

BAB III

PEMBAHASAN

3.1. Tinjauan Perusahaan

Dalam tinjauan perusahaan, penulis akan menjelaskan tentang sejarah perusahaan, struktur organisasi yang menggambarkan tugas dan tanggung jawab setiap bagian, yang ada dalam Kantor Notaris & P.P.A.T MUJTAHID, SH. Berikut adalah hasil dari pengamatan yang penulis lakukan, yaitu:

3.1.1. Sejarah Perusahaan

Kantor Notaris & P.P.A.T MUJTAHID, SH., adalah salah satu dari banyaknya Kantor Notaris yang berkedudukan di daerah Cikampek, Jawa Barat. Kegiatan kantor setiap harinya sama seperti Kantor Notaris & P.P.A.T pada umumnya yaitu membuat akta-akta otentik mengenai semua perbuatan, perjanjian, dan penetapan yang diharuskan oleh perundang-undangan dan dikehendaki oleh yang berkepentingan untuk dinyatakan dalam akta otentik agar diakui sah secara hukum.

Kantor Notaris & P.P.A.T MUJTAHID, SH., berdiri sejak tahun 2003, beralamat di Jalan Jend. A. Yani No. 86, Telp. 304767, Kecamatan Cikampek, Kabupaten Karawang, Jawa Barat. Merupakan sebuah lembaga profesi sekaligus pejabat umum yang resmi berdasarkan Surat Keputusan Menteri Hukum Republik Indonesia nomor C-663 – HT.03.01 – Th. 1998, tentang Pengangkatan Notaris tanggal 1 Desember 1998. Pada tanggal 31 Desember 1998 dilakukan pelantikan dan Pengambilan Sumpah Notaris sebagaimana tertera dalam Berita Acara Pengambilan Sumpah Notaris nomor 877/189/Pem.

Pengangkatan atau penunjukan sebagai Pejabat Pembuat Akta Tanah, yang selanjutnya disebut P.P.A.T sudah dilakukan sejak tanggal 3 Juni 1996 berdasarkan Surat Keputusan Menteri Negara Agraria/ Kepala Badan Pertanahan Nasional nomor 5-XI-1996, dan pada tanggal 31 Juli 1996 dilakukan pengambilan sumpah atau janji P.P.A.T sebagaimana tertuang dalam Berita Acara Pengambilan Sumpah/ Janji P.P.A.T nomor 640-4452-1996, dengan daerah kerja Kabupaten Karawang.

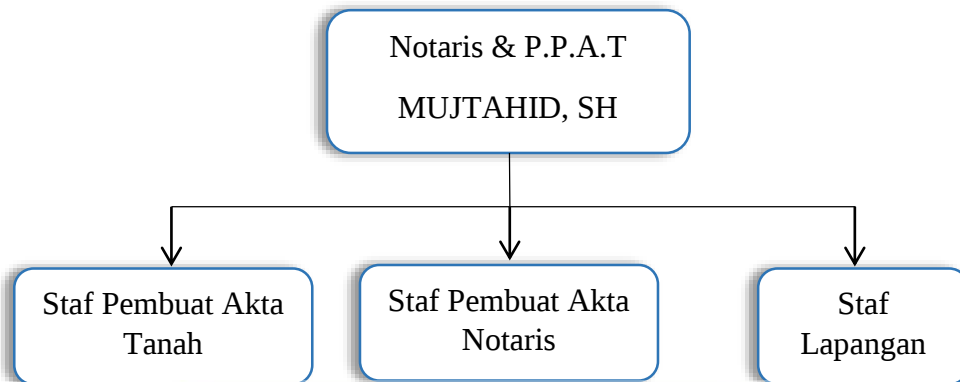
Kantor Notaris & P.P.A.T MUJTAHID, SH., di dirikan dengan tujuan untuk mempermudah masyarakat dalam mengurus berkas atau dokumen berharga dan membuat dokumen-dokumen perizinan. Hingga saat ini Kantor Notaris & P.P.A.T MUJTAHID, SH masih bertahan dan akan terus meningkatkan kualitas terhadap pelayanan kepada masyarakat.

Meskipun letak dan posisi kantor berdekatan dengan Kantor Notaris & P.P.A.T lainnya, Kantor Notaris & P.P.A.T. MUJTAHID, SH., hingga kini masih ramai dan banyak yang menggunakan layanan jasa dikantor tersebut. Tidak hanya melayani masyarakat umum saja, tetapi juga sudah bekerjasama dengan beberapa PT, Koperasi, dan Bank. Kantor Notaris & P.P.A.T MUJTAHID, SH., kini telah memiliki karyawan sebanyak enam orang. Karyawan tersebut terdiri dari staf pembuat akta notaris, staf pembuat akta tanah, dan staf lapangan.

3.1.2. Struktur Organisasi

Dalam suatu lembaga atau perusahaan struktur organisasi sangat diperlukan untuk mempermudah suatu kegiatan berjalan lancar, selain itu struktur organisasi diperlukan sebagai pembagian wewenang dan tanggung jawab setiap anggota yang ada di suatu lembaga atau perusahaan tersebut. Struktur organisasi dibuat untuk memudahkan komunikasi dan pengarahan dalam setiap kegiatan dan diharapkan dapat

bekerja secara profesional dan optimal. Gambaran tentang struktur organisasi yang ada pada Kantor Notaris & P.P.A.T MUJTAHID, SH., secara lengkap dapat dilihat pada gambar dibawah ini:



Sumber: Kantor Notaris & P.P.A.T MUJTAHID, SH

Gambar III.1

Struktur Organisasi Kantor Notaris & P.P.A.T MUJTAHID, SH

Deskripsi fungsi dari wewenang dan tanggung jawab masing-masing bagian dalam struktur organisasi yang terdapat di Kantor Notaris & P.P.A.T MUJTAHID, SH., adalah sebagai berikut:

1. Notaris
 - a. Sebagai pemimpin sekaligus pemilik kantor, memiliki wewenang untuk mengambil segala keputusan yang berkaitan dengan kantor maupun pekerjaan.
 - b. Bertanggung jawab dalam memimpin dan mengarahkan karyawan untuk optimal dalam bekerja.
 - c. Menerima dan memeriksa laporan keuangan
2. Staf pembuat akta tanah
 - a. Bertanggung jawab dalam pembuatan akta tanah, seperti akta jual beli (AJB), hibah, akta pembagian hak bersama (APHB), dan masih banyak lagi, yang bertujuan untuk peralihan hak atas tanah yang dimiliki.

- b. Membuat laporan keuangan setiap kali ada akta tanah yang harus di tandatangani oleh notaris.

3. Staf pembuat akta notaris

- a. Bertanggung jawab dalam pembuatan akta notaris, seperti akta perjanjian pengikatan jual beli (PPJB), kuasa, pendirian perseroan terbatas (PT), pendirian perseroan komanditer (CV), dan masih banyak lagi, yang bertujuan sebagai bukti dokumen perizinan yang sah dalam hukum.
- b. Membuat laporan keuangan setiap kali ada akta notaris yang harus di tandatangani oleh notaris.

4. Staf lapangan

- a. Bertanggung jawab melanjutkan proses pembuatan akta dan dokumen-dokumen perizinan dari kantor kepada pihak bagian hukum dalam bidang masing-masing.
- b. Bertanggung jawab menjaga berkas-berkas yang sedang diproses diluar kantor agar tidak hilang sampai proses selesai.

3.2. Tinjauan Kasus

Dalam tinjauan kasus ini, penulis menemukan fakta bahwa prosedur sistem yang ada saat ini mulai dari pengajuan balik nama hingga pembuatan laporan keuangan atas proses balik nama sangat memerlukan waktu yang cukup lama, oleh karena itu penulis hanya berfokus pada prosedur sistem pendapatan atas proses balik nama.

Penulis akan menjelaskan tentang proses bisnis sistem berjalan, *activity diagram* sistem berjalan, dokumen masukan dan dokumen keluaran, permasalahan pokok, serta pemecahan masalah yang ada pada kantor tersebut:

3.2.1. Proses Bisnis Sistem Berjalan

Proses bisnis sistem berjalan merupakan proses yang saat ini diterapkan. Proses bisnis sistem berjalan yang ada pada Kantor Notaris & P.P.A.T Mujtahid, SH., terbagi menjadi tiga proses, dan setiap prosesnya saling berkaitan, sebagai berikut:

1. Proses Pengajuan Balik Nama

Klien datang dengan membawa berkas persyaratan berupa fotokopi KTP, Kartu Keluarga, Fotokopi SPPT/PBB, dan Setifikat Tanah asli. Kemudian dokumen tersebut diserahkan kepada staf pembuat akta tanah untuk dicek apakah berkas tersebut sudah lengkap. Jika berkas belum lengkap, maka berkas tersebut akan diserahkan kembali kepada klien untuk dilengkapi. Jika sudah lengkap staf pembuat akta tanah memberikan kesepakatan harga balik nama. Setelah disepakati oleh kedua pihak staf pembuat akta tanah memberikan sertifikat tanah kepada staf lapangan untuk dicek. Staf lapangan menyerahkan sertifikat tanah klien kepada pihak Badan Pertanahan Nasional (BPN), setelah data sesuai, BPN membuat tanda terima pengecekan yang akan diserahkan kepada staf lapangan. Staf lapangan tersebut menyerahkan tanda terima pengecekan kepada staf pembuat akta tanah, kemudian staf pembuat akta tanah menyimpan tanda terima pengecekan sebagai arsip dan membuat akta, lalu melakukan penandatanganan akta antara pihak klien yang menyerahkan hak dengan klien yang menerima hak dan disaksikan oleh staf pembuat akta tanah.

2. Proses Pembayaran

Staf pembuat akta tanah memberikan hasil perhitungan berkaitan dengan dana atau harga balik nama seperti pajak penjual (SSP), pajak pembeli (SSB/BPHTB), pembayaran SPPT/PBB, biaya pengecekan sertifikat tanah, biaya Zona Tanah (ZNT), dan biaya notaris kepada klien. Setelah itu klien membayarkan sejumlah dana yang dikeluarkan untuk harga balik nama yang sudah disepakati. Staf pembuat

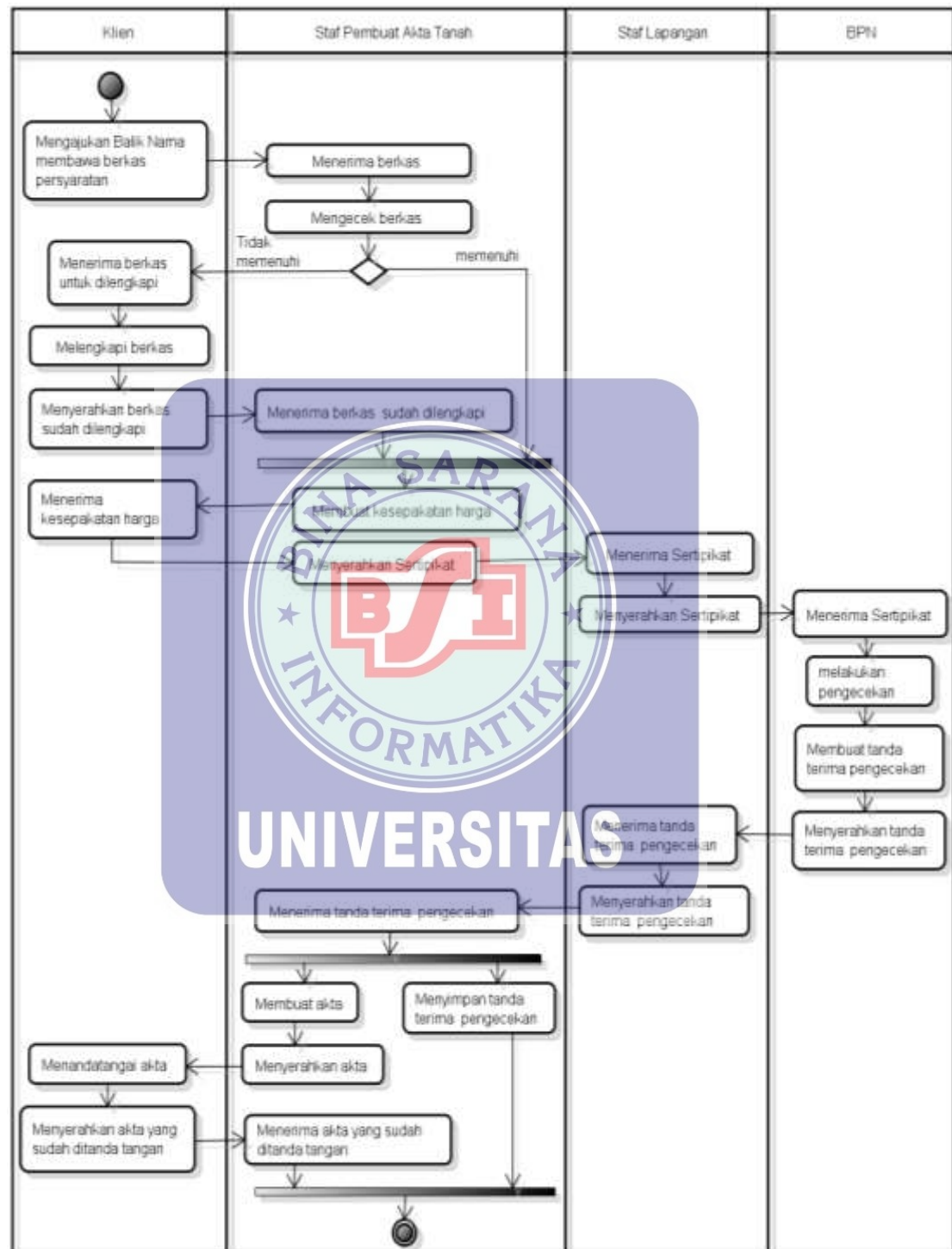
akta tanah membuat kwitansi 2 rangkap atas pembayaran balik nama dan tanda terima berkas 2 rangkap. Kwitansi dan tanda terima warna putih diserahkan kepada klien, sedangkan kwitansi dan tanda terima warna merah muda disimpan sebagai arsip. Setelah itu staf pembuat akta tanah melakukan pembayaran pajak penjual (SSP) ke bank, bank menerima pembayaran tersebut kemudian membuat tanda terima pembayaran pajak 2 rangkap yang diserahkan kepada staf pembuat akta tanah. Setelah menerima tanda terima tersebut, tanda terima warna putih diberikan kepada klien dan tanda terima warna merah muda disimpan. Staf pembuat akta tanah memberikan SPPT/PBB kepada staf lapangan untuk dibayarkan, staf lapangan membayarkan SPPT/PBB ke kantor pajak. Kantor pajak akan mencetak lalu menyerahkan bukti pembayaran pajak kepada staf lapangan, kemudian bukti pajak tersebut diserahkan kepada staf pembuat akta tanah. Setelah itu staf pembuat akta tanah membuat SSB/BPHTB dan dokumen ZNT. Dokumen ZNT kemudian diberikan kepada staf lapangan, staf lapangan menyerahkannya ke pihak BPN, setelah diproses oleh pihak BPN akan diberikan kembali kepada staf lapangan, lalu diserahkan ke staf pembuat akta tanah. Bukti pembayaran pajak, SSB/BPHTB, dan dokumen ZNT nantinya akan digunakan sebagai syarat balik nama.

3. Prosedur Laporan Keuangan atas Proses Balik Nama

Setelah seluruh dokumen siap, staf pembuat akta tanah membuat laporan keuangan kemudian menyerahkan laporan tersebut beserta dokumen dan sejumlah dana biaya notaris untuk ditanda tangani oleh notaris. Setelah notaris menerima laporan, beserta dokumen dan sejumlah dana biaya notaris, selanjutnya notaris memeriksa laporan dan semua dokumen untuk ditanda tangani. Setelah selesai laporan dan semua dokumen diserahkan kembali ke staf pembuat akta tanah.

3.2.2. Activity Diagram

1. Activity diagram pengajuan balik nama

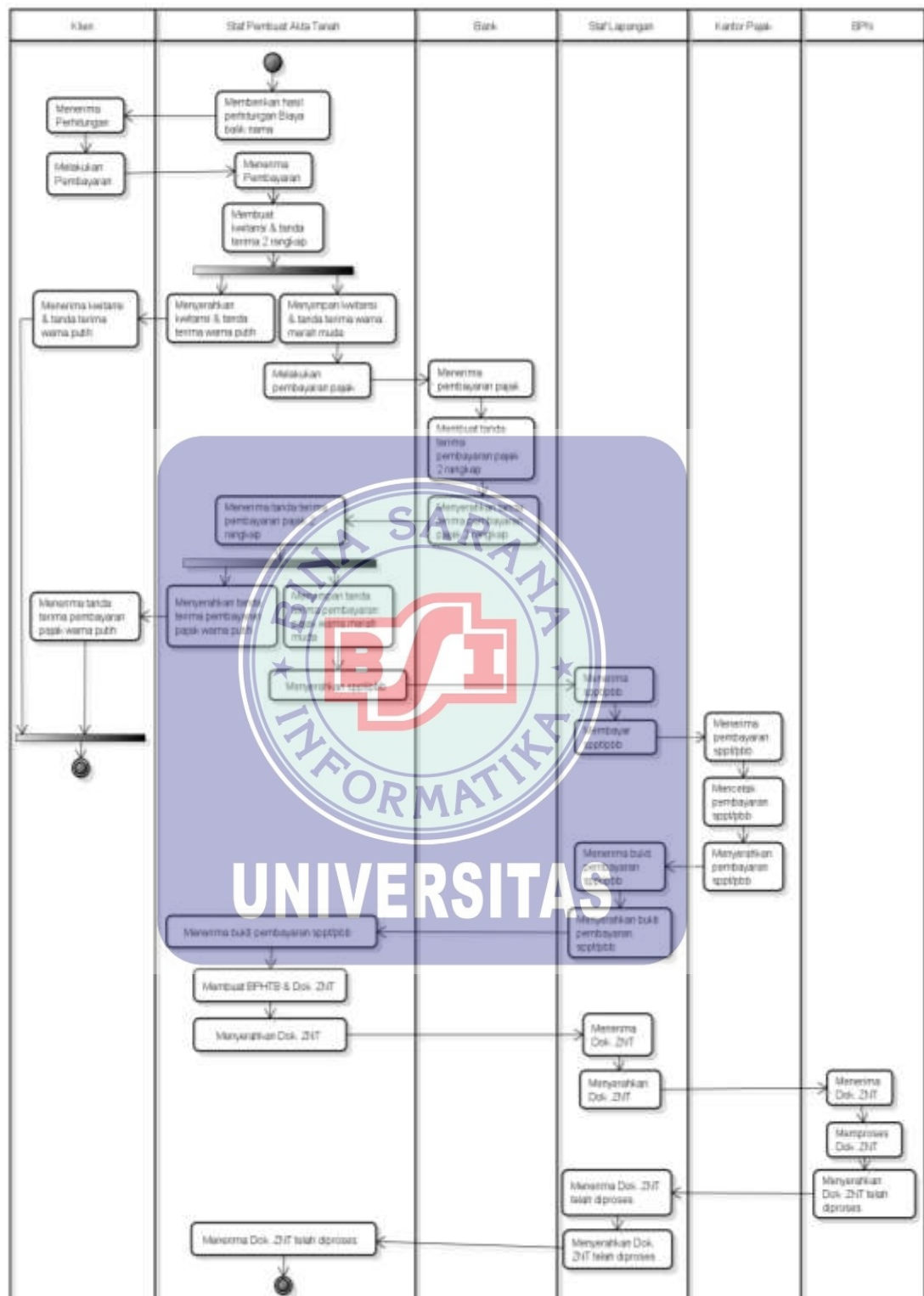


Sumber: Proses pengajuan balik nama Kantor Notaris & PPAT MUJTAHID, SH

Gambar III.2

Activity Diagram Pengajuan Balik Nama

2. Activity diagram pembayaran

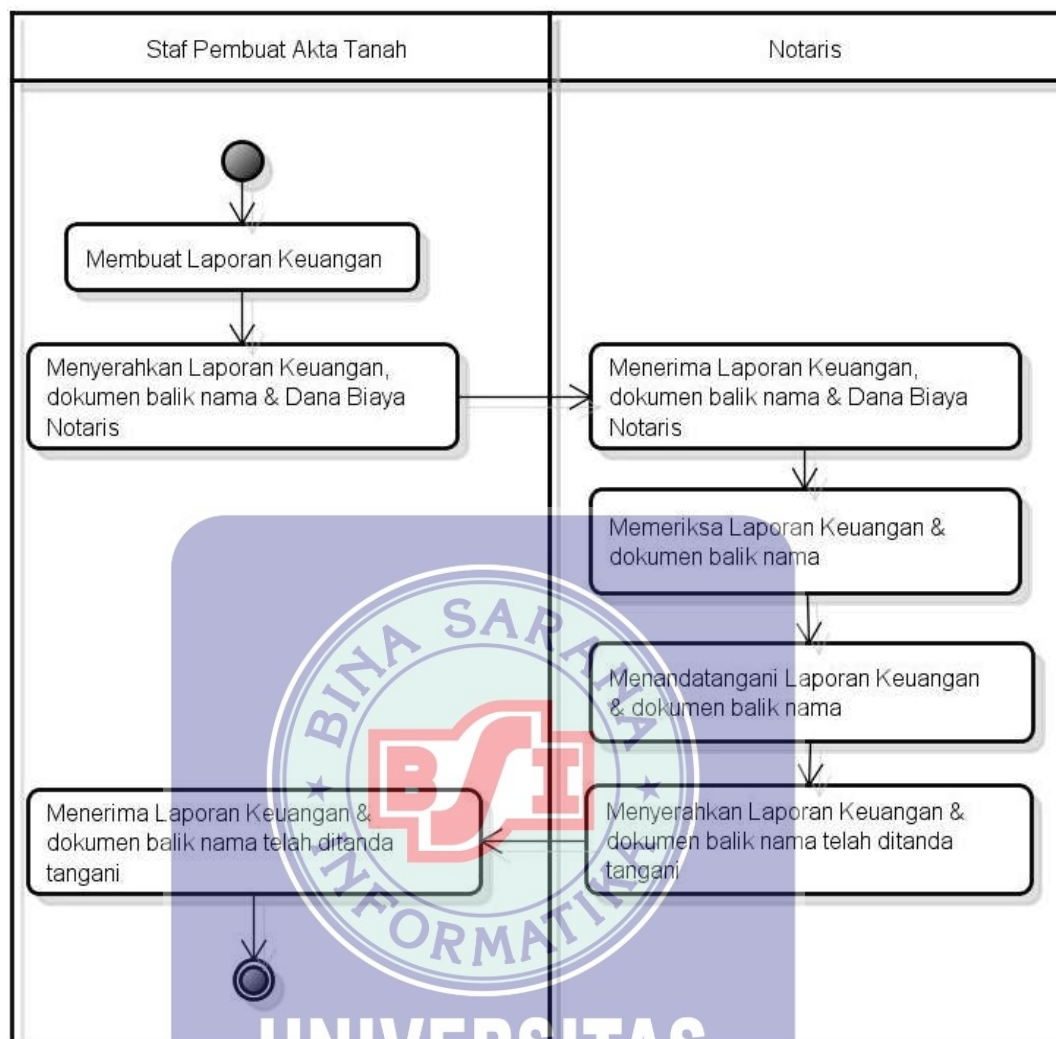


Sumber: Proses Pembayaran balik nama Kantor Notaris & PPAT MUJTAHID, SH

Gambar III.3

Activity Diagram Pembayaran

3. Activity diagram laporan keuangan atas proses balik nama



Sumber: Proses laporan Balik Nama Kantor Notaris & PPAT MUJTAHID, SH

Gambar III.4

Activity Diagram Laporan Baik Nama

3.2.3. Dokumen Masukan

Dokumen masukan adalah semua dokumen yang masuk yang digunakan sebagai dasar data untuk menunjang pemrosesan suatu proses sesuai dengan kebutuhan kegiatannya. Adapun dokumen-dokumen masukan yang ada pada proses balik nama adalah sebagai berikut:

1. Nama Dokumen : Fotokopi KTP dan Kartu Keluarga
- Fungsi : Sebagai Data Pribadi Klien
- Sumber : Klien
- Tujuan : Untuk mengetahui Data diri klien dan pembuatan akta
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran A.1

2. Nama Dokumen : Sertifikat Tanah Asli
- Fungsi : Sebagai Data Bukti Kepemilikan Tanah
- Sumber : Klien
- Tujuan : Untuk membuat akta
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran A.2

3. Nama Dokumen : Tanda Terima Pajak Penjual (SSP) warna merah muda
- Fungsi : Sebagai bukti pembayaran pajak penjual (SSP)
- Sumber : Bank
- Tujuan : Untuk arsip data balik nama
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran A.3

4. Nama Dokumen : Kwitansi warna merah muda
- Fungsi : Sebagai bukti pembayaran
- Sumber : Kantor
- Tujuan : Untuk Mengetahui jumlah dana yang dibayarkan
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran A.4
5. Nama Dokumen : Tanda terima berkas warna merah muda
- Fungsi : Sebagai bukti penerimaan berkas dari klien
- Sumber : Kantor
- Tujuan : Untuk Mengetahui berkas apa saja yang diserahkan
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran A.5
6. Nama Dokumen : Tanda terima pengecekan sertifikat tanah
- Fungsi : Sebagai bukti bahwa sertifikat tanah telah dicek
- Sumber : BPN
- Tujuan : Untuk Mengetahui bahwa sertifikat tanah tidak bermasalah
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap kali pengecekan sertifikat tanah
- Format : Lampiran A.6

7. Nama Dokumen : Fotokopi SPPT/PBB
- Fungsi : Sebagai salah satu syarat balik nama dan arsip
- Sumber : Klien
- Tujuan : Untuk membuat akta dan pembayaran SPPT/PBB
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran A.7
8. Nama Dokumen : Laporan Balik Nama
- Fungsi : Sebagai salah satu bukti transaksi yang telah dilakukan
- Sumber : Karyawan Notaris
- Tujuan : Untuk memberikan informasi transaksi kepada notaris
- Media : Kertas
- Jumlah : -
- Frekuensi : Setiap Transaksi
- Format : Lampiran A8

3.2.4. Dokumen Keluaran

Dokumen keluaran adalah dokumen-dokumen yang keluar atau bukti hasil dari prosedur yang dilakukan dalam proses balik nama secara keseluruhan. Adapun dokumen-dokumen keluaran tersebut sebagai berikut:

1. Nama Dokumen : Kwitansi warna putih
- Fungsi : Sebagai bukti pembayaran
- Sumber : Kantor
- Tujuan : Untuk Mengetahui jumlah dana yang dibayarkan

- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran B.1
2. Nama Dokumen : Tanda terima warna putih
- Fungsi : Sebagai bukti bahwa berkas klien ada di kantor kami
- Sumber : Kantor
- Tujuan : Untuk mengetahui berkas apa saja yg diserahkan
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran B.2
3. Nama Dokumen : Tanda Terima Pajak Penjual (SSP) Warna Putih
- Fungsi : Sebagai bukti bahwa telah dibayarnya paja penjual
- Sumber : Bank
- Tujuan : Untuk salah satu bukti bahwa berkas telah diproses
- Media : Kertas
- Jumlah : Satu Lembar
- Frekuensi : Setiap Klien Baru
- Format : Lampiran B.3
4. Nama Dokumen : bukti pembayaran SPPT/PBB
- Fungsi : Sebagai bukti telah dibayarnya SPPT/PBB
- Sumber : Kantor Pajak
- Tujuan : Untuk syarat pelengkap balik nama
- Media : Kertas

Jumlah	: Satu Lembar
Frekuensi	: Setiap Klien Baru
Format	: Lampiran B.4
5. Nama Dokumen	: SSB/BPHTB
Fungsi	: Sebagai Bukti pajak pembeli
Sumber	: Kantor
Tujuan	: Untuk syarat pelengkap balik nama
Media	: Kertas
Jumlah	: Satu Lembar
Frekuensi	: Setiap Klien Baru
Format	: Lampiran B.5
6. Nama Dokumen	: Dokumen ZNT
Fungsi	: Sebagai salah satu syarat balik nama
Sumber	: Kantor
Tujuan	: Untuk data pelengkap Proses Balik Nama
Media	: Kertas
Jumlah	: Satu Lembar
Frekuensi	: Setiap Klien Baru
Format	: Lampiran B.6

3.2.5. Permasalahan Pokok

Prosedur sistem yang baik merupakan keinginan semua perusahaan ataupun lembaga. Segingga kegiatan yang dilakukan dapat berjalan sesuai harapan. Namun seiring perkembangan zaman yang semakin pesat terutama dalam teknologi akan mempengaruhi prosedur sistem yang sudah ada dan akhirnya menimbulkan masalah.

Berdasarkan hasil pengamatan pada sistem pendapatan atas proses balik nama dalam Kantor Notaris & P.P.A.T MUJTAHID, SH., penulis menemukan beberapa permasalahan yang terdapat pada sistem pendapatan tersebut, diantaranya yaitu:

1. Tidak adanya penggunaan program dalam pengolahan datanya, pencatatan masih dalam bentuk buku, sehingga apabila hilang tidak ada salinannya.
2. Penggunaan kertas yang sangat banyak, sehingga menyebabkan penumpukan dan kerusakan pada arsip atau dokumen penyimpanan.
3. Tidak adanya bagian keuangan secara keseluruhan yang membuat pembukuan laporan pendapatan masih dipegang oleh masing karyawan, sehingga tidak stabilnya keuangan dan dimungkinkan dapat dipakai untuk keperluan karyawan masing - masing, dan tidak diketahui kas yang masuk.

3.2.6. Pemecahan Masalah

Dengan adanya permasalahan yang ada pada sistem pendapatan, maka untuk meningkatkan mutu dan pelayanan pada Kantor Notaris & P.P.A.T MUJTAHID, SH., perlu adanya sistem baru yang mampu mengatasi masalah-masalah tersebut. Adapun alternatif pemecahan masalah yang akan penulis sampaikan adalah sebagai berikut:

1. Dibuatkannya sistem berbasis *desktop* dengan menggunakan *Java Netbeans 8.1* sehingga dalam pencatatan yang dilakukan dapat lebih cepat dan efektif.
2. Penyimpanan arsip atau dokumen tidak hanya menggunakan kertas, dapat juga dalam bentuk *harddisk*, sehingga dapat mengurangi penggunaan kertas dan kerusakan arsip atau dokumen.
3. Menambah tenaga kerja khususnya yang memahami bagian keuangan, sehingga pembuatan laporan dapat lebih akurat.

3.3. Analisis Kebutuhan Software

Berdasarkan prosedur sistem pendapatan atas proses balik nama yang ada pada Kantor Notaris & P.P.A.T MUJTAHID, SH., penulis menganalisa kebutuhan *software*, yang nantinya akan digunakan dalam pembuatan program, karena sebuah program dapat berjalan dengan baik jika dalam analisa kebutuhan saat perancangan tepat dan sesuai permasalahan yang ada. Analisa kebutuhan tersebut digambarkan melalui perancangan *usecase diagram* dan *activity diagram*, sebagai berikut:

3.3.1. Analisis Kebutuhan

1. Admin

- A.1. Admin dapat *Login*
- A.2. Admin dapat mengelola Data Klien
- A.3. Admin dapat mengelola Data Pengajuan Balik Nama

2. Bagian Keuangan

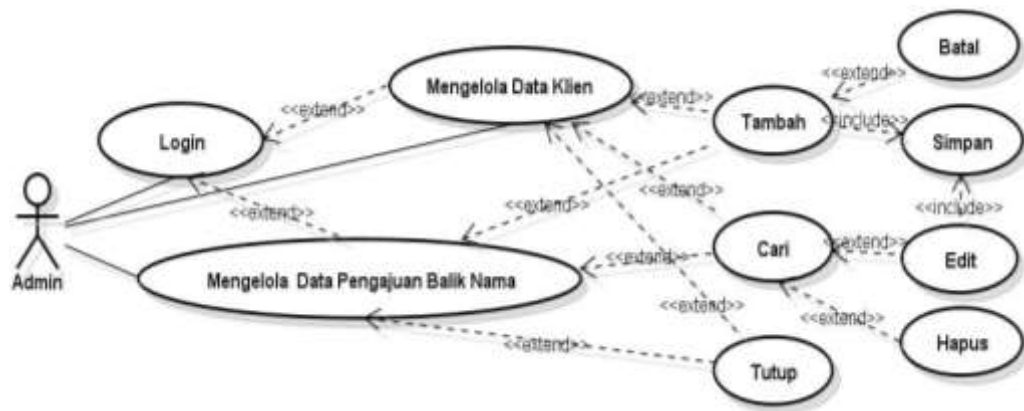
- B.1. Bagian Keuangan dapat *Login*
- B.2. Bagian Keuangan dapat mengelola Data Perkiraan
- B.3. Bagian Keuangan dapat mengelola Pembayaran Jasa Balik Nama
- B.4. Bagian Keuangan dapat melihat Laporan Jurnal
- B.5. Bagian Keuangan dapat mengelola Laporan Pembayaran

3. Notaris

- C.1. Notaris dapat *Login*
- C.2. Notaris dapat mengelola Data Pengguna
- C.3. Notaris dapat melihat Laporan Pembayaran
- C.4. Notaris dapat melihat laporan Jurnal

3.3.2. Usecase Diagram

1. Usecase Diagram Admin



Sumber: Hasil Penelitian (2019).

Gambar III.5

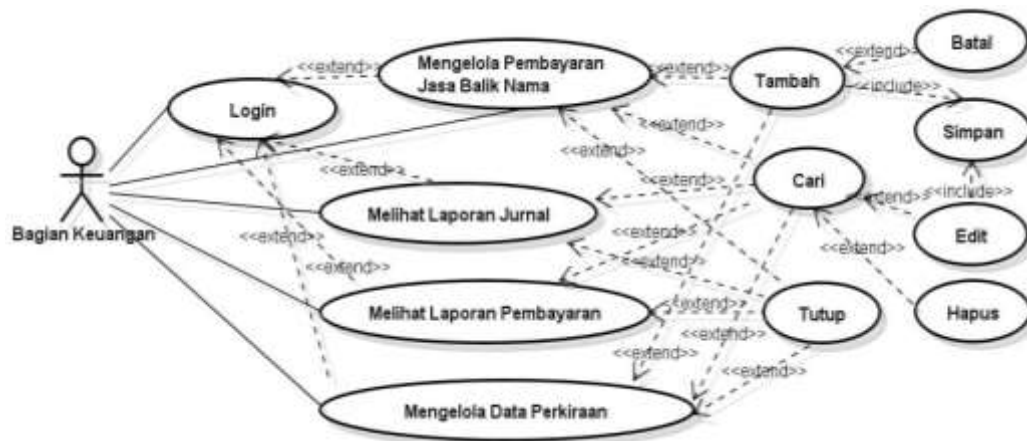
Usecase Diagram Admin

Tabel III.1.

Deskripsi Usecase Diagram Halaman Admin

Usecase Narrative Halaman Admin	
Kebutuhan	A1-A3
Deskripsi	Sistem ini memungkinkan Admin untuk melakukan <i>login</i> dan masuk ke halaman Admin untuk mengelola menu master data klien dan menu transaksi pengajuan balik nama.
Skenario Utama	
Aktor	Admin
Kondisi Awal	Aktor membuka aplikasi halaman admin
Aksi Aktor	Reaksi Sistem
1. Aktor melakukan <i>Login</i>	Sistem akan menampilkan tampilan <i>login</i> , kemudian aktor akan memasukkan <i>Id Pengguna</i> dan <i>Password</i> .
2. Aktor memilih menu master data klien	Sistem akan menampilkan <i>form</i> data klien
3. Aktor memilih menu transaksi data pengajuan balik nama	Sistem akan menampilkan <i>form</i> data pengajuan balik nama
Kondisi Akhir	Jika perintah sesuai maka sistem akan menampilkan seperti permintaan

2. Usecase Diagram Bagian Keuangan



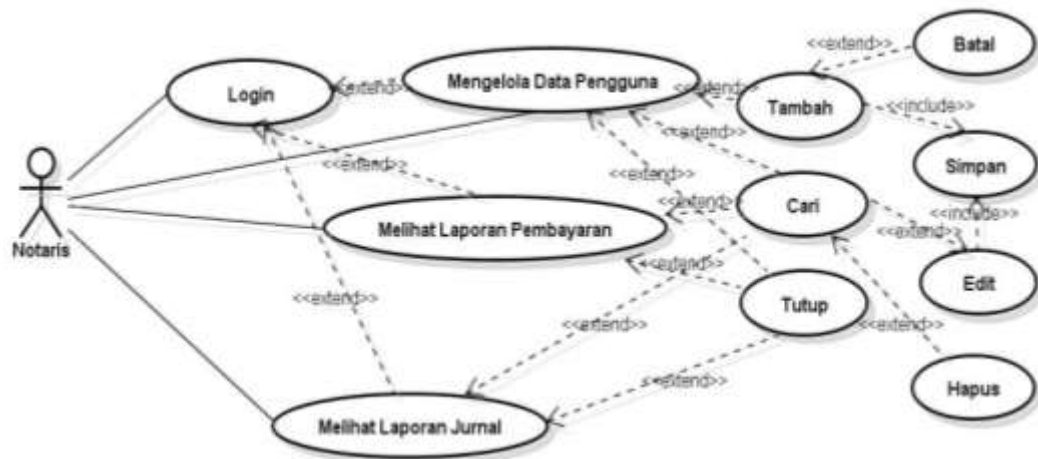
Sumber: Hasil Penelitian (2019).

Gambar III.6
Usecase Diagram Bagian Keuangan

Tabel III.2.
Deskripsi Usecase Halaman Bagian Keuangan

Usecase Narrative Halaman Bagian Keuangan	
Kebutuhan	B1-B5
Deskripsi	Sistem ini memungkinkan Bagian Keuangan untuk melakukan <i>login</i> dan masuk ke halaman Bagian Keuangan untuk mengelola menu transaksi pembayaran, melihat laporan jurnal, laporan pembayaran.
Skenario Utama	
Aktor	Bagian Keuangan
Kondisi Awal	Aktor membuka aplikasi halaman bagian keuangan
Aksi Aktor	Reaksi Sistem
1. Aktor melakukan <i>Login</i>	Sistem akan menampilkan tampilan <i>login</i> , kemudian aktor akan memasukkan <i>Id Pengguna</i> dan <i>Password</i> .
2. Aktor memilih menu transaksi pembayaran	Sistem akan menampilkan <i>form</i> pembayaran jasa balik nama
3. Aktor memilih menu laporan jurnal	Sistem akan menampilkan <i>form</i> laporan jurnal
4. Aktor memilih menu laporan pembayaran	Sistem akan menampilkan <i>form</i> laporan pembayaran
Kondisi Akhir	Jika perintah sesuai maka sistem akan menampilkan seperti permintaan

3. Usecase Diagram Notaris



Sumber: Hasil Penelitian (2019).

Gambar III.7
Usecase Diagram Notaris

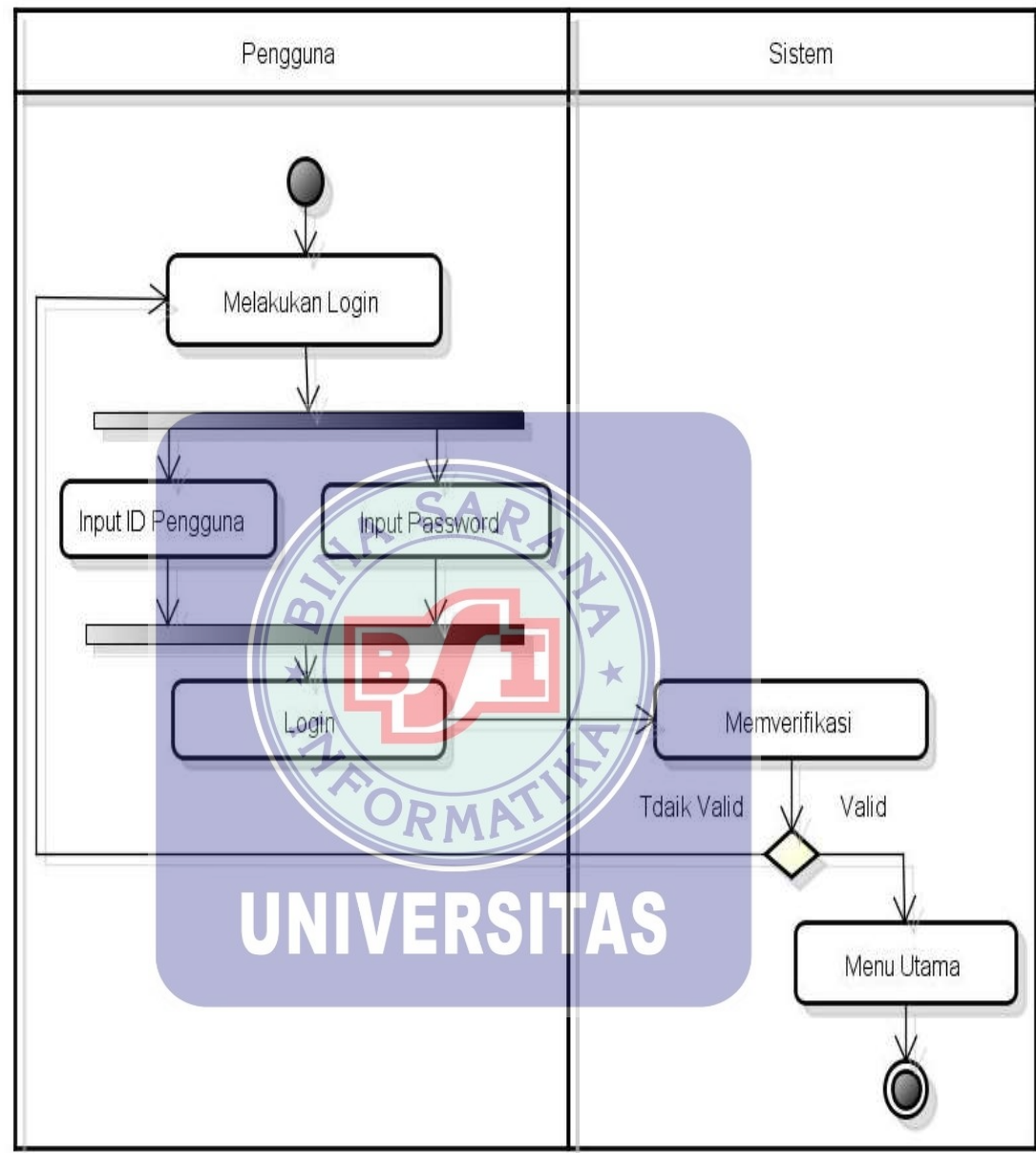
Tabel III.3.

Deskripsi Usecase Diagram Halaman Notaris

Usecase Narrative Halaman Notaris	
Kebutuhan	A1-A3
Deskripsi	Sistem ini memungkinkan Notaris untuk melakukan <i>login</i> dan masuk ke halaman Notaris untuk mengelola menu master pengguna, melihat laporan pembayaran.
Skenario Utama	
Aktor	Notaris
Kondisi Awal	Aktor membuka aplikasi halaman notaris
Aksi Aktor	Reaksi Sistem
1. Aktor melakukan <i>Login</i>	Sistem akan menampilkan tampilan <i>login</i> , kemudian aktor akan memasukkan <i>Id Pengguna</i> dan <i>Password</i> .
2. Aktor memilih menu master pengguna	Sistem akan menampilkan <i>form</i> pengguna
3. Aktor memilih laporan pembayaran	Sistem akan menampilkan <i>form</i> laporan pembayaran
4. Aktor memilih menu laporan jurnal	Sistem akan menampilkan <i>form</i> laporan jurnal
Kondisi Akhir	Jika perintah sesuai maka sistem akan menampilkan seperti permintaan.

3.3.3. Activity Diagram

1. Activity Diagram Login

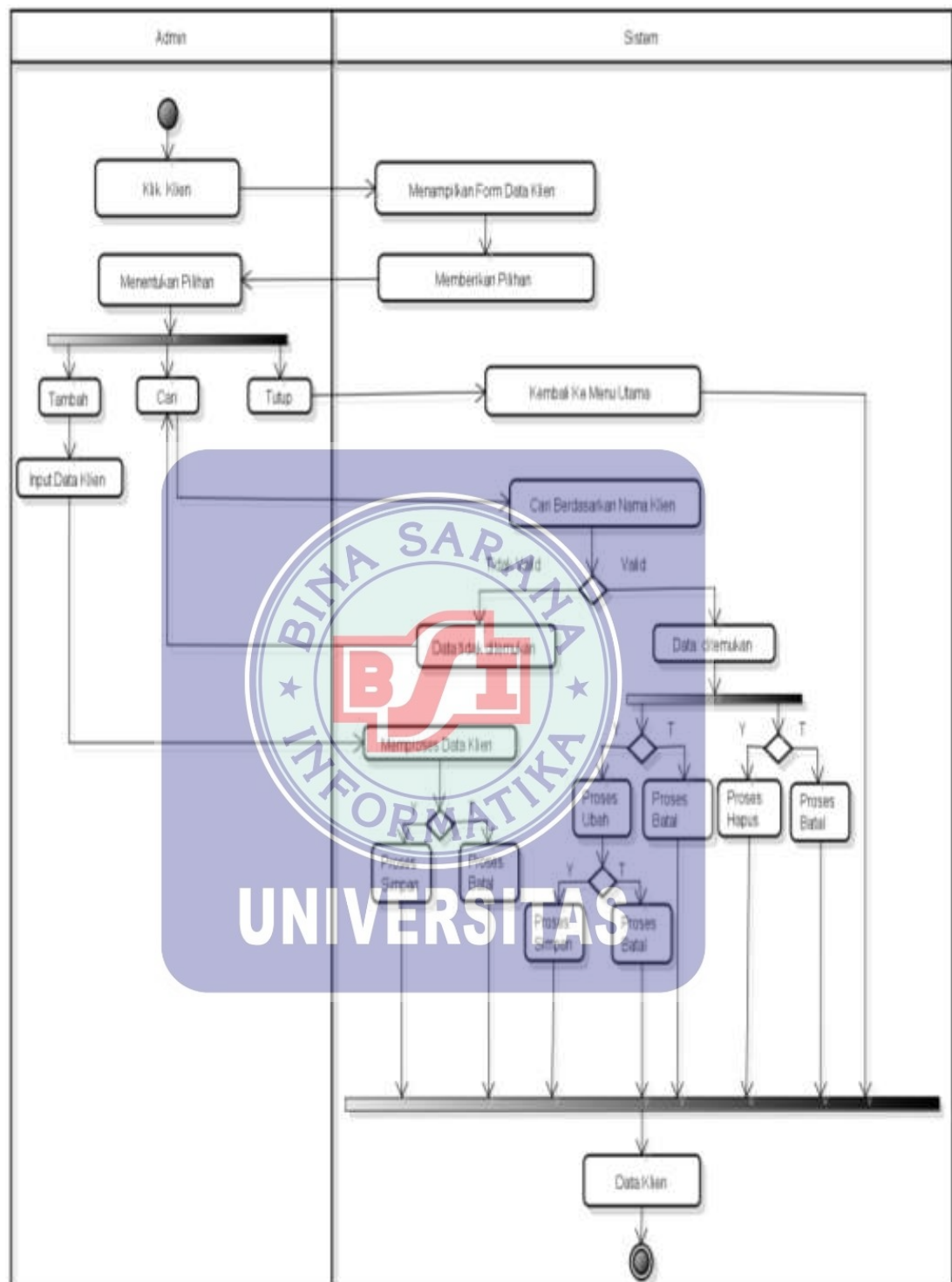


Sumber: Hasil Penelitian (2019).

Gambar III.17

Activity Diagram Login

2. Activity Diagram Mengelola Data Klien

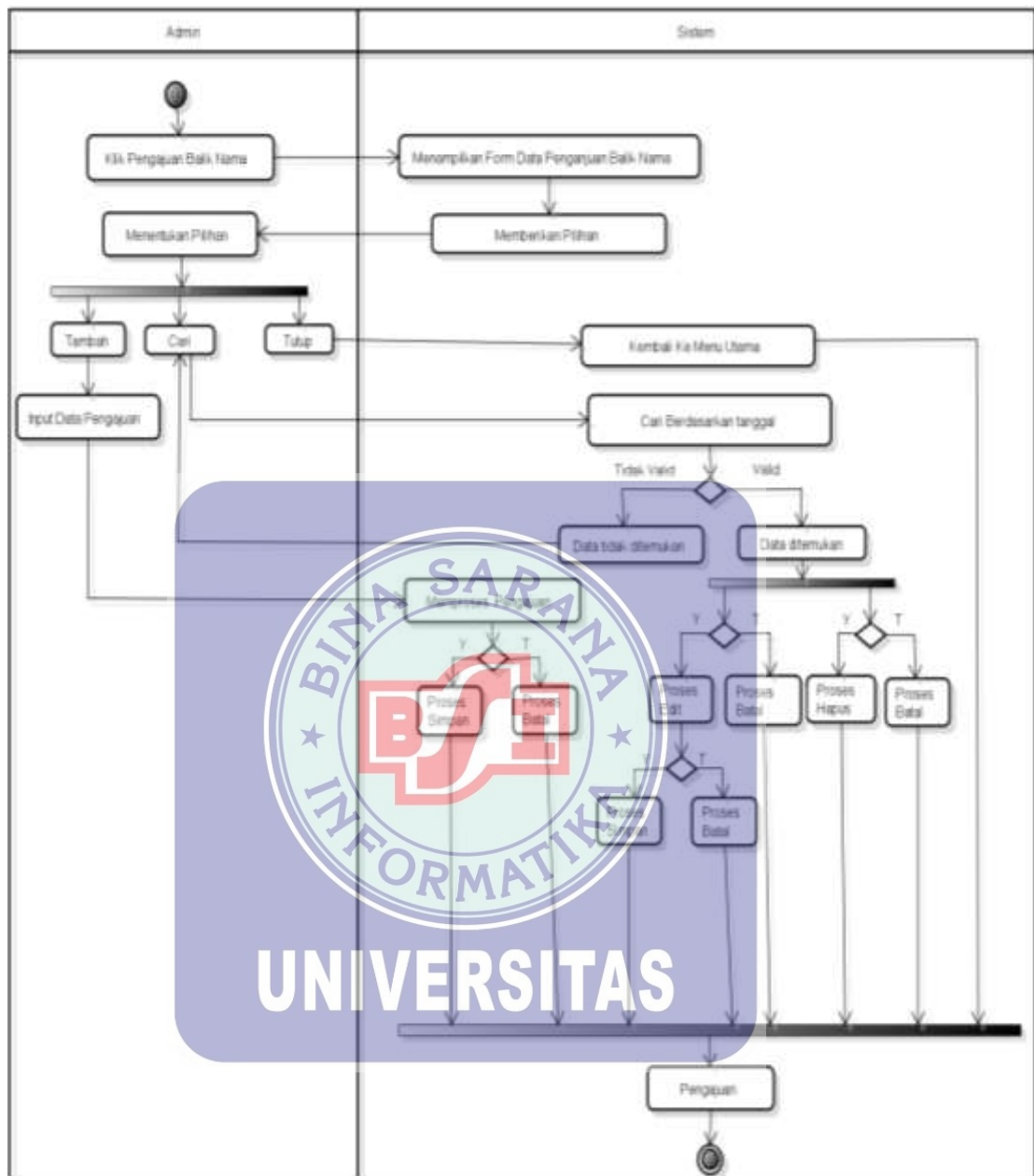


Sumber: Hasil Penelitian (2019).

Gambar III.18

Activity Diagram Mengelola Data Klien

3. Activity Diagram Mengelola Data Pengajuan Balik Nama

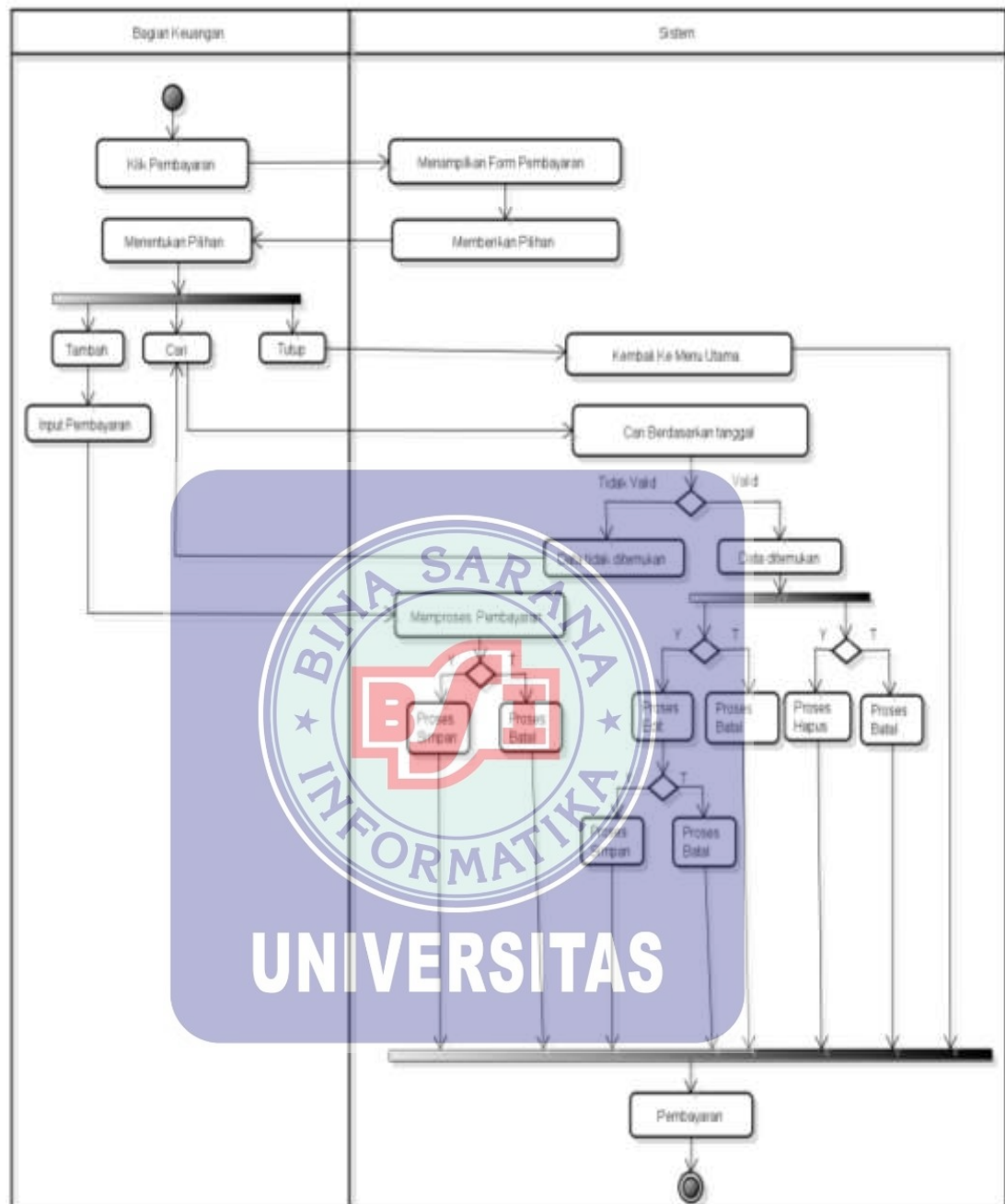


Sumber: Hasil Penelitian (2019).

Gambar III.19

Activity Diagram Mengelola Data Pengajuan Balik Nama

4. Activity Diagram Mengelola Pembayaran Jasa Balik Nama

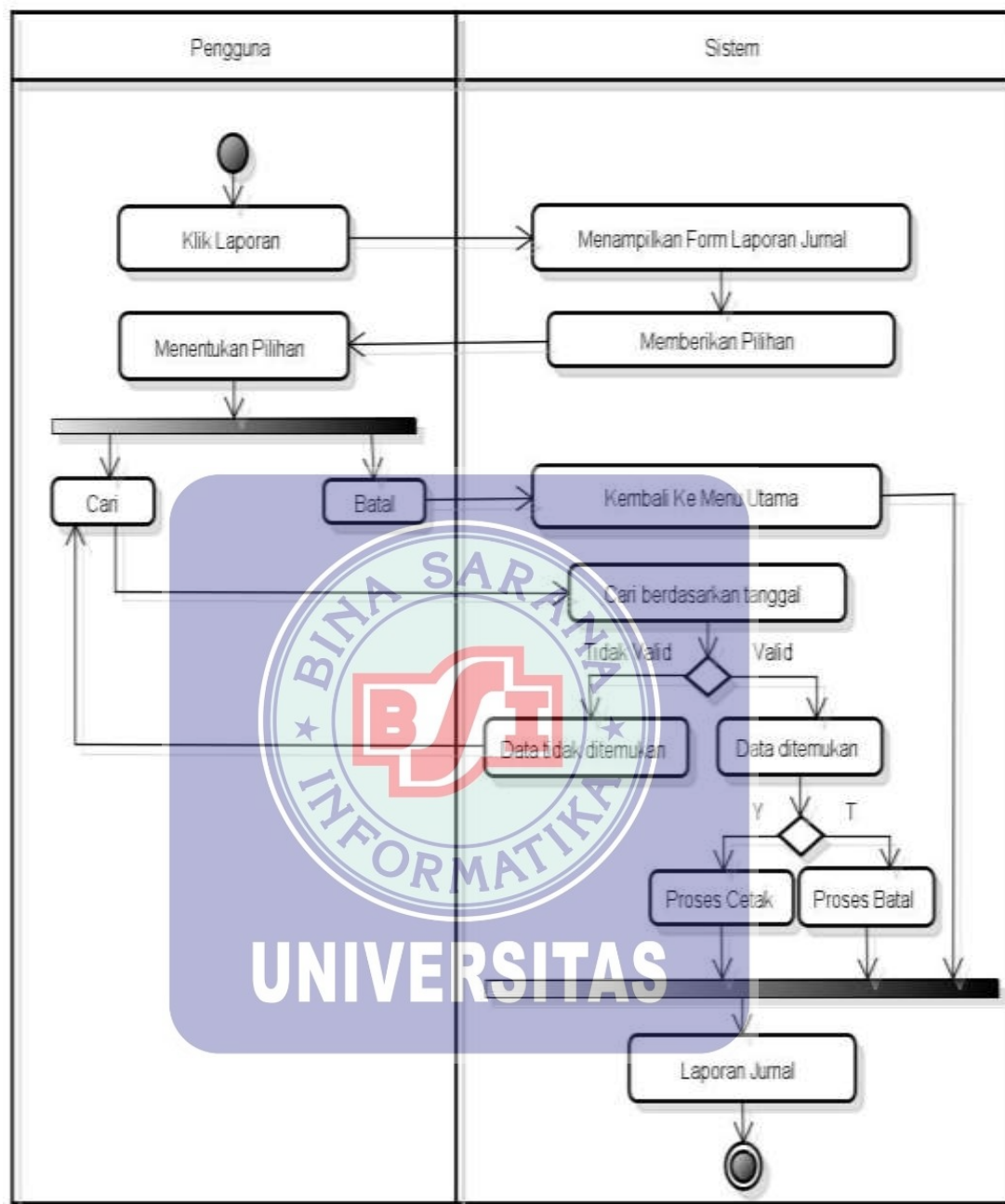


Sumber: Hasil Penelitian (2019).

Gambar III.20

Activity Diagram Mengelola Data Pembayaran Jasa Balik Nama

5. Activity Diagram Melihat Laporan Jurnal

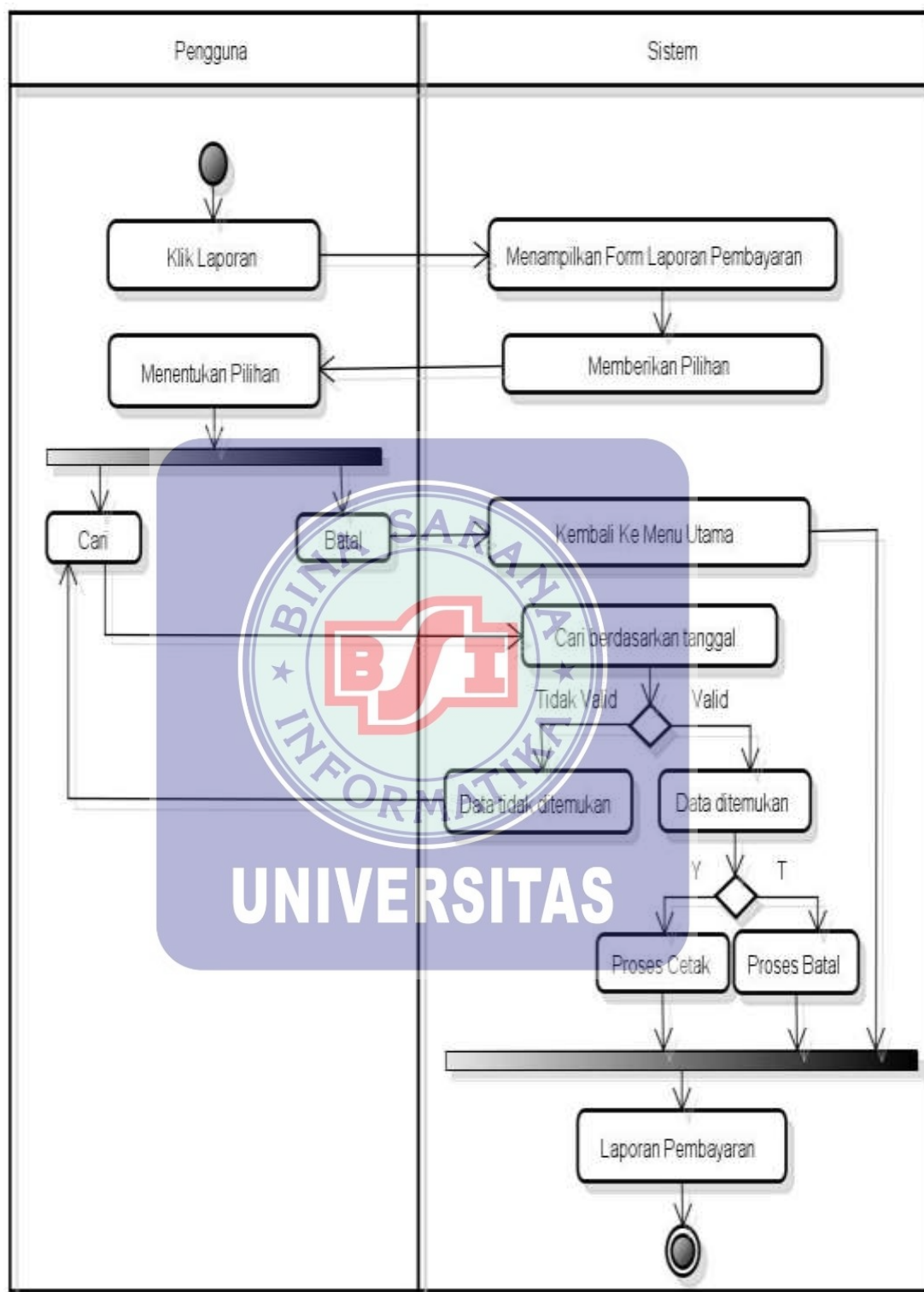


Sumber: Hasil Penelitian (2019).

Gambar III.21

Activity Diagram Melihat laporan Jurnal

6. Activity Diagram Melihat Laporan Pembayaran

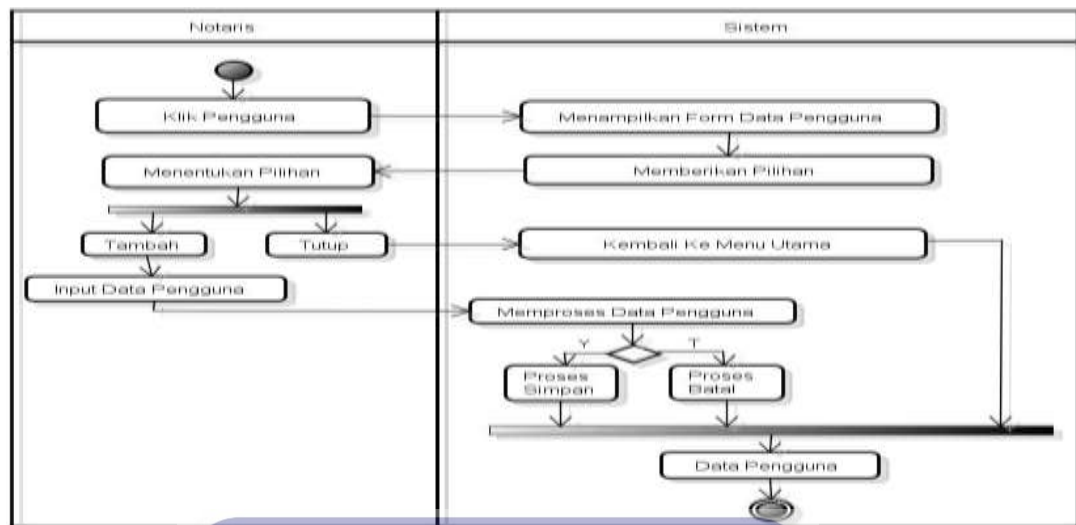


Sumber: Hasil Penelitian (2019).

Gambar III.22

Activity Diagram Melihat Laporan Pembayaran

7. Activity Diagram Mengelola Data Pengguna

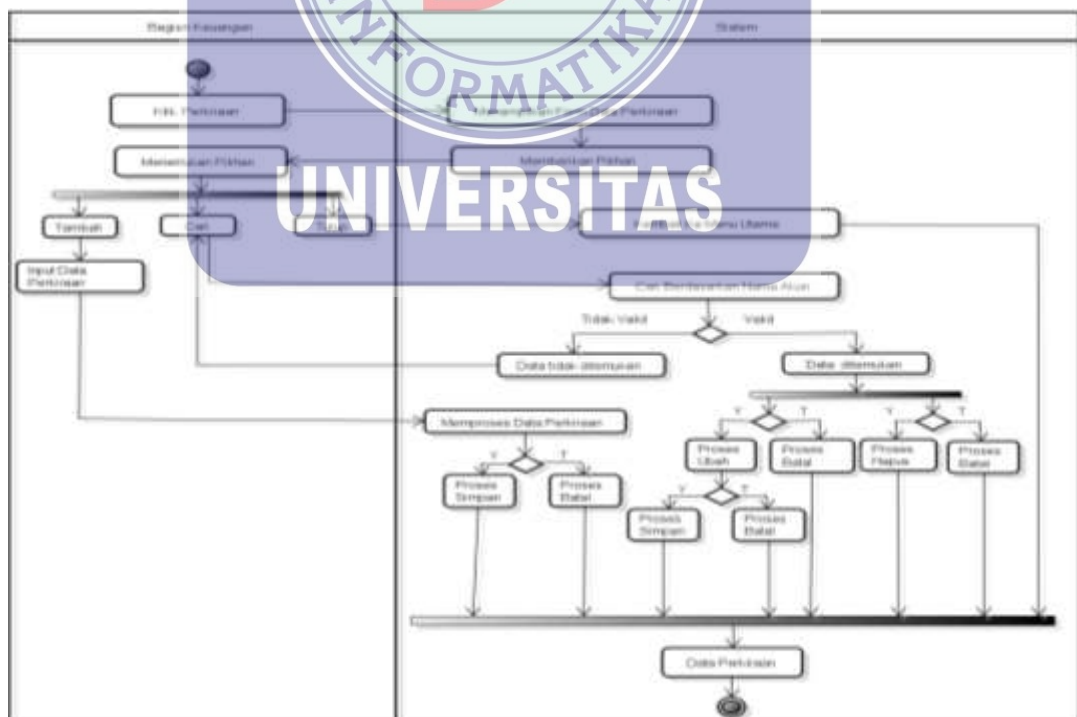


Sumber: Hasil Penelitian (2019).

Gambar III.23

Activity Diagram Mengelola Data Pengguna

8. Activity Diagram Mengelola Data Perkiraan



Sumber: Hasil Penelitian (2019).

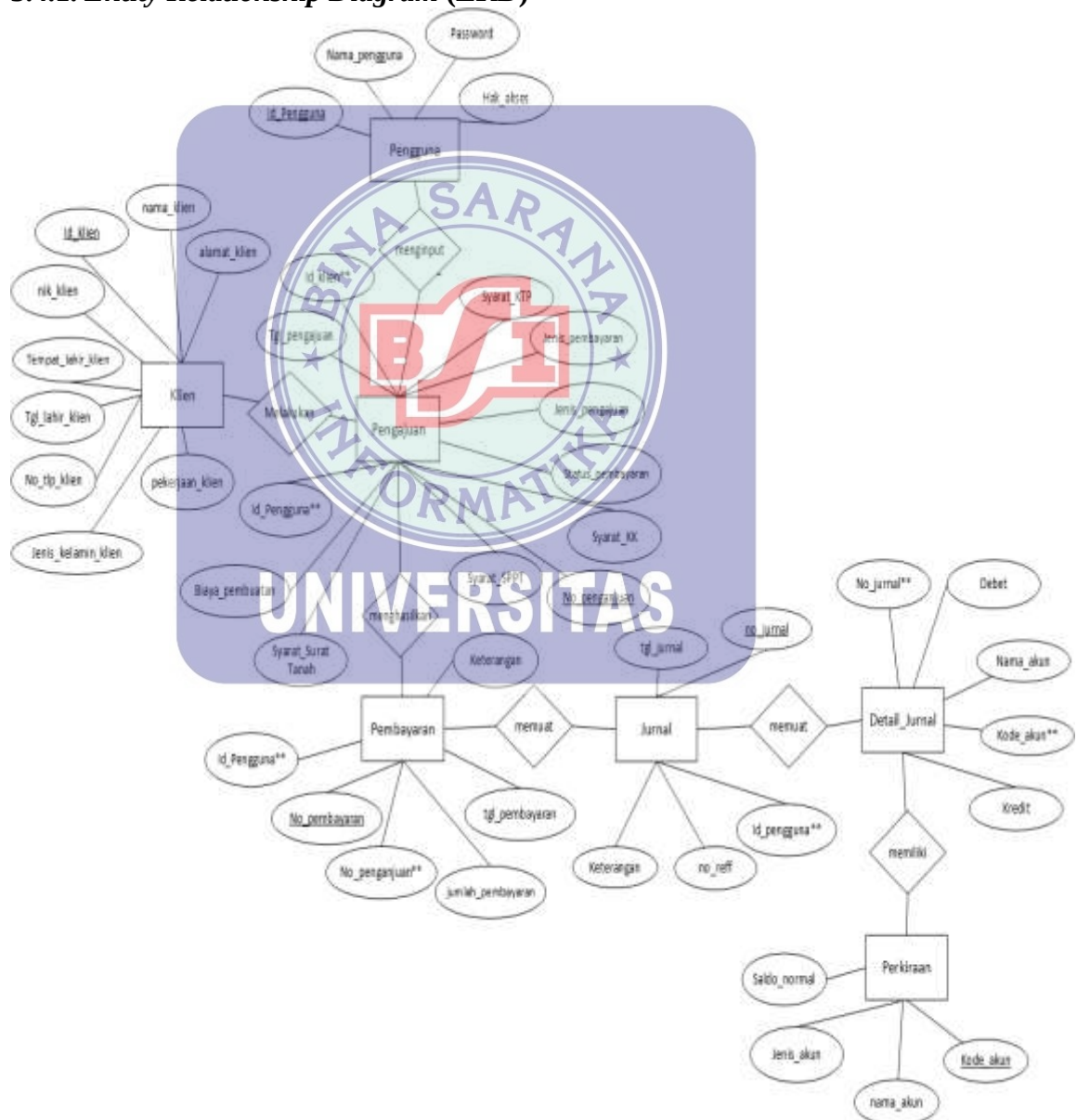
Gambar III.24

Activity Diagram Mengelola Data Perkiraan

3.4. Desain

Pada Desain, berisi penggambaran lengkap *Entity Relationship Diagram* (ERD), *Logical Record Structure* (LRS), Spesifikasi *File*, *Sequence Diagram*, *Deployment Diagram* dan *User Interface* yang penulis rancang sesuai Analisis Kebutuhan Software, yaitu sebagai berikut.

3.4.1. Entity Relationship Diagram (ERD)

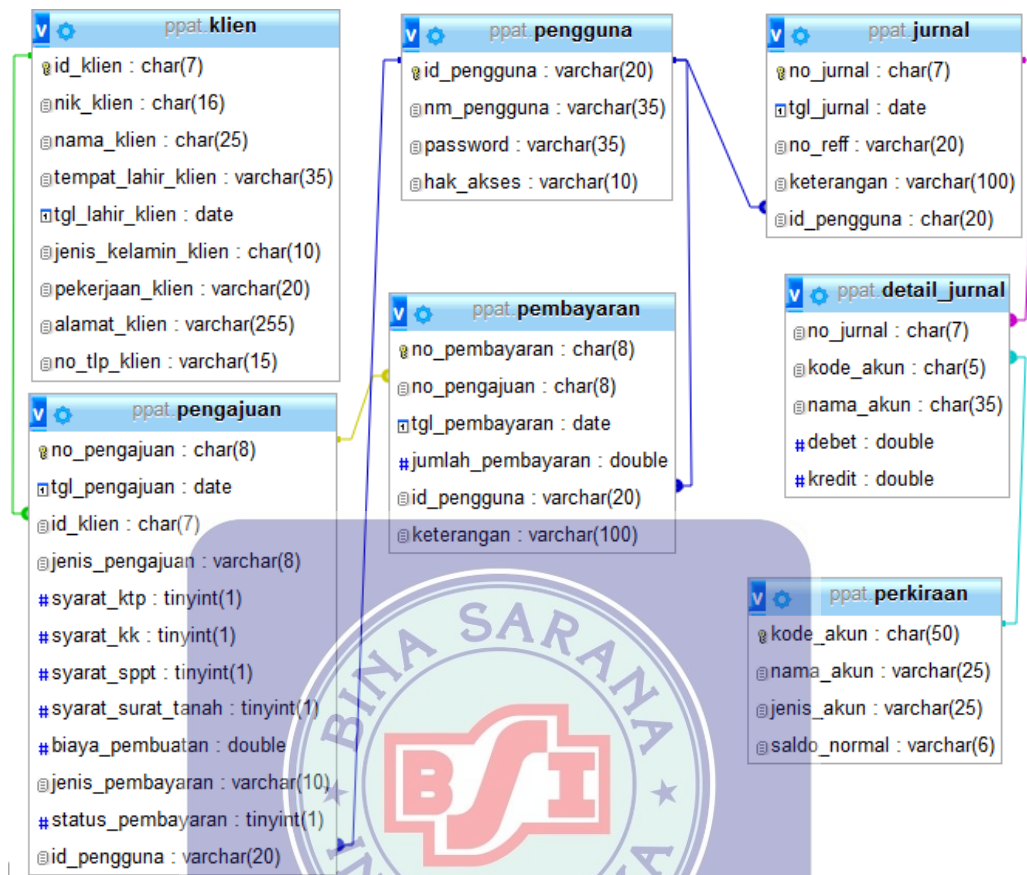


Sumber: Hasil Penelitian (2019).

Gambar III.25

Entity Relationship Diagram (ERD)

3.4.2. Logical Record Structure (LRS)



Sumber: Hasil Penelitian (2019).

Gambar III.26
UNIVERSITAS
Logical Record Structure (LRS)

3.4.3. Spesifikasi File

1. Spesifikasi File Klien

Nama Database : ppat

Nama File : Klien

Tipe File : File Master

Akses File : Random

Panjang Record : 383 Karakter

Field Key : id_klien

Tabel III.13.

Spesifikasi File Tabel Klien

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1.	Id Klien	id_klien	Char	7	Primary Key
2.	Nama Klien	nama_klien	Char	16	
3.	NIK	nik_klien	Char	25	
4.	Tempat Lahir	tempat_lahir_klien	Varchar	35	
5.	Tanggal Lahir	tgl_lahir_klien	Date	-	
6.	Jenis Kelamin	jenis_kelamin_klien	Char	10	
7.	Pekerjaan	pekerjaan_klien	Varchar	20	
8.	Alamat	nalamat_klien	Varchar	255	
9.	No Telp	no_tlp_klien	Varchar	15	

2. Spesifikasi File Pengguna

Nama Database : ppat

Nama File : Pengguna

Tipe File : File Master

Akses File : Random

Panjang Record : 100 Karakter

Field Key : id_pengguna

Tabel III.14.

Spesifikasi File Tabel Pengguna

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1.	ID Pengguna	id-pengguna	Varchar	20	Primary Key
2.	Nama Pengguna	nama_pengguna	Varchar	35	
3.	Password	Password	Varchar	35	
4.	Hak Akses	hak_akses	Varchar	10	

3. Spesifikasi File Pengajuan

Nama *Database* : ppat

Nama *File* : Pengajuan

Tipe *File* : *File* Transaksi

Akses *File* : *Random*

Panjang *Record* : 58 Karakter

Field Key : no_pengajuan

Tabel III.15.

Spesifikasi *File* Tabel Pengajuan Balik Nama

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1.	No Pengajuan	no_pengajuan	Char	8	<i>Primary Key</i>
2.	Tanggal Pengajuan	tgl_pengajuan	Date	-	
3.	Id Klien	id_klien	Char	7	<i>Foreign Key</i>
4.	Jenis pengajuan	jenis_pengajuan	Varchar	8	
5.	Fotokopi KTP	syarat_ktp	Tinyint	1	
6.	Fotokopi KK	syarat_kk	Tinyint	1	
7.	Fotokopi SPPT	syarat_sppt	Tinyint	1	
8.	Surat Tanah	syarat_surat_tanah	Tinyint	1	
9.	Biaya Pembuatan	biaya_pembuatan	Double	-	
10.	Cara Bayar	jenis_pembayaran	Varchar	10	
11.	Status Pembayaran	status_pembayaran	Tinyint	1	
12.	Id Pengguna	id_pengguna	Varchar	20	<i>Foreign Key</i>

4. Spesifikasi *File* Pembayaran

Nama *Database* : ppat

Nama *File* : Pembayaran

Tipe *File* : *File* transaksi

Akses *File* : *Random*

Panjang *Record* : 136 Karakter

Field Key : no_pembayaran

Tabel III.16.

Spesifikasi *File* Tabel Pembayaran

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1.	No Pembayaran	no_pembayaran	Char	8	<i>Primary Key</i>
2.	No Pengajuan	no_pengajuan	Char	8	<i>Foreign Key</i>
3.	Tanggal Pembayaran	tgl_pembayaran	Date	-	
4.	Jumlah Pembayaran	jumlah_pembayaran	Double	-	
5.	Id Pengguna	id_pengguna	Varchar	20	<i>Foreign Key</i>
6.	Keterangan	keterangan	Varchar	100	

5. Spesifikasi *File* JurnalNama *Database* : ppatNama *File* : jurnalTipe *File* : *File* transaksiAkses *File* : *Random*Panjang *Record* : 177 Karakter*Field Key* : no_jurnal

Tabel III.17.

Spesifikasi *File* Tabel Jurnal

No	Elemen Data	Nama File	Tipe	Size	Keterangan
1.	No Jurnal	no_jurnal	Char	7	<i>Primary Key</i>
2.	Tanggal	tgl_jurnal	Date	-	
3.	No Reff	no_reff	Varchar	20	
4.	Keterangan	keterangan	Varchar	100	
5.	Id Pengguna	id_pengguna	Varchar	50	<i>Foreign Key</i>

6. Spesifikasi *File* Detail Jurnal

Nama *Database* : ppat

Nama *File* : detail_jurnal

Tipe *File* : *File* transaksi

Akses *File* : *Random*

Panjang *Record* : 177 Karakter

Field Key : -

Tabel III.18.

Spesifikasi *File* Tabel Detail Jurnal

No	Elemen Data	Nama File	Tipe	Size	Keterangan
1.	No Jurnal	no_jurnal	Char	7	<i>Foreign Key</i>
2.	Kode Akun	kode_akun	Char	5	<i>Foreign Key</i>
3.	Nama Akun	nama_akun	Char	35	
4.	Debet	debet	Double	-	
5.	Kredit	kredit	Double	-	

7. Spesifikasi *File* Perkiraan

Nama *Database* : ppat

Nama *File* : Perkiraan

Tipe *File* : *File* Master

Akses *File* : *Random*

Panjang *Record* : 106 Karakter

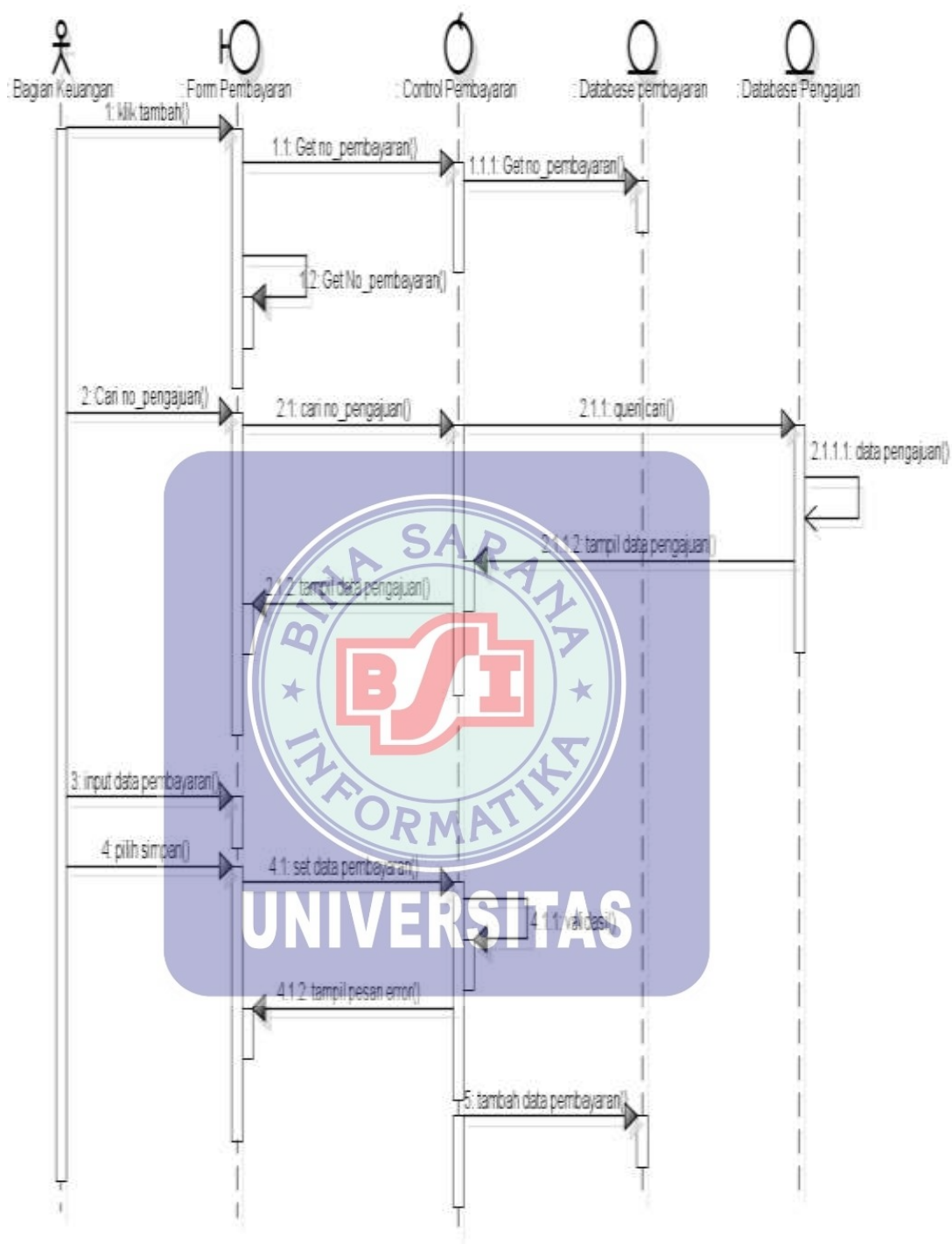
Field Key : kode_akun

Tabel III.19.

Spesifikasi *File* Tabel Perkiraan

No	Elemen Data	Akronim	Tipe	Panjang	Keterangan
1.	Kode Akun	kode_akun	Char	50	<i>Primary Key</i>
2.	Nama Akun	nama_akun	Varchar	25	
3.	Klasifikasi	jenis_akun	Varchar	25	
4.	Saldo Normal	saldo_normal	Varchar	6	

3.4.4. Sequence Diagram

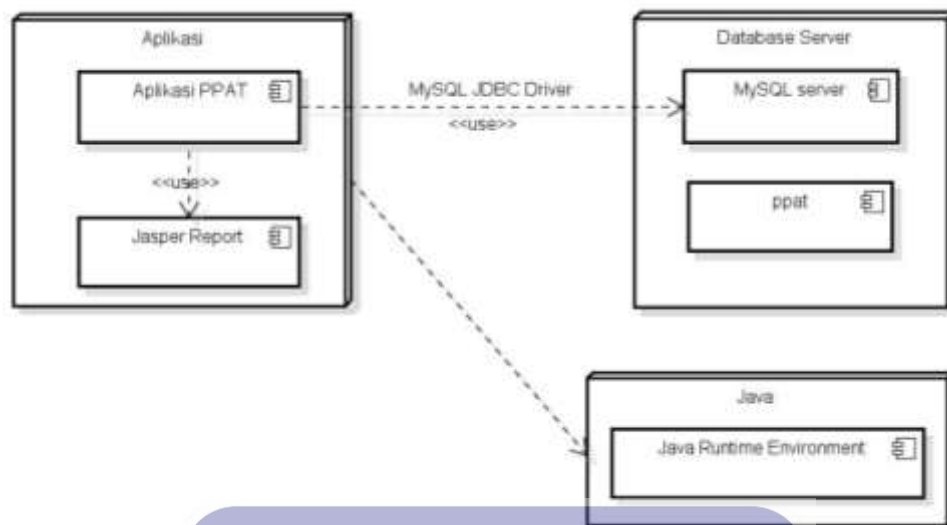


Sumber: Hasil Penelitian (2019).

Gambar III.27

Sequence Diagram Pembayaran

3.4.5. Deployment Diagram



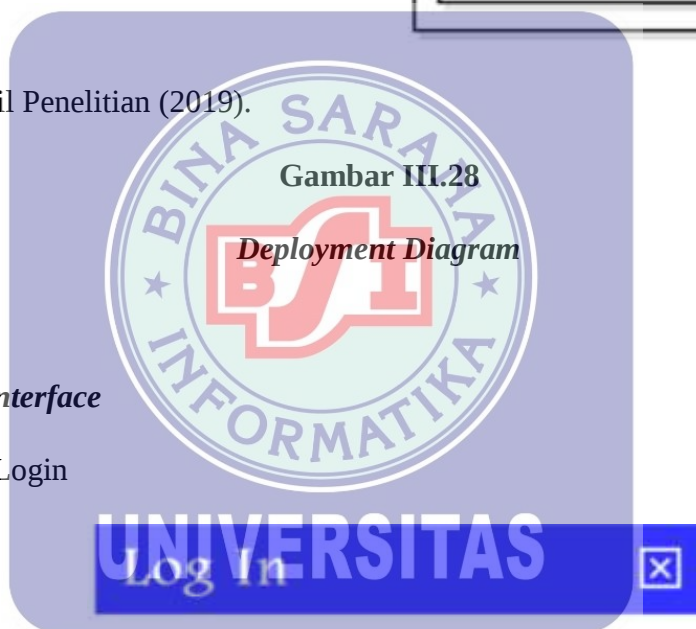
Sumber: Hasil Penelitian (2019).

Gambar III.28

Deployment Diagram

3.4.6. User Interface

1. Halaman Login



Masukan Id Pengguna dan Password anda

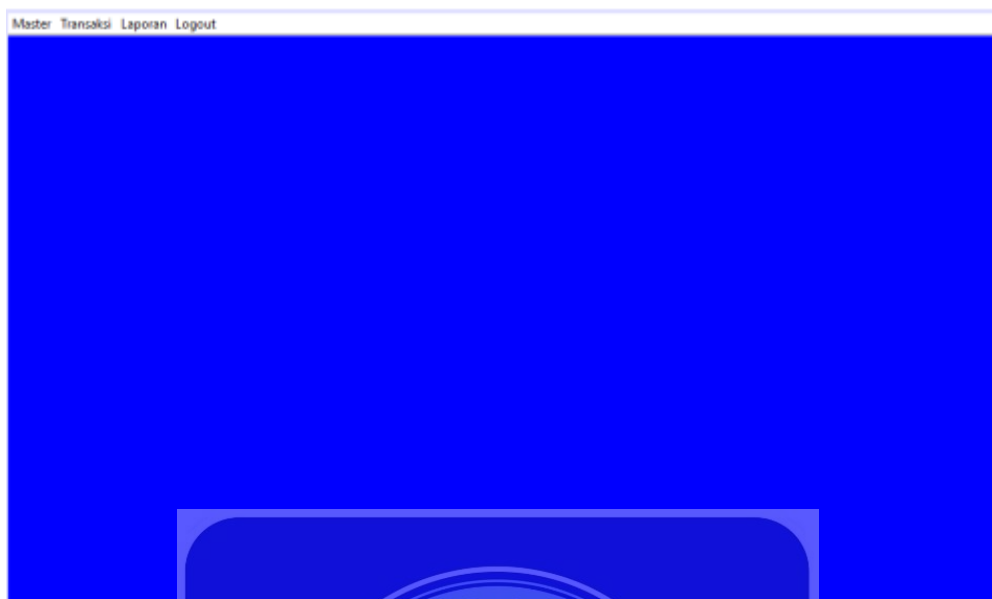
Id Pengguna	
Password	
Log In Keluar	

Sumber: Hasil Penelitian (2019).

Gambar III.29

User Interface halaman Login

2. Halaman Menu Utama



Sumber: Hasil Penelitian (2019).

Gambar III.30
User Interface halaman Menu Utama

3. Halaman Data Klien

Data Klien	
Input / Edit	Daftar
Id Klien	KL000005
Nama Klien	
NIK	
Tempat Lahir	
Tanggal Lahir	
Jenis Kelamin	Laki-laki
Pekerjaan	
Alamat	
No Telp	
Tambah Simpan Ubat Batal Batal Batal Batal	

Sumber: Hasil Penelitian (2019).

Gambar III.31

User Interface halaman Data Klien

4. Halaman Data Pengguna

ID Pengguna	Nama Pengguna	Password	Hak Akses
dida	dida	12345678	Notaris
ikoh3	ikoh	12345678	Kemangan
jun	jun	12345678	Admin

Sumber: Hasil Penelitian (2019).

Gambar III.32
User Interface halaman Data Pengguna

5. Halaman Data Pengajuan Balik Nama

Sumber: Hasil Penelitian (2019).

Gambar III.33
User Interface halaman Data Pengajuan Balik Nama

6. Halaman Pembayaran Jasa Balik Nama

Sumber: Hasil Penelitian (2019).

Gambar III.34

User Interface halaman Pembayaran Jasa Balik Nama

7. Halaman Laporan Jurnal

No Jurnal	Tanggal	No Ref	Keterangan	Kode Akun	Nama Akun	Debet	Kredit
JU00001	2019-07-26	BY00001	Pembayaran Jas.	10001	Kas	Rp0.000.000,00	Rp0,00
				50001	Pendapatan Jasa	Rp0,00	Rp0.000.000,00
JU00002	2019-07-26	BY00002	Pembayaran Jas.	10001	Kas	Rp3.000.000,00	Rp0,00
				40001	Piutang Jasa	Rp3.000.000,00	Rp0,00
				50001	Pendapatan Jasa	Rp0,00	Rp6.000.000,00
JU00003	2019-07-27	BY00003	Pembayaran Jas.	10001	Kas	Rp3.000.000,00	Rp0,00
				40001	Piutang Jasa	Rp0,00	Rp3.000.000,00

Sumber: Hasil Penelitian (2019).

Gambar III.35

User Interface halaman Laporan Jurnal

8. Halaman Laporan Pembayaran

No Pembayaran	Tanggal	No Pengajuan	ID Klien	Nama Klien	Jumlah Pembaya...	Keterangan
BY00001	2019-07-26	PN00002	KL00002	Meirika	5000000.0	Lunas
BY00002	2019-07-26	PN00001	KL00003	MUHAMAD RAFI	3000000.0	DP
BY00003	2019-07-27	PN00001	KL00003	MUHAMAD RAFI	3000000.0	Pelunasan

Sumber: Hasil Penelitian (2019).

Gambar III.36

User Interface halaman Laporan Pembayaran

9. Halaman Data Perkiraan

Kode Akun	Nama Akun	Klasifikasi	Saldo Normal
10001	Kas	Kas	Debet
40001	Pendapatan Jasa	Pendapatan	Kredit
40002	Pinang Jasa	Pinang	Debet

Sumber: Hasil Penelitian (2019).

Gambar III.37

User Interface halaman Data Perkiraan

3.5. Implementasi

Pada tahap ini penulis akan mengimplementasikan hasil dari perancangan sistem yang telah dijelaskan sebelumnya, dengan cara menampilkan *code generation* dan melakukan pengujian terhadap sistem informasi akuntansi pendapatan balik nama pada Kantor Notaris & PPAT MUJTAHID, SH., menggunakan metode *blackbox testing* serta menguraikan spesifikasi sistem komputer *hardware* dan *software*.

3.5.1. Code Generation

```

/*
 * 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.
 */

package forms;

import java.awt.Color;
import java.awt.Dimension;
import java.awt.Font;
import java.awt.Toolkit;
import java.awt.event.KeyEvent;
import java.text.DecimalFormat;
import java.text.NumberFormat;
import java.util.Date;
import java.util.List;
import java.util.ListIterator;
import java.util.Locale;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import model.DetailJurnal;

```

```

import model.Jurnal;
import model.Klien;
import model.Pembayaran;
import model.Pengajuan;
import ppat.PPAT;

public class FormPembayaran extends javax.swing.JFrame {

    String no_pengajuan = "";
    String Status = "";

    Klien klien = new Klien();
    Pengajuan pengajuan = new Pengajuan();
    Pembayaran pembayaran = new Pembayaran();

    DefaultTableModel model = null;

    NumberFormat number = NumberFormat.getCurrencyInstance(new Locale("id",
"ID"));

    NumberFormat nf = new DecimalFormat("#");

    /**
     * Creates new form FormPebayaran
     */
    public FormPembayaran() {

        initComponents();

        Toolkit tk = Toolkit.getDefaultToolkit();
        Dimension d = tk.getScreenSize();
        int x, y;
        x = (int) ((d.getWidth() - getSize().width) / 2);
        y = (int) ((d.getHeight() - getSize().height) / 2);

        setLocation(x, y);

```

```

model = (DefaultTableModel) tblpengajuan.getModel();

tblpengajuan.getTableHeader().setFont(new Font("Andalus", 1, 16));
tblpengajuan.setRowHeight(30);

daftarPembayaran();
aktif();
tabs.setSelectedIndex(1);
this.getContentPane().setBackground(Color.WHITE);
PPAT.flatButton(bttambah);
PPAT.flatButton(btsimpan);
PPAT.flatButton(btncari);

PPAT.flatButton(btbatalkan);
PPAT.flatButton(btcarikan);
PPAT.flatButton(btutup);
}

/**
 * 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() {

    buttonGroup1 = new javax.swing.ButtonGroup();
    bttambah = new javax.swing.JButton();
    btsimpan = new javax.swing.JButton();
    btbatalkan = new javax.swing.JButton();
    btcarikan = new javax.swing.JButton();
    btutup = new javax.swing.JButton();
    tabs = new javax.swing.JTabbedPane();
    jPanel1 = new javax.swing.JPanel();

```

```

tnama_klien = new javax.swing.JTextField();
lbidpengguna = new javax.swing.JLabel();
lbnamapengguna = new javax.swing.JLabel();
tno_pembayaran = new javax.swing.JTextField();
tnik = new javax.swing.JTextField();
lbnamapengguna1 = new javax.swing.JLabel();
lbnamapengguna3 = new javax.swing.JLabel();
ttgl_pembayaran = new com.toedter.calendar.JDateChooser();
tno_pengajuan = new javax.swing.JTextField();
lbidpengguna1 = new javax.swing.JLabel();
tketerangan = new javax.swing.JTextField();
lbnamapengguna10 = new javax.swing.JLabel();

btcari = new javax.swing.JButton();
tbiaya = new javax.swing.JTextField();
lbnamapengguna11 = new javax.swing.JLabel();
tjumlah_bayar = new javax.swing.JTextField();
lbnamapengguna12 = new javax.swing.JLabel();
lbnamapengguna13 = new javax.swing.JLabel();
jPanel2 = new javax.swing.JPanel();
jScrollPane1 = new javax.swing.JScrollPane();
tblpengajuan = new javax.swing.JTable();
btncari = new javax.swing.JButton();
tsampai_tgl1 = new com.toedter.calendar.JDateChooser();
tdari_tgl1 = new com.toedter.calendar.JDateChooser();
jPanel3 = new javax.swing.JPanel();
jLabel1 = new javax.swing.JLabel();

```

```

setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_ON_CLOSE);

```

```

bttambah.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
bttambah.setText("Tambah");
bttambah.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        bttambahActionPerformed(evt);
    }
});

btsimpan.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
btsimpan.setText("Simpan ");
btsimpan.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btsimpanActionPerformed(evt);}
});

btbatal.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
btbatal.setText("Batal");
btbatal.addActionListener(new java.awt.event.ActionListener() {

    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btbatalActionPerformed(evt);
    }
});

bttutup.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
bttutup.setText("Tutup");
bttutup.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        bttutupActionPerformed(evt);
    }
});

tabs.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
jPanel1.setBackground(new java.awt.Color(255, 255, 255));
tnama_klien.setEditable(false);
tnama_klien.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
tnama_klien.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tnama_klienActionPerformed(evt);
    }
});

```

```

    }
});

lbidpengguna.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbidpengguna.setText("No Pembayaran");

lbnamapengguna.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbnamapengguna.setText("Nama Klien");

tno_pembayaran.setEditable(false);
tno_pembayaran.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
tno_pembayaran.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyPressed(java.awt.event.KeyEvent evt) {
        tno_pembayaranKeyPressed(evt);
    }
});
tnik.setEditable(false);
tnik.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
tnik.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tnikActionPerformed(evt);
    }
});

lbnamapengguna1.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbnamapengguna1.setText("NIK");

lbnamapengguna3.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbnamapengguna3.setText("Tanggal Pembayaran");

ttgl_pembayaran.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N

tno_pengajuan.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N

```

```
lbidpengguna1.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbidpengguna1.setText("No Pengajuan");
```

```
tketerangan.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
tketerangan.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tketeranganActionPerformed(evt);
    }
});
```

```
lbnamapengguna10.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbnamapengguna10.setText("Keterangan");
```

```
btcari.setText("...");
btcari.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btcariActionPerformed(evt);
    }
});
```

```
tbiaya.setEditable(false);
tbiaya.setFont(new java.awt.Font("Andalus", 0, 18)); // NOI18N
tbiaya.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tbiayaActionPerformed(evt);
    }
});
```

```
lbnamapengguna11.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbnamapengguna11.setText("Total Biaya");
```

```
tjumlah_bayar.setFont(new java.awt.Font("Andalus", 0, 18)); // NOI18N
tjumlah_bayar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        tjumlah_bayarActionPerformed(evt);
    }
});
```

```

    }
});

lbnamapengguna12.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
lbnamapengguna12.setText("Jumlah Pembayaran");

lbnamapengguna13.setFont(new java.awt.Font("Andalus", 1, 24)); // NOI18N
lbnamapengguna13.setText("Pembayaran ");

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)

.addGroup(jPanel1Layout.createSequentialGroup()
.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createSequentialGroup()

.addComponent(lbidpengguna)
.addComponent(lbnamapengguna)
.addComponent(lbnamapengguna1)
.addComponent(lbnamapengguna3)
.addComponent(lbidpengguna1)
.addComponent(lbnamapengguna10)
.addComponent(lbnamapengguna11)
.addComponent(lbnamapengguna12))

```

`.addGap(24, 24, 24)`

`.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)`

`.addComponent(tnama_klien,
javax.swing.GroupLayout.PREFERRED_SIZE, 301,
javax.swing.GroupLayout.PREFERRED_SIZE)`

`.addComponent(tnik,
javax.swing.GroupLayout.PREFERRED_SIZE, 186,
javax.swing.GroupLayout.PREFERRED_SIZE)`

`.addComponent(tketerangan,
javax.swing.GroupLayout.PREFERRED_SIZE, 301,
javax.swing.GroupLayout.PREFERRED_SIZE)`

`.addComponent(tno_pembayaran,
javax.swing.GroupLayout.PREFERRED_SIZE, 129,
javax.swing.GroupLayout.PREFERRED_SIZE)`

`.addComponent(tbiaya,
javax.swing.GroupLayout.PREFERRED_SIZE, 301,
javax.swing.GroupLayout.PREFERRED_SIZE)`

`.addComponent(tjumlah_bayar,
javax.swing.GroupLayout.PREFERRED_SIZE, 301,
javax.swing.GroupLayout.PREFERRED_SIZE)`

`.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)`

`.addGroup(jPanel1Layout.createSequentialGroup()
.addComponent(tno_pengajuan,
javax.swing.GroupLayout.PREFERRED_SIZE, 129,
javax.swing.GroupLayout.PREFERRED_SIZE)`

`.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)`

`.addComponent(btccari,
javax.swing.GroupLayout.PREFERRED_SIZE, 0, Short.MAX_VALUE))`

```

        .addComponent(ttgl_pembayaran,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addComponent(lbnamapengguna13))
        .addContainerGap(515, Short.MAX_VALUE))
    );

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)

    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(13, 13, 13)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.BASELINE)
            .addComponent(tno_pembayaran,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(lbidpengguna,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.CENTER)
            .addComponent(btcari, javax.swing.GroupLayout.PREFERRED_SIZE,
30, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(tno_pengajuan,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(lbidpengguna1,
        javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE))

```

30,

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

            .addComponent(ttgl_pembayaran,
            javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)

```

30,

```

            .addComponent(lbnamapengguna3,
            javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE))

```

30,

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

```

```

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

            .addComponent(lbnamapengguna,
            javax.swing.GroupLayout.Alignment.TRAILING,
            javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)

```

30,

```

            .addComponent(tnama_klien,
            javax.swing.GroupLayout.Alignment.TRAILING,
            javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE))

```

30,

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

```

```

        .addComponent(lbnamapengguna1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(tnik, javax.swing.GroupLayout.PREFERRED_SIZE, 30,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(40, 40, 40)
        .addComponent(lbnamapengguna13,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.BASELINE)
        .addComponent(tbiaya, javax.swing.GroupLayout.PREFERRED_SIZE,
30, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(lbnamapengguna11,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.BASELINE)
        .addComponent(tjumlah_bayar,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(lbnamapengguna12,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.BASELINE)

```

```

        .addComponent(tketerangan,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(lbnamapengguna10,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(331, 331, 331))
    );

    tabs.addTab("Input / Edit", jPanel1);

    jPanel2.setBackground(new java.awt.Color(255, 255, 255));

    tblpengajuan.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
    tblpengajuan.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 [] {
            "No Pembayaran", "Tanggal", "No Pengajuan", "ID Klien", "Nama Klien",
            "Jumlah Pembayaran", "Keterangan"
        }
    ));

    boolean[] canEdit = new boolean [] {
        false, false, false, false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }

```

```

});
tblpengajuan.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        tblpengajuanMouseClicked(evt);
    }
});
jScrollPane1.setViewportView(tblpengajuan);

btncari.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
btncari.setText("Cari");
btncari.setPreferredSize(new java.awt.Dimension(49, 26));
btncari.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btncariActionPerformed(evt);
    }
});

tsampai_tgl1.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
tdari_tgl1.setFont(new java.awt.Font("Andalus", 0, 16)); // NOI18N
javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addGap(11, 11, 11)
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 969, Short.MAX_VALUE))

```

```

        .addGroup(jPanel2Layout.createSequentialGroup()
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
                Short.MAX_VALUE)
            .addComponent(tdari_tgl1,
                javax.swing.GroupLayout.PREFERRED_SIZE,                201,
                javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(tsampai_tgl1,
                javax.swing.GroupLayout.PREFERRED_SIZE,                201,
                javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(btncari,
                javax.swing.GroupLayout.PREFERRED_SIZE,                86,
                javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap())

    );
    jPanel2Layout.setVerticalGroup(
        jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
    )

        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
            jPanel2Layout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
                    Short.MAX_VALUE)

        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
            t.LEADING, false)
            .addComponent(tsampai_tgl1,
                javax.swing.GroupLayout.Alignment.TRAILING,
                javax.swing.GroupLayout.DEFAULT_SIZE, 31, Short.MAX_VALUE)

```

```

        .addComponent(tdari_tgl1,
        javax.swing.GroupLayout.Alignment.TRAILING,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(btncari,
        javax.swing.GroupLayout.Alignment.TRAILING,
        javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE))
    );

    tabs.addTab("Data", jPanel2);

    jPanel3.setBackground(new java.awt.Color(102, 102, 255));

    jLabel1.setFont(new java.awt.Font("Andalus", 1, 36)); // NOI18N
    jLabel1.setForeground(new java.awt.Color(255, 255, 255));
    jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

    jLabel1.setText("Pembayaran Jasa Balik Nama");
    javax.swing.GroupLayout jPanel3Layout = new
    javax.swing.GroupLayout(jPanel3);
    jPanel3.setLayout(jPanel3Layout);
    jPanel3Layout.setHorizontalGroup(
        jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
        jPanel3Layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addContainerGap())
    );

```

```

        .addContainerGap()
        .addComponent(jLabel1,      javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addContainerGap()
    );
    jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING
)

        .addGroup(jPanel3Layout.createSequentialGroup())
        .addContainerGap()
        .addComponent(jLabel1,      javax.swing.GroupLayout.PREFERRED_SIZE,
51, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new
    javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addComponent(tabs)
                    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                        .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                        .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    )
                )
            )
        )
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(tabs)
            .addContainerGap()
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addComponent(jLabel2, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addContainerGap()
                    .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                )
            )
        )
    );
    getContentPane().setLayout(layout);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setTitle("GUI");
    pack();
    setVisible(true);
}

}

```

```

        .addComponent(btambah,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(btsimpan,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addComponent(btbatall,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(62, 62, 62)
        .addComponent(bttutup,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap()
        .addComponent(jPanel3, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
        .addComponent(jPanel3,javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(tabs, javax.swing.GroupLayout.PREFERRED_SIZE, 470,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, Short.MAX_VALUE)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASEL
INE)

```

```

        .addComponent(bttambah,
javafx.swing.GroupLayout.PREFERRED_SIZE,
javafx.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btsimpan,
javafx.swing.GroupLayout.PREFERRED_SIZE,
javafx.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btbatalkan, javafx.swing.GroupLayout.PREFERRED_SIZE,
45, javafx.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(bttutup, javafx.swing.GroupLayout.PREFERRED_SIZE,
45, javafx.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap()
    );

    pack();
} // </editor-fold>

private void aktif() {
    btbatalkan.setEnabled(!"".equals(this.Status));
    bttambah.setEnabled(!"".equals(this.Status));
    btsimpan.setEnabled(!"".equals(this.Status));

    tabs.setEnabledAt(0, !"".equals(this.Status));
    tabs.setEnabledAt(1, !"".equals(this.Status));

    try {
        tjumlah_bayar.setEditable(pengajuan.getJenis_pembayaran().equals("Dengan
DP"));
    } catch (Exception ex) {
    }

}

private void bersihForm() {
    tno_pengajuan.setText("");
    ttgl_pembayaran.setDate(null);
    tno_pembayaran.setText("");

```

```

    tnik.setText("");
    tnama_klien.setText("");
    tbiaya.setText("");
    tjumlah_bayar.setText("");
    tketerangan.setText("");
}

private void setData() {
    pembayaran.setNo_pembayaran(tno_pembayaran.getText());
    pembayaran.setNo_pengajuan(tno_pengajuan.getText());
    pembayaran.setTgl_pembayaran(new
java.sql.Date(ttgl_pembayaran.getDate().getTime()));
    pembayaran.setId_pengguna(PPAT.id_pengguna);
    pembayaran.setKeterangan(tketerangan.getText());
    pembayaran.setJumlah_pembayaran(Double.parseDouble(tjumlah_bayar.getText()));
;
}

private void daftarPembayaran() {
    List<Pembayaran> lp = pembayaran.daftarPembayaran();
    ListIterator<Pembayaran> lip = lp.listIterator();

    model.setRowCount(0);
    tblpengajuan.repaint();
    while (lip.hasNext()) {
        pembayaran = (Pembayaran) lip.next();
        pengajuan = pengajuan.cariNoPengajuan(pembayaran.getNo_pengajuan());
        klien = klien.cariIdKlien(pengajuan.getId_klien());
        Object[] data = {

            pembayaran.getNo_pembayaran(),
            pembayaran.getTgl_pembayaran(),
            pembayaran.getNo_pengajuan(),
            pengajuan.getId_klien(),

```

```

        klien.getNama_klien(),
        pembayaran.getJumlah_pembayaran(),
        pembayaran.getKeterangan()

    };
    model.addRow(data);
}
}

```

```

private void daftarPembayaran(Date tgl1, Date tgl2) {
    List<Pembayaran> lp = pembayaran.cariPembayaranPeriode(new
java.sql.Date(tgl1.getTime()), new java.sql.Date(tgl2.getTime()));
    ListIterator<Pembayaran> lip = lp.listIterator();
    model.setRowCount(0);
    tblpengajuan.repaint();

    while (lip.hasNext()) {
        pembayaran = (Pembayaran) lip.next();
        pengajuan = pengajuan.cariNoPengajuan(pembayaran.getNo_pengajuan());
        klien = klien.cariIdKlien(pengajuan.getId_klien());
        Object[] data = {
            pembayaran.getNo_pembayaran(),
            pembayaran.getTgl_pembayaran(),
            pembayaran.getNo_pengajuan(),
            pengajuan.getId_klien(),
            klien.getNama_klien(),
            pembayaran.getJumlah_pembayaran(),
            pembayaran.getKeterangan()
        };
        model.addRow(data);
    }
}

```

```

    }
}

private void cariPengajuan(String NoPengajuan) {
    tnama_klien.setText("");
    tnik.setText("");
    tbiaya.setText("");
    tketerangan.setText("");

    System.out.println(NoPengajuan);

    if (!"".equals(tno_pengajuan.getText())) {
        pengajuan = pengajuan.cariNoPengajuan(NoPengajuan);
        System.out.println(pengajuan.getId_klien());
        if (pengajuan.getId_klien() != null) {
            if (pengajuan.getStatus_pembayaran() == 0) {
                klien = klien.cariIdKlien(pengajuan.getId_klien());
                tnik.setText(klien.getNik_klien());
                tnama_klien.setText(klien.getNama_klien());
                tbiaya.setText(nf.format(pengajuan.getBiaya_pembuatan()));
                if ("Tunai".equals(pengajuan.getJenis_pembayaran())) {
                    tjumlah_bayar.setText(nf.format(pengajuan.getBiaya_pembuatan()));
                } else {
                    double total =
pembayaran.totalPembayaran(tno_pengajuan.getText());
                    tjumlah_bayar.setText(nf.format(pengajuan.getBiaya_pembuatan() -
total));}
                } else {
                    tno_pengajuan.setText("");
                    JOptionPane.showMessageDialog(rootPane, "Pengajuan yang Anda cari
sudah dilakukan pebayaran dengan Lunas!", "Info",
JOptionPane.INFORMATION_MESSAGE);
                }
            } else {

```

```

        tno_pengajuan.setText("");
        JOptionPane.showMessageDialog(rootPane, "Pengajuan yang Anda cari
tidak ada!", "Error", JOptionPane.ERROR_MESSAGE);
    }
}
aktif();
}

```

```

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

```

```

private void bttambahActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here
    this.Status = "Tambah";
    this.no_pengajuan = "";
    bersihForm();
    tno_pembayaran.setText(pembayaran.noAuto());
    aktif();
    tabs.setSelectedIndex(0);
}

```

```

private void btbatalActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    this.Status = "";
    this.no_pengajuan = "";
    bersihForm();
    daftarPembayaran();
    aktif();
    tabs.setSelectedIndex(1);
}

```

```

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

```

```

}

private void tblpengajuanMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    int row = tblpengajuan.getSelectedRow();
    this.no_pengajuan = (String) model.getValueAt(row, 0);
    aktif();
}

private void btsimpanActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if ("".equals(tno_pembayaran.getText())
        || "".equals(tnama_klien.getText())
        || ttgl_pembayaran.getDate() == null
        || "".equals(tketerangan.getText())
        || "".equals(tjumlah_bayar.getText())) {
        JOptionPane.showMessageDialog(rootPane, "Mohon isi data dulu", "info",
JOptionPane.ERROR_MESSAGE);
        return;
    }

    if (!PPAT.periksaInput(tjumlah_bayar.getText(), "[0-9]{2,19}$")) {
        JOptionPane.showMessageDialog(rootPane, "Biaya pembuatan hanya boleh
berupa angka", "Info", JOptionPane.WARNING_MESSAGE);
        return;
    }

    setData();
    if ("Tambah".equals(Status)) {
        if (pembayaran.tambahPembayaran() > 0) {
            if
                (pengajuan.getBiaya_pembuatan()
                    -
pembayaran.totalPembayaran(tno_pengajuan.getText()) == 0) {
                pengajuan = pengajuan.cariNoPengajuan(tno_pengajuan.getText());
                pengajuan.setStatus_pembayaran(1);
            }
        }
    }
}

```

```

        pengajuan.ubahPengajuan(tno_pengajuan.getText());
    }

    Jurnal jurnal = new Jurnal();
    DetailJurnal detail = new DetailJurnal();

    System.out.println(pengajuan.getJenis_pembayaran());

    String No = jurnal.noAuto();
    if ("Dengan DP".equalsIgnoreCase(pengajuan.getJenis_pembayaran())) {
        //Jurnal Piutang Pendapatan Jasa

        jurnal.setNo_jurnal(No);
        jurnal.setTgl_jurnal(new java.sql.Date(new java.util.Date().getTime()));
        jurnal.setNo_reff(tno_pembayaran.getText());
        jurnal.setId_pengguna(PPAT.id_pengguna);
        jurnal.setKeterangan("Pembayaran Jasa Tn/Nyoya " +
klien.getNama_klien());

        jurnal.tambahJurnal();

        if (pembayaran.totalPembayaran(tno_pengajuan.getText()) <
pengajuan.getBiaya_pembuatan()) {
            if (pembayaran.banyakPembayaran(tno_pengajuan.getText()) <= 1) {

                double piutang = pengajuan.getBiaya_pembuatan() –
Double.parseDouble(tjumlah_bayar.getText());
                detail.setNo_jurnal(No);
                detail.setKode_akun("10001");
                detail.setNama_akun("Kas");
                detail.setDebet(Double.parseDouble(tjumlah_bayar.getText()));
                detail.setKredit(0.0);

                detail.tambahDetailJurnal();
                detail.setNo_jurnal(No);

```

```

        detail.setKode_akun("40001");
        detail.setNama_akun("Piutang Jasa");
        detail.setDebet(piutang);
        detail.setKredit(0.0);

        detail.tambahDetailJurnal();
        detail.setNo_jurnal(No);
        detail.setKode_akun("50001");
        detail.setNama_akun("Pendapatan Jasa");
        detail.setDebet(0.0);
        detail.setKredit(pengajuan.getBiaya_pembuatan());
        detail.tambahDetailJurnal();
    }
} else {
    detail.setNo_jurnal(No);
    detail.setKode_akun("10001");
    detail.setNama_akun("Kas");
    detail.setDebet(Double.parseDouble(tjumlah_bayar.getText()));
    detail.setKredit(0.0);

    detail.tambahDetailJurnal();
    detail.setNo_jurnal(No);
    detail.setKode_akun("40001");
    detail.setNama_akun("Piutang Jasa");
    detail.setDebet(0.0);
    detail.setKredit(Double.parseDouble(tjumlah_bayar.getText()));
    detail.tambahDetailJurnal();}

} else {
    jurnal.setNo_jurnal(No);
    jurnal.setTgl_jurnal(new java.sql.Date(new java.util.Date().getTime()));
    jurnal.setNo_reff(tno_pembayaran.getText());
    jurnal.setId_pengguna(PPAT.id_pengguna);

```

```
jurnal.setKeterangan("Pembayaran Jasa Tn/Nyoya " +
klien.getNama_klien());
```

```
jurnal.tambahJurnal();
detail.setNo_jurnal(No);
detail.setKode_akun("10001");
detail.setNama_akun("Kas");
detail.setDebet(Double.parseDouble(tjumlah_bayar.getText()));
detail.setKredit(0.0);
```

```
detail.tambahDetailJurnal();
detail.setNo_jurnal(No);
detail.setKode_akun("50001");
detail.setNama_akun("Pendapatan Jasa");
detail.setDebet(0.0);
detail.setKredit(Double.parseDouble(tjumlah_bayar.getText()));
detail.tambahDetailJurnal();
```

```
}
```

```
JOptionPane.showMessageDialog(rootPane, "Data berhasil disimpan",
"info", JOptionPane.INFORMATION_MESSAGE);
```

```
} else {
```

```
JOptionPane.showMessageDialog(rootPane, "Data gagal disimpan\n" +
pembayaran.getPesanError(), "info", JOptionPane.ERROR_MESSAGE);
```

```
}
```

```
}
```

```
this.no_pengajuan = "";
```

```
this.Status = "";
```

```
daftarPembayaran();
```

```
bersihForm();
```

```
aktif();
```

```

    tabs.setSelectedIndex(1);
}

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

private void tno_pembayaranKeyPressed(java.awt.event.KeyEvent evt) {
    // TODO add your handling code here:
    if (evt.getKeyCode() == KeyEvent.VK_ENTER) {
        cariPengajuan(tno_pembayaran.getText());
    }
}

private void btcariActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    tno_pengajuan.setText(new FormCariPengajuan(this, true).cariPengajuan());
    cariPengajuan(tno_pengajuan.getText());
}

private void btncariActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if (tdari_tgl1.getDate() == null || tsampai_tgl1.getDate() == null) {
        daftarPembayaran();
    } else {
        daftarPembayaran(tdari_tgl1.getDate(), tsampai_tgl1.getDate());
    }
}

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

private void tbiayaActionPerformed(java.awt.event.ActionEvent evt) {

```

```

    // TODO add your handling code here:
}

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

/**
 * @param args the command line arguments
 */

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 ("Metal".equals(info.getName())) {

                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

```

```

private javax.swing.ButtonGroup buttonGroup1;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JLabel lblidpengguna;
private javax.swing.JLabel lblidpengguna1;
private javax.swing.JLabel lblnamapengguna;
private javax.swing.JLabel lblnamapengguna1;
private javax.swing.JLabel lblnamapengguna10;
private javax.swing.JLabel lblnamapengguna11;
private javax.swing.JLabel lblnamapengguna12;
private javax.swing.JLabel lblnamapengguna13;
private javax.swing.JLabel lblnamapengguna3;
private javax.swing.JTabbedPane tabs;
private javax.swing.JTextField tbiaya;
private javax.swing.JTable tblpengajuan;
private com.toedter.calendar.JDateChooser tdari_tgl1;
private javax.swing.JTextField tjumlah_bayar;
private javax.swing.JTextField tketerangan;
private javax.swing.JTextField tnama_klien;
private javax.swing.JTextField tnik;
private javax.swing.JTextField tno_pembayaran;
private javax.swing.JTextField tno_pengajuan;
private com.toedter.calendar.JDateChooser tsampai_tgl1;
private com.toedter.calendar.JDateChooser ttgl_pembayaran;

// End of variables declaration
}

```

3.5.2. Blackbox Testing

1. Pengujian Terhadap *Form Login*

Tabel III.20.

Hasil Pengujian Black Box Testing Halaman Login

No.	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	<i>Id</i> Pengguna dan Password tidak diisi, kemudian klik tombol <i>login</i>	<i>Id</i> Pengguna: (kosong) <i>password</i> : (kosong)	Sistem akan menolak dan menampilkan “ <i>Id</i> Pengguna dan Password tidak boleh kosong”	Sesuai Harapan	Valid
2	<i>Id</i> Pengguna diisi benar dan <i>password</i> diisi salah, kemudian klik tombol <i>login</i>	<i>Id</i> Pengguna: (ikoh) <i>password</i> : (12345678)	Sistem akan menolak dan menampilkan “ <i>password</i> keliru”	Sesuai Harapan	Valid
3	<i>Id</i> Pengguna diisi salah dan <i>password</i> diisi benar, kemudian klik tombol <i>login</i>	<i>Id</i> Pengguna: (ikoh3) <i>password</i> : (150698)	Sistem akan menolak dan menampilkan “ <i>Id</i> Pengguna keliru”	Sesuai Harapan	Valid
4	Menginputkan data <i>login</i> yang benar lalu klik tombol <i>login</i>	<i>Id</i> Pengguna: (mujtahid) <i>password</i> : (12345678) benar	Sistem menerima akses <i>login</i> kemudian masuk sebagai notaris	Sesuai Harapan	Valid

2. Pengujian Terhadap Halaman Data Klien

Tabel III.21.

Hasil Pengujian Black Box Testing Halaman Data Klien

No.	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Klik tambah, form tidak diisi lalu klik simpan	Form data Klien (kosong)	Sistem akan menolak dan menampilkan informasi: “Mohon isi data dulu”	Sesuai Harapan	Valid
2	Klik tambah, Nama Klien diisi angka lalu klik simpan	Nama Klien (1234567)	Sistem akan menolak dan menampilkan informasi: “Nama klien hanya boleh berupa alfabet”	Sesuai Harapan	Valid
3	Klik tambah, NIK diisi alphabet atau angka kurang dari 16 digit lalu klik simpan	NIK (dida atau 987654)	Sistem akan menolak dan menampilkan informasi: “NIK hanya boleh angka 16 digit”	Sesuai Harapan	Valid
4	Klik tambah, Tempat Lahir diisi angka, lalu klik simpan	Tempat Lahir: (654321)	Sistem akan menolak dan menampilkan informasi: “Tempat lahir hanya boleh berupa alfabet”	Sesuai Harapan	Valid
5	Klik tambah, No telp diisi alphabet atau angka kurang dari 8 digit lalu klik simpan	No telp (jun atau 0986 8901)	Sistem akan menolak dan menampilkan informasi: “No telp hanya boleh angka min 8 digit max 15 digit”	Sesuai Harapan	Valid
6	Menginputkan data klien yang benar lalu klik tombol simpan	Tanggal pengajuan: (12 Jul 19) biaya Pembuatan , dan sebagainya	Sistem menerima dan menampilkan “data pengajuan balik nama kemudian masuk sebagai admin	Sesuai Harapan	Valid

3. Pengujian Terhadap Halaman Pembayaran Jasa Balik Nama

Tabel III.22.

Hasil Pengujian *Black Box Testing* Halaman Pembayaran Jasa Balik Nama

No.	Skenario Pengujian	Test Case	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Klik tambah, form tidak diisi lalu klik simpan	Form pembayaran (kosong)	Sistem akan menolak dan menampilkan informasi: "Mohon isi data dulu"	Sesuai Harapan	Valid
2	Klik tambah, jumlah Pembayaran diisi alfabet lalu klik simpan	Jumlah Pembayaran (kiesiah)	Sistem akan menolak dan menampilkan informasi: "Biaya pembuatan hanya boleh berupa angka"	Sesuai Harapan	Valid
3	Klik tambah, No pengajuan tidak dipilih lalu klik pilih	No pengajuan (Kosong)	Sistem akan menolak dan menampilkan informasi: "Belum ada data yang dipilih"	Sesuai Harapan	Valid
4	Menginputkan Pembayaran yang benar lalu klik tombol <i>simpan</i>	No pengajuan: (PN003) dan seterusnya	Sistem menerima dan menampilkan "data pembayaran balik nama."	Sesuai Harapan	Valid

3.5.3. Spesifikasi *Hardware* dan *Software*

Tabel III.23.

Spesifikasi *Hardware* dan *Software*

Kebutuhan	Keterangan
Sistem	<i>Windows 7 32-bit atau 64-bit atau di atasnya</i>
<i>Processor</i>	<i>Intel Core 2 Duo @1.9GHz</i>
RAM	2 GB
<i>Harddisk</i>	2 GB (instalasi Java dan Program)
<i>Monitor</i>	14" LED
<i>Keyboard</i>	102 Key
<i>Printer</i>	<i>INKJET</i>
<i>Mouse</i>	<i>Standard</i>
<i>Software</i>	Kebutuhan menjalankan aplikasi: Editor : NetBeans IDE 8.1 Web Server : Xampp v.3.2.1 DBMS : MySQL