BAB IV

ANALISIS DAN PERANCANGAN SISTEM

4.1 Analisis Sistem yang Sedang Berjalan

Analisis merupakan suatu kegiatan yang bertujuan untuk mempelajari serta mengevaluasi suatu bentuk permasalahan yang ada pada sebuah sistem. Dalam analisis sistem akan ditemukan masalah yang mungkin akan mempengaruhi kerja sistem. Agar sistem yang dirancang dapat berjalan sebagaimana mestinya, perlu dilakukan analisis terhadap kinerja sistem yang pada akhirnya bertujuan untuk pengembangan sistem.

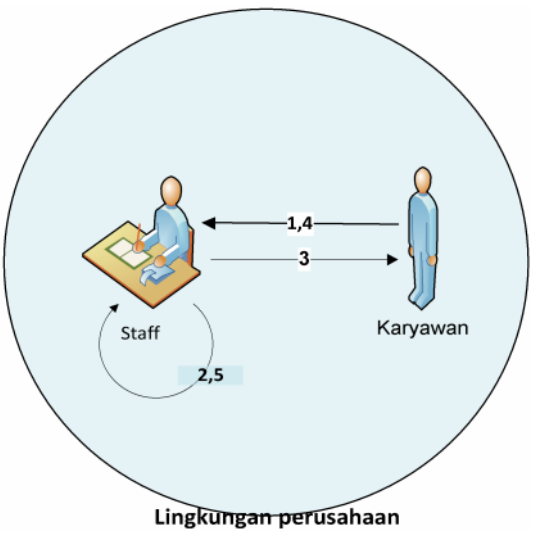
Adapun sistem yang sedang berjalan pada Option Cell sampai saat ini yaitu sistem informasi pengolahan data penggajian dan absensi yang berjalan pada perusahaan tersebut masih dilakukan secara manual. Pengarsipan data karyawan, pengarsipan daftar absensi kehadiran, hingga pada proses perhitungan gaji masih dilakukan pencatatan manual dengan proses penghitungan menggunakan Microsoft excel.

4.1.1 Analisis Prosedur yang Sedang Berjalan

Analisis prosedur merupakan proses analisa terhadap kegiatan-kegiatan yang dilakukan oleh Option Cell proses absensi dan perhitungan gaji. Analisis terhadap prosedur yang sedang berjalan hendaknya perlu dilakukan terlebih dahulu sebelum dilakukannya perancangan terhadap sistem informasi yang akan diusulkan. Harus diketahui pula hal-hal yang menjadi tujuan pemakai sehingga masalah tersebut dapat didefinisikan secara jelas.

4.1.1.1 Proses Bisnis Berjalan

Proses bisnis adalah suatu kumpulan aktivitas atau pekerjaan terstruktur yang saling terkait untuk menyelesaikan suatu masalah tertentu atau yang menghasilkan produk atau layanan. Proses bisnis absensi yang sedang berjalan adalah sebagai berikut :

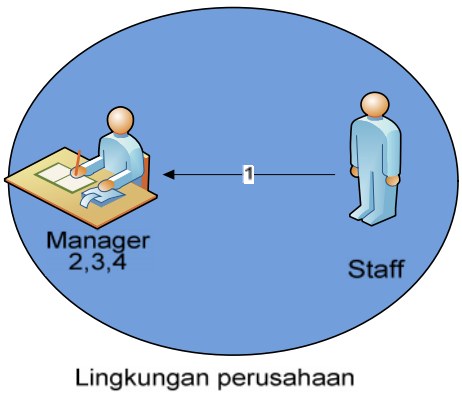


Gambar 4. 1 Proses bisnis absenssi yang sedang berjalan

Keterangan :

1. Karyawan datang ke kantor dan melakukan absensi pada staff.
2. Staff mencatat data absensi karyawan dan memeriksa keterlambatan
3. Jika karyawan tersebut terlambat, staff akan menyampaikan data keterlambatannya, dan langusng meminta denda.
4. Karyawan membayarkan denda pada staff
5. Staff mencatat data absensi, keterlambatan, dan pembayaran.

Proses bisnis penggajian yang sedang berjalan adalah sebagai berikut :



Gambar 4. 2 Proses bisnis penghitungan penggajian yang sedang berjalan

Keterangan :

1. Staff menyerahkan data absensi pada manager.
2. Manager memeriksa data setiap karyawan.
3. Manager melakukan perhitungan untuk penambahan bonus atau pengurangan denda pada bulan tersebut.
4. Manager mencatat data perhitungan penggajian untuk kemudian dilakukan pembayaran gaji pada setiap karyawan.

Proses bisnis laporan yang sedang berjalan adalah sebagai berikut :



Gambar 4. 3 Proses bisnis pelaporan yang sedang berjalan

Keterangan :

1. Manager melakukan data perhitungan penggajian.

2. Manager menyerahkan data penghitungan total penggajian pada staff agar dapat dimasukkan dalam laporan keuangan
3. Staff mencatat data laporan penggajian

4.1.1.2 Prosedur Penggajian dan Absensi Berjalan

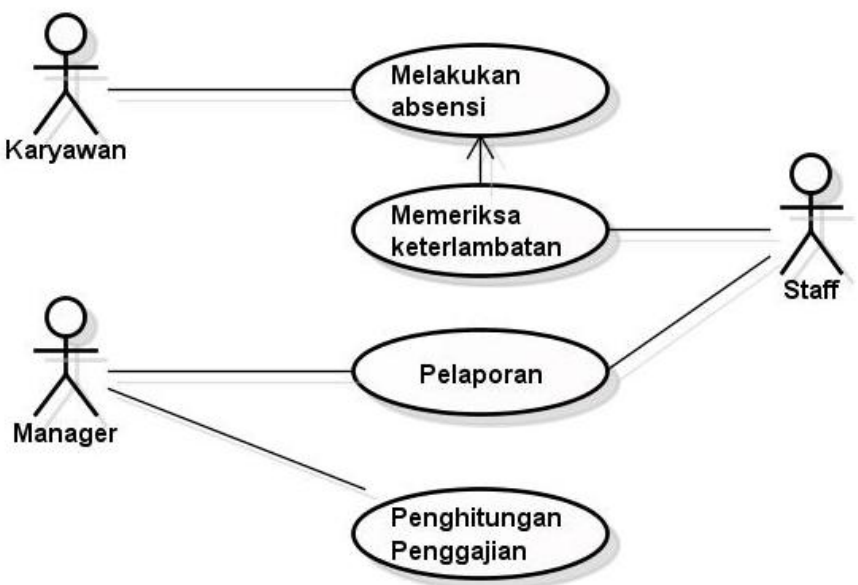
Prosedur penggajian yang sedang berjalan di option cell adalah sebagai berikut:

1. Jam masuk kerja dibagi menjadi dua shift, shift 1 masuk mulai dari jam 10 pagi s/d jam 6 sore, shift 2 masuk mulai dari jam 2 siang s/d jam 9 malam. Jumlah hari kerja dalam satu bulan yaitu 25 – 27 hari, tanggal merah dan hari besar tidak libur kecuali hari raya idul fitri.
2. Lembur dihitung perjam Rp 10 ribu.
3. Besarnya gaji ditentukan dari jabatan dan status pernikahan
    a) Direktur gaji pokok Rp 2 jt/ bulan.
    b) Kepala toko gaji pokok 1,5 jt/ bulan.
    c) *Sales counter* gaji pokok 850 ribu/ bulan.
    d) Uang makan 600 ribu/ bulan.
    e) Uang *transport* 180 ribu/ bulan.
    f) Tunjangan istri 150 ribu/ bulan.
    g) Tunjangan anak 100 ribu/ bulan.
4. Karyawan yang mendapatkan gaji pokok yaitu karyawan yang masa kerjanya telah 1 bulan, karyawan yang baru masuk kurang dari 1 bulan belum mendapatkan gaji, hanya menerima uang transport dan uang makan.

5. Untuk keterlambatan masuk kerja maksimal 30 menit, diatas 30 menit karyawan yang terlambat mendapat kan denda sebesar 15 ribu per 30 menit dipotong dari total gaji.
6. Untuk status karyawan yang sudah menikah dan mempunyai anak, hanya 1 anak yang termasuk kedalam tunjangan anak.

4.1.1.3 *Use Case* Diagram

Diagram *use case* bisnis memperlihatkan hubungan-hubungan antara aktor-aktor bisnis, *use case* bisnis, dan pekerja-pekerja bisnis untuk suatu organisasi atau perusahaan. Diagram ini memberikan model lengkap tentang apa yang dilakukan perusahaan, siapa yang ada di dalam perusahaan, serta siapa yang berada di luar perusahaan. Berikut adalah gambar model diagram use case bisnis yang berjalan :



Gambar 4. 4 *Use case* sistem yang sedang berjalan

1. Identifikasi *use case*

   Identifikasi *use case* adalah *use case* yang bekerja pada sistem dan *use case* yang bekerja pada sistem adalah sebagai berikut :

Tabel 4. 1 Identifikasi u*se case* sistem berjalan

No	*Use Case*	Deskripsi
UC-01	Melakukan absensi	Fungsionalitas untuk melakukan absensi dari setiap karyawan
UC-02	Memeriksa data keterlambatan	Fungsionalitas melakukan pencatatan dan pemeriksaan data absensi dan keterlambatan dari setiap karyawan
UC-03	Pelaporan	Fungsionalitas untuk melakukan pelaporan
UC-04	Perhitungan penggajian	Fungsionalitas untuk melakukan proses perhitungan penggajian untuk setiap karyawan

2. Identifikasi aktor

   Aktor yang berperan dalam menjalankan sistem adalah sebagai berikut :

Tabel 4. 2 Identifikasi aktor sistem berjalan

No	Aktor	Deskripsi
P-01	Karyawan	Aktoryang berperan sebagai karyawan pada kantor dan berhak melakukan data absensi
P-02	Staff	Aktor yang berperan melakukan pemeriksaan data keterlambatan dan mencatat data absensi
P-03	Manager	Aktor yang berfungsi untuk melakukan proses perhitungan penggajian dan bertanggung jawab penuh dengan seluruh kegiatan pada perusahaan

4.1.1.4 Skenario *Use Case*

Skenario *Use Case* digunakan untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. Adapun tahapan dari skenario *use case* pada sistem yang sedang berjalan adalah sebagai berikut :

4.1.1.3.1 Skenario u*se case* melakukan absensi

Tabel 4. 3 Skenario *use case*melakukan absensi

Identifikasi	
Judul	Melakukan absensi
No. Use Case	UC-01
Deskripsi Use Case	Karyawan datang ke kantor dan melakukan data absensi
Aktor	Karyawan
Trigger	Pelanggan datang untuk melakukan absensi
Kondisi awal	Staff siap untuk melakukan pencatatan dan pemeriksaan keterlambatan
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Karyawan datang ke kantor untuk melakukan absensi	
	2. Staff melakukan pemeriksaan data absensi dan keterlambatan

4.1.1.3.2 Skenario *use case* memeriksa data keterlambatan

Tabel 4. 4 Skenario *use case* memeriksa data keterlambatan

Identifikasi	
Judul	Memeriksa data keterlambatan
No. Use Case	UC-02
Deskripsi Use Case	Staff melakukan pemeriksaan data keterlambatan karyawan
Aktor	Staff
Trigger	Proses absensi telah selesai
Kondisi awal	Menunggu proses absensi selesai
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Staff memeriksa data absensi dari karyawan	
2. Jika karyawan terlambat, staff meminta denda pada karyawan untuk kompensasi keterlambatannya	
	3. Menyimpan data absensi karyawan
4. Karyawan melakukan pembayaran sesuai dengan ketentuan	

4.1.1.3.3 Skenario *use case* pelaporan

Tabel 4. 5 Skenario *use case* pelaporan

Identifikasi	
Judul	Pelaporan
No. Use Case	UC-03
Deskripsi Use Case	Membuat laporan absensi dan penggajian
Aktor	Staff, Manager
Trigger	Manager membutuhkan data pelaporan untuk keperluan penghitungan gaji
Kondisi awal	Staff siap untuk membuat laporan
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Staff membuat laporan absensi untuk keperluan penggajian	
2. Staff menyerahkan laporan kepada manager	
	3. Managermemeriksa laporan
	4. Manager menyimpan arsip laporan

4.1.1.3.4 Skenario *use case* penghitungan penggajian

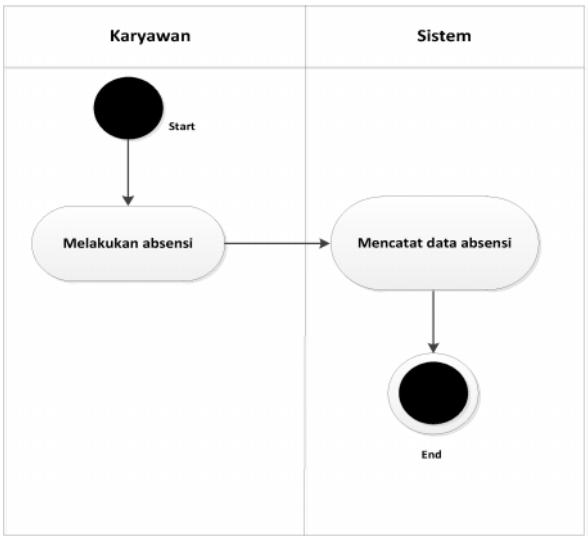
Tabel 4. 6 Skenario *use case* penghitungan penggajian

Identifikasi	
Judul	Penghitungan penggajian
No. Use Case	UC-04
Deskripsi Use Case	Melakukan penghitungan penggajian untuk setiap karyawan
Aktor	Manager
Trigger	Manager akan melakukan penghitungan penggajian
Kondisi awal	Data absensi telah siap dan tersedia
Skenario Utama	
Aksi- Aktor	**Reaksi Sistem**
1. Manager memeriksa data absensi yang ada untuk melakukan perhitungan penggajian	
2. Manager melakukan perhitungan data penggajian	
	3. Menyimpan data penggajian

4.1.1.5 *Activity Diagram*

Activity diagram adalah diagram yang menggambarkan aliran fungsionalitas dari sistem. Berikut *activity diagram* proses absensi pada sistem yang berjalan :

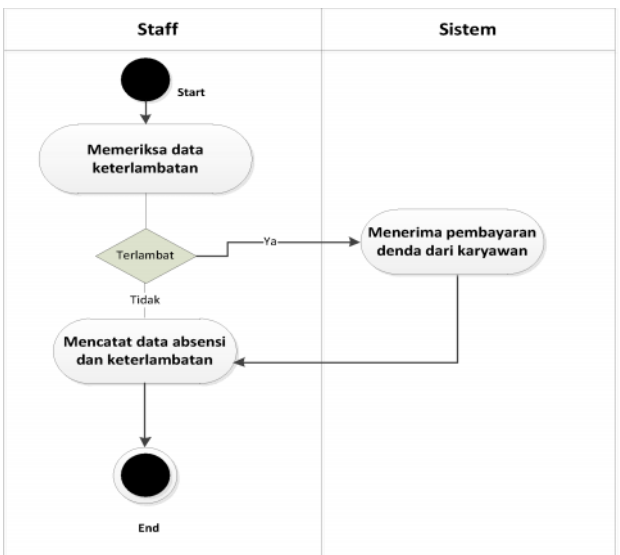
4.1.1.4.1 *Activity diagram* proses absensi



Gambar 4. 5 *Activity diagram* proses absensi

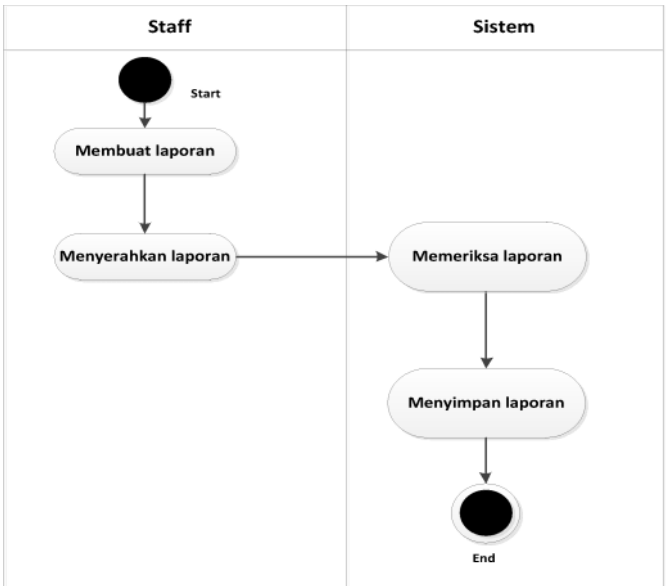
4.1.1.4.2 *Activity diagram* memeriksa data keterlambatan

Activity diagram pada proses pemeriksaan data keterlambatan



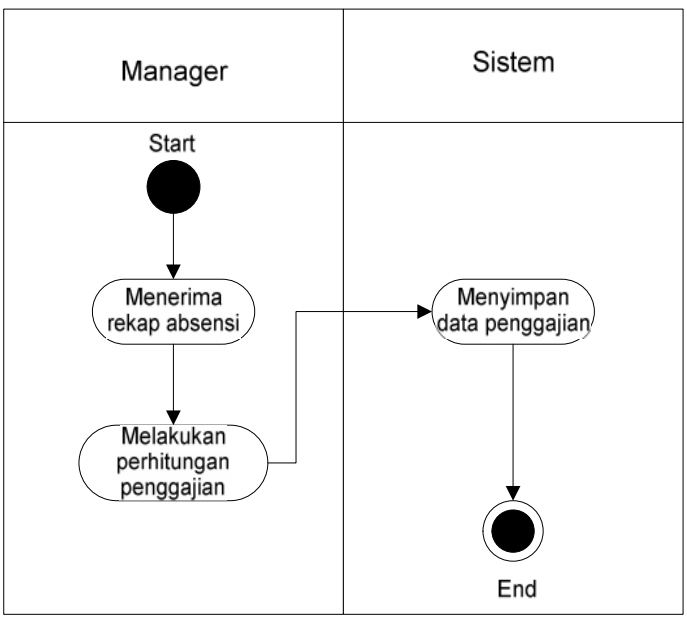
Gambar 4. 6 *Activity diagram* pemeriksaan data keterlambatan

4.1.1.4.3 *Activity diagram* pelaporan



Gambar 4. 7 *Activity diagram* pelaporan

4.1.1.4.4 *Activity Diagram* penghitungan penggajian



Gambar 4. 8 *Activity diagram* penghitungan penggajian

4.1.2 Evaluasi Proses Bisnis yang Sedang Berjalan

Setelah dilakukan penelitian pada proses bisnis berjalan di Option Cell, terdapat beberapa kelemahan yang ditemukan pada proses yang sedang berjalan. Kelemahan-kelemahan tersebut adalah sebagai berikut:

Tabel 4.7 Evaluasi proses bisnis yang berjalan

No	Permasalahan	Penyelesaian
1.	Proses penggajian nya masih menggunakan microsoft excel, sehingga proses penghitungan gaji pegawainya membutuhkan waktu yang cukup lama	Merancang aplikasi agar dapat memberikan kemudahan dalam melakukan proses perhitungan penggajian
2.	Media penyimpanan data pegawai masih berupa arsip sehingga memperlambat dalam proses pencarian data dan pembuatan laporan gaji pegawai karena tidak adanya media penyimpanan data/ database khusus penggajian	Merancang dan membangun aplikasi yang memiliki dan mengelola *database* secara baik dan terintegrasi sehingga membantu dalam kegiatan proses penggajian serta perhitungannya berdasarkan absensi yang ada.
3.	Pembuatan slip gaji dan laporan gajinya menggunakan microsoft excel sehingga proses pembuatan nya lambat dan membutuhkan waktu yang cukup lama. Selain itu juga sering terjadinya kesalahan.	Merancang dan membangun aplikasi yang memiliki cetak slip gaji dan laporan gaji berdasarkan periode yang diinginkan.

4.2 Perancangan Sistem

Perancangan sistem merupakan suatu kegiatan pengembangan prosedur dan proses yang sedang berjalan untuk menghasilkan suatu sistem yang baru, atau memperbaharui sistem yang ada untuk meningkatkan efektifitas kerja agar dapat memenuhi hasil yang digunakan dengan tujuan memanfaatkan teknologi dan fasilitas yang tersedia. Pada bagian ini penyusun akan memberikan usulan yang merupakan sistem informasi secara komputerisasi yang diharapkan akan membantu dan mempermudah pekerjaan.

4.2.1 Tujuan Perancangan Sistem

Perancangan sistem bertujuan untuk memberikan gambaran yang jelas dan rancang bangun yang sesuai dengan kebutuhan pemakai sistem itu sendiri. Perancangan sistem dilakukan apabila tahap analisis sistem telah dilakukan. Maka untuk selanjutnya seorang analis sistem merancang bagaimana membentuk sistem yang baru ataupun memperbaharui sistem yang lama. Tahap inilah yang dinamakan dengan istilah dari perancangan sistem.

Adapun tujuan perancangan sistem yang diusulkan yaitu untuk menghasilkan suatu rancangan sistem yang dapat membantu proses pengolahan data karyawan dalam menangani absensi dan perhitungan penggajian yang semuanya terintegrasi dengan *database* dan aplikasi tersebut dapat mencapai sasaran kebutuhan dan tujuannya yakni membantu perusahaan dalam pengelolaan data.

4.2.2 Gambaran Umum Sistem yang Diusulkan

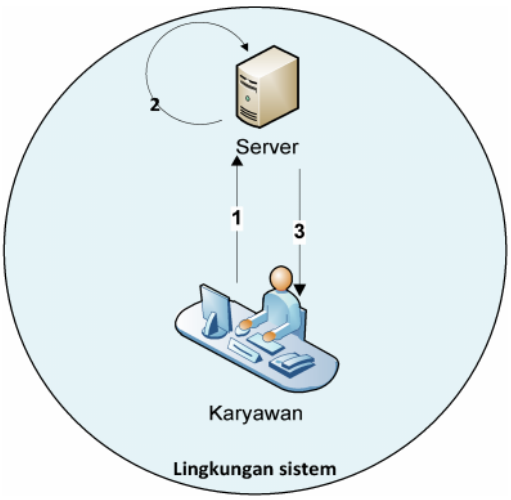
Gambaran umum tentang sistem yang diusulkan pada proses perancangan ini adalah suatu sistem yang merubah seluruh proses bisnis yang manual menjadi terkomputerisasi. Proses bisnis yang dimaksud adalah pengolahan data karyawan, proses absensi, penghitungan data penggajian, hingga proses laporan kepada manager ditangani oleh sistem informasi ini. Sistem ini diharapkan dapat membantu efektifitas dan optimalisasi waktu pada seluruh proses bisnis yang berjalan serta dapat membantu mengatasi permasalahan yang ada dan yang paling penting dapat menghasilkan informasi yang cepat, tepat dan akurat.

4.2.3 Perancangan Prosedur yang Diusulkan

Perancangan prosedur merupakan awal dari pembuatan sistem informasi yang akan dibuat, dimana dapat dilihat proses-proses apa saja yang nantinya diperlukan dalam pembuatan suatu sistem informasi. Sedangkan perancangan prosedur yang diusulkan merupakan tahap untuk memperbaiki atau meningkatkan efisiensi kerja. Tahap perancangan sistem yang digambarkan merupakan tahap perancangan untuk membangun suatu sistem dan mengkonfigurasikan komponen-komponen perangkat lunak dan perangkat keras sehingga menghasilkan sistem informasi yang baik. Sistem informasi yang dirancang tersebut menjadi satu komponen. Tahapan perancangan prosedur ini akan dijelaskan dengan menggunakan pemodelan sistem informasi berorientasi objek dengan *Unified Modeling Language* (UML).

4.2.3.1 Proses Bisnis Diusulkan

Proses bisnis absensi yang diusulkan adalah sebagai berikut :

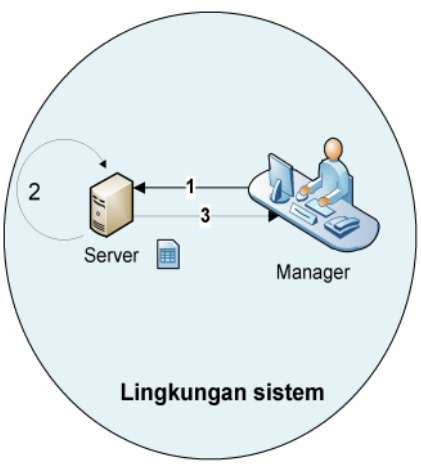


Gambar 4. 9 Proses bisnis absensi yang diusulkan

Keterangan :

1. Karyawan datang ke kantor dan melakukan absensi pada sistem
2. Sistem memeriksa data karyawan dan menyimpan data absensi pada karyawan tersebut
3. Sistem memberitahu karyawan data absensi karyawan tersebut.

Proses bisnis perhitungan data penggajian yang diusulkan adalah sebagai berikut :



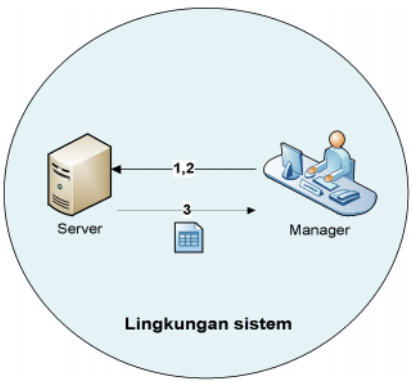
Gambar 4. 10 Proses bisnis penghitungan penggajian yang diusulkan

Keterangan :

1. Manager meminta sistem melakukan perhitungan penggajian

2. Sistem memeriksa data inputan dari manager dan melakukan proses kalkulasi penggajian
3. Sistem menampilkan data penggajian hasil kalkulasi.

Proses bisnis pelaporan yang diusulkan adalah sebagai berikut :



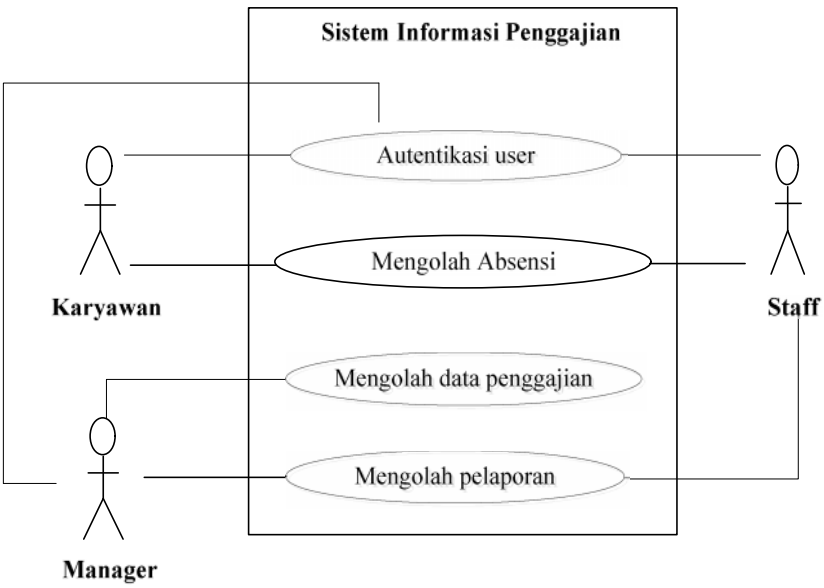
Gambar 4. 11 Proses bisnis pelaporan yang diusulkan

Keterangan :

1. Manager melakukan login ke sistem
2. Managermeminta pelaporan dengan memasukkan data periode laporan dan jenis laporan yang diinginkan.
3. Sistem secara terkomputerisasi akan menampilakan laporan sesuai dengan kondisi yang diinginkan.

4.2.3.2 *Use Case* Diagram

Diagram *use case* bisnis memperlihatkan hubungan-hubungan antara aktor-aktor bisnis, *use case* bisnis, dan pekerja-pekerja bisnis untuk suatu organisasi atau perusahaan. Diagram ini memberikan model lengkap tentang apa yang dilakukan perusahaan, siapa yang ada di dalam perusahaan, serta siapa yang berada di luar perusahaan. Berikut adalah gambar model diagram use case bisnis yang diusulkan :



Gambar 4. 12 *Use case* diagram yang diusulkan

1. Identifikasi *use case*

   Identifikasi *use case* adalah *use case* yang bekerja pada sistem dan *use case* yang bekerja pada sistem adalah sebagai berikut :

Tabel 4. 7 Identifikasi u*se case* sistem yang diusulkan

No	*Use Case*	Deskripsi
UC-01	Autentikasi user	Fungsionalitas untuk menentukan hak akses user pada saat login ke sistem
UC-02	Mengolah absensi	Fungsionalitas untuk mengolah data absensi
UC-03	Mengolah data penggajian	Fungsionalitas untuk mengolah data penggajian
UC-04	Mengolah pelaporan	Fungsionalitas untuk melakukan pemeriksaan dan pencetakan laporan.

2. Identifikasi aktor

   Aktor yang berperan dalam menjalankan sistem adalah sebagai berikut :

Tabel 4. 8 Identifikasi aktor sistem diusulkan

No	Aktor	Deskripsi
P-01	Karyawan	Aktor yang berperan sebagai karyawan pada kantor dan berhak melakukan data absensi
P-02	Staff	Aktor yang berperan melakukan pemeriksaan data keterlambatan dan mencatat data absensi
P-03	Manager	Aktor yang berfungsi untuk melakukan proses perhitungan penggajian dan bertanggung jawab penuh dengan seluruh kegiatan pada perusahaan

4.2.3.3 Skenario *Use Case*

Skenario *Use Case* digunakan untuk memudahkan dalam menganalisa skenario yang akan kita gunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut. Adapun tahapan dari skenario *use case* pada sistem yang diusulkan adalah sebagai berikut :

4.2.3.3.1 Skenario u*se case* autentikasi user

Tabel 4. 9 Skenario *use case* autentikasi user

Identifikasi		
Judul	Autentikasi user	
No. Use case	UC-01	
Deskripsi use case	Aktor login ke sistem untuk menentukan hak aksesnya	
Aktor	Karyawan, Staff, Manager	
Trigger	Aktor menekan tombol login	
Kondisi awal	Aplikasi menampilkan sebuah form untuk login	
Skenario Utama		
Aksi- aktor		**Reaksi sistem**

1. Aktor mengisi *field* kosong pada *form* login.	
2. Aktor menekantombol login.	3. Sistem memeriksa keabsahan *user* berdasarkan *username* dan *password* pada *database*.
	4. Sistem mengkonfirmasi keabsahan user.
Kondisi akhir skenario utama	Sistem menampilkan halaman utama berdasarkan hak akses dari user
Skenario Alternatif - 1	
Aksi- aktor	**Reaksi sistem**
1. Aktor menekan tombol login tanpa mengisi *field-field* pada form.	2. *Form* login ditampilkan dan ada pesan kesalahan yang menyatakan *username* atau *password* harus diisi.
	3. *Form* utama tidak ditampilkan.
Kondisi akhir skenario alternatif -1	Aplikasi menampilkan *form login*.

4.2.3.3.2 Skenario u*se case* mengolah absensi

Tabel 4. 10 Skenario *use case* mengolah absensi

Identifikasi	
Judul	Mengolah absensi
No. Use case	UC-02
Deskripsi use case	Aktor menambah data absensi
Aktor	Karyawan, Staff
Trigger	Aktor memilih menekan tombol absen
Kondisi awal	Aplikasi menampilkan *formlogin*
Skenario Utama	
Aksi- aktor	**Reaksi sistem**

1. Aktor mengisi data *username* dan *password*	
2. Aktor memilih absen (absen masuk atau keluar)	
3. Aktor men-*submit* data absensi.	4. Sistem memeriksa apakah *field-field* pada *form* pengisian data telah diisi dan sesuai.
	5. Sistem menambahkan data absensi yang baru pada *database*
	6. *Formlogin* kembali tampil
Kondisi akhir skenario utama	Sistem menyimpan data transaksi absensi yang baru pada *database*
Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**
1. Aktor men-*submit* data dengan menekan sebuah *button* di mana ada *field-field* pada form yang seharusnya diisi tetapi dikosongkan atau ada data yang tidak sesuai.	2. *Formlogin* ditampilkan dengan pesan bahwa ada data *field* yang harus diisi namun masih kosong, atau ada data yang tidak sesuai.
	3. Data absensi baru tidak disimpan pada *database.*
Kondisi akhir skenario alternatif -1	*Formlogin* ditampilkan

4.2.3.3.3 Skenario u*se case* mengolah data penggajian

Tabel 4. 11 Skenario *use case* mengolah data penggajian

Identifikasi	
Judul	Mengolahdata penggajian
No. Use case	UC-03

Deskripsi use case	Aktor meminta sistem melakukan kalkulasi penggajian.
Aktor	Manager
Trigger	Aktor memilih menu untuk melakukan perhitungan data penggajian.
Kondisi awal	Aplikasi menampilkan sebuah untuk melakukan perhitungan penggajian.

Skenario Utama	
Aksi- aktor	**Reaksi sistem**
1. Aktor mengisi *field* pada *form*untuk kondisi data penggajian yang diinginkan.	
2. Aktor menekan tombol untuk melakukan perhitungandata penggajian.	3. Sistem memeriksa apakah *field-field* pada *form* pengisian data telah diisi dan sesuai.
	4. Sistem melakukan perhitungan data penggajian sesuai dengan kondisi yang diberikan.
	5. Sistem menyimpan data penggajian yang baru dikalkulasi ke dalam *database* dan menampilkannya.
Kondisi akhir skenario utama	Data penggajian yang baru dikalkulasi disimpan pada *database* dan ditampilkan pada user

Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**

1. Aktor men-*submit* data dengan menekan sebuah *button* di mana ada *field-field* pada form yang seharusnya diisi tetapi dikosongkan atau ada data yang tidak sesuai.	2. *Form*perhitungan data penggajian ditampilkan dengan pesan bahwa ada data *field* yang harus diisi namun masih kosong, atau ada data yang tidak sesuai.
	3. Data penggajian baru tidak disimpan pada *database.*
Kondisi akhir skenario alternatif -1	*Form*kalkulasi penggajian ditampilkan

4.2.3.3.4 Skenario u*se case* mengolah pelaporan

Tabel 4. 12 Skenario *use case* mengolah pelaporan

Identifikasi	
Judul	Mengolah laporan
No. Use case	UC-04
Deskripsi use case	Aktor mencetak laporan yang terjadi pada range waktu yang diinginkan.
Aktor	Manager.
Trigger	Aktor memilih menu untuk mencetak laporan.
Kondisi awal	Aplikasi menampilkan form untuk menentukan periode dan jenis laporan yang diinginkan.

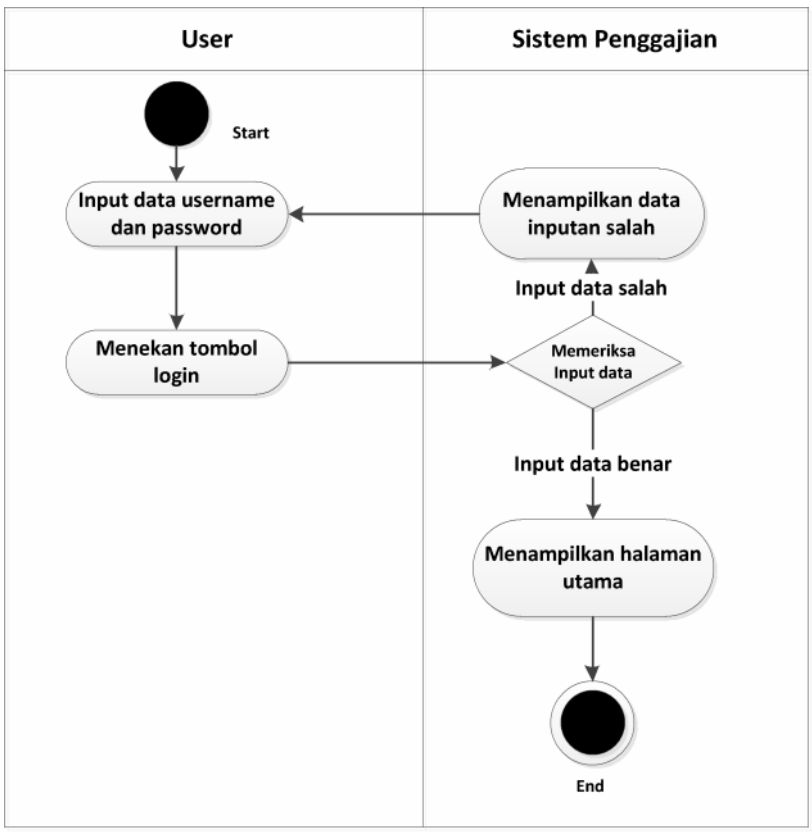
Skenario Utama	
Aksi- aktor	**Reaksi sistem**

1. Aktor mengisi periode waktu dan jenis laporan yang diinginkan untuk dicetak.	
2. Aktor menekan tombol untuk mencetak laporan	
	3. Sistem mencetak laporan berdasarkan sesuai dengan jenis dan periode laporan yang dipilih.
Kondisi akhir skenario utama	Laporan dicetak
Skenario Alternatif –1	
Aksi- aktor	**Reaksi sistem**
1. Aktor memilih periode yang tidak memiliki data samasekali.	
2. Aktor menekan tombol untuk mencetak laporan	
	3. Sistem menampilkan pesan bahwa data pada periode tersebut kosong.
Kondisi akhir skenario alternatif -1	Sistem menampilkan *form* untuk melakukan pengisian periode laporan yang diinginkan.

4.2.3.4 *Activity Diagram*

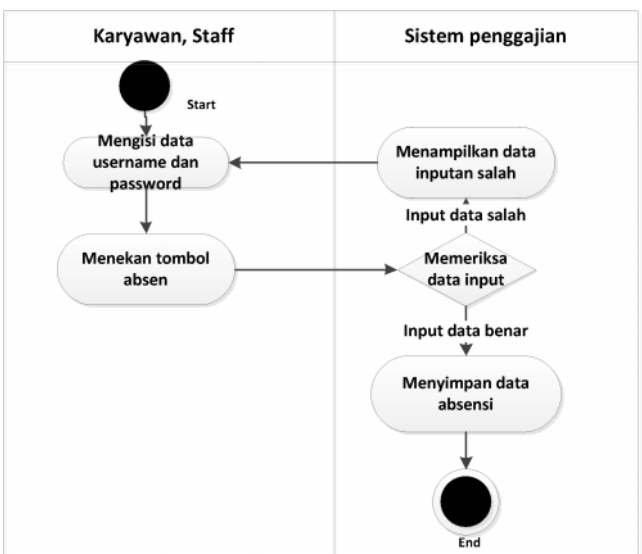
Activity diagram (diagram aktivitas) adalah diagram yang menggambarkan aliran fungsionalitas dari sistem. Berikut *activity diagram* proses perbaikan pada sistem yang diusulkan :

4.2.3.4.1 *Activity Diagram* autentikasi user



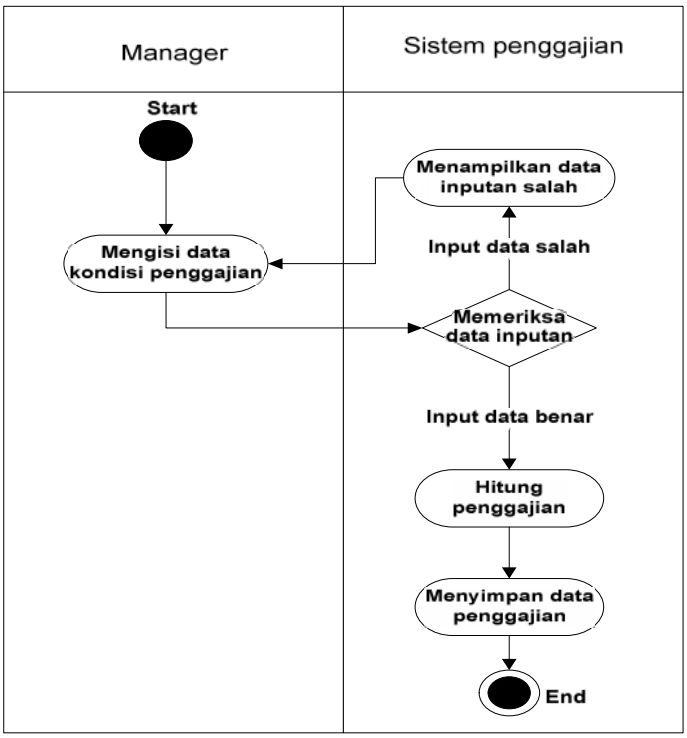
Gambar 4.13 *Activity diagram* autentikasi *user*

4.2.3.4.2 *Activity Diagram* mengolah absensi



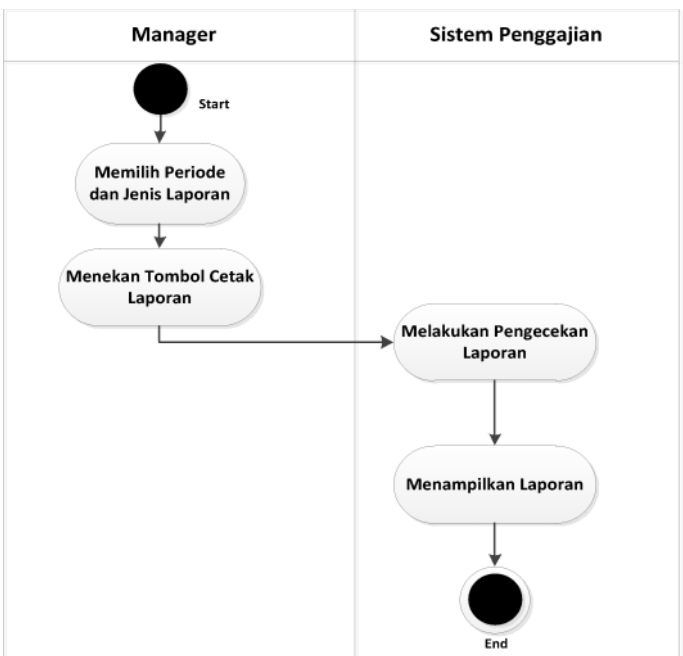
Gambar 4. 14 *Activity diagram* mengolah absensi

4.2.3.4.3 *Activity Diagram* mengolah data penggajian



Gambar 4. 15 *Activity diagram* mengolah penggajian

4.2.3.4.4 *Activity Diagram* mengolah pelaporan

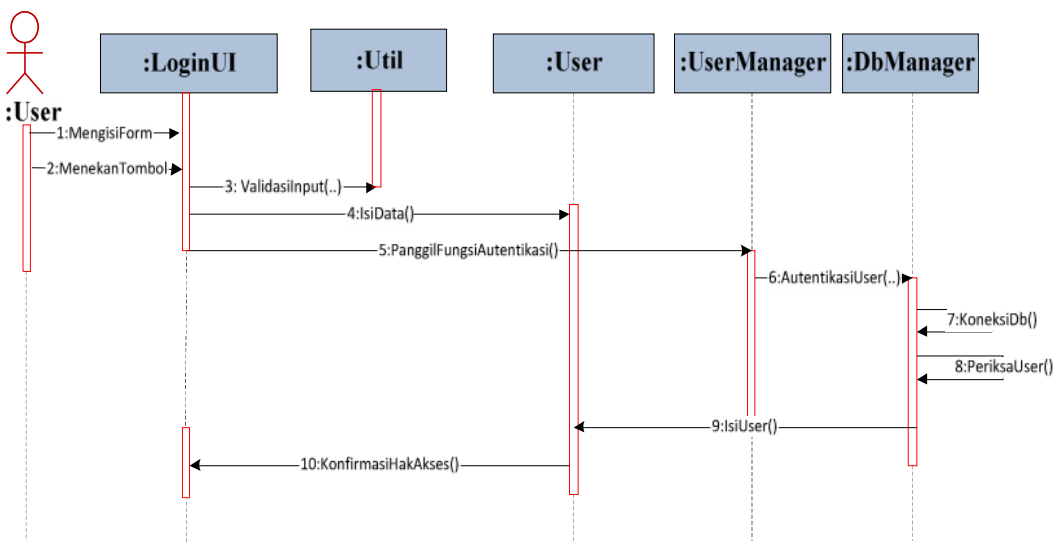


Gambar 4. 16 *Activity diagram* mengolah pelaporan

4.2.3.5 SequenceDiagram

Sequence Diagram digunakan untuk menggambarkan interaksi antar objek dalam waktu yang berurutan. Tetapi pada dasarnya *sequence diagram* digunakan dalam lapisan abstraksi model objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antar object, juga interaksi antar objek, dan menunjukkan sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Komponen utama *sequence diagram* terdiri atas objek yang dituliskan dengan kotak segiempat bernama, pesan diwakili oleh garis dengan tanda panah, dan waktu yang ditunjukkan dengan proses vertikal. Berikut adalah *sequence diagram* yang diusulkan :

4.2.3.5.1 *Sequence Diagram* autentikasi user



Gambar 4. 17 *Sequence diagram* Autentikasi User

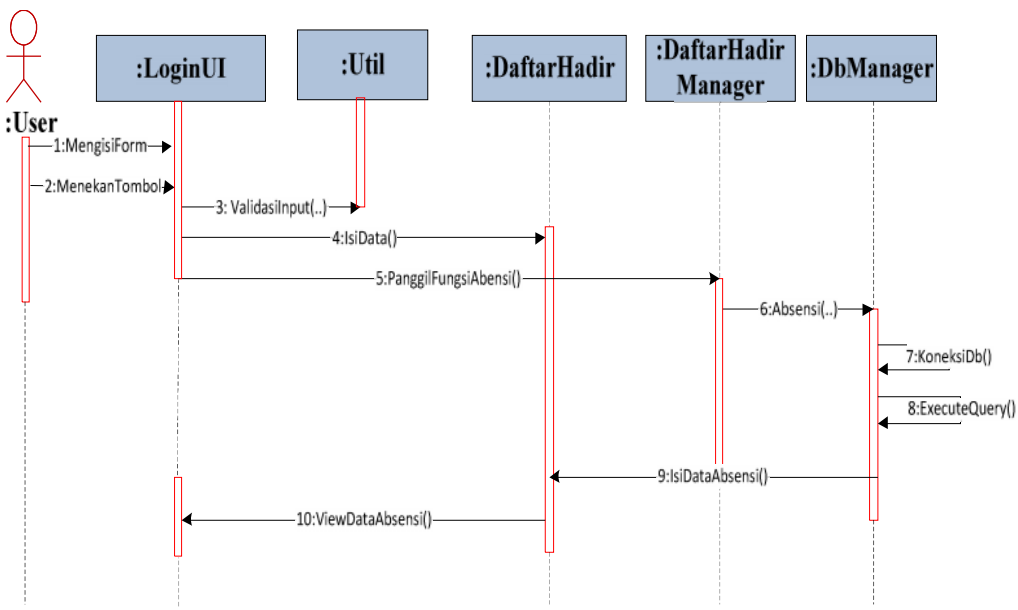
Keterangan:

Pada Gambar 4.16 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. LoginUI

2. Util
3. User
4. UserManager
5. DbManager

4.2.3.5.2 *Sequence Diagram* mengolah absensi



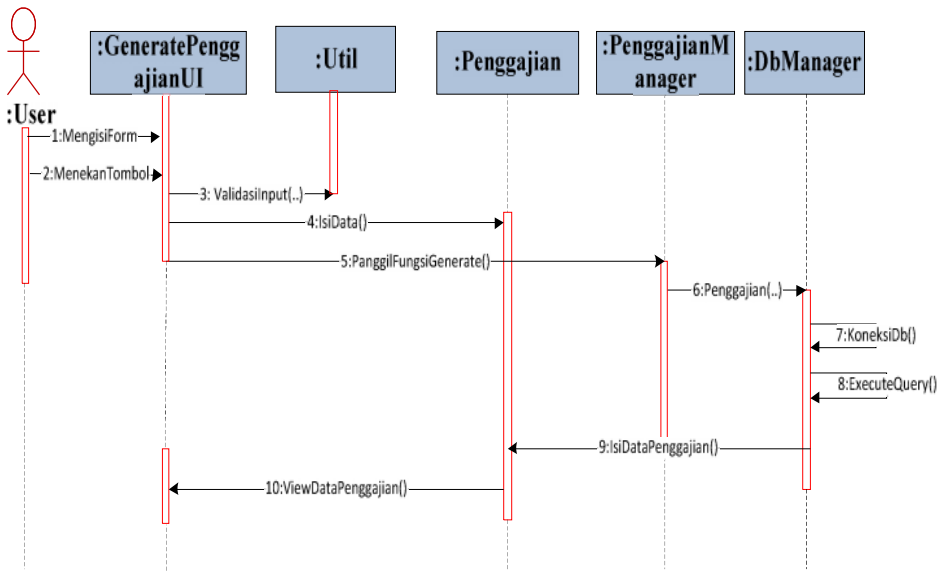
Gambar 4. 18 *Sequence diagram* Mengolah absensi

Keterangan:

Pada Gambar 4.17 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. LoginUI
2. Util
3. DaftarHadir
4. DaftarHadirManager
5. DbManager

4.2.3.5.3 *Sequence Diagram* mengolah penggajian



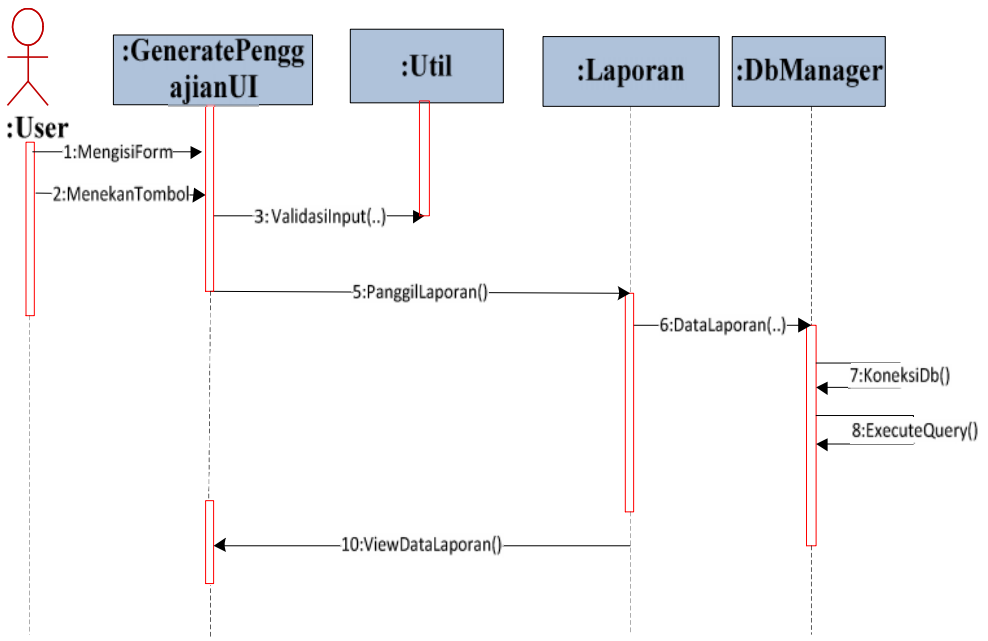
Gambar 4. 19 *Sequence diagram* Mengolah Penggajian

Keterangan:

Pada Gambar 4.18 dapat dilihat bahwa ada 5 kelas yang saling berinteraksi, yakni:

1. GeneratePenggajianUI
2. Util
3. Penggajian
4. PenggajianManager
5. DbManager

4.2.3.5.4 *Sequence Diagram* mengolah pelaporan



Gambar 4.20 *Sequence diagram* Mengolah Pelaporan

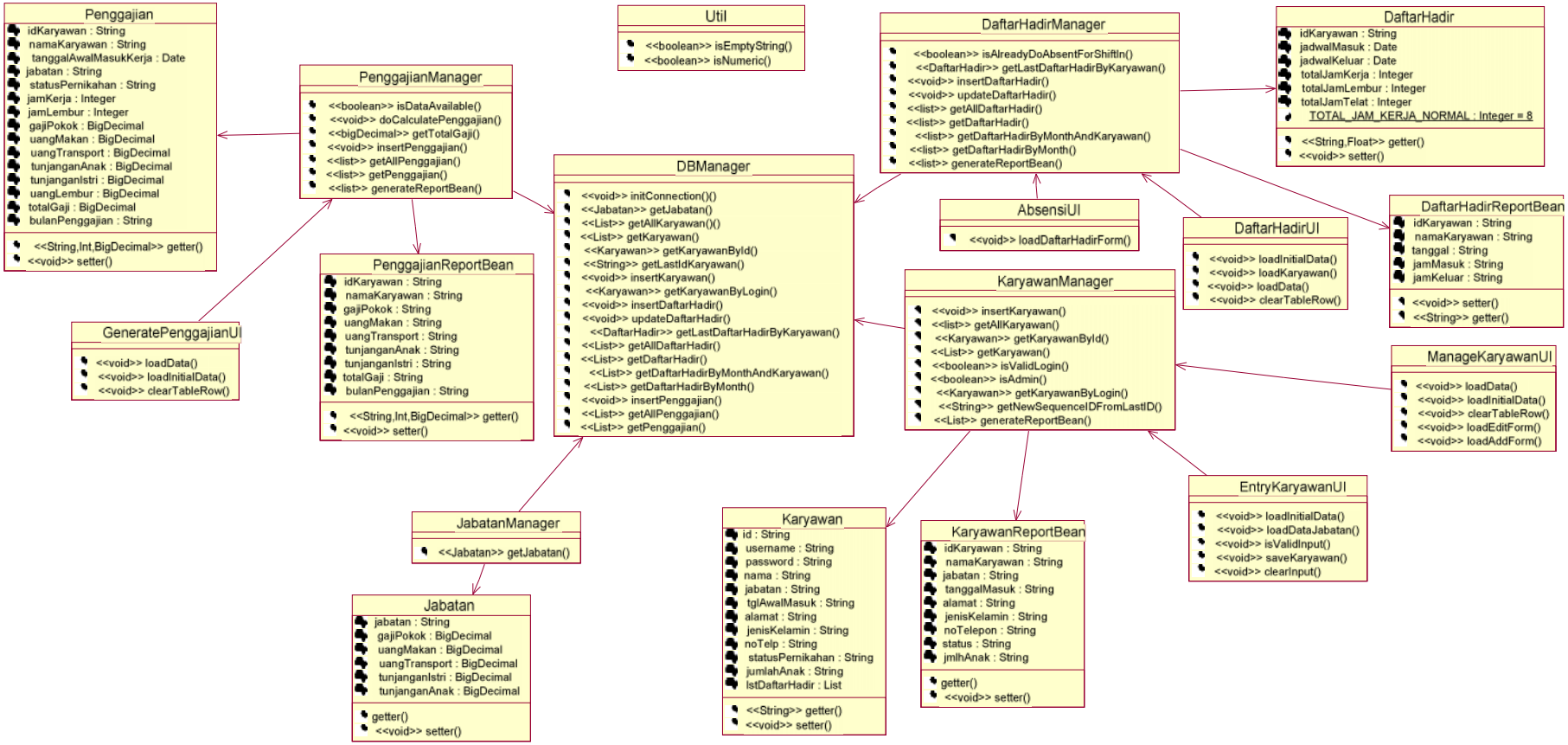
Keterangan:

Pada Gambar 4.19 dapat dilihat bahwa ada 4 kelas yang saling berinteraksi, yakni:

1. GeneratePenggajianUI
2. Util
3. Laporan
4. DbManager

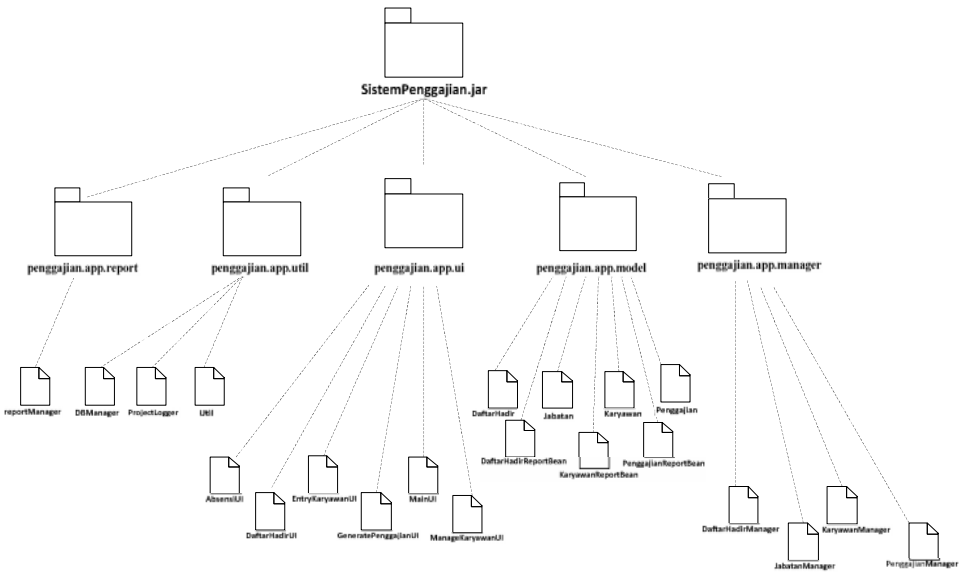
4.2.3.6 Class Diagram

Menggambarkan struktur statis *class* di dalam sistem. *Class* merepresentasikan sesuatu yang ditangani oleh sistem. Dengan melihat karakteristik sistem pemasaran produk dari bagian penjualan beserta proses-proses yang terjadi, maka dapat dibuat *class diagram* Berikut *class diagram* yang diusulkan pada Option Cell:



Gambar 4. 21 *Class Diagram* Applikasi Penggajian

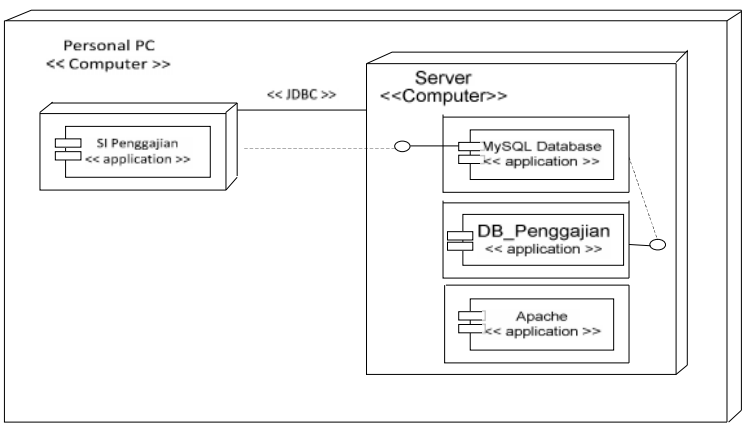
4.2.3.7 Component Diagram



Gambar 4. 22 *Component Diagram*

4.2.3.8 Deployment Diagram

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Diagram ini memuat simpul-simpul beserta komponen-komponen yang ada didalamnya. *Deployment diagram* berhubungan dengan diagram komponen dimana *deployment diagram* memuat satu atau lebih komponen-komponen.



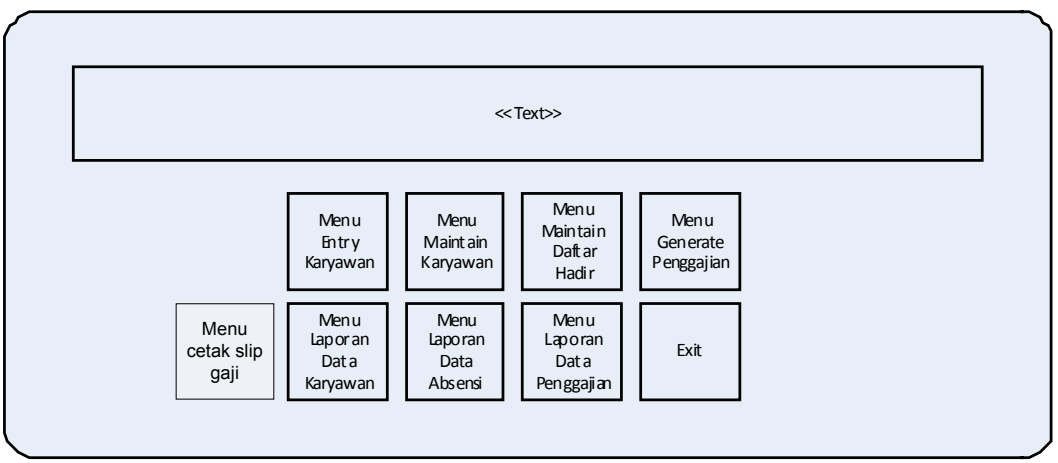
Gambar 4.23 *Deployment diagram* sistem yang diusulkan

4.2.4 Perancangan Antar Muka

Perancangan antar muka merupakan perancangan yang dibuat sebelum program aplikasi dibuat, perancangan antar muka pada sistem informasi penggajian untuk Option Cell sebagai berikut:

4.2.4.1 Perancangan Struktur Menu

Perancangan menu dibuat sebagai alat antar muka dengan pengguna untuk memudahkan pengoperasian perangkat lunak. Berikut rancangan menu perangkat lunak ini :



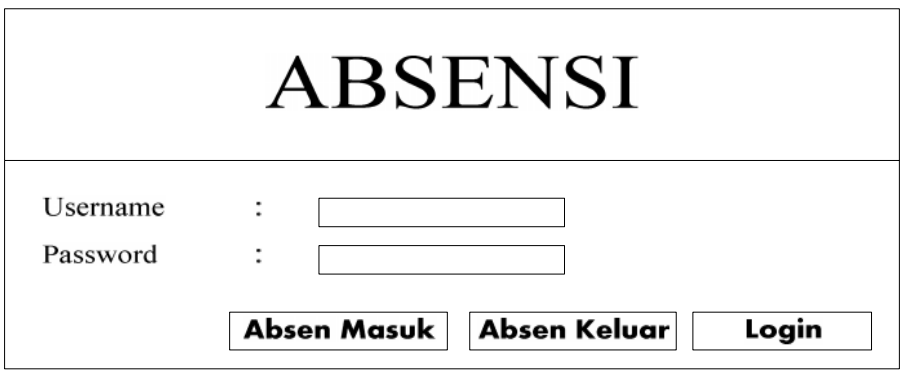
Gambar 4. 24 Perancangan Struktur Menu

4.2.4.2 Perancangan Input

Perancangan input merupakan perancangan tampilan yang akan digunakan guna memasukkan data pada sistem untuk kemudian diproses. Dalam perancangan input ini, data yang dimasukkan akan mempengaruhi hasil yang ditampilkan. Adapun perancangan-perancangan input yang ada dalam perancangan ini adalah:

1. Rancangan Login/Absensi

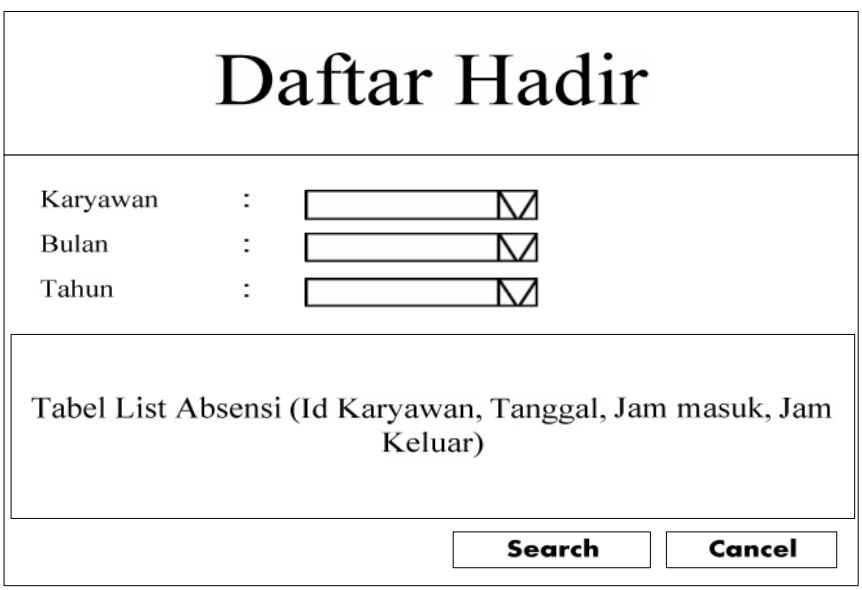
Tampilan yang dirancang ini berfungsi sebagai tampilan untuk melakukan login sekaligus melakukan absensi masuk dan absensi keluar.



Gambar 4. 25 Rancangan Login/Absensi

2. Rancangan Daftar Hadir

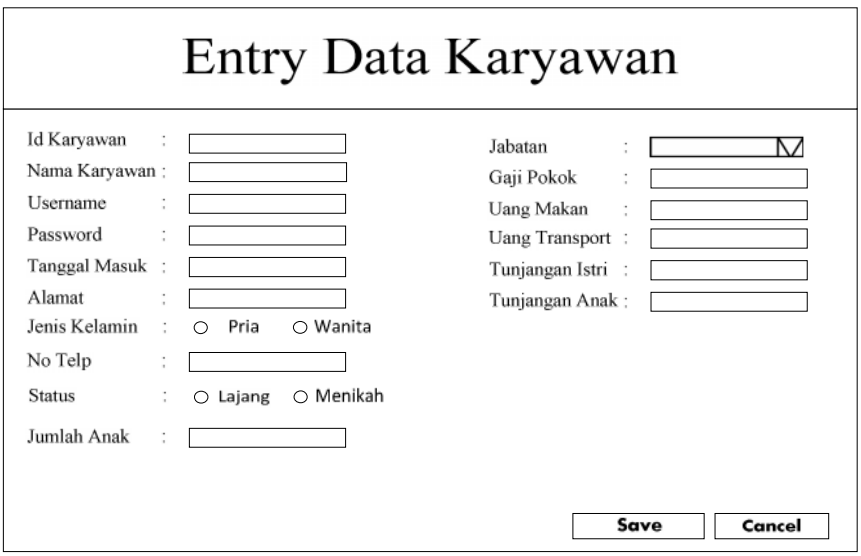
Tampilan yang dirancang ini berfungsi sebagai tampilan untuk memeriksa data absensi yang ada.



Gambar 4. 26 Rancangan daftar hadir

3. Rancangan tampilan Entry data karyawan

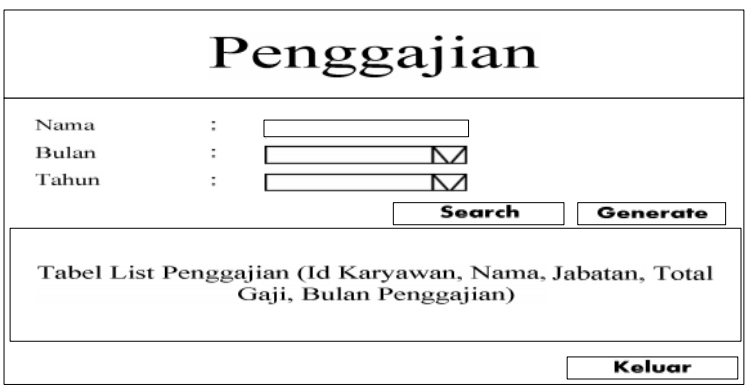
Tampilan yang dirancang ini berfungsi sebagai tampilan untuk menambahkan data karyawan yang baru.



Gambar 4. 27 Rancangan Entry Data Karyawan

4. Rancangan tampilian penggajian

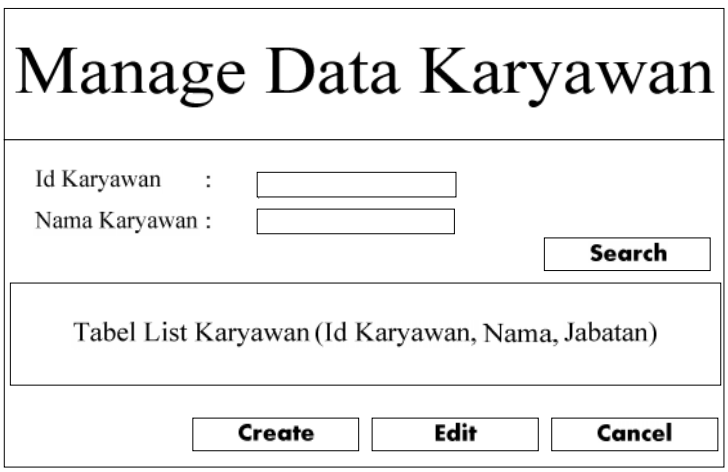
Tampilan yang dirancang ini berfungsi sebagai tampilan untuk melakukan kalkulasi penggajian dan menampilkannya.



Gambar 4. 28 Rancangan tampilan penggajian

4. Tampilan untuk mengolah data karyawan

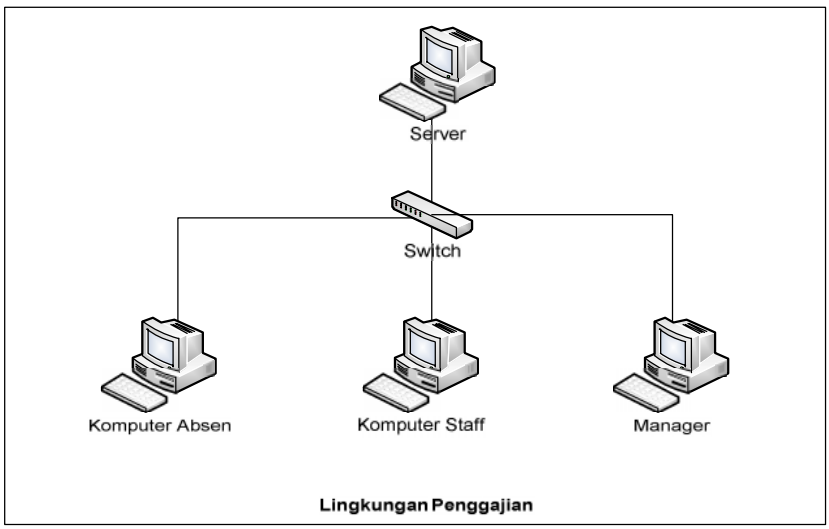
   Tampilan ini berfungsi untuk menampilkan data karyawan dan mengubahnya.



Gambar 4. 29 Rancangan tampilan mengolah data karyawan

4.2.5 Perancangan Arsitektur Jaringan

Arsitektur jaringan yang digunakan adalah sistem *client/server* mempunyai dua komponen utama yaitu komputer *server* dan komputer *client*. *Server* merupakan komputer induk yang melakukan pemrosesan terbanyak untuk memenuhi permintaan-permintaan dari komputer client dan bertindak sebagai *server database* yang menyimpan data. *Client* yaitu komputer yang melakukan pengiriman permintaan-permintaan data pada *server* kemudian menampilkan data tersebut pada *interface* aplikasi yang dimilikinya.



Gambar 4.30 Arsitektur jaringan yang diusulkan

Gambar 4.24 menunjukkan arsitektur jaringan. Disini penulis menggunakan topologi star dengan alasan jika salah satu komputer *client* rusak, komputer yang lain masih dapat diakses dan kontrol menagement lebih mudah karena semuanya terpusat ke server.