

BAB III

PEMBAHASAN

3.1. Tinjauan Sekolah

Dalam kegiatan rutin yang terjadi di MI Al-Wasthiyah Kotabaru terdapat aturan yang ditentukan dalam pelaksanaannya, dalam hal ini struktur organisasi yang menggambarkan garis perintah serta fungsi-fungsi pelaksanaan. Penulis akan menerangkan profil sekolah, visi misi, serta struktur organisasi ditempat penulis mengadakan riset dalam pembuatan Tugas Akhir ini.

3.1.1. Sejarah Sekolah

Madrasah Ibtidaiyah (MI) Al-Wasthiyah Kotabaru adalah Sekolah Dasar Swasta yang berdiri sejak tahun 1965. Sekolah ini didirikan oleh K.H Zaenal Abidin (Almarhum). Sejak awal berdiri hingga tahun 1980 MI Al-Wasthiyah Kotabaru merupakan lembaga pendidikan terpadu antara pendidikan formal dan pendidikan tidak formal. Pendidikan tidak formal yang dimaksud adalah pendidikan pondok pesantren.

MI Al-Wasthiyah Kotabaru berlokasi di Jalan Raya Pangulah Kampung Gandoang Desa Pangulah Selatan Kecamatan Kotabaru Kabupaten Karawang Provinsi Jawa Barat. MI Al-Wasthiyah Kotabaru memiliki luas tanah 520 m² dengan luas keseluruhan bangunan 265.7 m².

Saat ini status Akreditasi MI Al-Wasthiyah Kotabaru adalah peringkat A dan Nomor Statistik Madrasah 111232150112 dengan 16 tenaga pendidik dan 4 karyawan dengan jumlah siswa 401, 210 siswa perempuan dan 191 siswa laki-laki.

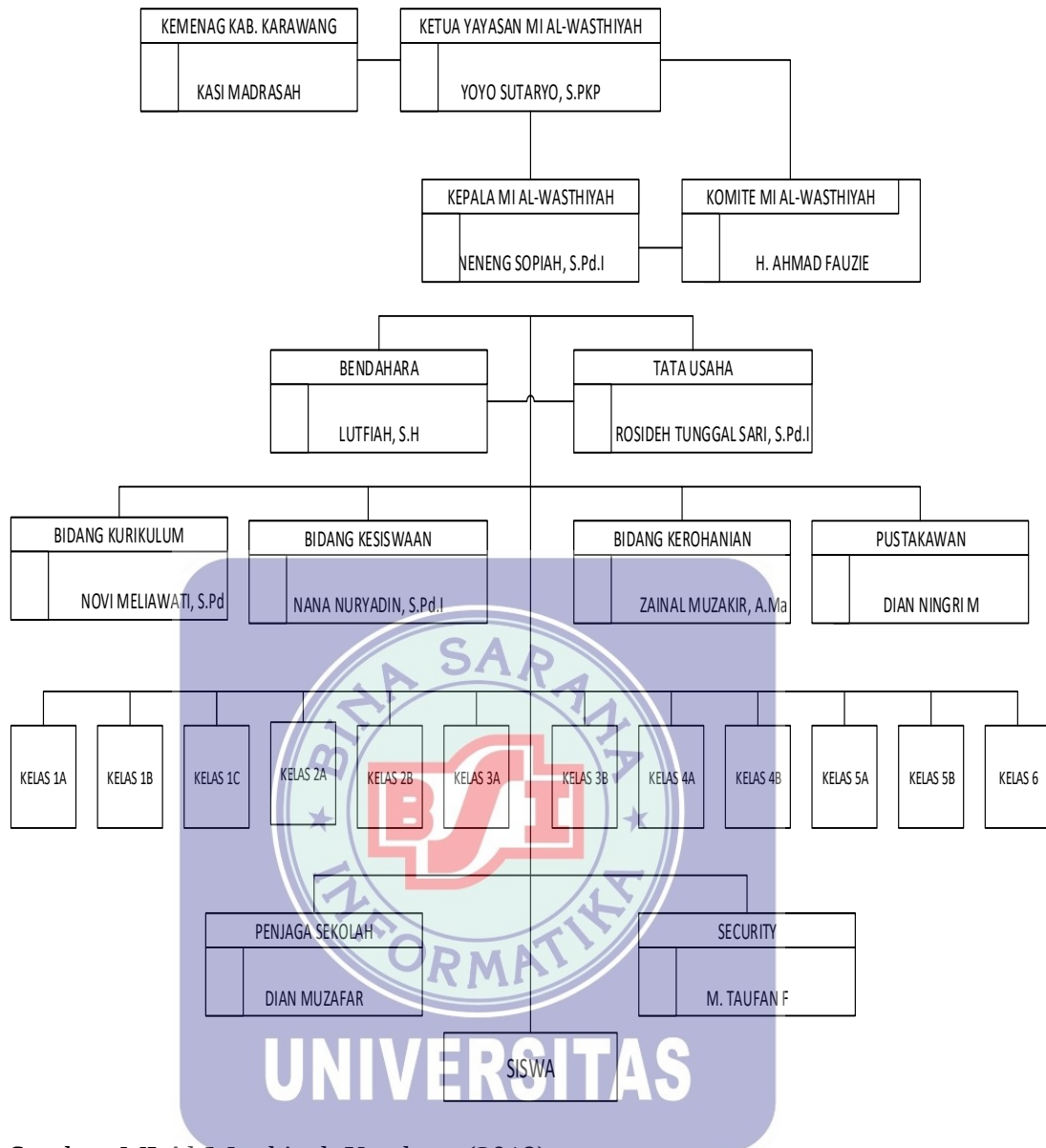
Visi dan Misi dari MI Al-Wasthiyah Kotabaru adalah sebagai berikut:

1. Visi: “Terwujudnya Manusia yang Berilmu, Beriman, Bertaqwa, Berkualitas dan Kompetitif”.
2. Misi:
 - a. Menumbuh kembangkan sikap dan amalah keagamaan Islam.
 - b. Menumbuhkan dan meningkatkan minat baca dan tulis.
 - c. Melaksanakan pembelajaran dan bimbingan yang kreatif, inovatif dan berkualitas.
 - d. Meningkatkan pencapaian rata-rata nilai Ujian Nasional (UN).
 - e. Mengembangkan kemampuan berbahasa Arab dan berbahasa Inggris.
 - f. Meningkatkan pencapaian prestasi akademik dan non akademik.
 - g. Memberdayakan lingkungan madrasah sebagai sumber belajar.
 - h. Menerapkan manajemen berbasis madrasah dengan melibatkan seluruh *Steakholder*.

3.1.2. Struktur Organisasi

Struktur Organisasi adalah suatu susunan dan hubungan antara tiap bagian secara posisi yang ada pada sekolah dalam menjalin kegiatan operasional untuk mencapai tujuan. Organisasi atau sekolah memegang peranan yang sangat penting. Setiap sekolah harus mempunyai struktur organisasi yang baik agar koordinasi dan tata kerja dapat tergambar dengan jelas, sehingga dapat berfungsi dengan maksimal.

Berikut struktur organisasi pada MI Al-Wasthiyyah Kotabaru adalah sebagai berikut:



Sumber: MI Al-Wasthiyah Kotabaru (2019)

Gambar III.1
Struktur Organisasi MI Al-Wasthiyah Kotabaru

Adapun tugas dan fungsi dari tiap bagian yang terdapat pada Struktur Organisasi MI Al-Wasthiyah Kotabaru:

1. Kementerian Agama (Kemenag) Karawang

Wewenang dan tanggung jawab dari Kementerian Agama yaitu:

- a. Perumusan dan penetapan visi, misi dan kebijakan teknis di bidang pelayanan dan bimbingan kehidupan beragama kepada masyarakat di kabupaten/kota.

- b. Pelaksanaan kebijakan teknis di bidang pengelolaan administrasi dan informasi.
- c. Pelayanan, bimbingan dan pembinaan di bidang haji dan umrah.
- d. Pelayanan, bimbingan dan pembinaan di bidang pendidikan madrasah, pendidikan agama dan keagamaan.
- e. Pembinaan kerukunan umat beragama.
- f. Pengkoordinasian perencanaan, pengendalian, pengawasan dan evaluasi program.
- g. Pelaksanaan hubungan dengan pemerintah daerah, instansi terkait dan lembaga masyarakat dalam rangka pelaksanaan tugas kementerian di kabupaten/kota.
(Pasal 8 PMA No. 13 Tahun 2013).

2. Ketua Yayasan

Wewenang dan tanggung jawab dari Ketua Yayasan yaitu:

- a. Merekrut kepala sekolah dan guru.
- b. Menentukan manajemen keuangan.
- c. Menandatangani kesepakatan kerja bersama dan SK sebagai legalitas dokumen.
- d. Menentukan kebijakan sekolah.

3. Kepala Sekolah

Wewenang dan tanggung jawab dari Kepala Sekolah, yaitu:

- a. Menjaga terlaksananya dan ketercapaian program kerja sekolah.
- b. Melakukan pengawasan dan penilaian kinerja tenaga pendidik dan kependidikan.
- c. Merencanakan, mengelola dan bertanggung jawabkan keuangan sekolah.
- d. Mengusulkan promosi dan mutu pendidik serta tenaga kependidikan.

- e. Mengangkat dan menetapkan personal struktur organisasi.
- f. Melegalisasi dokumen organisasi.
- g. Mengembangkan SDM.

4. Komite Sekolah

Wewenang dan tanggung jawab dari Komite Sekolah, yaitu:

- a. Membantu meningkatkan kelancaran penyelenggaraan kegiatan belajar mengajar di sekolah baik sarana, prasarana maupun teknis pendidikan.
- b. Melakukan penilaian sekolah untuk pengembangan pelaksanaan kurikulum, baik intra maupun ekstra kulikuler dan pelaksanaan manajemen sekolah, kepala sekolah atau wakil kepala sekolah, guru dan siswa.
- c. Melakukan pembahasan tentang usulan Rancangan Anggaran Pendapatan dan Belanja Sekolah.
- d. Memberikan penghargaan atas keberhasilan manajemen sekolah.
- e. Meminta sekolah agar mengadakan pertemuan untuk kepentingan tertentu.

5. Bendahara

Wewenang dan tanggung jawab dari Bendahara, yaitu:

- a. mengkoordinir dan melaksanakan pengumpulan sumbangan dari orang tua/wali siswa.
- b. Mempersiapkan rapat dengan pengurus Komite Sekolah dan orang tua/wali siswa dalam upaya dukungan dana.
- c. Mendayagunakan uang rutin sesuai dengan mata anggaran yang relevan.
- d. Menyampaikan pertanggungjawaban penggunaan uang rutin ke Kepala Sekolah.
- e. Membuat pertanggung jawaban keuangan sekolah dengan sebaik-baiknya.

6. Tata Usaha

Wewenang dan tanggung jawab dari Tata Usaha, yaitu:

- a. Merekap data serta mengarsipkan dokumen yang berkaitan dengan tata usaha.
- b. Membantu ketua tata usaha dalam menjalankan tugas administrasi sekolah.
- c. Menyusun laporan administrasi sekolah dan melaporkannya kepada ketua tata usaha secara berkala.

7. Bidang Kurikulum

Wewenang dan tanggung jawab dari Bidang Kurikulum, yaitu:

- a. Menyusun program pengajaran (program tahunan dan semester).
- b. Menyusun kalender pendidikan.
- c. Menyusun jadwal pelajaran.
- d. Menyusun program dan jadwal pelaksanaan Ujian Akhir Sekolah/Nasional.
- e. Menyusun jadwal penerimaan buku laporan pendidikan (Raport) dan penerimaan STTB/Ijasah dan STK.
- f. Mengatasi hambatan terhadap Kegiatan Belajar Mengajar (KBM).
- g. Mengkoordinasikan pelaksanaan KBM dan laporan pelaksanaan KBM.
- h. Menyusun laporan pelaksanaan pelajaran secara berkala.

8. Bidang Kesiswaan

Wewenang dan tanggung jawab dari Bidang Kesiswaan, yaitu:

- a. Menegakan tata tertib sekolah
- b. Mengadakan pemilihan siswa untuk mewakili sekolah dalam kegiatan diluar sekolah.
- c. Menyusun laporan pelaksanaan kegiatan kesiswaan secara berkala.
- d. Mengatur dan menyelenggarakan hubungan sekolah dengan orang tua siswa.

9. Bidang Kerohanian

Wewenang dan tanggung jawab dari Bidang Kerohanian, yaitu:

- a. Membahas dan menyelesaikan segala masalah yang ada menyangkut kerohanian.
- b. Membuat dan mengkoordinir kegiatan kerohanian di sekolah.
- c. Menjadi media penyampaian gagasan yang bermanfaat bagi seluruh warga sekolah.

10. Pustakawan

Wewenang dan tanggung jawab dari Pustakawan, yaitu:

- a. Pengurusan pelaksanaan perpustakaan.
- b. Perencanaan dan pengembangan perpustakaan.
- c. Pemeliharaan dan perbaikan buku perpustakaan.
- d. Penyimpanan buku-buku perpustakaan.
- e. Melaksanakan inventarisasi perpustakaan.
- f. Melayani pemakai perpustakaan.
- g. Mengusahakan pengadaan buku baru.
- h. Menjaga dan melaksanakan kegiatan keamanan, kebersihan, ketertiban, keindahan dan kekeluargaan.

11. Security

Wewenang dan tanggung jawab dari Security, yaitu:

- a. melaksanakan tugas pengamanan sekolah.
- b. memonitor lingkungan sekolah.
- c. mengawasi dan menjaga keamanan lahan parkir sekolah.

12. Penjaga Sekolah

Wewenang dan tanggung jawab dari Penjaga Sekolah, yaitu:

- a. melaksanakan tugas kebersihan.

- b. Membantu menyediakan kebutuhan barang-barang yang diperlukan sekolah.
- c. Melakukan tugas belanja makan/minum, foto *copy*, mengantar surat dan tugas sejenis lainnya.
- d. Memelihara dan menjaga barang-barang milik sekolah.

13. Siswa

Wewenang dan tanggung jawab dari Siswa, yaitu:

- a. Menuntut ilmu sebaik-baiknya.
- b. Mempertanggungjawabkan hasil pembelajarannya.
- c. Mematuhi peraturan yang sudah ditetapkan oleh pihak sekolah.

3.2. Tinjauan Kasus

Kebutuhan akan teknologi informasi dan komunikasi di era modern saat ini sangatlah penting untuk berbagai perusahaan dan instansi dalam mengambil keputusan ataupun mengetahui efektivitas suatu proses yang berjalan, sehingga dapat terlihat langkah apa saja yang harus diambil dalam menangani segala permasalahan. Madrasah Ibtidaiyah (MI) Al-Wasthiyah Kotabaru merupakan Sekolah Dasar Islam. Pencatatan dan pengolahan data gaji pada MI Al-Wasthiyah yang kurang maksimal dapat menghambat Sekolah dalam mengambil suatu keputusan. Pada kesempatan kali ini penulis bermaksud untuk membuat sistem yang manual menjadi lebih praktis dan efisien dengan mengubahnya menjadi sistem yang terkomputerisasi agar dapat mempermudah Sekolah dalam mendapatkan suatu informasi yang nantinya akan menjadi bahan pertimbangan menentukan langkah apa yang harus diambil. Dalam tinjauan kasus ini penulis membatasi ruang dengan menjelaskan proses bisnis sistem berjalan, *activity diagram* berjalan, dokumen yang digunakan, permasalahan beserta pemecahannya.

3.2.1. Proses Bisnis Sistem Berjalan

Prosedur yang dilakukan sebelum memulai suatu kegiatan untuk menyelesaikan suatu pekerjaan disebut prosedur sistem. Sesuai dengan ruang lingkup yang dibahas dalam tugas akhir ini, maka proses bisnis sistem berjalan yang dibahas yaitu tentang sistem informasi akuntansi pembayaran gaji guru dan karyawan pada MI Al-Wasthiyah Kotabaru sebagai berikut:

1. Rekap absensi guru dan karyawan

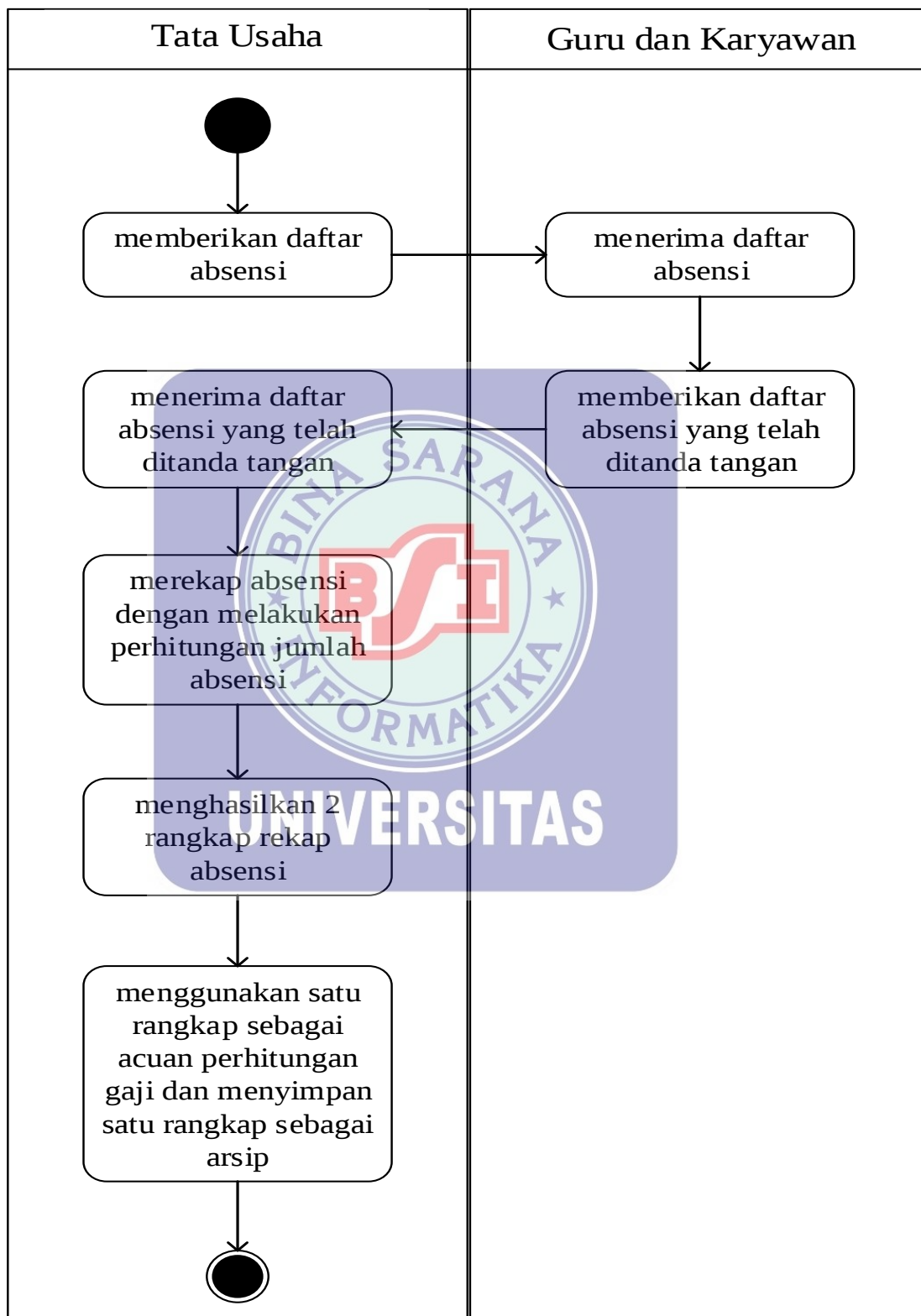
Bagian tata usaha memberikan daftar absensi setiap hari untuk ditanda tangani guru dan karyawan sebagai bukti absensi hadir. Setiap akhir bulan tata usaha merekap absensi guru dengan melakukan perhitungan jumlah absensi. Hasil dari rekap absensi tersebut menghasilkan dua rangkap rekap absensi, satu rangkap untuk disimpan sebagai arsip dan satu lagi untuk menjadi acuan perhitungan gaji guru dan karyawan.

2. Perhitungan dan pembayaran gaji

Bagian tata usaha menghitung gaji guru dan karyawan yang mengacu dari daftar rekap absensi guru dan karyawan. Kemudian tata usaha memberikan lembar penerimaan gaji kepada kepala sekolah untuk ditanda tangani. Kemudian lembar penerimaan gaji tersebut diberikan kembali kepada tata usaha untuk memberikan gaji serta lembar penerimaan gaji kepada guru dan karyawan untuk ditandatangani sebagai bukti gaji telah diterima oleh guru dan karyawan. Setelahnya guru dan karyawan memberikan kembali lembar penerimaan gaji kepada tata usaha untuk disimpan sebagai arsip.

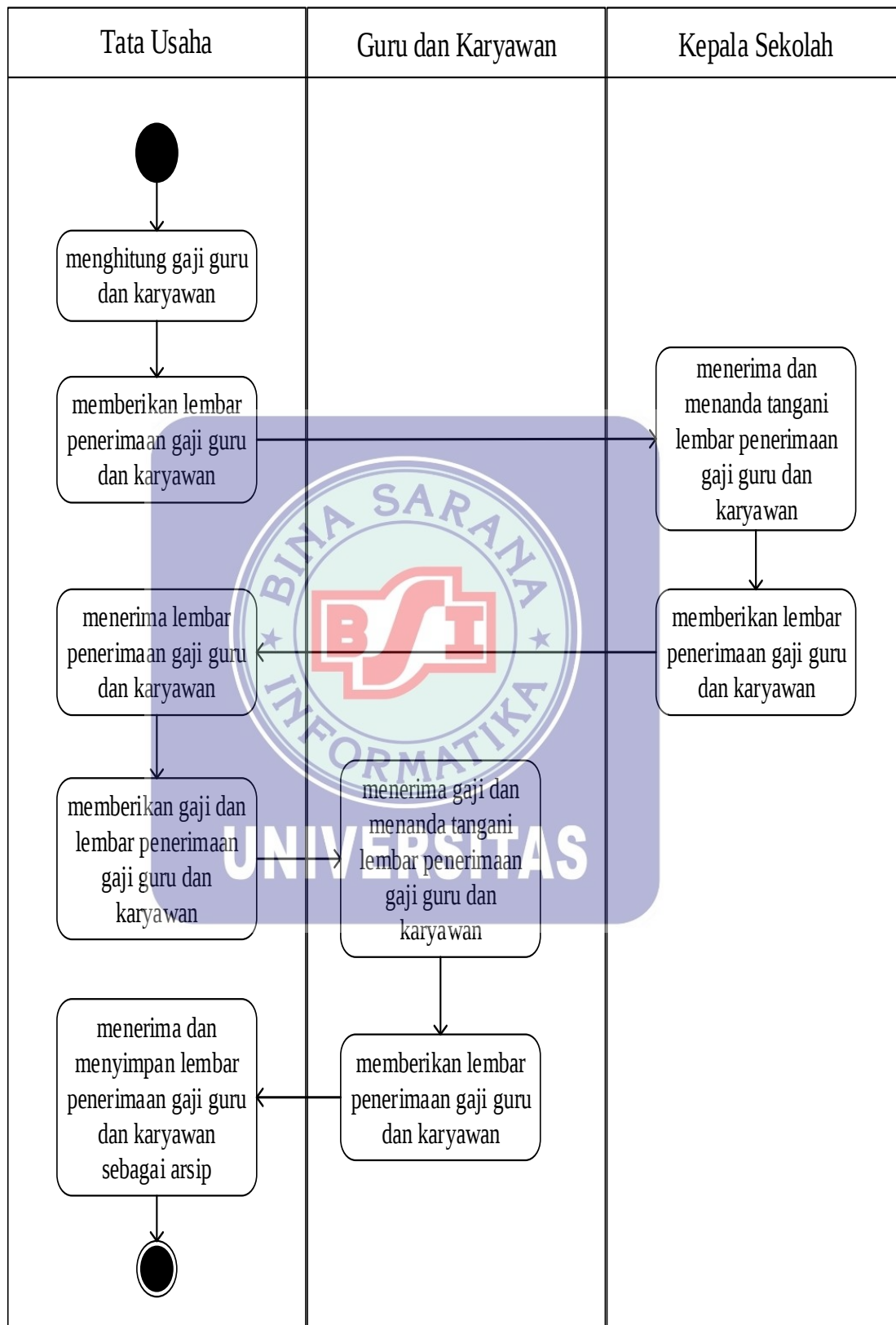
3.2.2. Activity Diagram Sistem Berjalan

1. Activity Diagram Rekap Absensi Guru dan Karyawan



Gambar III.2
Activity Diagram Rekap Absensi Guru dan Karyawan

2. Activity Diagram Perhitungan dan Pembayaran Gaji



Gambar III.3
Activity Diagram Perhitungan dan Pembayaran Gaji

3.2.3. Dokumen Masukan (*Input*)

Dokumen masukan adalah semua dokumen yang digunakan sebagai dasar untuk memperoleh data-data yang nantinya akan diproses untuk menghasilkan suatu keluaran yang disebut dengan *output*. Dokumen yang digunakan adalah sebagai berikut:

1. Nama Dokumen : Daftar absensi

Fungsi : Sebagai bukti kehadiran

Sumber : Tata usaha

Tujuan : Guru dan karyawan

Media : Kertas

Frekuensi : Setiap hari

Bentuk : Lampiran A1

2. Nama Dokumen : Daftar rekap absensi

Fungsi : Sebagai keterangan kehadiran

Sumber : Tata usaha

Tujuan : Tata usaha dan Kepala Sekolah

Media : Kertas

Frekuensi : Setiap bulan

Bentuk : Lampiran A2

3.2.4. Dokumen Keluaran (*Output*)

Dokumen keluaran adalah segala bentuk dokumen yang akan mendukung kegiatan manajemen serta merupakan dokumen dari hasil catatan laporan. Dokumen keluaran yang digunakan adalah sebagai berikut:

1. Nama Dokumen : Lembar penerimaan gaji guru dan karyawan

Fungsi : Sebagai bukti penerimaan gaji

Sumber : Tata Usaha

Tujuan : Guru dan Karyawan

Media : Kertas

Frekuensi : Setiap bulan

Bentuk : Lampiran B1

2. Nama Dokumen : Lembar penerimaan gaji guru dan karyawan

Fungsi : Sebagai bahan persetujuan bukti penerimaan gaji

Sumber : Tata Usaha

Tujuan : Kepala Sekolah

Media : Kertas

Frekuensi : Setiap satu semester

Bentuk : Lampiran B2

3.2.5. Permasalahan Pokok

Berdasarkan hasil pengamatan yang penulis lakukan di MI Al-Wasthiyah Kotabaru, penulis mengetahui permasalahan yang ada MI Al-Wasthiyah Kotabaru dalam sistem pembayaran gaji guru dan karyawan yaitu sebagai berikut:

1. Belum adanya Slip Gaji sehingga kurang adanya bukti mengenai kejelasan komponen gaji yang diterima selama bekerja dalam suatu periode.
2. Belum adanya Laporan untuk kepala sekolah sehingga tidak efektif untuk proses pengecekan pembayaran gaji tersebut pada MI Al-Wasthiyah.
3. Keamanan dokumen kurang terjamin karena belum adanya *database* sebagai penyimpanan dokumen tersebut.

4. Sistem manual membuat data-data khususnya data penggajian menjadi lebih sulit dicari kembali dan rentan terhadap resiko terjadi hilangnya salah satu data penggajian karena terselip dengan berkas data yang lain.

3.2.6. Pemecahan Masalah

Pemecahan masalah yang dihadapkan pada MI Al-Wasthiyah Kotabaru lebih mendasar dan terletak pada sumber daya manusia serta sistem yang dijalankan di sekolah tersebut, sehingga manajemen sekolah harus melakukan tindakan efektif dan efisien terhadap permasalahan yang timbul serta dapat mengambil langkah-langkah yang sesuai dengan manajemen sekolah. Adapun pemecahan permasalahan adalah sebagai berikut:

1. Dibuatkannya Slip Gaji sebagai bukti mengenai kejelasan komponen gaji yang diterima selama bekerja dalam suatu periode.
2. Dibuatkannya Laporan pembayaran gaji agar menghasilkan informasi yang lebih jelas untuk kepala sekolah.
3. Beralih dari sistem manual ke sistem komputerisasi berbasis desktop dengan menggunakan aplikasi *Java Netbeans* dan server *XAMPP*.
4. Sistem penggajian yang diusulkan memiliki hak akses sehingga hanya orang-orang tertentu saja yang dapat membuka data penggajian atau data-data lainnya sehingga data tersebut tidak rusak dan tidak mudah hilang.

2.3. Analisa Kebutuhan Software

Analisa kebutuhan merupakan langkah awal untuk menentukan gambaran perangkat yang akan dihasilkan ketika pengembang melaksanakan sebuah proyek pembuatan perangkat lunak.

3.3.1. Analisis Kebutuhan

Berdasarkan proses penggajian sistem berjalan pada MI Al-Wasthiyah Kotabaru, maka tahapan berikutnya adalah analisis kebutuhan. Berikut ini spesifikasi kebutuhan dari sistem penggajian pada MI Al-Wasthiyah Kotabaru.

A. Bagian Administrasi

A.1. Administrasi dapat melakukan *login*

A.2. Dapat mengelola data pegawai

A.3. Dapat mengelola data jabatan

A.4. Dapat mengelola data akun

A.4. Dapat mengelola data gaji

A.5. Dapat mencetak laporan gaji

A.6. Dapat mencetak jurnal

B. Kepala Sekolah

B.1. Kepala sekolah dapat melakukan *login*

B.2. Dapat mengelola data admin

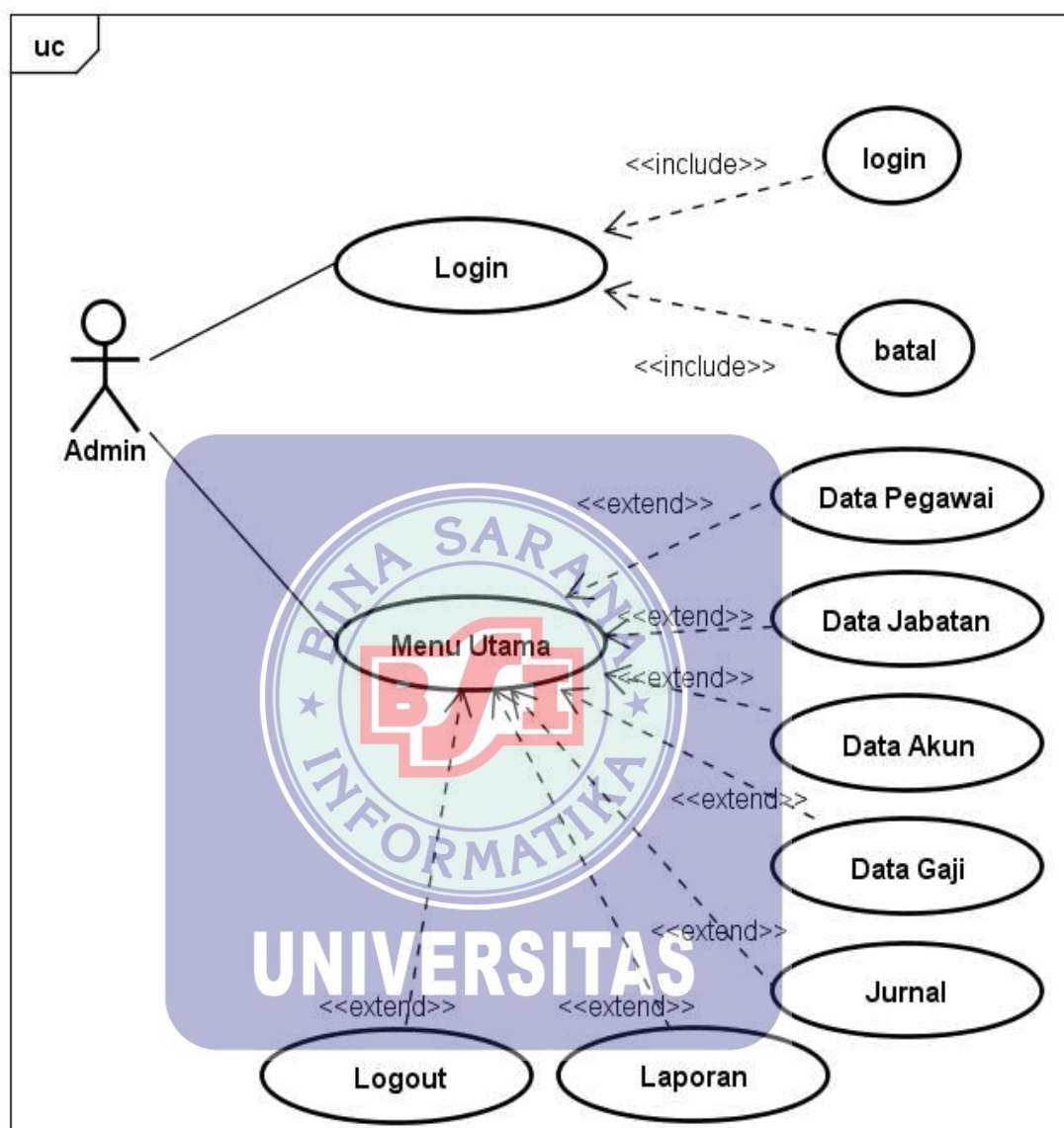
B.3. Dapat mencetak laporan gaji

B.4. Dapat mencetak jurnal



3.3.2. Use Case Diagram

1. Use Case Diagram Login Admin



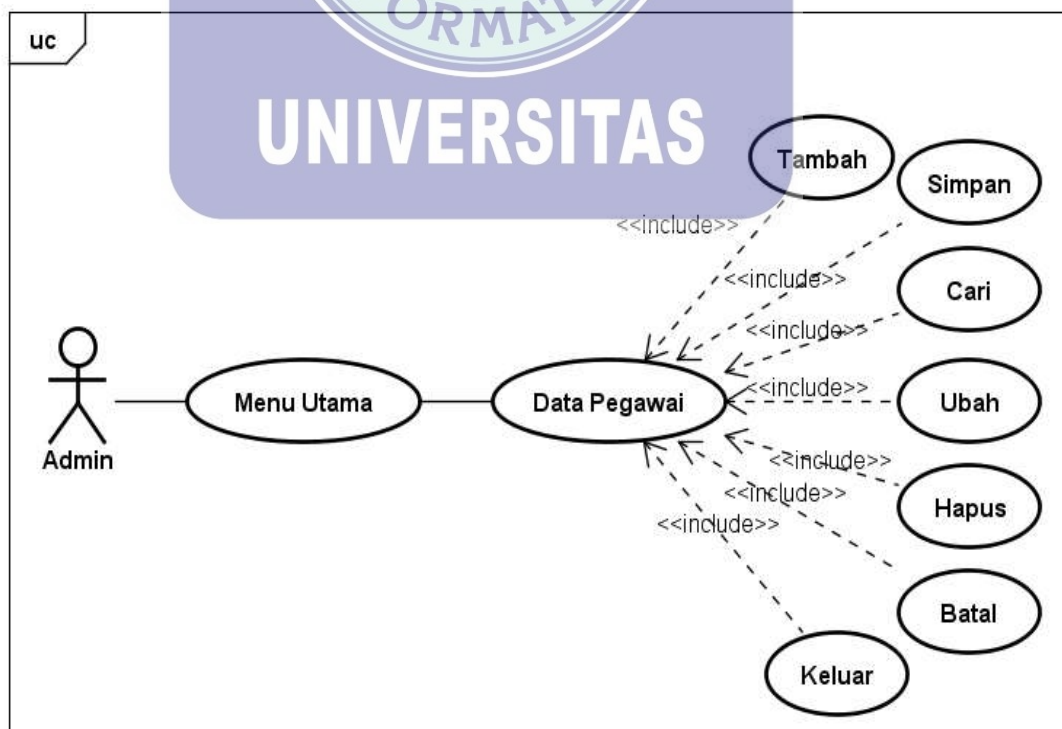
Gambar III.4
Use Case Diagram Login Admin

Tabel III.1
Deskripsi Use Case Diagram Login Admin

Deskripsi Use Case Diagram Login Admin	
Tujuan	Administrasi dapat melakukan pengolahan data yang terdapat di menu utama
Deskripsi	Sistem ini memungkinkan aktor untuk mengakses sistem pembayaran gaji

Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor melakukan <i>login</i> atau aktivitas
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol <i>login</i>	Sistem akan masuk ke menu utama
2. Aksi aktor memilih tombol batal	Sistem akan membersihkan <i>field username</i> dan <i>password</i>
3. Aksi aktor memilih data pegawai	Sistem akan menampilkan informasi data pegawai
4. Aksi aktor memilih data jabatan	Sistem akan menampilkan informasi data jabatan
5. Aksi aktor memilih data akun	Sistem akan menampilkan informasi data akun
5. Aksi aktor memilih data gaji	Sistem akan menampilkan informasi data gaji
6. Aksi aktor memilih jurnal	Sistem akan menampilkan informasi jurnal
7. Aksi aktor memilih laporan	Sistem akan menampilkan informasi laporan
8. Aksi aktor memilih <i>logout</i>	Sistem akan menampilkan form <i>login</i>
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

2. Use Case Diagram Data Pegawai

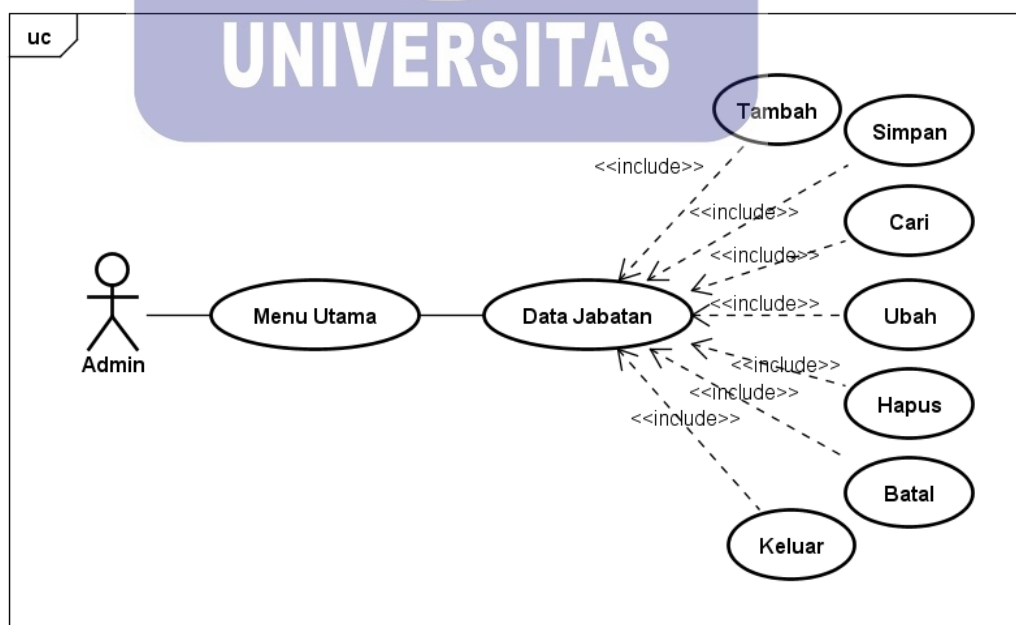


Gambar III.5
Use Case Diagram Data Pegawai

Tabel III.2
Deskripsi Use Case Diagram Data Pegawai

Deskripsi Use Case Diagram Data Pegawai	
Tujuan	Administrasi dapat mengakses data pegawai
Deskripsi	Sistem ini memungkinkan aktor untuk melakukan pengolahan data pegawai
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka data pegawai
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol tambah	Sistem akan menampilkan <i>field</i> untuk diinput
2. Aksi aktor memilih tombol simpan	Sistem akan menyimpan data
3. Aksi aktor memilih tombol cari	Sistem akan menampilkan data yang dicari
4. Aksi aktor memilih tombol ubah	Sistem akan mengubah data
5. Aksi aktor memilih tombol hapus	Sistem akan menghapus data
6. Aksi aktor memilih tombol batal	Sistem akan membersihkan <i>field</i>
7. Aksi aktor memilih tombol keluar	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

3. Use Case Diagram Data Jabatan

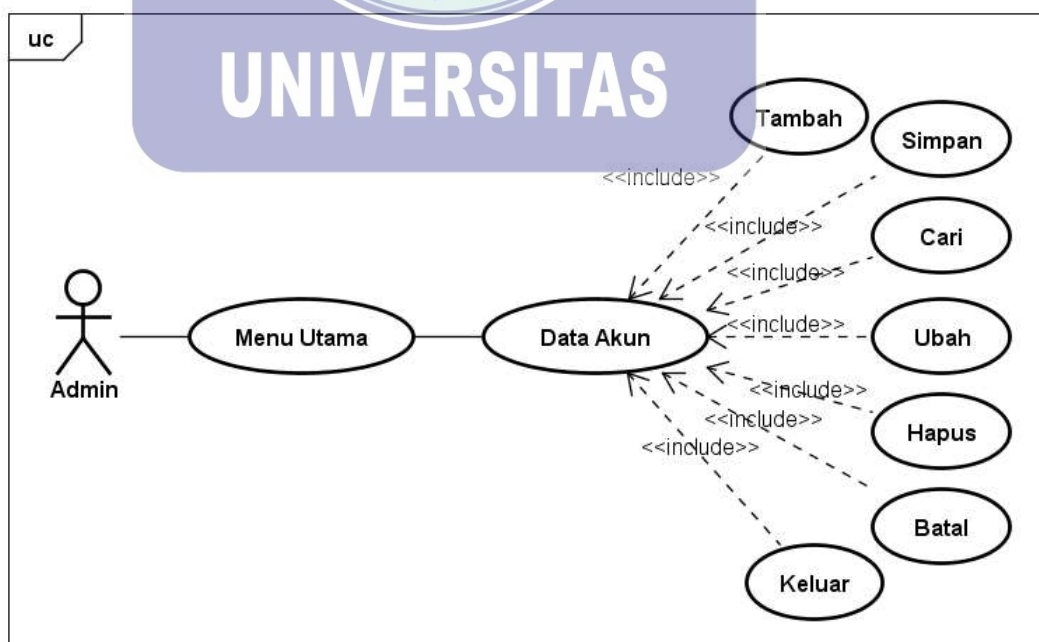


Gambar III.6
Use Case Diagram Data Jabatan

Tabel III.3
Deskripsi Use Case Diagram Data Jabatan

Deskripsi Use Case Diagram Data Jabatan	
Tujuan	Administrasi dapat mengakses data jabatan
Deskripsi	Sistem ini memungkinkan aktor untuk melakukan pengolahan data jabatan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka data jabatan
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol tambah	Sistem akan menampilkan <i>field</i> untuk diinput
2. Aksi aktor memilih tombol simpan	Sistem akan menyimpan data
3. Aksi aktor memilih tombol cari	Sistem akan menampilkan data yang dicari
4. Aksi aktor memilih tombol ubah	Sistem akan mengubah data
5. Aksi aktor memilih tombol hapus	Sistem akan menghapus data
6. Aksi aktor memilih tombol batal	Sistem akan membersihkan <i>field</i>
7. Aksi aktor memilih tombol keluar	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

4. Use Case Diagram Data Akun

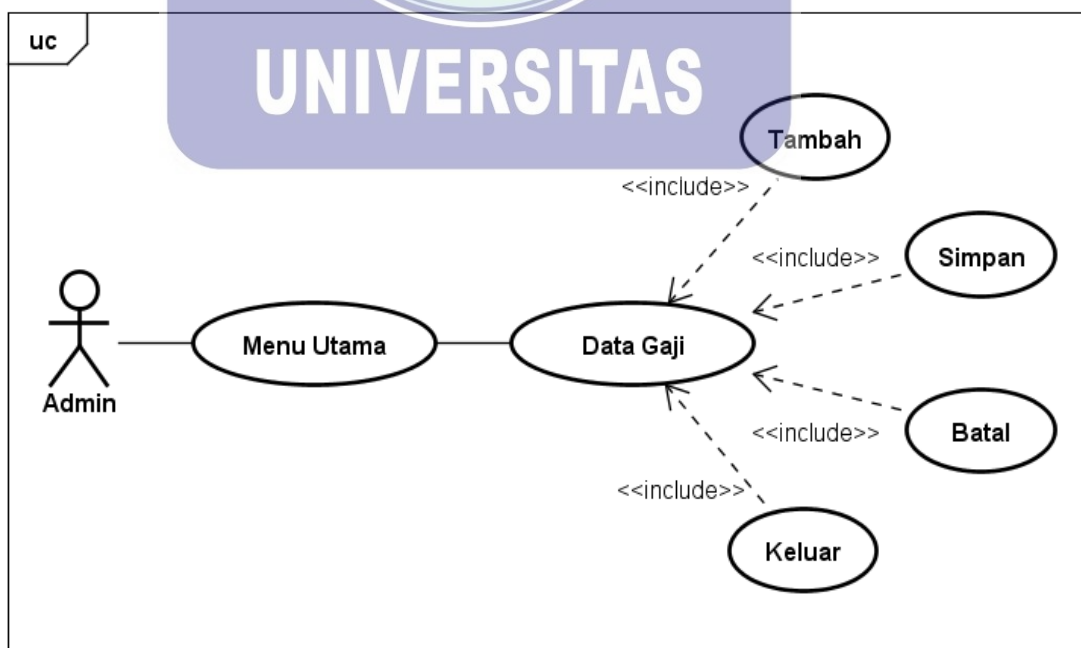


Gambar III.7
Use Case Diagram Data Akun

Tabel III.4
Deskripsi Use Case Diagram Data Akun

Deskripsi Use Case Diagram Data Akun	
Tujuan	Administrasi dapat mengakses data akun
Deskripsi	Sistem ini memungkinkan aktor untuk melakukan pengolahan data akun
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka data akun
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol tambah	Sistem akan menampilkan <i>field</i> untuk diinput
2. Aksi aktor memilih tombol simpan	Sistem akan menyimpan data
3. Aksi aktor memilih tombol cari	Sistem akan menampilkan data yang dicari
4. Aksi aktor memilih tombol ubah	Sistem akan mengubah data
5. Aksi aktor memilih tombol hapus	Sistem akan menghapus data
6. Aksi aktor memilih tombol batal	Sistem akan membersihkan <i>field</i>
7. Aksi aktor memilih tombol keluar	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

5. Use Case Diagram Data Gaji

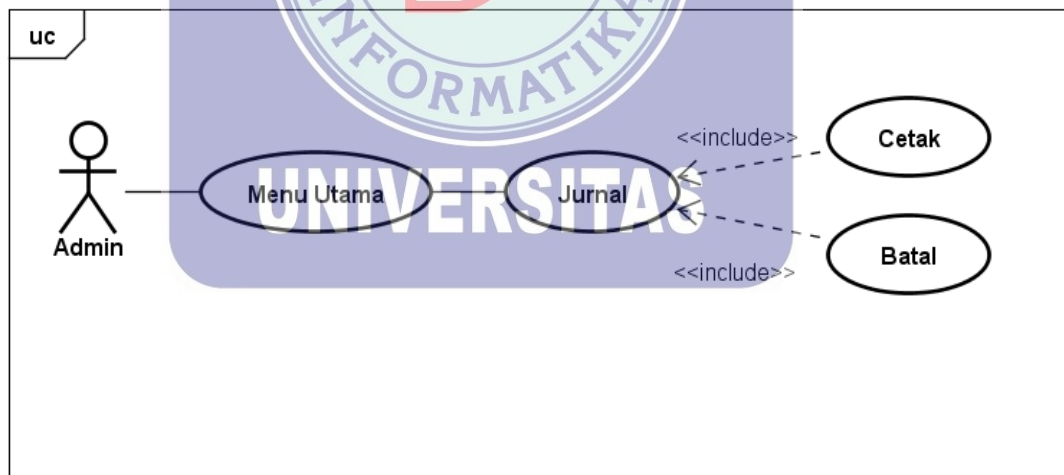


Gambar III.8
Use Case Diagram Data Gaji

Tabel III.5
Deskripsi Use Case Diagram Data Gaji

Deskripsi Use Case Diagram Data Gaji	
Tujuan	Administrasi dapat mengakses data gaji
Deskripsi	Sistem ini memungkinkan aktor untuk melakukan pengolahan data gaji
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka data gaji
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol tambah	Sistem akan menampilkan <i>field</i> untuk diinput
2. Aksi aktor memilih tombol simpan	Sistem akan menyimpan data
6. Aksi aktor memilih tombol batal	Sistem akan membersihkan <i>field</i>
7. Aksi aktor memilih tombol keluar	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

6. Use Case Diagram Jurnal Admin



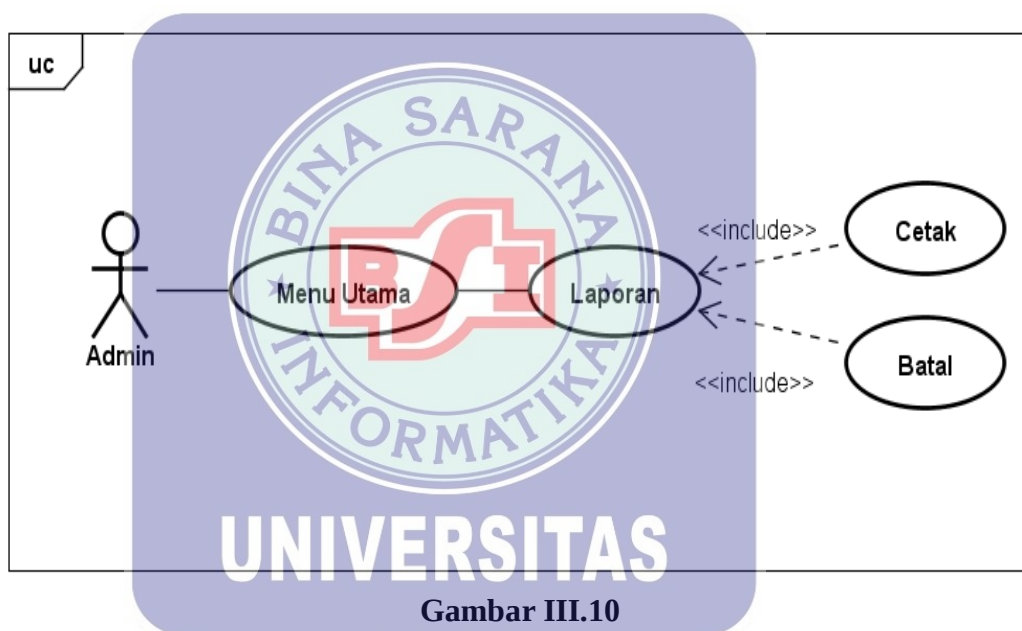
Gambar III.9
Use Case Diagram Jurnal Admin

Tabel III.6
Deskripsi Use Case Diagram Jurnal Admin

Deskripsi Use Case Diagram jurnal	
Tujuan	Administrasi dapat mengakses jurnal
Deskripsi	Sistem ini memungkinkan aktor untuk melihat serta mencetak jurnal

Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka jurnal
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol cetak	Sistem akan mencetak jurnal
2. Aksi aktor memilih tombol batal	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

7. Use Case Diagram Laporan Admin



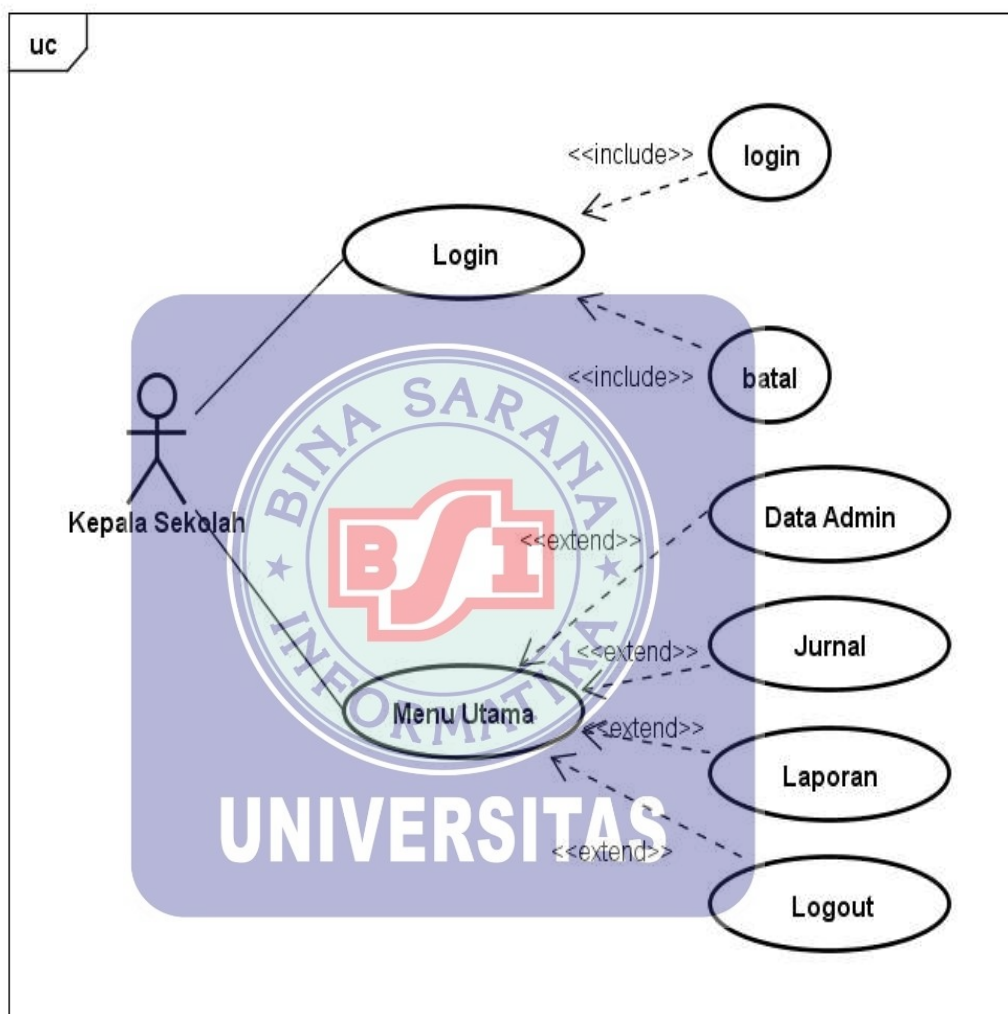
Gambar III.10
Use Case Diagram Laporan Admin

Tabel III.7
Deskripsi Use Case Diagram Laporan Admin

Deskripsi Use Case Diagram laporan	
Tujuan	Administrasi dapat mengakses laporan
Deskripsi	Sistem ini memungkinkan aktor untuk melihat serta mencetak laporan
Skenario Utama	
Aktor	Administrasi
Kondisi Awal	Aktor membuka laporan
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol cetak	Sistem akan mencetak laporan

2. Aksi aktor memilih tombol batal	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

8. Use Case Diagram Login Kepala Sekolah



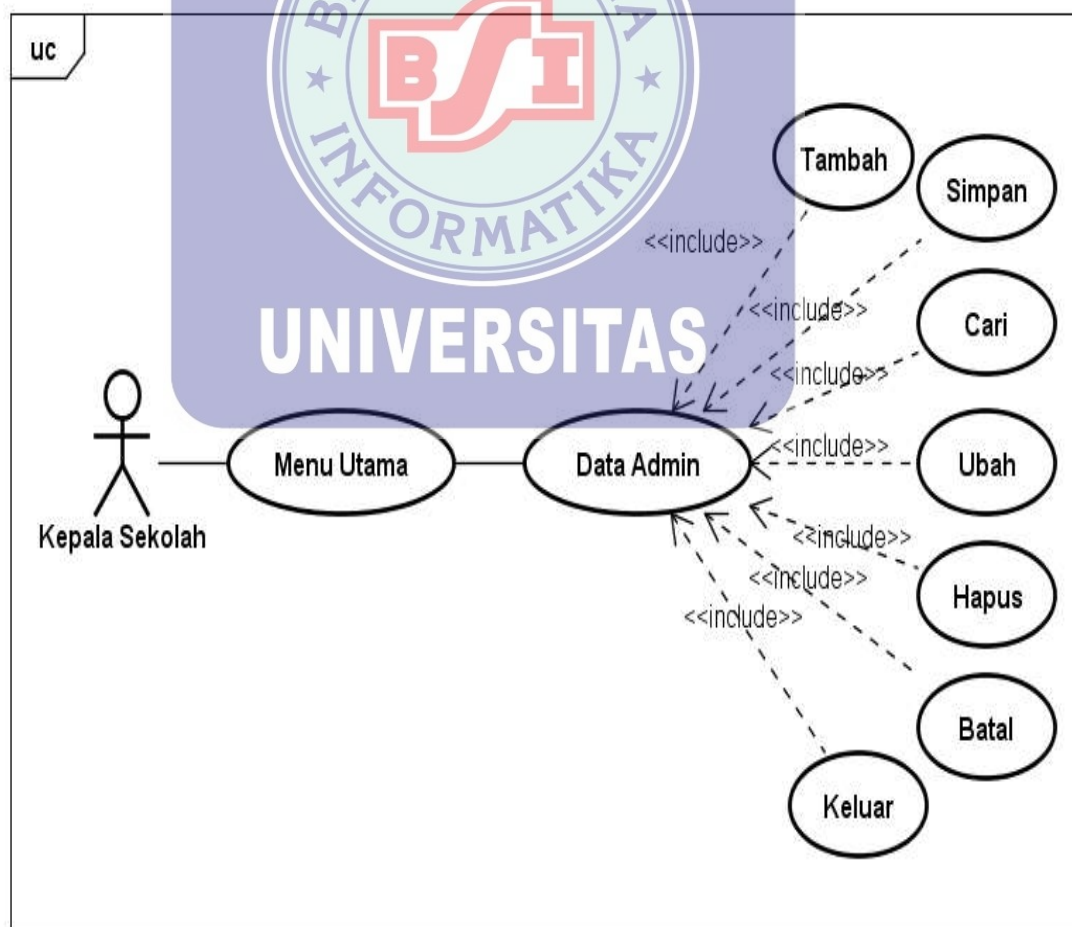
Gambar III.11
Use Case Diagram Login Kepala Sekolah

Tabel III.8
Deskripsi Use Case Diagram Login Kepala Sekolah

Deskripsi Use Case Diagram Login Kepala Sekolah	
Tujuan	Kepala Sekolah dapat melakukan pengolahan data yang terdapat di menu utama
Deskripsi	Sistem ini memungkinkan aktor untuk mengakses sistem pembayaran gaji

Skenario Utama	
Aktor	Kepala Sekolah
Kondisi Awal	Aktor melakukan <i>login</i> atau aktivitas
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol <i>login</i>	Sistem akan masuk ke menu utama
2. Aksi aktor memilih tombol batal	Sistem akan membersihkan <i>field username</i> dan <i>password</i>
3. Aksi aktor memilih data admin	Sistem akan menampilkan informasi data admin
4. Aksi aktor memilih jurnal	Sistem akan menampilkan informasi jurnal
5. Aksi aktor memilih laporan	Sistem akan menampilkan informasi laporan
6. Aksi aktor memilih <i>logout</i>	Sistem akan menampilkan form <i>login</i>
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

9. Use Case Diagram Data Admin

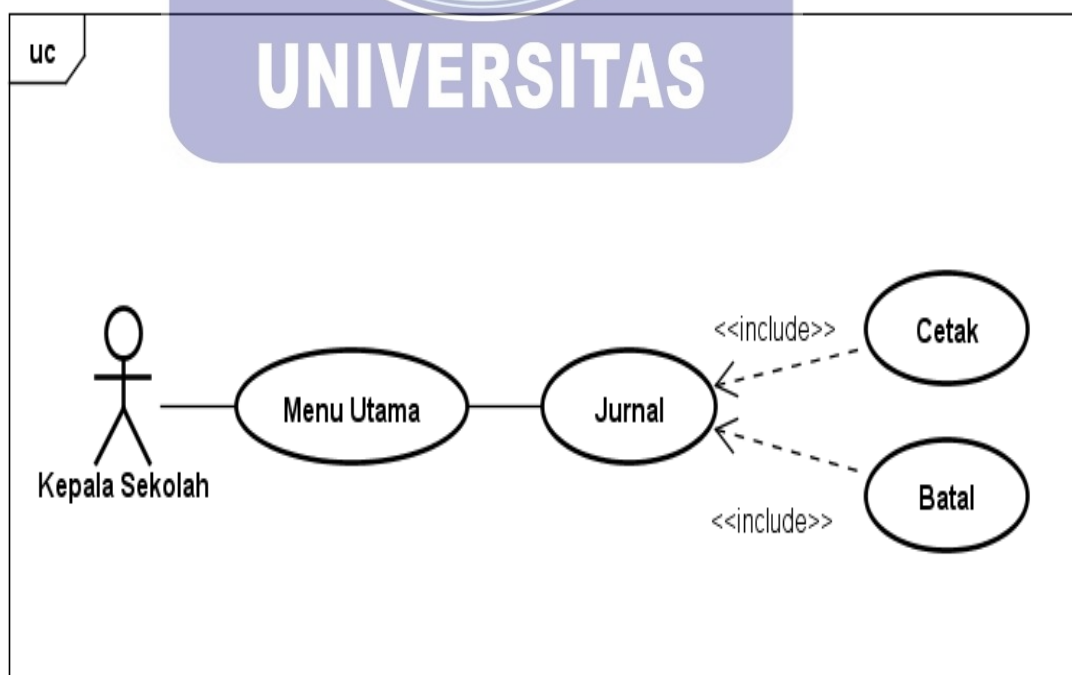


Gambar III.12
Use Case Diagram Data Admin

Tabel III.9
Deskripsi Use Case Diagram Data Admin

Deskripsi Use Case Diagram Data Admin	
Tujuan	Administrasi dapat mengakses data admin
Deskripsi	Sistem ini memungkinkan aktor untuk melakukan pengolahan data admin
Skenario Utama	
Aktor	Kepala Sekolah
Kondisi Awal	Aktor membuka data admin
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol tambah	Sistem akan menampilkan <i>field</i> untuk diinput
2. Aksi aktor memilih tombol simpan	Sistem akan menyimpan data
3. Aksi aktor memilih tombol cari	Sistem akan menampilkan data yang dicari
4. Aksi aktor memilih tombol ubah	Sistem akan mengubah data
5. Aksi aktor memilih tombol hapus	Sistem akan menghapus data
6. Aksi aktor memilih tombol batal	Sistem akan membersihkan <i>field</i>
7. Aksi aktor memilih tombol keluar	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

10. Use Case Diagram Jurnal Kepala Sekolah

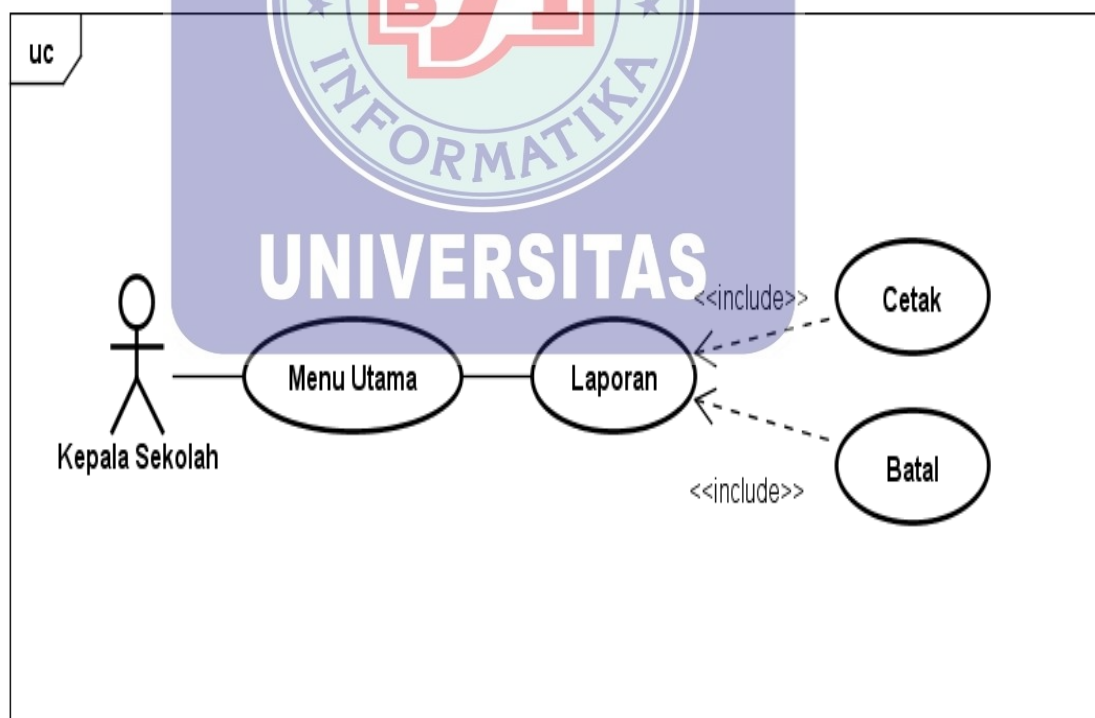


Gambar III.13
Use Case Diagram Jurnal Kepala Sekolah

Tabel III.10
Deskripsi Use Case Diagram Jurnal Kepala Sekolah

Deskripsi Use Case Diagram Jurnal Kepala Sekolah	
Tujuan	Kepala Sekolah dapat mengakses jurnal
Deskripsi	Sistem ini memungkinkan aktor untuk melihat serta mencetak jurnal
Skenario Utama	
Aktor	Kepala Sekolah
Kondisi Awal	Aktor membuka jurnal
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol cetak	Sistem akan mencetak jurnal
2. Aksi aktor memilih tombol batal	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan

11. Use Case Diagram Laporan Kepala Sekolah



Gambar III.14
Use Case Diagram Laporan Kepala Sekolah

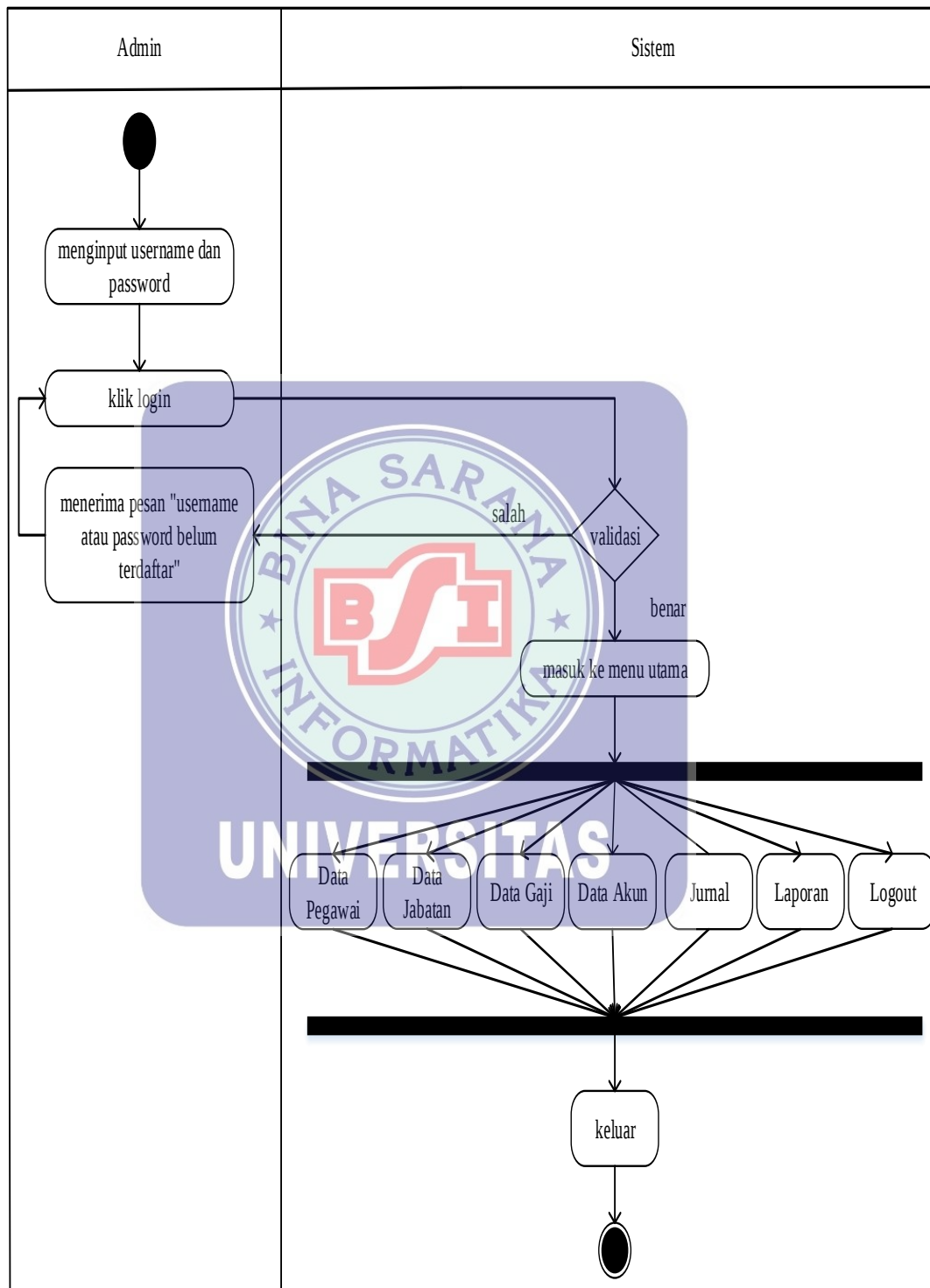
Tabel III.11
Deskripsi Use Case Diagram Laporan Kepala Sekolah

Deskripsi Use Case Diagram Laporan Kepala Sekolah	
Tujuan	Kepala Sekolah dapat mengakses laporan
Deskripsi	Sistem ini memungkinkan aktor untuk melihat serta mencetak laporan
Skenario Utama	
Aktor	Kepala Sekolah
Kondisi Awal	Aktor membuka laporan
Aksi Aktor	Reaksi Aktor
1. Aksi aktor memilih tombol cetak	Sistem akan mencetak laporan
2. Aksi aktor memilih tombol batal	Sistem akan kembali ke menu utama
kondisi akhir	Jika perintah sesuai maka sistem akan menampilkan apa yang diinginkan



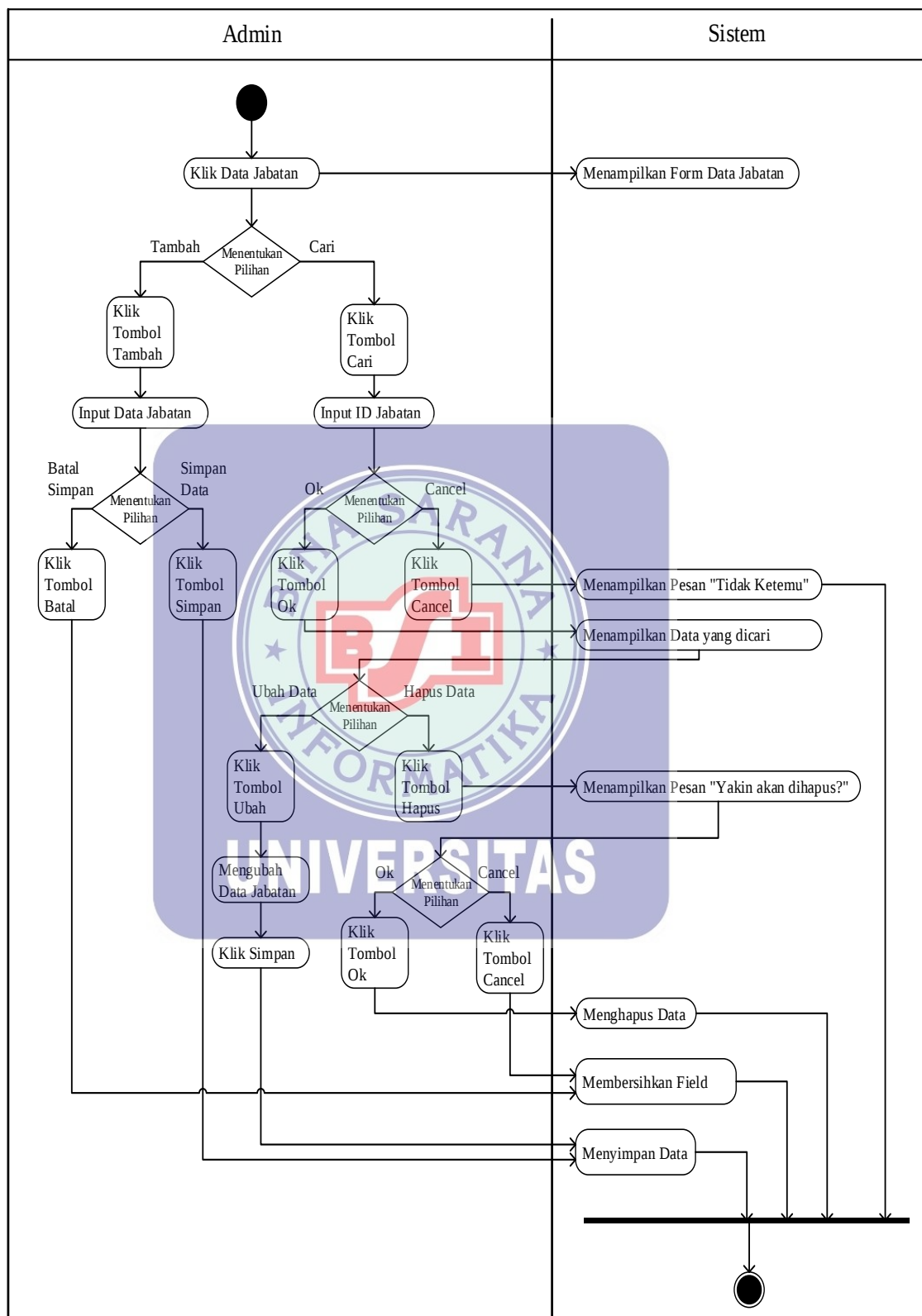
3.3.3. Activity Diagram

1. Activity Diagram Login Admin



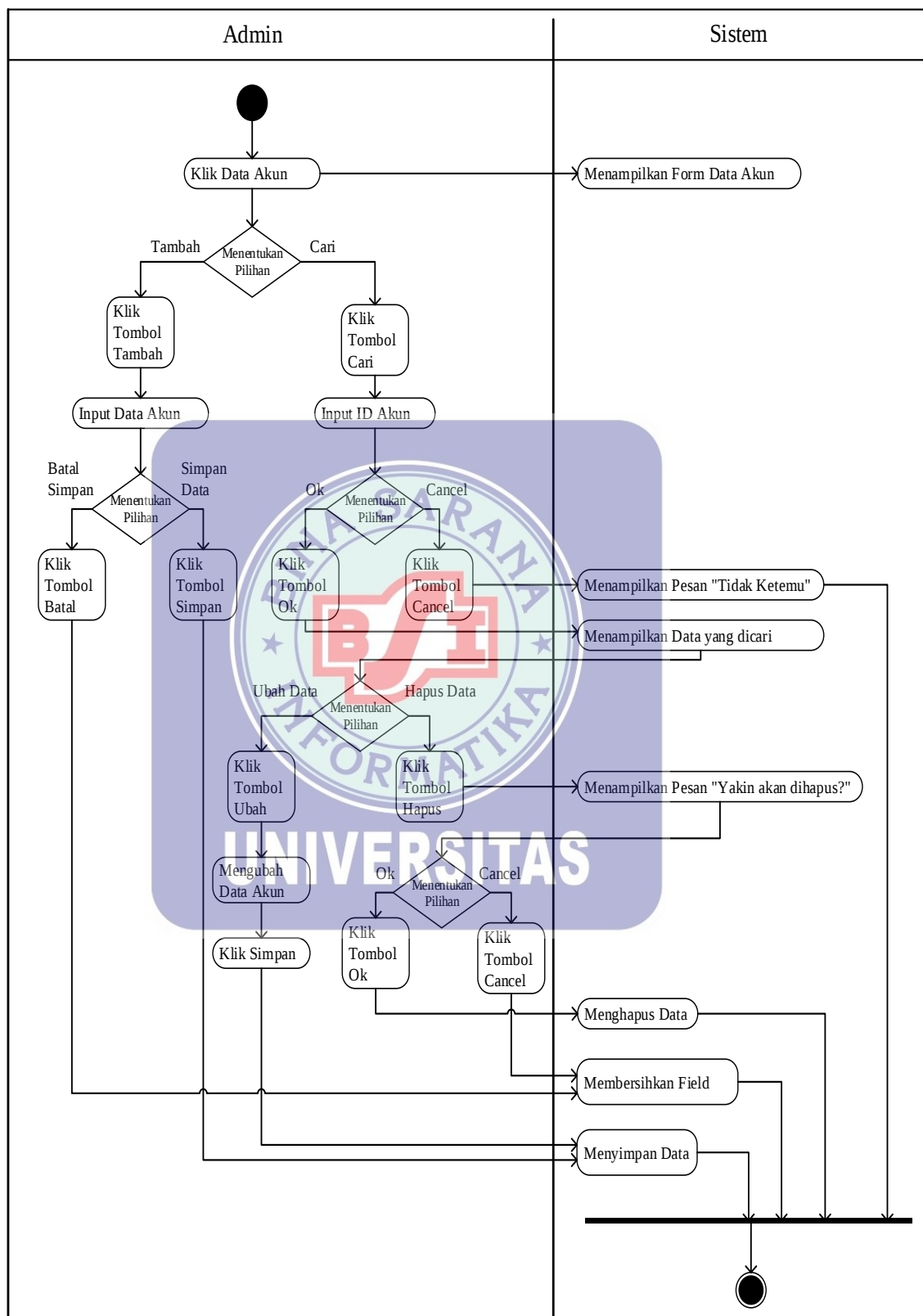
Gambar III.15
Activity Diagram Login Admin

3. Activity Diagram Data Jabatan



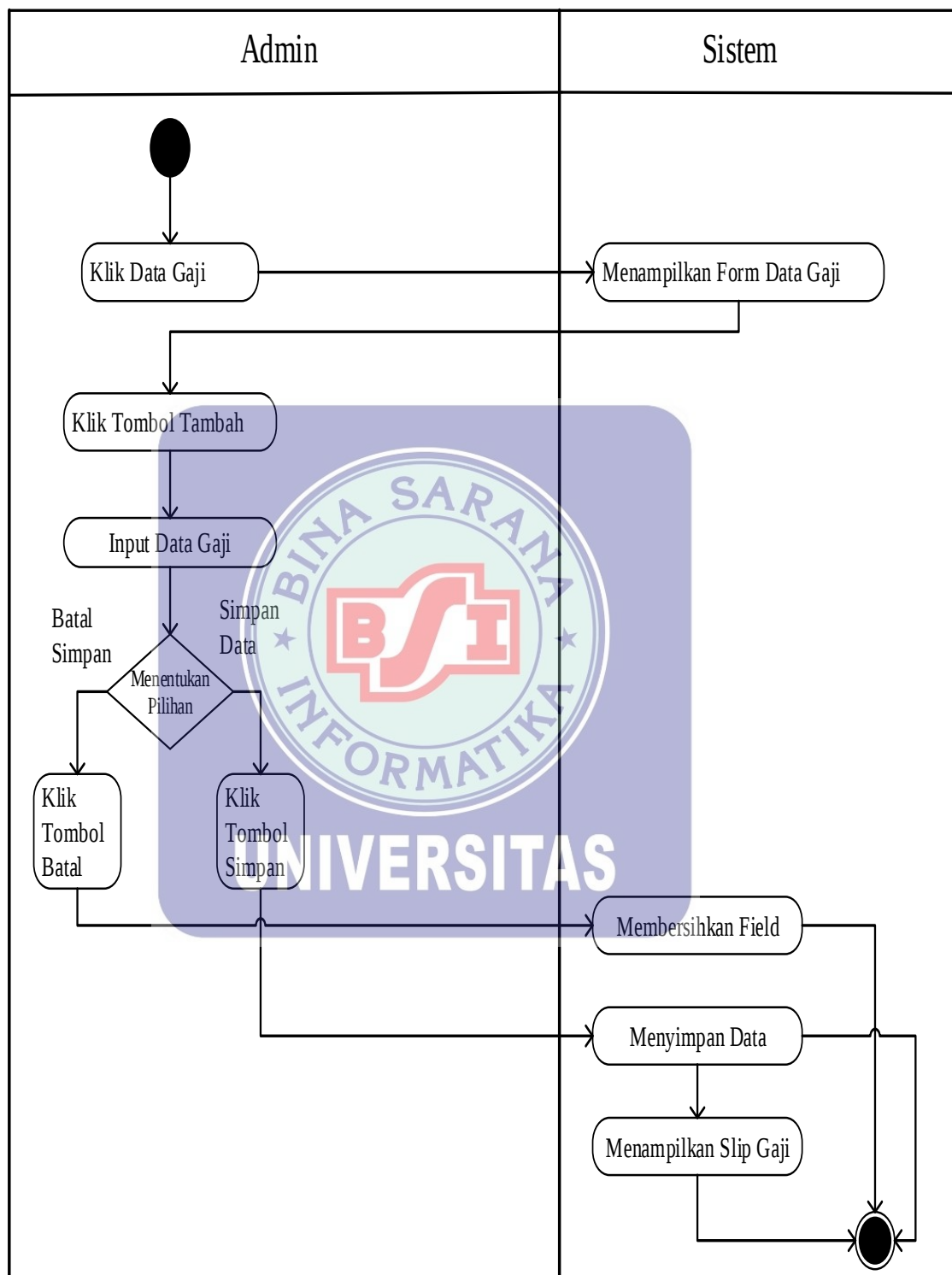
Gambar III.17
Activity Diagram Data Jabatan

4. Activity Diagram Data Akun



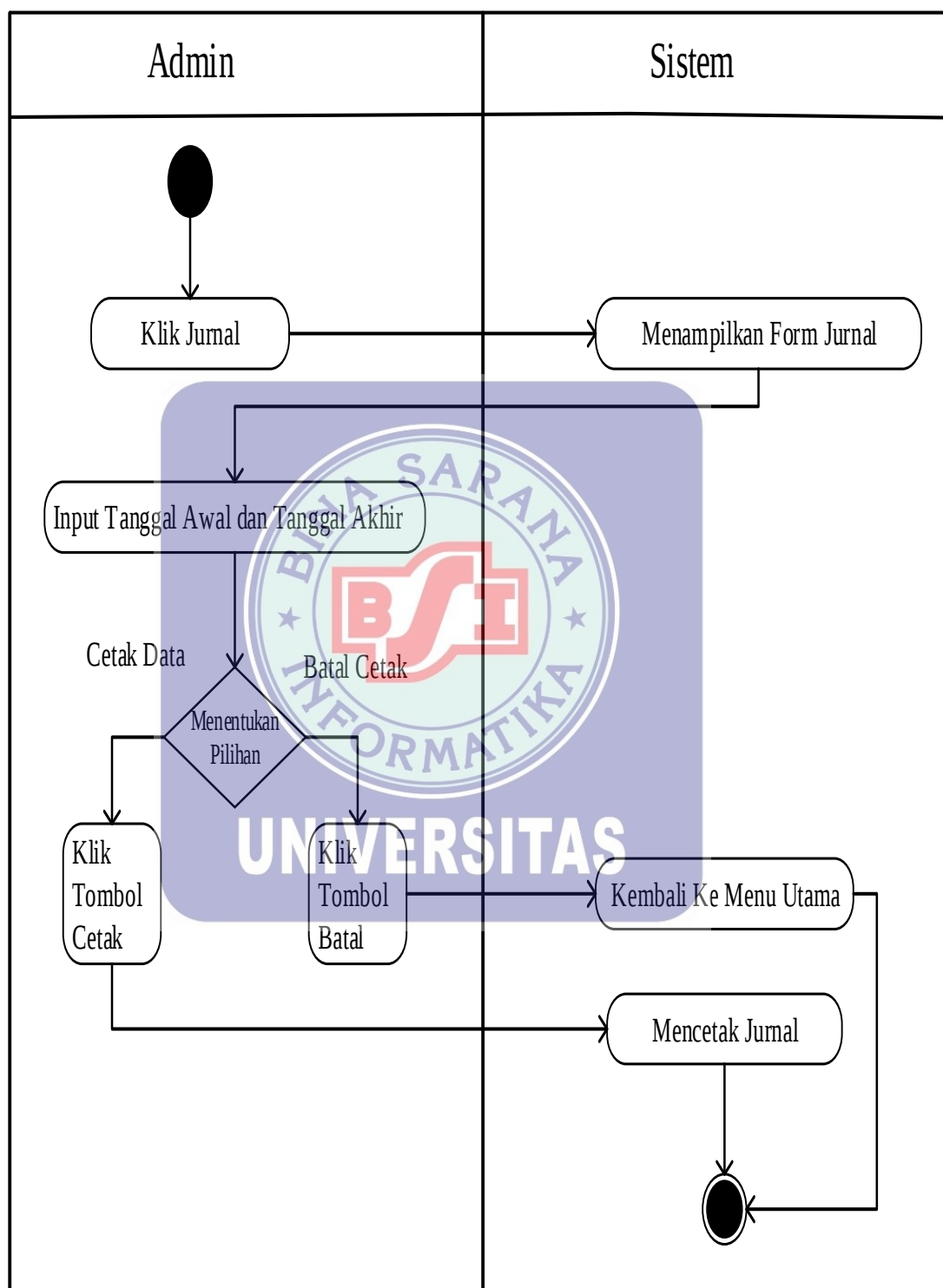
Gambar III.18
Activity Diagram Data Akun

5. Activity Diagram Data Gaji



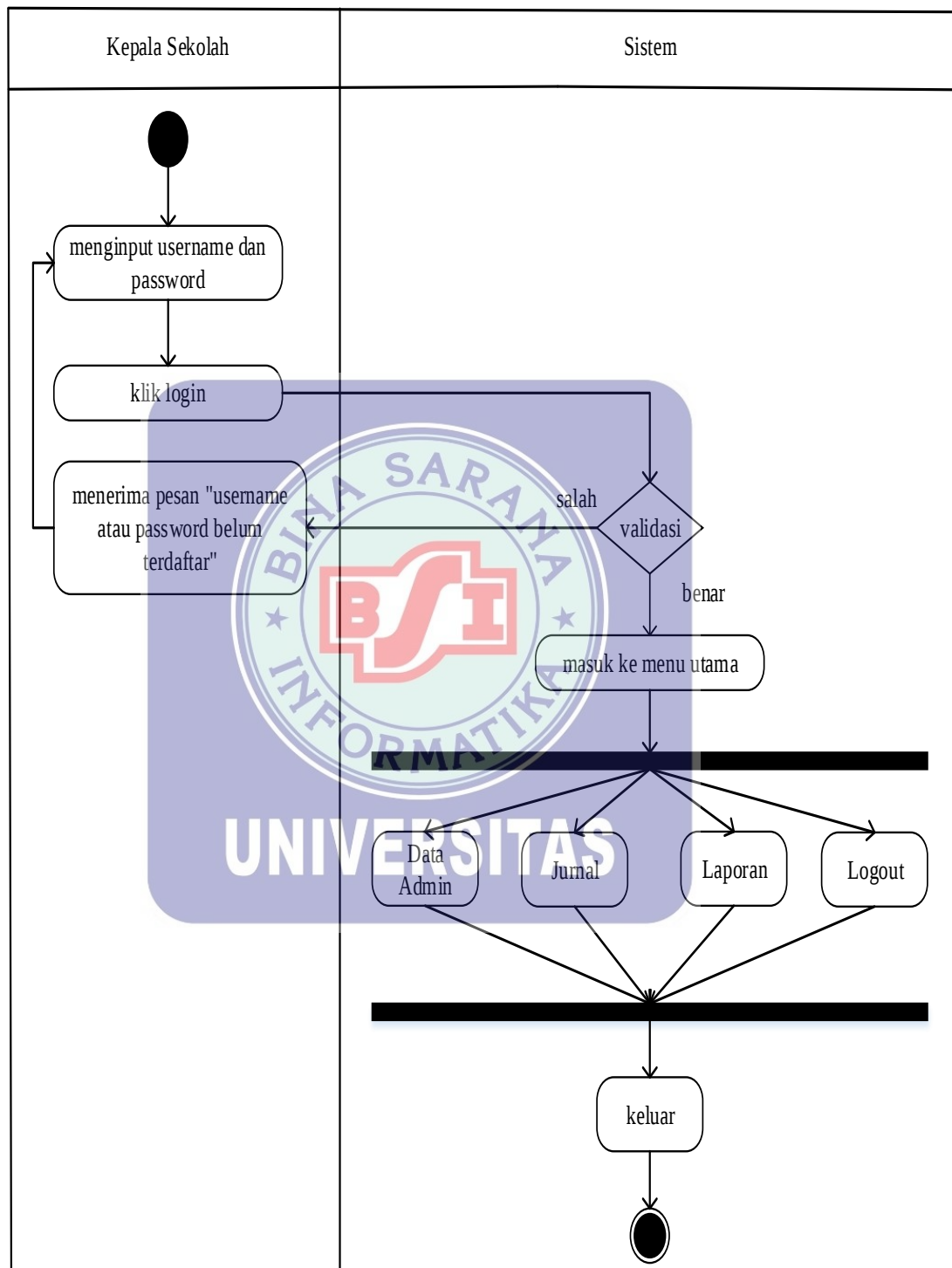
Gambar III.19
Activity Diagram Data Gaji

6. Activity Diagram Jurnal Admin



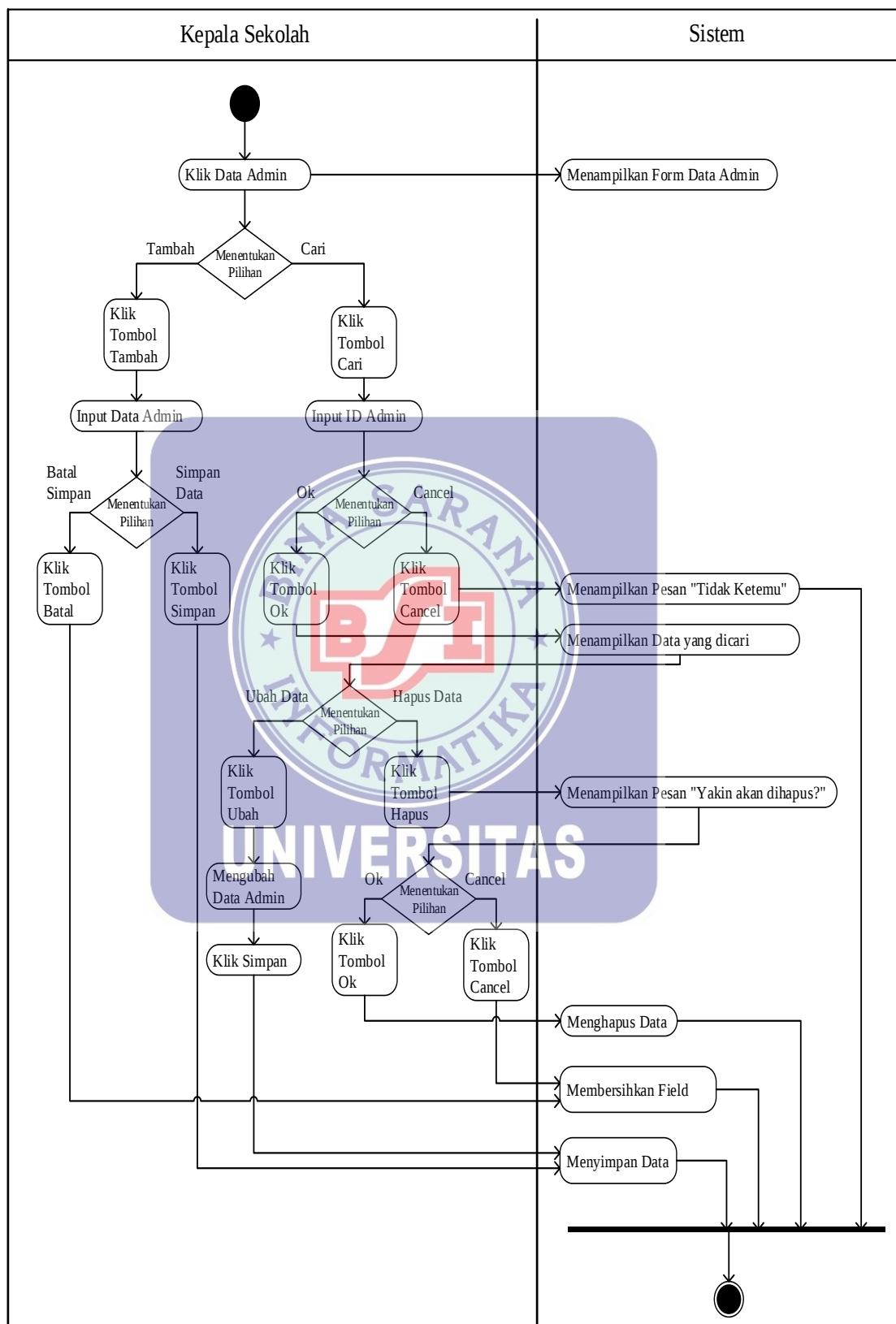
Gambar III.20
Activity Diagram Jurnal Admin

8. Activity Diagram Login Kepala Sekolah



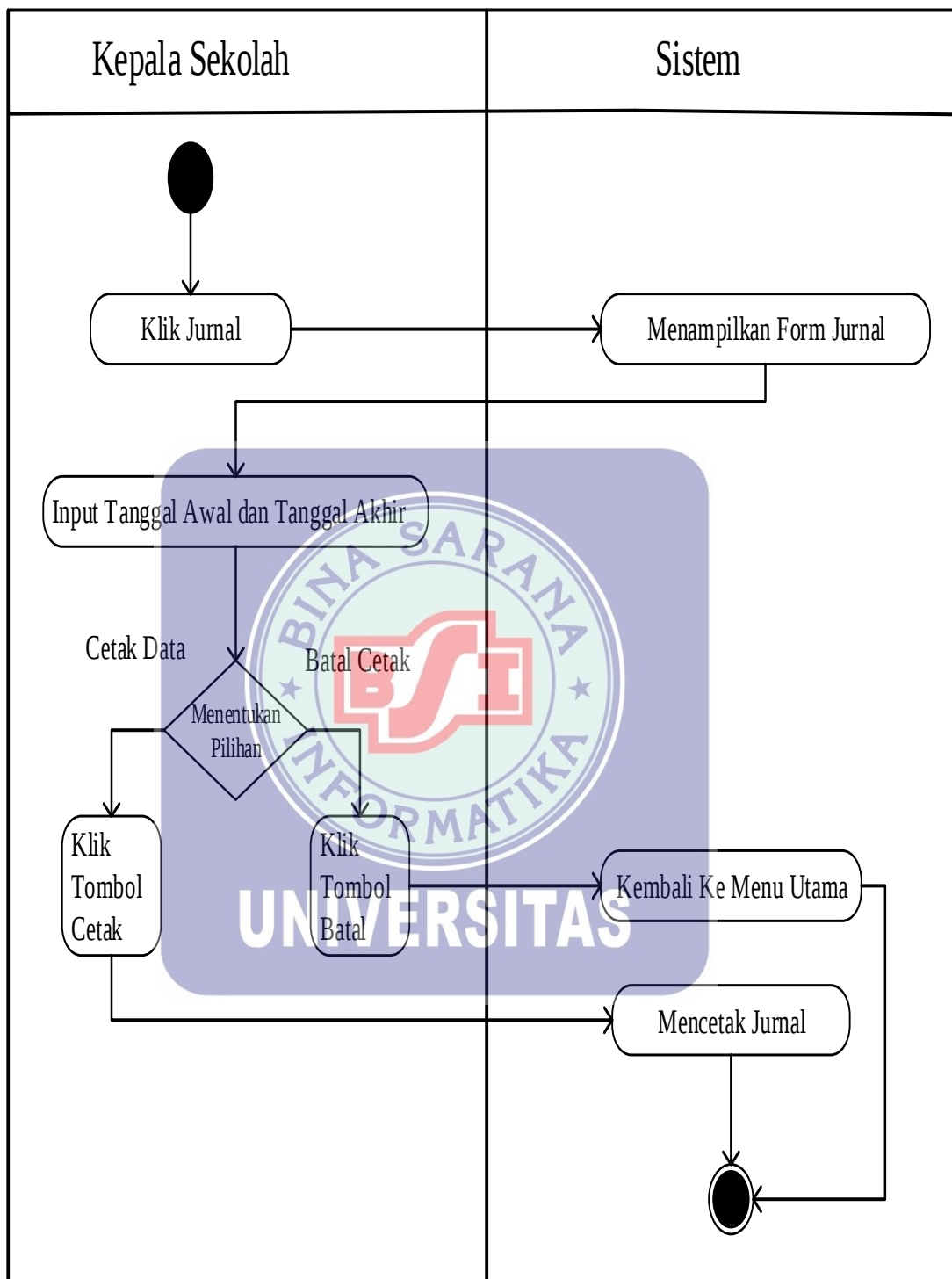
Gambar III.22
Activity Diagram Login Kepala Sekolah

9. Activity Diagram Data Admin



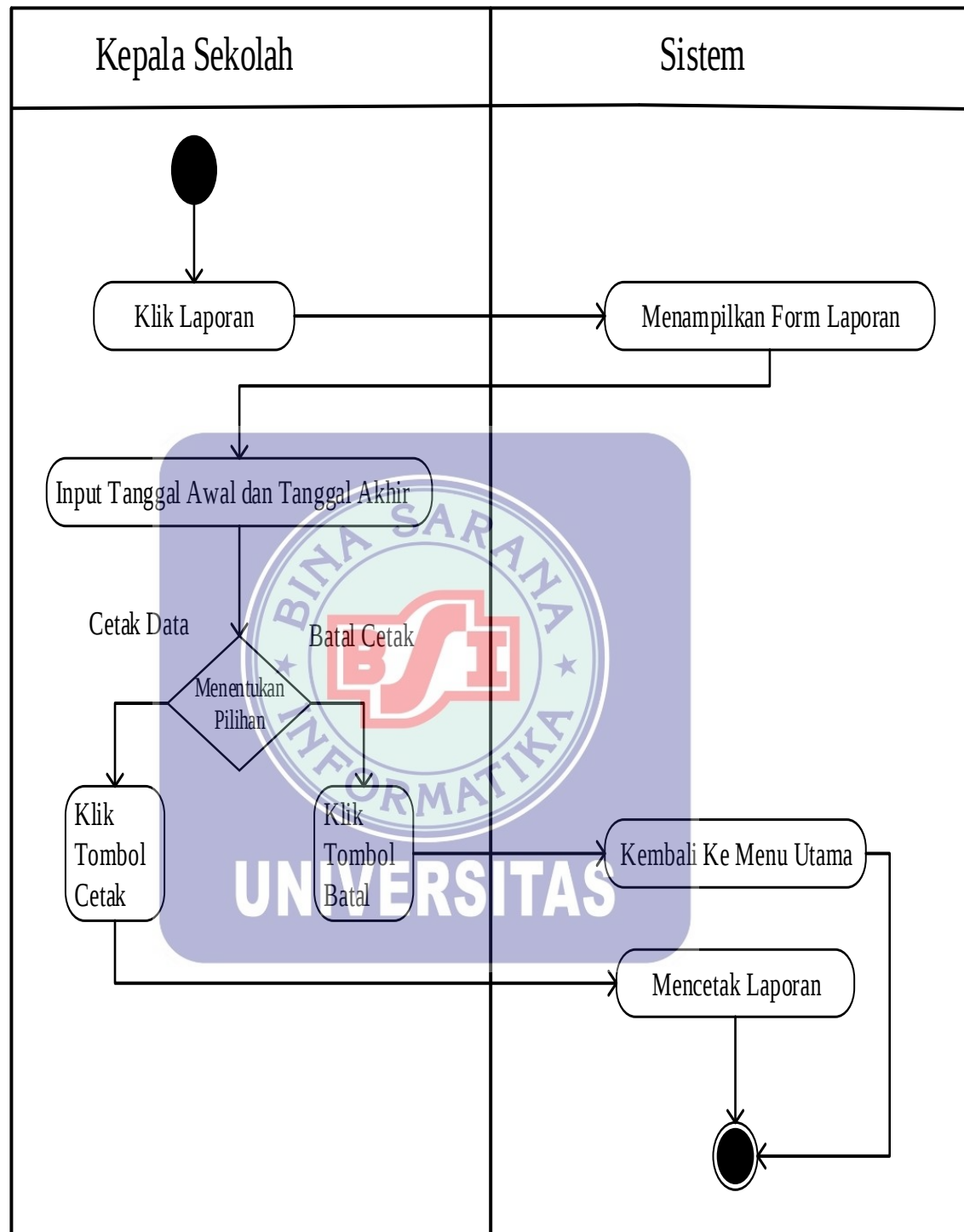
Gambar III.23
Activity Diagram DataAdmin

10. Activity Diagram Jurnal Kepala Sekolah



Gambar III.24
Activity Diagram Jurnal Kepala Sekolah

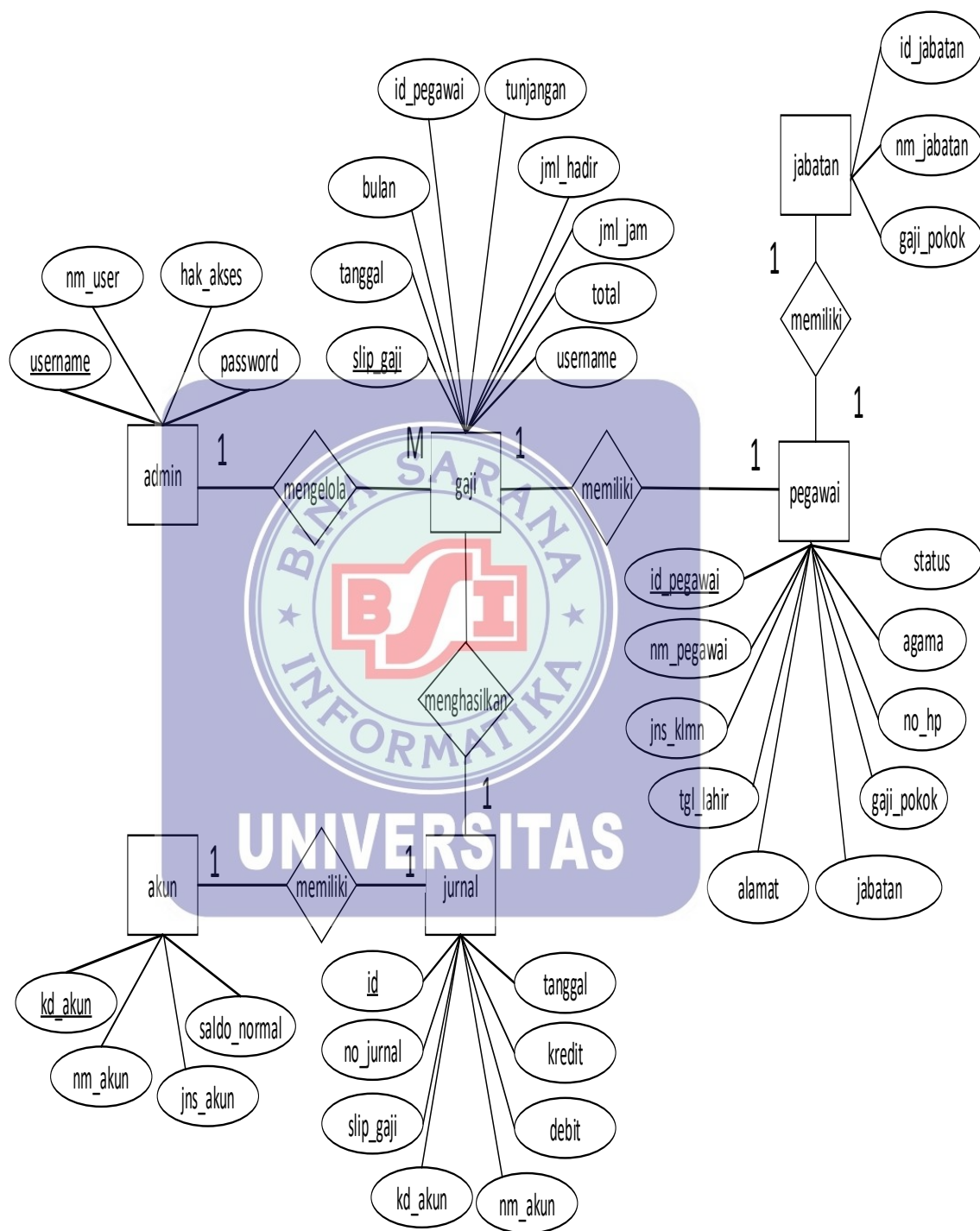
11. Activity Diagram Laporan Kepala Sekolah



Gambar III.25
Activity Diagram Laporan Kepala Sekolah

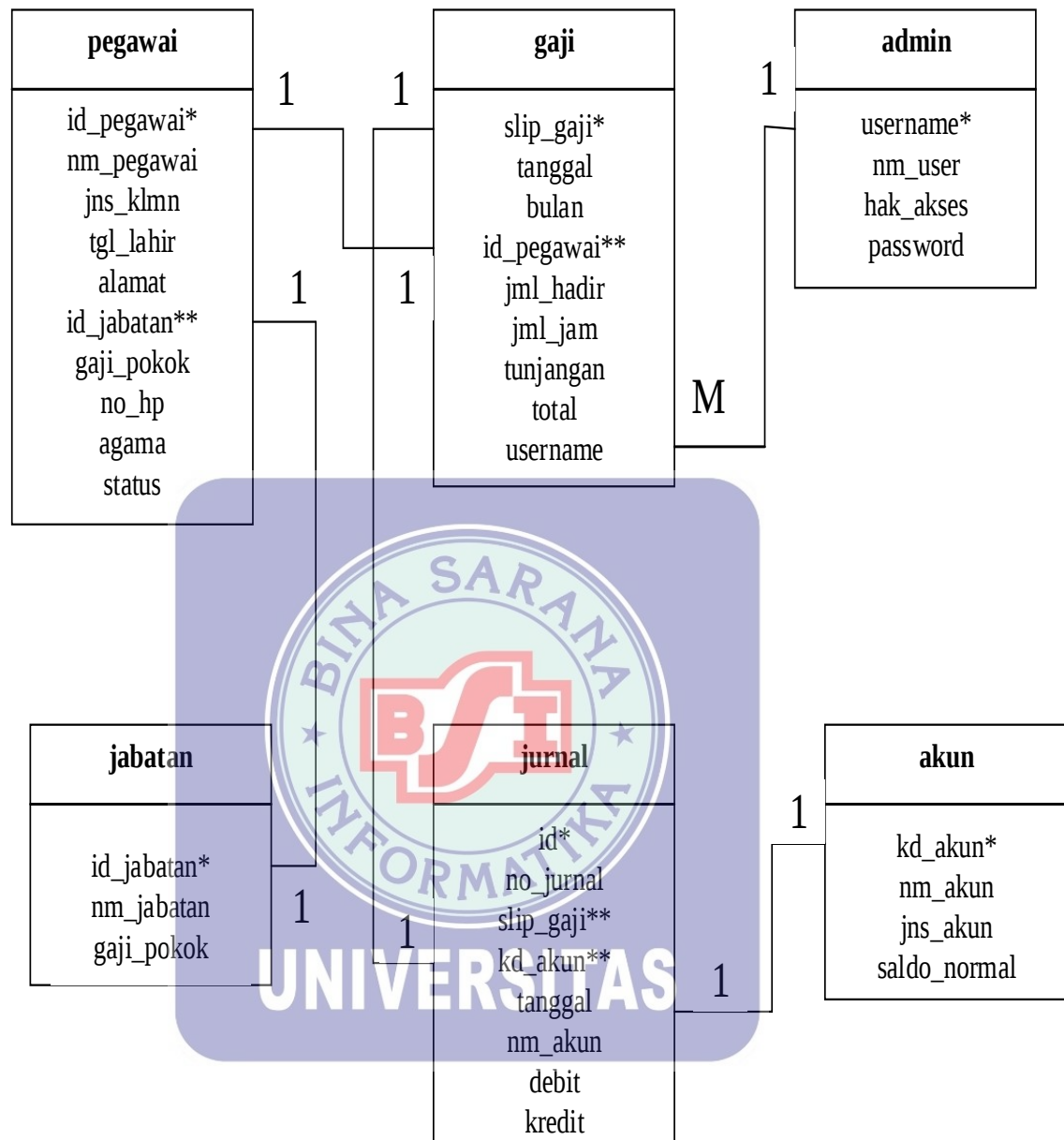
3.4. Desain

3.4.1. Entity Relationship Diagram (ERD)



Gambar III.26
Entity Relationship Diagram (ERD)

3.4.2. Logical Record Structure (LRS)



Gambar III.27
Logical Record Structure (LRS)

3.4.3. Spesifikasi File

a. Spesifikasi File Data Admin

Nama Database : penggajian(1)

Nama File : Tabel Admin

Akronim : admin

Tipe File : Master

Akses File : Random

Panjang Record : 100 Karakter

Field Key : username

Tabel III.12
Spesifikasi FileData Admin

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1	identitas user	username	Varchar	25	primary key
2	nama user	nm_user	Varchar	30	
3	Password	password	Varchar	30	
4	hak akses	hak_akses	Varchar	15	

b. Spesifikasi File Data Pegawai

Nama Database : penggajian(1)

Nama File : Tabel Pegawai

Akronim : pegawai

Tipe File : Master

Akses File : Random

Panjang Record : 190 Karakter

Field Key : id_pegawai

Tabel III.13
Spesifikasi File Data Pegawai

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1	identitas pegawai	id_pegawai	Varchar	15	primary key
2	nama pegawai	nm_pegawai	Varchar	30	
3	jenis kelamin	jns_klmn	Varchar	10	
4	tanggal lahir	tgl_lahir	Date		
5	alamat pegawai	Alamat	Varchar	50	
6	jabatan pegawai	id_jabatan	Varchar	20	foreign key
7	nomor hp	no_hp	Bigint	20	
8	Agama	Agama	Varchar	20	
9	Status	Status	Varchar	25	

c. Spesifikasi *File Data Jabatan*

Nama *Database* : penggajian(1)

Nama *File* : Tabel Jabatan

Akronim : jabatan

Tipe *File* : Master

Akses *File* : *Random*

Panjang *Record* : 40 Karakter

Field Key : id_jabatan

Tabel III.14
Spesifikasi *File Data Jabatan*

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1	identitas jabatan	id_jabatan	Varchar	5	<i>primary key</i>
2	nama jabatan	nm_jabatan	Varchar	25	
3	gaji pokok	gaji_pokok	Int	10	

d. Spesifikasi *File Data Gaji*

Nama *Database* : penggajian(1)

Nama *File* : Tabel Gaji

Akronim : gaji

Tipe *File* : Transaksi

Akses *File* : *Random*

Panjang *Record* : 80 Karakter

Field Key : id_gaji

Tabel III.15
Spesifikasi *File Data Gaji*

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1	identitas gaji	slip_gaji	Varchar	15	<i>primary key</i>
2	tanggal cetak slip gaji	tanggal	Date		
3	bulan cetak gaji	bulan	Varchar	15	

4	identitas pegawai	id_pegawai	Varchar	5	<i>foreign key</i>
5	jumlah hadir	jml_hadir	Int	5	
6	jumlah jam	jml_jam	Int	5	
7	Tunjangan	tunjangan	Int	10	
8	total gaji	total	Double		
9	identitas user	username	Varchar	25	<i>foreign key</i>

e. Spesifikasi File Data Jurnal

Nama Database : penggajian(1)

Nama File : Tabel Jurnal

Akronim : jurnal

Tipe File : Transaksi

Akses File : Random

Panjang Record : 52 Karakter

Field Key : id_jurnal

Tabel III.16
Spesifikasi File Data Jurnal

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1	identitas akun	id	Int	11	<i>Primary key</i>
2	nomor jurnal	no_jurnal	Varchar	15	
3	identitas gaji	slip_gaji	Varchar	15	<i>Foreign key</i>
4	identitas akun	kd_akun	Varchar	15	
5	Tanggal	tanggal	Date		
6	nama akun	nm_akun	Varchar	15	
7	Debet	debit	Double		
8	Kredit	kredit	Double		

f. Spesifikasi File Data Akun

Nama Database : penggajian(1)

Nama File : Tabel Akun

Akronim : akun

Tipe File : Master

Akses File : *Random*

Panjang Record : 50 Karakter

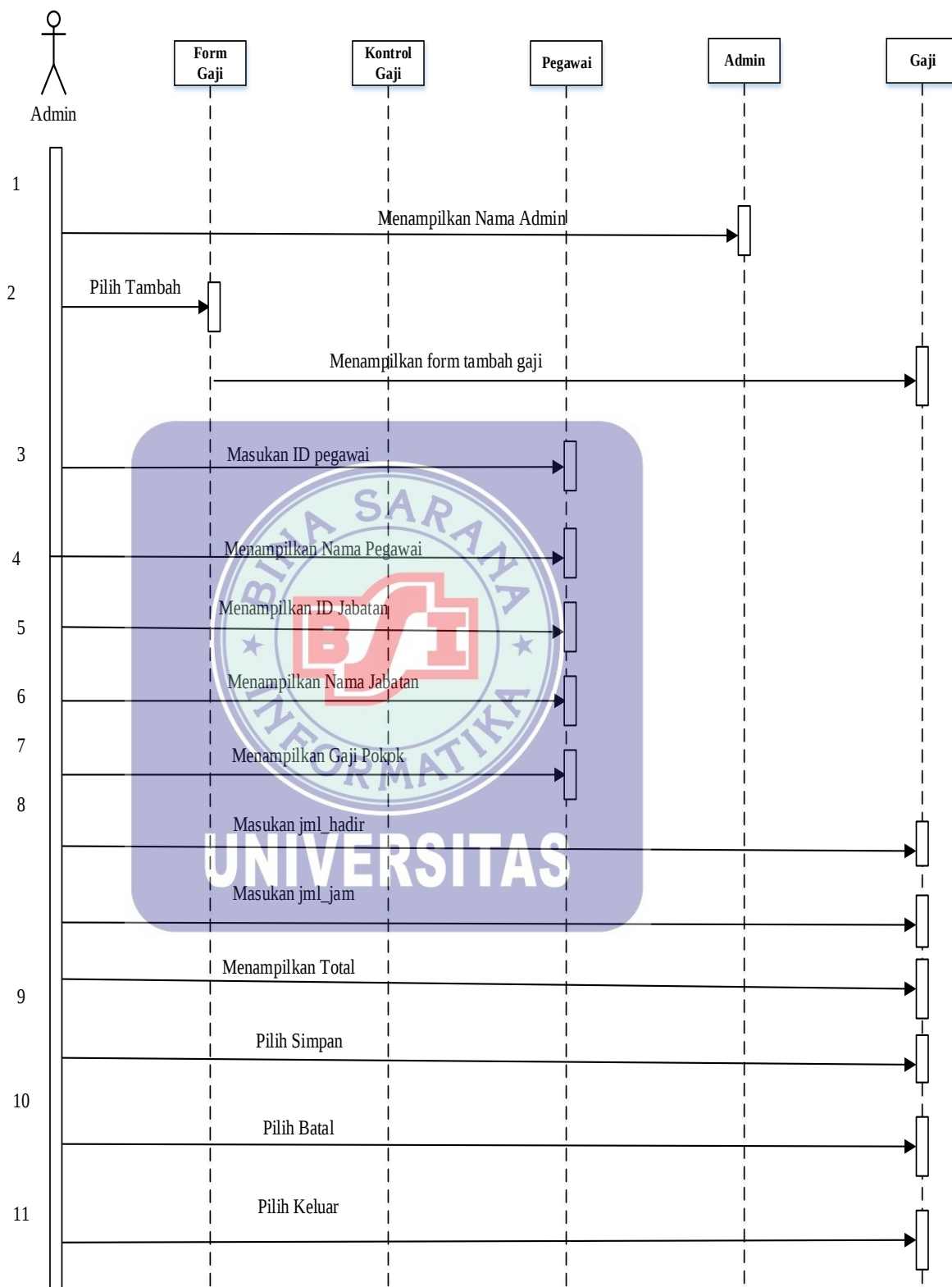
Field Key : *kd_akun*

Tabel III.17
Spesifikasi File Data Akun

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1	identitas akun	kd_akun	Int	5	<i>Primary key</i>
2	nama akun	nm_akun	Varchar	15	
3	jenis akun	jns_akun	Varchar	15	
4	saldo normal	saldo_normal	Varchar	15	<i>Foreign key</i>

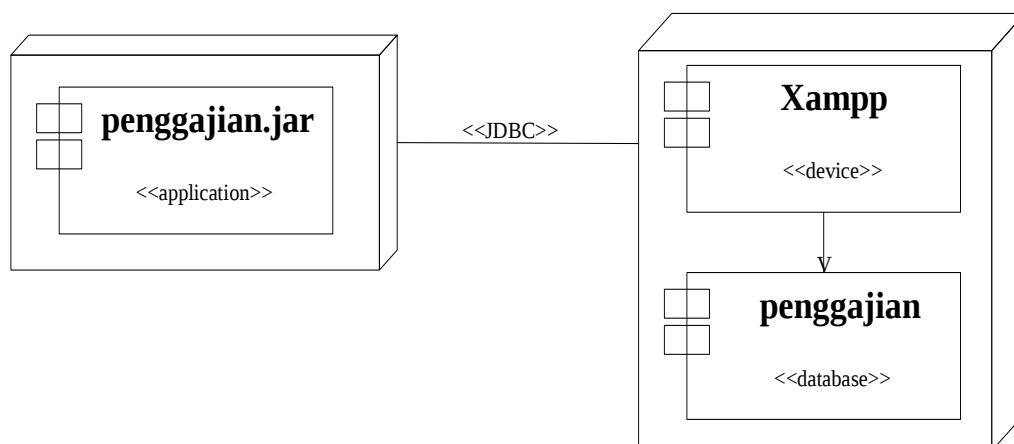


3.4.4. Sequence Diagram



Gambar III.28
Sequence Diagram Sistem Pembayaran Gaji

3.4.5. Deployment Diagram



Gambar III.29
Deployment Diagram Sistem Pembayaran Gaji

3.4.6. User Interface

The screenshot shows a login form window. At the top, there are two logos: a circular logo for 'BINA SARANA INFORMATIKA' with a red 'BSI' acronym, and a green pentagonal logo for 'UNIVERSITAS IKHLAS BERAMAL' featuring a book and a star. Below the logos, the form has two input fields: 'Username' and 'Password'. To the right of the 'Username' field is a 'Login' button. To the right of the 'Password' field is a 'Keluar' button. The window has standard Windows-style title bar controls (minimize, maximize, close) in the top right corner.

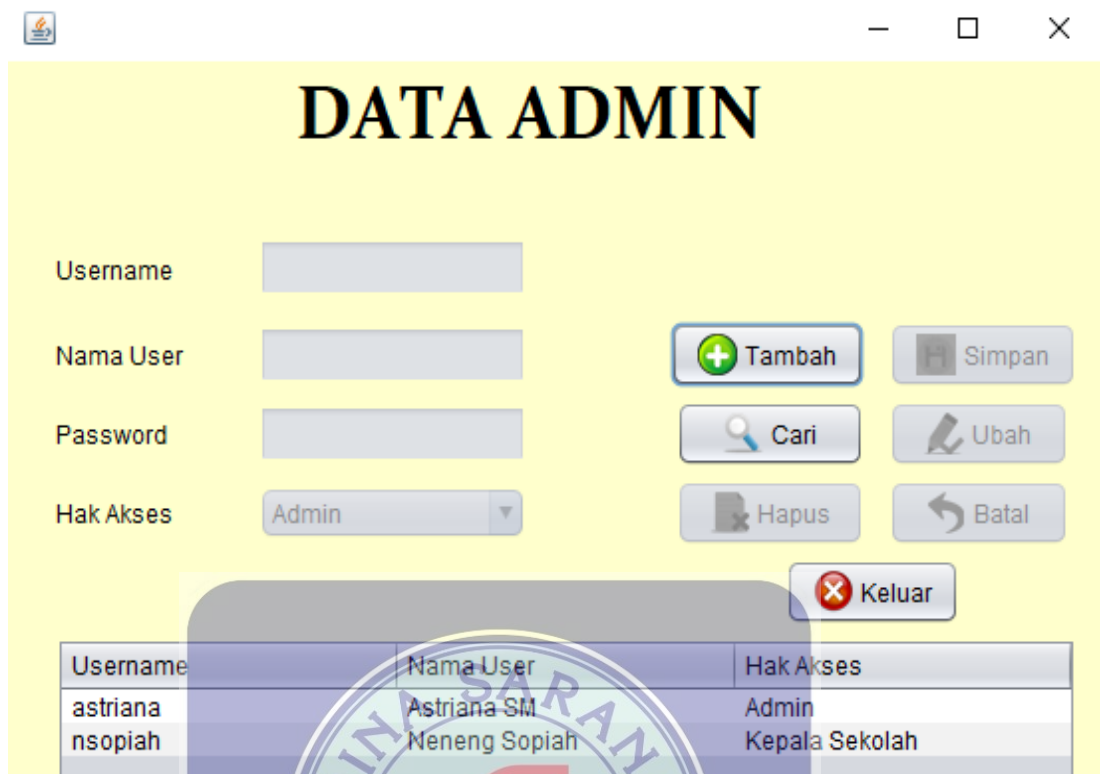
Gambar III.30
User Interface Form Login



Gambar III.31
User Interface Menu Utama Admin

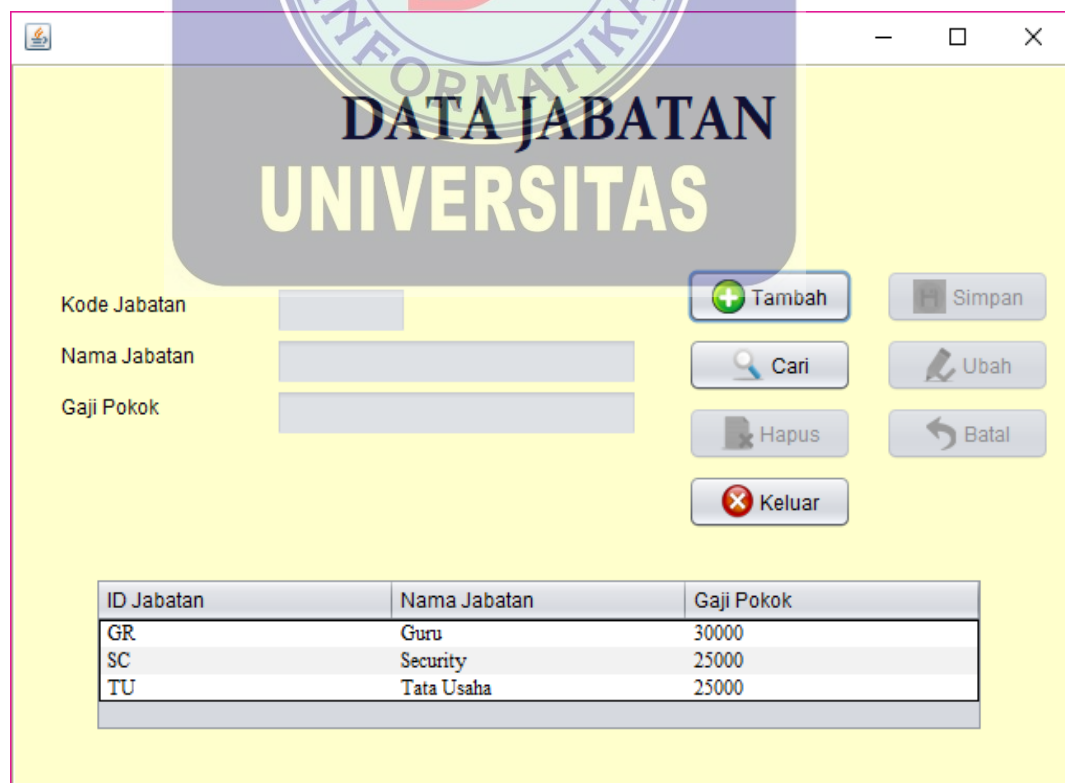


Gambar III.32
User Interface Menu Utama Kepala Sekolah



Username	Nama User	Hak Akses
astriana	Astriana SM	Admin
nsopiah	Neneng Sopiah	Kepala Sekolah

Gambar III.33
User Interface Data Admin



ID Jabatan	Nama Jabatan	Gaji Pokok
GR	Guru	30000
SC	Security	25000
TU	Tata Usaha	25000

Gambar III.34
User Interface Data Jabatan

DATA PEGAWAI

ID Pegawai

Nama Pegawai

Jenis Kelamin

Tanggal Lahir

Alamat

ID Jabatan

Nama Jabatan

Gaji Pokok

No Handphone

Agama

Status

+ Tambah Simpan

Cari Ubah

Hapus Batal

Keluar

ID Pegawai	Nama Pegawai	Jenis Kelamin	ID Jabatan	Nama Jabatan	Gaji Pokok
G01	Siti Ramidah	Perempuan	GR	Guru	30000
SC01	Ujang Mulyana	Laki-laki	SC	Security	25000
TU01	Meti Setia Sari	Perempuan	TU	Tata Usaha	25000
TU02	Udin	Laki-laki	TU	Tata Usaha	25000

Gambar III.35
User Interface Data Pegawai

DATA AKUN UNIVERSITAS

Kode Akun

Nama Akun

Jenis Akun

Saldo Normal

+ Tambah Simpan

Cari Ubah

Hapus Batal

Keluar

Kode Akun	Nama Akun	Jenis Akun	Saldo Normal
101	Kas	Kas	Debit
511	Biaya Gaji	Beban	Kredit

Gambar III.36
User Interface Data Akun

DATA GAJI

Admin :

ID Slip Gaji :

Tanggal Cetak :

Bulan :

ID Pegawai :

Nama Pegawai :

Jabatan :

Gaji Pokok :

Jumlah Hadir :

Jumlah Jam :

Tunjangan :

Total Gaji :

Tanggal	No Jurnal	ID Slip Gaji	Kode Akun	Nama Akun	Debit	Kredit
2019-09-08	J0809190...	G080919...	511	Biaya Gaji	1540000	0
2019-09-08	J0809190...	G080919...	101	Kas	0	1540000

ID Slip Gaji	Tanggal	Bulan	ID Pegawai	Hadir	Jam	Tunjangan	Total	Admin
G080919001	2019-09-08	September	G01	0	48	100000	1540000	Astria SM

Gambar III.37
User Interface Data Gaji

UNIVERSITAS JURNAL

Tanggal Awal :

Tanggal Akhir :

Tanggal	No Jurnal	Slip Gaji	Kode Akun	Nama Akun	Debit	Kredit
2019-08-31	J3108190...	G3108190...	511	Biaya Gaji	770000	0
2019-08-31	J3108190...	G3108190...	101	Kas	0	770000

Gambar III.38
User Interface Jurnal

LAPORAN

Tanggal Awal :

Tanggal Akhir :

Cetak Batal

Tanggal	Bulan	Slip Gaji	ID Pegawai	Jumlah H...	Jumlah J...	Total	Nama Ad...
G310819...	2019-08-31	August	G01	0	25	770000	Astiana SM

Gambar III.39
User Interface Laporan

3.5. Implementasi

3.5.1. Code Generation

```
package gaji;

import java.io.File;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.HashMap;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import javax.swing.table.DefaultTableModel;
import net.sf.jasperreports.engine.JRException;
import net.sf.jasperreports.engine.JasperCompileManager;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
```

```

import net.sf.jasperreports.engine.JasperReport;
import net.sf.jasperreports.engine.design.JasperDesign;
import net.sf.jasperreports.engine.xml.JRXmlLoader;
import net.sf.jasperreports.view.JasperViewer;

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
/**
 *
 * @author asus
 */
public class fgaji extends javax.swing.JFrame {

    koneksi kon = new koneksi();

    /**
     * Creates new form fgaji
     */
    public fgaji() {
        initComponents();
        txtadmin.setText(SesiAdmin.getU_nama());
        txtidpeg.setText("");
        txtbulan.setText("");
        txtnmpeg.setText("");
        txtidjab.setText("");
        txtgapok.setText("");
        txthadir.setText("");
        txtjam.setText("");

        kon.getConnection();
        jSimpan.setEnabled(false);
        jTambah.setEnabled(true);
        jBatal.setEnabled(false);
        jKeluar.setEnabled(true);

        kon.getConnection();
        regrid();
        regrid2();
        //nonaktif();
        txtjab.setVisible(false);

        DateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd");
        Calendar cal = Calendar.getInstance();
        DateFormat mm = new SimpleDateFormat("MMMM");
        Calendar call = Calendar.getInstance();
        txttgl.setText(dateFormat.format(cal.getTime()));
    }
}

```

```

txtbulan.setText(mm.format(call.getTime()));

}
private void aktif() {
    txtidpeg.setEnabled(true);
    txtnmpeg.setEnabled(true);
    txtidjab.setEnabled(true);
    txtgapok.setEnabled(true);
    //txthadir.setEnabled(true);
    //txtjam.setEnabled(true);
}
private void nonaktif() {
    txtidpeg.setEnabled(false);
    txtnmpeg.setEnabled(false);
    txtidjab.setEnabled(false);
    txtgapok.setEnabled(false);
    //txthadir.setEnabled(false);
    //txtjam.setEnabled(false);
}
private void regrid() {
    DefaultTableModel tgaji = new DefaultTableModel();
    tgaji.addColumn("ID Slip Gaji");
    tgaji.addColumn("Tanggal");
    tgaji.addColumn("Bulan");
    tgaji.addColumn("ID Pegawai");
    tgaji.addColumn("Hadir");
    tgaji.addColumn("Jam");
    tgaji.addColumn("Tunjangan");
    tgaji.addColumn("Total");
    tgaji.addColumn("Admin");

    try {
        kon.rs = kon.st.executeQuery("select * from gaji");
        while (kon.rs.next()) {
            tgaji.addRow(new Object[]{
                kon.rs.getString(1),
                kon.rs.getString(2),
                kon.rs.getString(3),
                kon.rs.getString(4),
                kon.rs.getString(5),
                kon.rs.getString(6),
                kon.rs.getString(7),
                kon.rs.getString(8),
                kon.rs.getString(9)});
        }
    } catch (Exception e) {
        JOptionPane.showMessageDialog(this, e);
    }
    jTable2.setModel(tgaji);
}

```

```

    }
    private void regrid2() {
        DefaultTableModel tslip = new DefaultTableModel();
        tslip.addColumn("Tanggal");
        tslip.addColumn("No Jurnal");
        tslip.addColumn("ID Slip Gaji");
        tslip.addColumn("Kode Akun");
        tslip.addColumn("Nama Akun");
        tslip.addColumn("Debet");
        tslip.addColumn("Kredit");

        try {
            kon.rs = kon.st.executeQuery("select * from jurnal");
            while (kon.rs.next()) {
                tslip.addRow(new Object[]{
                    kon.rs.getString(5),
                    kon.rs.getString(2),
                    kon.rs.getString(3),
                    kon.rs.getString(4),
                    kon.rs.getString(6),
                    kon.rs.getString(7),
                    kon.rs.getString(8),});
            }
        } catch (Exception e) {
            JOptionPane.showMessageDialog(this, e);
        }
        {
            jTable1.setModel(tslip);
        }
    }
    private void bersih() {
        txtidgaji.setText("");
        txtidpeg.setText("");
        txtnmpeg.setText("");
        txtidjab.setText("");
        txtgapok.setText("");
        txthadir.setText("");
        txtjam.setText("");
        txtttl.setText("");
    }
    public String norut() {
        String urutan = null;
        java.util.Date skrg = new java.util.Date();
        java.text.SimpleDateFormat kal = new
        java.text.SimpleDateFormat("ddMMyy");
        String tgl = (kal.format(skrg));
        try {
            Connection con = kon.getConnection();
            Statement st = con.createStatement();

```

```

        java.sql.ResultSet sql = st.executeQuery("SELECT (right (slip_gaji,3)+1) as
nomor FROM gaji where slip_gaji like '%" + tgl + "%' order by slip_gaji desc");
        if (sql.next()) {
            urutan = sql.getString(1);
            while (urutan.length() < 3) {
                urutan = "0" + urutan;
            }
            urutan = "G" + tgl + urutan;
        } else {
            urutan = "G" + tgl + "001";
        }
        txtidgaji.setText(urutan);
    } catch (Exception e) {
        // JOptionPane.showMessageDialog(null,e);
    }
    return urutan;
}

public String nojur() {
    String urutan = null;
    java.util.Date skrg = new java.util.Date();
    java.text.SimpleDateFormat kal = new
java.text.SimpleDateFormat("ddMMyy");
    String tgl = (kal.format(skrg));
    try {
        Connection con = kon.getConnection();
        Statement st = con.createStatement();
        java.sql.ResultSet sql = st.executeQuery("SELECT (right (no_jurnal,3)+1) as
nomor FROM jurnal where no_jurnal like '%" + tgl + "%' order by no_jurnal desc");
        if (sql.next()) {
            urutan = sql.getString(1);
            while (urutan.length() < 3) {
                urutan = "0" + urutan;
            }
            urutan = "J" + tgl + urutan;
        } else {
            urutan = "J" + tgl + "001";
        }
        txtnojurnal.setText(urutan);
    } catch (Exception e) {
        // JOptionPane.showMessageDialog(null,e);
    }
    return urutan;
}

public String idpeg;
public String namapeg;

```

```

public String idjab;

public String getidpeg() {
    return idpeg;
}
public String getnmpeg() {
    return namapeg;
}
public String getidjab() {
    return idjab;
}
private void GetDataJab() {
    try {
        Statement stat = (Statement) kon.getConnection().createStatement();
        String sql = "select * from jabatan where id_jabatan='" + txtjab.getText() +
        """,
        ResultSet res = stat.executeQuery(sql);

        while (res.next()) {
            txtidjab.setText(res.getString("nm_jabatan"));
            txtgapok.setText(res.getString("gaji_pokok"));
        }
    } catch (SQLException err) {
        JOptionPane.showMessageDialog(null, err.getMessage());
    }
}
/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jLabel6 = new javax.swing.JLabel();
    jLabel7 = new javax.swing.JLabel();
    txtidgaji = new javax.swing.JTextField();
    txtidpeg = new javax.swing.JTextField();
    txtbulan = new javax.swing.JTextField();
    txtnmpeg = new javax.swing.JTextField();
    jLabel9 = new javax.swing.JLabel();
    jLabel13 = new javax.swing.JLabel();

```

```

jLabel16 = new javax.swing.JLabel();
jLabel17 = new javax.swing.JLabel();
txtidjab = new javax.swing.JTextField();
txtgapok = new javax.swing.JTextField();
txthadir = new javax.swing.JTextField();
txtjam = new javax.swing.JTextField();
jSimpan = new javax.swing.JButton();
jTambah = new javax.swing.JButton();
jBatal = new javax.swing.JButton();
jScrollPane1 = new javax.swing.JScrollPane();
jTable1 = new javax.swing.JTable();
txttl = new javax.swing.JTextField();
jKeluar = new javax.swing.JButton();
txttl = new javax.swing.JTextField();
btncaripegawai = new javax.swing.JButton();
txtadmin = new javax.swing.JTextField();
jLabel8 = new javax.swing.JLabel();
jScrollPane2 = new javax.swing.JScrollPane();
jTable2 = new javax.swing.JTable();
txtnojurnal = new javax.swing.JTextField();
txtjab = new javax.swing.JTextField();
jLabel18 = new javax.swing.JLabel();
txttunjangan = new javax.swing.JTextField();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

jPanel1.setBackground(new java.awt.Color(255, 255, 204));
jPanel1.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

jLabel1.setFont(new java.awt.Font("Constantia", 1, 36)); // NOI18N
jLabel1.setText("DATA GAJI");
jPanel1.add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(312,
11, 230, -1));

jLabel2.setText("ID Slip Gaji");
jPanel1.add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
140, -1, -1));

jLabel3.setText("Tanggal Cetak");
jPanel1.add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
163, -1, 20));

jLabel4.setText("ID Pegawai");
jPanel1.add(jLabel4, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
224, -1, -1));

jLabel5.setText("Bulan");

```

```

jPanel1.add(jLabel5, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
195, -1, -1));

jLabel6.setText("Nama Pegawai");
jPanel1.add(jLabel6, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
250, -1, -1));

jLabel7.setText("Jabatan");
jPanel1.add(jLabel7, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
273, -1, -1));

txtidgaji.setEditable(false);
txtidgaji.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtidgajiActionPerformed(evt);
    }
});
jPanel1.add(txtidgaji, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
137, 186, -1));

txtidpeg.setEditable(false);
txtidpeg.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtidpegActionPerformed(evt);
    }
});
jPanel1.add(txtidpeg, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
216, 186, -1));

txtbulan.setEditable(false);
txtbulan.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtbulanActionPerformed(evt);
    }
});
jPanel1.add(txtbulan, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
189, 186, -1));

txtnmpeg.setEditable(false);
txtnmpeg.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtnmpegActionPerformed(evt);
    }
});
jPanel1.add(txtnmpeg, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
244, 186, -1));

jLabel9.setText("Gaji Pokok");

```

```

jPanel1.add(jLabel9, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
299, -1, -1));

jLabel13.setText("Jumlah Hadir");
jPanel1.add(jLabel13, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
325, -1, -1));

jLabel16.setText("Jumlah Jam");
jPanel1.add(jLabel16, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
351, -1, -1));

jLabel17.setText("Total Gaji");
jPanel1.add(jLabel17, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
420, -1, 20));

txtidjab.setEditable(false);
txtidjab.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtidjabActionPerformed(evt);
    }
});
jPanel1.add(txtidjab, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
270, 186, -1));

txtgapok.setEditable(false);
jPanel1.add(txtgapok, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
296, 186, -1));

txthadir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txthadirActionPerformed(evt);
    }
});
txthadir.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyReleased(java.awt.event.KeyEvent evt) {
        txthadirKeyReleased(evt);
    }
});
jPanel1.add(txthadir, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
322, 186, -1));

txtjam.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyReleased(java.awt.event.KeyEvent evt) {
        txtjamKeyReleased(evt);
    }
});
jPanel1.add(txtjam, new org.netbeans.lib.awtextra.AbsoluteConstraints(120,
350, 186, -1));

```

```

        jSimpan.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/icon/tombol_simpan.jpeg"))); //
NOI18N
        jSimpan.setText("Simpan");
        jSimpan.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jSimpanActionPerformed(evt);
            }
        });
        jPanel1.add(jSimpan, new org.netbeans.lib.awtextra.AbsoluteConstraints(450,
370, 95, -1));

        jTambah.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/icon/tombol_tambah.png"))); //
NOI18N
        jTambah.setText("Tambah");
        jTambah.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jTambahActionPerformed(evt);
            }
        });
        jPanel1.add(jTambah, new org.netbeans.lib.awtextra.AbsoluteConstraints(330,
370, -1, -1));

        jBatal.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/icon/tombol_batal.png"))); //
NOI18N
        jBatal.setText("Batal");
        jBatal.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jBatalActionPerformed(evt);
            }
        });
        jPanel1.add(jBatal, new org.netbeans.lib.awtextra.AbsoluteConstraints(570,
370, 95, -1));

        jTable1.setModel(new javax.swing.table.DefaultTableModel(
            new Object [][] {
                {null, null, null, null, null, null, null},
                {null, null, null, null, null, null, null},
                {null, null, null, null, null, null, null},
                {null, null, null, null, null, null, null}
            },
            new String [] {
                "Tanggal", "No Jurnal", "No Transaksi", "Kode Akun", "Nama Akun",
                "Debet", "Kredit"
            }
        ));
        jScrollPane1.setViewportView(jTable1);

```

```

jPanel1.add(jScrollPane1,
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 450, 530, 120));

txtttl.setEditable(false);
txtttl.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtttlActionPerformed(evt);
    }
});
jPanel1.add(txtttl, new org.netbeans.lib.awtextra.AbsoluteConstraints(120, 410,
186, 20));

jKeluar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/icon/tombol_keluar.png"))); //
NOI18N
jKeluar.setText("Keluar");
jKeluar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        jKeluarActionPerformed(evt);
    }
});
jPanel1.add(jKeluar, new org.netbeans.lib.awtextra.AbsoluteConstraints(700,
370, 95, -1));

txttgl.setEditable(false);
jPanel1.add(txttgl, new org.netbeans.lib.awtextra.AbsoluteConstraints(120, 163,
186, -1));

btncaripegawai.setText("Pilih");
btncaripegawai.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btncaripegawaiActionPerformed(evt);
    }
});
jPanel1.add(btncaripegawai,
org.netbeans.lib.awtextra.AbsoluteConstraints(312, 215, 60, -1));

txtadmin.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtadminActionPerformed(evt);
    }
});
jPanel1.add(txtadmin, new org.netbeans.lib.awtextra.AbsoluteConstraints(610,
96, 210, -1));

jLabel8.setText("Admin :");
jPanel1.add(jLabel8, new org.netbeans.lib.awtextra.AbsoluteConstraints(556,
96, 50, 20));

```

```

jTable2.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null, null, null},
        {null, null, null, null, null, null, null, null, null, null}
    },
    new String [] {
        "ID Slip Gaji", "Tanggal", "Bulan", "Nama", "Jabatan", "Gaji Pokok",
        "Hadir", "Jam", "Title 9", "Total"
    }
));
jScrollPane2.setViewportViewView(jTable2);

jPanel1.add(jScrollPane2,                                     new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 580, 790, 120));
jPanel1.add(txtnojurnal,                                     new
org.netbeans.lib.awtextra.AbsoluteConstraints(480, 470, 26, -1));

txtjab.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        txtjabActionPerformed(evt);
    }
});
jPanel1.add(txtjab, new org.netbeans.lib.awtextra.AbsoluteConstraints(160,
490, 10, 20));

jLabel18.setText("Tunjangan");
jPanel1.add(jLabel18, new org.netbeans.lib.awtextra.AbsoluteConstraints(20,
380, -1, 20));

txttunjangan.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyReleased(java.awt.event.KeyEvent evt) {
        txttunjanganKeyReleased(evt);
    }
});
jPanel1.add(txttunjangan,                                     new
org.netbeans.lib.awtextra.AbsoluteConstraints(120, 380, 186, -1));

getContentPane().add(jPanel1,                               new
org.netbeans.lib.awtextra.AbsoluteConstraints(0, -1, 830, 730));

setSize(new java.awt.Dimension(846, 751));
setLocationRelativeTo(null);
} // </editor-fold>

private void txtidpegActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

```

```

    }

    private void txtbulanActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void txtnmpegActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void txthadirActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void txtidgajiActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
    }

    private void jSimpanActionPerformed(java.awt.event.ActionEvent evt) {
        String row_txtno = txtidgaji.getText();
        String row_txttanggal = txttgl.getText();
        String row_txtbln = txtbulan.getText();
        String row_txtidpeg = txtidpeg.getText();
        String row_txtjmlhadir = txthadir.getText();
        String row_txtjmljam = txtjam.getText();
        String row_txttunjangan = txttunjangan.getText();
        String row_txttotal = txtttl.getText();
        String row_txtnoj = txtnojurnal.getText();
        String row_txtadmin = txtadmin.getText();

        if (!"".equals(row_txtno) && !"".equals(row_txtidpeg) &&
            !"".equals(row_txtjmlhadir) && !"".equals(row_txtjmljam)) {
            try {
                Connection con = kon.getConnection();
                Statement stm = con.createStatement();
                stm.executeUpdate("INSERT INTO gaji(slip_gaji, tanggal, bulan,
id_pegawai, jml_hadir, jml_jam, tunjangan, total, username) VALUES "
                    + "(" + row_txtno + ", " + row_txttanggal + ", " + row_txtbln + ", "
                    + row_txtidpeg + ", " + row_txtjmlhadir + ", " + row_txtjmljam + ", "
                    + row_txttunjangan + ", " + row_txttotal + ", " + row_txtadmin + ")");

                stm.executeUpdate("INSERT INTO jurnal(id, no_jurnal, slip_gaji,
kd_akun, tanggal, nm_akun, debit, kredit) VALUES "
                    + "('0', " + row_txtnoj + ", " + row_txtno + ", " + 511 + ", "
                    + row_txttanggal + ", 'Biaya Gaji', " + row_txttotal + ", " + 0 + ")");

                nojur();
                stm.executeUpdate("INSERT INTO jurnal(id, no_jurnal, slip_gaji,
kd_akun, tanggal, nm_akun, debit, kredit) VALUES "

```

```

        + "('0', "" + row_txtnoj + "", "" + row_txtno + "", "" + 101 + "", "" +
row_txttanggal + "", 'Kas', "" + 0 + "", "" + row_txttotal + "")");

con.close();

regrid();
//norut();
regrid2();
try {
    HashMap hash = new HashMap();
    hash.put("kode", txtidgaji.getText());

    File file1 = new File("src/laporan/slipgaji.jrxml");
    JasperDesign jasperDesign = JRXmlLoader.load(file1);
    JasperReport jasperReport =
JasperCompileManager.compileReport(jasperDesign);
    JasperPrint jasperPrint = JasperFillManager.fillReport(jasperReport,
hash, kon.getConnection());
    JasperViewer.viewReport(jasperPrint, false);

    //bersih();
} catch (JRException e) {
    JOptionPane.showMessageDialog(null, "Error " + e);
}
this.dispose();

} catch (SQLException e) {
    JOptionPane.showMessageDialog(null, "Error " + e);
}
} else {
    JOptionPane.showMessageDialog(null, "Terdapat inputan yang kosong ! ",
"Kesalahan", JOptionPane.WARNING_MESSAGE);
}
regrid();
regrid2();
}

private void jTambahActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    //aktif();
    norut();
    nojur();
    jBatal.setEnabled(true);
    jSimpan.setEnabled(true);
    jTambah.setEnabled(false);
    jKeluar.setEnabled(false);
}

private void jBatalActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

```

```

    aktif();
    jBatal.setEnabled(true);
    jSimpan.setEnabled(false);
    jTambah.setEnabled(true);
    jKeluar.setEnabled(true);
    bersih();
}
private void jKeluarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dispose();
}
private void txtttlActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}
private void btncaripegawaiActionPerformed(java.awt.event.ActionEvent evt) {
    boolean closable = true;
    caripeg a = new caripeg(this, rootPaneCheckingEnabled);
    a.kons = this;
    a.setVisible(true);

    if (!a.isVisible()) {
        txtidpeg.setText(idpeg);
        txtnmpeg.setText(namapeg);
        txtjab.setText(idjab);
    }
    if (!"GR".equals(txtjab.getText())) {
        txthadir.setEnabled(true);
        txtjam.setEnabled(false);
        txtjam.setText("0");
        //txthadir.requestFocus();
    } else {
        txtjam.setEnabled(true);
        txthadir.setEnabled(false);
        txthadir.setText("0");
        //txtjam.requestFocus();
    }
    GetDataJab();
}
private void txthadirKeyReleased(java.awt.event.KeyEvent evt) {
    int jml = Integer.parseInt(txthadir.getText());
    int gapok = Integer.parseInt(txtgapok.getText());
    int tunjangan = Integer.parseInt(txttunjangan.getText());
    int subtotal;
    subtotal = (jml * gapok) + tunjangan;
    txtttl.setText(String.valueOf(subtotal));
}
private void txtjamKeyReleased(java.awt.event.KeyEvent evt) {
    int jml = Integer.parseInt(txtjam.getText());
    int gapok = Integer.parseInt(txtgapok.getText());

```

```

int tunjangan = Integer.parseInt(txttunjangan.getText());
int subtotal;
subtotal = (jml * gapok) + tunjangan;
txtttl.setText(String.valueOf(subtotal));
}
private void txtadminActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}
private void txtjabActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}
private void txtidjabActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}
private void txttunjanganKeyReleased(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    int jml=0;
    if("GR".equals(txtjab.getText())){
        jml = Integer.parseInt(txtjam.getText());
    }else{
        jml = Integer.parseInt(txthadir.getText());
    }
    int gapok = Integer.parseInt(txtgapok.getText());
    int tunjangan = Integer.parseInt(txttunjangan.getText());
    int subtotal;
    subtotal = (jml * gapok) + tunjangan;
    txtttl.setText(String.valueOf(subtotal));
}
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc="Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.
    *
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
        java.util.logging.Logger.getLogger(fgaji.class.getName()).log(java.util.logging.Level
.SEVERE, null, ex);

```

```

    } catch (InstantiationException ex) {
java.util.logging.Logger.getLogger(fgaji.class.getName()).log(java.util.logging.Level
.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {
java.util.logging.Logger.getLogger(fgaji.class.getName()).log(java.util.logging.Level
.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
java.util.logging.Logger.getLogger(fgaji.class.getName()).log(java.util.logging.Level
.SEVERE, null, ex);
    }
//</editor-fold>
/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new fgaji().setVisible(true);
    }
});
}
}

```

3.5.2. Blackbox Testing

Tabel III.18
Blackbox Testing Login

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	Username dan Password tidak diisi dan kemudian diklik Login	Username (kosong) dan Password (kosong)	Sistem akan menolak akses user dan menampilkan pesa "Username atau password belum terdaftar"	Sesuai	Valid
2	Username diisi dan Password tidak diisi kemudian diklik Login	Username (Diisi) dan Password (Kosong)	Sistem akan menolak akses user dan menampilkan pesan "Username atau password belum terdaftar"	Sesuai	Valid
3	Username diisi dengan salah dan Password diisi dengan benar	Username Coba (Salah) dan Password (Benar)	Sistem akan menolak akses user dan menampilkan pesan "Username atau password belum terdaftar"	Sesuai	Valid
4	Username diisi dengan benar dan Password diisi dengan salah	Username Coba (Benar) dan Password (Salah)	Sistem akan menolak akses user dan menampilkan pesa "Username atau Password Salah!"	Sesuai	Valid

5	<i>Username</i> diisi dengan benar dan <i>Password</i> diisi dengan benar	<i>Username</i> Coba (Benar) dan <i>Password</i> (Benar)	Sistem akan menerima akses <i>Login</i> dan menampilkan Menu Utama	Sesuai	Valid
---	---	--	--	--------	-------

Tabel III.19
Blackbox Testing Data Gaji

No	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	<i>Input</i> Data Gaji (Data yang di input tidak lengkap) lalu klik Simpan	Data tidak diisi lengkap	Sistem tidak akan menyimpan ketika kolom tidak terisi semua, maka akan menampilkan pesan “Terdapat inputan yang kosong !”	Sesuai	Valid
2	<i>Input</i> Numerik (Jumlah Jam / Jumlah Hadir) dengan Huruf	Data Numerik diisi dengan Huruf	Sistem tidak mendukung untuk menampilkan <i>field</i> Total	Sesuai	Valid
3	<i>Input</i> Data Gaji (data yang diisi lengkap) lalu klik Simpan	Data diisi lengkap	Sistem menyimpan dan menampilkan pesan “Data Tersimpan.”	Sesuai	Valid

3.5.3. Spesifikasi *Hardware* dan *Software*

Processor : Intel(R) Celeron(R) CPU N3350 @ 1.10GHz

RAM : 2.5 GB

Monitor : 1366 x 768 *pixel*

Keyboard : 108 *keys*

Printer : Epson L300

Mouse : Standard Mouse

Sistem Operasi : Windows 10 Home Single Language 64-bit

Bahasa Pemrograman : Java Netbeans 8.02

DBMS : MySQL

Antivirus : Smadav