yang terlalu besar atau kompleks bagi profiler lain. NetBeans IDE 6.0 akan mendukung Profiling Point yang memungkinkan kita memprofilkan titik yang tepat dari eksekusi dan mengukur waktu eksekusi.

2.5.2.5.3 NetBeans C/C++ Pack

NetBeans C/C++ Pack menambahkan dukungan terhadap pengembang C/C++ ke NetBeans IDE 5.5.Paket ini memperbolehkan pengembang menggunakan sekumpulan kompiler dan alat sendiri bersama dengan NetBeans IDE untuk membangun aplikasi native untuk MS Windows, Linux, dan Solaris.Paket ini membuat editor mengenali bahasa C/C++ dan menyediakan project template, browser kelas yang dinamis, dukungan pembuatan file dan fungsionalitas debugger.Para pengembang juga dapat mengembangkan paket tersebut dengan fungsionalitas tambahan mereka sendiri.

2.5.2.5.4 NetBeans Enterprise Pack

NetBeans Enterprise Pack memperluas dukungan terhadap pengembangan aplikasi perusahaan dan web service di NetBeans IDE 5.5.

Enterprise Pack ini mengembangkan kemampuan untuk menulis, menguji, dan debug aplikasi dengan arsitektur berorientasi layanan (Service-Oriented Architecture) menggunakan XML, BPEL, dan Java web service. Paket ini menambahkan alat desain visual untuk pemodelan UML, skema XML, dan web service orchestration, juga dukungan untuk web service dengan menggunakan identitas yang aman.Paket ini juga menginstal dan mengkonfigurasi runtime yang diperlukan, termasuk mesin BPEL dan server manajemen identitas yang terintegrasi dengan Sun Java System Application Server.

2.5.2.5.5 NetBeans Ruby Pack

Versi NetBeans 6.0 mendatang akan mengijinkan pengembangan IDE menggunakan Ruby dan Jruby, sebagaimana Rails untuk dua implementasi Ruby yang lain. Preview NetBeans Ruby Pack tersedia sejak rilis Milestone 7 NetBeans 6.

a. Ruby Pack memasukkan fungsionalitas editor seperti:
b. pengeditan dasar
c. pewarnaan sintaks untuk Ruby
d. pelengkapan kode
e. occurence highlighting
f. pop-up dokumentasi yang terintegrasi untuk pemanggilan Ruby API

g. analisis semantik dengan highlighting parameter dan variabel lokal yang tidak terpakai

2.5.2.5.6 NetBeans JavaScript Editor

NetBeans JavaScript Editor menyediakan perluasan dukungan terhadap JavaScript dan CSS. Fitur-fiturnya antara lain:

a. Editor JavaScript
b. *syntaxhighlighting*
c. pelengkapan kode untuk objek dan fungsi *native*
d. semua fitur dalam editor NetBeans
e. pembuatan kerangka kelas JavaScript secara otomatis
f. pembuatan pemanggilan AJAX dari *template*
g. Ekstensi editor CSS
h. pelengkapan kode untuk nama-nama *style*
i. navigasi cepat melalui panel *navigator*
j. penampilan deklarasi aturan CSS di List View
k. penampilan struktur file di Tree View
l. mengurutkan outline view berdasarkan nama, tipe, atau urutan deklarasi (List & Tree)
m. pembuatan deklarasi aturan (hanya Tree)
n. pemfaktoran kembali sebagian nama rule (hanya Tree)

2.5.2.5.7 Lokalisasi

Sejak Januari 2007, paket bahasa Cina yang disederhanakan, Cina tradisional, Jerman, Jepang, Portugal (Brazil) dan Spanyol telah tersedia untuk NetBeans 5.5.1. Sementara itu, paket bahasa Albania, Azerbaijan, Ceska, Belanda, Yunani, Indonesia, Italia, dan Swedia sedang dalam proses pengerjaan.

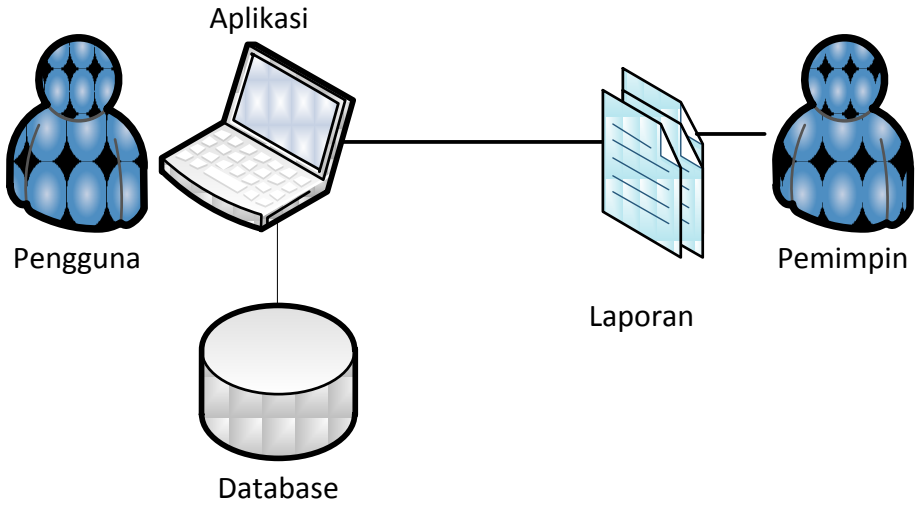
BAB 3

Pengembangan Sistem Penggajian

3.1 Deskripsi Global Perangkat Lunak

3.1.1 Perspektif Produk

Produk yang dikembangkan merupakan perangkat lunak berbasis *desktop*yang berjalan menggunakan *database* sebagai media penyimpanan datanya.Perangkat lunak ini digunakan sebagai pengolah sistem penggajian yang hanya dapat digunakan oleh satu pengguna pada waktu yang bersamaan. Sistem penggajian ini akan menghasilkan hasil akhir yang berupa laporan penggajian. Laporan penggajian tersebut untuk pemimpin perusahaan.



Gambar 3.1 Perpektif Produk

3.1.2 Fungsi Produk

Secara umum aplikasi ini adalah aplikasi yang berfungsi sebagai pengolah data pengajian.Sehingga pengguna dimudahkan dalam mengolah data penggajian dan pembuatan laporan penggajian dengan menggunakan aplikasi ini.

Fungsi Utama:

1. Autentikasi Pengguna
2. Pengolahan Data Hari Libur
3. Pengolahan Data Pendidikan
4. Pengolahan Data Bahasa
5. Pengolahan Data Divisi
6. Pengolahan Data Jabatan
7. Pengolahan Data Karyawan
8. Pengolahan Data Klien
9. Pengolahan Data Proyek
10. Pengolahan Data Hak Akses
11. Pengolahan Data Pengguna
12. Pengolahan Data Transportasi
13. Pengolahan Data Kehadiran
14. Pengolahan Data Lembur
15. Pembuatan Laporan

3.1.3 Karakteristik Pengguna

Pengguna aplikasi penggajian ini adalah administrator.Tugas administrator tersebut sebagai pengelola sistem penggajian.

Pengguna	Tanggung Jawab	Hak Akses	Tingkat Pendidikan	Tingkat Keterampilan	Pengalaman	Jenis Pelatihan
Administrator	Mengelola Data Sistem Penggajian	Hak Akses Penuh	S1	-	-	Pelatihan penggunaan sistem penggajian ini.

Tabel 3.1 Karakteristik Pengguna

3.1.4 Batasan – Batasan

Batasan – batasan yang digunakan pada pengembangan perangkat lunak ini adalah :

1. Sistem penggajian ini harus dapat berjalan pada berbagai sistem operasi tanpa melakukan kompilasi ulang.
2. Sistem penggajian ini harus dapat menggunakan seluruh jenis *database* tanpa harus merubah sistem.
3. Sistem penggajian ini harus dapat menambah pengguna dan hak akses yang berbeda sesuai dengan keinginan administrator.
4. Sistem penggajian ini harus berupa aplikasi *desktop*.

3.2 Deskripsi Rinci Kebutuhan Sistem Lama

Bagian ini berisikan semua kebutuhan perangkat lunak pada sistem yang lama dan diuraikan secara rinci untuk keperluan pengembangan ke sistem yang

baru. Kebutuhan perangkat lunak tersebut meliputi kebutuhan antarmuka eksternal,deskripsi data, kebutuhan fungsional dan non-fungsional.

3.2.1 Kebutuhan Antar Muka Eksternal

3.2.1.1 Antarmuka Pemakai

Aplikasi sistem penggajian ini merupakan aplikasi berbasis *desktop* dan *database*, sehingga untuk menggunakan aplikasi ini pengguna memerlukan komputer dan *database management system*yang telah terinstall pada komputernya.Pada aplikasi pengguna dapat mengolah data penggajian melalui menu-menu yang telah terdapat pada aplikasi.

3.2.1.2 Antarmuka Perangkat Keras

Aplikasi sistem penggajian ini merupakan aplikasi *standalone* yaitu aplikasi yang berdisi sendiri, sehingga pengguna hanya memerlukan komputer saja untuk menjalankannya.

3.2.1.3 Antarmuka Perangkat Lunak

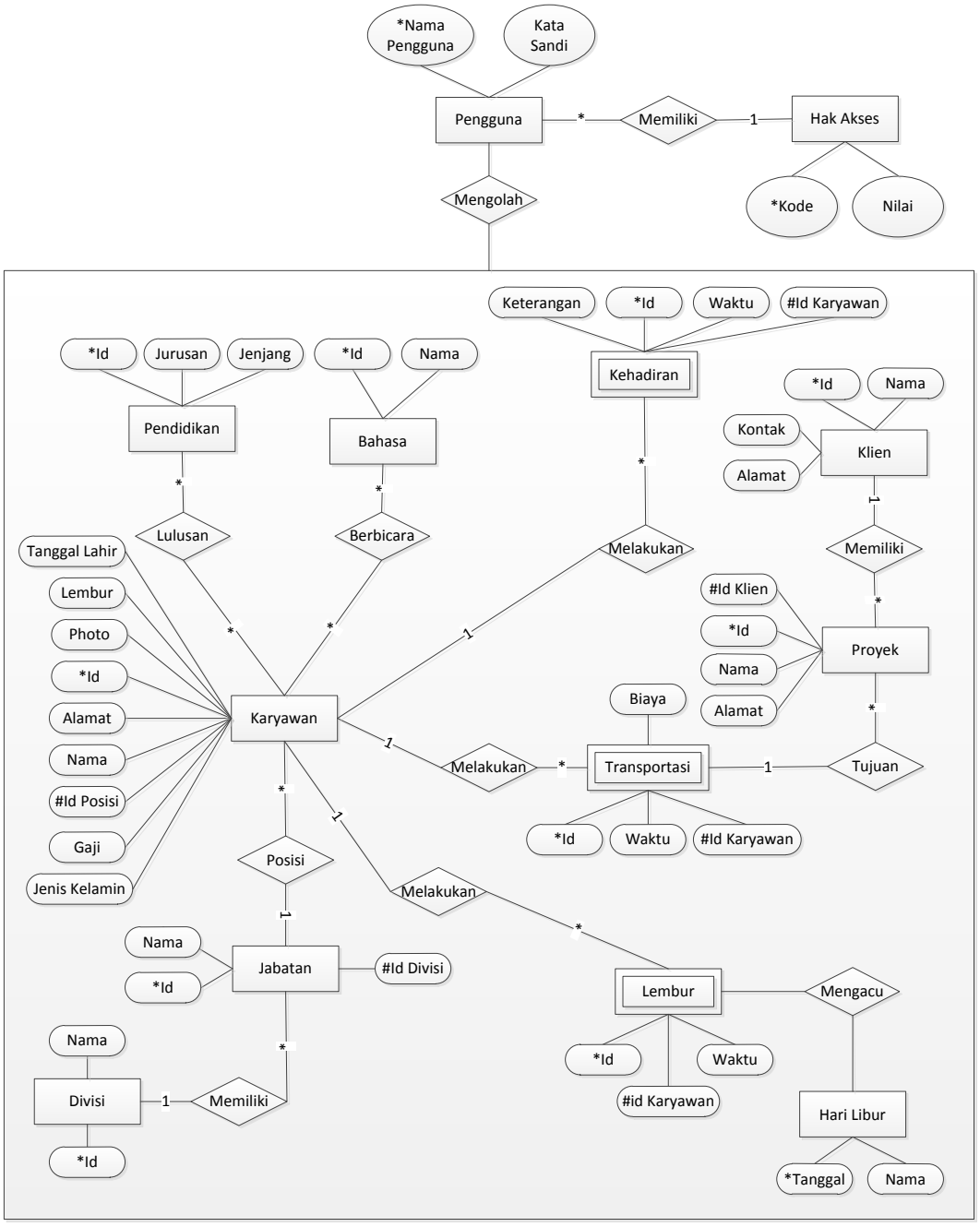
Aplikasi penggajian ini merupakan aplikasi yang berjalan di atas *desktop*.Aplikasi ini merupakan aplikasi yang berbasis GUI (*Graphical Unit Interface*).

3.2.1.4 Antarmuka Komunikasi

Aplikasi penggajian ini merupakan aplikasi yang menggunakan database, sehingga aplikasi ini membutuhkan database server yang terkoneksi melalui jaringan komunikasi.Protokol jaringan yang digunakan adalah TCP/IP.

3.2.2 Entity Relational Diagram

Berikut ini adalah *Entity Relational Diagram* untuk aplikasi sistem penggajian yang lama :

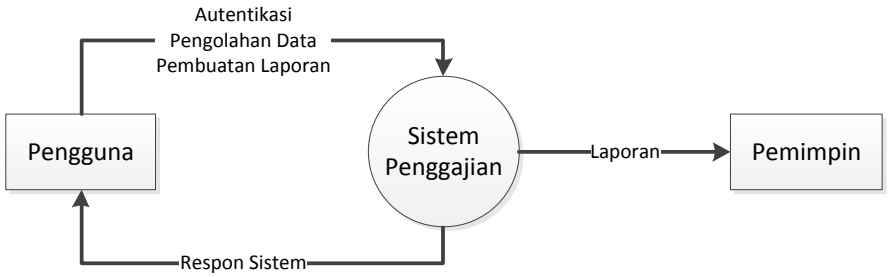


Gambar 3.2 Entity Relational Diagram

3.2.3 Kebutuhan Fungsional

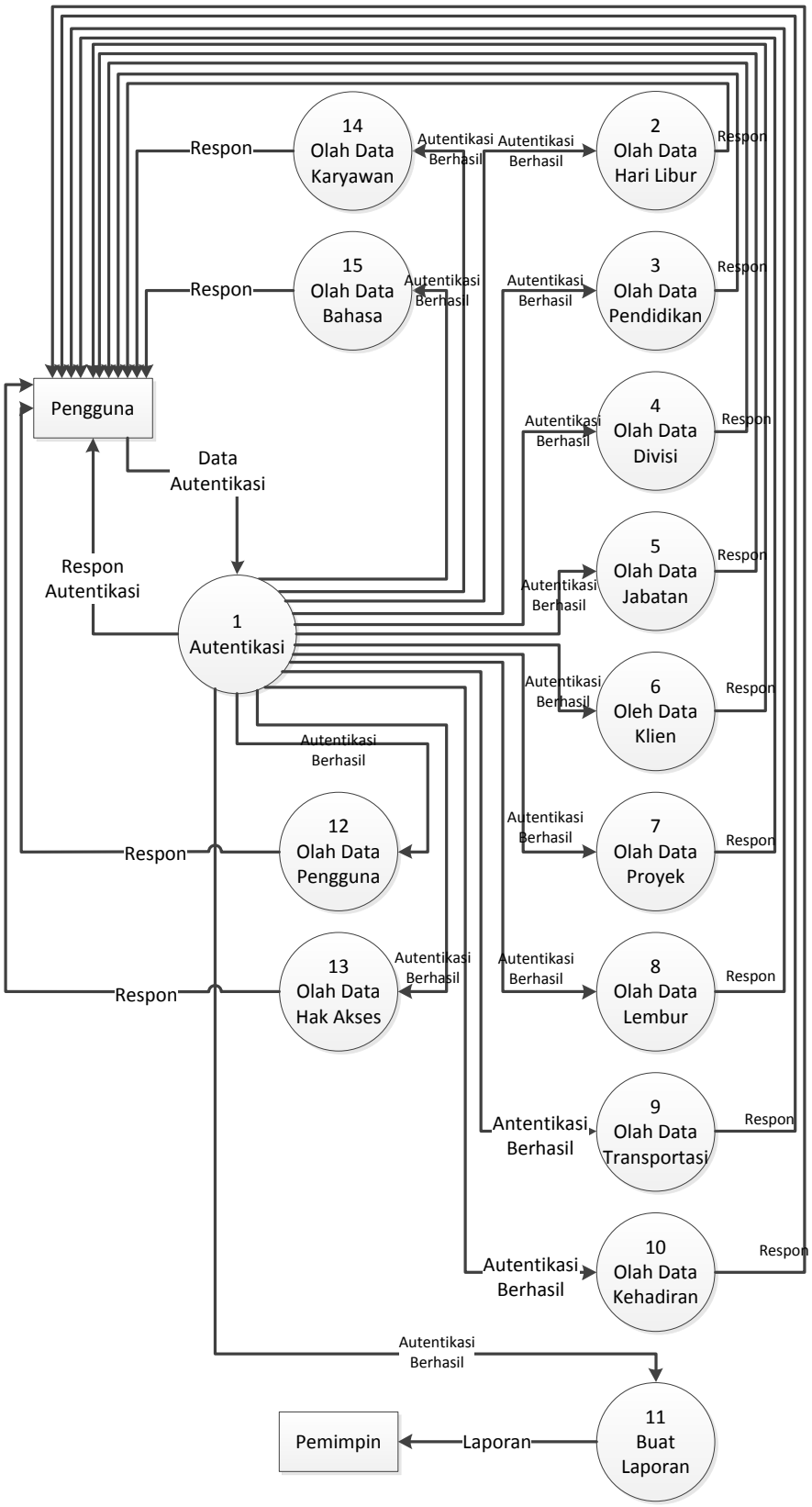
3.2.3.1 Aliran Informasi

Diagram konteks atau juga disebut sebagai model sistem *fundamental*merepresentasikanseluruh elemen sistem sebagai sebuah *bubble* tunggal dengan data input dan yang ditunjukkan oleh anak panah yang masuk dan keluar secara berurutan.



Gambar 3.3 Diagram Konteks

3.2.3.2 Data Flow Diagram Level 1



Gambar 3.4 Data Flow Diagram Level 1

Terdapat 11 proses inti pada sistem penggajian ini :

a. 1. Autentikasi

   Proses ini merupakan proses autentikasi terhadap pengguna yang akan menggunakan sistem penggajian ini. Pada proses ini pengguna akan dicek apakah pengguna merupakan pengguna sistem atau bukan.

b. 2. Olah Data Hari Libur

   Proses ini merupakan proses pengolahan data hari libur. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data hari libur.

c. 3. Olah Data Pendidikan

   Proses ini merupakan proses pengolahan data pendidikan. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data pendidikan.

d. 4. Olah Data Divisi

   Proses ini merupakan proses pengolahan data divisi. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data divisi.

e. 5. Olah Data Jabatan

   Proses ini merupakan proses pengolahan data jabatan. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data jabatan.

f. 6. Olah Data Klien

Proses ini merupakan proses pengolahan data klien. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data klien.

g. 7. Olah Data Proyek

Proses ini merupakan proses pengolahan data proyek. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data proyek.

h. 8. Olah Data Lembur

Proses ini merupakan proses pengolahan data lembur karyawan. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data lembur karyawan.

i. 9. Olah Data Transportasi

Proses ini merupakan proses pengolahan data transportasi. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data tranportasi karyawan.

j. 10. Olah Data Kehadiran

Proses ini merupakan proses pengolahan data kehadiran karyawan. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data kehadiran karyawan.

k. 11. Buat Laporan

Proses ini merupakan proses pembuatan laporan untuk pemimpin. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi.

l. 12. Olah Data Pengguna

Proses ini merupakan proses pengolahan data pengguna sistem penggajian. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data pengguna.

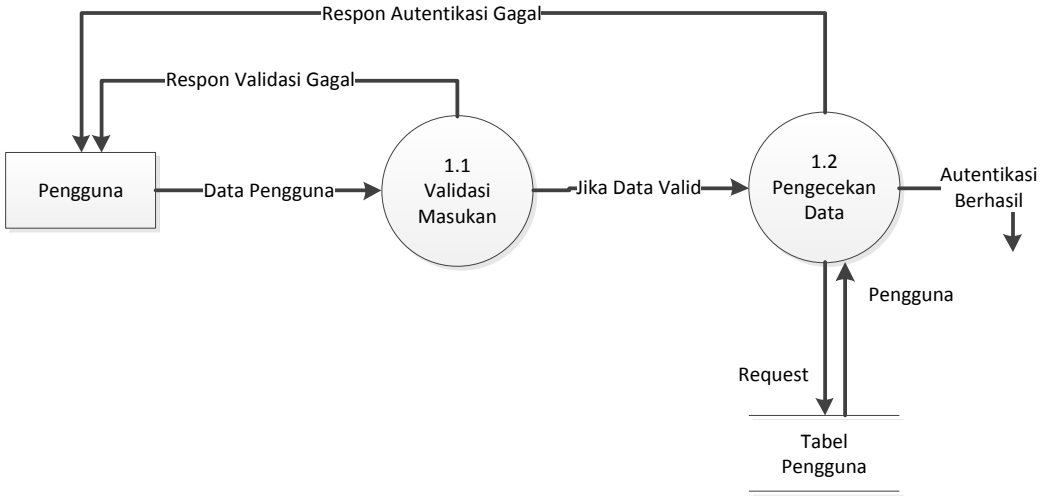
m. 13. Olah Data Hak Akses

Proses ini merupakan proses pengolahan data hak akses pengguna. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data lembur hak akses pengguna.

n. 14. Oleh Data Karyawan

Proses ini merupakan proses pengolahan data karyawan. Proses ini dapat dilakukan jika pengguna telah melewati proses autentikasi. Pada proses ini pengguna data melakukan menipulasi terhadap data karyawan.

3.2.3.3 Data Flow Diagram Level 2 Autentikasi



Gambar 3.5 Data Flow Diagram Level 2 Autentikasi

Pada proses autentikasi terdapat 2 proses inti :

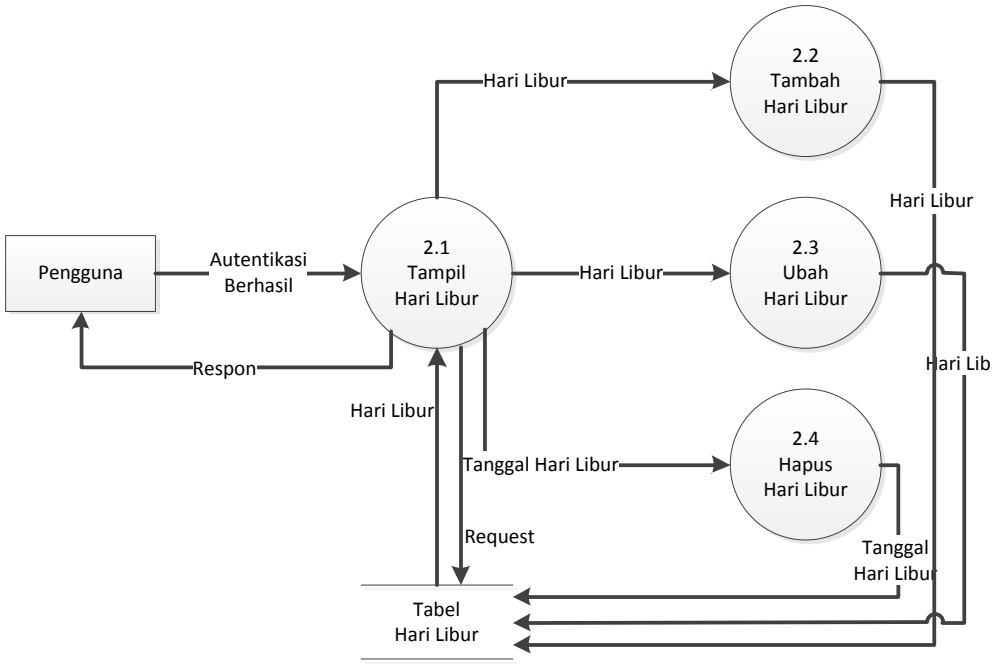
a. 1.1 Validasi Masukan

Proses ini merupakan proses pengecekan validasi masukan pengguna, pada proses ini sistem akan mengecek apakah data yang dimasukkan oleh pengguna valid atau tidak, seperti tidak memasukkan data yang kosong, dan lain-lain.

b. 1.2 Pengecekan Data

Pada proses ini sistem akan melakukan pengecekan ke database apakah penggua terdaftar di sistem, jika yam aka user dapat melanjutkan proses, jika tidak maka sistem akan mengembalikan respon autentikasi gagal.

3.2.3.4 Data Flow Diagram Level 2 Olah Data Hari Libur



Gambar 3.6Data Flow Diagram Level 2 Olah Data Hari Libur

Pada proses oleh data hari libur terdapat 4 inti proses :

a. 2.1 Tampil Hari Libur

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel hari libur yang terdapat dalam *database* ke sistem penggajian.

b. 2.2 Tambah Hari Libur

Proses ini merupakan proses penambahan data hari libur ke tabel hari libur yang terdapat pada *database*.

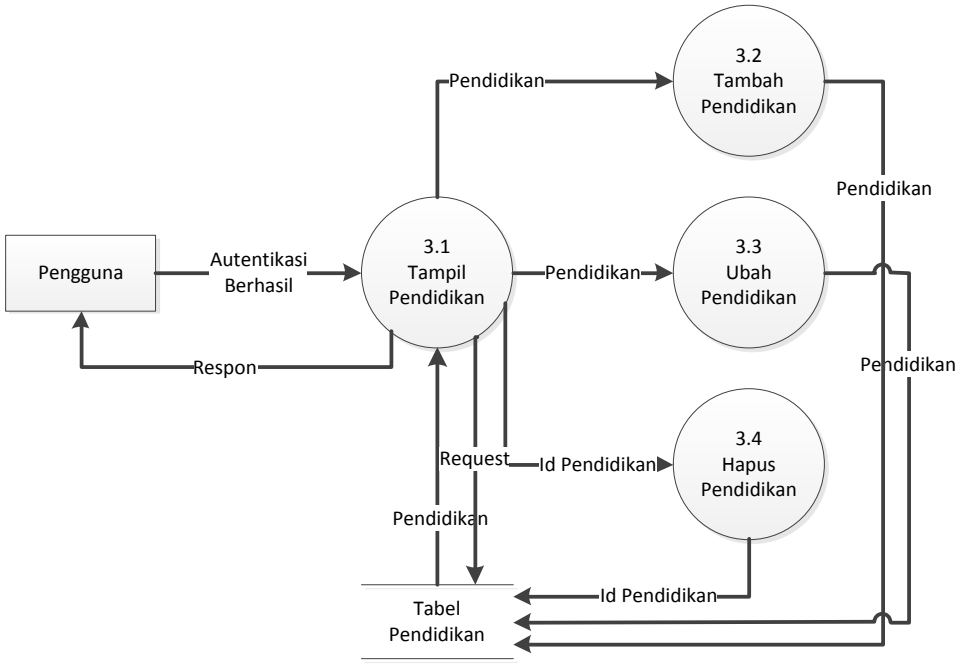
c. 2.3 Ubah Hari Libur

Proses ini merupakan proses pengubahan data hari libur yang terdapat pada tabel hari libur yang ada pada *database*.

d. 2.4 Hapus Hari Libur

Proses ini merupakan proses penghapusan data hari libur yang terdapat pada tabel hari libur yang ada pada *database*.

3.2.3.5 Data Flow Diagram Level 2 Proses Olah Data Pendidikan



Gambar 3.7Data Flow Diagram Level 2 Proses Olah Data Pendidikan

Pada proses oleh data pendidikan terdapat 4 inti proses :

a. 3.1 Tampil Pendidikan

   Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel pendidikan yang terdapat dalam *database* ke sistem penggajian.

b. 3.2 Tambah Pendidikan

   Proses ini merupakan proses penambahan data pendidikan ke tabel pendidikan yang terdapat pada *database*.

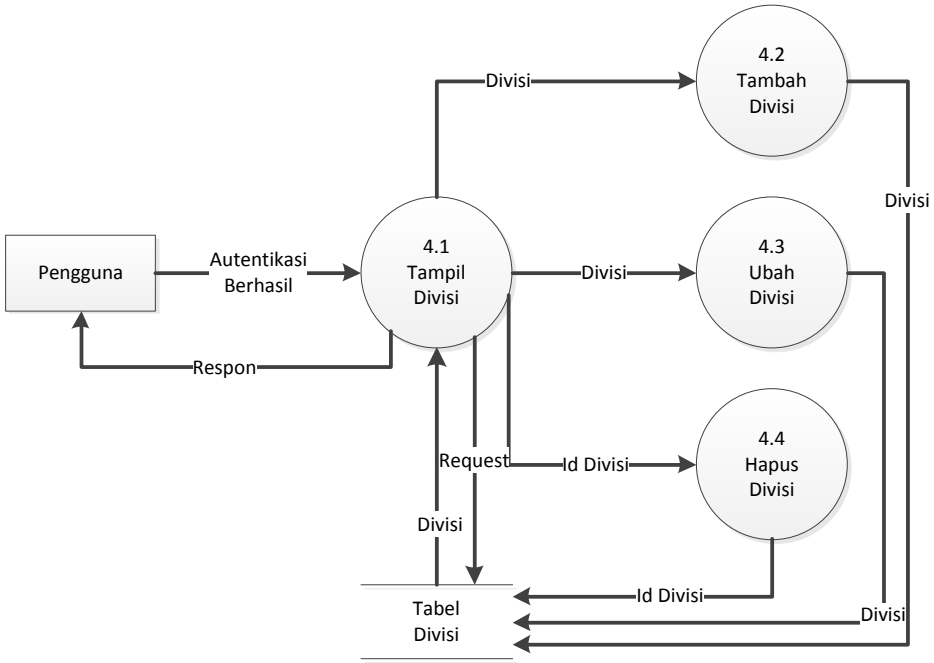
c. 3.3 Ubah Pendidikan

   Proses ini merupakan proses pengubahan data pendidikan yang terdapat pada tabel pendidikan yang ada pada *database*.

d. 3.4 Hapus Pendidikan

   Proses ini merupakan proses penghapusan data pendidikan yang terdapat pada tabel pendidikan yang ada pada *database*.

3.2.3.6 Data Flow Diagram Level 2 Olah Data Divisi



Gambar 3.8Data Flow Diagram Level 2 Olah Data Divisi

Pada proses oleh data divisi terdapat 4 inti proses :

a. 4.1 Tampil Divisi

   Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel divisi yang terdapat dalam *database* ke sistem penggajian.

b. 4.2 Tambah Divisi

   Proses ini merupakan proses penambahan data divisi ke tabel divisi yang terdapat pada *database*.

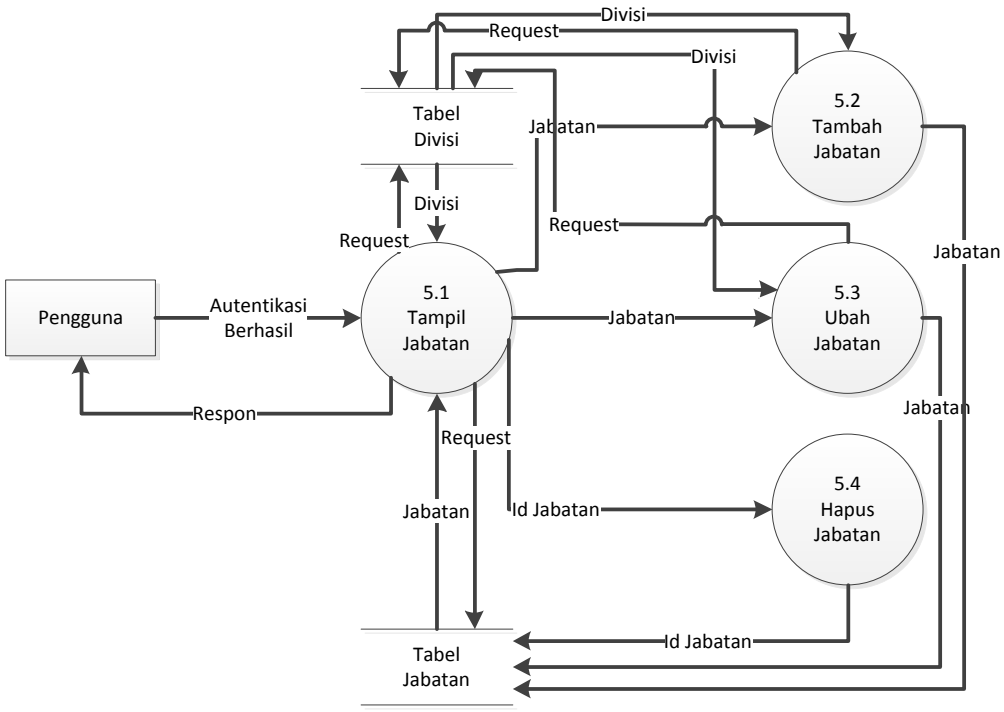
c. 4.3 Ubah Divisi

   Proses ini merupakan proses pengubahan data divisi yang terdapat pada tabel divisi yang ada pada *database*.

d. 4.4 Hapus Divisi

Proses ini merupakan proses penghapusan data divisi yang terdapat pada tabel divisi yang ada pada *database*.

3.2.3.7 Data Flow Diagram Level 2 Olah Data Jabatan



Gambar 3.9 Data Flow Diagram Level 2 Olah Data Jabatan

Pada proses oleh data jabatan terdapat 4 inti proses :

a. 5.1 Tampil Jabatan

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel jabatan yang terdapat dalam *database* ke sistem penggajian.

b. 5.2 Tambah Jabatan

Proses ini merupakan proses penambahan data jabatan ke tabel jabatan yang terdapat pada *database*.

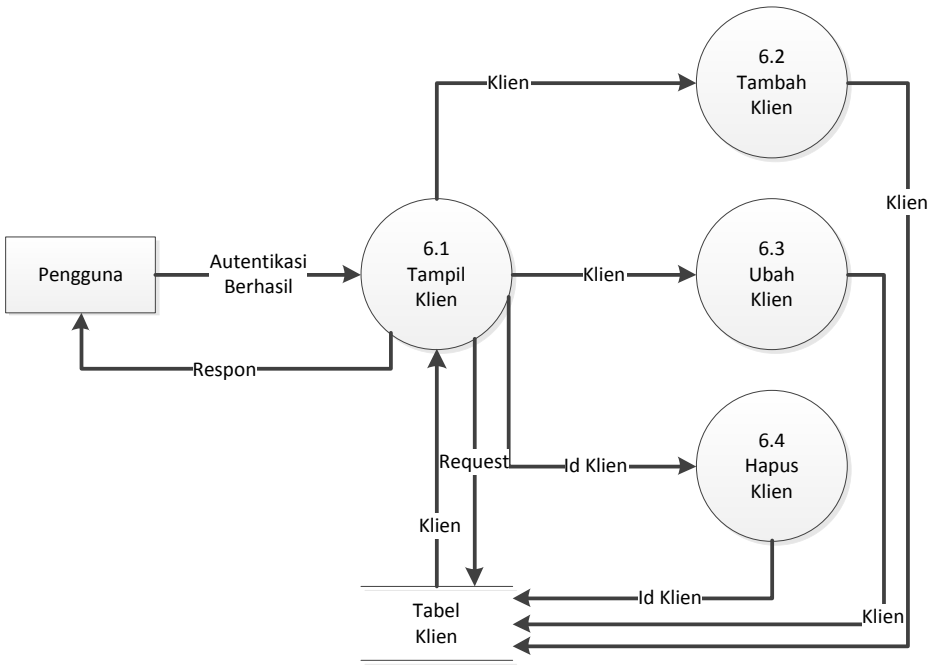
c. 5.3 Ubah Jabatan

Proses ini merupakan proses pengubahan data jabatan yang terdapat pada tabel jabatan yang ada pada *database*.

d. 5.4 Hapus Jabatan

Proses ini merupakan proses penghapusan data jabatan yang terdapat pada tabel jabatan yang ada pada *database*.

3.2.3.8 Data Flow Diagram Level 2 Olah Data Klien



Gambar 3.10 Data Flow Diagram Level 2 Olah Data Klien

Pada proses oleh data klien terdapat 4 inti proses :

a. 6.1 Tampil Klien

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel klien yang terdapat dalam *database* ke sistem penggajian.

b. 6.2 Tambah Klien

Proses ini merupakan proses penambahan data klien ke tabel klien yang terdapat pada *database*.

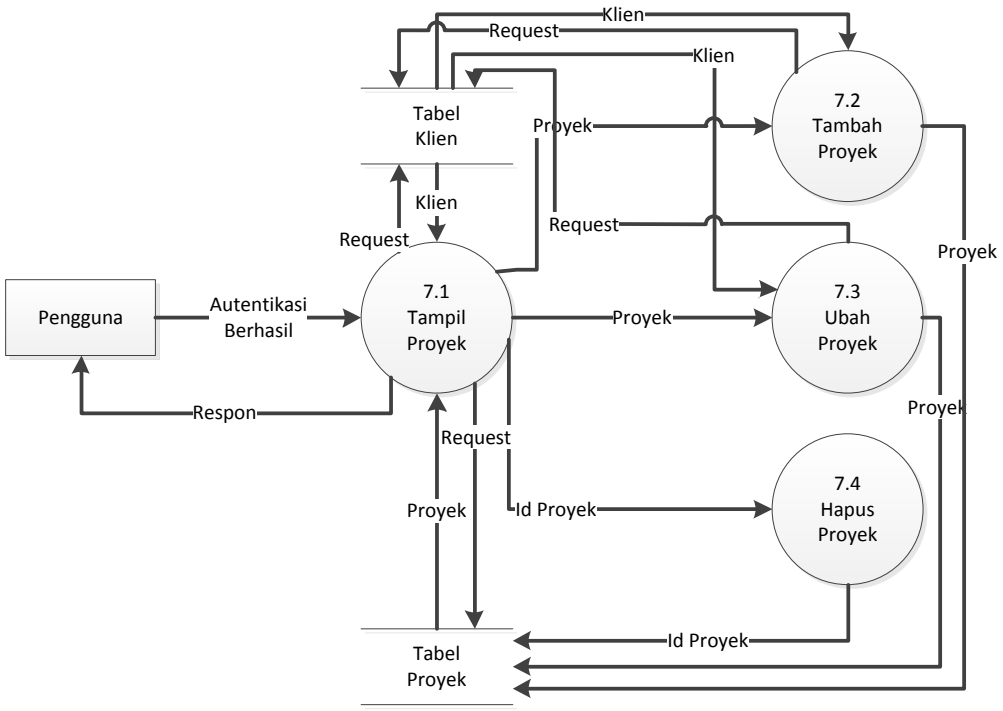
c. 6.3 Ubah Klien

Proses ini merupakan proses pengubahan data klien yang terdapat pada tabel klien yang ada pada *database*.

d. 6.4 Hapus Klien

Proses ini merupakan proses penghapusan data klien yang terdapat pada tabel klien yang ada pada *database*.

3.2.3.9 Data Flow Diagram Level 2 Olah Data Proyek



Gambar 3.11 Data Flow Diagram Level 2 Olah Data Proyek

Pada proses oleh data proyek terdapat 4 inti proses :

a. 7.1 Tampil Proyek

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel proyek yang terdapat dalam *database* ke sistem penggajian.

b. 7.2 Tambah Proyek

Proses ini merupakan proses penambahan data proyek ke tabel proyek yang terdapat pada *database*.

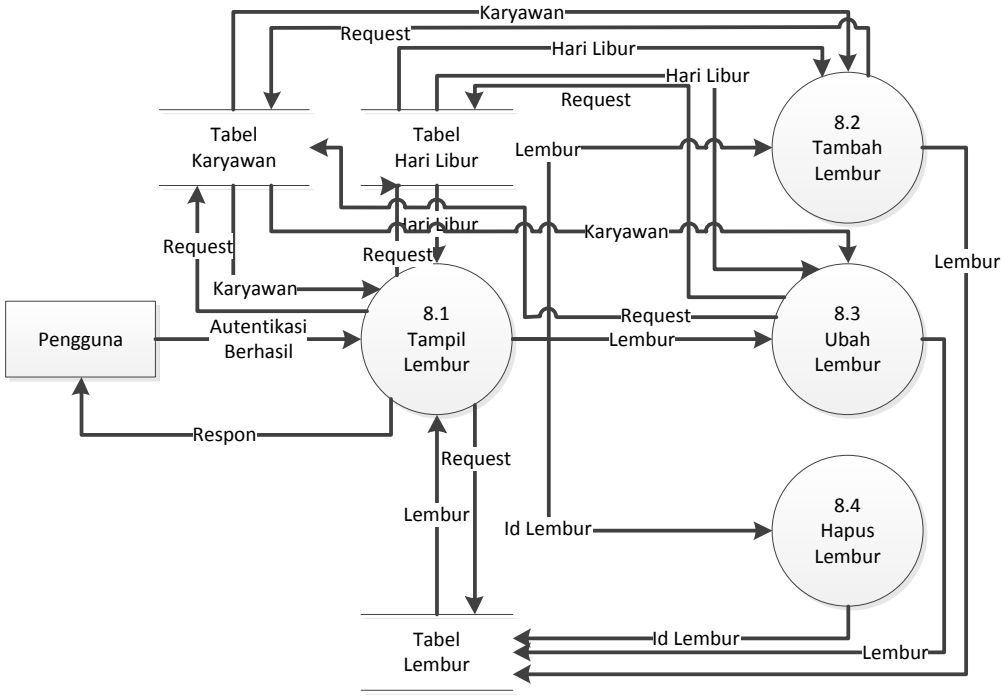
c. 7.3 Ubah Proyek

Proses ini merupakan proses pengubahan data proyek yang terdapat pada tabel proyek yang ada pada *database*.

d. 7.4 Hapus Proyek

Proses ini merupakan proses penghapusan data proyek yang terdapat pada tabel proyek yang ada pada *database*.

3.2.3.10 Data Flow Diagram Level 2 Olah Data Lembur



Gambar 3.12 Data Flow Diagram Level 2 Olah Data Lembur

Pada proses oleh data lembur terdapat 4 inti proses :

a. 8.1 Tampil Lembur

   Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel lembur yang terdapat dalam *database* ke sistem penggajian.

b. 8.2 Tambah Lembur

   Proses ini merupakan proses penambahan data lembur ke tabel lembur yang terdapat pada *database*.

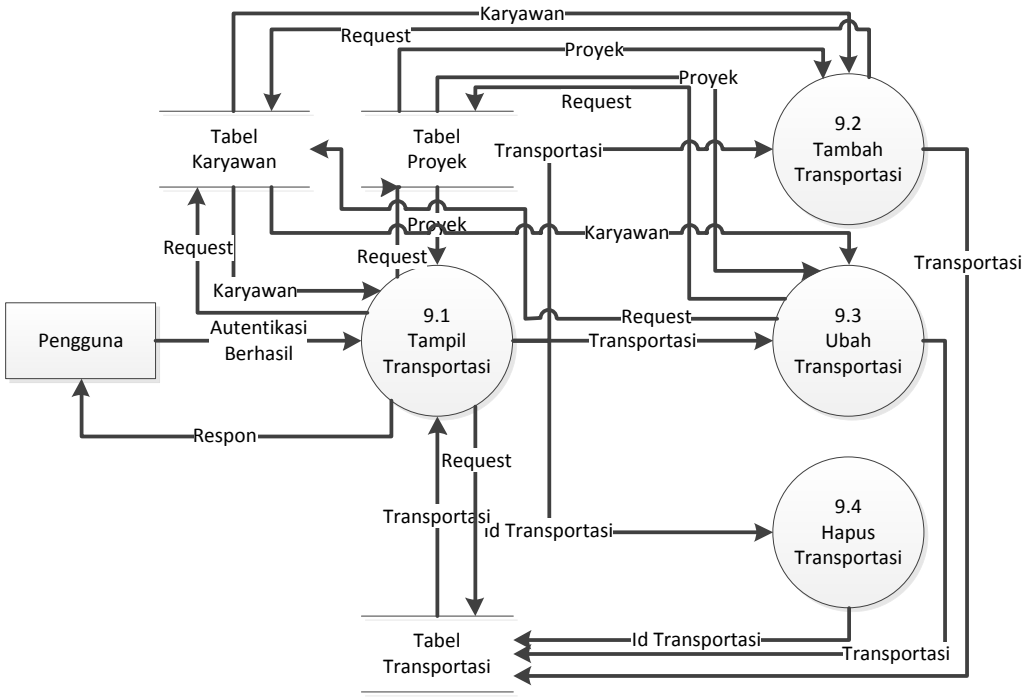
c. 8.3 Ubah Lembur

   Proses ini merupakan proses pengubahan data lembur yang terdapat pada tabel lembur yang ada pada *database*.

d. 8.4 Hapus Lembur

   Proses ini merupakan proses penghapusan data lembur yang terdapat pada tabel lembur yang ada pada *database*.

3.2.3.11 Data Flow Diagram Level 2 Olah Data Transportasi



Gambar 3.13 Data Flow Diagram Level 2 Olah Data Transportasi

Pada proses oleh data transsportasi terdapat 4 inti proses :

a. 9.1 Tampil Transportasi

   Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel transportasi yang terdapat dalam *database* ke sistem penggajian.

b. 9.2 Tambah Transportasi

   Proses ini merupakan proses penambahan data transportasi ke tabel trasnportasi yang terdapat pada *database*.

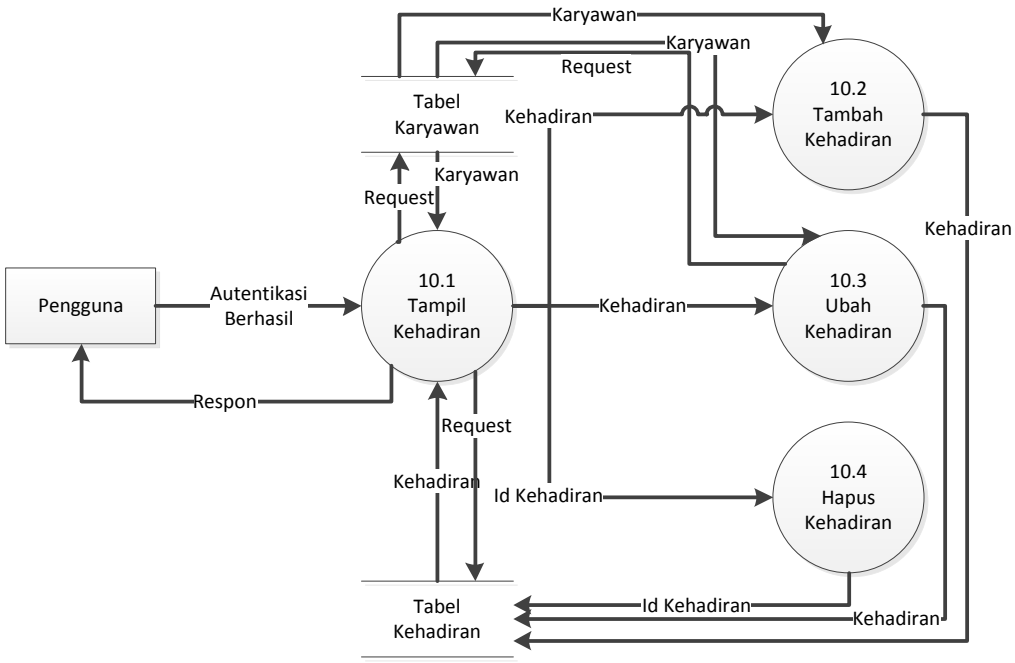
c. 9.3 Ubah Transportasi

   Proses ini merupakan proses pengubahan data transportasi yang terdapat pada tabel transportasi yang ada pada *database*.

d. 9.4 Hapus Transportasi

Proses ini merupakan proses penghapusan data transportasi yang terdapat pada tabel transportasi yang ada pada *database*.

3.2.3.12 Data Flow Diagram Level 2 Olah Data Kehadiran



Pada proses oleh data kehadiran terdapat 4 inti proses :

a. 10.1 Tampil Kehadiran

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel kehadiran yang terdapat dalam *database* ke sistem penggajian.

b. 10.2 Tambah Kehadiran

Proses ini merupakan proses penambahan data kehadiran ke tabel kehadiran yang terdapat pada *database*.

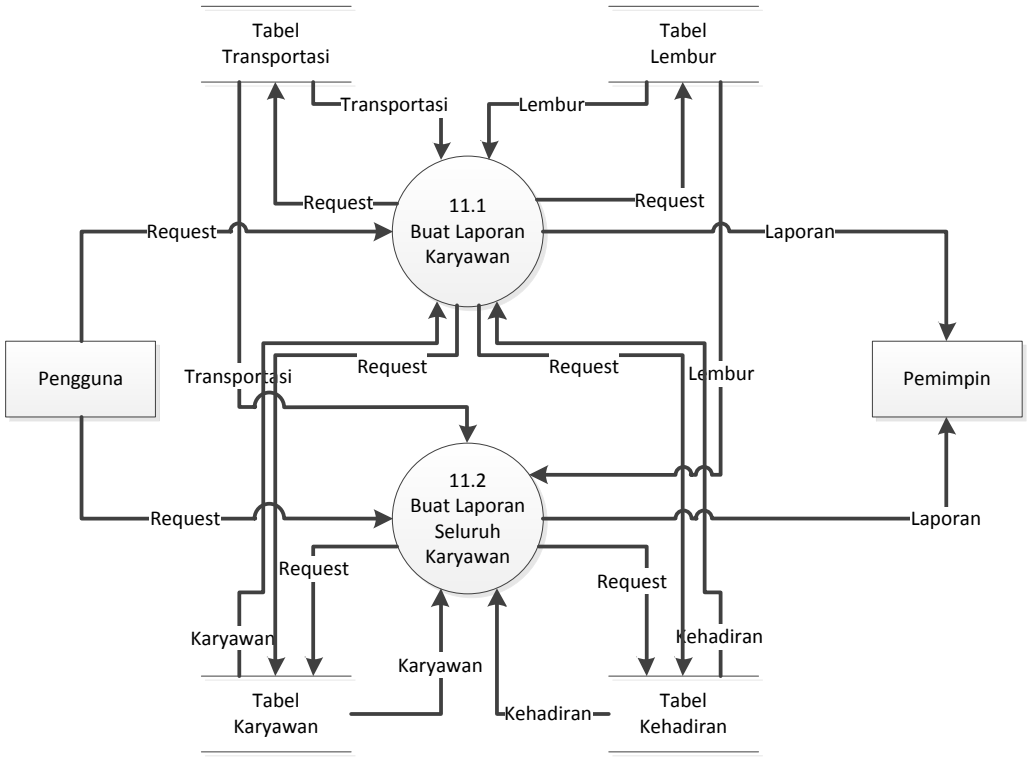
c. 10.3 Ubah Kehadiran

Proses ini merupakan proses pengubahan data kehadiran yang terdapat pada tabel kehadiran yang ada pada *database*.

d. 10.4 Hapus Kehadiran

Proses ini merupakan proses penghapusan data kehadiran yang terdapat pada tabel kehadiran yang ada pada *database*.

3.2.3.13 Data Flow Diagram Level 2 Buat Laporan



Gambar 3.14 Data Flow Diagram Level 2 Buat Laporan

Pada proses buat laporan terdapta 2 inti proses :

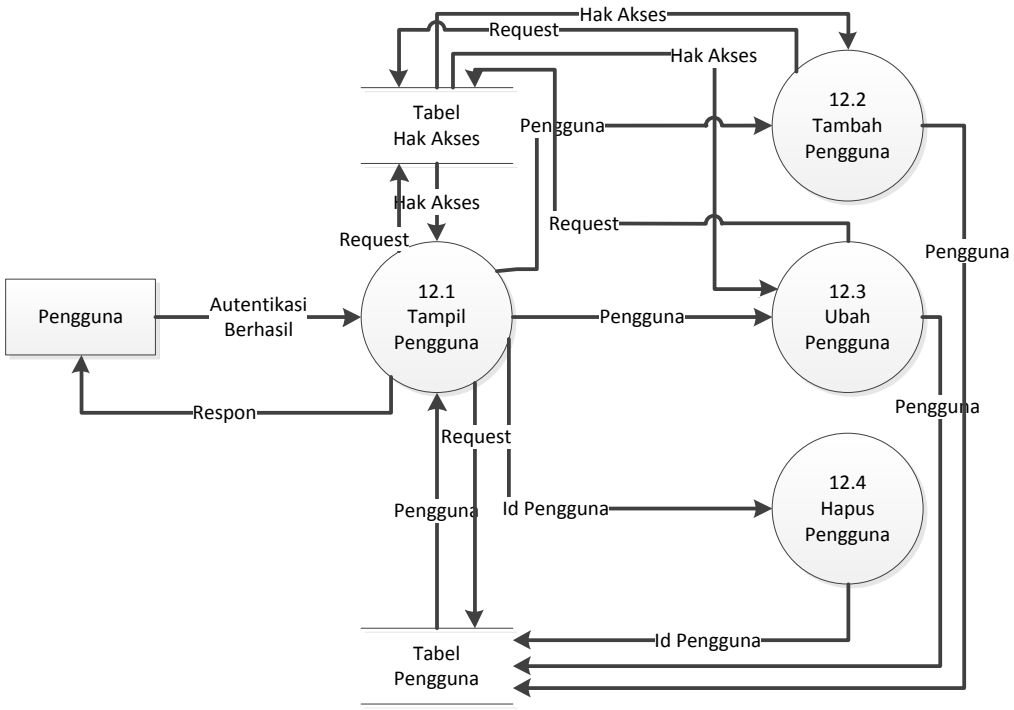
a. 11.1 Buat Laporan Karyawan

Proses ini merupakan proses membuat laporan untuk seorang karyawan. Laporan ini nantinya akan dilaporkan ke pemimpin perusahaan.

b. 11.2 Buat Laporan Seluruh Karyawan

Proses ini merupakan proses membuat laporan kumulatif seluruh karyawan. Laporan ini nantinya akan dilaporkan ke pemimpin perusahaan.

3.2.3.14 Data Flow Diagram Level 2 Olah Data Pengguna



Gambar 3.15 Data Flow Diagram Level 2 Olah Data Pengguna

Pada proses oleh data pengguna terdapat 4 inti proses :

a. 12.1 Tampil Pengguna

   Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel pengguna yang terdapat dalam *database* ke sistem penggajian.

b. 12.2 Tambah Pengguna

   Proses ini merupakan proses penambahan data pengguna ke tabel pengguna yang terdapat pada *database*.

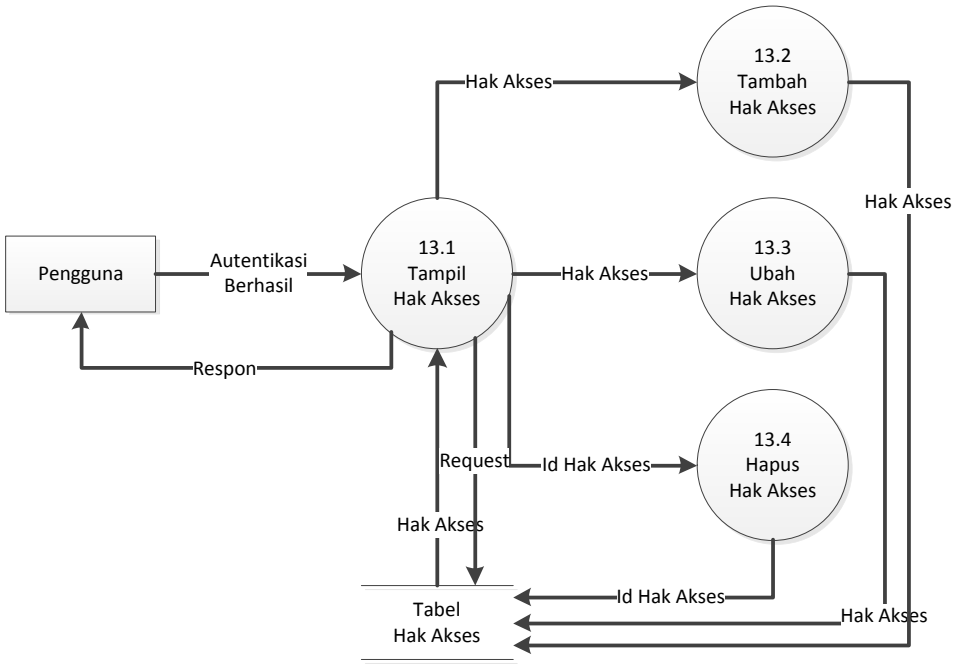
c. 12.3 Ubah Pengguna

   Proses ini merupakan proses pengubahan data pengguna yang terdapat pada tabel pengguna yang ada pada *database*.

d. 12.4 Hapus Pengguna

Proses ini merupakan proses penghapusan data pengguna yang terdapat pada tabel pengguna yang ada pada *database*.

3.2.3.15 Data Flow Diagram Level 2 Olah Data Hak Akses



Gambar 3.16 Data Flow Diagram Level 2 Olah Data Hak Akses

Pada proses oleh data hak akses terdapat 4 inti proses :

a. 13.1 Tampil Hak Akses

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel hak akses yang terdapat dalam *database* ke sistem penggajian.

b. 13.2 Tambah Akses

Proses ini merupakan proses penambahan data hak akses ke tabel hak akses yang terdapat pada *database*.

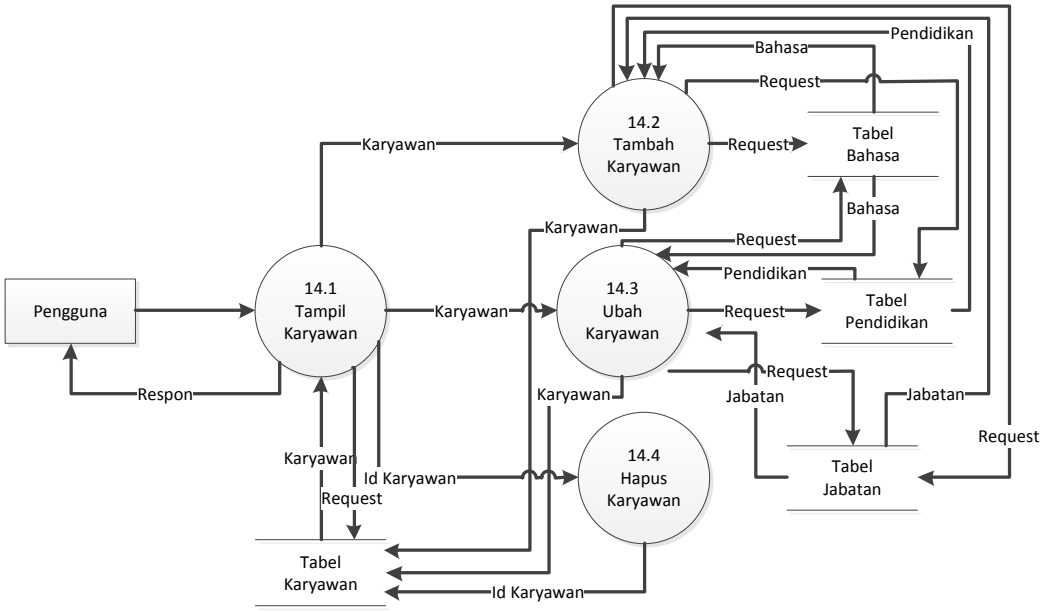
c. 13.3 Ubah Hak Akses

Proses ini merupakan proses pengubahan data hak akses yang terdapat pada tabel hak akses yang ada pada *database*.

d. 13.4 Hapus Hak Akses

Proses ini merupakan proses penghapusan data hak akses yang terdapat pada tabel hak akses yang ada pada *database*.

3.2.3.16 Data Flow Diagram Level 2 Olah Data Karyawan



Gambar 3.17 Data Flow Diagram Level 2 Olah Data Karyawan

Pada proses oleh data karyawan terdapat 4 inti proses :

a. 14.1 Tampil Karyawan

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel karyawan yang terdapat dalam *database* ke sistem penggajian.

b. 14.2 Tambah Karyawan

Proses ini merupakan proses penambahan data karyawan ke tabel karyawan yang terdapat pada *database*.

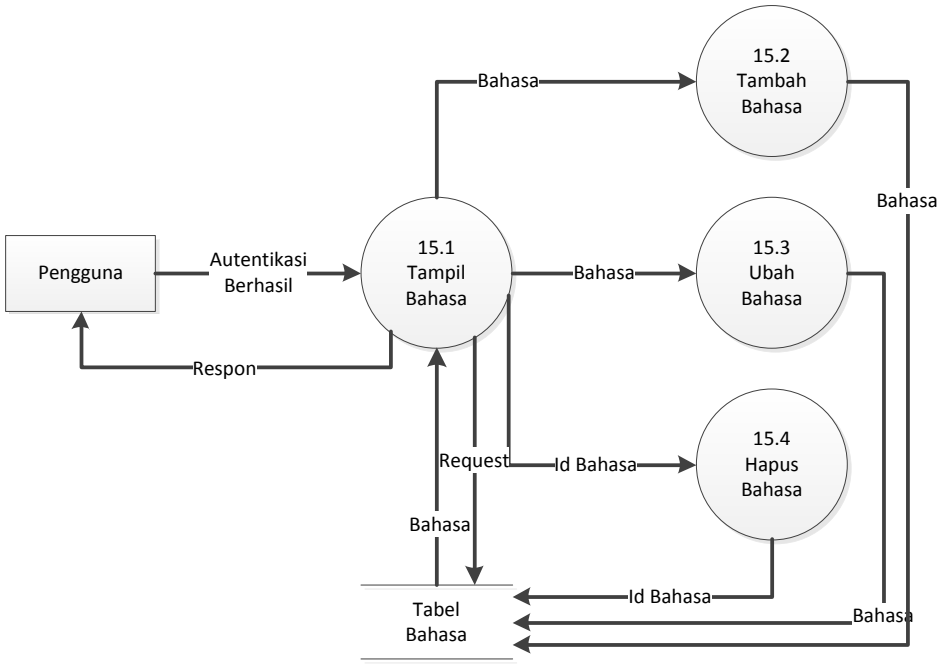
c. 14.3 Ubah Karyawan

Proses ini merupakan proses pengubahan data karyawan yang terdapat pada tabel karyawan yang ada pada *database*.

d. 14.4 Hapus Karyawan

Proses ini merupakan proses penghapusan data karyawan yang terdapat pada tabel karyawan yang ada pada *database*.

3.2.3.17 Data Flow Diagram Level 2 Olah Data Bahasa



Gambar 3.18 Data Flow Diagram Level 2 Olah Data Bahasa

Pada proses oleh data bahasa terdapat 4 inti proses :

e. 15.1 Tampil Bahasa

Proses ini merupakan proses menampilkan seluruh data yang terdapat pada tabel bahasa yang terdapat dalam *database* ke sistem penggajian.

f. 15.2 Tambah Bahasa

Proses ini merupakan proses penambahan data bahasa ke tabel bahasa yang terdapat pada *database*.

g. 15.3 Ubah Bahasa

Proses ini merupakan proses pengubahan data bahasa yang terdapat pada tabel bahasa yang ada pada *database*.

h. 15.4 Hapus Bahasa

Proses ini merupakan proses penghapusan data bahasa yang terdapat pada tabel bahasa yang ada pada *database*.

3.3 Deskripsi Rinci Kebutuhan Sistem Baru

Bagian ini berisikan semua kebutuhan perangkat lunak pada sistem yang lama dan diuraikan secara rinci untuk keperluan pengembangan ke sistem yang baru. Kebutuhan perangkat lunak tersebut meliputi kebutuhan antarmuka eksternal,deskripsi data, kebutuhan fungsional dan non-fungsional.

3.3.1 Kebutuhan Antar Muka Eksternal

3.3.1.1 Antarmuka Pemakai

Aplikasi sistem penggajian ini merupakan aplikasi berbasis *desktop* dan *database*, sehingga untuk menggunakan aplikasi ini pengguna memerlukan komputer dan *database management system*yang telah terpasang pada komputernya.Pada aplikasi pengguna dapat mengolah data penggajian melalui menu-menu yang telah terdapat pada aplikasi.

3.3.1.2 Antarmuka Perangkat Keras

Aplikasi sistem penggajian ini merupakan aplikasi *standalone* yaitu aplikasi yang berdisi sendiri, sehingga pengguna hanya memerlukan komputer saja untuk menjalankannya.

3.3.1.3 Antarmuka Perangkat Lunak

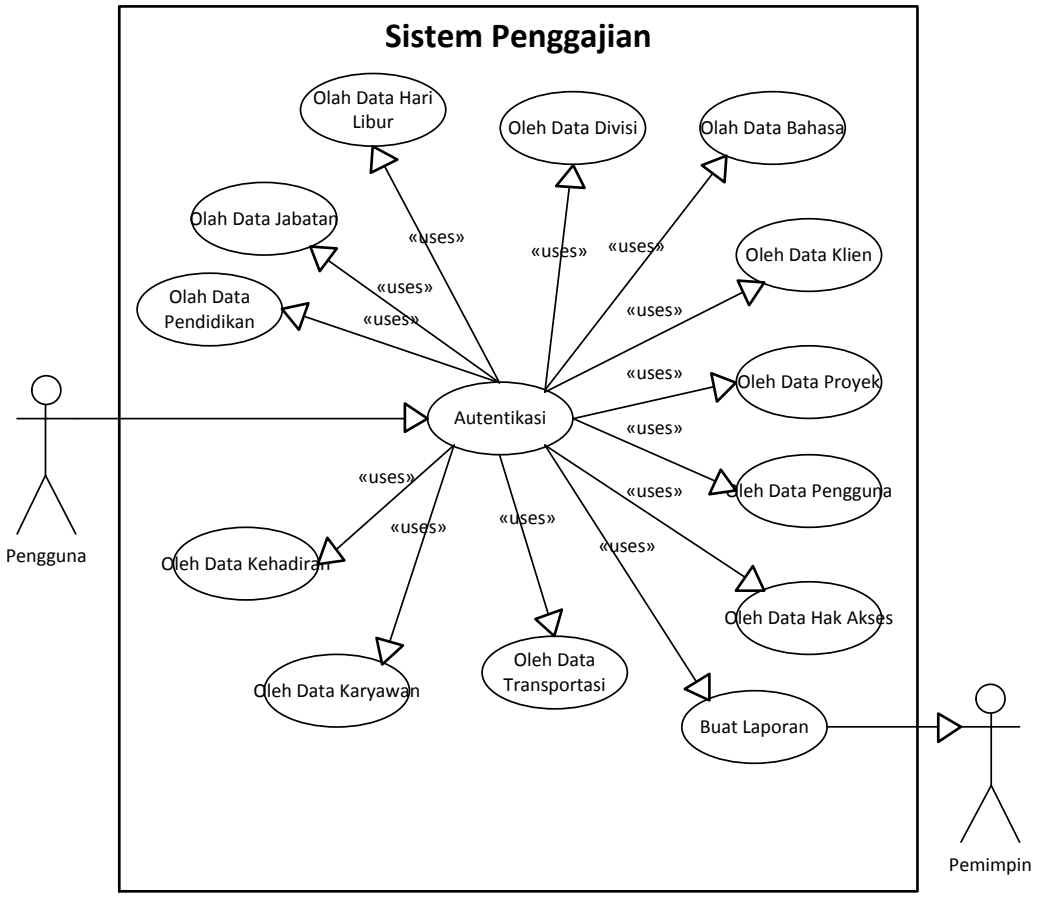
Aplikasi penggajian ini merupakan aplikasi yang berjalan di atas *desktop*.Aplikasi ini merupakan aplikasi yang berbasis GUI (*Graphical Unit Interface*).

3.3.1.4 Antarmuka Komunikasi

Aplikasi penggajian ini merupakan aplikasi yang menggunakan database, sehingga aplikasi ini membutuhkan database *server* yang terkoneksi melalui jaringan komunikasi.Protokol jaringan yang digunakan adalah TCP/IP.

3.3.2 Kebutuhan Fungsional

Sistem penggajian yang akan dikembangan memiliki 14 fungsi utama seperti yang terlihat pada usecase dibawah ini :



Gambar 3.19 Usecases Sistem Penggajian

3.3.2.1 Autentikasi

Autentikasi merupakan proses pengecekan apakah pengguna terdaftar atau tidak, jika terdaftar maka pengguna dapat menggunakan sistem jika tidak, maka pengguna tidak dapat menggunakan sistem.

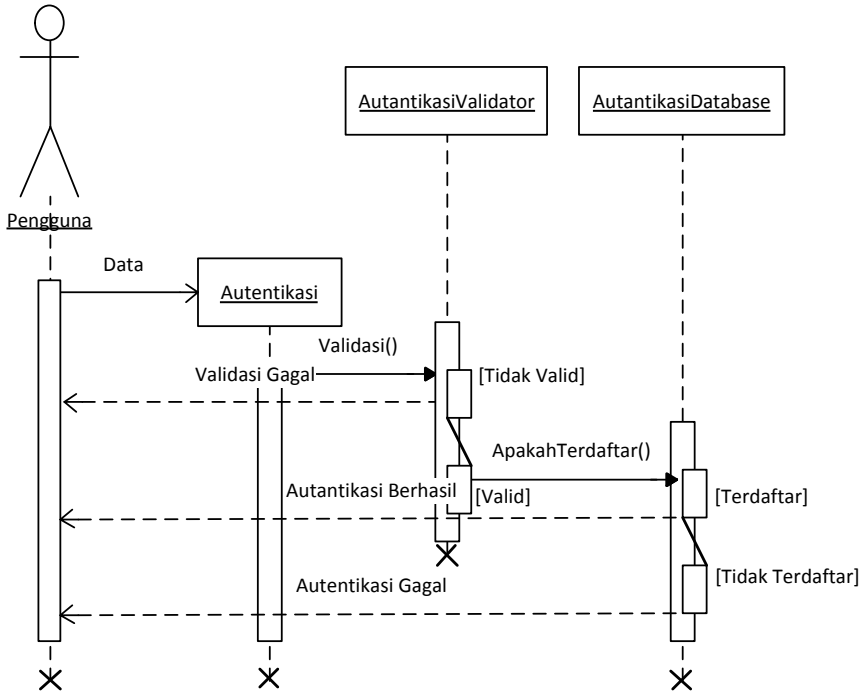
3.3.2.1.1 Skenario

Identifikasi	
Nomor	1
Nama	Autentikasi
Tujuan	Melakukan autentikasi terhadap pengguna
Deskripsi	Sistem menerima inputan nama penggunua dan kata sandi, lalu sistem akan melakukan pengecekan apakah pengguna telah terdaftar atau tidak, jika terdaftar maka pengguna akan dapat menggunakan sistem, jika tidak maka pengguna tidak dapat menggunakan sistem.
Tipe	

Aktor	Pengguna
Skenario Utama	
Kondisi Awal	Pengguna masuk ke form autantikasi
Aksi Aktor	**Reaksi Sistem**
Pengguna memasukkan data nama pengguna dan kata sandi	Mengirim data ke sistem
	Melakukan pengecekan jika berhasil maka pengguna masuk ke aplikasi jika gagal, maka pengguna akan mendapat peringatan
Skenario Alternatif – Autantikasi Gagal	
Aksi Aktor	**Reaksi Sistem**
Pengguna melakukan hal yang sama	Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Aplikasi akan ditampilkan

Tabel 3.2 Skenario Autentikasi

3.3.2.1.2 Dinamika Objek



Gambar 3.20 Dinamika Objek Autentikasi

Pertama pengguna memasukkan data autentikasi, setelah itu dilakukan proses validasi, jika gagal maka pengguna akan diperingatkan, jika berhasil maka sistem akan melakukan proses pengecekan data autentikasi. Jika pengguna

terdaftar di database maka proses autentikasi berhasil jika tidak maka proses autentikasi gagal dan pengguna tidak dapat masuk ke sistem.

3.3.2.2 Olah Data Bahasa

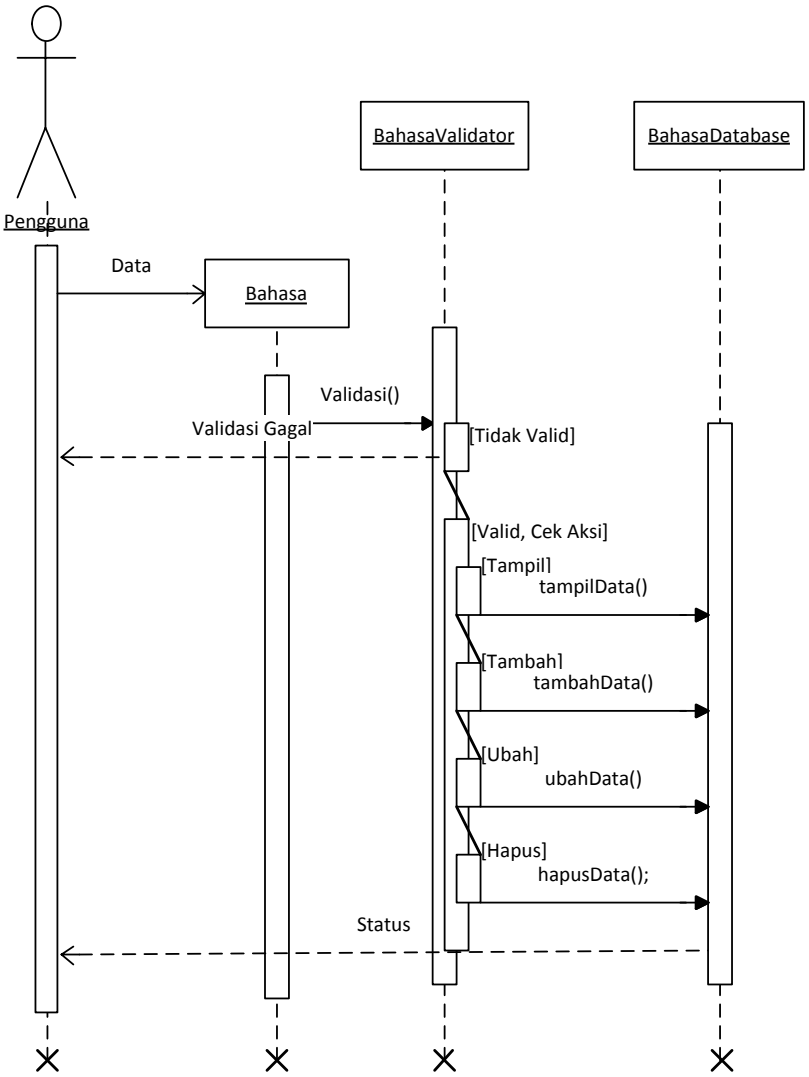
Fungsi ini merupakan fungsi yang digunakan untuk mengolah data bahasa. Fungsi ini hanya dapat dijalankan jika fungsi autentikasi telah terlewati, sehingga pengguna harus terdaftar terlebih dahulu jika akan menjalankan fungsi ini.

3.3.2.2.1 Skenario

Identifikasi		
Nomor	2	
Nama	Olah Data Bahasa	
Tujuan	Melakukan pengolahan data bahasa	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah data bahasa	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – ValidasiGagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.3 Skenario Olah Data Bahasa

3.3.2.2.2 Dinamika Objek



Gambar 3.21 Dinamika Objek Olah Data Bahasa

Pertama pengguna memasukkan data , lalu memerintahkan aksi yang akan dilakukan setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang telah diperintahkan oleh pengguna.

3.3.2.3 Olah Data Pendidikan

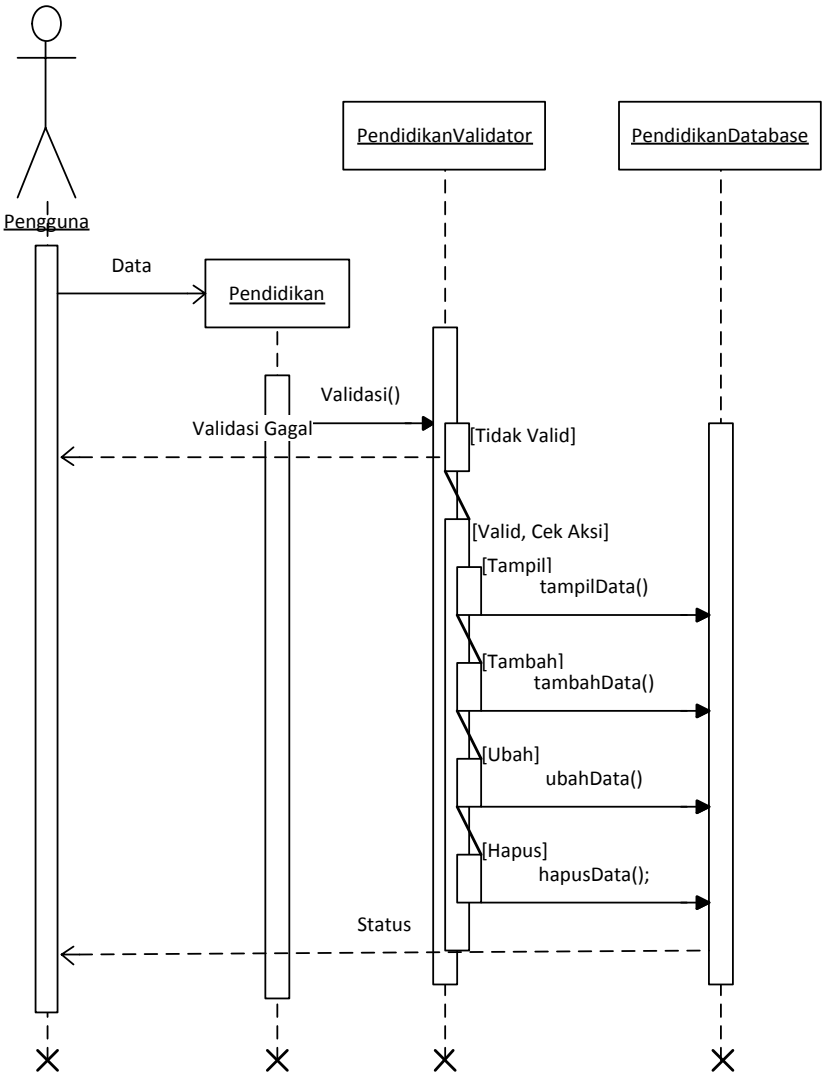
Fungsi ini merupakan fungsi untuk melakukan proses pengolahan data pendidikan. Fungsi ini hanya dapat dijalankan jika sudah melewati fungsi autentikasi.

3.3.2.3.1 Skenario

Identifikasi		
Nomor	3	
Nama	Olah Data Pendidikan	
Tujuan	Melakukan pengolahan data pendidikan	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah data pendidikan	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.4 Skenario Olah Data Pendidikan

3.3.2.3.2 Dinamika Objek



Gambar 3.22 Dinamika Objek Olah Data Pendidikan

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.4 Olah Data Hari Libur

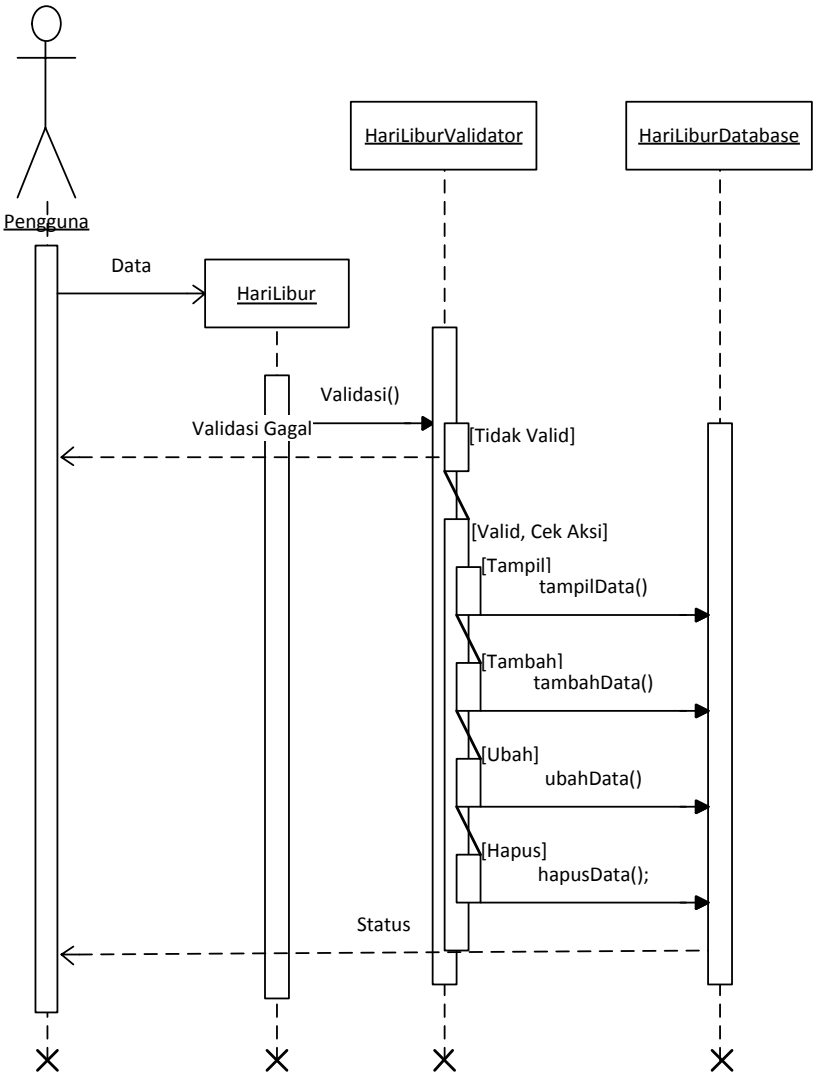
Fungsi ini merupakan fungsi pengolahan data hari libur. Fungsi ini hanya dapat dilakukan jika pengguna telah melewati proses autentikasi.

3.3.2.4.1 Skenario

Identifikasi		
Nomor	4	
Nama	Olah Data Hari Libur	
Tujuan	Melakukan pengolahan data hari libur	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah data hari libur	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.5 Skenario Olah Data Hari Libur

3.3.2.4.2 Dinamika Objek



Gambar 3.23 Dinamika Objek Olah Data Hari Libur

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.5 Olah Data Divisi

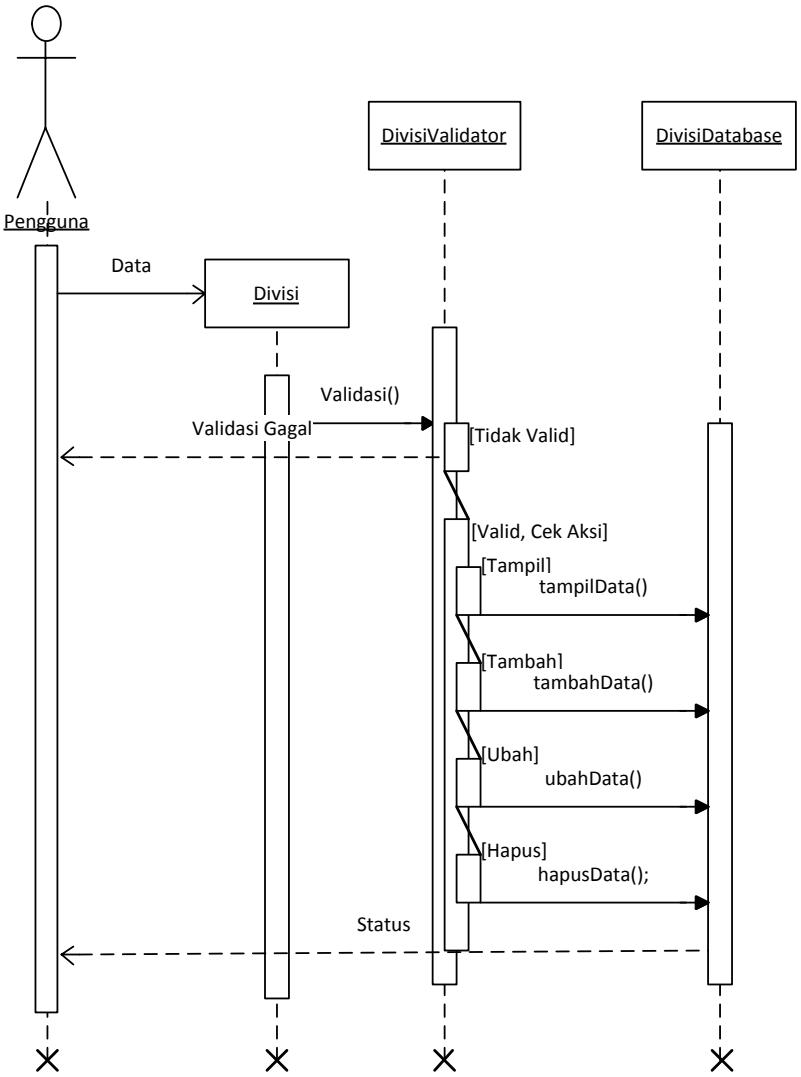
Fungsi ini merupakan proses untuk melakukan pengolahan data divisi. Fungsi ini hanya dapat dijalankan jika pengguna telah melewati proses autentikasi.

3.3.2.5.1 Skenario

Identifikasi		
Nomor	5	
Nama	Olah Data Divisi	
Tujuan	Melakukan pengolahan data divisi	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah data divisi	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.6 Skenario Olah Data Divisi

3.3.2.5.2 Dinamika objek



Gambar 3.24 Dinamika Objek Olah Data Divisi

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.6 Olah Data Jabatan

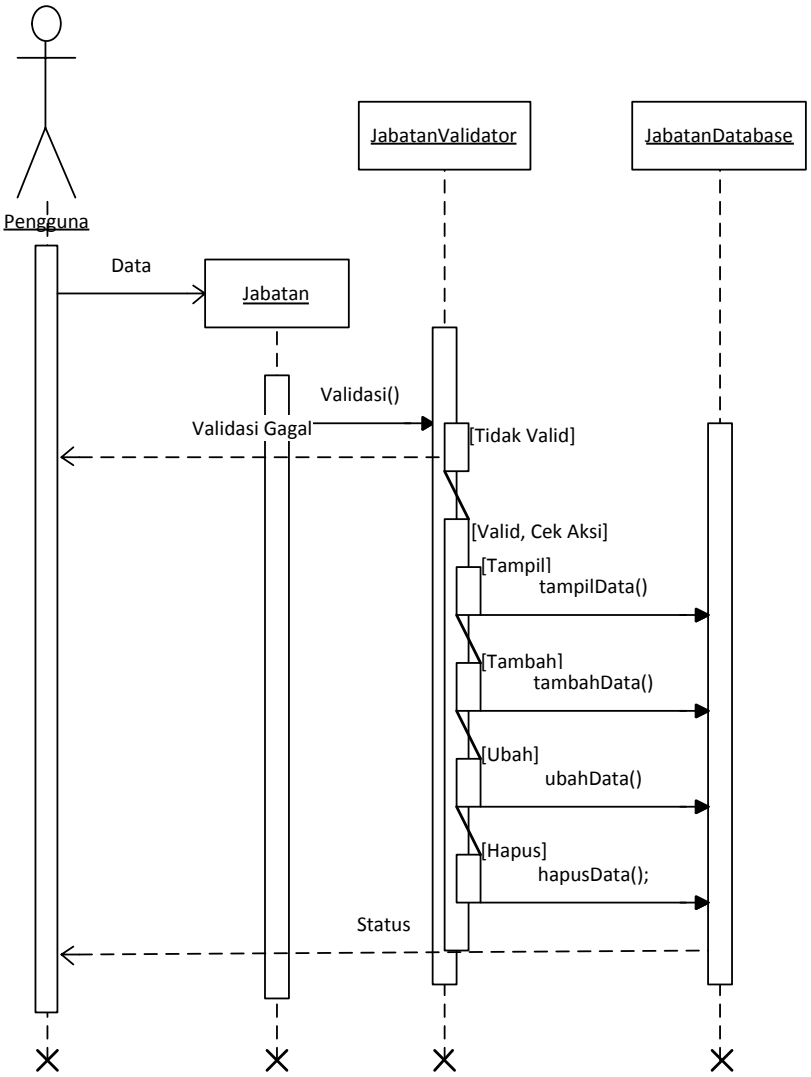
Fungsi ini merupakan proses untuk melakukan pengolahan data jabatan. Fungsi ini hanya akan dapat dijalankan jika pengguna telah melewati fungsi autentikasi.

3.3.2.6.1 Skenario

Identifikasi		
Nomor	6	
Nama	Olah Data Jabatan	
Tujuan	Melakukan pengolahan data jabatan	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah data jabatan	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.7 Skenario Olah Data Jabatan

3.3.2.6.2 Dinamika Objek



Gambar 3.25 Dinamika Objek Olah Data Jabatan

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.7 Olah Data Klien

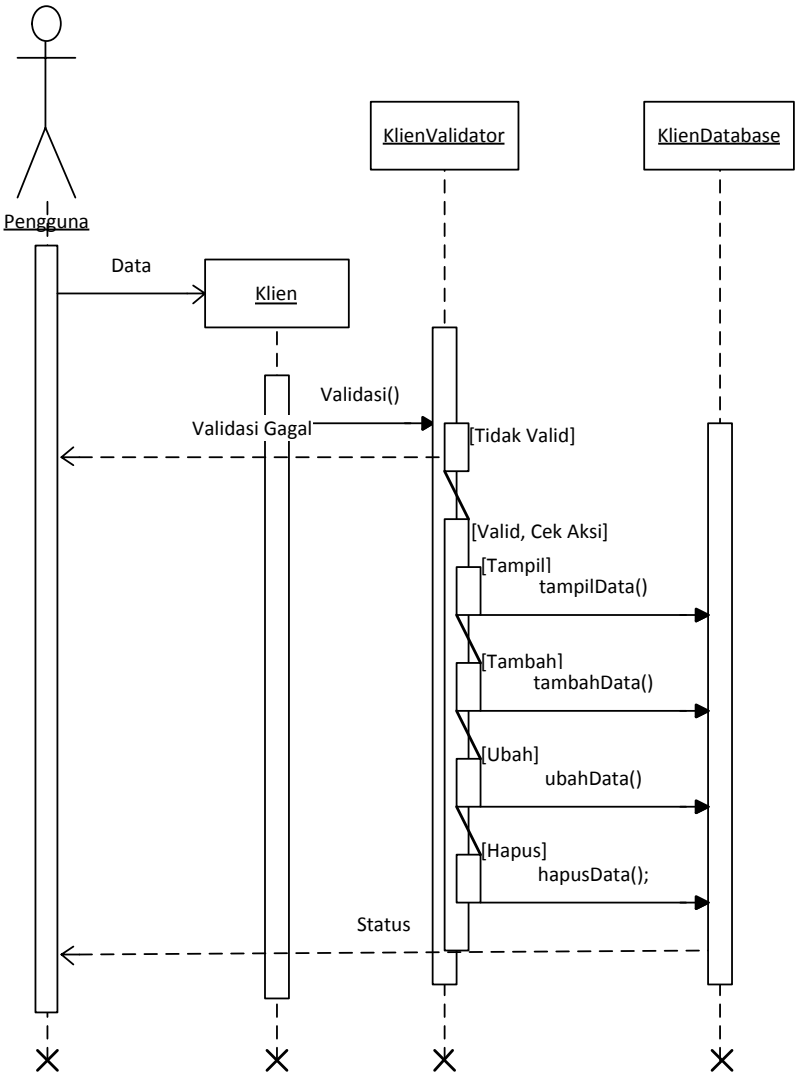
Fungsi ini merupakan proses untuk melakukan pengolahan data klien. Fungsi ini hanya data dijalankan jika pengguna telah melewati fungsi autentikasi.

3.3.2.7.1 Skenario

Identifikasi		
Nomor	7	
Nama	Olah Data Klien	
Tujuan	Melakukan pengolahan data klien	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah data klien	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.8 Skenario Pengolahan Data Klien

3.3.2.7.2 Dinamika Objek



Gambar 3.26 Dinamika Objek Olah Data Klien

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.8 Olah Data Proyek

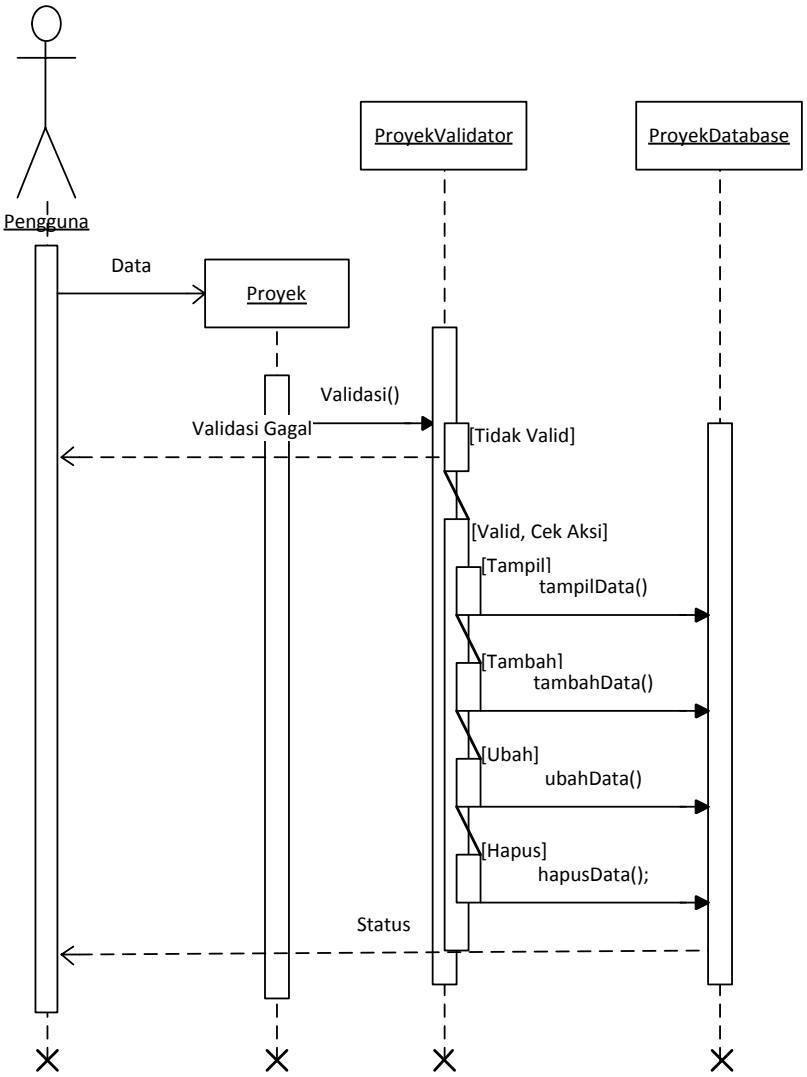
Fungsi ini merupakan fungsi untuk melakukan proses pengolahan data proyek. Fungsi ini hanya dapat dijalankan oleh pengguna yang telah melewati fungsi autentikasi.

3.3.2.8.1 Skenario

Identifikasi		
Nomor	8	
Nama	Olah Data Proyek	
Tujuan	Melakukan pengolahan data proyek	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah data proyek	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.9 Skenario Olah Data Proyek

3.3.2.8.2 Dinamika Objek



Gambar 3.27 Dinamika Objek Olah Data Proyek

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.9 Olah Data Pengguna

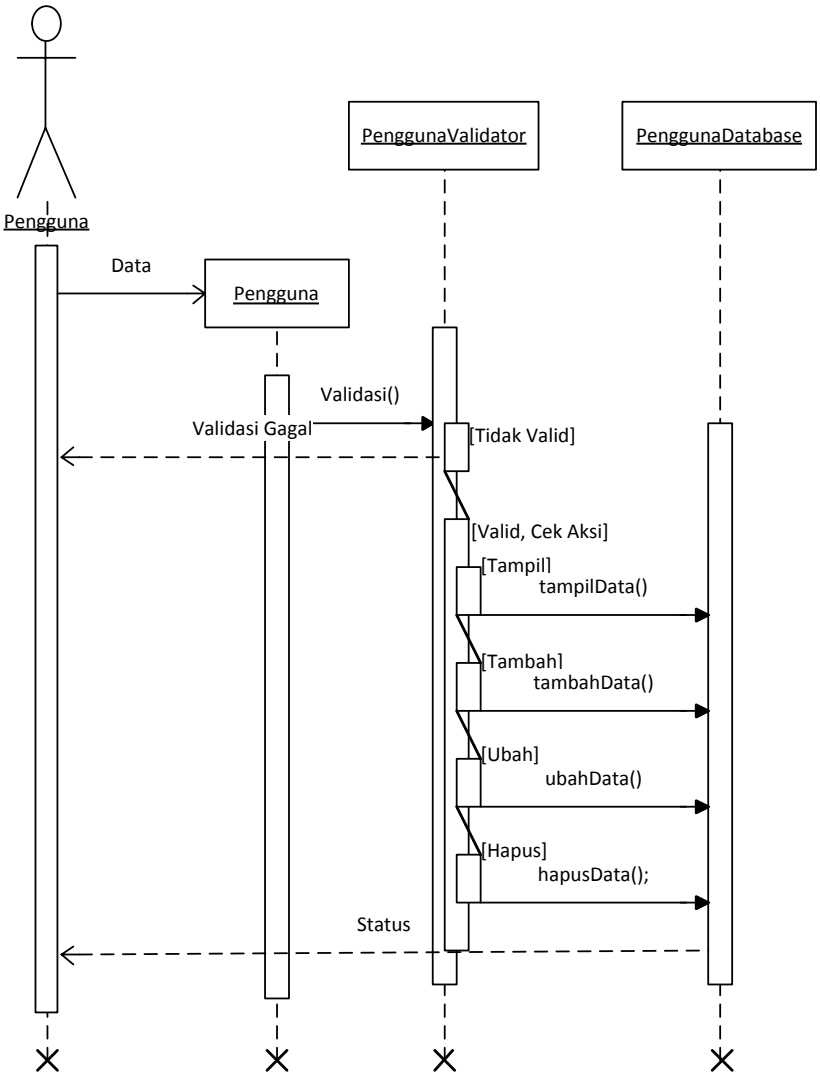
Fungsi ini merupakan proses yang digunakan untuk melakukan pengolahan data pengguna sistem penggajian. Fungsi ini hanya dapat dijalankan oleh pengguna yang telah melewati proses autentikasi.

3.3.2.9.1 Skenario

Identifikasi		
Nomor	9	
Nama	Olah Data Pengguna	
Tujuan	Melakukan pengolahan data pengguna	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah pengguna	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.10 Skenario Olah Data Pengguna

3.3.2.9.2 Dinamika Objek



Gambar 3.28 Dinamika Objek Olah Data Pengguna

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.10 Olah Data Hak Akses

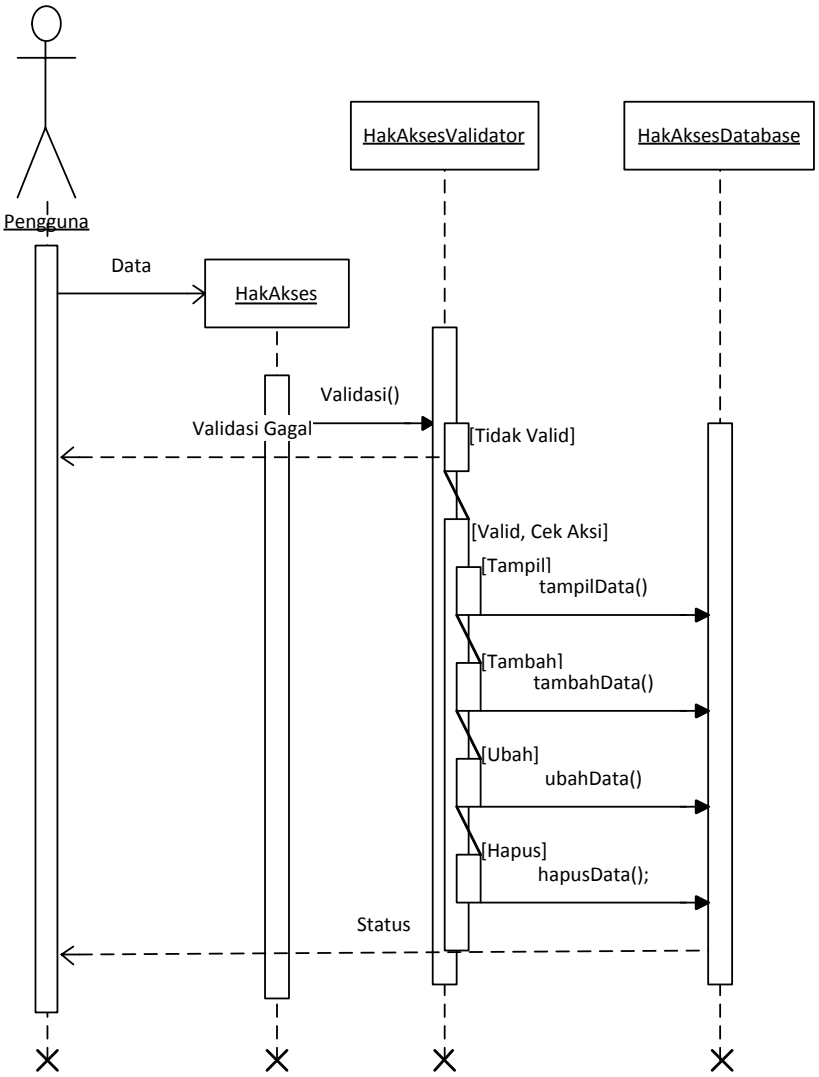
Fungsi ini merupakan proses yang dilakukan untuk mengolah data hak akses terhadap pengguna sistem penggajian. Fungsi ini hanya dapat dijalankan jika pengguna telah melewati fungsi autentikasi.

3.3.2.10.1 Skenario

Identifikasi		
Nomor	10	
Nama	Olah Data Hak Akses	
Tujuan	Melakukan pengolahan data hak akses	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah hak akses	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.11 Skenario Olah Data Hak Akses

3.3.2.10.2 Dinamika Objek



Gambar 3.29 Dinamika Objek Olah Data Hak Akses

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.11 Olah Data Kehadiran

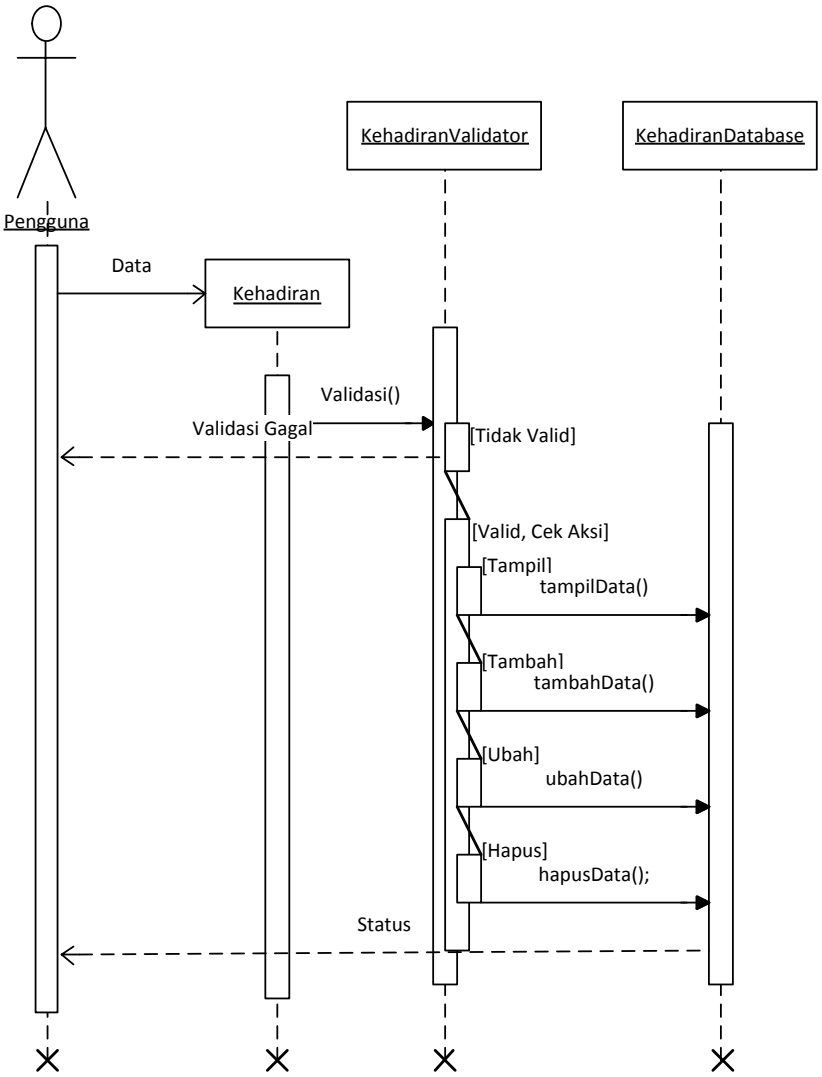
Fungsi ini merupakan proses untuk mengolah data kehadiran karyawan. Fugnsi ini hanya dapat dijalankan jika pengguna telah melewati fungsi autentikasi.

3.3.2.11.1 Skenario

Identifikasi		
Nomor	11	
Nama	Olah Data Kehadiran	
Tujuan	Melakukan pengolahan data kehadiran	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah skenario	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.12 Skenario Olah Data Kehadiran

3.3.2.11.2 Dinamika Objek



Gambar 3.30 Dinamika Objek Olah Data Kehadiran

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.12 Olah Data Transportasi

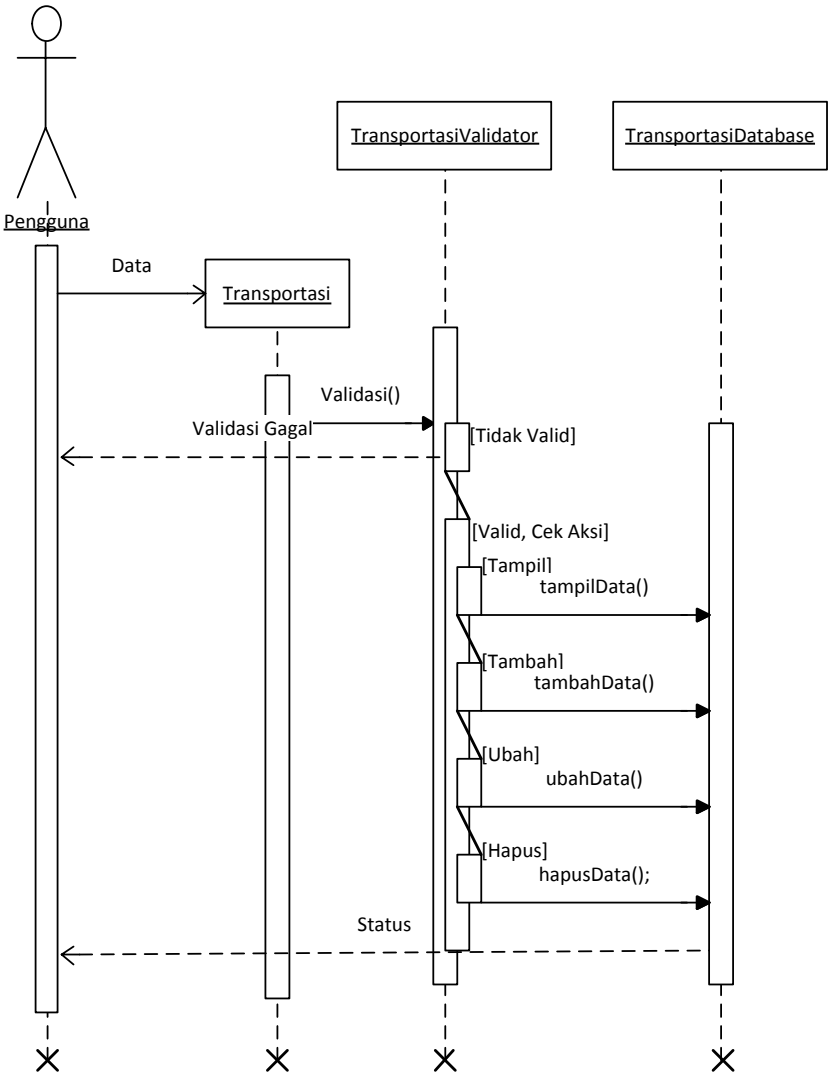
Fungsi ini merupakan fungsi yang digunakan untuk melakukan proses pengolahan data transportasi. Fugsi ini hanya dapat dijalankan jika pengguna telah melewati fungsi autentikasi.

3.3.2.12.1 Skenario

Identifikasi		
Nomor	12	
Nama	Olah Data Transportasi	
Tujuan	Melakukan pengolahan data transportasi	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah transportasi	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.13 Skenario Olah Data Transportasi

3.3.2.12.2 Dinamika Objek



Gambar 3.31 Dinamika Objek Olah Data Transportasi

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.13 Olah Data Karyawan

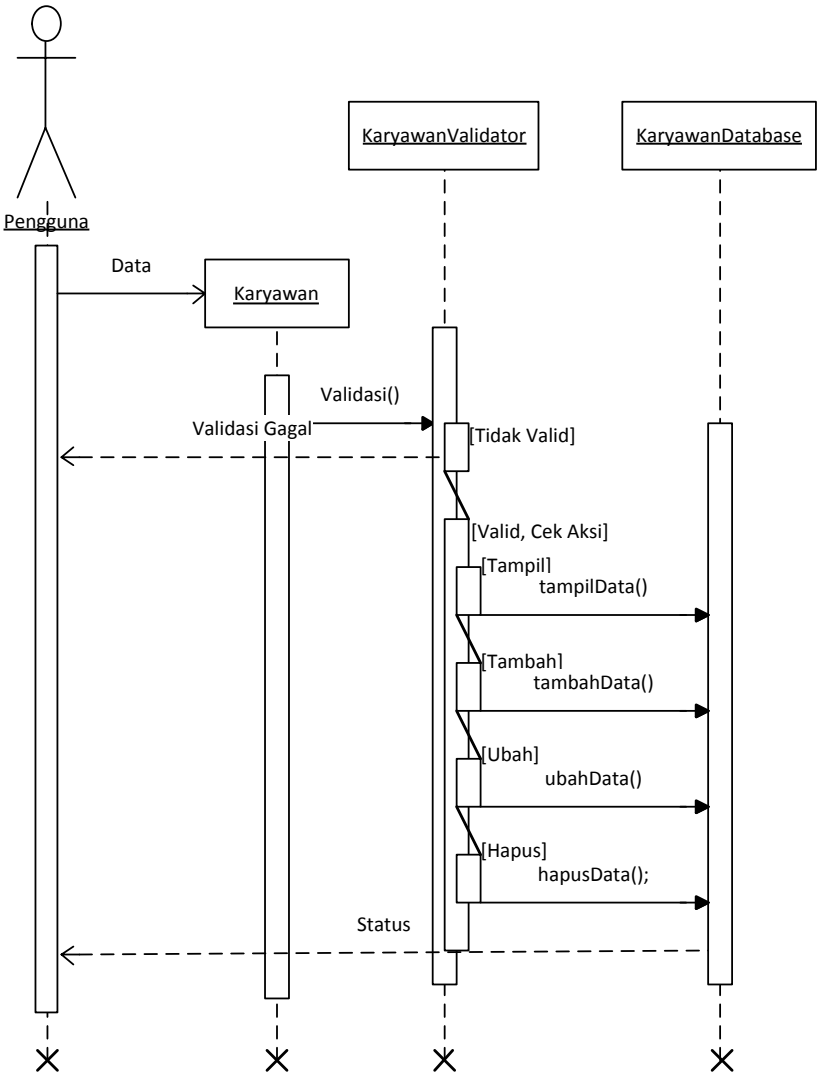
Fungsi ini merupakan fungsi yang dilakukan untuk mengolah data karyawan.Fungsi ini hanya dapat dijalankan oleh pengguna yang telah melewati fungsi autentikasi.

3.3.2.13.1 Skenario

Identifikasi		
Nomor	13	
Nama	Olah Data Karyawan	
Tujuan	Melakukan pengolahan data karyawan	
Deskripsi	Sistem akan menerima input data dan aksi yang akan dijalankan oleh pengguna, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan pengguna, jika data valid maka aksi yang diperintahkan pengguna akan dijalankan oleh sistem	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form olah karyawan	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Tampilan aplikasi akan diupdate sesuai dengan aksi yang telah dilakukan oleh pengguna	

Tabel 3.14 Skenario Olah Data Karyawan

3.3.2.13.2 Dinamika Objek



Gambar 3.32 Dinamika Objek Olah Data Karyawan

Pertama pengguna memasukkan data, lalu memerintahkan aksi yang akan dilakukan, setelah itu sistem akan melakukan proses validasi terhadap data yang dimasukkan, jika valid maka sistem akan melakukan aksi yang diperintahkan oleh pengguna.

3.3.2.14 Buat Laporan

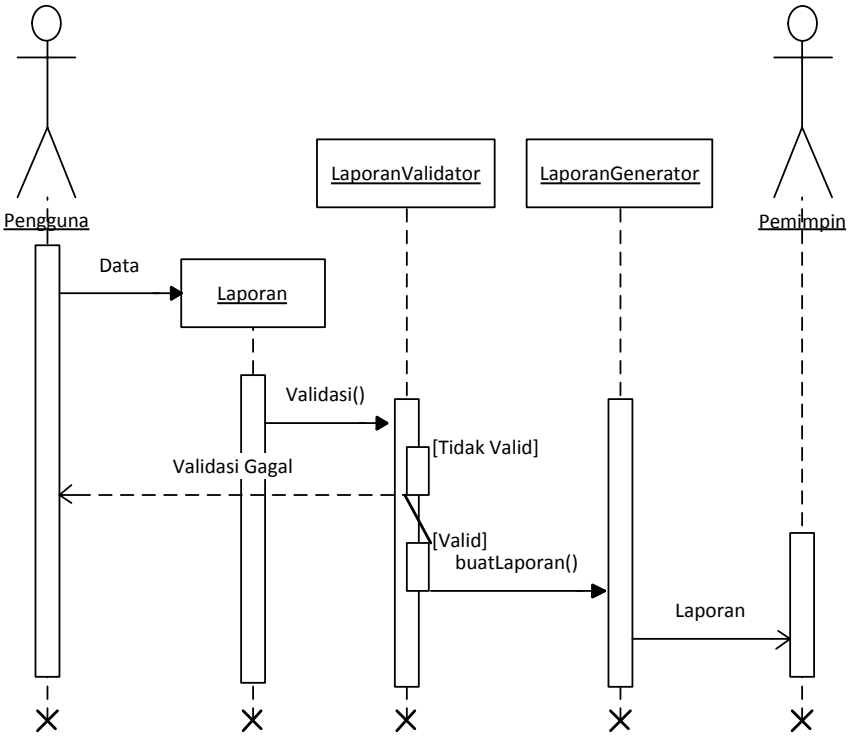
Fungsi ini merupakan fungsi yang digunakan untuk membuat laporan. Fungsi ini hanya dapat dilakukan oleh pengguna yang telah melewati proses autentikasi.

3.3.2.14.1 Skenario

Identifikasi		
Nomor	14	
Nama	Buat Laporan	
Tujuan	Membuat Laporan	
Deskripsi	Sistem menerima masukan data yang diperlukan, setelah itu sistem melakukan validasi terhadap data yang dimasukkan oleh penggua, jika data valid maka laporan akan dibuat.	
Tipe		
Aktor	Pengguna	
Skenario Utama		
Kondisi Awal	Pengguna masuk ke form buat laporan	
Aksi Aktor		**Reaksi Sistem**
Pengguna memasukkan data		Mengirim data ke sistem
Pengguna memerintah aksi yang akan dilakukan		Melakukan validasi terhadap data dan jika valid maka proses aksi yang diperintahankan akan dilakukan
Skenario Alternatif – Validasi Gagal		
Aksi Aktor		**Reaksi Sistem**
Pengguna melakukan hal yang sama		Sistem akan melakukan reaksi yang sama
Kondisi Akhir	Laporan akan ditampilkan	

Tabel 3.15 Skenario Buat Laporan

3.3.2.14.2 Dinamika Objek

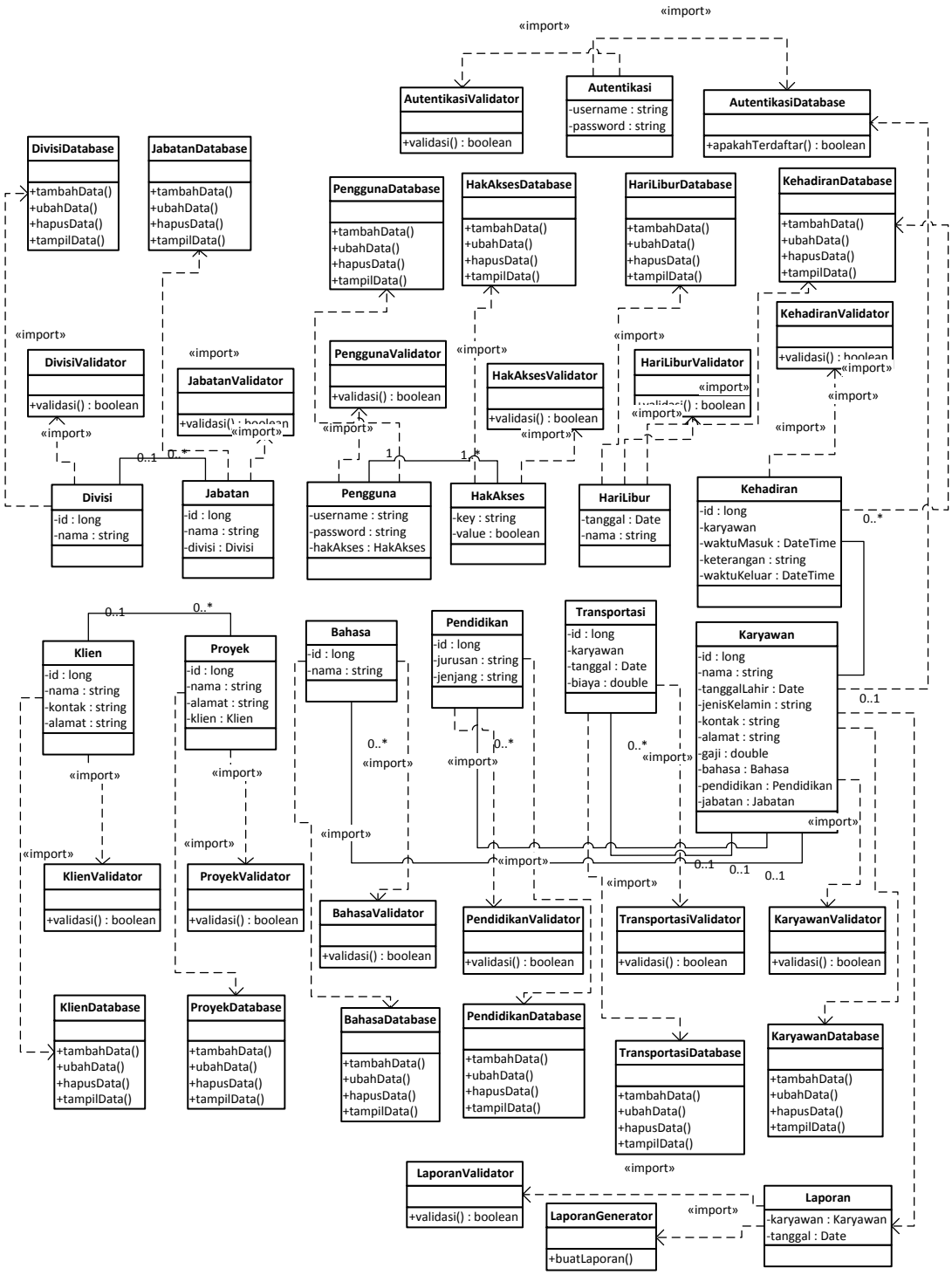


Gambar 3.33 Dinamika Objek Buat Laporan

Pertama pengguna memasukkan data yang dubutuhkan untuk membuat laporan, setelah itu sistem akan melakukan validasi terhadap data yang dimasukkan, jika valid, maka laporan akan dibuat.

3.3.3 Deskripsi Kelas – Kelas

3.3.3.1 Diagram Kelas

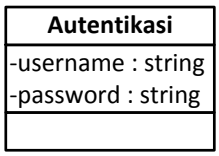


Gambar 3.34 Diagram Kelas

3.3.3.2 Spesifikasi Kelas

3.3.3.2.1 Kelas Autentikasi

Kelas Autentikasi merupakan data yang digunakan saat pengguan melakukan proses autentikasi.



Gambar 3.35 Kelas Autentikasi

3.3.3.2.1.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
username	String	45	a-z, A-Z	Null
password	String	45	-	Null

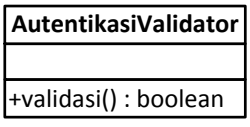
Tabel 3.16 Deskripsi Atribut Kelas Autentikasi

3.3.3.2.1.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.2 Kelas AutentikasiValidator

Kelas AutentikasiValidator merupakan kelas yang digunakan untuk memvalidasi data kelas Autentikasi.



Gambar 3.36 Kelas AutentikasiValidator

3.3.3.2.2.1 Deskripsi Atribut

Tidak memiliki atribut.

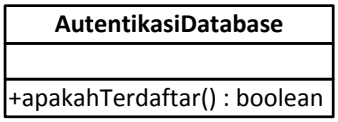
3.3.3.2.2.2 Deskripsi Layanan

Operari – operasi yang ada didalam kelas ini adalah :

1. validasi(), digunakan untuk melakukan validasi terhadap data Autentikasi.

3.3.3.2.3 Kelas AutentikasiDatabase

Kelas AutentikasiDatabase merupakan kelas yang digunakan sebagai operasi yang berhubungan dengan Autentikas dan *Database*.



Gambar 3.37 Kelas AutentikasiDatabase

3.3.3.2.3.1 Deskripsi Atribut

Tidak memiliki atribut.

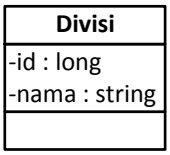
3.3.3.2.3.2 Deskripsi Layanan

Operasi – operasi yang ada pada kelas ini adalah :

1. apakahTerdaftar(), digunakan untuk mengecek apakah data Autentikasi terdaftar pada *database*.

3.3.3.2.4 Kelas Divisi

Kelas divisi merupakan kelas yang merepresentasikan Divisi di perusahaan.



Gambar 3.38 Kelas Divisi

3.3.3.2.4.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
id	Long		-	Auto Increment
nama	String	45	a-z, A-Z	Null

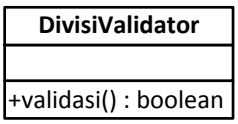
Tabel 3.17 Deskripsi Atribut Kelas Divisi

3.3.3.2.4.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.5 Kelas DivisiValidator

Kelas DivisiValidator merupakan kelas yang digunakan untuk melakukan proses validasi terhadap kelas Divisi.



Gambar 3.39 Kelas DivisiValidator

3.3.3.2.5.1 Deskripsi Atribut

Tidak Memiliki Atribut

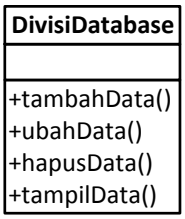
3.3.3.2.5.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data Divisi.

3.3.3.2.5.3 Kelas DivisiDatabase

Kelas DivisiDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data di *database* dari kelas Divisi.



Gambar 3.40 Kelas DivisiDatabase

3.3.3.2.5.4 Deskripsi Atribut

Tidak memiliki atribut.

3.3.3.2.5.5 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. tambahData(), digunakan untuk menambah data divisi yang ada di *database*.
2. ubahData(), digunakan untuk mengubah data divisi yang ada di *database*.
3. hapurData(), digunakan untuk menghapus data divisi yang ada di *database*.

4. tampilDate(), digunakan untuk menampilkan data divisi yang terdapat pada *database*.

3.3.3.2.6 Kelas Jabatan

Kelas Jabatan merupakan kelas yang merepresentasikan jabatan di perusahaan.



Gambar 3.41 Kelas Jabatan

3.3.3.2.6.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
id	Long	-	-	Auto Increment
nama	String	45	a-z, A-Z	Null
divisi	Divisi	-	-	Null

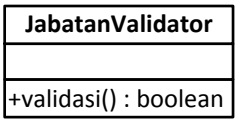
Tabel 3.18 Deskripsi Atribut Kelas Jabatan

3.3.3.2.6.2 Deskripsi Layanan

Tidak memiliki layanan

3.3.3.2.7 Kelas JabatanValidator

Kelas JabatanValidator merupakan kelas yang digunakan untuk melakukan validasi terhadap data Jabatan.



Gambar 3.42 Kelas JabatanValidator

3.3.3.2.7.1 Deskripsi Atribut

Tidak memiliki atribut.

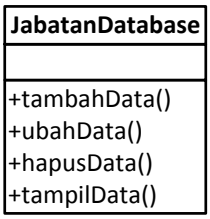
3.3.3.2.7.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data Jabatan.

3.3.3.2.8 Kelas JabatanDatabase

Kelas JabatanDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi *database* terhadap data kelas Jabatan.



Gambar 3.43 Kelas JabatanDatabase

3.3.3.2.8.1 Deskripsi Atribut

Tidak memiliki atribut.

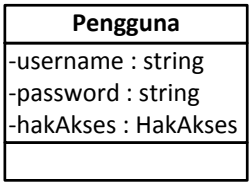
3.3.3.2.8.2 Deskripsi Layanan

Operasi – operasi yang dimiliki kelas ini :

1. tambahData(), digunakan untuk menambah data jabatan ke *database*.
2. ubahData(), digunakan untuk mengubah data jabatan di *database*.
3. hapusData(), digunakan untuk menghapus data jabatan di *database*.
4. tampilData(), digunakan untuk menampilkan data jabatan yang ada di *database*.

3.3.3.2.9 Kelas Pengguna

Kelas pengguna merupakan kelas yang merepresentasikan aktor pengguna pada sistem penggajian ini.



Gambar 3.44 Kelas Pengguna

3.3.3.2.9.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
username	String	45	a-z, A-Z	Null
Password	String	45	a-z, A-Z	Null
hakAkses	Array of HakAkses	-	-	Null

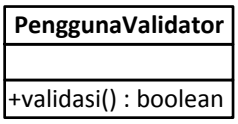
Tabel 3.19 Deskripsi Atribut Kelas Pengguna

3.3.3.2.9.2 Deskrispsi Layanan

Tidak memiliki layanan.

3.3.3.2.10 Kelas PenggunaValidator

Kelas PenggunaValidator merupakan kelas yang digunakan untuk melakukan validasi terhadap data kelas pengguna.



Gambar 3.45 Kelas PenggunaValidator

3.3.3.2.10.1 Deskrisp Atribut

Tidak memiliki atribut.

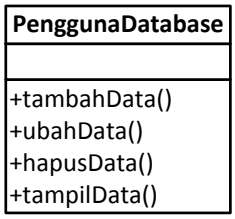
3.3.3.2.10.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data Pengguna.

3.3.3.2.11 Kelas PenggunaDatabase

Kelas PenggunaDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data pengguna yang ada pada *database*.



Gambar 3.46 Kelas PenggunaDatabase

3.3.3.2.11.1 Deskripsi Atribut

Tidak memiliki atribut.

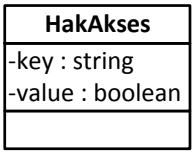
3.3.3.2.11.2 Deskripsi Layanan

Operas – operasi yang terdapat pada kelas ini :

1. tambahData(), digunakan untuk menambah data pengguna ke *database*.
2. ubahData(), digunakan untuk mengubah data pengguna di *database*.
3. hapusData(), digunakan untuk menghapus data pengguna di *database*.
4. tampilData(), digunakan untuk menampilkan data yang terdapat pada *database*.

Kelas HakAkses

Kelas HakAkses merupakan kelas yang merepresentasikan hak akses pengguna pada sistem penggajian ini.



Gambar 3.47 Kelas HakAkses

3.3.3.2.11.3 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
key	String	45	a-z, A-Z	Null
value	Boolean	-	-	false

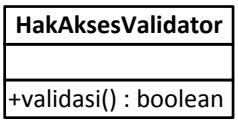
Tabel 3.20 Deskripsi Atribut Kelas Hak Akses

3.3.3.2.11.4 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.12 Kelas HakAksesValidator

Kelas HakAksesValidator merupakan kelas yang digunakan untuk memvalidasi data HakAkses.



Gambar 3.48 Kelas HakAksesValidator

3.3.3.2.12.1 Deskripsi Atribut

Tidak memiliki atribut.

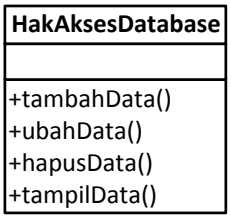
3.3.3.2.12.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data HakAkses.

3.3.3.2.13 Kelas HakAksesDatabase

Kelas HakAksesDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data HakAkses di *database*.



Gambar 3.49 Kelas HakAksesDatabase

3.3.3.2.13.1 Deskripsi Atribut

Tidak memiliki atribut.

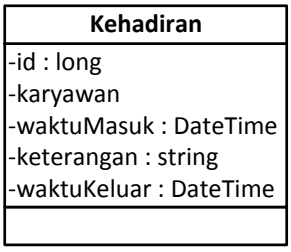
3.3.3.2.13.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. tambahData(), digunakan untuk menambah data hak akses ke *database*.
2. ubahData(), digunakan untuk mengubah data yang ada hak akses di *database*.
3. hapusData(), digunakan untuk menghapus data hak akses yang ada di *database*.
4. tampilData(), digunakan untuk menampilkan data hak akses yang ada di *database*.

3.3.3.2.14 Kelas Kehadiran

Kelas Kehadiran merupakan kelas yang merepresentasikan data kehadiran.



Gambar 3.50 Kelas Kehadiran

3.3.3.2.14.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Id	Long	-	-	Auto increment
Karyawan	Karyawan	-	-	Null

waktuMasuk	DateTime	-	-	Null
waktuKeluar	DateTime	-	-	Null
Keterangan	String	255	a-z, A-Z	Null

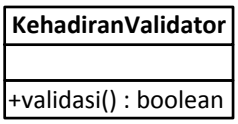
Tabel 3.21 Deskripsi Atribut Kelas Kehadiran

3.3.3.2.14.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.15 Kelas KehadiranValidator

Kelas kehadiran validator merupakan kelas yang digunakan untuk melakukan validasi terhadap kelas Kehadiran



Gambar 3.51 Kelas KehadiranValidator

3.3.3.2.15.1 Deskripsi Atribut

Tidak memiliki atribut.

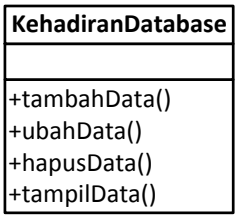
3.3.3.2.15.2 Dekripsi Layanan

Operasi – operasi yang dimiliki kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data Kehadiran.

3.3.3.2.16 Kelas KehadiranDatabase

Kelas KehadiranDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data kehadiran di *database*.



Gambar 3.52 Kelas KehadiranDatabase

3.3.3.2.16.1 Deskripsi Atribut

Tidak memiliki atribut.

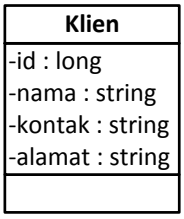
3.3.3.2.16.2 Deskripsi Layanan

Operasi – operasi yang dimiliki kelas ini :

1. tambahData(), digunakan untuk menambah data Kehadiran ke *database*.
2. ubahData(), digunakan untuk mengubah data kehadiran di *database*.
3. hapusData(), digunakan untuk mengubah data kehadiran di *database*.
4. tampilData(), digunakan untuk menampilkan data kehadiran dari *database*.

3.3.3.2.17 Kelas Klien

Kelas Klien merupakan kelas yang merepresentasikan data klien di perusahaan.



Gambar 3.53 Kelas Klien

3.3.3.2.17.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Id	Long	-	-	Auto increment
Nama	String	45	a-z, A-Z	Null
Kontak	String	25	a-z,A-Z, @ .	Null
Alamat	String	255		Null

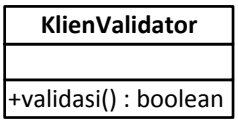
Tabel 3.22 Deskripsi Atribut Klien

3.3.3.2.17.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.18 Kelas KlienValidator

Kelas KlienValidator merupakan kelas yang digunakan untuk melakukan validasi terhadap data Klien.



Gambar 3.54 Kelas KlienValidator

3.3.3.2.18.1 Deskripsi Atribut

Tidak memiliki atribut

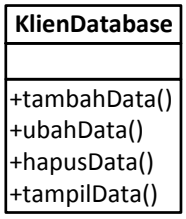
3.3.3.2.18.2 Deskripsi Layanan

Operasi – operasi yang dimiliki kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data Klien.

3.3.3.2.19 Kelas KlienDatabase

Kelas KlienDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data klien yang ada dalam *database*.



Gambar 3.55 Kelas KlienDatabase

3.3.3.2.19.1 Deskripsi Atribut

Tidak memiliki atribut.

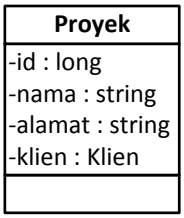
3.3.3.2.19.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. tambahData(), digunakan untuk menambah data klien ke *database*.
2. ubahData(), digunakan untuk mengubah data klien yang ada di *database*.
3. hapusData(), digunakan untuk menghapus data klien yang ada di *database*.
4. tampilData(), digunakan untuk menampilkan data klein yang ada di *database*.

3.3.3.2.20 Kelas Proyek

Kelas Proyek merupakan kelas yang merepresentasikan proyek di perusahaan.



Gambar 3.56 Kelas Proyek

3.3.3.2.20.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Id	Long	-	-	Auto increment
Nama	String	45	a-z, A-Z	Null
Alamat	String	255		Null
Klien	Klien			Null

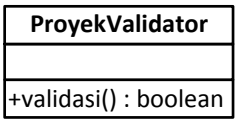
Tabel 3.23 Deskripsi Atribut Kelas Proyek

3.3.3.2.20.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.21 Kelas ProyekValidator

Kelas ProyekValidator merupakan kelas yang digunakan untuk melakukan prose validasi terhadap data Proyek.



Gambar 3.57 Kelas ProyekValidator

3.3.3.2.21.1 Deskripsi Atribut

Tidak memiliki atribut.

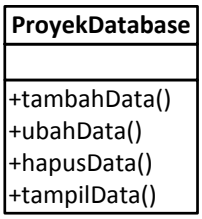
3.3.3.2.21.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data proyek.

3.3.3.2.22 Kelas ProyekDatabase

Kelas ProyekDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data proyek di *database*.



Gambar 3.58 Kelas ProyekDatabase

3.3.3.2.22.1 Deskripsi Atribut

Tidak memiliki atribut.

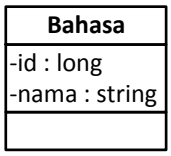
3.3.3.2.22.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. tambahData(), digunakan untuk menambah data proyek ke *database*.
2. ubahData(), digunakan untuk mengubah data proyek di *database*.
3. hapusData(), digunakan untuk menghapus data proyek di *database*.
4. tampilData(), diguanakan untuk menampilkan data proyek yang ada di *database*.

3.3.3.2.23 Kelas Bahasa

Kelas Bahasa merupakan kelas yang merepresentasikan bahasa.



Gambar 3.59 Kelas Bahasa

3.3.3.2.23.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Id	Long	-	-	Auto increment
Nama	String	45	a-z, A-Z	Null

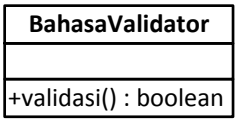
Tabel 3.24 Deskripsi Atribut Kelas Bahasa

3.3.3.2.23.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.24 Kelas BahasaValidator

Kelas BahasaValidator merupakan kelas yang digunakan untuk melakukan validasi terhadap data bahasa.



Gambar 3.60 Kelas BahasaValidator

3.3.3.2.24.1 Deskripsi Atribut

Tidak memiliki atribut.

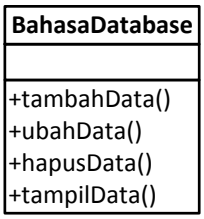
3.3.3.2.24.2 Deskripsi Layanan

Operasi – operasi yang ada pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data bahasa.

3.3.3.2.25 Kelas BahasaDatabase

Kelas BahasaDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data bahasa di *database*.



Gambar 3.61 Kelas BahasaDatabase

3.3.3.2.25.1 Deskripsi Atribut

Tidak memiliki atribut.

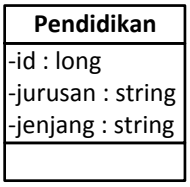
3.3.3.2.25.2 Deskripsi Layanan

Operasi – operasi yang terdapat di kelas ini :

1. tambahData(), digunakan untuk menambah data bahasa ke *database*.
2. ubahData(), digunakan untuk mengubah data bahasa di *database*.
3. hapusData(), digunakan untuk menghapus data bahasa di *database*.
4. tampilData(), digunakan untuk menampilkan data bahasa yang ada di *database*.

3.3.3.2.26 Kelas Pendidikan

Kelas pendidikan merupakan kelas yang merepresentasikan data pendidikan.



Gambar 3.62 Kelas Pendidikan

3.3.3.2.26.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Id	Long	-	-	Auto increment
Jurusan	String	45	a-z, A-Z	Null
Jenjang	String	10	a-z,A-Z,0-9	Null

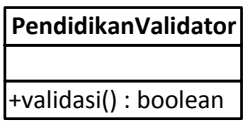
Tabel 3.25 Deskripsi Atribut Kelas Pendidikan

3.3.3.2.26.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.27 Kelas PendidikanValidator

Kelas PendidikanValidator merupakan kelas yang digunakan untuk melakukan proses validasi terhadap data pendidikan.



Gambar 3.63 Kelas PendidikanValidator

3.3.3.2.27.1 Deskripsi Atribut

Tidak memiliki atribut.

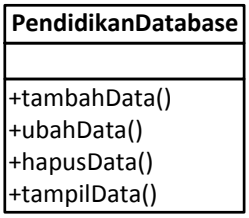
3.3.3.2.27.2 Deskripsi Layanan

Operasi – operasi yang terdapat di kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data pendidikan.

3.3.3.2.28 Kelas PendidikanDatabase

Kelas PendidikanDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data pendidikan di *database*.



Gambar 3.64 Kelas PendidikanDatabase

3.3.3.2.28.1 Deskripsi Atribut

Tidak memiliki atribut.

3.3.3.2.28.2 Deskripsi Layanan

Operasi – operasi yang ada pada kelas ini :

1. tambahData(), digunakan untuk menambah data pendidikan ke *database*.
2. ubahData(), digunakan untuk mengubah data pendidikan di *database*.

3. hapusData(), digunakan untuk menghapus data pendidikan di *database*.
4. tampilData(), digunakan untuk menampilkan data pendidikan yang ada di *database*.

3.3.3.2.29 Kelas Transportasi

Kelas Transportasi merupakan kelas yang merepresentasikan data transportasi.



Gambar 3.65 Kelas Transportasi

3.3.3.2.29.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Id	Long	-	-	Auto increment
Karyawan	Karyawan	-	-	Null
Tanggal	Date	-	-	Null
Biaya	Double	-	-	0.0

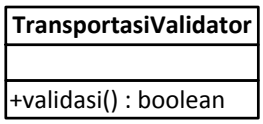
Tabel 3.26 Deskripsi Atribut Kelas Transportasi

3.3.3.2.29.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.30 Kelas TransportasiValidator

Kelas TransportasiValidator merupakan kelas yang digunakan untuk melakukan validasi terhadap data transportasi.



Gambar 3.66 Kelas Transportasi

3.3.3.2.30.1 Deskripsi Atribut

Tidak memiliki atribut.

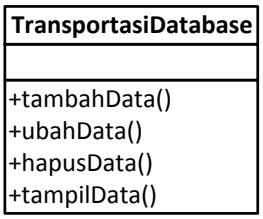
3.3.3.2.30.2 Deskripsi Layanan

Operasi – operasi yang ada pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data transportasi.

3.3.3.2.31 Kelas TransportasiDatabase

Kelas TransportasiDatabase merupakan kelas yang digunakan untuk melakukan proses manipulasi data di *database*.



Gambar 3.67 Kelas TransportasiDatabase

3.3.3.2.31.1 Deskripsi Atribut

Tidak memiliki atribut.

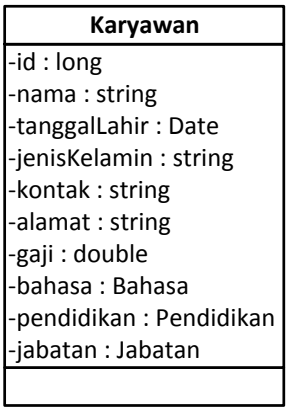
3.3.3.2.31.2 Deskripsi Layanan

Operasi – operasi yang dimiliki oleh kelas ini :

1. tambahData(), digunakan untuk menambah data transportasi ke *database*.
2. ubahData(), digunakan untuk mengubah data transportasi di *database*.
3. hapusData(), digunakan untuk menghapus data transportasi di *database*.
4. tampilData(), digunakan untuk menampilkan data transportasi yang ada di *database*.

3.3.3.2.32 Kelas Karyawan

Kelas karyawan merupakan kelas yang merepresentasikan karyawan di perusahaan.



Gambar 3.68 Kelas Karyawan

3.3.3.2.32.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Id	Long	-	-	Auto increment
Nama	String	45	a-z, A-Z	Null
tanggalLahir	Date	-	-	Null
jenisKelamin	String	1	‘L’,’P’	Null
Kontak	String	45	a-z,A-Z,0-9	Null

Alamat	String	255	a-z,A-Z,0-9	Null
Gaji	Double	-	-	0
Bahasa	Array of Bahasa	-	-	Null
Pendidikan	Array of Pendidikan	-	-	Null
Jabatan	Jabatan	-	-	-

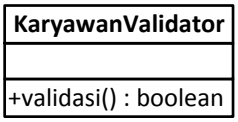
Tabel 3.27 Deskripsi Atribut Kelas Karyawan

3.3.3.2.32.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.33 Kelas KaryawanValidator

Kelas KaryawanValidator merupakan kelas yang digunakan untuk melakukan proses validasi terhadap data karyawan.



Gambar 3.69 Kelas KaryawanValidator

3.3.3.2.33.1 Deskripsi Atribut

Tidak memiliki atribut

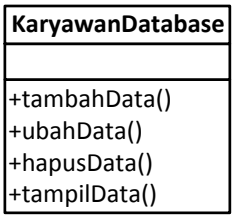
3.3.3.2.33.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. validasi(), digunakan untuk melakukan proses validasi terhadap data karyawan.

3.3.3.2.34 Kelas KaryawanDatabase

Kelas KaryawanDatabase digunakan untuk melakukan proses manipulasi data karyawan di *database*.



Gambar 3.70 Kelas KaryawanDatabase

3.3.3.2.34.1 Deskripsi Atribut

Tidak memiliki atribut..

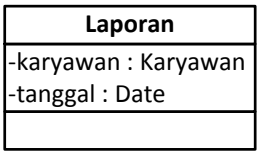
3.3.3.2.34.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

1. tambahData(), digunakan untuk menambah data karyawan ke *database*.
2. ubahData(), digunakan untuk mengubah data karyawan di *database*.
3. hapusData(), digunakan untuk menghapus data karyawan di *database*.
4. tampilData(), digunakan untuk menampilkan data karyawan yang ada di *database*.

3.3.3.2.35 Kelas Laporan

Kelas laporan merupakan kelas yang digunakan untuk membuat laporan.



Gambar 3.71 Kelas Laporan

3.3.3.2.35.1 Deskripsi Atribut

Nama	Tipe	Panjang	Range	Nilai Tetap
Karyawan	Karyawan	-	-	Null
Tanggal	Date	-	-	Null

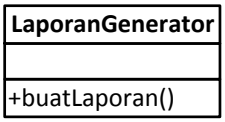
Tabel 3.28 Deskripsi Atribut Kelas Laporan

3.3.3.2.35.2 Deskripsi Layanan

Tidak memiliki layanan.

3.3.3.2.36 Kelas LaporanGenerator

Kelas LaporanGenerator merupakan kelas yang digunakan untuk membuat laporan.



Gambar 3.72 Kelas LaporanGenerator

3.3.3.2.36.1 Deskripsi Atribut

Tidak memiliki atribut.

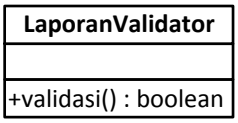
3.3.3.2.36.2 Deskripsi Layanan

Operasi – operasi yang ada di kelas ini :

1. buatLaporan(), digunakan untuk membuat laporan.

3.3.3.2.37 Kelas LaporanValidator

Kelas LaporanValidator merupakan kelas yang digunakan untuk melakukan proses validasi terhadap data laporan



Gambar 3.73 Kelas LaporanValidator

3.3.3.2.37.1 Deskripsi Atribut

Tidak memiliki atribut.

3.3.3.2.37.2 Deskripsi Layanan

Operasi – operasi yang terdapat pada kelas ini :

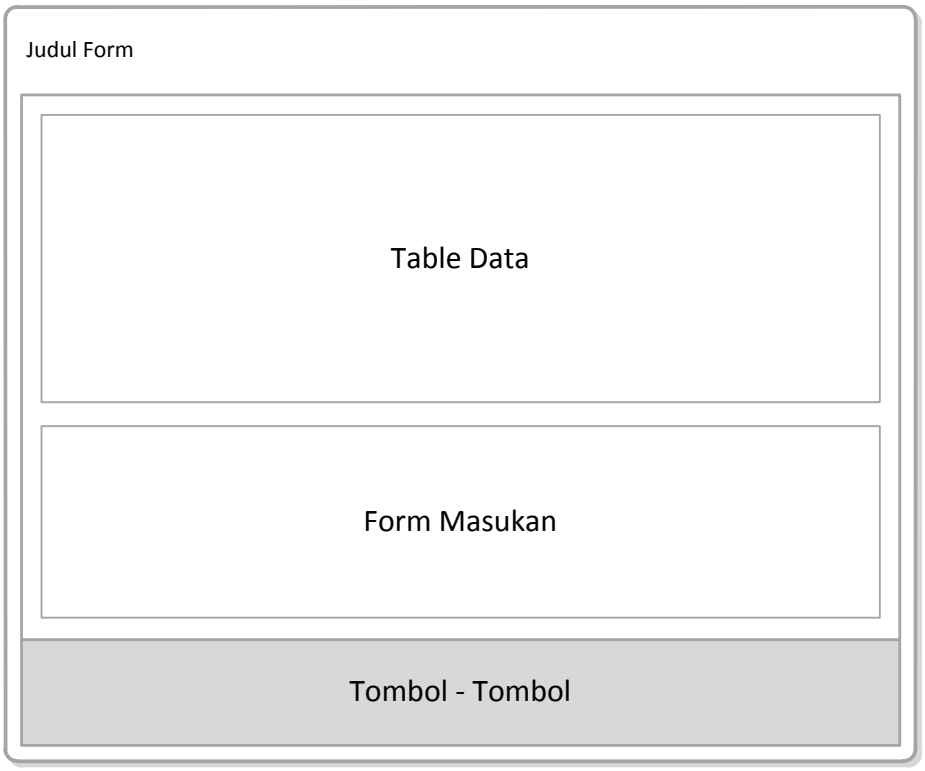
1. validasi(), digunakan untuk melakukan proses validasi terhadap data laporan.

3.3.4 Deskripsi Perancangan

Bagian ini berikikan rancangan lengkap tentang sistem informasi penggajian yang akan dikembangkan. Dari mulai deskripsi layar dan diagram semantik.

3.3.4.1 Deskripsi Layar

Secara umum layout akan seperti pada gambar dibawah ini :



Gambar 3.74 Deskripsi Layar

Pada bagian Table Data akan berisikan tentang *grid* atau biasa disebut tabel, fungsi table ini akan menampilkan seluruh data yang ada. *Form* masukan akan berisikan alat input yang dapat diisi oleh pengguna dan pada bagian tombol – tombol akan berisikan tombol yang memuat aksi.

3.3.4.1.1 Autentikasi



Gambar 3.75 Deskripsi Layar Autentikasi

3.3.4.1.1.1 Deskripsi Objek

Objek	Jenis	Keterangan
Gambar	Image	Icon gambar login
Nama Pengguna	TextBox	Masukkan data nama pengguna
Kata Sandi	TextBox	Masukkan data kata sandi
Masuk	Button	Tombol aksi masuk
Keluar	Button	Tombol aksi keluar

Tabel 3.29 Deskripsi Objek Autentikasi

3.3.4.1.1.2 Algoritma

Input : nama pengguna dan kata sandi

Proses :

1. Jika tombol masuk di tekan maka, sistem akan melakukan proses autentikasi

2. Jika tombol keluar ditekan kala sistem akan menutup aplikasi.

3.3.4.1.2 Olah Data Pendidikan



Gambar 3.76 Deskripsi Layar Olah Data Pendidikan

3.3.4.1.2.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Pendidikan	Table	Tabel yang berisikan data pendidikan
Jurusan	Textbox	Masukkan data jurusan
Jenjang	Textbox	Masukkan data jenjang
Simpan	Button	Tombol aksi simpan data pendidikan
Ubah	Button	Tombol aksi ubah data pendidikan
Hapus	Button	Tombol aksi hapus data pendidikan
Tutup	Button	Tombol aksi keluar

Gambar 3.77 Deskripsi Objek Olah Data Pendidikan

3.3.4.1.2.2 Algoritma

Input : jurusan dan jenjang

Proses :

1. Jika tombol simpan ditekan maka sistem akan menyimpan data pendidikan yang dimasukkan.
2. Jika tombol ubah ditekan maka sistem akan mengubah data yang terseleksi di tabel yang ada di *database* dengan data yang dimasukkan.
3. Jika tombol hapus ditekan maka sistem akan menghapus data yang terseleksi di tabel dari *database*.
4. Jika tombol keluar ditekan maka *form* olah data pendidikan akan ditutup.

3.3.4.1.3 Olah Data Hari Libur



Gambar 3.78 Deskripsi Layar Olah Data Hari Libur

3.3.4.1.3.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Hari Libur	Table	Tabel yang berisikan data hari libur
Tanggal	Textbox	Masukkan data tanggal
Nama	Textbox	Masukkan data nama
Simpan	Button	Tombol aksi simpan data hari libur

Ubah	Button	Tombol aksi ubah data hari libur
Hapus	Button	Tombol aksi hapus data hari libur
Tutup	Button	Tombol aksi keluar

Tabel 3.30 Deskripsi Objek Olah Data Bahasa

3.3.4.1.3.2 Algoritma

Input : tanggal dan nama

Proses :

1. Jika tombol simpan ditekan maka sistem akan menyimpan data hari libur yang dimasukkan.
2. Jika tombol ubah ditekan maka sistem akan mengubah data yang terseleksi di tabel yang ada di *database* dengan data yang dimasukkan.
3. Jika tombol hapus ditekan maka sistem akan menghapus data yang terseleksi di tabel dari *database*.
4. Jika tombol keluar ditekan maka *form* olah data hari libur akan ditutup.

3.3.4.1.4 Olah Data Bahasa



Gambar 3.79 Deskripsi Layar Olah Data Bahasa

3.3.4.1.4.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Bahasa	Table	Tabel yang berisikan data bahasa
Nama	Textbox	Masukkan data nama bahasa
Simpan	Button	Tombol aksi simpan data bahasa
Ubah	Button	Tombol aksi ubah data bahasa
Hapus	Button	Tombol aksi hapus data bahasa
Tutup	Button	Tombol aksi keluar

Tabel 3.31 Deskripsi Objek Oleh Data Bahasa

3.3.4.1.4.2 Algoritma

Inpu : nama

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data bahasa yang ke *database*.

2. Jika ditekan tombol ubah maka sistem akan mengubah data bahasa yang terseleksi di table dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data bahasa yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data bahasa.

3.3.4.1.5 Olah Data Divisi



Gambar 3.80 Deskripsi Layar Olah Data Divisi

3.3.4.1.5.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Divisi	Table	Tabel yang berisikan data divisi
Nama	Textbox	Masukkan data nama divisi
Simpan	Button	Tombol aksi simpan data divisi
Ubah	Button	Tombol aksi ubah data divisi

Hapus	Button	Tombol aksi hapus data divisi
Tutup	Button	Tombol aksi keluar

Tabel 3.32 Deskripsi Objek Olah Data Divisi

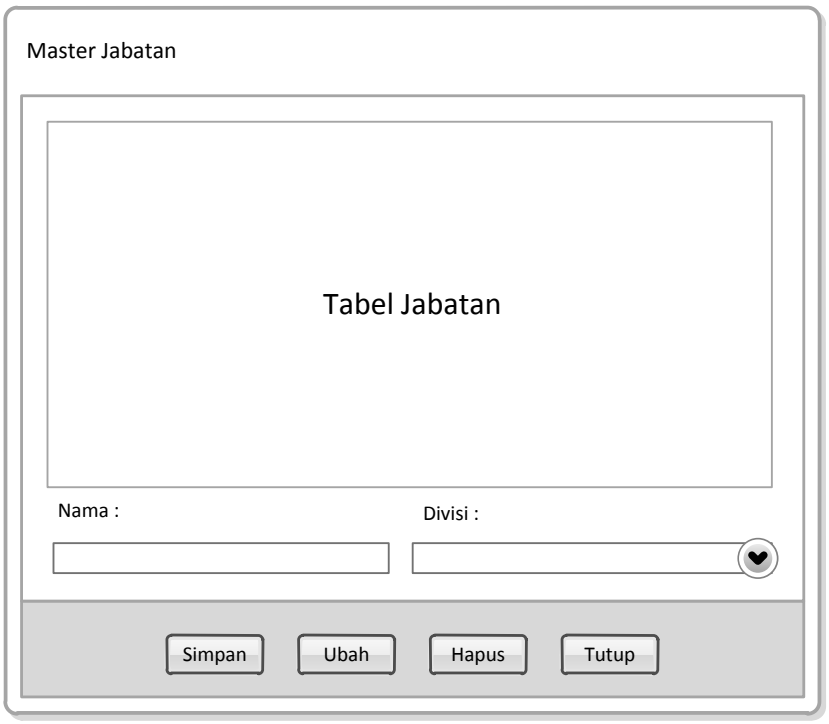
3.3.4.1.5.2 Algoritma

Input : nama

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data divisi yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data divisi yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data divisi yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data divisi.

3.3.4.1.6 Olah Data Jabatan



Gambar 3.81 Deskripsi Layar Olah Data Jabatan

3.3.4.1.6.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Jabatan	Table	Tabel yang berisikan data jabatan
Nama	Textbox	Masukkan data nama jabatan
Divisi	ComboBox	Lookup data dari database Divisi
Simpan	Button	Tombol aksi simpan data jabatan
Ubah	Button	Tombol aksi ubah data jabatan
Hapus	Button	Tombol aksi hapus data jabatan
Tutup	Button	Tombol aksi keluar

Tabel 3.33 Deskripsi Objek Olah Data Jabatan

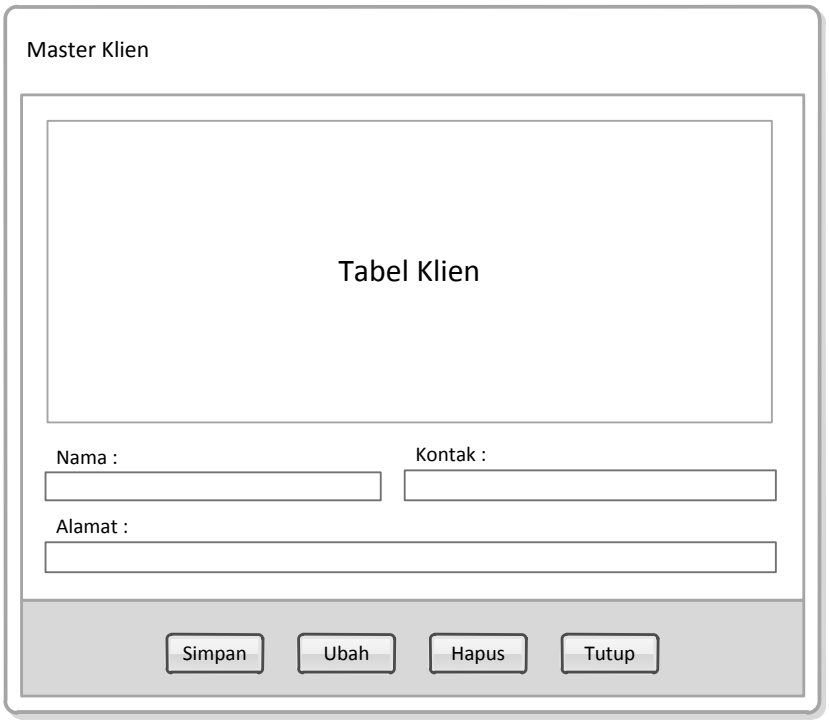
3.3.4.1.6.2 Algoritma

Input : nama dan divisi

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data jabatan yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data jabatan yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data jabatan yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data jabatan.

3.3.4.1.7 Olah Data Klien



Gambar 3.82 Deskripsi Layar Olah Data Klien

3.3.4.1.7.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Klien	Table	Tabel yang berisikan data klien
Nama	Textbox	Masukkan data nama klien
Kontak	Textbox	Masukkan data kontak klien
Alamat	Textbox	Masukkan data alamat klien
Simpan	Button	Tombol aksi simpan data jabatan
Ubah	Button	Tombol aksi ubah data jabatan
Hapus	Button	Tombol aksi hapus data jabatan
Tutup	Button	Tombol aksi keluar

Tabel 3.34 Deskripsi Objek Olah Data Klien

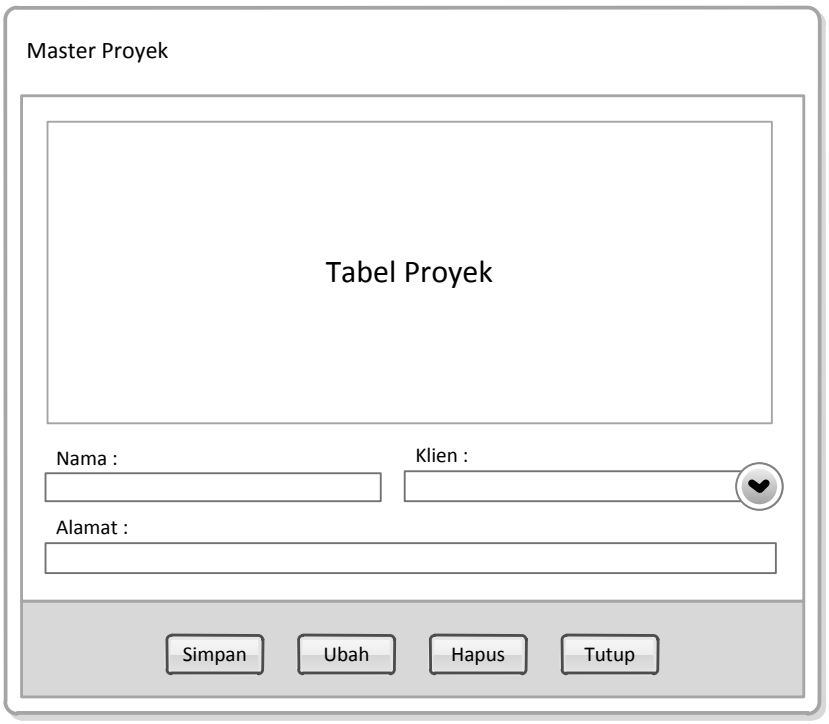
3.3.4.1.7.2 Algoritma

Input : nama, kontak dan alamat.

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data klien yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data klien yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data klien yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data klien.

3.3.4.1.8 Olah Data Proyek



Gambar 3.83 Deskripsi Layar Olah Data Proyek

3.3.4.1.8.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Klien	Table	Tabel yang berisikan data proyek
Nama	Textbox	Masukkan data nama proyek
Klien	ComboBox	Lookup data klien
Alamat	Textbox	Masukkan data alamat proyek
Simpan	Button	Tombol aksi simpan data proyek
Ubah	Button	Tombol aksi ubah data proyek
Hapus	Button	Tombol aksi hapus data proyek
Tutup	Button	Tombol aksi keluar

Tabel 3.35 Deskripsi Objek Olah Data Proyek

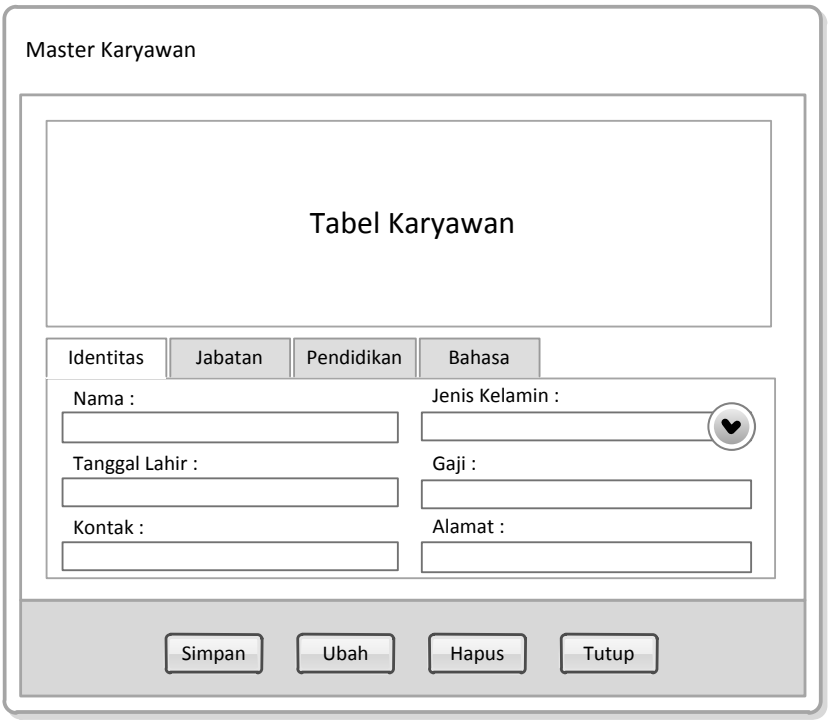
3.3.4.1.8.2 Algoritma

Input : nama, klien dan alamat.

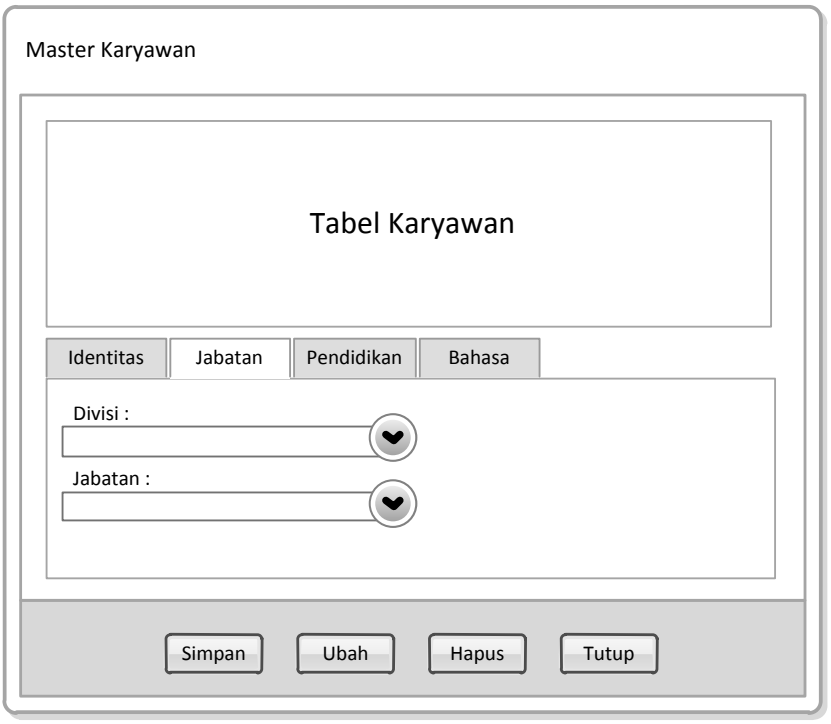
Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data proyek yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data proyek yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data proyek yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data proyek.

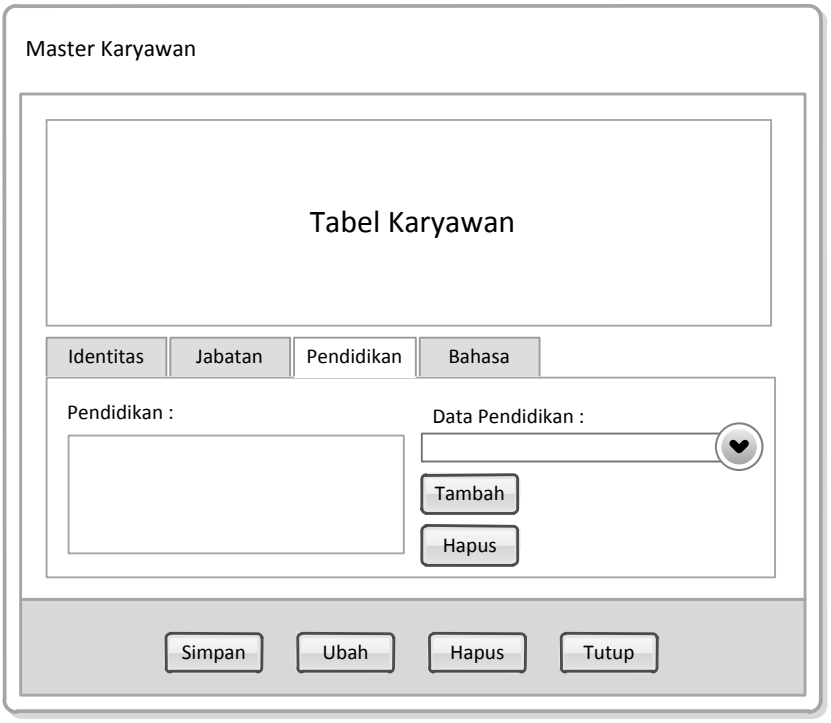
3.3.4.1.9 Olah Data Karyawan



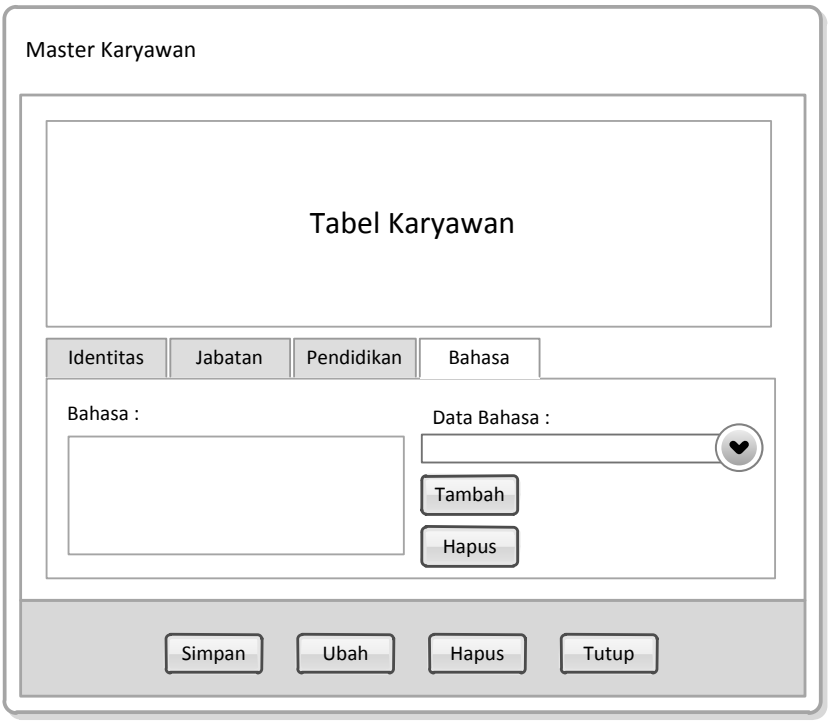
Gambar 3.84 Deskripsi Layar Olah Data Karyawan 1



Gambar 3.85 Deskripsi Layar Olah Data Karyawan 2



Gambar 3.86 Deskripsi Layar Olah Data Karyawan 3



Gambar 3.87 Deskripsi Layar Olah Data Karyawan 4

3.3.4.1.9.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Karyawan	Table	Tabel yang berisikan data karywan
Nama	Textbox	Masukkan data nama karyawan
Jenis Kelamin	Combobox	Masukkan data jenis kelamin karyawan
Tanggal Lahir	Textbox	Masukkan data tangal lahir karyawan
Gaji	Textbox	Masukkan data gaji karyawan
Kontak	Textbox	Masukkan data kontak karyawan
Alamat	Textbox	Masukkan data alamat karyawan
Divisi	Combobox	Lookup data divisi dari database
Jabatan	Combobox	Lookup data jabatan dari database berdasarkan divisi
Pendidikan	Listbox	Tambah data pendidikan dari combobox data pendidikan
Bahasa	Listbox	Tambah data bahasa dari combobox data bahasa
Simpan	Button	Tombol aksi simpan data karyawan
Ubah	Button	Tombol aksi ubah data karyawan
Hapus	Button	Tombol aksi hapus data karyawan
Tutup	Button	Tombol aksi keluar

Tabel 3.36 Deskripsi Objek Olah data Karyawan

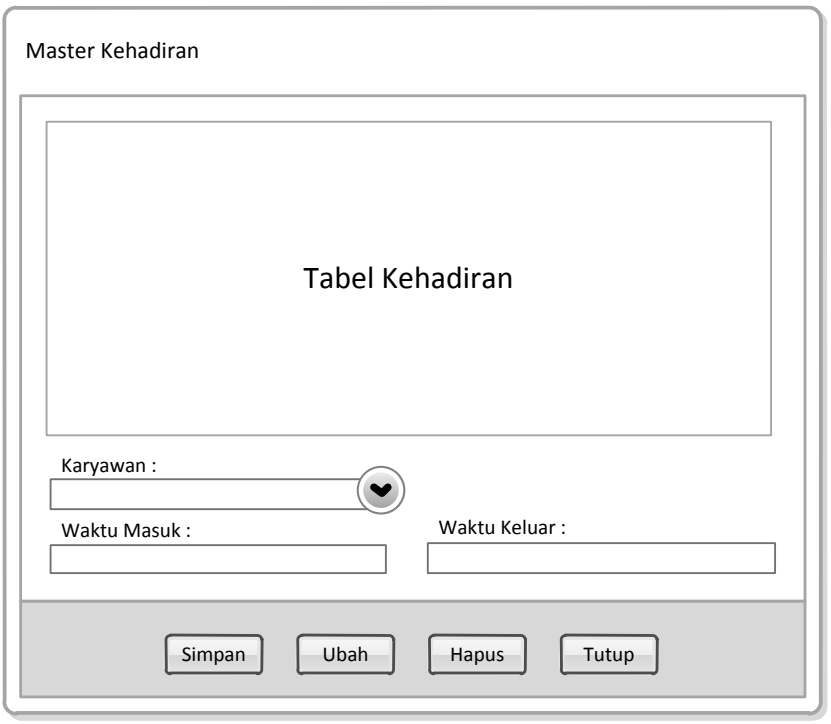
3.3.4.1.9.2 Algoritma

Input : nama, jenis kelamin, tanggal lahir, gaji, alamat, kontak, jabatan, bahasa dan pendidikan.

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data karyawan yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data karyawan yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data karyawan yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data karyawan.

3.3.4.1.10 Olah Data Kehadiran



Gambar 3.88 Deskripsi Layar Olah Data Kehadiran

3.3.4.1.10.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Kehadiran	Table	Tabel yang berisikan data kehadiran
Karyawan	Combobox	Lookup data karyawan
Waktu Masuk	Textbox	Masukkan data waktu masuk
Waktu Keluar	Textbox	Masukkan data waktu keluar
Simpan	Button	Tombol aksi simpan data kehadiran
Ubah	Button	Tombol aksi ubah data kehadiran
Hapus	Button	Tombol aksi hapus data kehadiran
Tutup	Button	Tombol aksi keluar

Tabel 3.37 Deskripsi Objek Olah Data Kehadiran

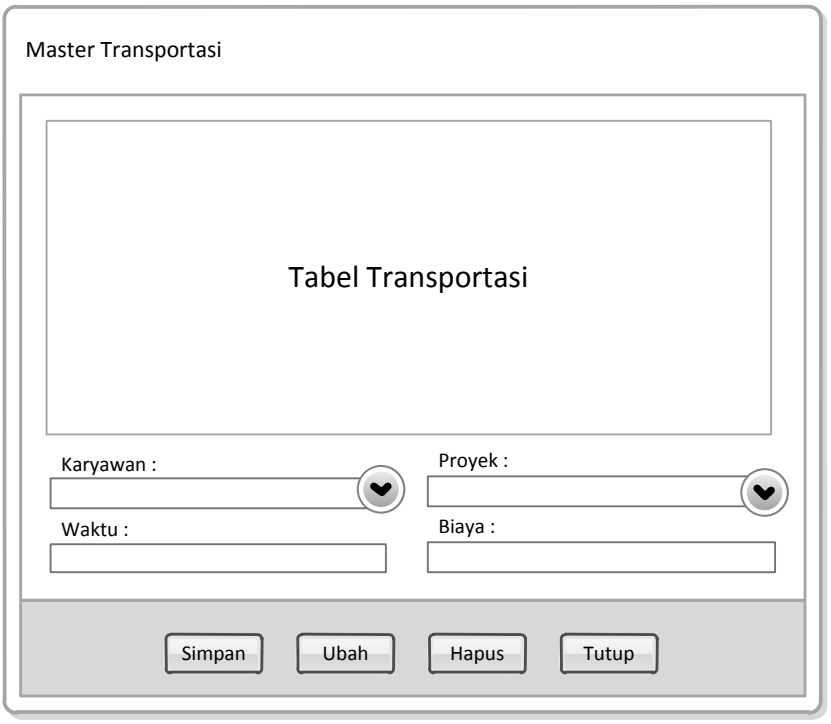
3.3.4.1.10.2 Algoritma

Input : karyawan, waktu masuk dan waktu keluar.

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data kehadiran yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data kehadiran yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data kehadiran yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data kehadiran.

3.3.4.1.11 Olah Data Transportasi



Gambar 3.89 Deskripsi Layar Olah Data Transportasi

3.3.4.1.11.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Kehadiran	Table	Tabel yang berisikan data kehadiran
Karyawan	Combobox	Lookup data karyawan
Proyek	Combobox	Lookup data proyek
Waktu	Textbox	Masukkan data waktu karyawan melakukan transportasi
Biaya	Textbox	Masukkan biaya transportasi
Simpan	Button	Tombol aksi simpan data kehadiran
Ubah	Button	Tombol aksi ubah data kehadiran
Hapus	Button	Tombol aksi hapus data kehadiran
Tutup	Button	Tombol aksi keluar

Tabel 3.38 Deskripsi Objek Olah Data Kehadiran

3.3.4.1.11.2 Algoritma

Input : karyawan, proyek, waktu, biaya

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data transportasi yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data transportasi yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data transportasi yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data transportasi.

3.3.4.1.12 Olah Data Pengguna



Gambar 3.90 Deskripsi Layar Olah Data Pengguna

3.3.4.1.12.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Kehadiran	Table	Tabel yang berisikan data kehadiran
Nama Pengguna	Textbox	Masukkan data nama pengguna
Kata Sandi	Passwordbox	Masukkan data kata sandi
Hak Akses	Combobox	Lookup data dari Hak Akses
Simpan	Button	Tombol aksi simpan data kehadiran
Ubah	Button	Tombol aksi ubah data kehadiran
Hapus	Button	Tombol aksi hapus data kehadiran
Tutup	Button	Tombol aksi keluar

Tabel 3.39 Deskripsi Objek Olah Data Pengguna

3.3.4.1.12.2 Algoritma

Input : nama pengguna, kata sandi dan hak akses

Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data pengguna yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data pengguna yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data pengguna yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data pengguna.

3.3.4.1.13 Olah Data Hak Akses



Gambar 3.91 Deskripsi Layar Olah Data Hak Akses

3.3.4.1.13.1 Deskripsi Objek

Objek	Jenis	Keterangan
Tabel Kehadiran	Table	Tabel yang berisikan data kehadiran
Nama	Textbox	Masukkan nama hak akses
Tambah	Checkbox	Cek jika hak akses boleh menambah data
Ubah	Checkbox	Cek jika hak akses boleh mengubah data
Hapus	Checkbox	Cek jika hak akses boleh menghapus data
Tampil	Checkbox	Cek jika hak akses bole menampilkan data
Simpan	Button	Tombol aksi simpan data kehadiran
Ubah	Button	Tombol aksi ubah data kehadiran
Hapus	Button	Tombol aksi hapus data kehadiran
Tutup	Button	Tombol aksi keluar

Tabel 3.40 Deskripsi Objek Olah Data Hak Akses

3.3.4.1.13.2 Algoritma

Input : nama dan detail hak akses

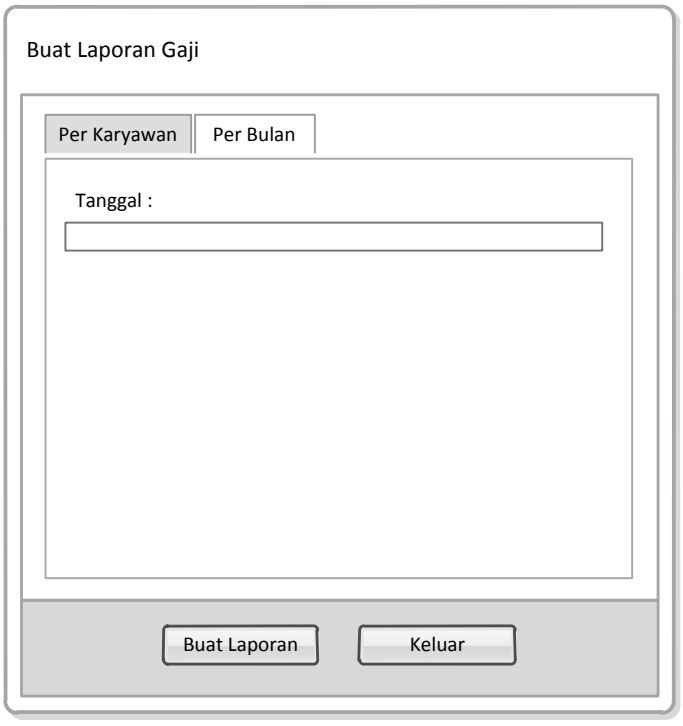
Proses :

1. Jika ditekan tombol simpan maka sistem akan menyimpan data hak akses yang ke *database*.
2. Jika ditekan tombol ubah maka sistem akan mengubah data hak akses yang terseleksi di tabel dengan data yang dimasukkan dan menyimpan perubahannya ke *database*.
3. Jika ditekan tombol hapus maka sistem akan menghapus data hak akses yang terseleksi di tabel.
4. Jika tombol keluar ditekan maka sistem akan menutup *form* olah data hak akses.

3.3.4.1.14 Buat Laporan



Gambar 3.92 Deskripsi Layar Buat Laporan 1



Gambar 3.93 Deskripsi Layar Buat Laporan 2

3.3.4.1.14.1 Deskripsi Objek

Objek	Jenis	Keterangan
Karyawan	Combobox	Lookup dari data karyawan
Tanggal	Textbox	Masukkan tahun dan bulan untuk laporan
Buat Laporan	Button	Tombol aksi membuat laporan
Keluar	Button	Tombol untuk keluar

Tabel 3.41 Deskripsi Objek Buat Laporan

3.3.4.1.14.2 Algoritma

Input : karyawan dan tanggal atau tanggal.

Proses :

1. Jika tombol buat laporn di tekan, maka sistem akan membuat laporan berdasarkan data yang diberikan oleh pengguna.
2. Jika tombol keluar ditekan maka sistem akan menutup *form* buat laporan.

3.3.4.1.15 Kesalahan



Gambar 3.94 Deskripsi Layar Kesalahan

3.3.4.1.15.1 Deskripsi Objek

Objek	Jenis	Keterangan
Sebab	Textarea	Isi pesan kesalahan
OK	Button	Tombol untuk menurup form kesalahan

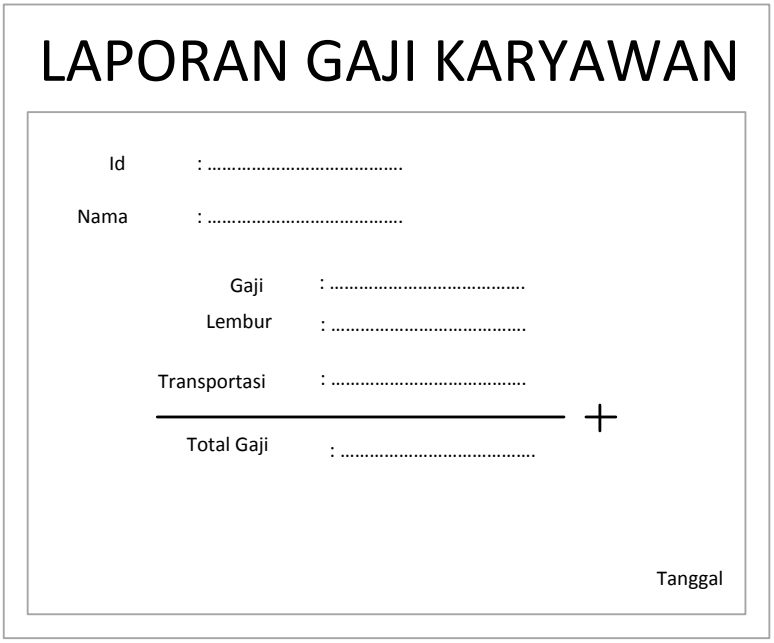
Tabel 3.42 Deskripsi Objek Kesalahan

3.3.4.1.15.2 Algoritma

Proses :

1. Jika ditekan tombol OK, maka sistem akan menutup *form* kesalahan.

3.3.4.1.16 Laporan Gaji Karyawan



Gambar 3.95 Deskripsi Layar Laporan Gaji Karyawan

3.3.4.1.16.1 Deskripsi Objek

Objek	Jenis	Keterangan
Id	Label	Id karyawan
Nama	Label	Nama karyawan
Gaji	Label	Gaji karyawan
Lembur	Label	Lembur karyawan
Transportasi	Label	Transportasi karyawan

Total Gaji	Label	Total gaji karyawan
Tanggal	Label	Bulan gaji

Tabel 3.43 Deskripsi Objek Laporan Gaji Karyawan

3.3.4.1.17 Laporan Per Bulan



Gambar 3.96 Deskripsi Layar Laporan Gaji Bulanan

3.3.4.1.17.1 Deskripsi Objek

Objek	Jenis	Keterangan
Id	Label	Id karyawan
Nama	Label	Nama karyawan
Gaji	Label	Gaji karyawan
Lembur	Label	Lembur karyawan
Transportasi	Label	Transportasi karyawan
Total Gaji	Label	Total gaji karyawan
Tanggal	Label	Bulan gaji

Tabel 3.44 Deskripsi Objek Laporan Per Bulan

3.3.4.2 Jaringan Semantik

Jaringan semantik untuk sistem penggajian yang baru adalah seperti pada gambar dibawah ini :



3.97 Jaringan Semantik

3.3.5 Implementasi

3.3.5.1 Perangkat Lunak Pendukung

Sistem penggajian ini dapat berjalan pada seluruh sistem operasi, namun agar sistem penggajian ini dapat berjalan, diperlukan perangkat lunak :

1. Java Runtime Edition versi 1.6
2. Database Management System

3.3.5.2 Kebutuhan Perangkat Keras

Kebutuhan minimal perangkat keras yang digunakan untuk menjalankan sistem penggajian ini adalah seperti yang tertera dibawah ini. Semakin tinggi spesifikasi perangkat keras, maka performa sistem penggajian akan lebih baik pula :

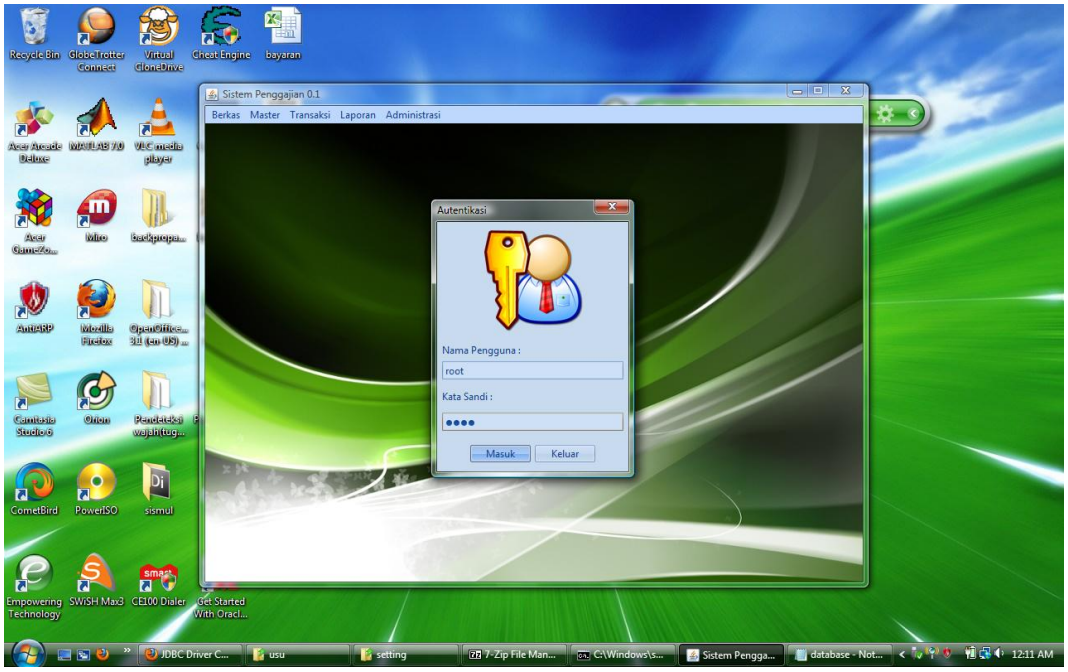
1. Processor 2 GHz
2. RAM 512 MB
3. Ruang HartDisk minimum 100MB

3.3.5.3 Implementasi Procedur dan Data

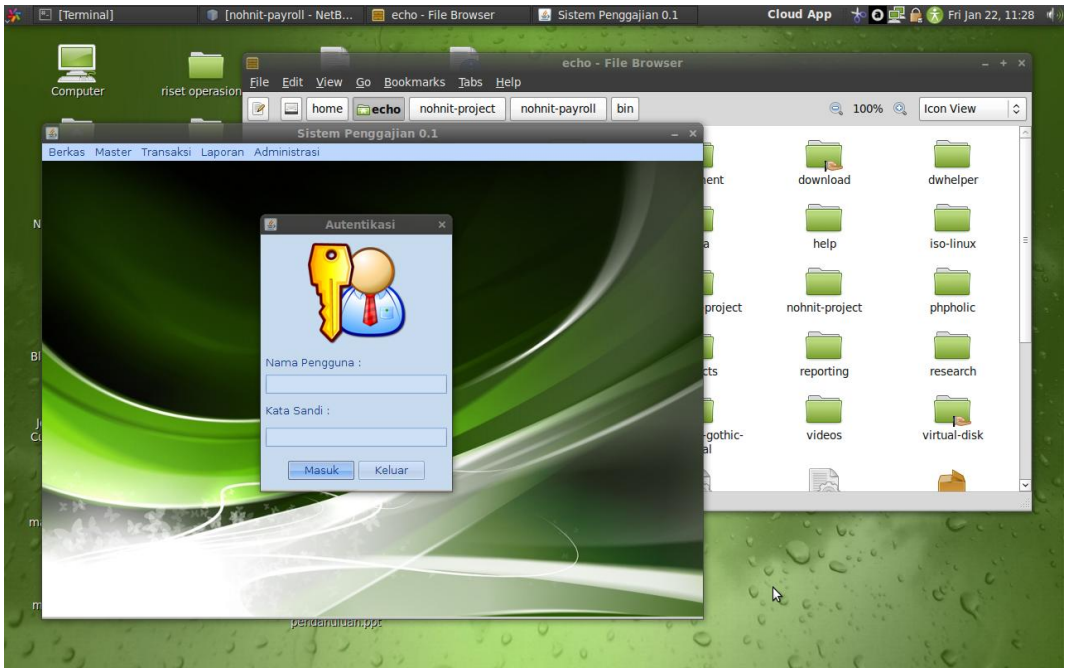
Untuk membuat sistem penggajian ini penulis menggunakan *Java Development Kit, NetBeans IDE, MySQL Server, Oracle Database, Hibernate Framework,* dan *Spring Framework.*

3.3.5.4 Implementasi Antar Muka

Berikut ini contoh beberapa tampilan antarmuka sistem penggajian yang dijalankan di sistem operasi Windows Vista dan LinuxMint.



Gambar 3.98 Sistem Penggajian di Windows



Gambar 3.99 Sistem Penggajian di LinuxMint

3.3.6 Pengujian

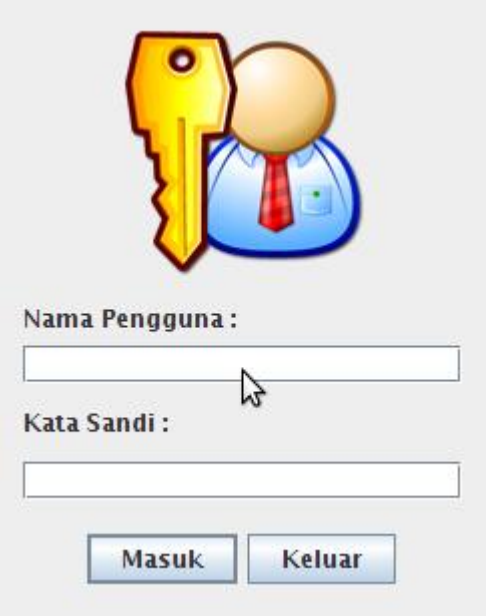
Proses pengujian adalah mencoba program dengan memasukan data kedalam *form* - *form* masukan yang telah disediakan. Pada tahap ini merupakan

kelanjutan dari tahap implementasi yaitu melakukan pengujian - pengujian terhadap aplikasi yang dibangun.

3.3.6.1 Kasus dan Pengujian Alpha

3.3.6.1.1 Pengujian Autentikasi

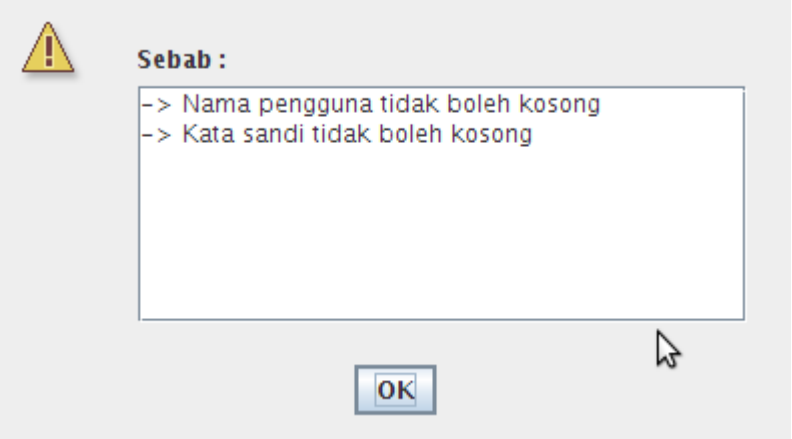
Pengujian autentikasi dilakukan dengan memasukan data yang salah dan juga masukan data yang benar.



Gambar 3.100 Tampilan Antar Muka Autentikasi

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pengguna = root Kata Sandi = root	Pengguna dapat masuk ke sistem penggajian	Sistem mengijinkan pengguna masuk ke sistem penggajian	Diterima

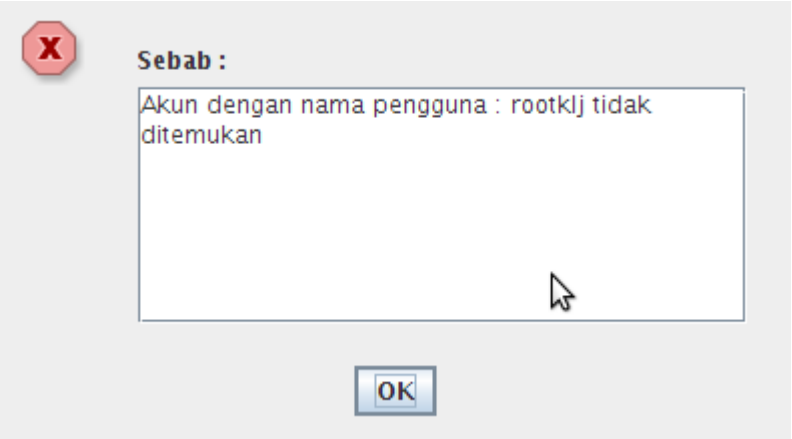
Tabel 3.45 Data Normal Autentikasi



Gambar 3.101 Tampilan Validasi Gagal Autentikasi

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pengguna = Kata Sandi =	Pengguna tidak dapat masuk dikarenakan nama pengguna dan kata sandi masih kosong	Sistem memberi peringatan jika nama pengguna dan kata sandi harus diisi	Diterima

Tabel 3.46 Data Salah Autentikasi



Gambar 3.102 Pengguna Tidak Terdaftar

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pengguna = rootklj Kata Sandi = asdasdasd	Pengguna tidak dapat masuk dikarenakan nama pengguna tidak terdaftar	Sistem memberi peringatan jika nama pengguna tidak terdaftar	Diterima

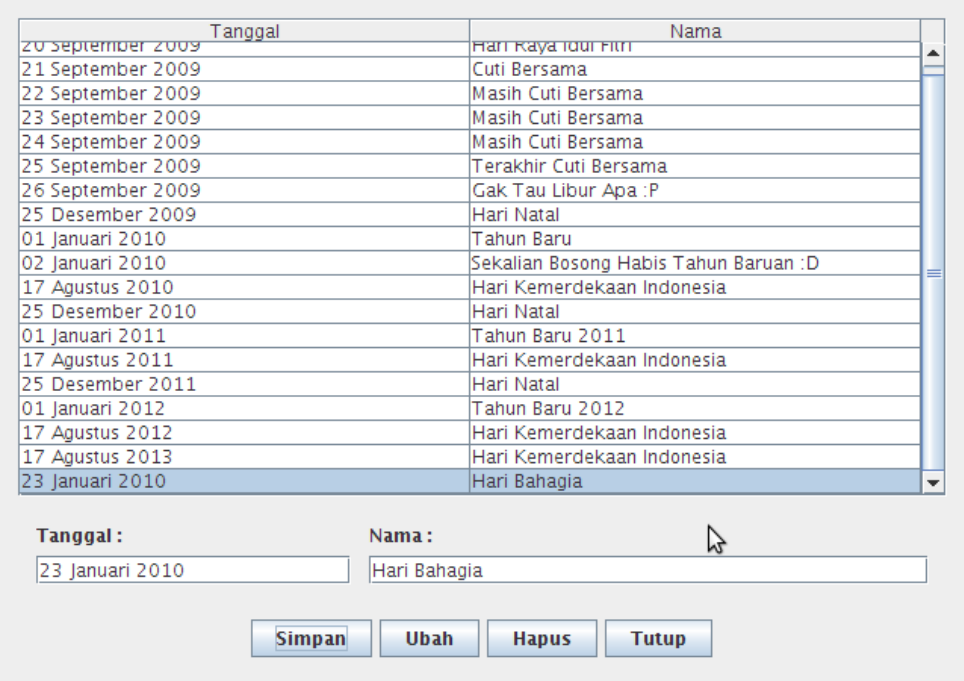
Tabel 3.47 Data Salah Autentikasi

3.3.6.1.2 Pengujian Oleh Data Hari Libur

Pengujian hari libur dilakukan dengan mamasukkan data-data yang salah dan data-data yang benar.



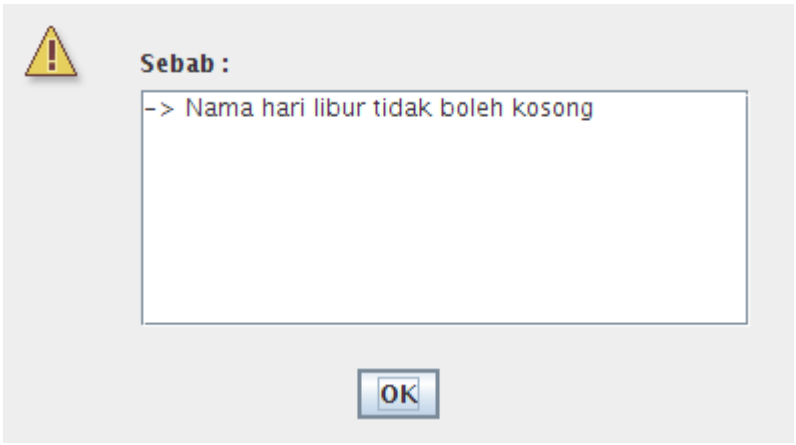
Gambar 3.103 Tampilan Antarmuka Olah Hari Libur



Gambar 3.104 Menambah Data Hari Libur

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tangal : 23 Januari 2010 Nama : Hari Bahagia	Data yang dimasukkan akan disimpan di *database*	Sistem menyimpan data di *database*	Diterima

Tabel 3.48 Data Normal Hari Libur



Gambar 3.105 Validasi Gagal Hari Libur

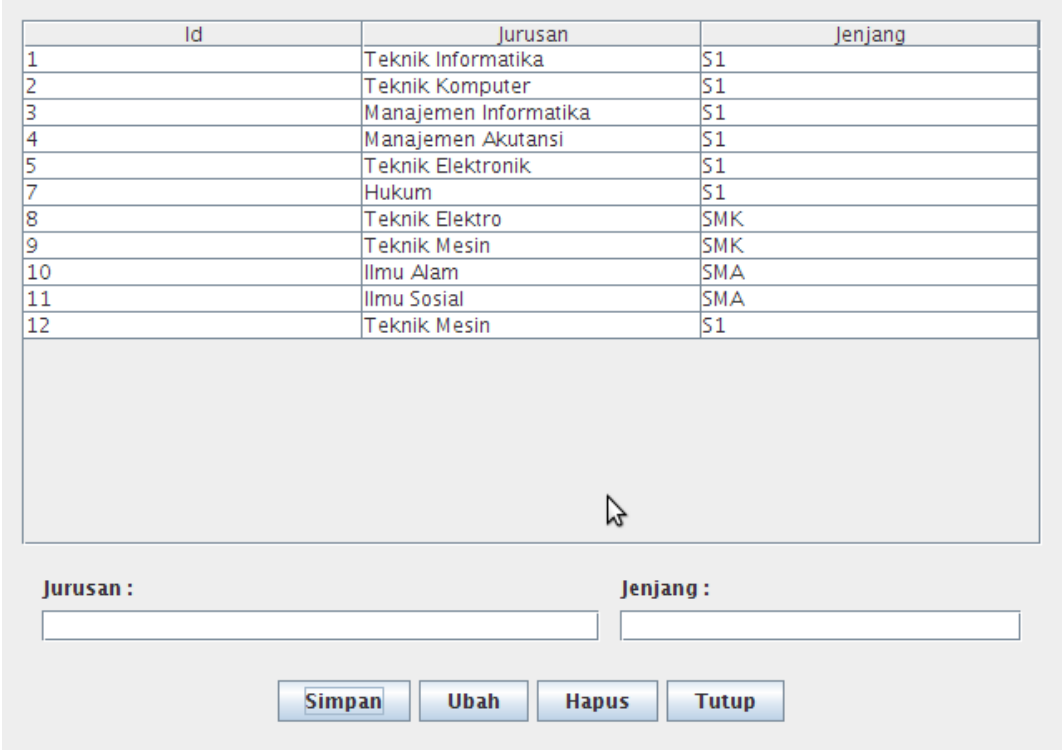
Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Tangal : 23 Januari 2010	Sistem harus menolak data masukan karena	Sistem menolak data masukan	Diterima

Nama :	nama hari libur masih kosong	dan menampilkan peringatan	

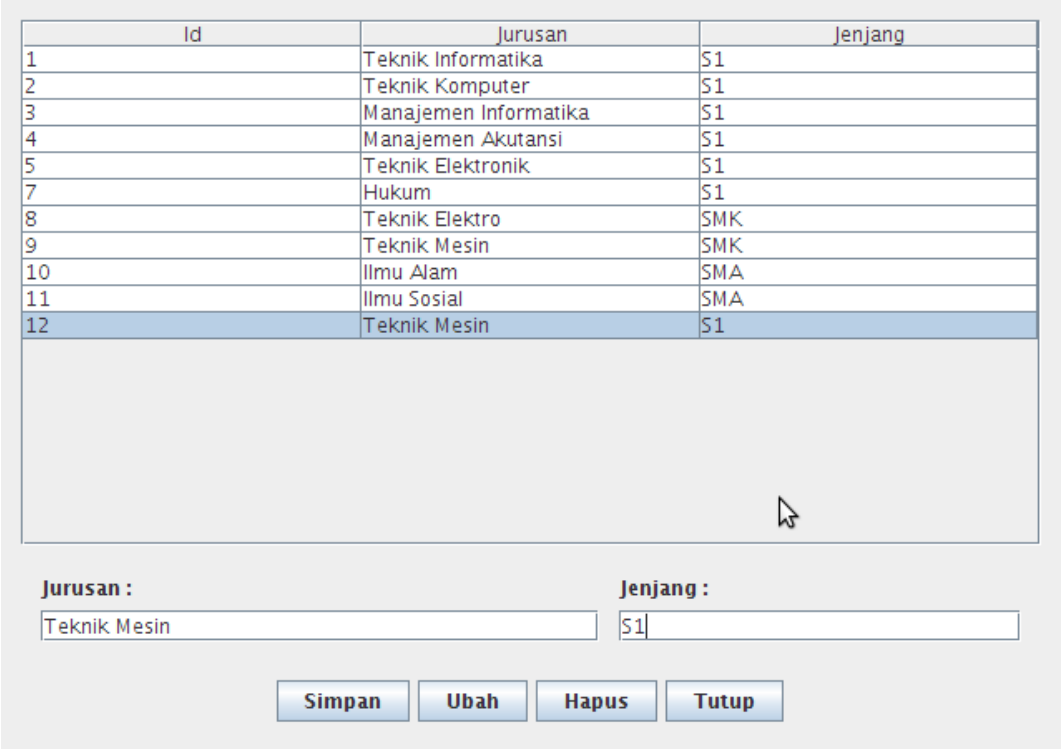
Tabel 3.49 Data Salah Hari Libur

3.3.6.1.3 Pengujian Olah Data Pendidikan

Pengujian pendidikan dilakukan dengan memasukkan data-data yang salah dan memasukkan data-data benar.



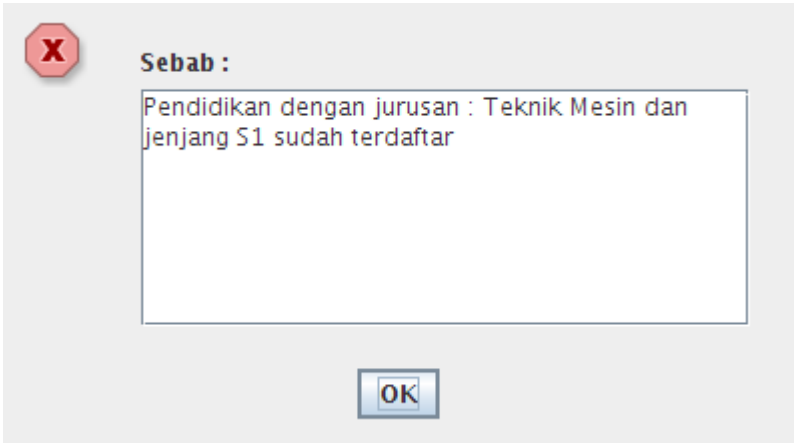
Gambar 3.106 Tampilan Antar Muka Olah Data Pendidikan



Gambar 3.107 Menambah Data Pendidikan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jurusan : Teknik Mesin Nama : S1	Sistem harus menyimpan data ke dalam *database* dan menampilkannya ke tabel	Sistem menyimpan data ke *database* dan menampilkannya ke tabel	Diterima

Tabel 3.50 Data Normal Olah Data Pendidikan



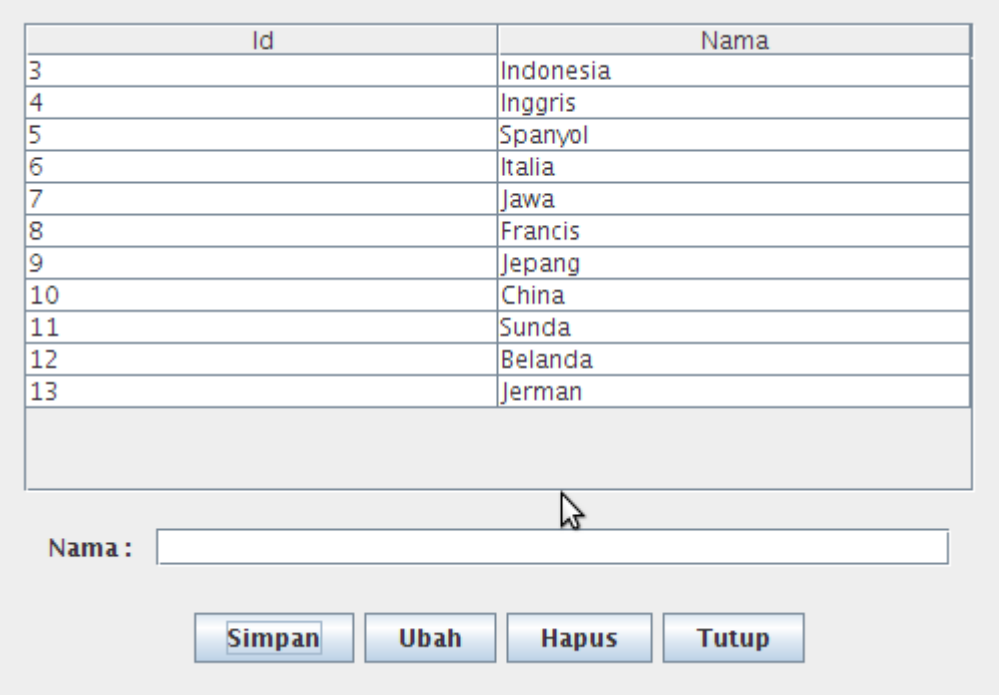
Gambar 3.108 Gagal Menambah Data Pendidikan

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Jurusan : Teknik Mesin Nama : S1	Sistem harus menolak data masukkan karena data masukkan tersebut sebelumnya telah dimasukkan sehingga ada di *database*	Sistem menolak inputan data dan memberi peringatan jika data sudah ada di *database*.	Diterima

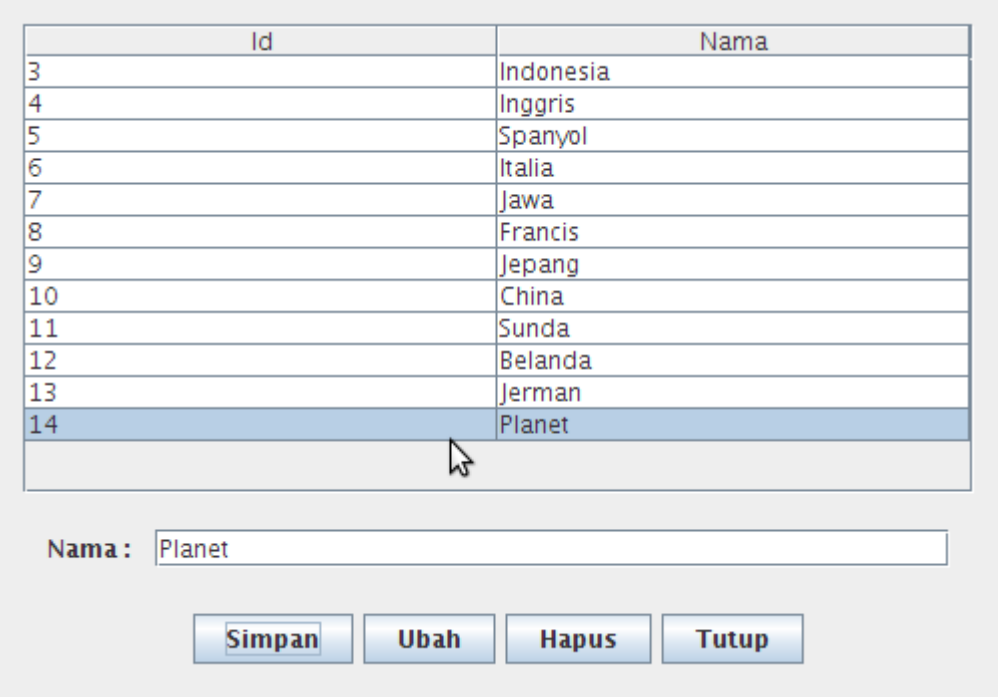
Tabel 3.51 Data Salah Uji Coba Data Pendidikan

3.3.6.1.4 Pengujian Olah Data Bahasa

Pengujian olah data bahasa dilakukan dengan memasukkan data-data yang salah dan data-data yang benar.



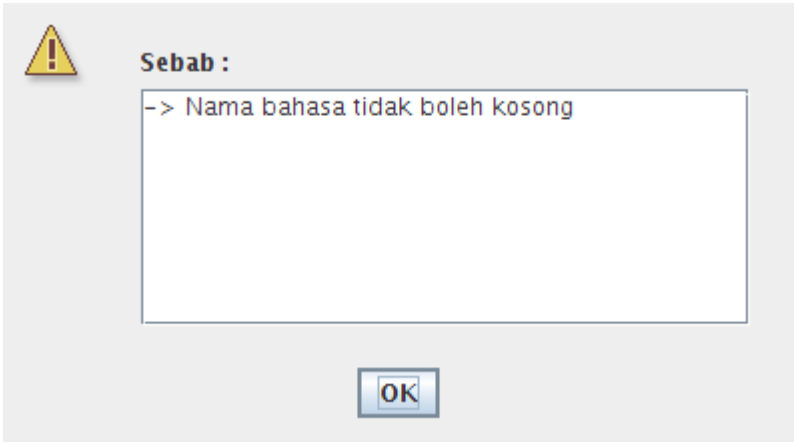
Gambar 3.109 Tampilan Olah Data Bahasa



Gambar 3.110 Menambah Data Bahasa

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Planet	Sistem harus menambahkan data ke *database* dan menampilkannya di tabel	Sistem menyimpan data ke *database* dan menampilkannya ke tabel	Diterima

Tabel 3.52 Data Normal Olah Data Bahasa



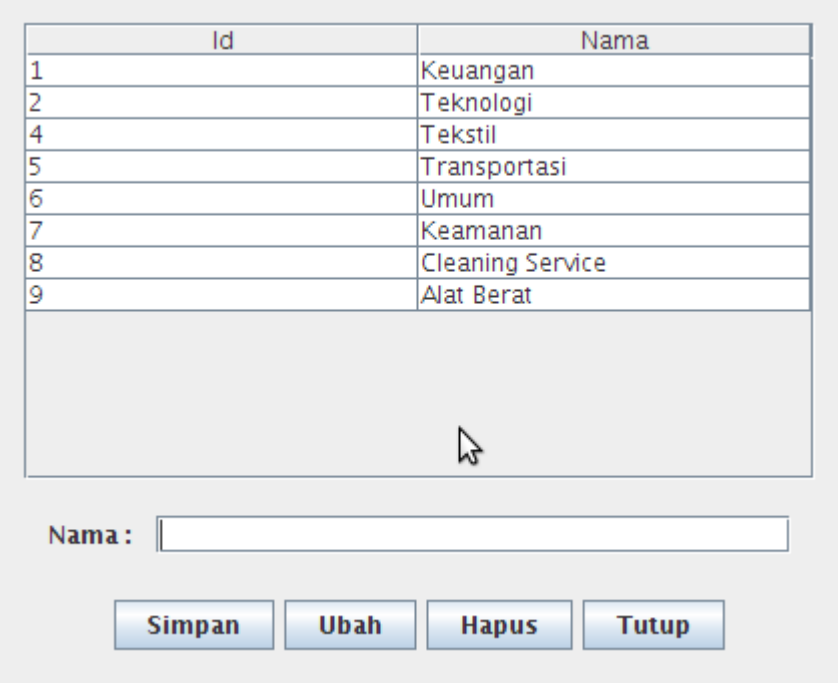
Gambar 3.111 Validasi Gagal Olah Data Bahasa

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama :	Sistem harus menolak masukan pengguna karena data nama masih kosong	Sistem menolak masukan pengguna dan menampilkan konfirmasi validasi gagal	Diterima

Gambar 3.112 Data Salah Olah Data Bahasa

3.3.6.1.5 Pengujian Olah Data Divisi

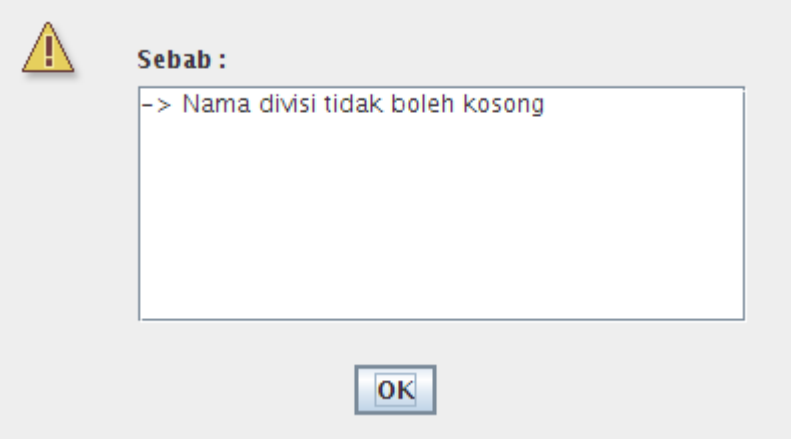
Pengujian olah data divisi dilakukan dengan cara memasukkan data-data salah dan data-data benar.



Gambar 3.113 Antar Muka Olah Data Divisi

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Alat Berat	Sistem menyimpan data yang dimasukkan ke *database* dan menampilkannya di tabel	Sistem menyimpan data ke *database* dan menampilkannya ke tabel	Diterima

Tabel 3.53 Data Normal Olah data Divisi



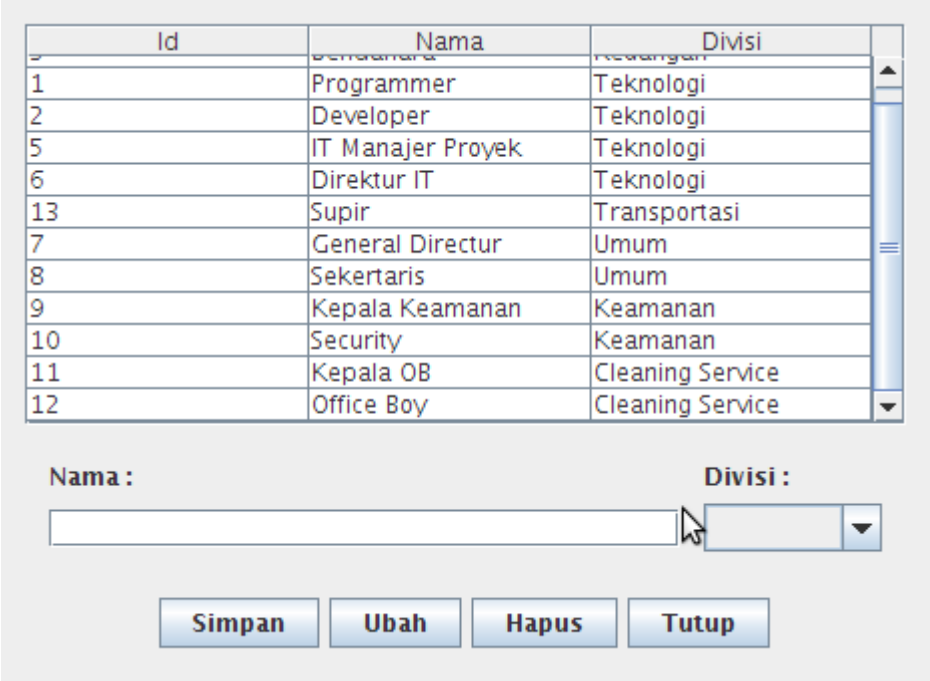
Gambar 3.114 Validasi Gagal Olah Data Divisi

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama :	Sistem menolak masukan pengguna karena nama divisi masih kosong	Sistem menolak masukkan pengguna dan menampilkan peringatan kesalahan	Diterima

Tabel 3.54 Data Salah Olah Data Divisi

3.3.6.1.6 Pengujian Olah Data Jabatan

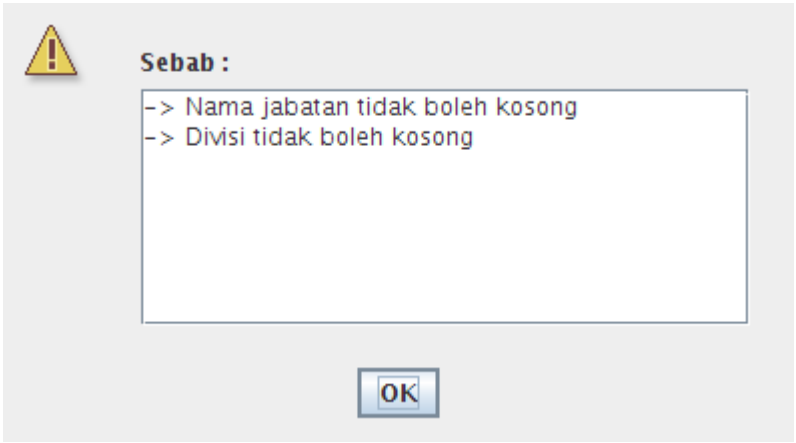
Pengujian olah data jabatan dilakukan dengan cara memasukkan data-data yang salah dan data-data yang benar.



Gambar 3.115 Antarmuka Olah Data Jabatan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Office Box Divisi : Cleaning Service	Sistem menyimpan data yang dimasukan oleh pengguna ke *database* dan menampilkannya ke tabel	Sistem menyimpan data yang dimasukkan ke *database* dan menampilkannya ke tabel	Diterima

Tabel 3.55 Data Normal Olah Data Jabatan



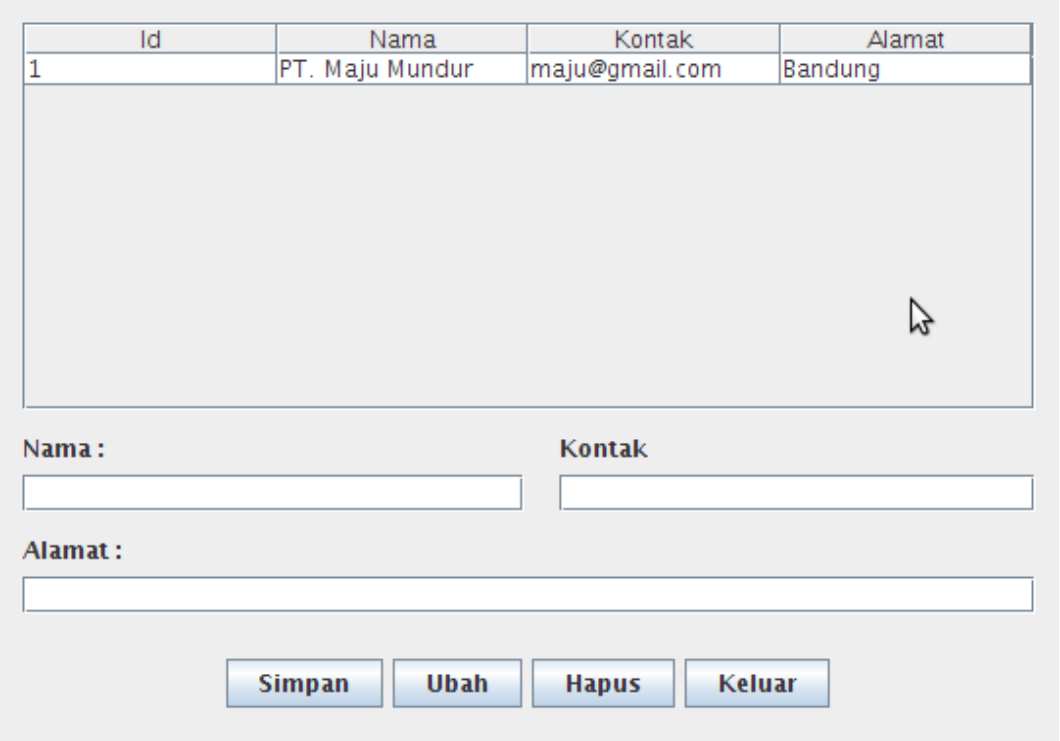
Gambar 3.116 Validasi Gagal Olah Data Jabatan

Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Divisi :	Sistem menolah masukan dari pengguna karena data nama dan divisi masih kosong	Sistem nenolak data masukan pengguna dan menampilkan pesan validasi gagal	Diterima

Tabel 3.56 Data Salah Olah Data Jabatan

3.3.6.1.7 Pengujian Olah Data Klien

Pengujian olah data klien dilakukan dengan cara memasukkan data-data salah dan data-data benar.

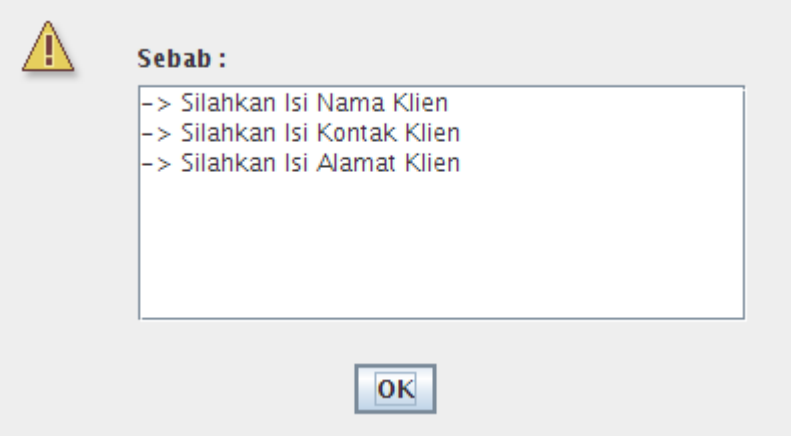


Gambar 3.117 Antarmuka Olah Data Klien

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : PT. Maju Mundur Kontak : maju@gmail.com	Sistem menyimpan data yang dimasukan oleh pengguna ke *database* dan	Sistem menyimpan data yang dimasukkan ke	Diterima

Alamat : Bandung	menampilkannya ke tabel	*database* dan menampilkannya ke tabel	

Tabel 3.57 Data Normal Olah Data Klien

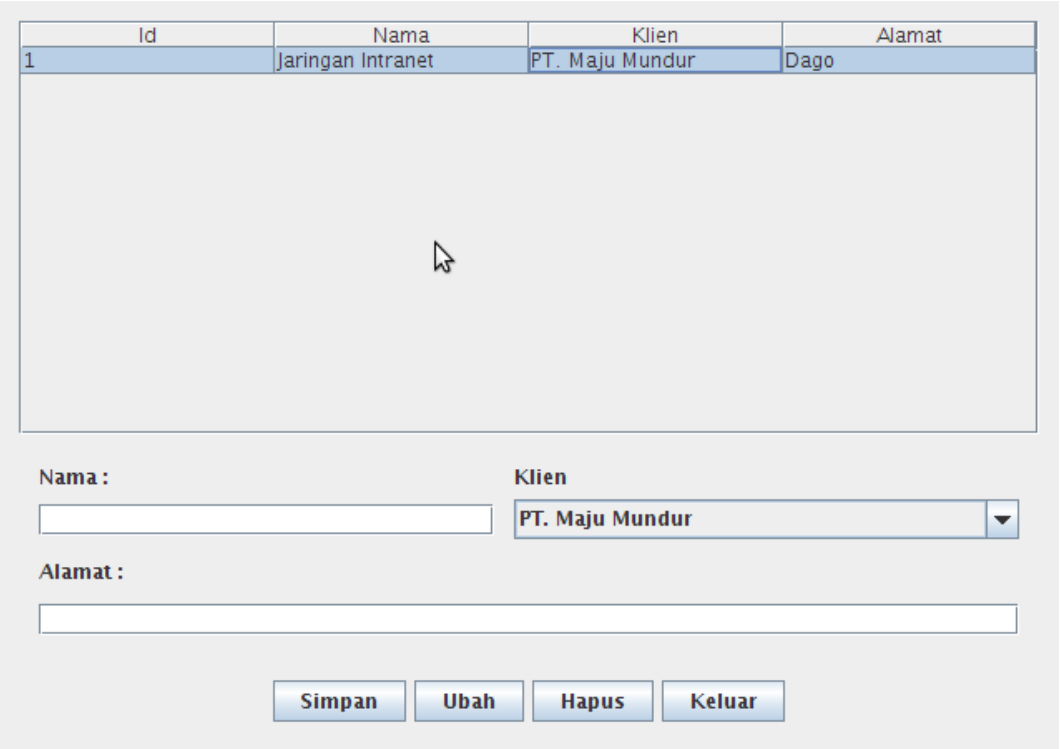


Gambar 3.118 Validasi Gagal Olah Data Klien

Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Kontak : Alamat :	Sistem menolak masukan dari pengguna karena data nama, alamat dan kontak masih kosong	Sistem nenolak data masukan pengguna dan menampilkan pesan validasi gagal	Diterima

Tabel 3.58 Data Salah Olah Data Klien

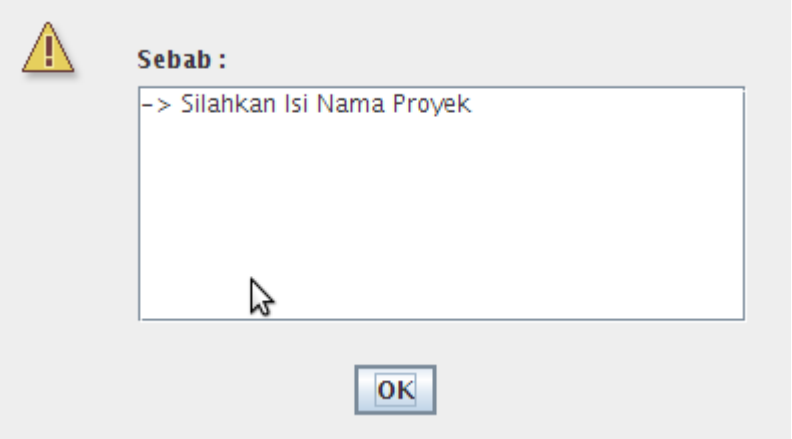
3.3.6.1.8 Pengujian Olah Data Proyek



Gambar 3.119 Antarmuka Olah Data Proyek

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Jaringan Intranet Klien : PT. Maju Mundur Alamat : Dago	Sistem menyimpan data yang dimasukan oleh pengguna ke *database* dan menampilkannya ke tabel	Sistem menyimpan data yang dimasukkan ke *database* dan menampilkann ya ke tabel	Diterima

Tabel 3.59 Data Normal Olah Data Proyek



Gambar 3.120 Validasi Gagal Olah Data Proyek

Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Klien : PT. Maju Mundur Alamat : Dago	Sistem menolak masukan dari pengguna karena data nama proyek masih kosong	Sistem nenolak data masukan pengguna dan menampilkan pesan validasi gagal	Diterima

Tabel 3.60 Data Salah Olah Data Proyek

3.3.6.1.9 Pengujian Olah Data Pengguna

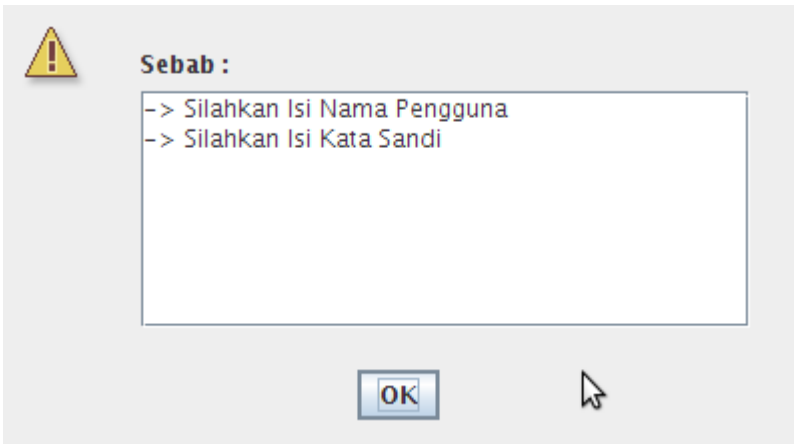
Pengujian olah data pengguna dilakukan dengan cara memasukkan data-data yang salah dan data-data yang benar.



Gambar 3.121 Antarmuka Olah Data Pengguna

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pengguna : Eko Kata Sandi : xxxxxxx Hak Akses : Admin	Sistem menyimpan data yang dimasukan oleh pengguna ke *database* dan menampilkannya ke tabel	Sistem menyimpan data yang dimasukkan ke *database* dan menampilkann ya ke tabel	Diterima

Tabel 3.61 Data Normal Olah Data Pengguna



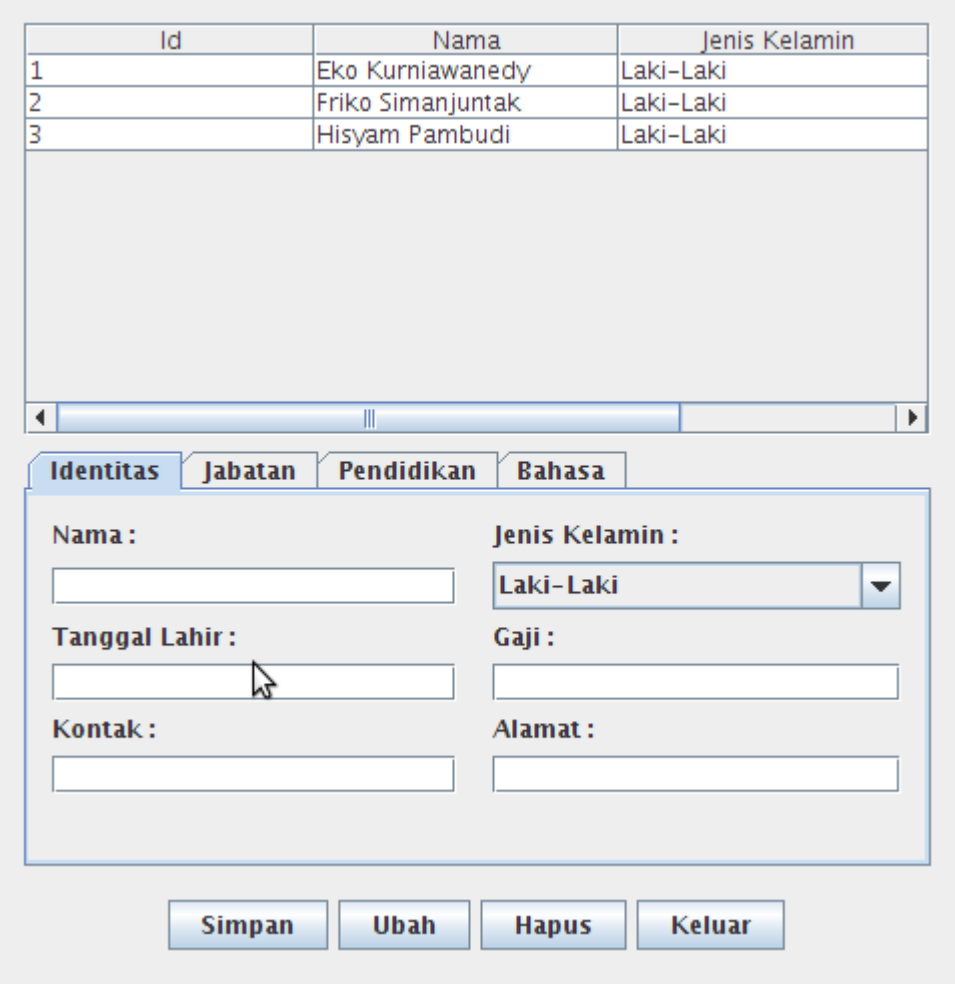
Gambar 3.122 Validasi Gagak Olah Data Pengguna

Kasus dan Hasil Uji (Data salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama Pengguna : Kata Sandi : Hak Akses : Admin	Sistem menolak masukan dari pengguna karena data nama pengguna dan kata sandi masih kosong	Sistem nenolak data masukan pengguna dan menampilkan pesan validasi gagal	Diterima

Tabel 3.62 Data Salah Olah Data Pengguna

3.3.6.1.10 Pengujian Olah Data Karyawan

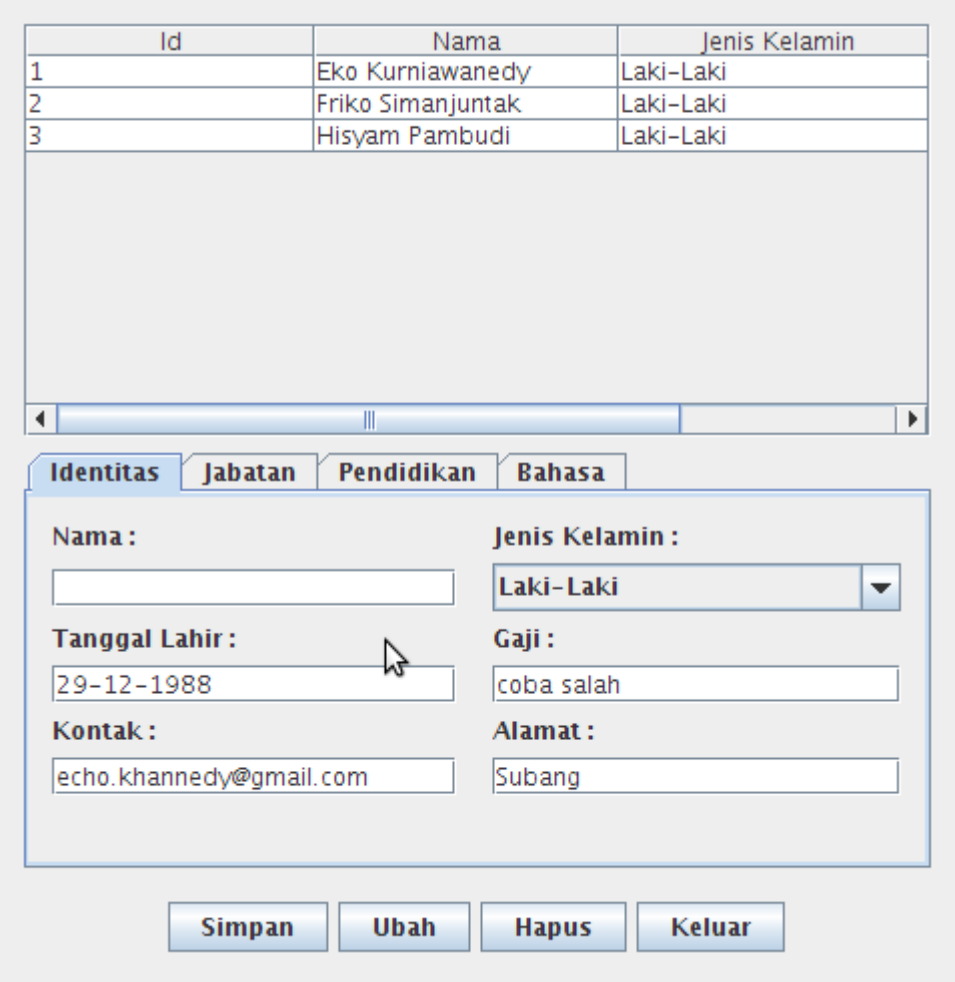
Pengujian olah data karyawan dilakukan dengan memasukkan data-data yang salah dan data-data yang benar.



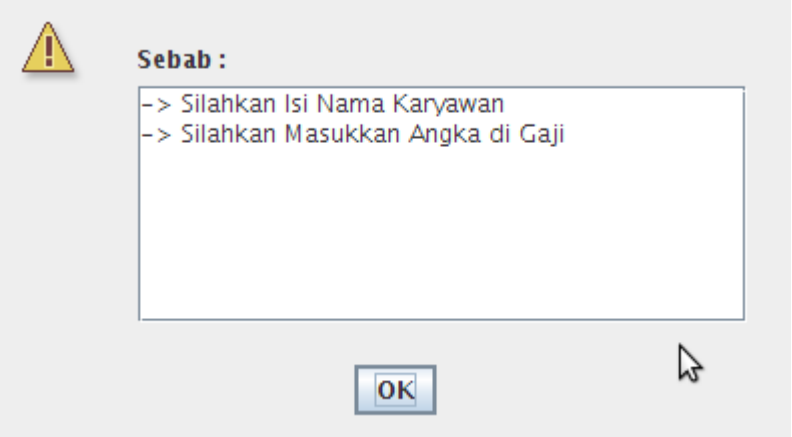
Tabel 3.63 Antarmuka Olah Data Karyawan

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Hisyam Pambudi Jenis Kelamin : Laki-laki Tanggal Lahir : 1-8-1988 Gaji : 5000000 Kontak : hisyam@gmail.com Alamat : Depok Jabatan : Programmer Pendidikan : Teknik Informatika S1 Bahasa : Inggris, Indonesia	Sistem menyimpan data yang dimasukan oleh pengguna ke *database* dan menampilkannya ke tabel	Sistem menyimpan data yang dimasukkan ke *database* dan menampilkannya ke tabel	Diterima

Tabel 3.64 Data Normal Olah Data Karyawan



Gambar 3.123 Data Salah di Pengolahan Data Karyawan



Gambar 3.124 Validasi Gagal Olah Data Karyawan

Kasus dan Hasil Uji (Data Salah)			
Data Masukan	**Yang Diharapkan**	**Pengamatan**	**Kesimpulan**
Nama : Jenis Kelamin : Laki-laki Tanggal Lahir : 1-8-1988 Gaji : coba salah Kontak : hisyam@gmail.com Alamat : Depok Jabatan : Programmer Pendidikan : Teknik Informatika S1 Bahasa : Inggris, Indonesia	Sistem menolak data masukkan dari pengguna karena nama karyawan masih kosong dan masukkan data gaji bukan number	Sistem menolak masukkan data pengguna dan menampilkan peringatan validasi gagal	Diterima

Tabel 3.65 Data Salah Olah Data Karyawan

3.3.7 Kesimpulan

Dari hasil pengujian, dapat disimpulkan jika sistem penggajian yang baru sudah mencapai tujuan yang telah ditentukan. Dan sudah siap untuk di gunakan dan di distribusikan.

BAB 4

KESIMPULAN DAN SARAN

4.1 Kesimpulan

Berdasarkan hasil kerja pengembangan sistem informasi penggajian ini, penulis mendapatkan kesimpulan bahwa :

a. Sistem penggajian yang baru dapat dengan mudah berjalan di sistem operasi apapun tanpa harus melakukan konfigurasi ulang ataupun pengaturan ulang, hal ini dikarenakan penulis menggunakan teknologi-teknologi yang netral terhadap sistem operasi manapun.
b. Sistem penggajian yang baru memiliki pengaturan pengguna yang dinamis dan dapat diubah-ubah oleh administrator, hal ini menyebabkan setiap perusahaan yang menggunakan sistem penggajian yang baru tersebut dapat dengan mudah menerapkan ketentuan perusahaan terhadap hak akses pengguna sistem penggajian tersebut.
c. Sistem penggajian yang baru dapat dengan mudah dimigrasikan dari satu database ke database lainnya. Hal ini menyebabkan perusahaan yang menggunakan sistem penggajian baru tersebut tidak perlu khawatir tentang database yang digunakan apakah didukung oleh sistem penggajian ini atau tidak.

4.2 Saran

Dari hasil pengembangan sistem informasi penggajian ini, penulis menyarankan agar pengguna sistem penggajian yang baru tidak menggunakan

komputer yang memiliki spesifikasi dibawah spesifikasi minimum sistem penggajian yang baru, hal ini dikarenakan teknologi yang digunakan untuk membangun sistem informasi yang baru cukup berat.

DAFTAR PUSTAKA

Ford, Neil. (2004), *Art of Java Web Development*, Manning, Greenwich, 45-134.

Schwartz, Baron. (2008), High Performance MySQL, O'Reilly, Sebastopol, 201-400.

Sabariah, Mira Kania. (2009), Diktat Kuliah Rekayasa Perangkat Lunak.