

## BAB III

### ANALISA DAN PERANCANGAN SISTEM

#### 3.1. Analisa Masalah

Langkah awal dalam pembuatan sistem adalah mempelajari permasalahan yang ada sebagai dasar untuk membuat sebuah solusi yang disajikan dalam bentuk aplikasi.

##### 3.1.1. Identifikasi Masalah

Pada langkah identifikasi masalah dilakukan tahapan-tahapan untuk mengetahui permasalahan yang ada yaitu dengan observasi dan wawancara ke Rumah Sakit Bedah Surabaya, sehingga bisa dilakukan tindakan solusi yang tepat untuk menyelesaikan permasalahan tersebut.

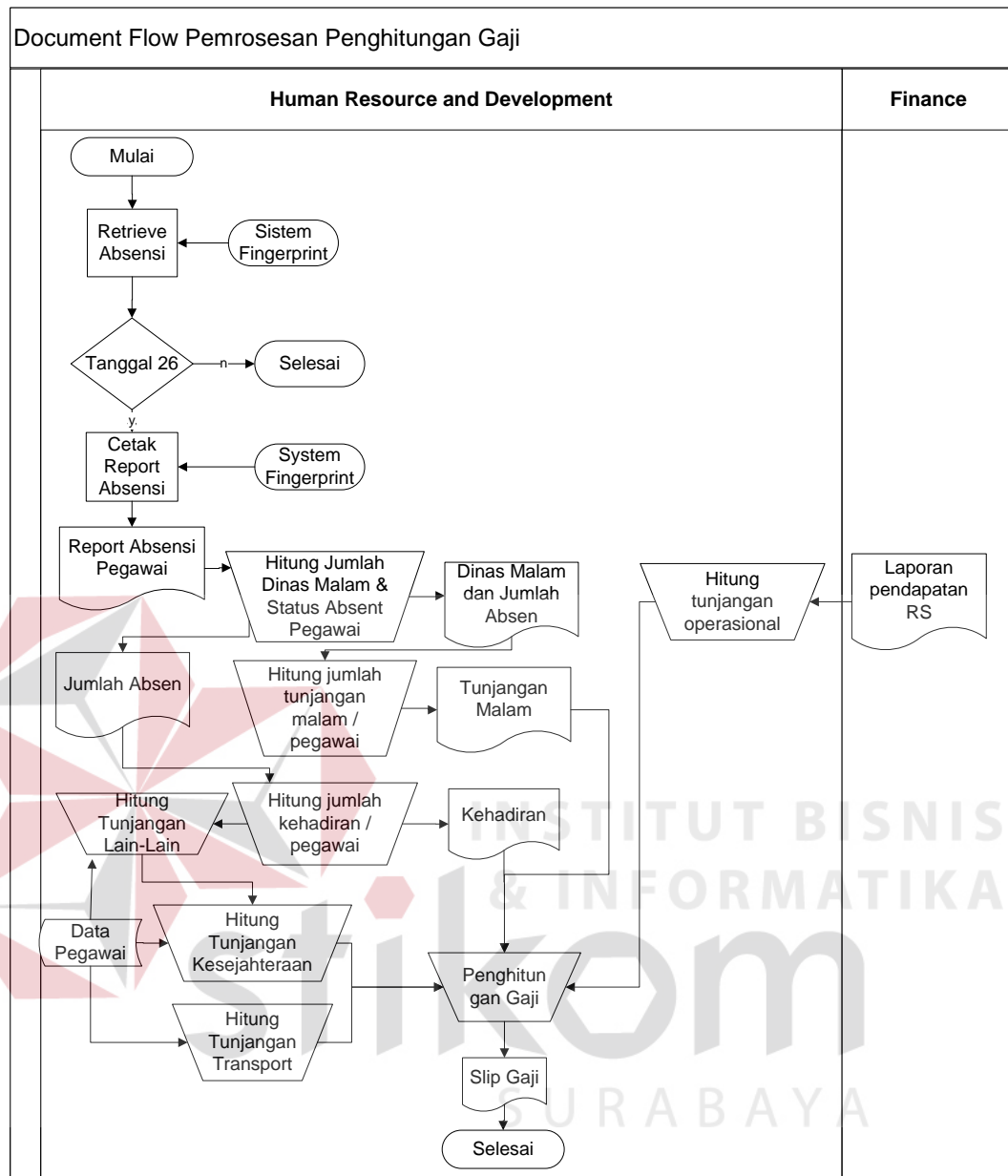
Dari hasil observasi dan wawancara dengan pihak HRD pada RS Bedah Surabaya selama ini penghitungan gaji karyawan dilakukan secara semi manual menggunakan *software* Microsoft Excel. Hal ini memang membantu pada awalnya tapi seiring berkembangnya RS Bedah Surabaya yang berarti bertambahnya jumlah karyawan dibutuhkan waktu lebih banyak untuk memproses penghitungan gaji karyawan yang biasanya penghitungan memakan waktu sampai 4 hari dengan pengerjaan secara *overtime*. Berikut ini adalah alur manual yang dilakukan oleh HRD RS Bedah Surabaya untuk penghitungan gaji karyawan:

1. Petugas HRD melakukan *retrieval data* setiap hari pada pukul 09:00.
2. Penghitungan absensi untuk bulan x dihitung sejak tanggal 25 pada bulan x-1 sampai pada tanggal 25 pada bulan x.

3. Terdapat *bug* pada program *fingerprint*, yaitu tidak bisa menghitung kehadiran untuk karyawan *shift* malam. Petugas harus memeriksa *data* mentah dari 200 lebih pegawai untuk memastikan tidak ada absensi yang terlewat.
4. Petugas mengolah *data* absensi di Excel dengan menginputkan satu persatu *data* kedalam *formula* yang telah di buat di Excel sehingga jadi laporan gaji pegawai pada bulan x.
5. Petugas membuat slip gaji secara manual dari Excel dengan mengganti nominal gaji dan nama pegawai sebanyak jumlah pegawai.
6. Gaji diberikan pada akhir bulan pada hari aktif yaitu senin – jumat. Apabila akhir bulan jatuh pada hari minggu, maka pemberian gaji akan dimajukan menjadi hari jum'at.

Solusi yang dapat menyelesaikan permasalahan diatas untuk membantu HRD RS Bedah Surabaya melakukan penghitungan gaji adalah sebuah aplikasi yang mampu *men-generate data* absensi dan melakukan penghitungan berdasarkan beberapa variabel yang telah ditetapkan sebelumnya sehingga menghasilkan laporan gaji yang diberikan untuk pegawai setiap bulannya.

*Document flow* pemrosesan penggajian pada RS Bedah Surabaya dapat dilihat pada gambar 3.1.



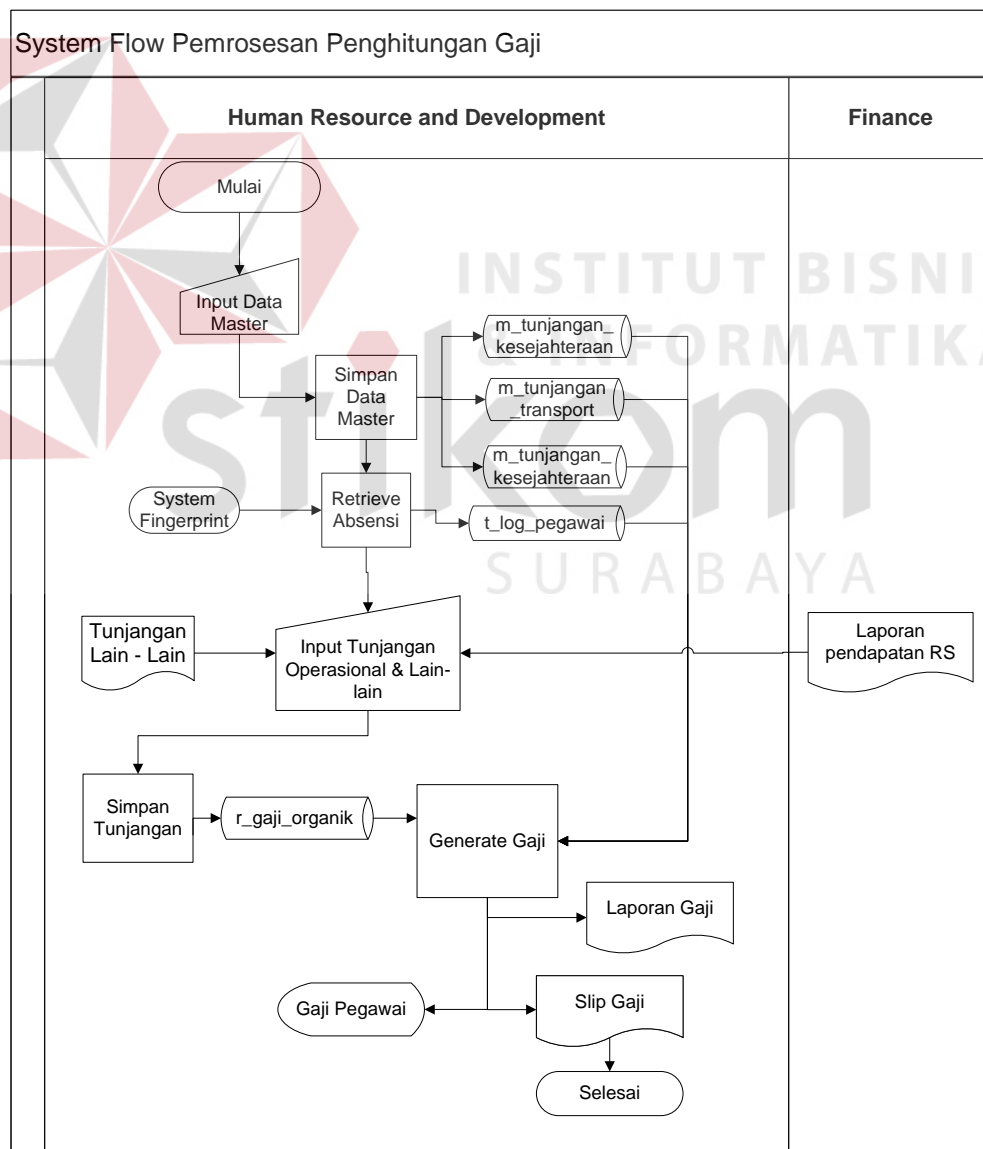
Gambar 3.1 Document Flow Proses Pemrosesan Gaji Secara Manual

### 3.1.2. Analisis Sistem

Sistem Informasi Penggajian Pegawai & Remunerasi Jasa Medis yang akan diimplementasikan pada RS Bedah Surabaya diharapkan membuat bagian HRD dan Keuangan dapat terbantu dalam menangani proses penggajian pegawai. Berikut adalah beberapa fitur yang dibutuhkan dalam sistem untuk menunjang HRD dan Keuangan dalam melakukan pengolahan gaji pegawai yaitu:

1. *Input variable* penghitungan gaji, seperti *data* pegawai, gaji pokok, tunjangan dan absensi.
2. Mengespor *data* dari *database fingerprint* kedalam *database* aplikasi.
3. Melakukan penghitungan gaji secara otomatis berdasarkan variabel penghitungan gaji yang telah di-*input*-kan sebelumnya.
4. Menghasilkan laporan slip gaji dan laporan remunerasi jasa medis.

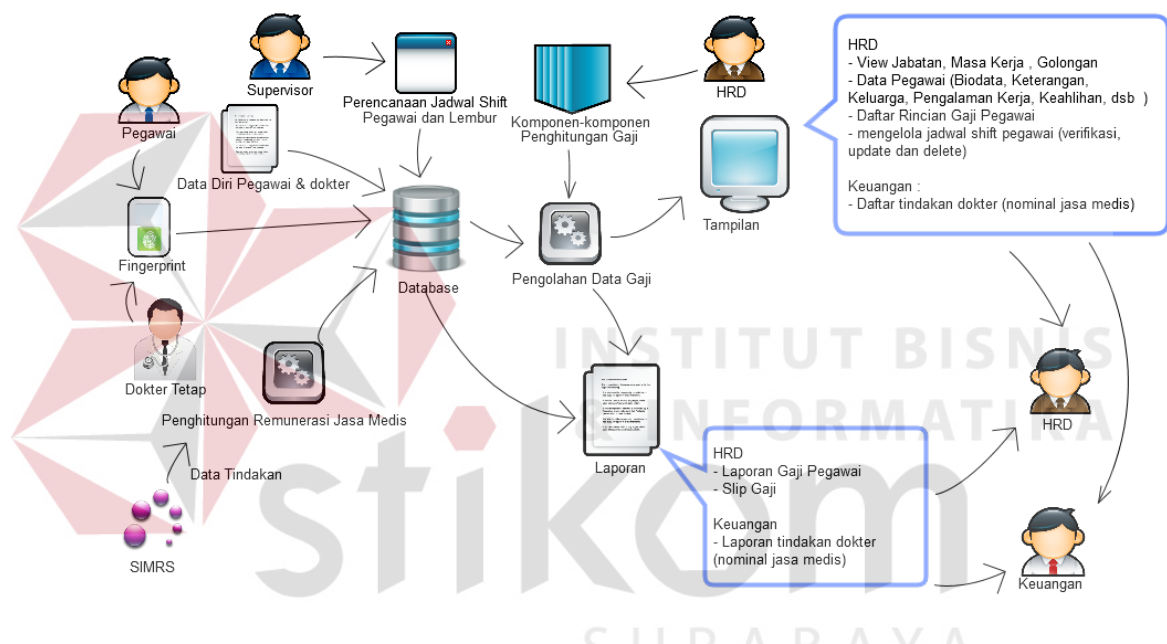
System flow pemrosesan gaji pada RSBS nantinya akan seperti pada gambar 3.2.



Gambar 3.2 System Flow Pemrosesan Gaji

### 3.2. Perancangan Sistem

Setelah mengetahui kebutuhan sistemnya, langkah selanjutnya adalah membuat perancangan dan desain sistem. Terdapat beberapa user dalam sistem ini yaitu Pegawai, Dokter, HRD, Manager dan Keuangan. Masing-masing user memiliki hak akses tersendiri dan memiliki *user stories* tersendiri. Gambaran umum dari Sistem Informasi Penggajian Pegawai dan Remunerasi Jasa Medis dapat dilihat pada gambar 3.3.



Gambar 3.3 Gambaran Umum Sistem Informasi Penggajian dan Remunerasi Jasa Medis

Berikut ini adalah *user stories* yang merupakan penjelasan detail dari gambaran umum sistem. *User stories* dari aplikasi ini dapat dilihat pada tabel 3.1

Tabel 3.1 User Stories

No	Story
U01	Sebagai <i>User</i> , saya harus login terlebih dahulu untuk menggunakan aplikasi.
U02	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> Pegawai.
U03	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> di menu Master Anak.

Tabel 3.1 User Stories (Lanjutan)

U04	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> di menu Master Pengalaman Kerja.
U05	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> di menu Master History Pendidikan.
U06	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> di menu Master Keterangan.
U07	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> unit
U08	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> jabatan
U09	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> golongan
U10	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> masa kerja
U11	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> shift
U12	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> status log
U13	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> tunjangan
U14	Sebagai <i>User</i> , saya bisa menambah, mengedit, dan menghapus <i>data</i> pada menu alokasi pegawai
U15	Sebagai <i>User</i> , saya bisa membuat perencanaan jadwal shift
U16	Sebagai <i>User</i> , saya bisa melakukan verifikasi atas perencanaan jadwal
U17	Sebagai <i>User</i> , saya bisa melakukan impor <i>data fingerprint</i> pegawai.
U18	Sebagai <i>User</i> , saya bisa melakukan verifikasi atas lembur pegawai
U19	Sebagai <i>User</i> , saya bisa menginput nilai tunjangan non-generate
U20	Sebagai <i>User</i> , saya bisa men-generate gaji pegawai
U21	Sebagai <i>User</i> , saya bisa mencetak slip gaji
U22	Sebagai <i>User</i> , saya bisa melihat nominal remunerasi medis
U23	Sebagai <i>User</i> , saya bisa mencetak nominal remunerasi medis
U24	Sebagai <i>User</i> , saya bisa mencetak report gaji pegawai
U25	Sebagai <i>User</i> , saya bisa <i>Log Out</i> dari sistem.

Terdapat 5 user yang terhubung langsung dengan sistem ini yaitu HRD, Supervisor, *Manager*, *Accounting* dan *Administrator*. Berikut ini adalah penjelasan dari tiap *user*:

1. HRD : HRD adalah unit yang mengelola sumber daya manusia pada RS Bedah Surabaya. User inilah yang paling banyak terlibat dalam proses transaksi dalam sistem ini karena memang sistem ini dibuat untuk

memudahkan HRD dalam mengolah *data* pegawai dari absensi hingga mencetak slip gaji.

2. Supervisor : Supervisor adalah pimpinan dalam sebuah unit kecil yang bertugas untuk mengkoordinasi *staff*. Didalam sistem ini supervisor hanya bertugas untuk mengatur jadwal shift pegawai.
3. Manager : *Manager* adalah pimpinan dalam setiap departemen. Tugas *manager* adalah untuk melakukan verifikasi dari penjadwalan shift yang telah dibuat oleh supervisor.
4. Accounting : Accounting adalah departemen yang bertugas memanajemen keuangan dalam RS Bedah Surabaya. tugas *accounting* salah satunya adalah melakukan pembayaran untuk dokter, baik dokter tetap maupun dokter tamu. Dari sana lah dibutuhkan sebuah sistem yang mampu menampilkan nominal jasa medis untuk mempermudah pekerjaan *accounting*.
5. Administrator : Administrator adalah unit yang ditunjuk oleh pihak manajemen pada RS Bedah Surabaya untuk mengatur keseimbangan sistem. User ini memiliki hak akses untuk masuk ke semua menu yang dimiliki oleh sistem.

Dari *user stories* yang ada, serta untuk memberikan solusi dari permasalahan yang ada, maka didapatkan modul-modul yang dibutuhkan dalam membangun sistem ini yaitu:

#### 1. Login

Modul login merupakan gerbang awal dari sistem dimana setiap pengguna harus melakukan login terlebih dahulu untuk mengakses sistem karena hak akses dari tiap pengguna berbeda-beda.

## 2. Master

Modul ini terdiri dari beberapa sub modul lagi yaitu Master Pegawai, Master Anak, Master History, Master Pengalaman Kerja, Master Pendidikan, Master Keterangan, Master Unit, Master Jabatan, Master Gaji, Master Shift, Master Gaji, Master Golongan, Master Masa Kerja, Master Shift, Master Status Log, Master Tunjangan Transport, Master Tunjangan Kesejahteraan, Tunjangan Jabatan dan Master Alokasi Pegawai. Modul ini berfungsi sebagai modul untuk menyiapkan berbagai macam *data* yang nantinya dibutuhkan oleh modul yang lain untuk memproses berbagai transaksi yang ada.

## 3. Penjadwalan Shift Pegawai

Dalam modul ini pengguna bisa melakukan penjadwalan *shift* untuk satu bulan kedepan dan dilakukan pada akhir bulan yang telah lewat dan perubahan *shift* dilakukan kapan saja asal ada otorisasi terlebih dahulu dari atasan masing-masing *unit*

## 4. Verifikasi Jadwal Shift

Setelah penjadwalan selama sebulan kedepan telah dibuat, maka atasan melakukan verifikasi pada modul ini perihal kelayakan dari hasil perencanaan jadwal *shift*.

## 5. Export Log Pegawai

Modul *export* adalah menarik *data* dari *database fingerprint* ke dalam *database* sistem ini. *Export log* akan dilakukan setiap hari oleh HRD agar *data* absensi pegawai bisa dipantau secara *up to date*



## 6. Verifikasi Overtime

Waktu pulang pegawai sangat bervariasi meskipun telah ditetapkan dalam kebijakan rumah sakit tentang jumlah jam kerja pegawai perharinya. Bisa saja pegawai tidak langsung pulang ketika jam pulang telah berakhir karena ada kepentingan selain kepentingan pekerjaan. Hal seperti inilah yang ditanggulangi dengan keberadaan modul ini. Dengan adanya verifikasi overtime, HRD bisa membedakan kelebihan jam kerja mana yang termasuk lembur atau tidak.

## 7. Penghitungan Gaji

Modul ini adalah modul utama dalam sistem dimana *data-data* yang telah tersedia diolah untuk menghitung gaji yang diterima oleh masing-masing pegawai pada bulan tersebut. Output yang dihasilkan dalam penghitungan ini akan berbentuk *Report* yang nantinya akan di-*generate* ke bentuk *Portable Document Format* (PDF) sehingga nantinya bisa dicetak.

### 3.2.1 Planning Iteration Phase

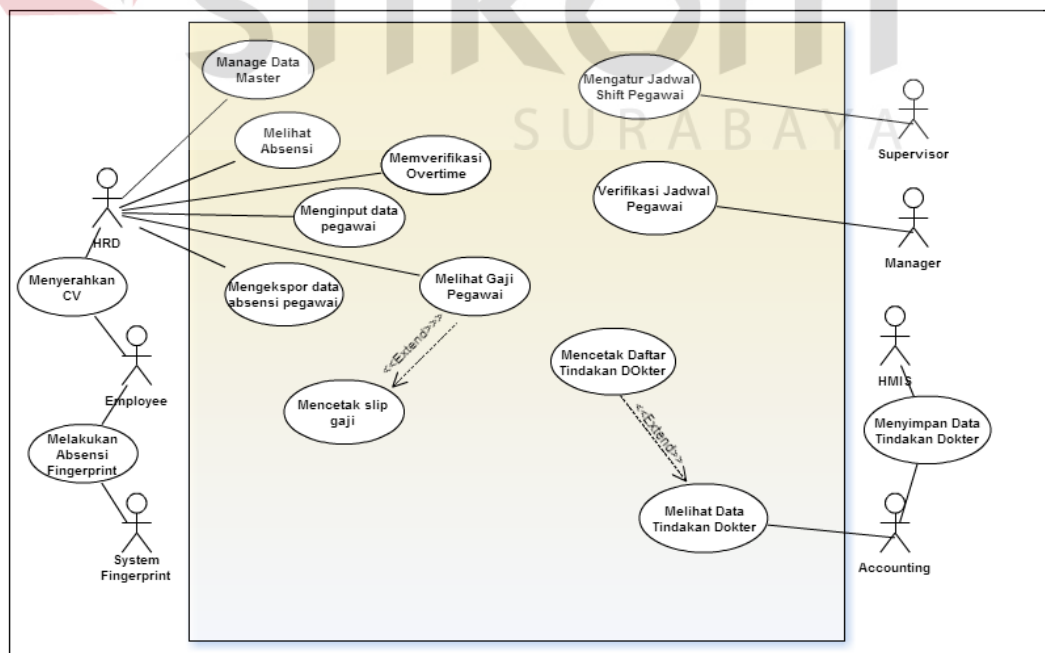
Setelah diketahui modul yang akan dibuat, maka disusunlah prioritas pengembangan modul. Jadwal pengerjaan modul-modul tersebut dibagi menjadi beberapa iterasi. Jadwal iterasi ditunjukkan pada tabel 3.2.

Tabel 3.2 Jadwal Iterasi

Iterasi	Modul	Implementasi
1	Login	<i>User stories</i> U1 dan U25
2	Master	<i>User stories</i> U2, U3, U4, U5, U6, U7, U8, U9, U10, U11, U12, U13, U14, dan U24
3	Penjadwalan Shift Pegawai	<i>User stories</i> U15
4	Verifikasi Jadwal Shift	<i>User stories</i> U16
5	Import Log Pegawai	<i>User stories</i> U17
6	Verifikasi Overtime	<i>User stories</i> U18
7	Penghitungan Gaji	<i>User stories</i> U19, U20, U21, U22, dan U23

### 3.2.2 Iteration to Release Phase

Runtutan pengerjaan aplikasi akan disesuaikan dengan urutan dari iterasi yang telah direncanakan. Selanjutnya penjelasan dalam tiap modul akan jabarkan dalam bentuk UML. Salah satu diagram UML yang akan digunakan adalah diagram *use case* dimana pada diagram ini digambarkan hubungan antara aktor dengan proses bisnis yang ada. Diagram *use case* dari sistem ini dapat dilihat pada gambar 3.4. Pada gambar tersebut, *use case* yang berada diluar kotak menandakan bahwa pekerjaan tersebut berlangsung di luar sistem. *Use case* yang terdapat dalam lingkup sistem bisa diartikan *use case* tersebut adalah beberapa fitur yang bisa diakses oleh berbagai *actor* yang terdapat pada *use case* tersebut. Pada *use case* melihat gaji pegawai dan *use case* melihat data tindakan dokter, aktor yang terkait dibebaskan untuk sekaligus menjalankan *use case* lain yang terhubung pada *use case* tersebut yaitu *use case* mencetak slip gaji atau *use case* mencetak daftar tindakan dokter.



Gambar 3.4 Use Case Diagram Sistem Informasi Penggajian & Remunerasi Jasa Medis

Selanjutnya penjelasan dari tiap-tiap modul yang telah dijelaskan sebelumnya akan dibahas menggunakan *use case diagram*, *flow of events*, *sequence diagram*, *class diagram*, *interface design* dan *unit testing*. Penjelasan dari modul-modul tersebut adalah sebagai berikut:

### 1. Iterasi Login

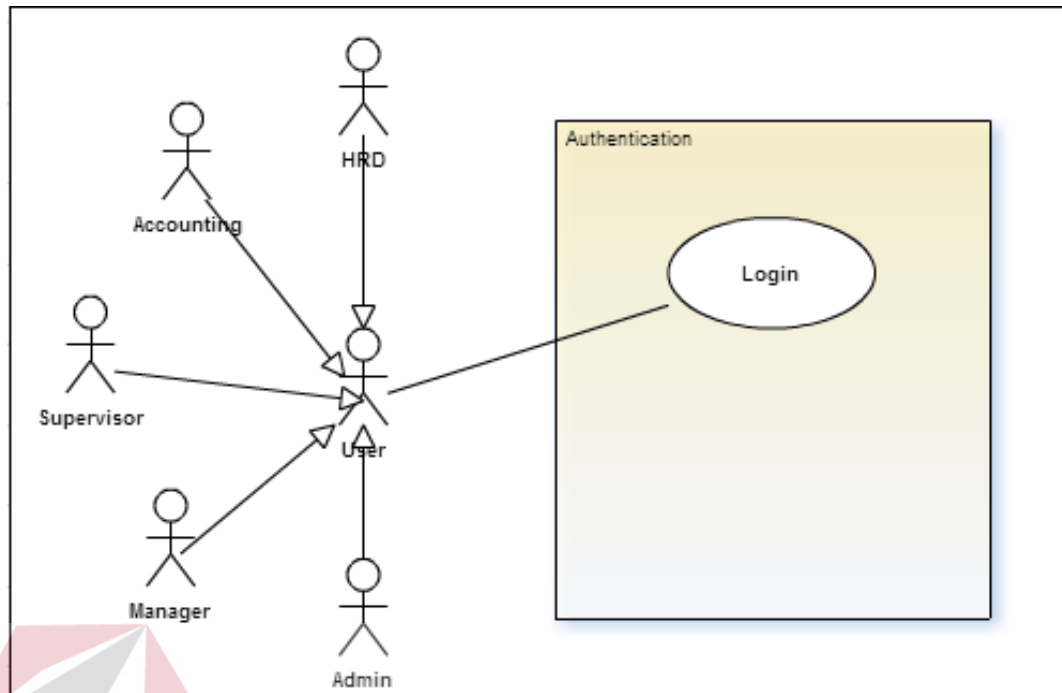
Dalam iterasi ini akan dijelaskan bagaimana seorang pengguna melakukan login dan sistem melakukan autentikasi. Pada tabel 3.3 akan ditampilkan tentang hak akses dari setiap pengguna dalam modul login ini.

Tabel 3.3 Hak Akses Modul Login

	HRD	Supervisor	Manager	Accounting	Admin
Login	√	√	√	√	√

#### A. Use Case Diagram Modul Login

Pada gambar 3.5 merupakan gambar dari diagram *use case* login dimana terdapat 5 aktor yaitu HRD, *Accounting*, *Supervisor*, *Manager* dan Admin yang memiliki sifat yang sama, kemudian digeneralisasi menjadi aktor *User*.



Gambar 3.5 Diagram Use Case Login

### B. Flow of Events Modul Login

Flow of event untuk *use case* login dapat dilihat pada tabel 3.4.

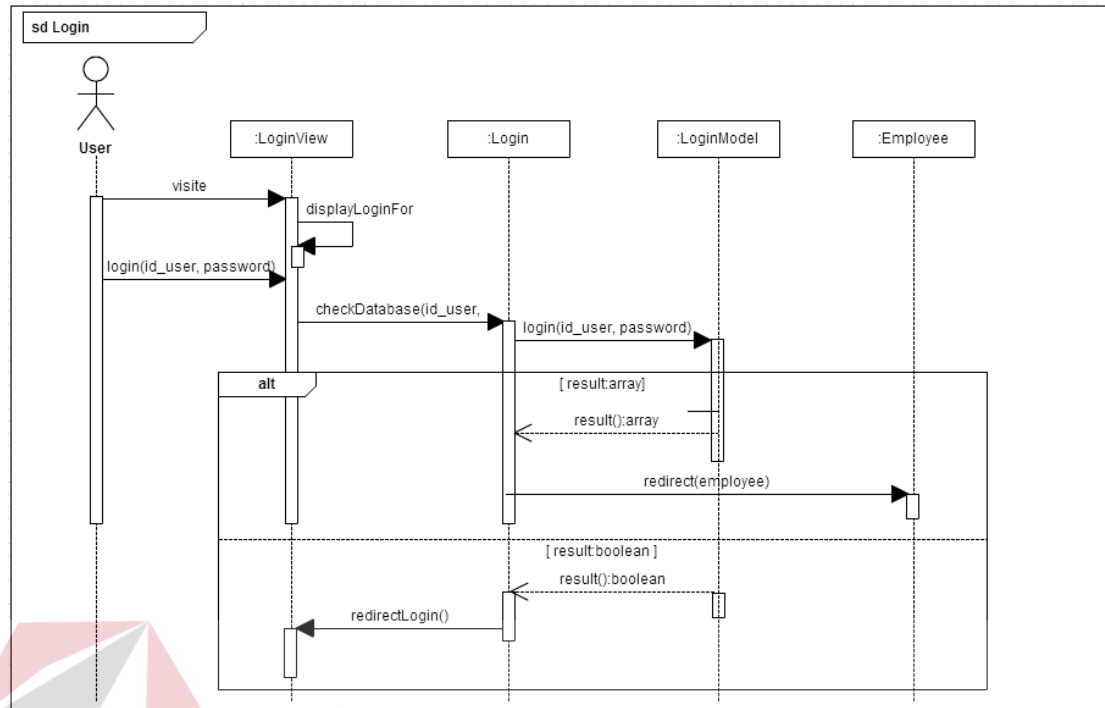
Tabel 3.4 Flow of Event Login

Deskripsi <i>Use Case Detail</i>	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> ini digunakan untuk melakukan autentikasi terhadap use yang akan masuk, dan disini juga akan diatur bagaimana hak akses masing-masing pengguna	
<b>Tujuan</b>	Pengguna bisa masuk kedalam Sistem Informasi Penggajian Pegawai dan Remunerasi Jasa Medis	
<b>Prasyarat</b>	Tidak ada	
<b>Kondisi akhir Sukses</b>	Pengguna bisa masuk ke dalam aplikasi	
<b>Kondisi akhir gagal</b>	Pengguna tidak bisa masuk ke aplikasi karena kesalahan <i>username</i> , kesalahan <i>password</i> , atau kesalahan <i>username</i> dan <i>password</i>	
<b>Aktor utama</b>	User	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	Pengguna berada pada halaman login aplikasi	
<b>Alur Utama</b>		
	1	Pengguna memasukkan kombinasi <i>username</i> dan <i>password</i> pada form login

<b>Deskripsi Use Case Detail</b>	<b>Keterangan</b>	
<b>Deskripsi</b>	<i>Use case</i> ini digunakan untuk melakukan autentikasi terhadap use yang akan masuk, dan disini juga akan diatur bagaimana hak akses masing-masing pengguna	
	2	Sistem mengirimkan <i>data user</i> dan melakukan pencocokan dengan <i>database</i> . Jika terjadi <i>error</i> pada koneksi <i>database</i> maka akan masuk ke langkah 2.2. jika kombinasi <i>username</i> dan <i>password</i> salah, maka akan masuk ke langkah 2.1.
	3	Sistem akan memindah halaman ke <i>home.php</i>
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	2.1	Sistem akan mengembalikan ke halaman awal dan kembali ke langkah 1
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	2.2	Sistem akan mengembalikan ke halaman awal dan kembali ke langkah 1

### C. Sequence Diagram Modul Login

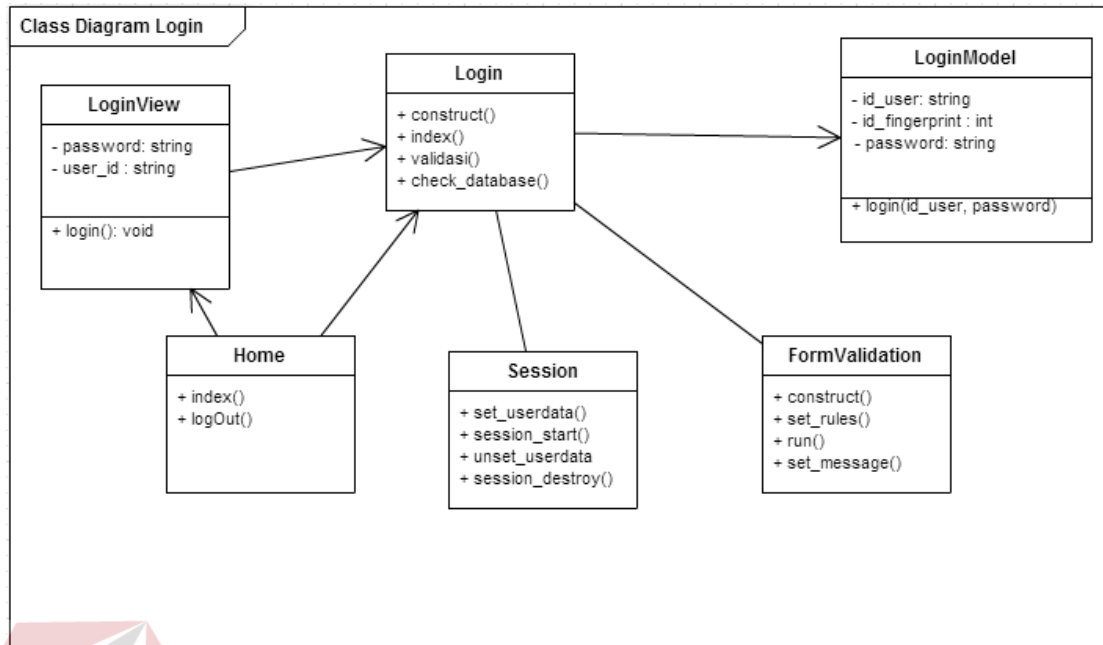
Sequence diagram untuk *use case login* dapat dilihat pada gambar 3.6. Pada gambar tersebut dijelaskan alur kemungkinan yang terjadi yang memungkinkan bisa atau tidaknya pengguna masuk ke dalam sistem. Apabila kolom *username* dan *password* sesuai dengan *data* yang terdapat pada *database*, maka pengguna akan dibawa pada tampilan halaman awal.



Gambar 3.6 Sequence Diagram pada Use Case Login

#### D. Class Diagram Modul Login

Relasi antar *class* pada modul *login* ditunjukkan oleh gambar 3.7. Pada gambar tersebut, terdapat beberapa *class* untuk pengamanan login yaitu Class Session, dimana Class Session memiliki fungsi untuk menyimpan sesi login sehingga meskipun aplikasi tidak sengaja tertutup, apabila sesi penggunaan aplikasi masih ada, aplikasi akan otomatis bisa diakses kembali tanpa melakukan login kembali. Sebaliknya apabila aplikasi tidak digunakan dalam jangka waktu yang lama, Class Session akan mengakhiri sesi tersebut dengan melakukan aksi *log out* apabila aplikasi tersebut dijalankan kembali.



Gambar 3.7 Class Diagram pada Use Case Login

### E. Desain Interface Modul Login

Desain *interface* untuk *login* terdiri dari dua *textbox* yaitu *id\_user* dan *password*. Untuk keamanan passwordnya sendiri, setiap passwordnya akan di-generate dalam bentuk MD5 sehingga password tidak bisa diketahui meskipun ada seseorang yang bisa melihat isi kolom password pada *database*. Desain *interface login* ditunjukkan pada gambar 3.8.

**Silahkan Login**

Username

Password

Gambar 3.8 Desain Interface Login

## F. Unit Testing Modul Login

Skenario tes pada modul login adalah dengan mengimplementasikan fitur unit testing yang dimiliki oleh framework code igniter untuk menguji fungsionalitas dari modul login ini. Skenario testing pada modul login dapat dilihat pada tabel 3.5.

Tabel 3.5 Output unit tes class loginTest

No	Tes
1	loginTest(\$username = id_user, \$password = password)
2	loginTest(\$username != id_user, \$password = password)
3	loginTest(\$username = id_user, \$password != password)
3	loginTest(\$username != id_user, \$password != password)

### 2. Iterasi Master

Didalam modul master sendiri terdapat beberapa sub-modul lagi yaitu Master Employee, Master Anak, Master Pengalaman Kerja, Master History Pendidikan, Master Keterangan, Master Unit, Master Jabatan, Master Gaji, Master Golongan, Master Masa Kerja, Master Shift, Master Status Log, Master Tunjangan serta Master Alokasi Pegawai. Pada tabel 3.6 akan ditampilkan mengenai hak akses dari setiap pengguna dalam modul master ini.

Tabel 3.6 Hak Akses Modul Master

	HRD	Supervisor	Manager	Accounting	Admin
M.Employee	√				√
M.Anak	√				√
M.Pengalaman Kerja	√				√
M.History Pendidikan	√				√
M.Keterangan	√				√
M.Unit	√				√
M.Jabatan	√				√
M.Gaji	√				√
M.Golongan	√				√
M.Masa Kerja	√				√

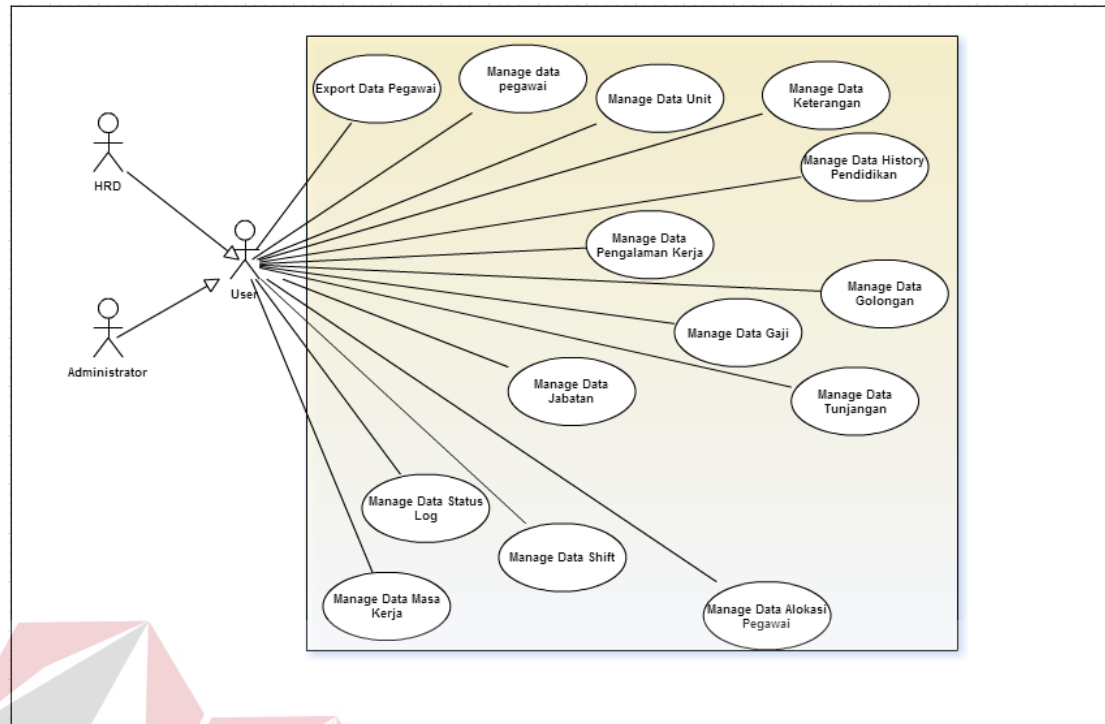


Tabel 3.6 Hak Akses Modul Master (lanjutan)

	HRD	Supervisor	Manager	Accounting	Admin
M.Shift	√				√
M.Status Log	√				√
M.Tunjangan Transport	√				√
M.Tunjangan Kesejahteraan	√				√
Alokasi Pegawai	√				√

### A. Diagram Use Case Modul Master

Berikut ini adalah diagram *use case* dari modul master yang dapat dilihat pada gambar 3.9. Dalam gambar tersebut bisa dilihat terdapat 2 aktor disana yang memiliki sifat yang sama sehingga digeneralisasi menjadi aktor *User* serta memiliki 14 *use case* yang berinteraksi dengan *User*. *Use case* tersebut yaitu, *use case export data pegawai*, *manage data pegawai*, *manage data unit*, *manage data keterangan*, *manage data history pendidikan*, *manage data pengalaman kerja*, *manage data golongan*, *manage data gaji*, *manage data tunjangan*, *manage data jabatan*, *manage data status log*, *manage data shift*, *manage data masa kerja* dan *manage data alokasi pegawai*. Apabila diperhatikan pada *use case* tersebut terdapat kata "manage" pada awalan nama *use case* yang berarti bahwa tiap *use case* memiliki proses dasar yang lazimnya terdapat pada modul master yaitu insert, update, delete & view.



Gambar 3.9 Diagram Use Case Modul Master

### A. Flow of Events Modul Master

Pada dasarnya ada 5 proses yang terdapat pada modul master ini yaitu *insert*, *update*, *delete*, *view* dan *export data* pegawai. *Flow of event* berikut ini akan menjelaskan tentang alur proses pada modul master.

#### a. Flow of Event Insert

*Flow of Event* untuk *insert* dapat dilihat pada tabel 3.7.

Tabel 3.7 Flow of Event Insert Pada Modul Master

Deskripsi Use case Detail	Keterangan
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>insert data</i>
<b>Tujuan</b>	<i>User</i> bisa menambahkan <i>data</i> ke <i>database</i> melalui aplikasi
<b>Prasyarat</b>	Login
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa menambahkan <i>data</i> .
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka <i>data</i> tidak akan ditampilkan di <i>View</i>
<b>Aktor utama</b>	<i>User</i>

Tabel 3.7 Flow of Event Insert Pada Modul Master (Lanjutan)

Deskripsi Use case Detail	Keterangan	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman master dari <i>data</i> yang akan ditambahkan	
<b>Alur Utama</b>		
	1	Use case dimulai ketika <i>user</i> berada pada form master.
	2	<i>Use case</i> dimulai ketika <i>user</i> mengisi teks box yang telah tersedia
	3	User menekan tombol "submit"
	4	Sistem akan melakukan koneksi <i>database</i> dan memasukkan <i>data</i> yang telah di- <i>input</i> . jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

## b. Flow of Event Update

*Flow of event* untuk melakukan proses *update* dapat dilihat pada tabel 3.8.

Tabel 3.8 Flow of Event Update Pada Modul Master

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>update data</i>	
<b>Tujuan</b>	<i>User</i> bisa mengubah <i>data</i> pada <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa mengubah <i>data</i> .	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka <i>data</i> tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman master dari <i>data</i> yang akan diubah.	
<b>Alur Utama</b>		
	1	Use case dimulai ketika <i>user</i> berada pada form master.

Tabel 3.8 Flow of Event Update Pada Modul Master (Lanjutan)

Deskripsi Use case Detail	Keterangan	
	2	<i>Use case</i> dimulai ketika <i>user</i> masuk kedalam mode view dan menekan tombol <input type="checkbox"/> didalam kolom "Actions" pada tabel yang terdapat di Form Master
	3	User mengubah <i>data</i> yang tampil pada teks boks yang tersedia.
	4	User menekan tombol "submit".
	5	Sistem akan melakukan koneksi <i>database</i> dan meng- <i>update data</i> yang telah diubah. jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 5.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.


## c. Flow of Event Delete

Flow of event untuk proses delete pada modul master dapat dilihat pada tabel 3.9.

Tabel 3.9 Flow of Event Delete Pada Modul Master

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>delete data</i>	
<b>Tujuan</b>	<i>User</i> bisa menghapus <i>data</i> pada <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa menghapus <i>data</i> .	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka <i>data</i> tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman master dari <i>data</i> yang akan dihapus	
<b>Alur Utama</b>		
	1	<i>Use case</i> dimulai ketika <i>user</i> berada pada form master.

Tabel 3.9 Flow of Event Delete Pada Modul Master (Lanjutan)

Deskripsi Use case Detail	Keterangan	
	2	<i>Use case</i> dimulai ketika <i>user</i> masuk kedalam mode view dan menekan tombol  didalam kolom "Actions" pada tabel yang terdapat di Form Master
	3	User menekan tombol "Oke" pada dialog yang muncul.
	4	Sistem akan melakukan koneksi <i>database</i> dan menghapus <i>data</i> sesuai id yang dituju. jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

## d. Flow of Event Export Data

Flow of event dari proses export *data* dapat dilihat pada tabel 3.10.

Tabel 3.10 Flow of Event Export Data

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>export data</i>	
<b>Tujuan</b>	<i>User</i> bisa memindah <i>data</i> dari <i>database fingerprint</i> ke <i>database</i> aplikasi.	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa mengimpor <i>data</i> .	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka <i>data</i> tidak akan ditampilkan di View	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman <i>Master Employee</i> .	
<b>Alur Utama</b>		
	1	<i>Use case</i> dimulai ketika <i>user</i> berada pada form <i>Master Employee</i> .
	2	<i>Use case</i> dimulai ketika <i>user</i> menekan tombol "Export Data"

Tabel 3.10 Flow of Event Export *Data*

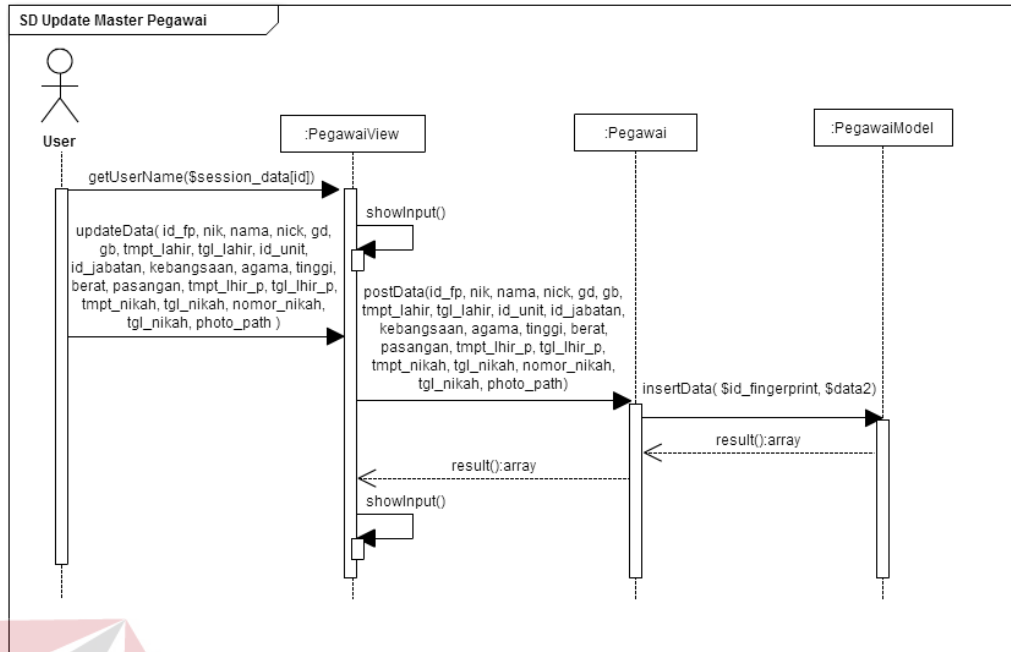
Deskripsi Use case Detail	Keterangan	
	3	Sistem akan melakukan koneksi <i>database</i> , melakukan proses <i>select</i> dari <i>database</i> sistem <i>fingerprint</i> dan melakukan proses insert ke <i>database</i> SIPP. Jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 3.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	3.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

## B. Sequence Diagram Modul Master

Berikut ini adalah *sequence diagram* tiap *form* yang terdapat dalam modul master yaitu sebagai berikut:

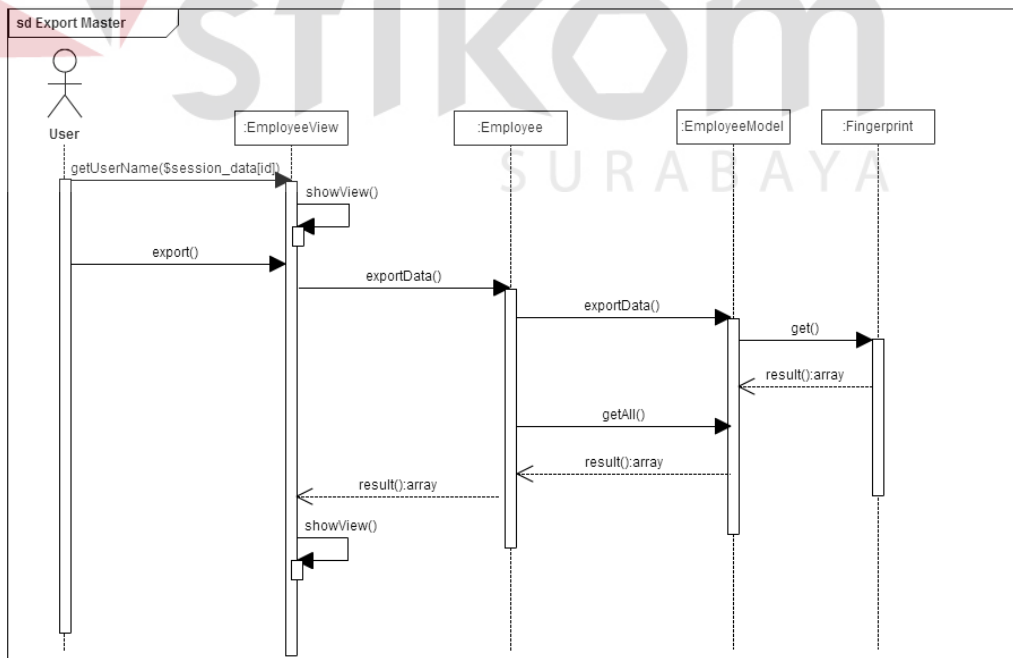
### a. Master Pegawai

*Sequence diagram* untuk *use case insert data master* pada *master pegawai* terdapat pada gambar 3.10. Pengguna melengkapi data karyawan pada proses ini karena data yang didapat pada proses impor hanya berupa id fingerprint, nama, dan gelar.



Gambar 3.10 Sequence Diagram Update pada Master Pegawai

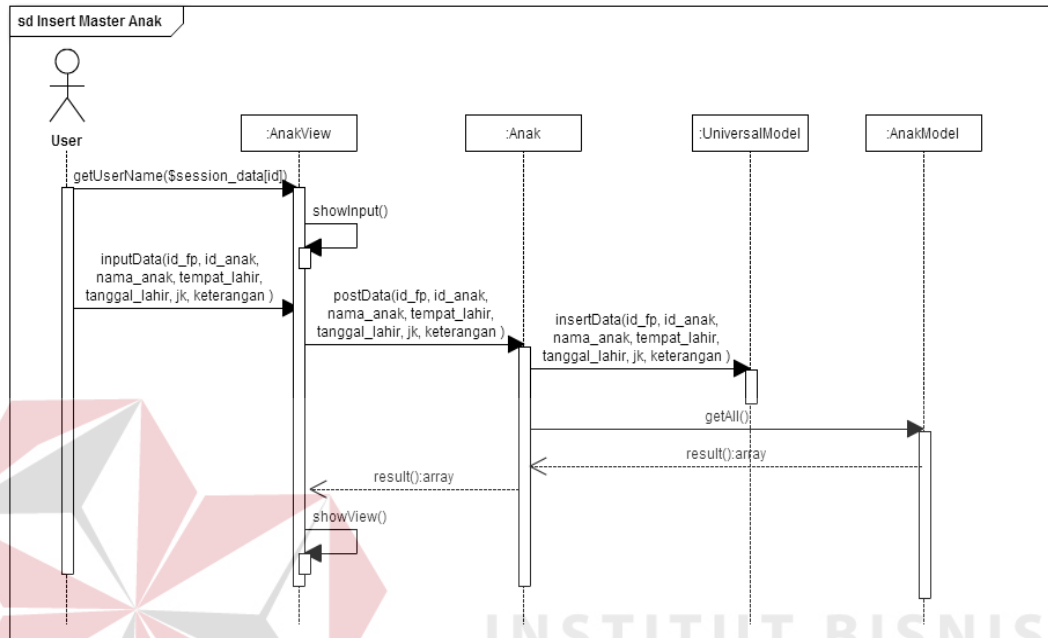
Sequence diagram proses *export data* pada *Master Pegawai* dapat dilihat pada gambar 3.11. Class *EmployeeModel* pada gambar tersebut berisi *function* yang berfungsi mengambil data dari *database* sistem *fingerprint*.



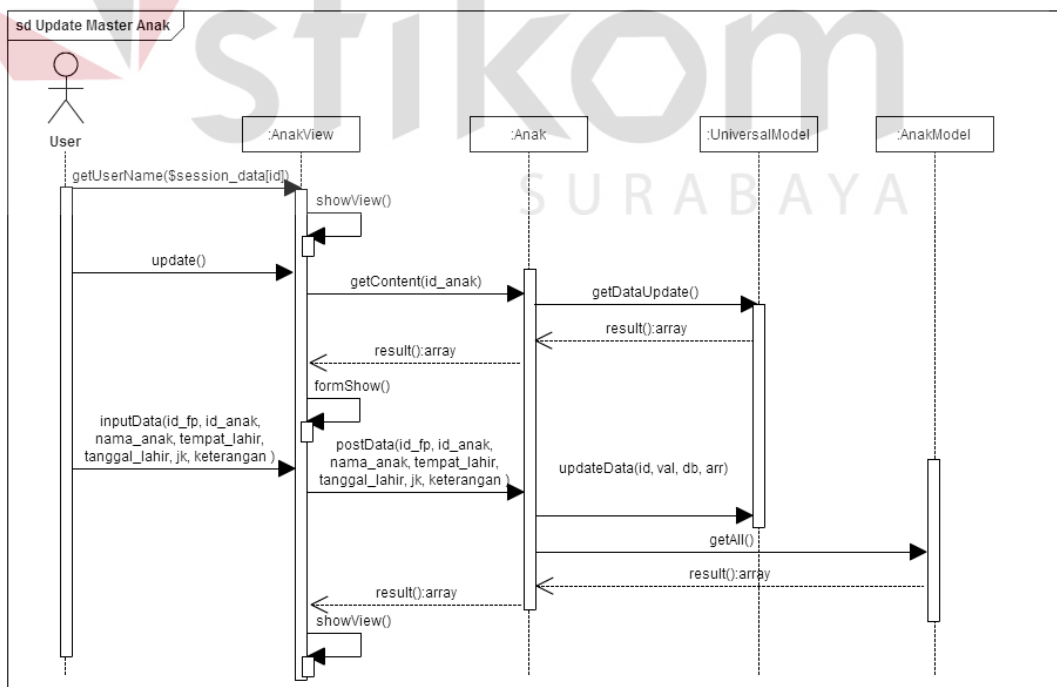
Gambar 3.11 Sequence Diagram pada Use Case Export Data Master

b. Master Anak

Sequence diagram untuk proses insert, update dan delete pada master anak terdapat pada gambar 3.12, gambar 3.13, dan gambar 3.14.

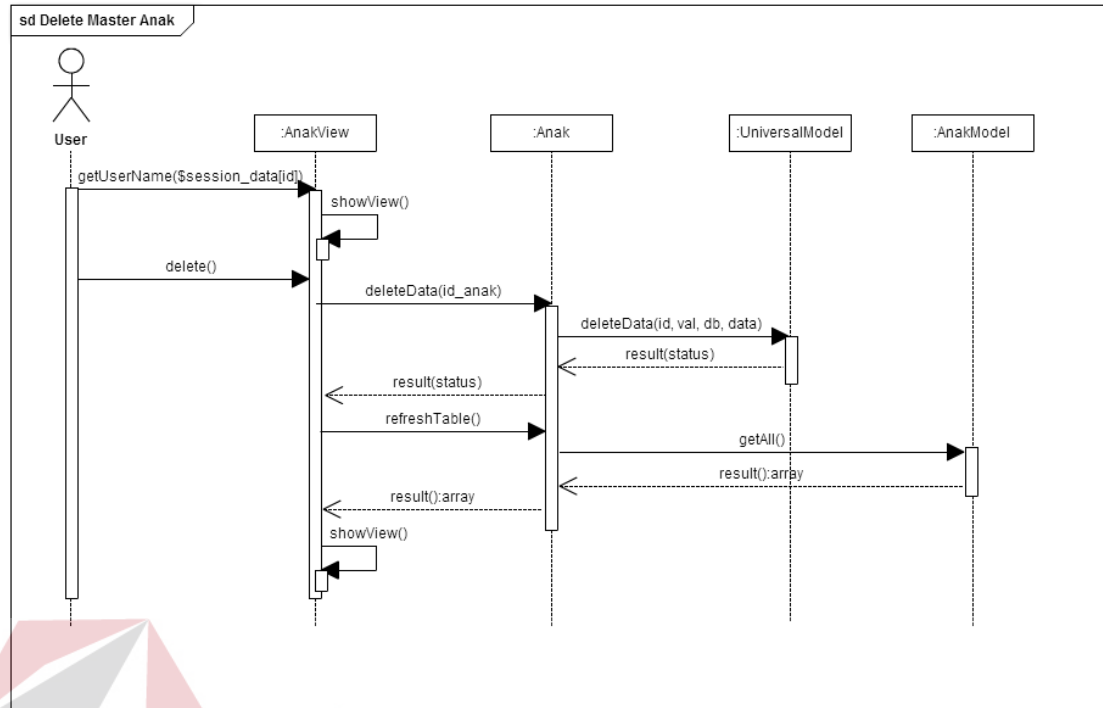


Gambar 3.12 Sequence Diagram Proses Insert Data pada Master Anak



Gambar 3.13 Sequence Diagram Proses Update Data pada Master Anak



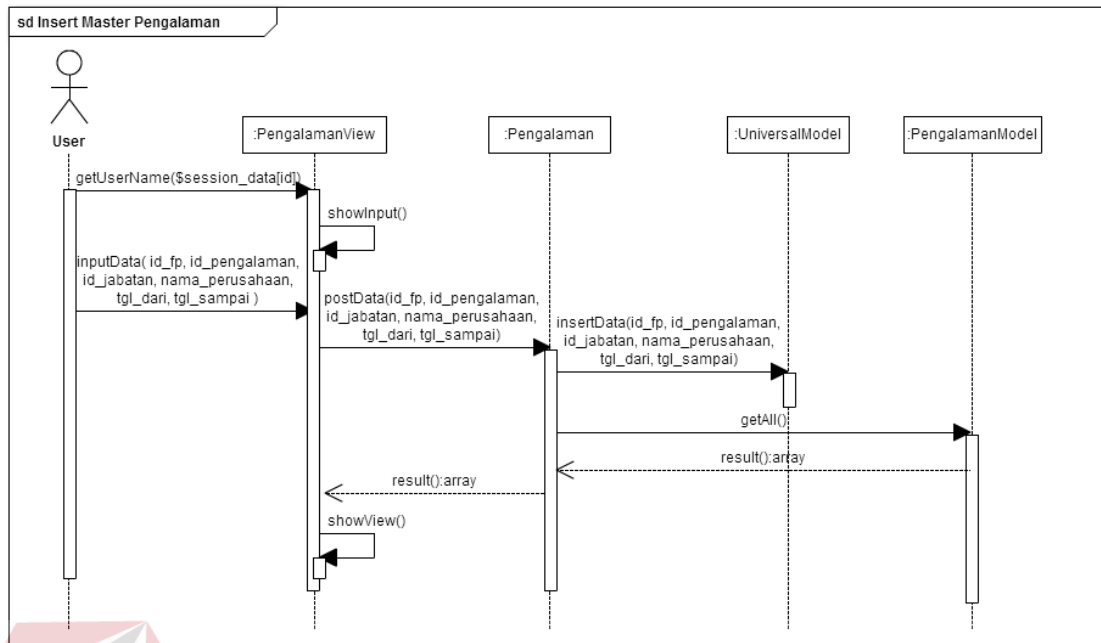


Gambar 3.14 Sequence Diagram Proses Delete pada Master Anak

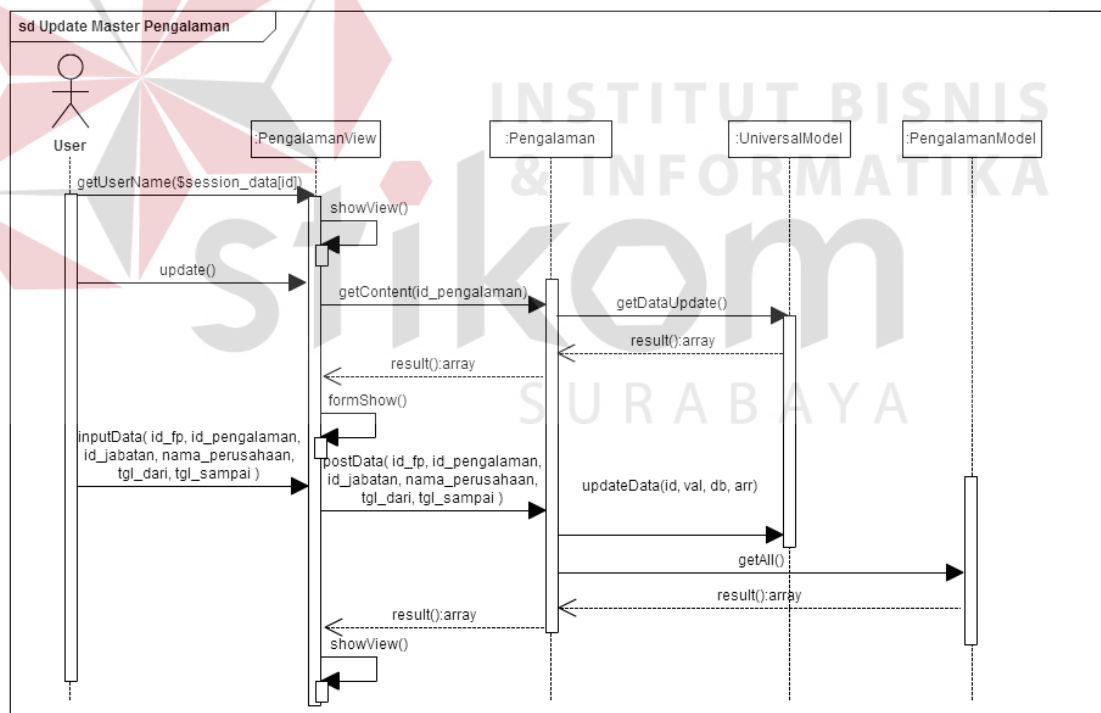
Variabel yang menjadi inputan pada proses insert dan Update pada Master Anak adalah `id_fingerprint`, `id_anak`, `nama_anak`, `tempat_lahir`, `tanggal_lahir`, `jk` (jenis kelamin) dan keterangan. Variabel `id_anak` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

### c. Master Pengalaman

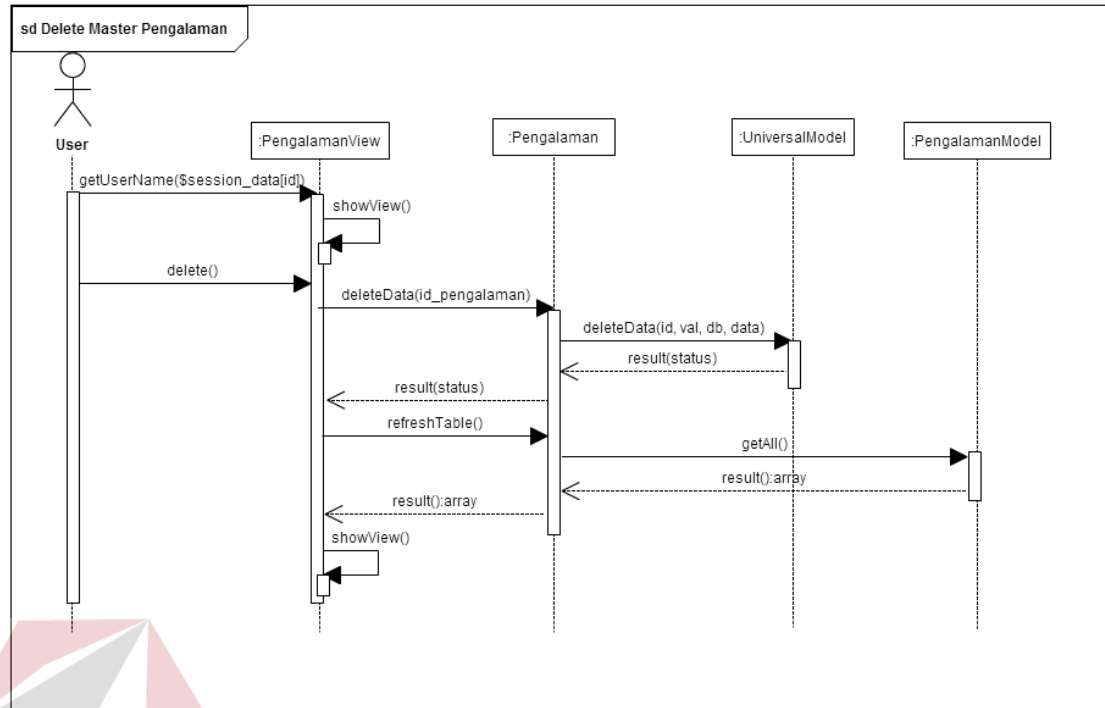
*Sequence diagram* untuk proses insert, update dan delete pada *master* pengalaman terdapat pada gambar 3.15, gambar 3.16, dan gambar 3.17.



Gambar 3.15 Sequence Diagram Proses Insert Data pada Master Pengalaman



Gambar 3.16 Sequence Diagram Proses Update Data pada Master Pengalaman

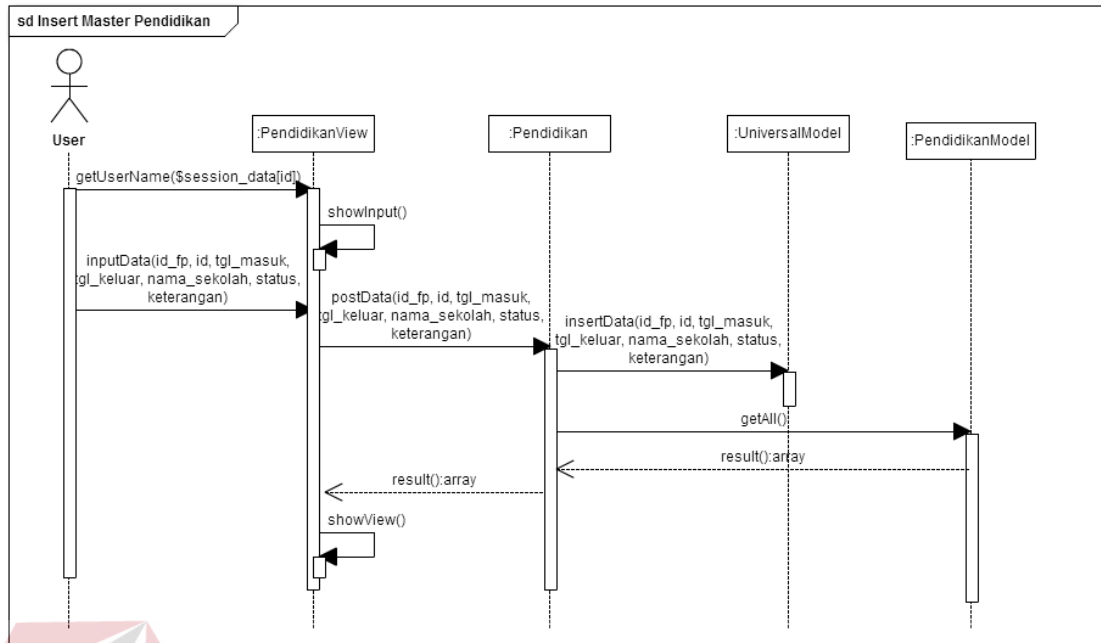


Gambar 3.17 Sequence Diagram Proses Delete Data pada Master Pengalaman

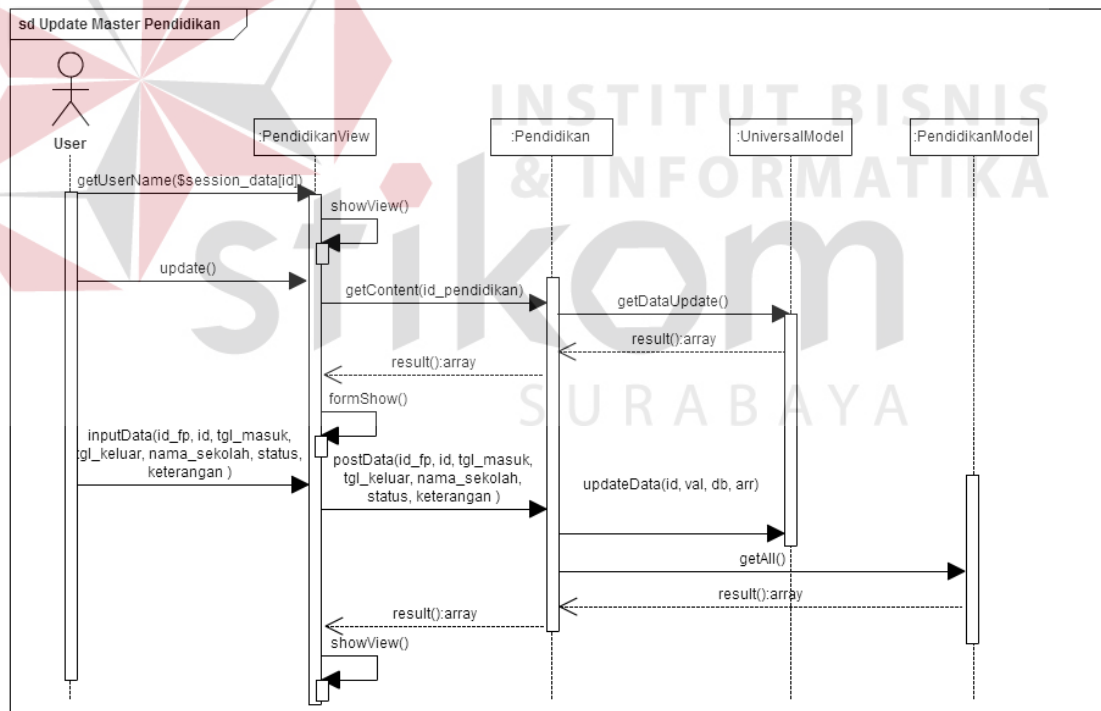
Variabel yang menjadi inputan pada proses insert dan Update pada Master Pengalaman adalah `id_fingerprint`, `id_pengalaman`, `id_jabatan`, `nama_perusahaan`, `tgl_dari` dan `tgl_sampai`. Variabel `id_pengalaman` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

#### d. Master Pendidikan

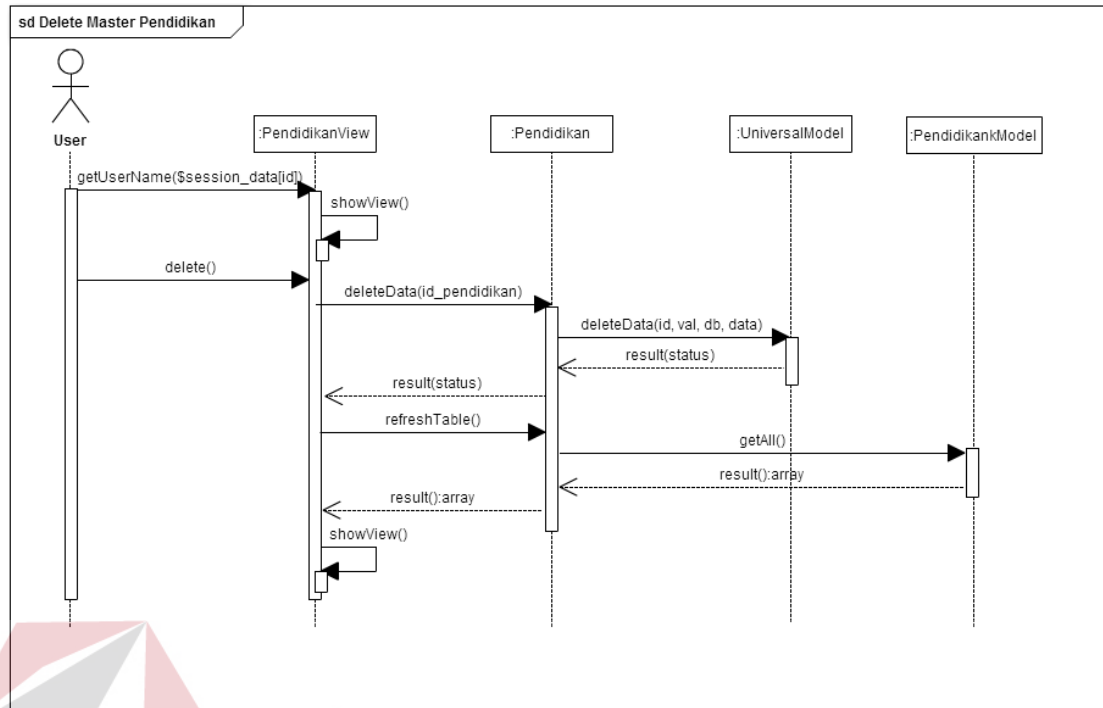
*Sequence diagram* untuk proses insert, update dan delete pada *master pendidikan* terdapat pada gambar 3.18, gambar 3.19, dan gambar 3.20.



Gambar 3.18 Sequence Diagram Proses Insert Data pada Master Pendidikan



Gambar 3.19 Sequence Diagram Proses Update Data pada Master Pendidikan

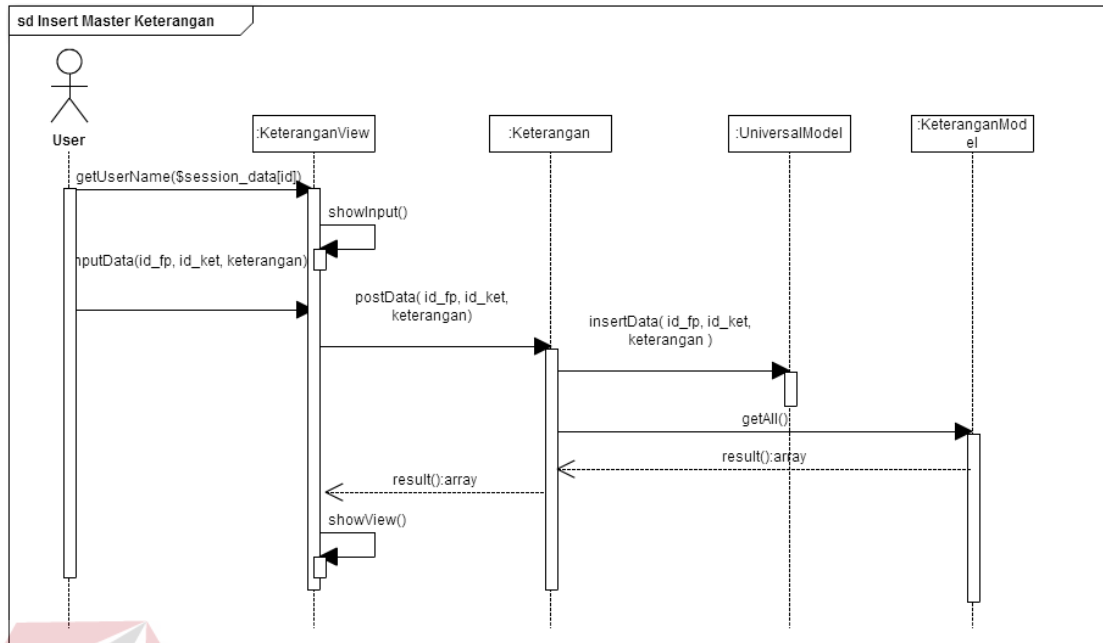


Gambar 3.20 Sequence Diagram Proses Delete Data pada Master Pendidikan

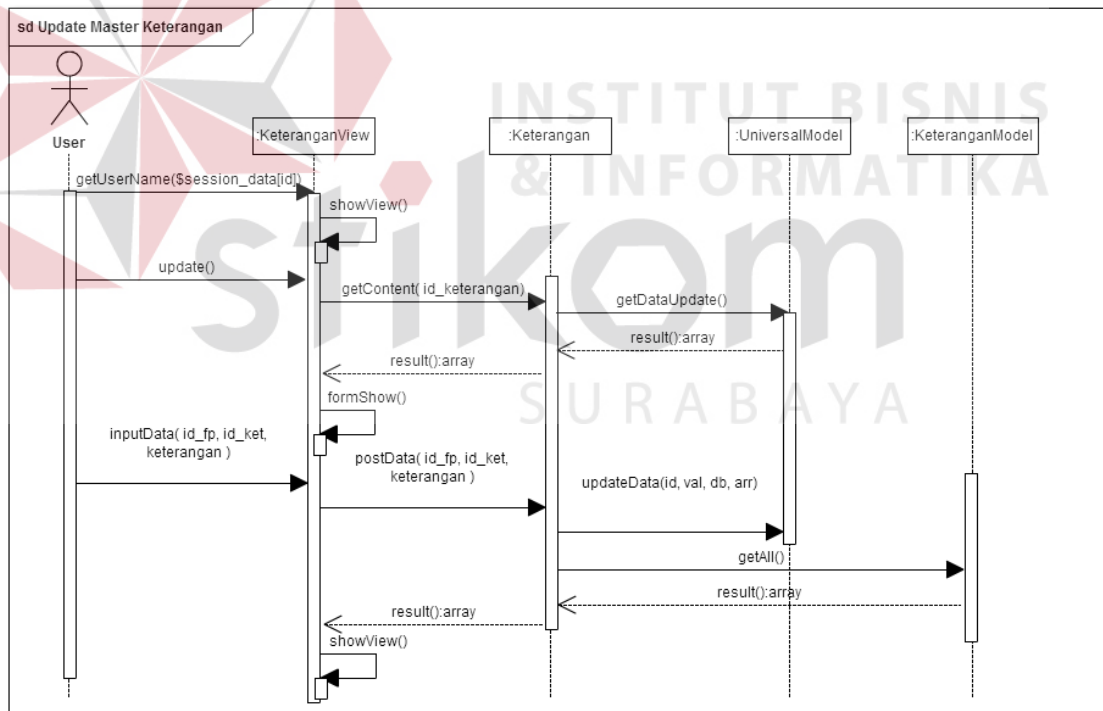
Variabel yang menjadi inputan pada proses insert dan Update pada Master Pendidikan adalah `id_fingerprint`, `id_pendidikan`, `tgl_masuk`, `tgl_keluar`, `status`, `nama_sekolah`, `keterangan`. Variabel `id_pendidikan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

e. Master Keterangan

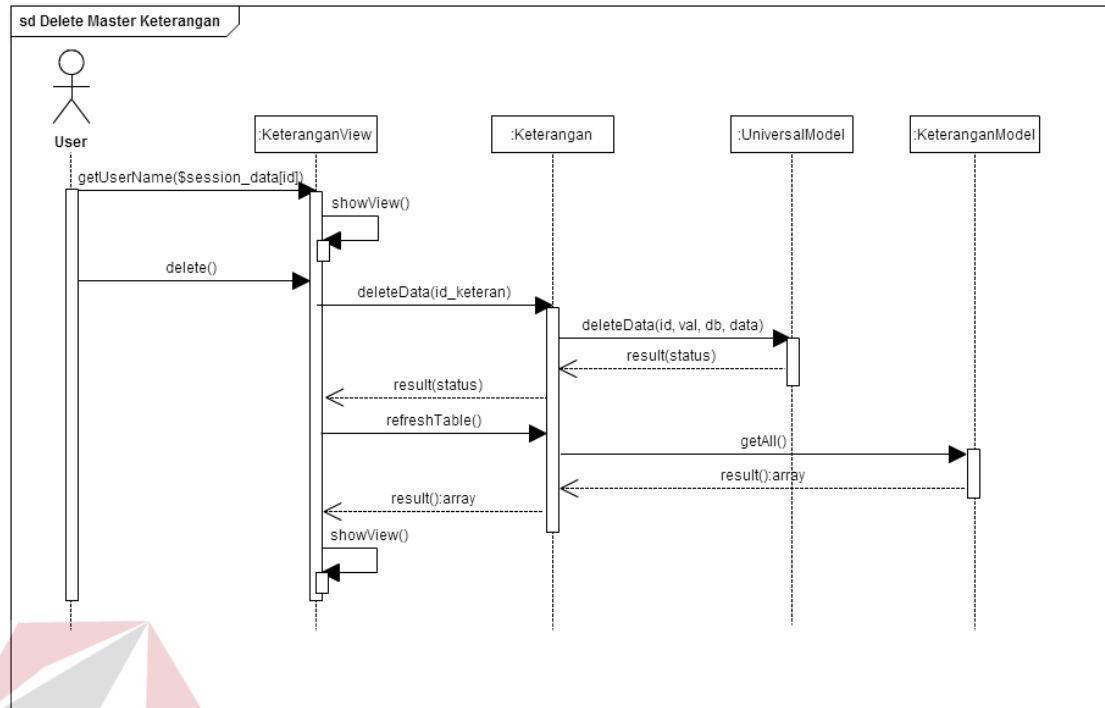
*Sequence diagram* untuk proses insert, update dan delete pada *master* keterangan terdapat pada gambar 3.21, gambar 3.22, dan gambar 3.23.



Gambar 3.21 Sequence Diagram Proses Insert Data pada Master Keterangan



Gambar 3.22 Sequence Diagram Proses Update Data pada Master Keterangan

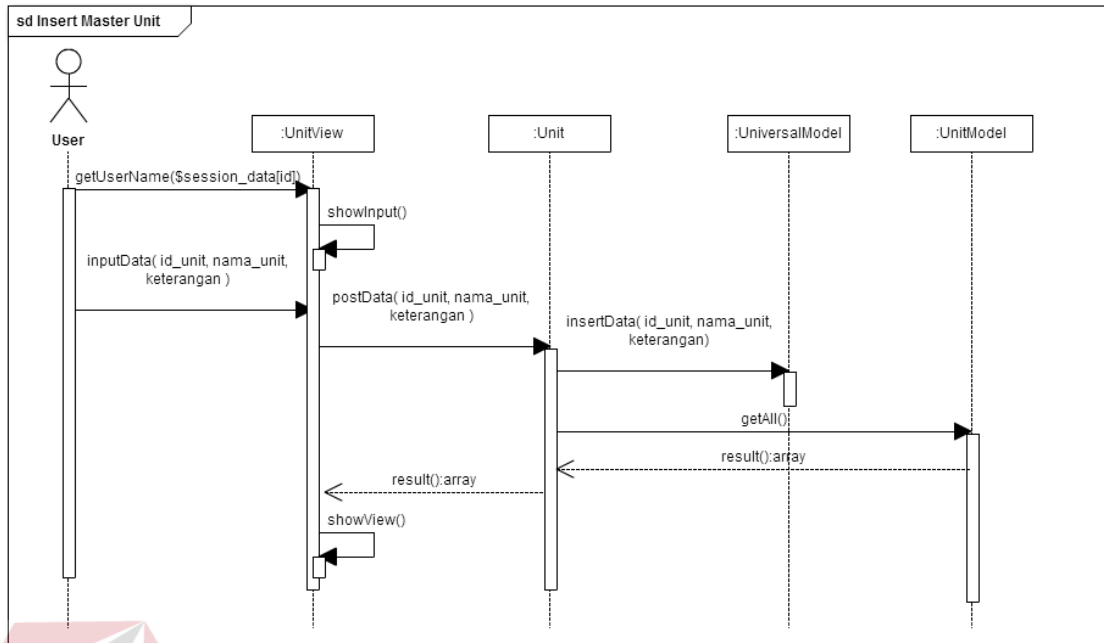


Gambar 3.23 Sequence Diagram Proses Delete Data pada Master Keterangan

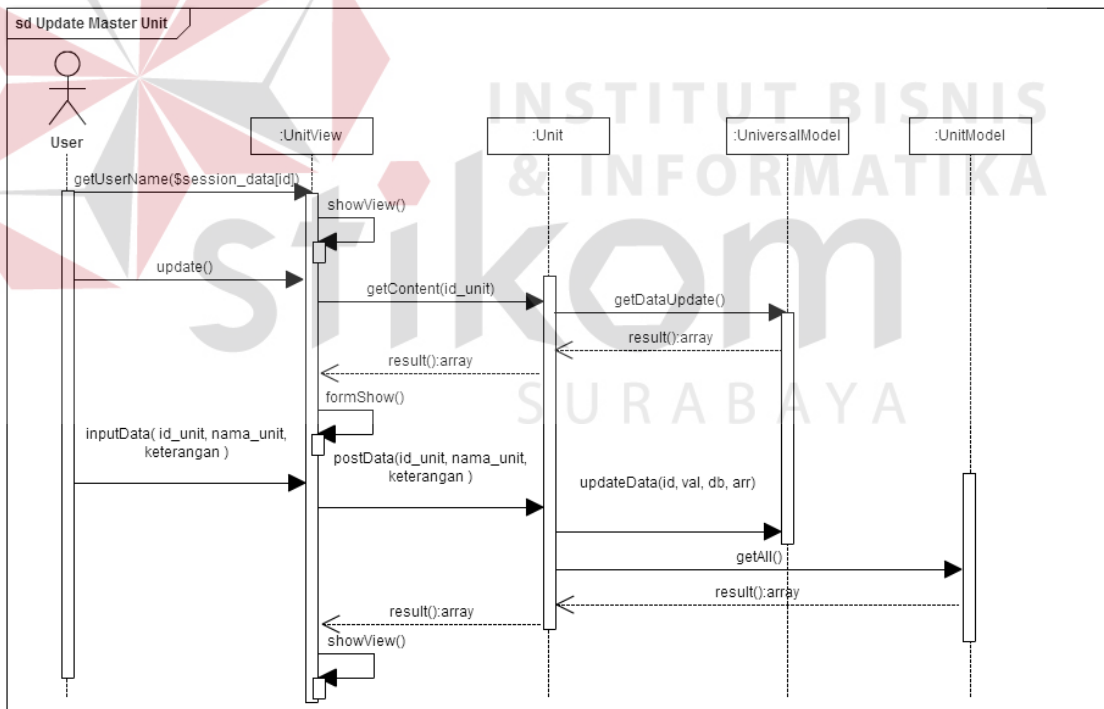
Variabel yang menjadi inputan pada proses insert dan Update pada Master Keterangan adalah `id_fingerprint`, `id_keterangan`, `keterangan`. Variabel `id_keterangan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

#### f. Master Unit

*Sequence diagram* untuk proses insert, update dan delete pada *master unit* terdapat pada gambar 3.24, gambar 3.25, dan gambar 3.26.

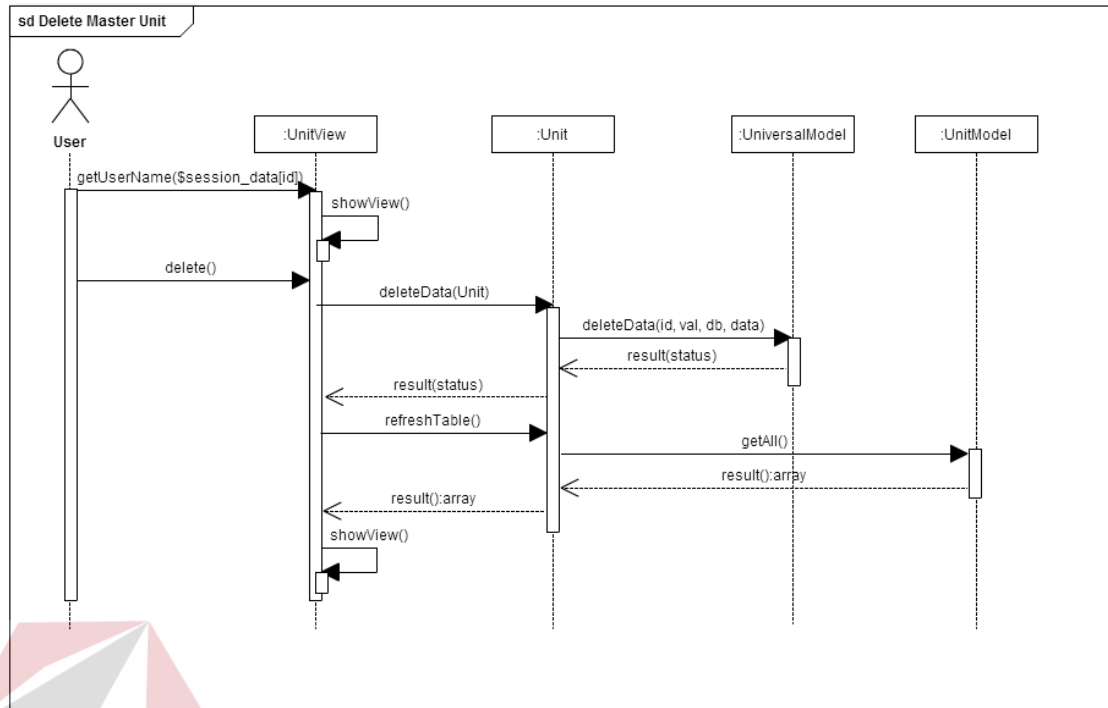


Gambar 3.24 Sequence Diagram Proses Insert Data pada Master Unit



Gambar 3.25 Sequence Diagram Proses Update Data pada Master Unit



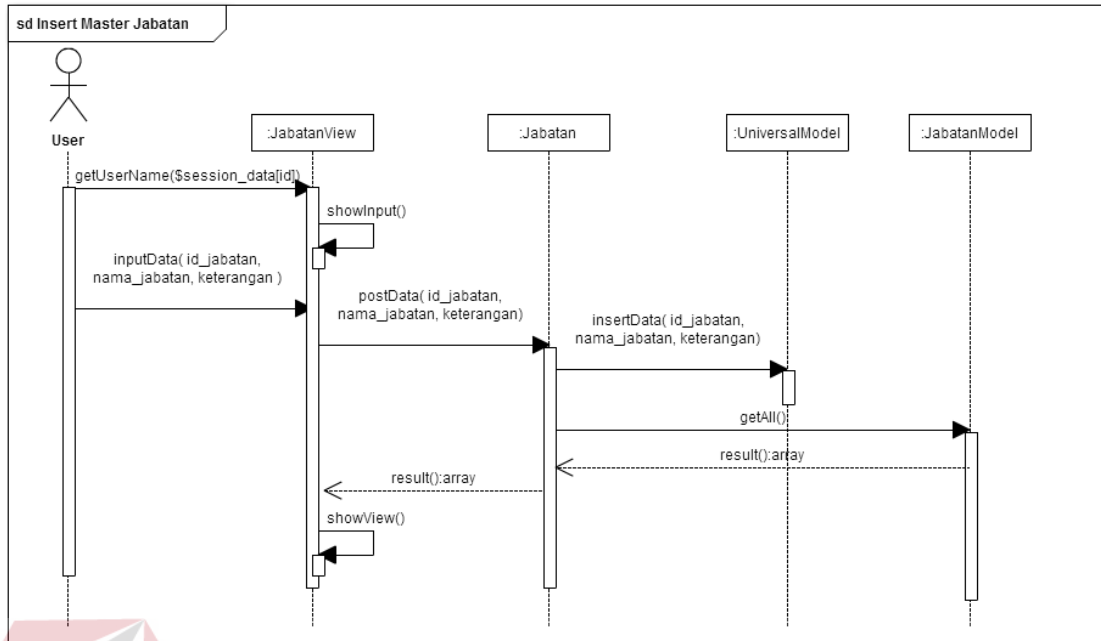


Gambar 3.26 Sequence Diagram Proses Delete Data pada Master Unit

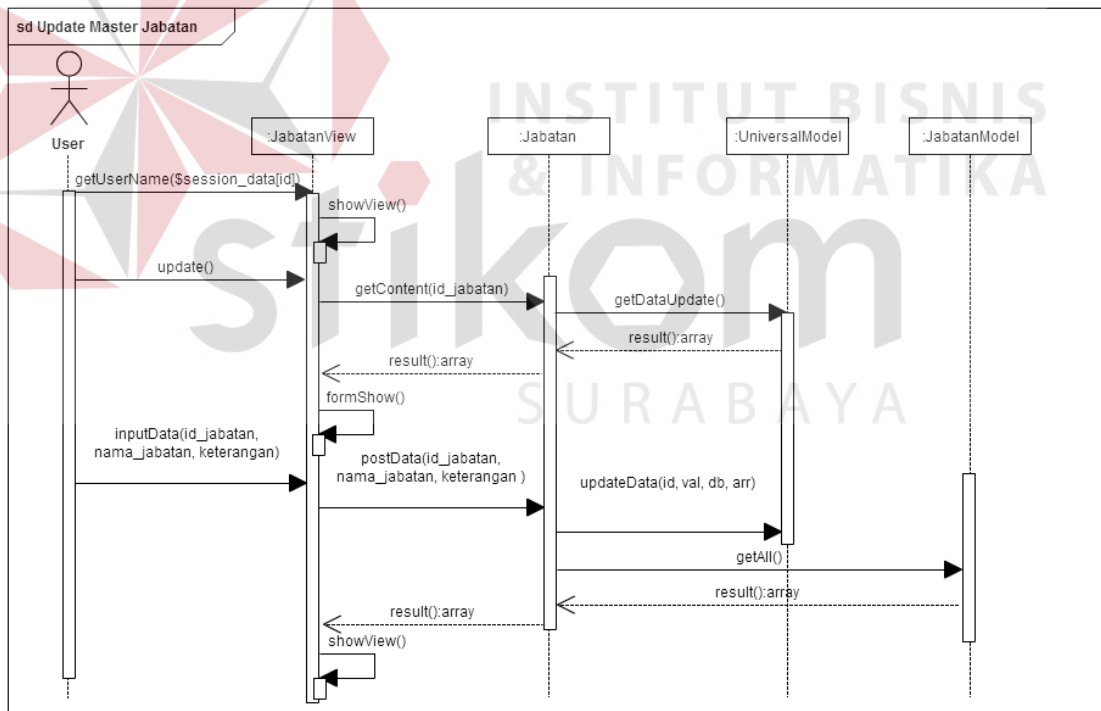
Variabel yang menjadi inputan pada proses insert dan Update pada Master Unit adalah id\_unit, nama\_unit, keterangan. Variabel id\_unit disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

g. Master Jabatan

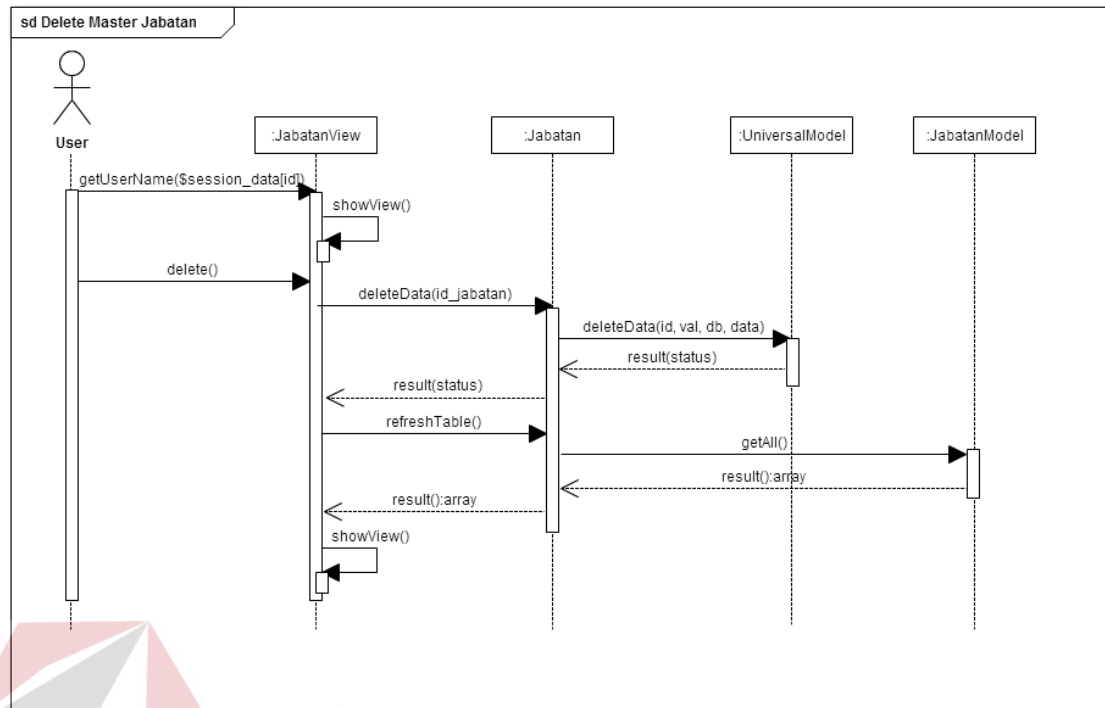
*Sequence diagram* untuk proses insert, update dan delete pada *master jabatan* terdapat pada gambar 3.27, gambar 3.28, dan gambar 3.29.



Gambar 3.27 Sequence Diagram Proses Insert Data pada Master Jabatan



Gambar 3.28 Sequence Diagram Proses Update Data pada Master Jabatan

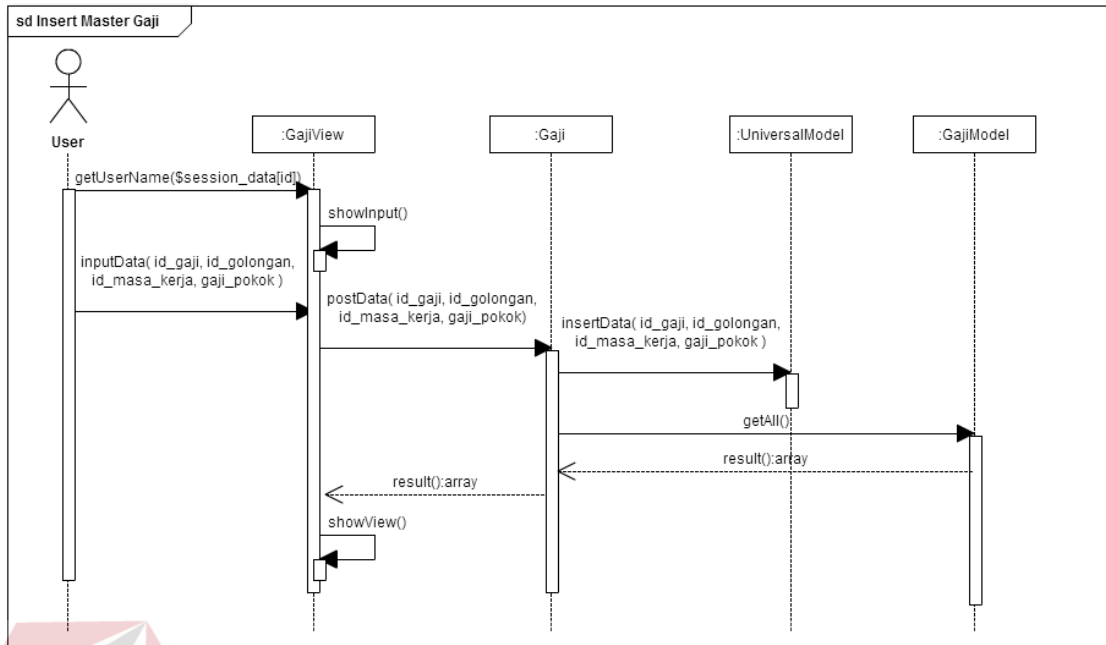


Gambar 3.29 Sequence Diagram Proses Delete Data pada Master Jabatan

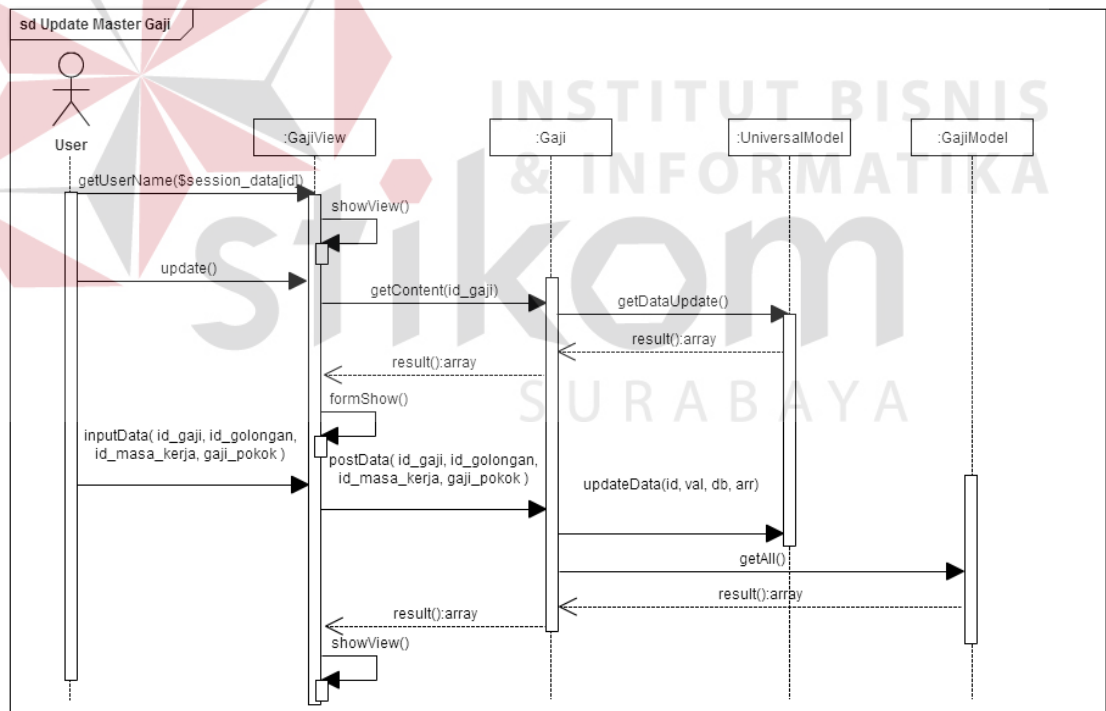
Variabel yang menjadi inputan pada proses insert dan Update pada Master Jabatan adalah `id_jabatan`, `nama_jabatan`, `keterangan`. Variabel `id_jabatan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

#### h. Master Gaji

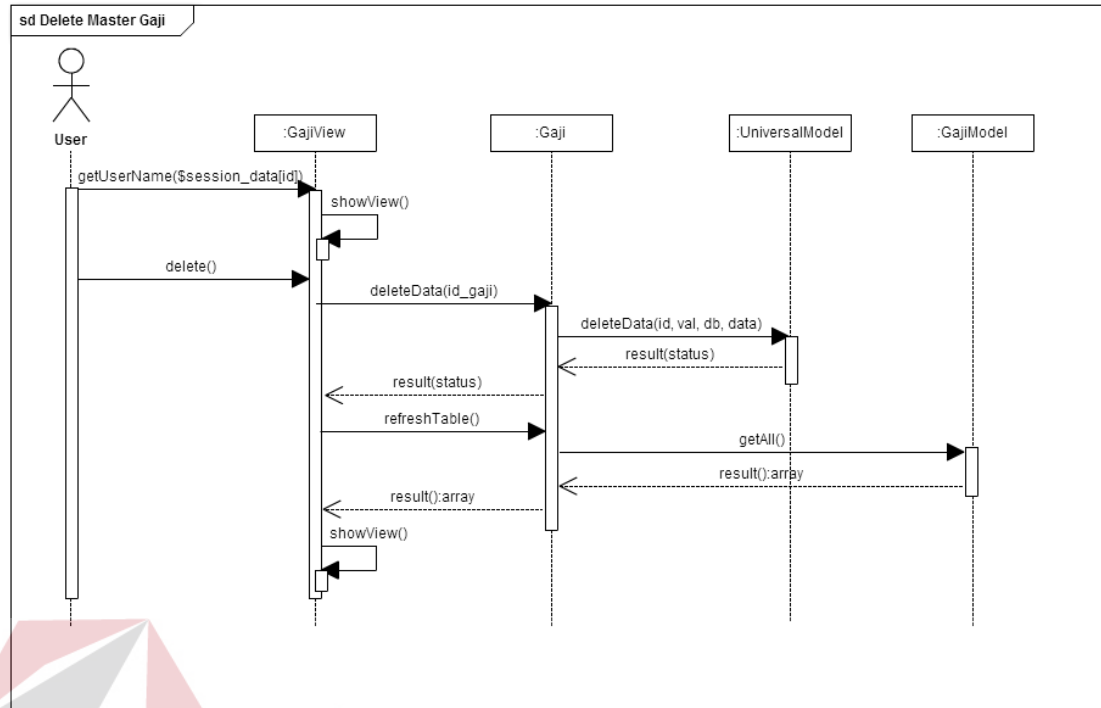
*Sequence diagram* untuk proses insert, update dan delete pada *master gaji* terdapat pada gambar 3.30, gambar 3.31, dan gambar 3.32.



Gambar 3.30 Sequence Diagram Proses Insert Data pada Master Gaji



Gambar 3.31 Sequence Diagram Proses Update Data pada Master Gaji

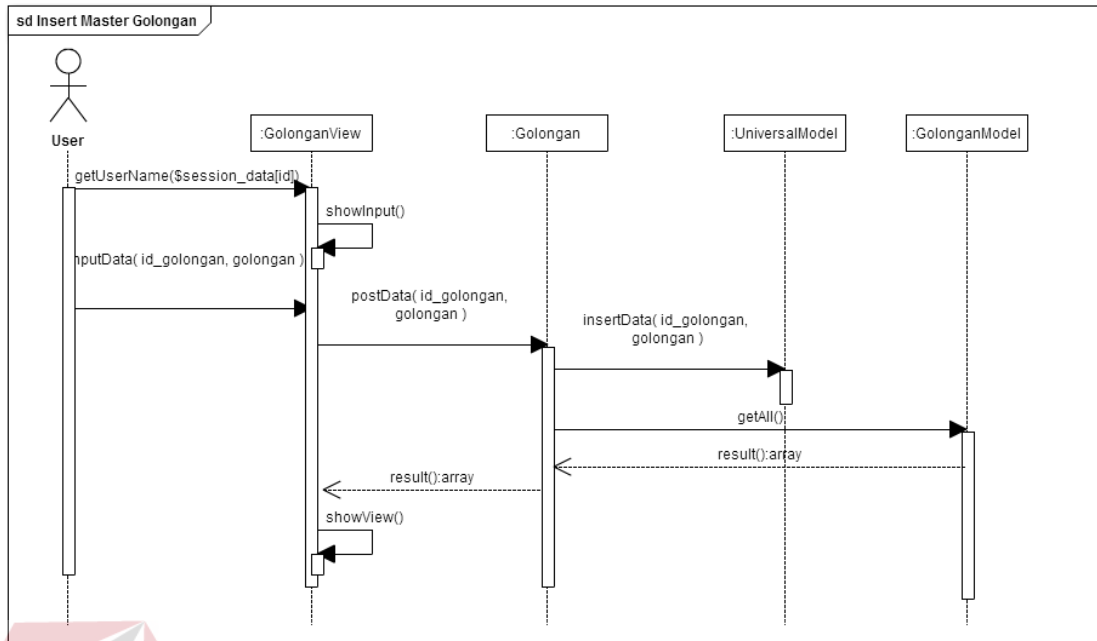


Gambar 3.32 Sequence Diagram Proses Delete Data pada Master Gaji

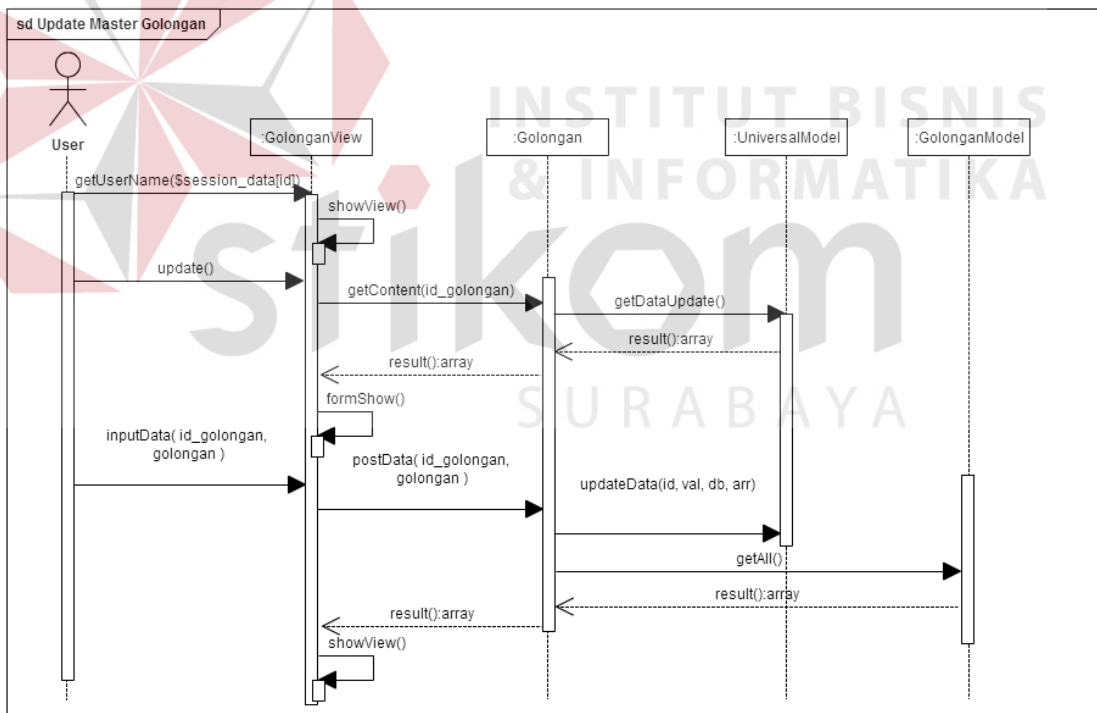
Variabel yang menjadi inputan pada proses insert dan Update pada Master Gaji adalah `id_gaji`, `id_golongan`, `id_masa_kerja`, `gaji_pokok`. Variabel `id_gaji` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

i. Master Golongan

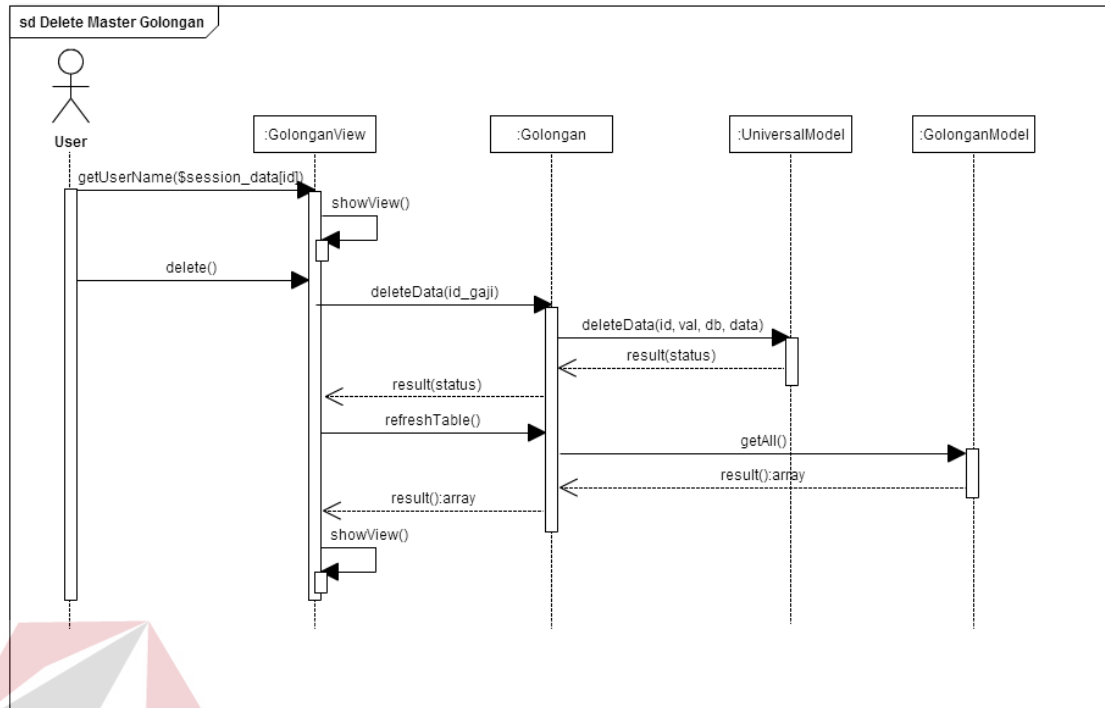
*Sequence diagram* untuk proses insert, update dan delete pada *master golongan* terdapat pada gambar 3.33, gambar 3.34, dan gambar 3.35.



Gambar 3.33 Sequence Diagram Proses Insert Data pada Master Golongan



Gambar 3.34 Sequence Diagram Proses Update Data pada Master Golongan

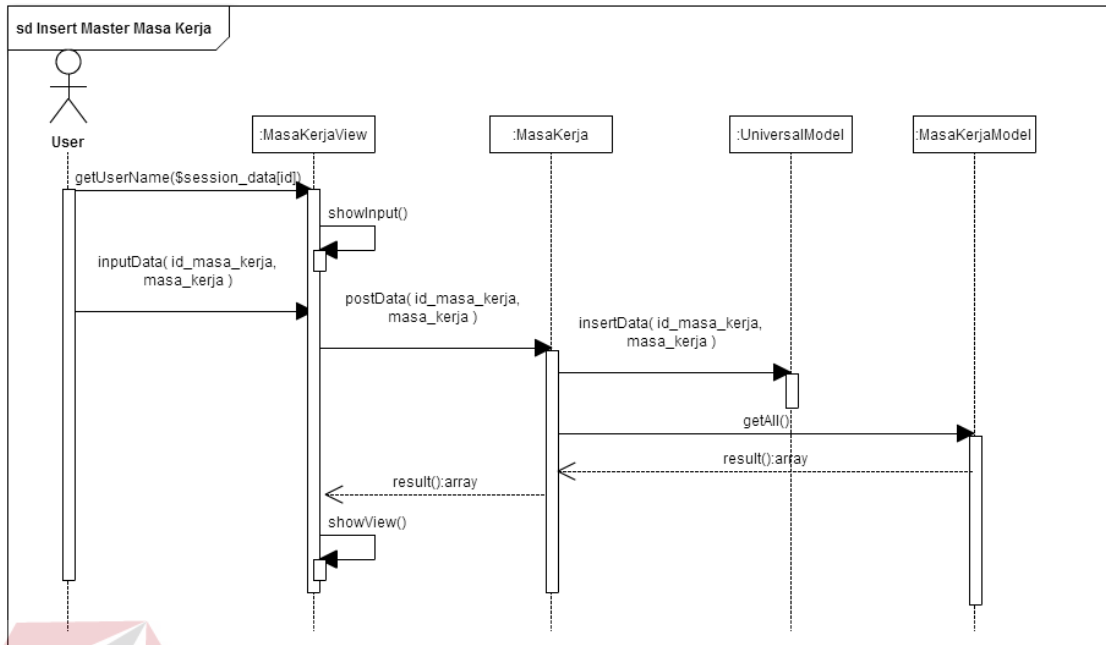


Gambar 3.35 Sequence Diagram Proses Delete Data pada Master Golongan

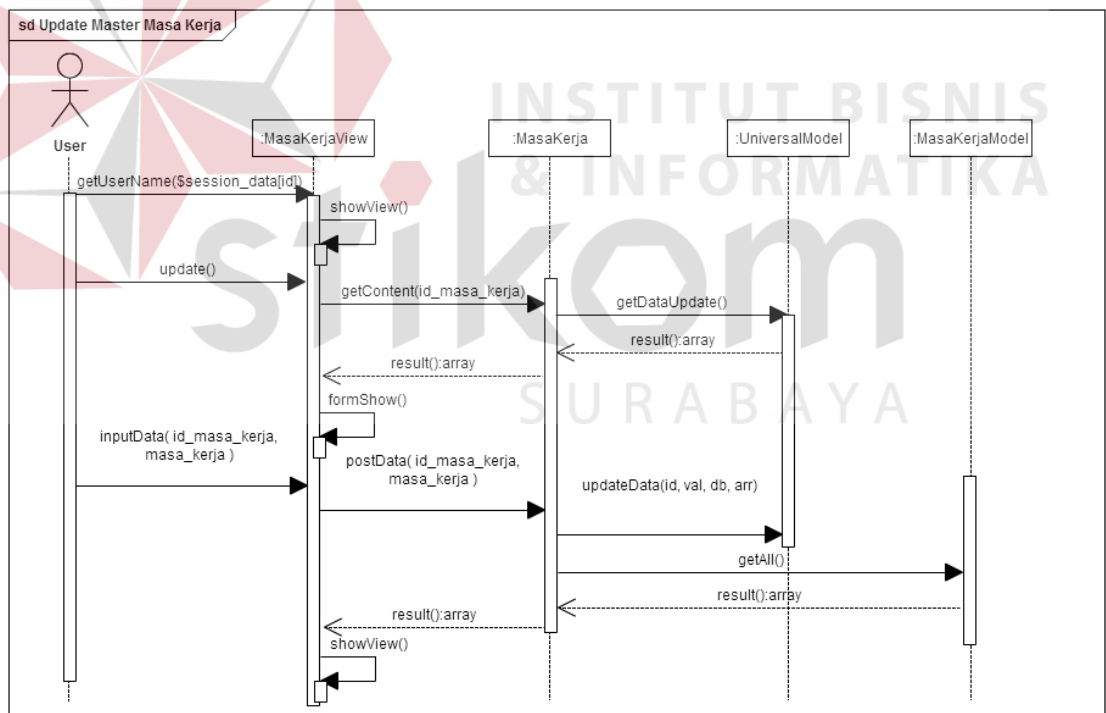
Variabel yang menjadi inputan pada proses insert dan Update pada Master Golongan adalah `id_golongan`, `golongan`. Variabel `id_golongan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

j. Master Masa Kerja

*Sequence diagram* untuk proses insert, update dan delete pada *master* kerja terdapat pada gambar 3.36, gambar 3.37, dan gambar 3.38.

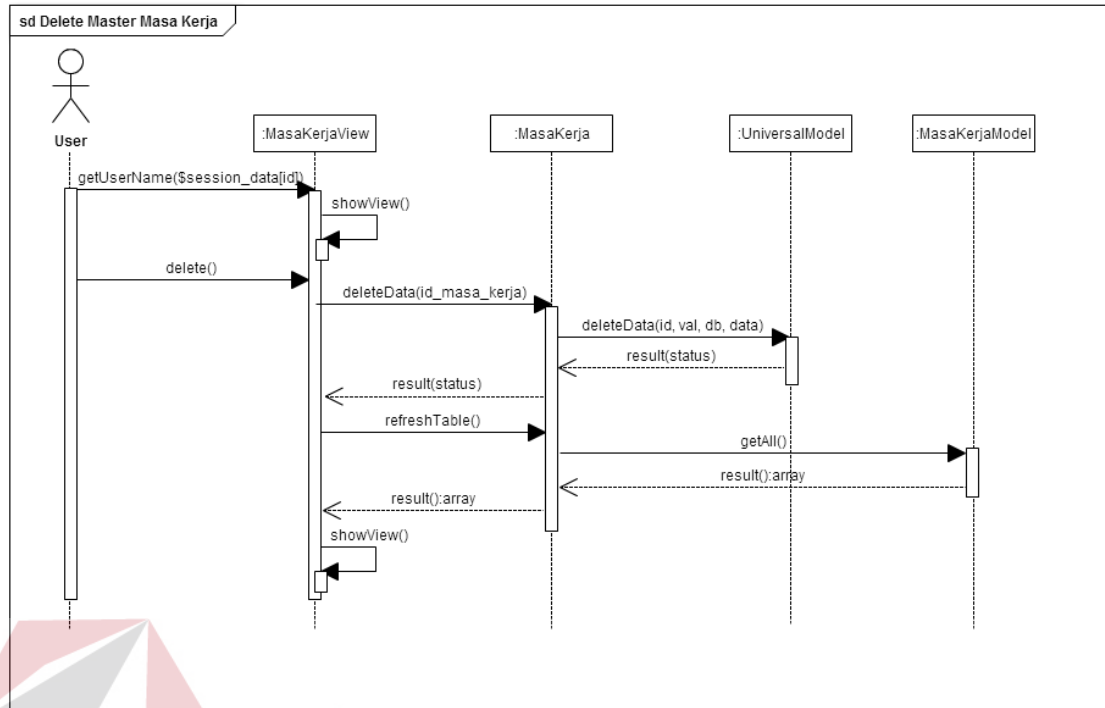


Gambar 3.36 Sequence Diagram Proses Insert Data pada Master Masa Kerja



Gambar 3.37 Sequence Diagram Proses Update Data pada Master Masa Kerja



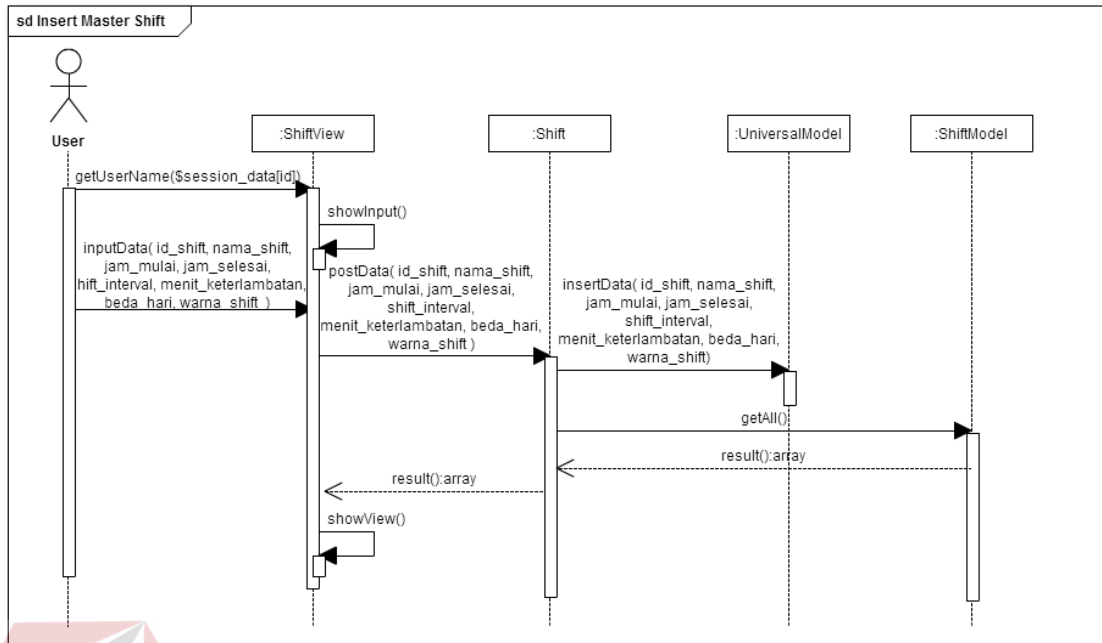


Gambar 3.38 Sequence Diagram Proses Delete Data pada Master Masa Kerja

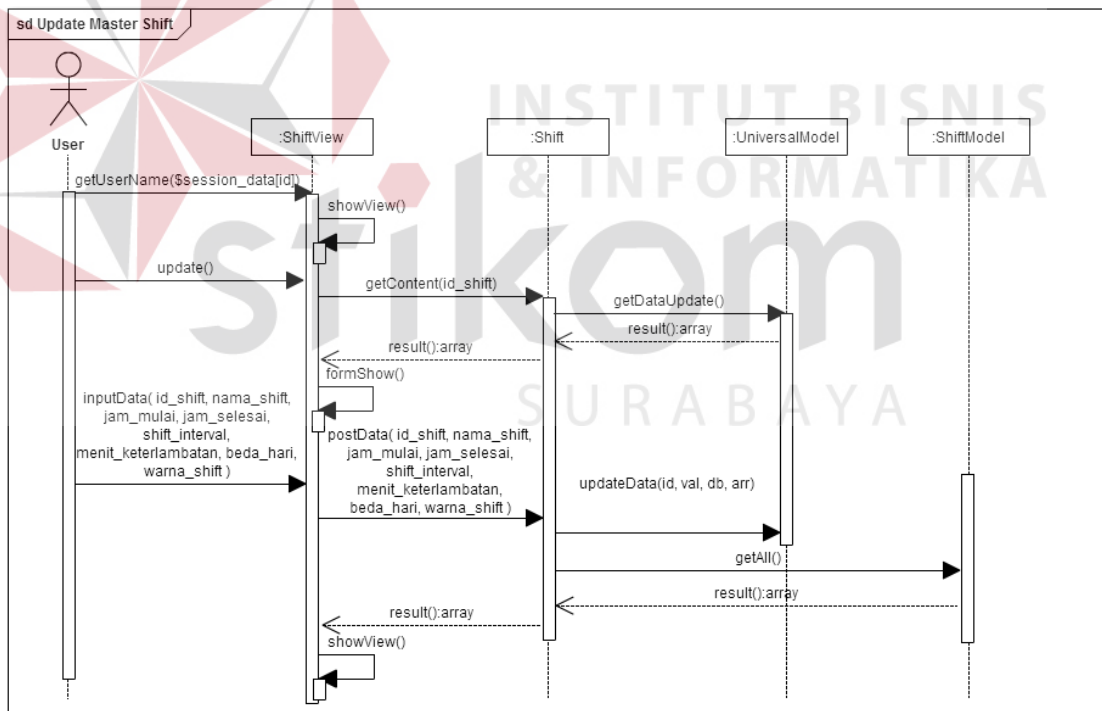
Variabel yang menjadi inputan pada proses insert dan Update pada Master Masa Kerja adalah `id_masa_kerja`, `masa_kerja`. Variabel `id_masa_kerja` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

#### k. Master Shift

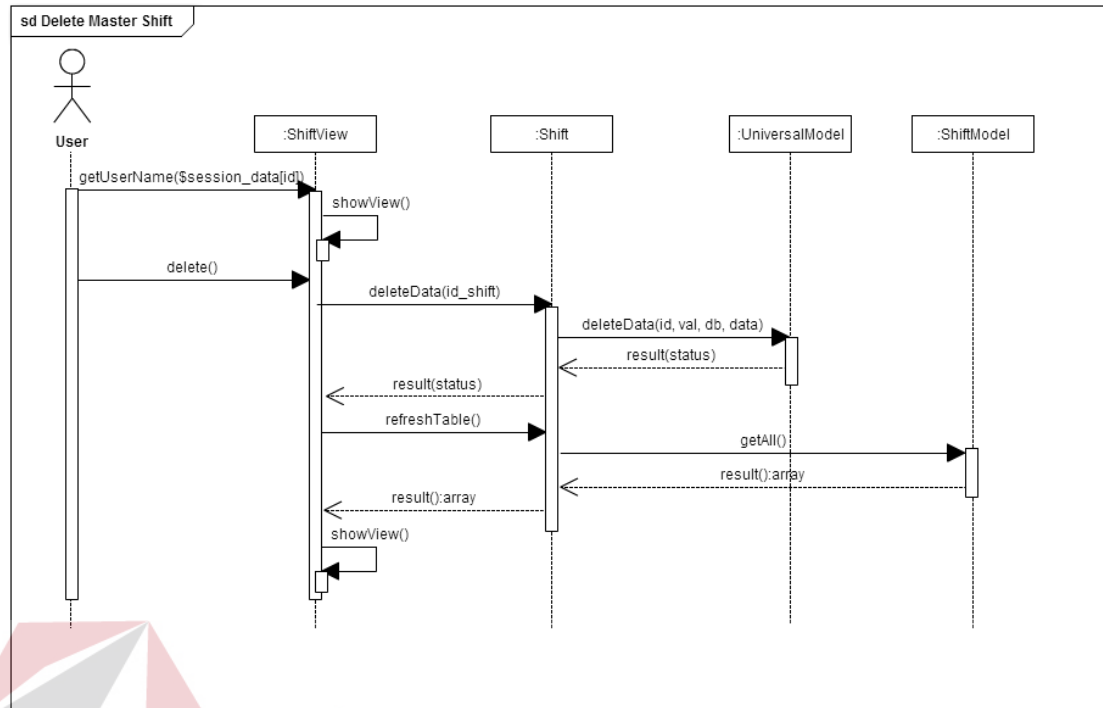
*Sequence diagram* untuk proses insert, update dan delete pada *master shift* terdapat pada gambar 3.39, gambar 3.40, dan gambar 3.41.



Gambar 3.39 Sequence Diagram Proses Insert Data pada Master Shift



Gambar 3.40 Sequence Diagram Proses Update Data pada Master Shift

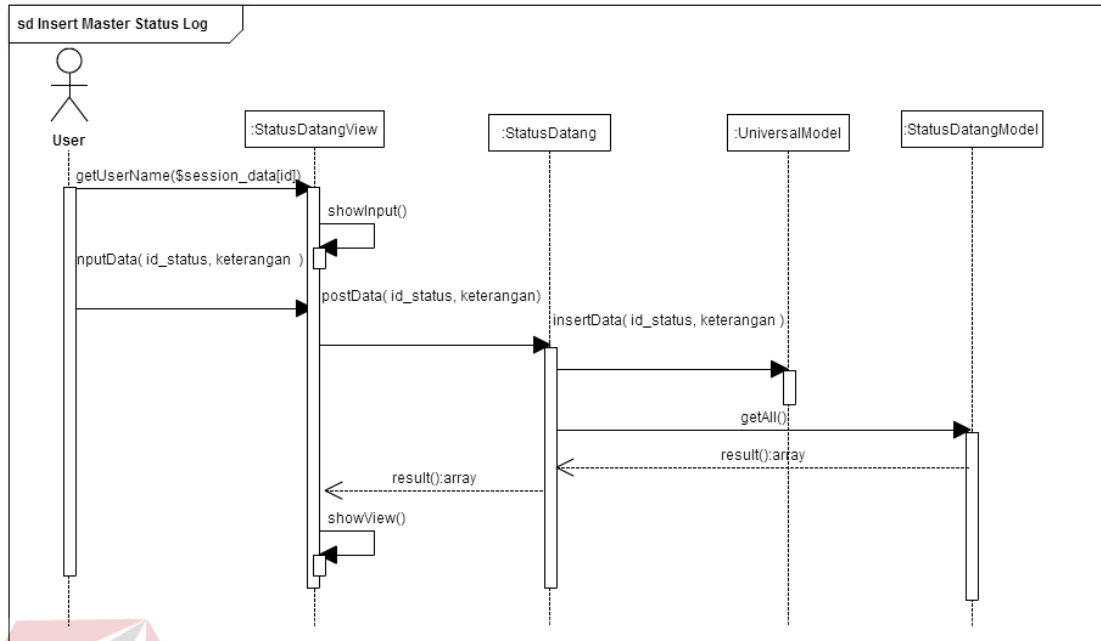


Gambar 3.41 Sequence Diagram Proses Delete Data pada Master Shift

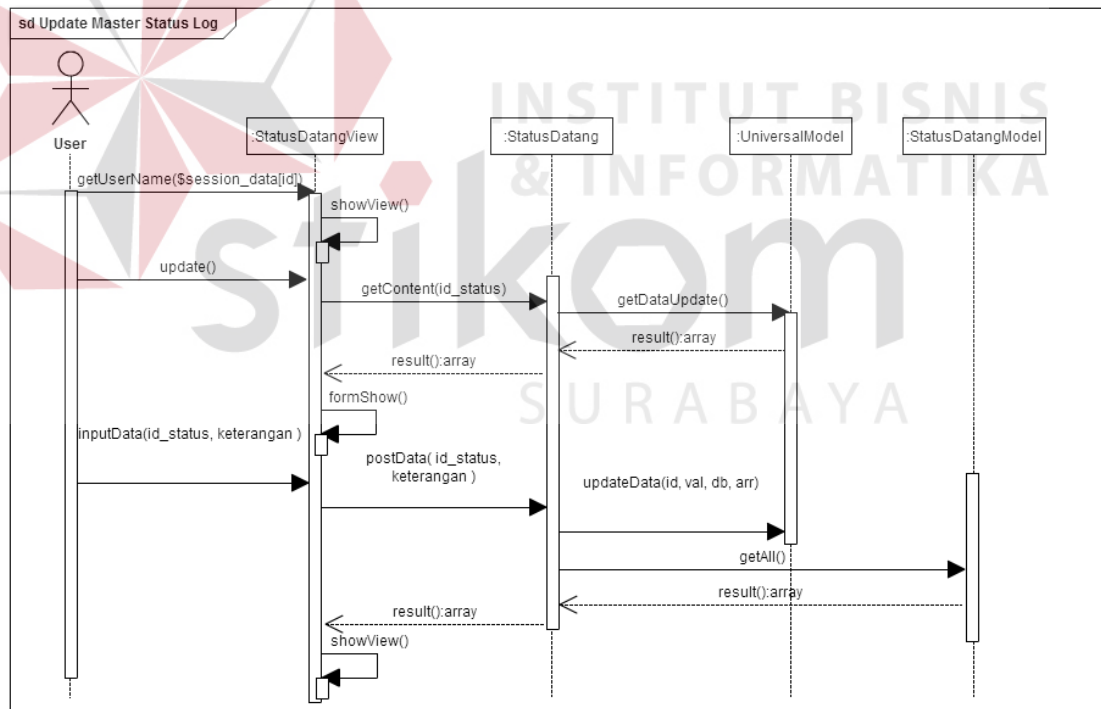
Variabel yang menjadi inputan pada proses insert dan Update pada Master Shift adalah `id_shift`, `nama_shift`, `jam_mulai`, `jam_selesai`, `shift_interval`, `menit_keterlambatan`, `beda_hari`, `warna_shift`. Variabel `id_keterangan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

#### 1. Master Status Log

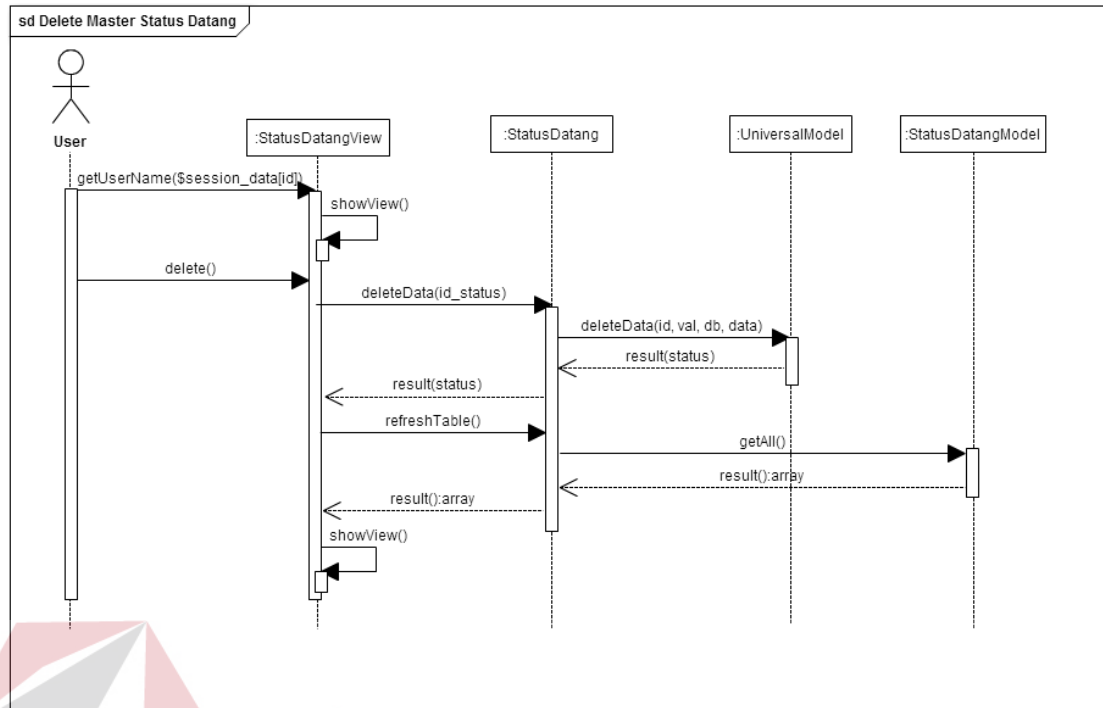
*Sequence diagram* untuk proses insert, update dan delete pada *master status log* terdapat pada gambar 3.42, gambar 3.43, dan gambar 3.44.



Gambar 3.42 Sequence Diagram Proses Insert Data pada Master Status Log



Gambar 3.43 Sequence Diagram Proses Update Data pada Master Status Log

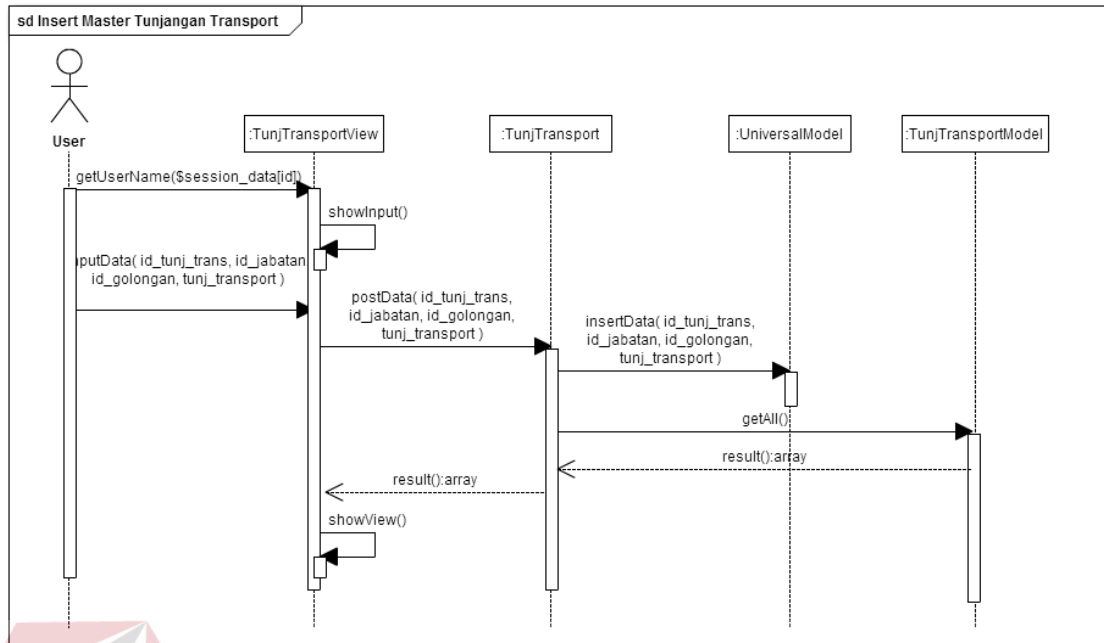


Gambar 3.44 Sequence Diagram Proses Delete Data pada Master Status Log

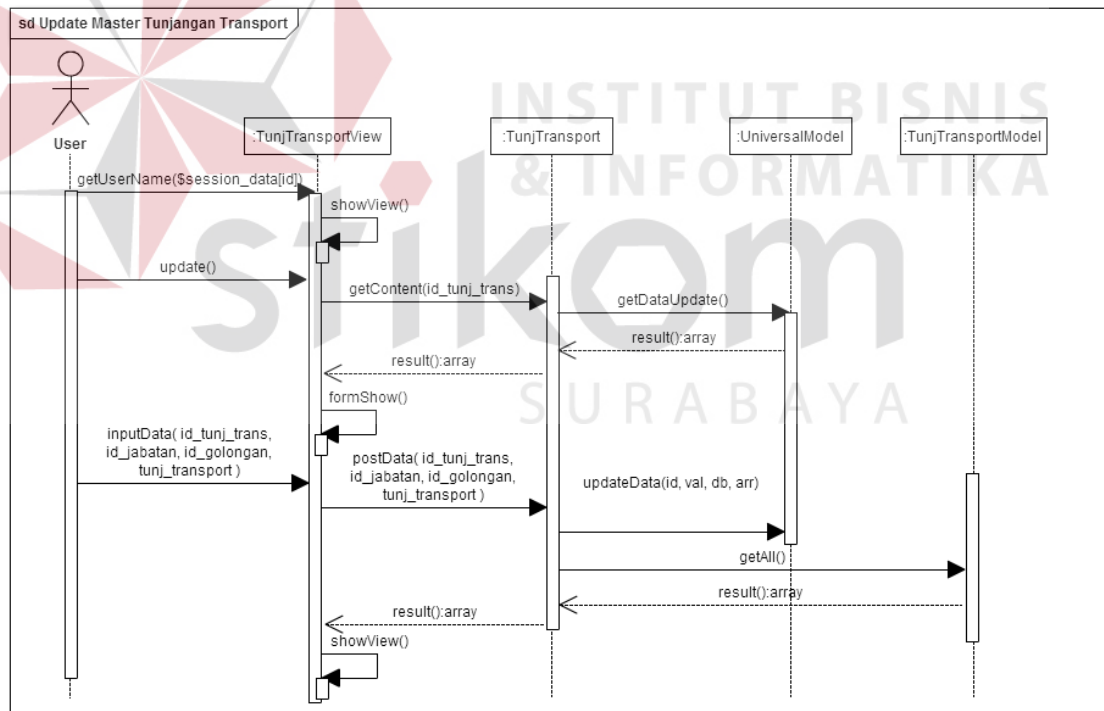
Variabel yang menjadi inputan pada proses insert dan Update pada Master Status Log adalah `id_status`, keterangan. Variabel `id_status` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

m. Master Tunjangan Transportasi

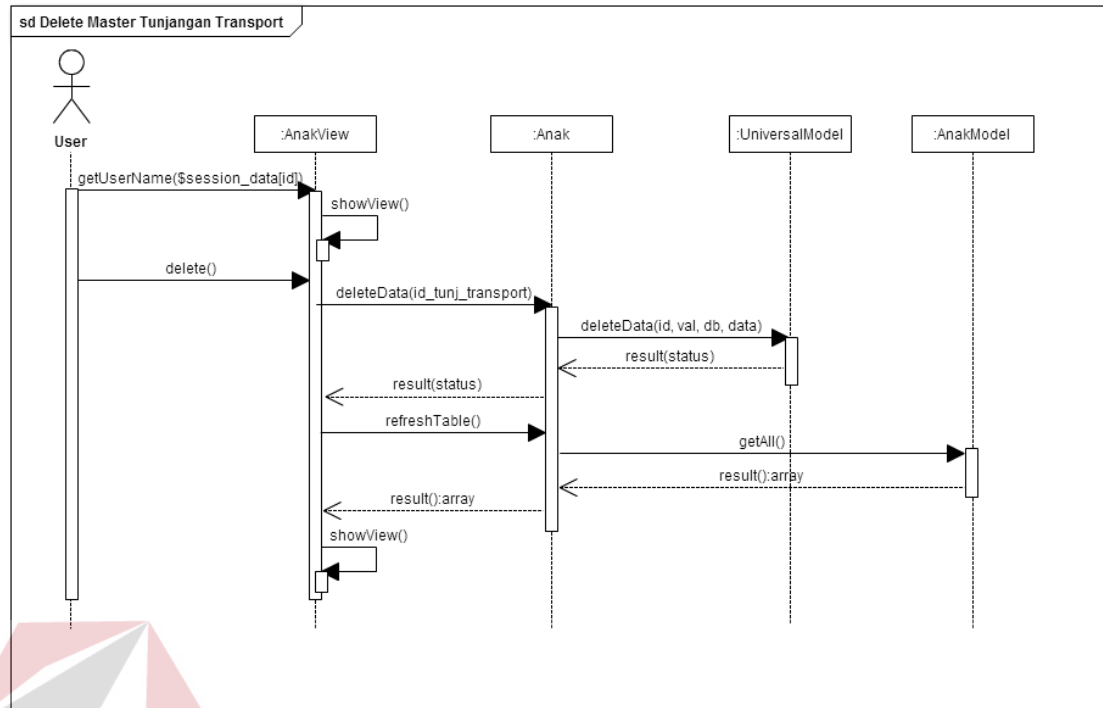
*Sequence diagram* untuk proses insert, update dan delete pada *master* transportasi terdapat pada gambar 3.45, gambar 3.46, dan gambar 3.47.



Gambar 3.45 Sequence Diagram Proses Insert Data pada Master Transportasi



Gambar 3.46 Sequence Diagram Proses Update Data pada Master Transportasi

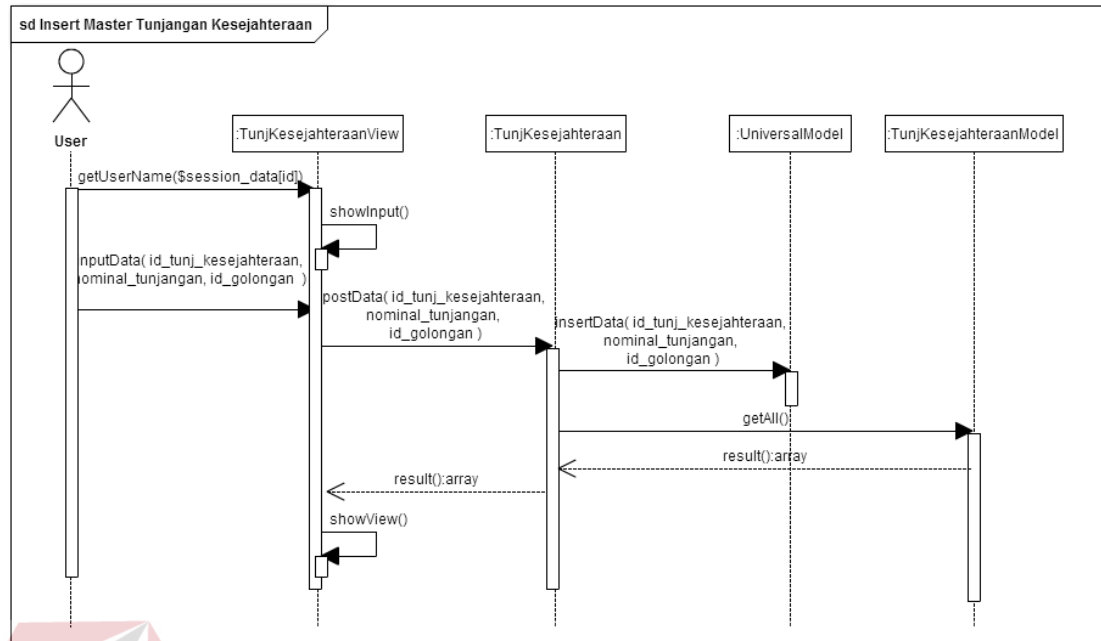


Gambar 3.47 Sequence Diagram Proses Delete Data pada Master Transportasi

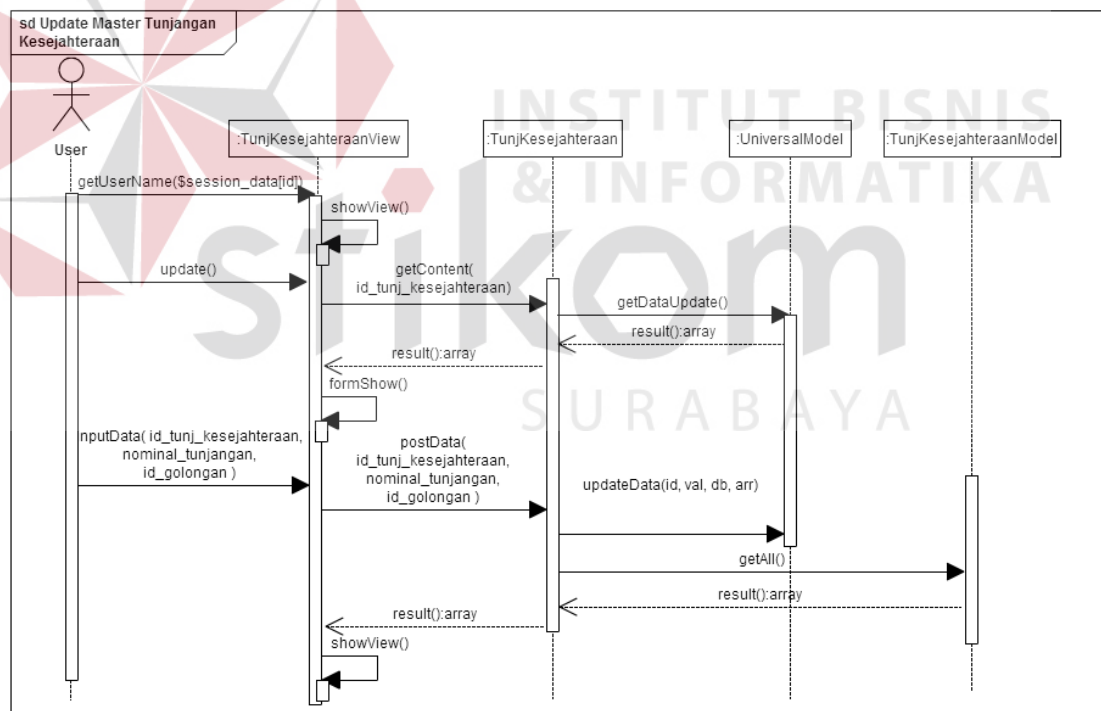
Variabel yang menjadi inputan pada proses insert dan Update pada Master Transportasi adalah `id_tunj_trans`, `id_jabatan`, `id_golongan`, `tunj_transport`. Variabel `id_keterangan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

#### n. Master Tunjangan Kesejahteraan

*Sequence diagram* untuk proses insert, update dan delete pada *master* kesejahteraan terdapat pada gambar 3.48, gambar 3.49, dan gambar 3.50.

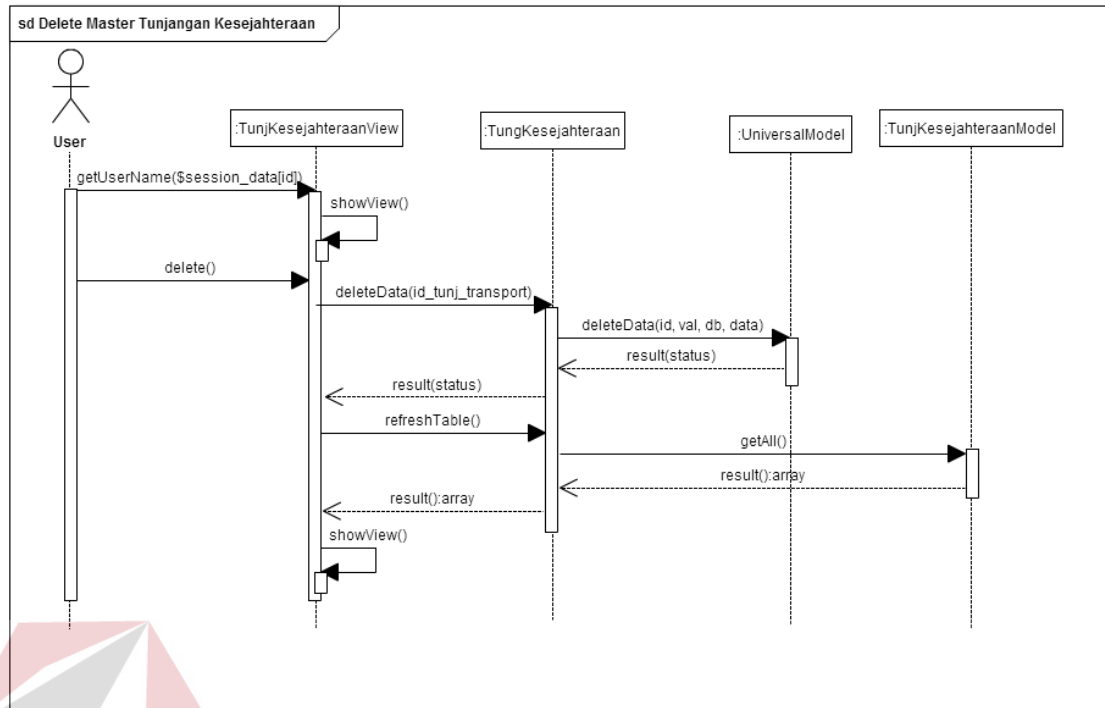


Gambar 3.48 Sequence Diagram Insert Data Master Tunjangan Kesejahteraan



Gambar 3.49 Sequence Diagram Update Data Master Tunjangan Kesejahteraan



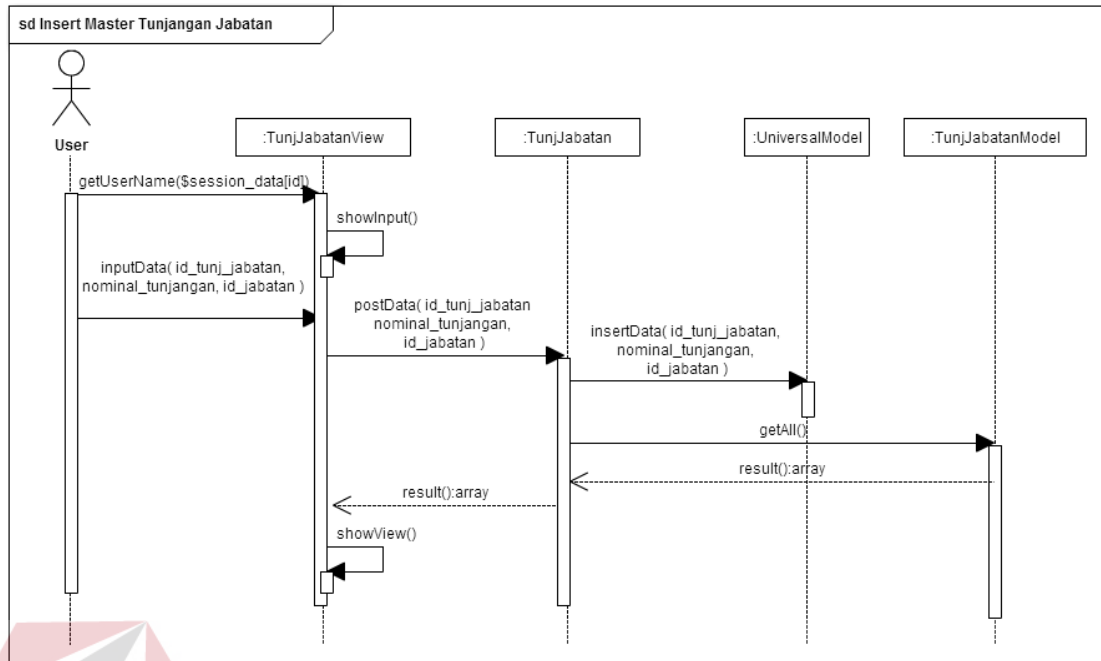


Gambar 3.50 Sequence Diagram Delete Data Master Tunjangan Kesejahteraan

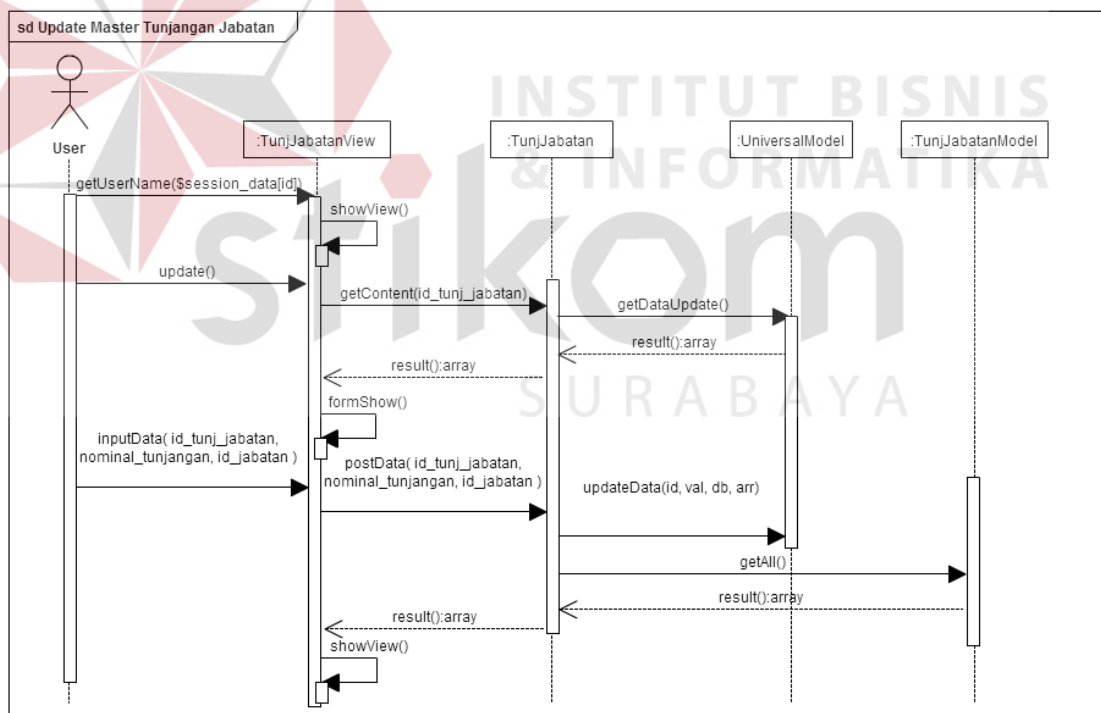
Variabel yang menjadi inputan pada proses insert dan Update pada Master Tunjangan Kesejahteraan adalah `id_tunj_kesejahteraan`, `nominal_tunjangan`, `id_golongan`. Variabel `id_tunj_kesejahteraan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

o. Master Tunjangan Jabatan

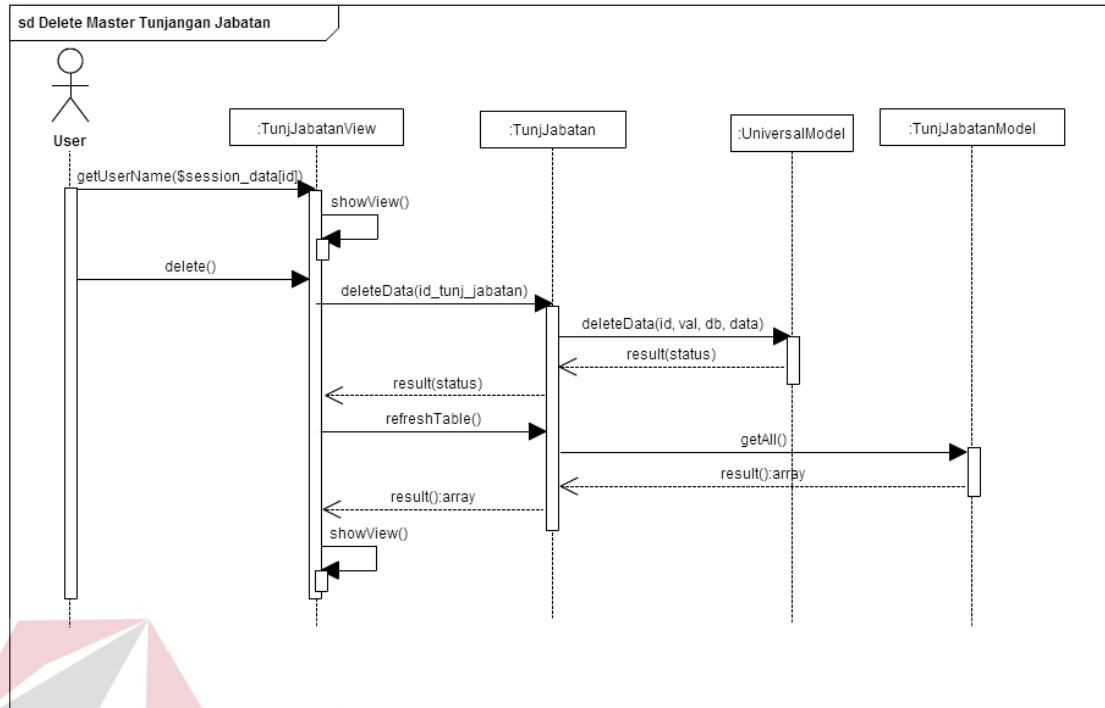
*Sequence diagram* untuk proses insert, update dan delete pada *master* kesejahteraan terdapat pada gambar 3.51, gambar 3.52, dan gambar 3.53.



Gambar 3.51 Sequence Diagram Insert Data pada Master Tunjangan Jabatan



Gambar 3.52 Sequence Diagram Insert Data pada Master Tunjangan Jabatan

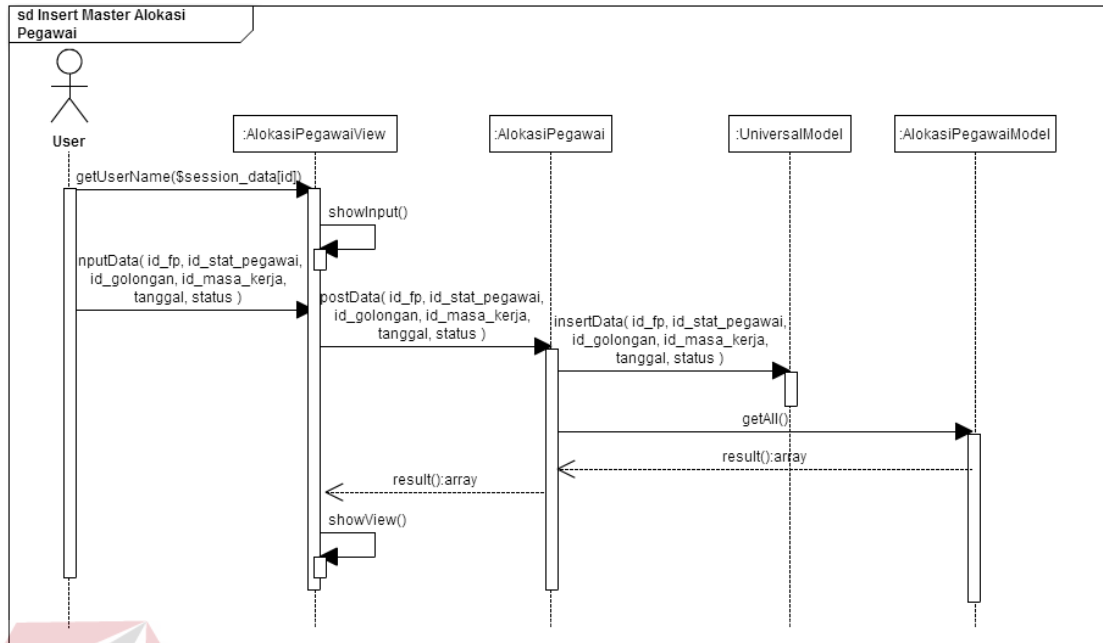


Gambar 3.53 Sequence Diagram Delete Data pada Master Tunjangan Jabatan

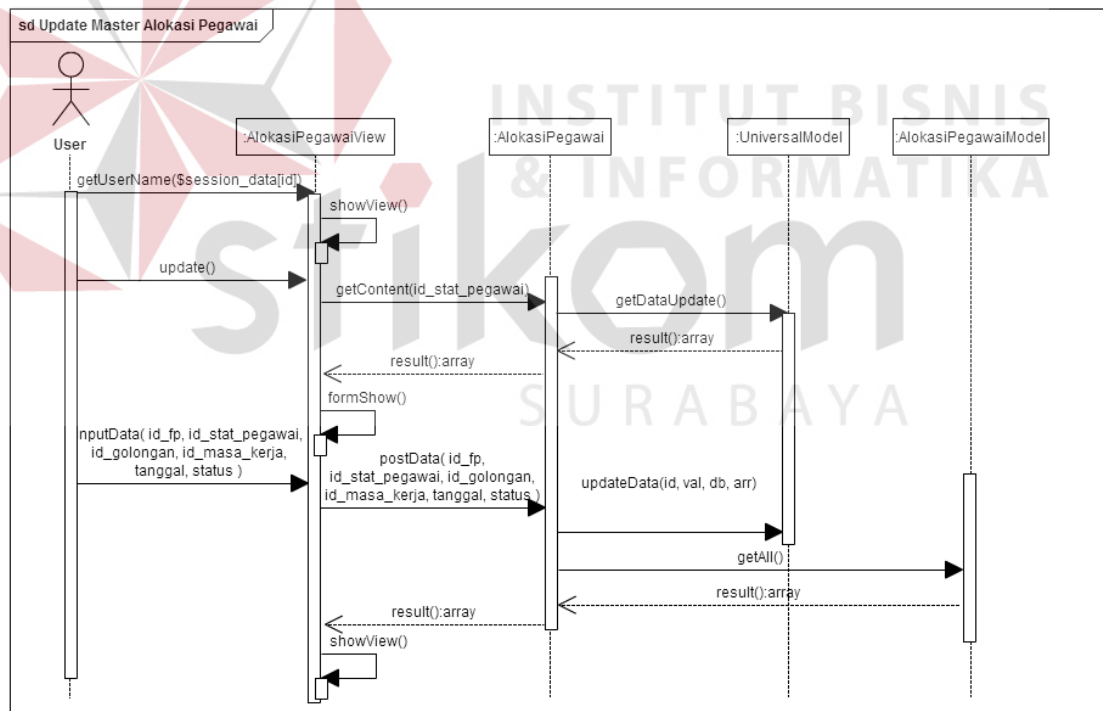
Variabel yang menjadi inputan pada proses insert dan Update pada Master Tunjangan Jabatan adalah `id_tunj_jabatan`, `nominal_tunjangan`, `id_jabatan`. Variabel `id_tunj_jabatan` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

p. Master Alokasi Pegawai

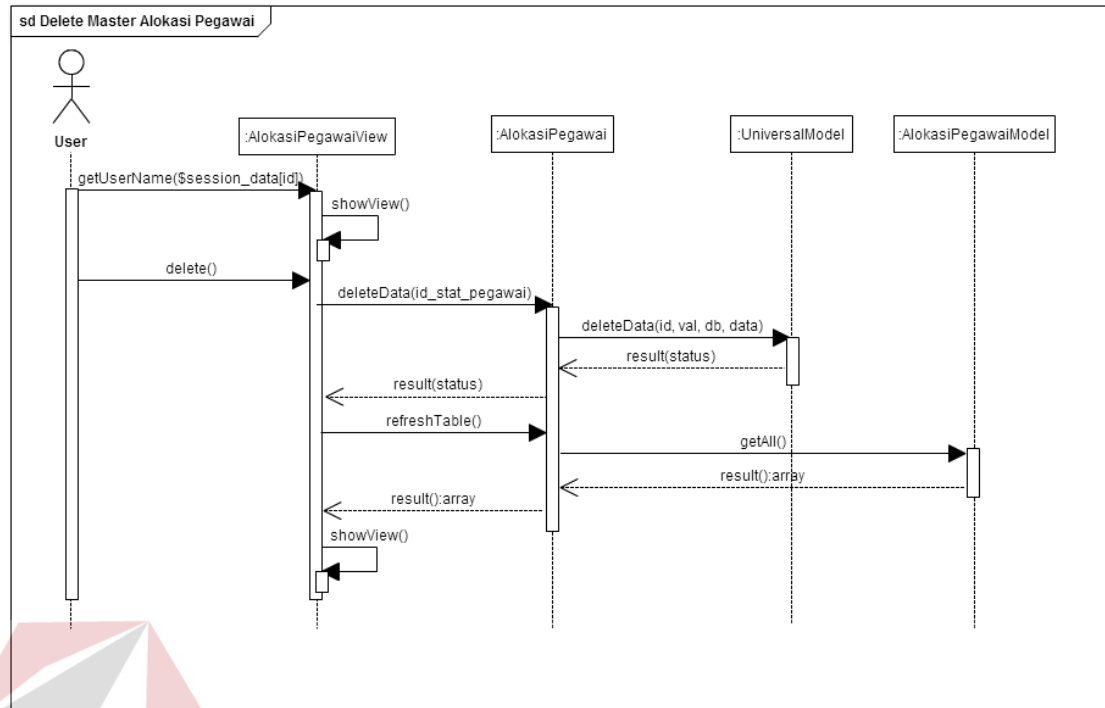
*Sequence diagram* untuk proses insert, update dan delete pada *master* alokasi pegawai terdapat pada gambar 3.54, gambar 3.55, dan gambar 3.56.



Gambar 3.54 Sequence Diagram Proses Insert Data pada Master Alokasi Pegawai



Gambar 3.55 Sequence Diagram Proses Update Data pada Master Alokasi Pegawai



Gambar 3. 56 Sequence Diagram Proses Delete Data pada Master Alokasi Pegawai

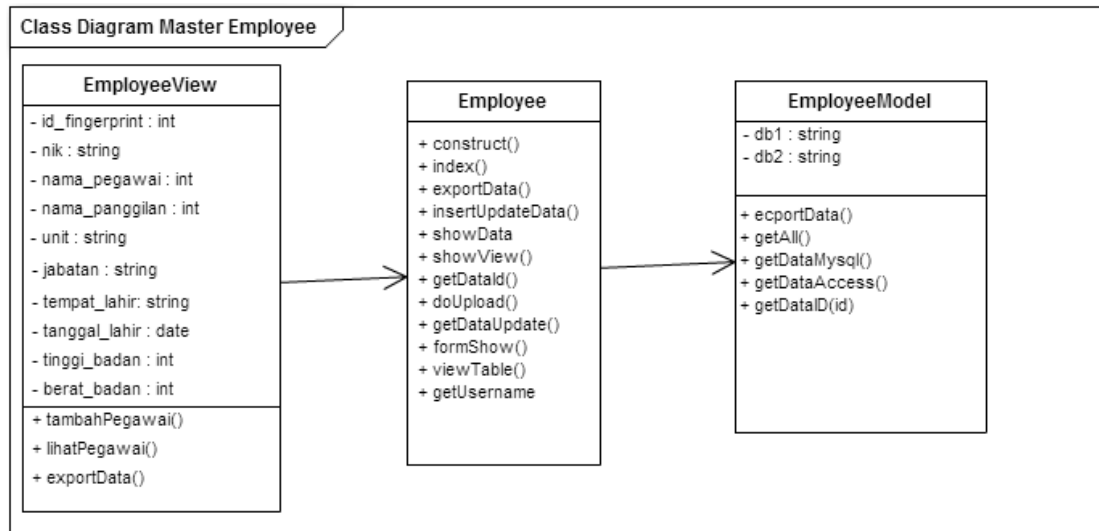
Variabel yang menjadi inputan pada proses insert dan Update pada Master Alokasi Pegawai adalah `id_fingerprint`, `id_stat_pegawai`, `id_golongan`, `id_masa_kerja`, `tanggal`, `status`. Variabel `id_stat_pegawai` disini digunakan sebagai acuan dalam melakukan proses *update* dan *delete*.

### C. Class Diagram Modul Master

Penggambaran dari relasi *class diagram* pada modul master adalah sebagai berikut:

#### a. Master Pegawai

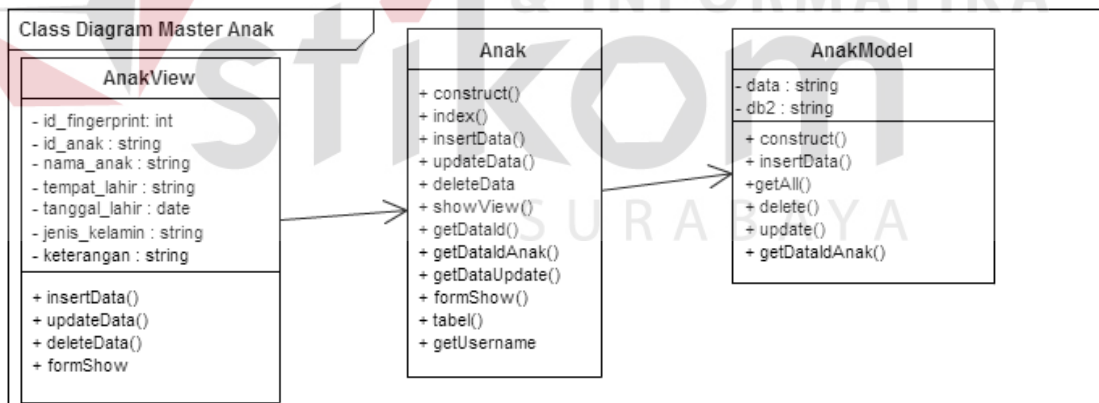
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master* pegawai dapat dilihat pada gambar 3.57.



Gambar 3.57 Class Diagram pada Master Pegawai

## b. Master Anak

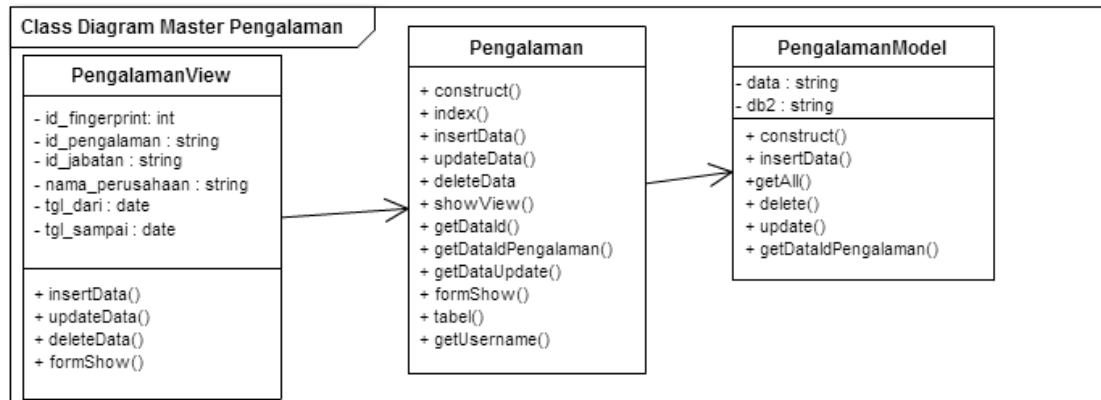
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master* anak dapat dilihat pada gambar 3.58.



Gambar 3.58 Class Diagram pada Master Anak

## c. Master Pengalaman

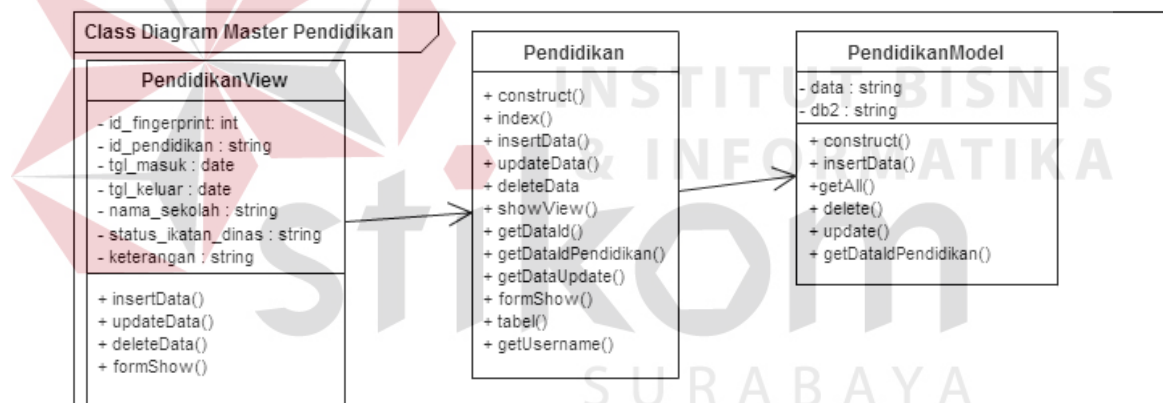
Master Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master* pengalaman dapat dilihat pada gambar 3.59



Gambar 3.59 Class Diagram pada Master Pengalaman

#### d. Master Pendidikan

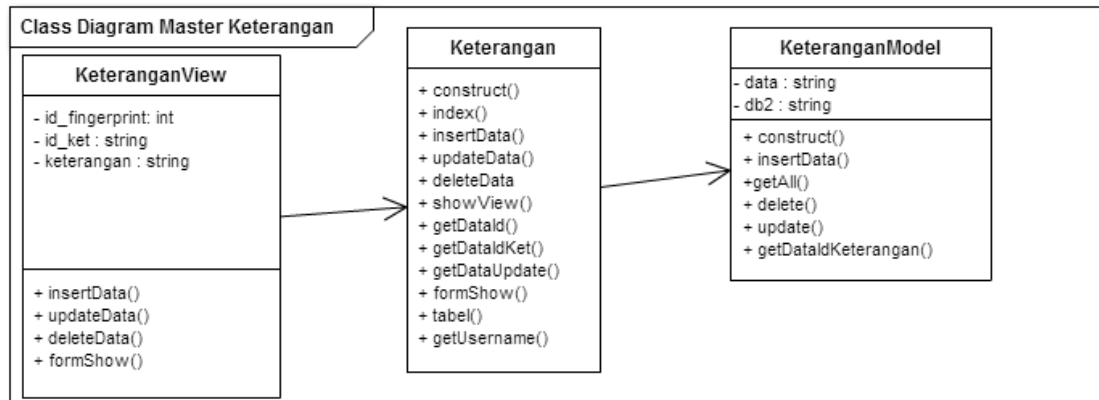
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master pendidikan* dapat dilihat pada gambar 3.60



Gambar 3.60 Class Diagram pada Master Pendidikan

#### e. Master Keterangan

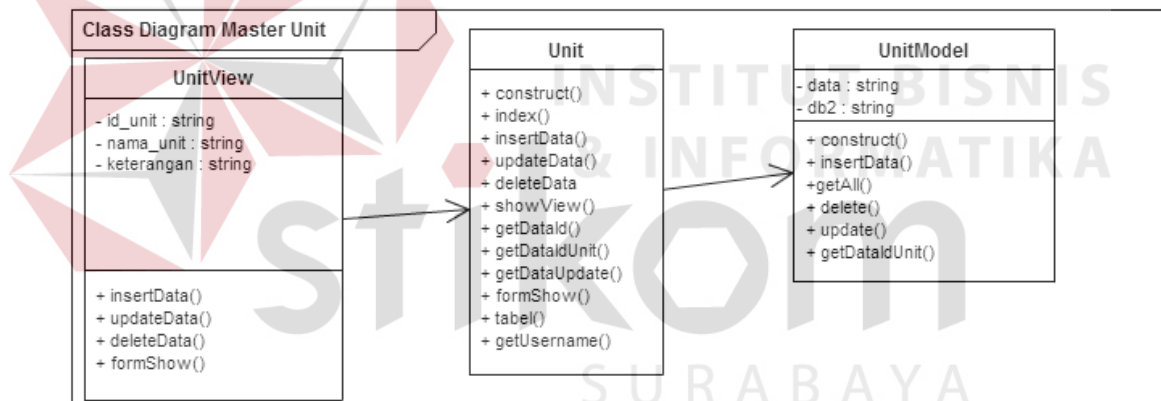
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master keterangan* dapat dilihat pada gambar 3.61



Gambar 3.61 Class Diagram pada Master Keterangan

## f. Master Unit

Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master unit* dapat dilihat pada gambar 3.62

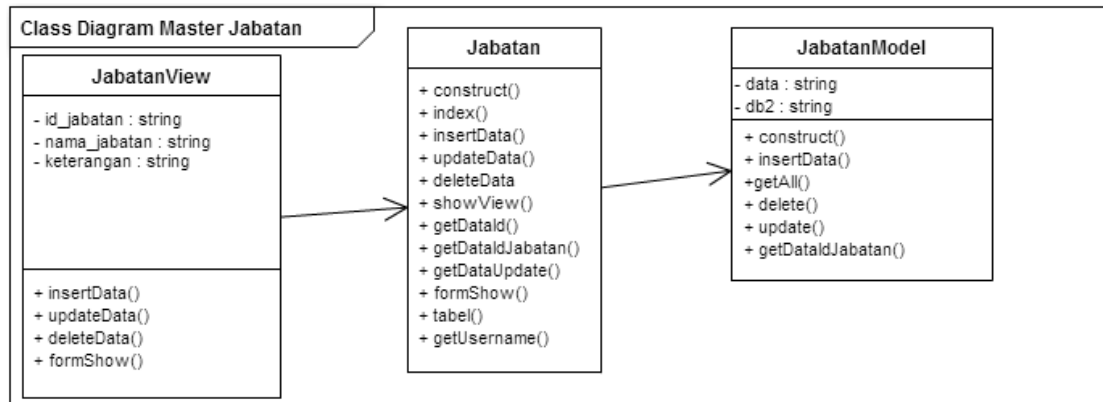


Gambar 3.62 Class Diagram pada Master Unit

## g. Master Jabatan

Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master jabatan* dapat dilihat pada gambar 3.63.

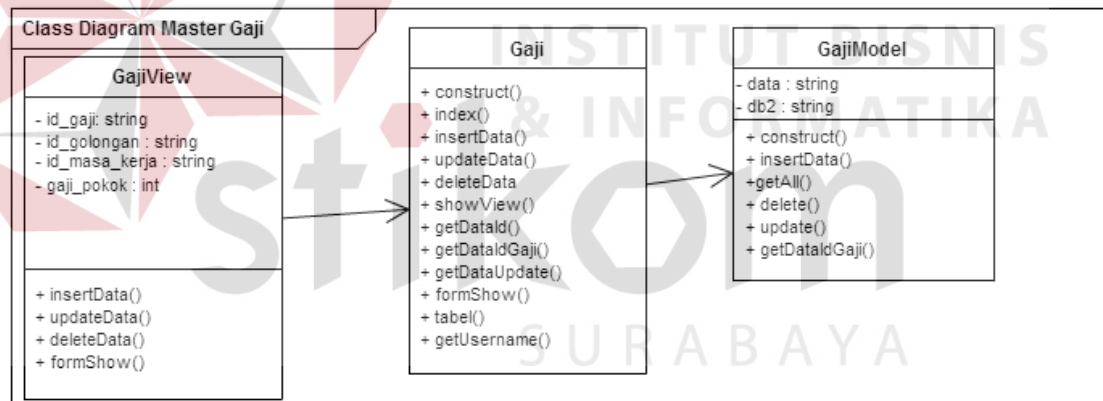




Gambar 3.63 Class Diagram pada Master Jabatan

## h. Master Gaji

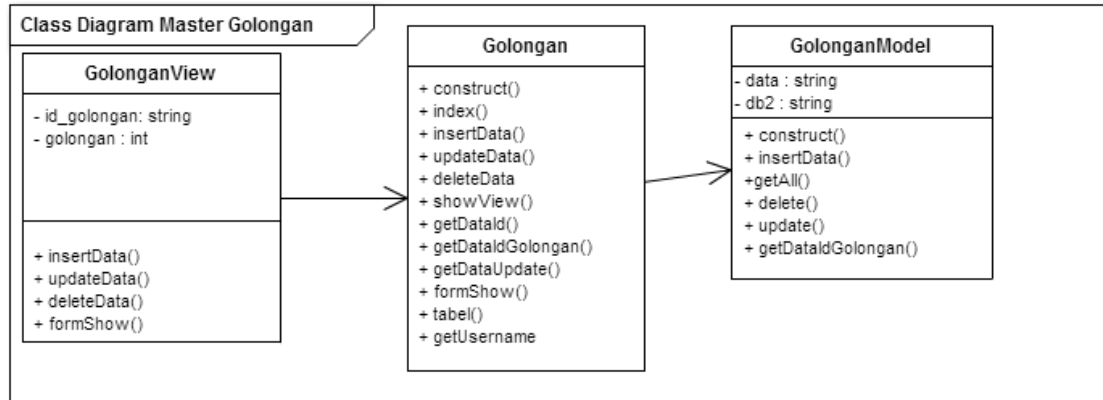
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master gaji* dapat dilihat pada gambar 3.64.



Gambar 3.64 Class Diagram pada Master Gaji

## i. Master Golongan

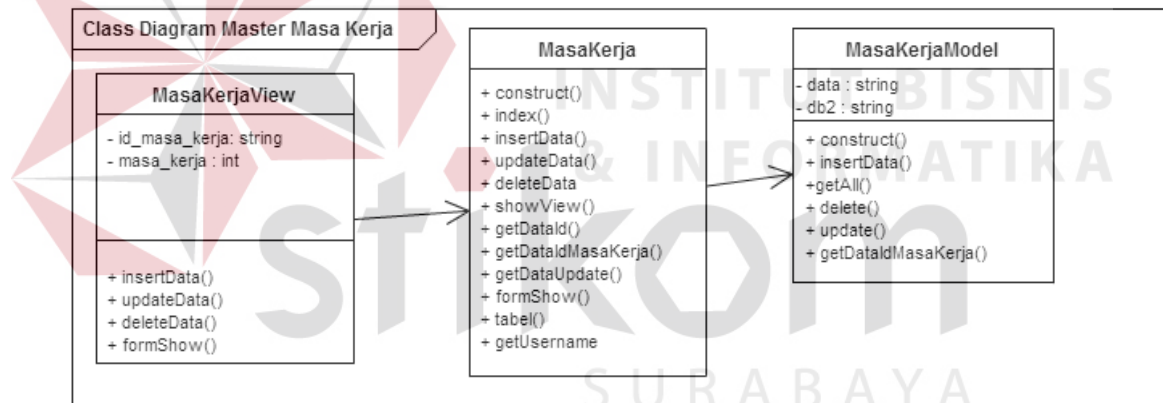
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master golongan* dapat dilihat pada gambar 3.65.



Gambar 3.65 Class Diagram pada Master Golongan

## j. Master Masa Kerja

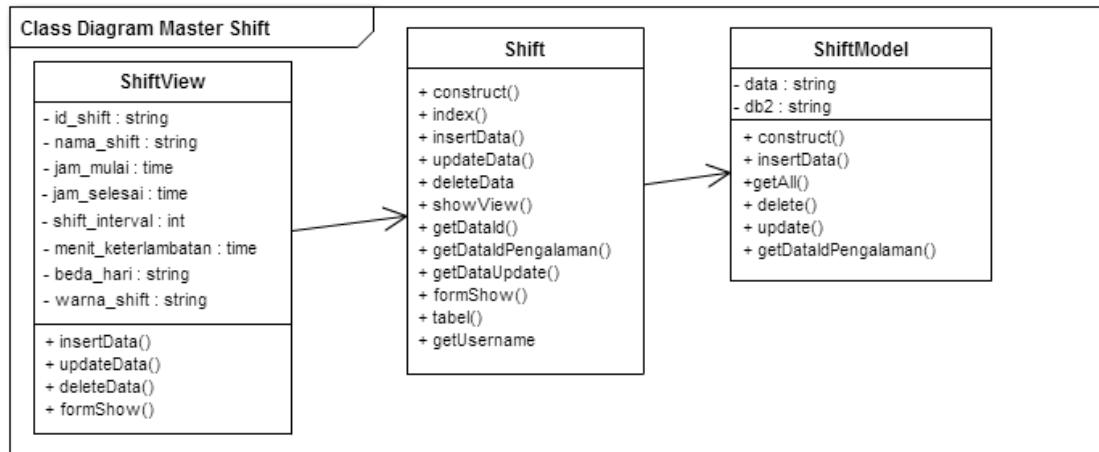
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master* masa kerja dapat dilihat pada gambar 3.66.



Gambar 3.66 Class Diagram pada Master Masa Kerja

## k. Master Shift

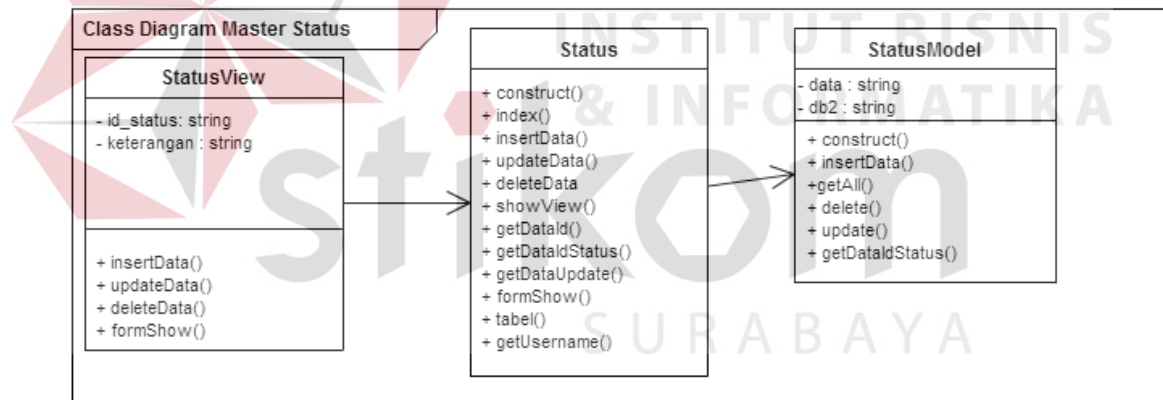
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master* shift dapat dilihat pada gambar 3.67.



Gambar 3.67 Class Diagram pada Master Shift

### l. Master Status Log

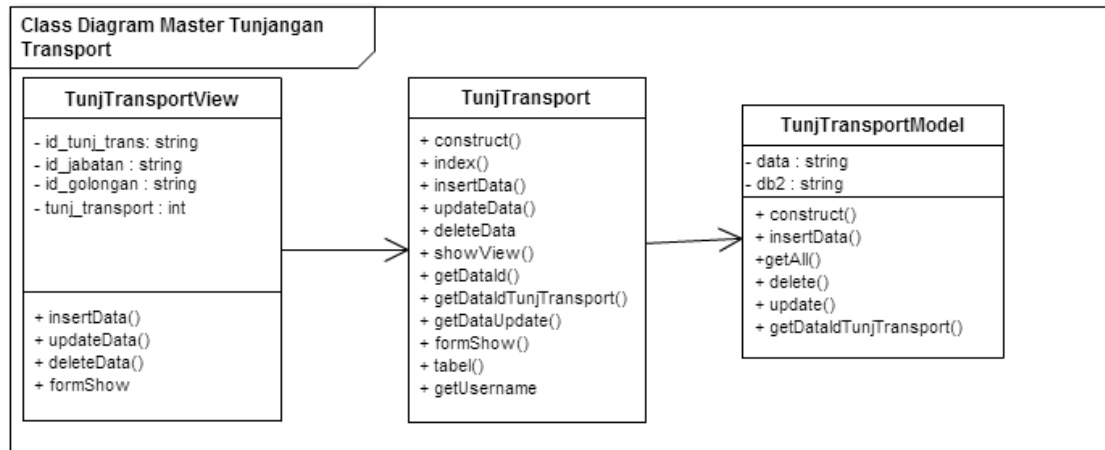
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master status log* dapat dilihat pada gambar 3.68.



Gambar 3.68 Class Diagram pada Master Status Log

### m. Master Tunjangan Transportasi

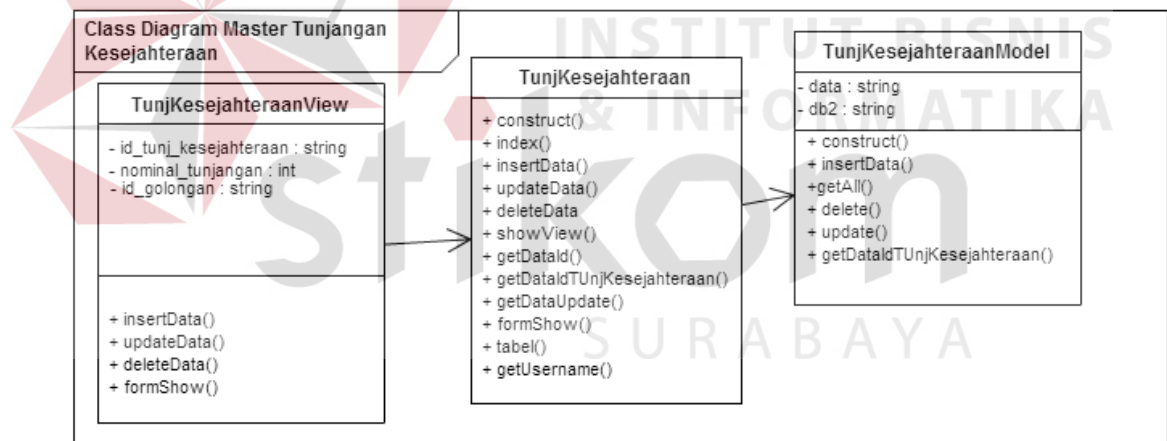
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master tunjangan transportasi* dapat dilihat pada gambar 3.69.



Gambar 3.69 Class Diagram pada Master Tunjangan Transportasi

#### n. Master Tunjangan Kesejahteraan

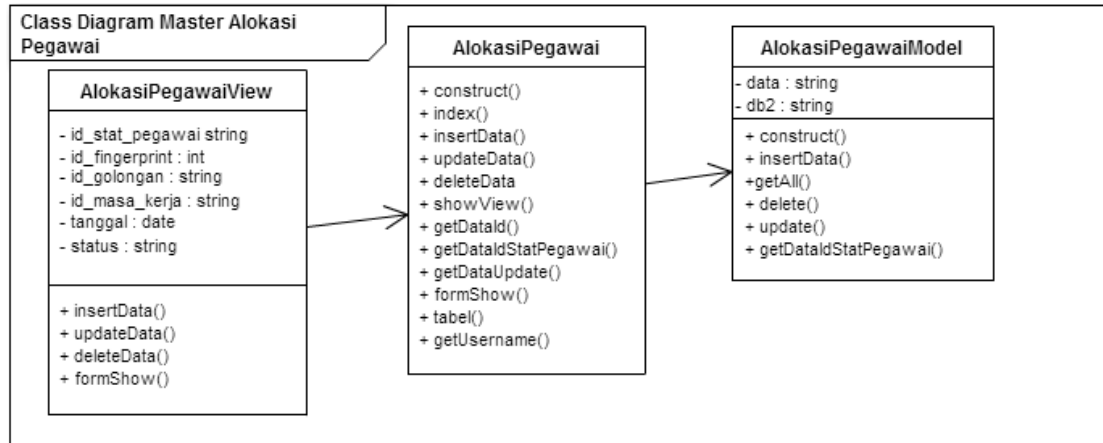
Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master* tunjangan kesejahteraan dapat dilihat pada gambar 3.70.



Gambar 3.70 Class Diagram pada Master Tunjangan Kesejahteraan

#### o. Master Alokasi Pegawai

Relasi *class diagram* yang menunjukkan berbagai variabel dan fungsi yang terdapat pada *master* alokasi pegawai dapat dilihat pada gambar 3.71.



Gambar 3.71 Class Diagram pada Master Alokasi Pegawai

#### D. Desain Interface Modul Master

Desain *user interface* ini akan menjadi acuan nantinya dalam mengembangkan aplikasi. Desain *user interface* pada modul master adalah sebagai berikut

##### a. Master Pegawai

Desain *user interface* yang nantinya akan menjadi acuan dalam membuat form untuk menambahkan data pada *master* pegawai dapat dilihat pada gambar 3.72, gambar 3.73 dan gambar 3.74, sedangkan untuk desain UI view master pegawai dapat dilihat pada gambar 3.75.

The screenshot shows a web application interface for 'Master Pegawai'. At the top, there is a navigation menu with 'Master', 'Transactional', and 'Payroll' options. A user login status is displayed as 'anda login sebagai : nama\_pegawai' with a 'Logout' link. The main content area is titled 'Master Pegawai' and contains three tabs: 'Data Diri Pegawai', 'Data Pelengkap', and 'Data Pernikahan'. The 'Data Diri Pegawai' tab is active, showing a form with the following fields: 'ID Fingerprint' (id\_fingerprint), 'NIK' (nik), 'Nama Pegawai' (nama\_pegawai), 'Nama Panggilan' (nama\_panggilan), 'Gelar Depan' (gelar\_depan), 'Gelar Belakang' (gelar\_belakang), and 'Unit / Jabatan' (id\_unit / id\_jabatan). There is a 'Photo\_path' field with a 'Pilih File' button and a 'Submit' button. A 'Forward >>' button is located at the bottom right of the form. The footer of the page contains the text 'Copyright'.

Gambar 3.72 Desain User Interface Form Input Data Diri Pegawai

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**

Tambah Data

Lihat Data

Detail Pegawai

Master Pegawai

Data Diri Pegawai Data Pelengkap Data Pernikahan

Tempat Lahir : tempat\_lahir

Tanggal Lahir : tanggal\_lahir

Kebangsaan : kebangsaan

Agama : agama

Tinggi Badan : tinggi\_badan

Berat Badan : berat\_badan

<< Previous Forward >>

Submit

Copyright

Gambar 3.73 Desain User Interface Form Input Data Pelengkap

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**

Tambah Data

Lihat Data

Detail Pegawai

Master Pegawai

Data Diri Pegawai Data Pelengkap Data Pernikahan

Nama Pasangan : nama\_pasangan

Tempat Lahir : tempat\_lahir\_pasangan

Tanggal Lahir : tgl\_lahir\_pasangan

Tempat Nikah : tempat\_nikah

Tanggal Menikah : tanggal\_nikah

Nomor Pernikahan : nomor\_nikah

Tanggal Surat Nikah : tgl\_nikah

<< Previous

Submit

Copyright

Gambar 3.74 Desain User Interface Form Input Data Pernikahan

Logo

anda login sebagai : nama\_pegawai [Logout](#)

Master   Transactional   Payroll

	Master Employee						
	ID Fingerprint	NIK	Nama Pegawai	Nama Panggilan	Posisi Jabatan Pegawai	Tempat/Tanggal Lahir	Kebangsaan
Tambah Pegawai	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
Lihat Pegawai	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
Export Data	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
Detail Pegawai	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
Anak	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
Pengalaman Kerja	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
History Pendidikan	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
Keterangan	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan
	id_fingerprint	nik	nama_pegawai	nama_panggilan	nama_unit/nama_jabatan	tempat_lahir/tanggal_lahir	Kebangsaan

Copyright

Gambar 3.75 Desain User Interface Form View Master Pegawai

Desain form *input* pegawai tetap satu form, tetapi nantinya form input dalam form tersebut bisa di-*slide* dengan menekan tombol *Forward* dan *Previous*.

#### b. Master Anak

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* anak dapat dilihat pada gambar 3.76 dan gambar 3.77.

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**

- Tambah Data
- Lihat Data
- Detail Pegawai

**Master Anak**

ID FP :

ID Anak :

Nama Anak :

Tempat Lahir :

Tanggal Lahir :

Jenis Kelamin :  Laki-Laki  Perempuan

Keterangan :

Copyright

Gambar 3.76 Desain User Interface Tambah Data pada Master Anak

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**

- Tambah Data
- Lihat Data

**Master Anak**

ID Fingerprint	ID Anak	Nama Anak	Tempat/Tanggal Lahir	Jenis Kelamin	Keterangan	Actions
id_fingerprint	id_anak	nama_anak	tempat_lahir."".tanggal_lahir	jenis_kelamin	keterangan	edit delete
id_fingerprint	id_anak	nama_anak	tempat_lahir."".tanggal_lahir	jenis_kelamin	keterangan	edit delete
id_fingerprint	id_anak	nama_anak	tempat_lahir."".tanggal_lahir	jenis_kelamin	keterangan	edit delete
id_fingerprint	id_anak	nama_anak	tempat_lahir."".tanggal_lahir	jenis_kelamin	keterangan	edit delete
id_fingerprint	id_anak	nama_anak	tempat_lahir."".tanggal_lahir	jenis_kelamin	keterangan	edit delete
id_fingerprint	id_anak	nama_anak	tempat_lahir."".tanggal_lahir	jenis_kelamin	keterangan	edit delete

Copyright

Gambar 3.77 Desain User Interface Lihat View pada Master Anak

### c. Master Pengalaman

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* pengalaman dapat dilihat pada gambar 3.78 dan gambar 3.79.



anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**

Tambah Data

Lihat Data

Detail Pegawai

Master Pengalaman

ID FP :

ID Pengalaman :

Nama Jabatan :

Di Perusahaan :

Dari Tanggal :

Sampai Tanggal :

Copyright

Gambar 3.78 User Interface Tambah Data pada Master Pengalaman

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**

Tambah Data

Lihat Data

Master Pengalaman

ID Pengalaman	ID Fingerprint	ID Jabatan	Di Perusahaan	Dari Tanggal	Sampai Tanggal	Actions
id_pengalaman	id_fingerprint	id_jabatan	nama_perusahaan	tanggal_dari	tanggal_sampai	edit delete
id_pengalaman	id_fingerprint	id_jabatan	nama_perusahaan	tanggal_dari	tanggal_sampai	edit delete
id_pengalaman	id_fingerprint	id_jabatan	nama_perusahaan	tanggal_dari	tanggal_sampai	edit delete
id_pengalaman	id_fingerprint	id_jabatan	nama_perusahaan	tanggal_dari	tanggal_sampai	edit delete
id_pengalaman	id_fingerprint	id_jabatan	nama_perusahaan	tanggal_dari	tanggal_sampai	edit delete
id_pengalaman	id_fingerprint	id_jabatan	nama_perusahaan	tanggal_dari	tanggal_sampai	edit delete

Copyright

Gambar 3.79 Desain User Interface Lihat View pada Master Pengalaman

#### d. Master Pendidikan

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* pendidikan dapat dilihat pada gambar 3.80 dan gambar 3.81.

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**  
 Tambah Data  
 Lihat Data  
 Detil Pegawai

**Master Pendidikan**

ID FP :

ID Pendidikan :

Tanggal Masuk :

Tanggal Lulus :

Nama Sekolah :

Ikatan Dinas :

Keterangan :

Copyright

Gambar 3.80 Desain User Interface Tambah Data pada Master Pendidikan

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

**Action**  
 Tambah Data  
 Lihat Data

**Master Pendidikan**

ID Pendidikan	ID Fingerprint	Tanggal Masuk	Tanggal Lulus	Nama Sekolah	Ikatan Dinas	Keterangan	Actions
id_pendidikan	id_fingerprint	tgl_masuk	tgl_lulus	nama_sekolah	status_ikatan_dinas	keterangan	edit delete
id_pendidikan	id_fingerprint	tgl_masuk	tgl_lulus	nama_sekolah	status_ikatan_dinas	keterangan	edit delete
id_pendidikan	id_fingerprint	tgl_masuk	tgl_lulus	nama_sekolah	status_ikatan_dinas	keterangan	edit delete
id_pendidikan	id_fingerprint	tgl_masuk	tgl_lulus	nama_sekolah	status_ikatan_dinas	keterangan	edit delete
id_pendidikan	id_fingerprint	tgl_masuk	tgl_lulus	nama_sekolah	status_ikatan_dinas	keterangan	edit delete
id_pendidikan	id_fingerprint	tgl_masuk	tgl_lulus	nama_sekolah	status_ikatan_dinas	keterangan	edit delete

Copyright

Gambar 3.81 Desain User Interface Lihat View pada Master Pendidikan

#### e. Master Keterangan

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* keterangan dapat dilihat pada gambar 3.82 dan gambar 3.83.

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

Action

Tambah Data

Lihat Data

Detail Pegawai

Master Keterangan

ID FP :

ID Keterangan :

Keterangan :

Submit Reset

Copyright

Gambar 3.82 Desain User Interface Tambah Data pada Master Keterangan

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

Action

Tambah Data

Lihat Data

Master Keterangan

ID Keterangan	ID Fingerprint	Keterangan	Actions
id_keterangan	id_fingerprint	keterangan	edit delete
id_keterangan	id_fingerprint	keterangan	edit delete
id_keterangan	id_fingerprint	keterangan	edit delete
id_keterangan	id_fingerprint	keterangan	edit delete
id_keterangan	id_fingerprint	keterangan	edit delete
id_keterangan	id_fingerprint	keterangan	edit delete

Copyright

Gambar 3.83 Desain User Interface Lihat View pada Master Keterangan

f. Master Unit

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* unit dapat dilihat pada gambar 3.84 dan gambar 3.85.

Gambar 3.84 Desain User Interface Tambah Data pada Master Unit

ID Unit	Nama Unit	Keterangan	Actions
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete
id_unit	nama_unit	keterangan	edit delete

Gambar 3.85 Desain User Interface Lihat View pada Master Unit

#### g. Master Jabatan

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master jabatan* dapat dilihat pada gambar 3.86 dan gambar 3.87.

anda login sebagai : nama\_pegawai [Logout](#)

Master Transactional Payroll

**Action**

Tambah Data

Lihat Data

Master Jabatan

ID Jabatan :

Nama Jabatan :

Keterangan :

Copyright

Gambar 3.86 Desain User Interface Tambah Data pada Master Jabatan

anda login sebagai : nama\_pegawai [Logout](#)

Master Transactional Payroll

**Action**

Tambah Data

Lihat Data

Master Jabatan

ID Jabatan	Nama Jabatan	Keterangan	Actions
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete

Copyright

Gambar 3.87 Desain User Interface Lihat View pada Master Jabatan

#### h. Master Gaji

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master gaji* dapat dilihat pada gambar 3.88.

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

Detil

Golongan

Masa Kerja

Master Gaji

ID Gaji :

Golongan / Masa Kerja :

Gaji Pokok :

ID Jabatan	Nama Jabatan	Keterangan	Actions
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete
id_jabatan	nama_jabatan	keterangan	edit delete

Copyright

Gambar 3.88 Desain User Interface pada Master Gaji

## i. Master Golongan

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* golongan dapat dilihat pada gambar 3.89.

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

Detil

Golongan

Masa Kerja

Master Golongan

ID Golongan :

Golongan :

ID Golongan	Golongan	Actions
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete

Copyright

Gambar 3.89 Desain User Interface pada Master Golongan

### j. Master Masa Kerja

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* masa kerja dapat dilihat pada gambar 3.90.

ID Golongan	Golongan	Actions
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete
id_golongan	golongan	edit delete

Gambar 3.90 Desain User Interface pada Master Masa Kerja

### k. Master Shift

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* shift dapat dilihat pada gambar 3.91 dan gambar 3.92.

Gambar 3.91 Desain User Interface Tambah Data pada Master Shift

Logo

anda login sebagai : nama\_pegawai [Logout](#)

Master
Transactional
Payroll

Action

Master Shift

Tambah Data

ID Shift	Nama Shift	Jam Masuk	Jam Pulang	Jumlah Jam Kerja	Menit Dispensasi	Beda Hari	Warna Shift	actions
id_shift	nama_shift	jam_masuk	jam_pulang	shift_interval	menit_keterlambatan	beda_hari	warna_shift	edit delete
id_shift	nama_shift	jam_masuk	jam_pulang	shift_interval	menit_keterlambatan	beda_hari	warna_shift	edit delete
id_shift	nama_shift	jam_masuk	jam_pulang	shift_interval	menit_keterlambatan	beda_hari	warna_shift	edit delete
id_shift	nama_shift	jam_masuk	jam_pulang	shift_interval	menit_keterlambatan	beda_hari	warna_shift	edit delete
id_shift	nama_shift	jam_masuk	jam_pulang	shift_interval	menit_keterlambatan	beda_hari	warna_shift	edit delete
id_shift	nama_shift	jam_masuk	jam_pulang	shift_interval	menit_keterlambatan	beda_hari	warna_shift	edit delete
id_shift	nama_shift	jam_masuk	jam_pulang	shift_interval	menit_keterlambatan	beda_hari	warna_shift	edit delete

Lihat Data

Copyright

Gambar 3.92 Desain User Interface Lihat View pada Master Shift

### 1. Master Status Log

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master status log* dapat dilihat pada gambar 3.93.

Logo

anda login sebagai : nama\_pegawai [Logout](#)

Master
Transactional
Payroll

Detil

Master Status Log

Golongan

ID Status :

Keterangan :

ID Status	Keterangan	Actions
id_status	keterangan	edit delete
id_status	keterangan	edit delete
id_status	keterangan	edit delete
id_status	keterangan	edit delete
id_status	keterangan	edit delete

Copyright

Gambar 3.93 Desain User Interface Tambah Data pada Master Status Log



Master status log berfungsi untuk memmanage data status log dimana status log sendiri berfungsi sebagai pembeda antara check in dan check out.

#### m. Master Tunjangan Transportasi

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* pegawai dapat dilihat pada gambar 3.94.

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

Detil  
Golongan  
Masa Kerja

Master Tunjangan Transportasi

ID Status :

Jabatan :  ▼

Golongan :  —

Nominal Tunjangan :

ID Tunjangan	Jabatan	Golongan	Nominal Tunjangan	Actions
id_tunjangan	nama_jabatan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	nama_jabatan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	nama_jabatan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	nama_jabatan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	nama_jabatan	id_golongan	nominal_tunjangan	edit delete

Copyright

Gambar 3.94 Desain User Interface pada Master Tunjangan Transport

#### n. Master Tunjangan Kesejahteraan

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* tunjangan kesejahteraan dapat dilihat pada gambar 3.95.

Logo

anda login sebagai : nama\_pegawai [Logout](#)

Master
Transactional
Payroll

Detil

Golongan

Masa Kerja

Master Tunjangan Kesejahteraan

ID Tunjangan :

Golongan :  ▼

Nominal Tunjangan :

ID Tunjangan	ID Golongan Tunjangan	Nominal Tunjangan	Actions
id_tunjangan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	id_golongan	nominal_tunjangan	edit delete
id_tunjangan	id_golongan	nominal_tunjangan	edit delete

Copyright

Gambar 3.95 Desain User Interface pada Master Tunjangan Kesejahteraan

## o. Master Tunjangan Jabatan

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* tunjangan jabatan dapat dilihat pada gambar 3.96.

Logo

anda login sebagai : nama\_pegawai [Logout](#)

Master
Transactional
Payroll

Detil

Tunj Transportasi

Tunj Kesejahteraan

Master Tunjangan Jabatan

ID Tunjangan :

Jabatan :  ▼

Nominal Tunjangan :

ID Tunjangan	ID Jabatan	Nominal Tunjangan	Actions
id_tunjangan	id_jabatan	nominal_tunjangan	edit delete
id_tunjangan	id_jabatan	nominal_tunjangan	edit delete
id_tunjangan	id_jabatan	nominal_tunjangan	edit delete
id_tunjangan	id_jabatan	nominal_tunjangan	edit delete
id_tunjangan	id_jabatan	nominal_tunjangan	edit delete

Copyright

Gambar 3.96 Desain User Interface pada Master Tunjangan Jabatan

Besarnya tunjangan jabatan yang didapat oleh pegawai bergantung pada tingkatan jabatan yang dia miliki, semakin tinggi jabatan, maka nilai tunjangan jabatan yang didapat akan semakin besar. Maka dari itu pada desain form master jabatan terdapat *combo box* yang berisi id\_jabatan yang nantinya sebagai acuan.

p. Master Alokasi Pegawai

Desain user interface yang nantinya akan menjadi acuan dalam membuat form *master* alokasi pegawai dapat dilihat pada gambar 3.97.

anda login sebagai : nama\_pegawai [Logout](#)

Logo

Master Transactional Payroll

Detil

Golongan

Masa Kerja

Master Alokasi Pegawai

ID Alokasi MG :

ID Fingerprint :

Golongan / Masa Kerja :  /

Tanggal SK :

ID Alokasi MG	ID Fingerprint	Golongan	Masa Kerja	Tanggal	Actions
id_stat_pegawai	id_fingerprint	id_golongan	id_masa_kerja	tanggal	edit delete
id_stat_pegawai	id_fingerprint	id_golongan	id_masa_kerja	tanggal	edit delete
id_stat_pegawai	id_fingerprint	id_golongan	id_masa_kerja	tanggal	edit delete
id_stat_pegawai	id_fingerprint	id_golongan	id_masa_kerja	tanggal	edit delete

Copyright



Gambar 3.97 Desain User Interface pada Master Alokasi Pegawai

## E. Unit Testing Modul Master

Uji coba dalam modul master ini untuk menguji apakah berbagai fungsi yang terdapat dalam modul master dapat berjalan sesuai dengan yang diharapkan.

Desain uji coba modul master dapat dilihat pada tabel 3.11

Tabel 3.11 Uji Coba Modul Master

Test Case ID	Tujuan	Input	Output yang diharapkan	User stories
1	Menambah data	Field yang tersedia pada tiap form	Data dalam <i>database</i> dapat ditambahkan sesuai dengan inputan.	U2, U3, U4, U5, U6, U7, U8, U9, U10, U11, U12, U13, U14
2	Mengedit data	User melakukan aksi klik pada tombol 	Data yang telah terdapat dalam <i>database</i> bisa dirubah.	U2, U3, U4, U5, U6, U7, U8, U9, U10, U11, U12, U13, U14
3	Menghapus data	User melakukan aksi klik pada tombol 	Data yang telah berada dalam <i>database</i> bisa dihapus.	U2, U3, U4, U5, U6, U7, U8, U9, U10, U11, U12, U13, U14
4	Mengkspor data	Menekan tombol export data pada <i>master employee</i>	Data dalam <i>database</i> karyawan dapat terisi data dari <i>database</i> fingerprint.	U2, U3, U4, U5, U6, U7, U8, U9, U10, U11, U12, U13, U14

### 3. Iterasi Penjadwalan Shift Pegawai

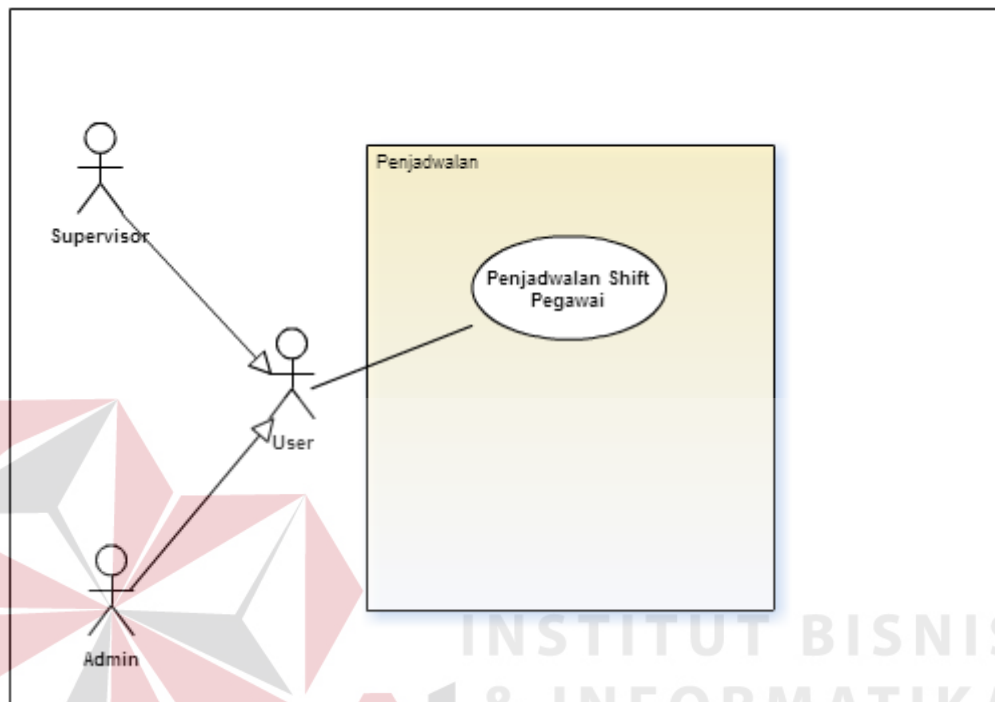
Proses perencanaan penjadwalan dilakukan oleh supervisor dari tiap unit untuk menentukan jadwal shift pegawai yang ada dalam unit tersebut. Penjadwalan inilah yang nantinya dijadikan acuan dalam penghitungan absensi pegawai dengan cara dicocokkan dengan realisasi absensi tiap hari. Pada tabel 3.12 akan ditunjukkan hak akses dari modul ini.

Tabel 3. 12 Hak Akses Modul Penjadwalan Shift Pegawai

	HRD	Supervisor	Manager	Accounting	Admin
Penjadwalan		√			√

### A. Diagram Use Case Modul Penjadwalan Shift Pegawai

Berikut ini adalah diagram *use case* dari modul master penjadwalan yang dapat dilihat pada gambar 3.98.



Gambar 3.98 Diagram Use Case Modul Penjadwalan Shift

### B. Flow of Events Modul Penjadwalan Shift Pegawai

Modul penjadwalan memiliki 3 fungsi utama yaitu input penjadwalan, view penjadwalan dan update penjadwalan. Modul telah dilengkapi dengan fitur *filter staff* yang memungkinkan hanya staff yang berada dibawah supervisor pada department tersebut yang hanya tampil untuk direncanakan penjadwalannya. Berikut ini adalah flow of event dalam modul penjadwalan shift pegawai.

a. Flow of Event Input Penjadwalan

Flow of event untuk input penjadwalan shift pegawai dapat dilihat pada tabel 3.14

Tabel 3.13 Flow of Event Insert Penjadwalan Shift Pegawai

<b>Deskripsi Use case Detail</b>	<b>Keterangan</b>	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>insert data</i>	
<b>Tujuan</b>	<i>User</i> bisa menambahkan data penjadwalan ke <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa menambahkan data.	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka data tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman Penjadwalan.	
<b>Alur Utama</b>		
	1	Use case dimulai ketika <i>user</i> berada pada form Penjadwalan.
	2	<i>Use case</i> dimulai ketika <i>user</i> mengisi teks box yang telah tersedia
	3	<i>User</i> menekan tombol "Generate Schedule"
	4	Sistem akan melakukan koneksi <i>database</i> dan memasukkan data yang telah di- <i>input</i> . jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

b. Flow of Event Input Penjadwalan

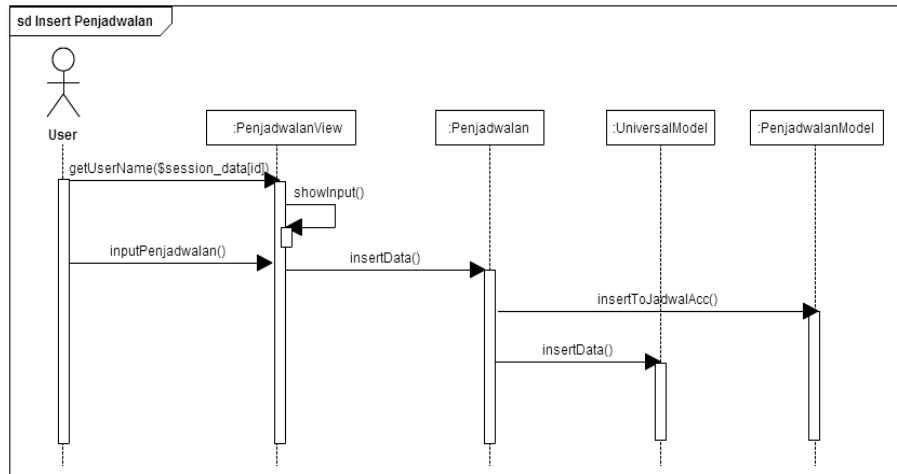
Flow of event untuk input penjadwalan shift pegawai dapat dilihat pada tabel 3.15.

Tabel 3.14 Flow of Event Update Pada Modul Penjadwalan

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>update data</i>	
<b>Tujuan</b>	<i>User</i> bisa mengubah data pada <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa mengubah data.	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka data tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman Penjadwalan.	
<b>Alur Utama</b>		
	1	<i>Use case</i> dimulai ketika <i>user</i> berada pada form penjadwalan.
	2	<i>Use case</i> dimulai ketika <i>user</i> masuk kedalam mode view dan menekan tombol <input type="checkbox"/> didalam kolom "Actions" pada tabel yang terdapat di Form Master
	3	<i>User</i> mengubah data yang tampil pada teks boks yang tersedia.
	4	<i>User</i> menekan tombol "submit".
	5	Sistem akan melakukan koneksi <i>database</i> dan meng- <i>update</i> data yang telah diubah. jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 5.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	5.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

### C. Sequence Diagram Modul Penjadwalan Shift Pegawai

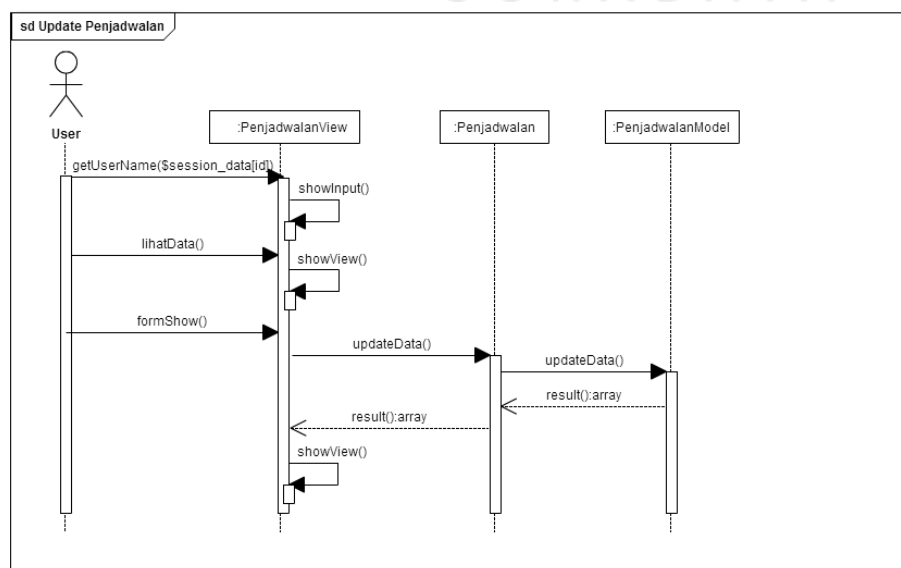
*Sequence diagram* untuk *use case insert* penjadwalan dapat dilihat pada gambar 3.99.



Gambar 3.99 Sequence Diagram pada Insert Penjadwalan

Pada sequence diagram tersebut, ketika *function* `insertData()` selesai dijalankan, proses akan dilanjutkan pada menyimpan data dalam *database* dan juga membuat sebuah id dalam tabel verifikasi penjadwalan. Id dalam tabel verifikasi penjadwalan ini nantinya akan menjadi acuan bagi seorang manajer dalam melakukan verifikasi pada rencana penjadwalan yang telah dibuat oleh supervisor masing-masing unit.

*Sequence diagram* untuk *use case update* penjadwalan dapat dilihat pada gambar 3.100.

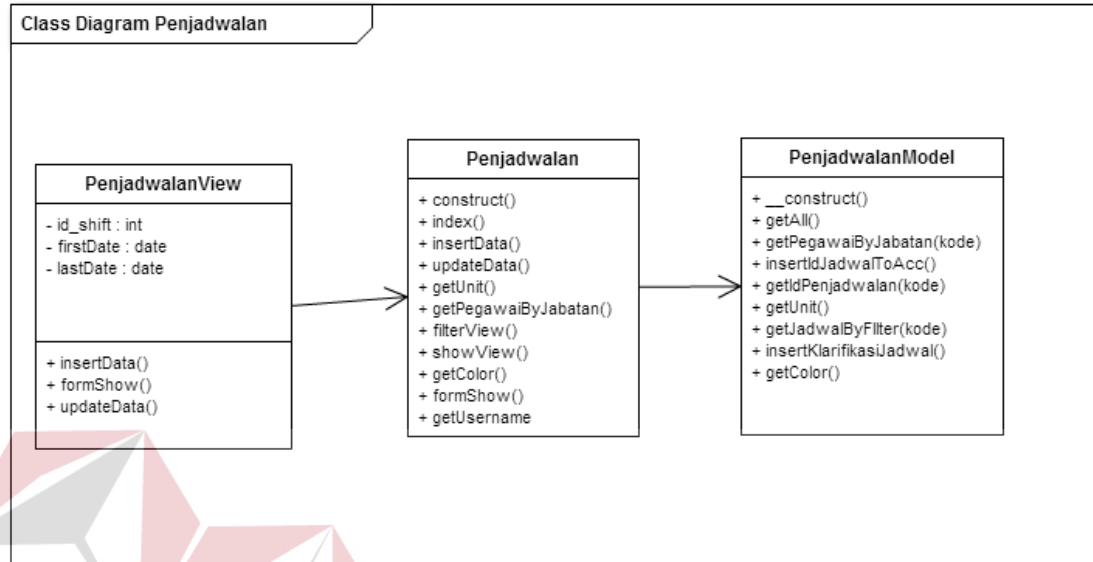


Gambar 3.100 Sequence Diagram pada Use Case Update Penjadwalan



#### D. Class Diagram Modul Penjadwalan Shift Pegawai

Class diagram untuk use case penjadwalan dapat dilihat pada gambar 3.101.



Gambar 3.101 Class Diagram pada Use Case Penjadwalan

#### E. Desain Interface Modul Penjadwalan Shift Pegawai

Desain interface untuk modul penjadwalan shift pegawai dapat dilihat pada gambar 3.102.

Gambar 3.102 Desain Interface pada Form Penjadwalan

## F. Unit Testing Modul Penjadwalan Shift Pegawai

Uji coba dalam modul master ini untuk menguji apakah berbagai fungsi yang terdapat dalam modul master dapat berjalan sesuai dengan yang diharapkan.

Desain uji coba modul penjadwalan dapat dilihat pada tabel 3.16.

Tabel 3.15 Uji Coba Pada Modul Penjadwalan

Test Case ID	Tujuan	Input	Output yang diharapkan	User stories	Status
1	Menambah data	Field yang tersedia pada tiap form	Data dalam <i>database</i> dapat ditambahkan sesuai dengan inputan.	U15	Sukses
2	Mengedit data	User melakukan aksi klik pada tombol <input type="checkbox"/>	Data yang telah terdapat dalam <i>database</i> bisa dirubah.	U15	Sukses

Output unit testing dari fungsi utama yang dimiliki oleh form penjadwalan dan master Employee dapat dilihat pada tabel 3.17.

Tabel 3.16 Output Unit Testing pada Modul Penjadwalan

No	Tes	Status
1	Insert_penjadwalan_test	Passed
2	Update_penjadwalan_test	Passed

## 4. Iterasi Verifikasi Jadwal Shift

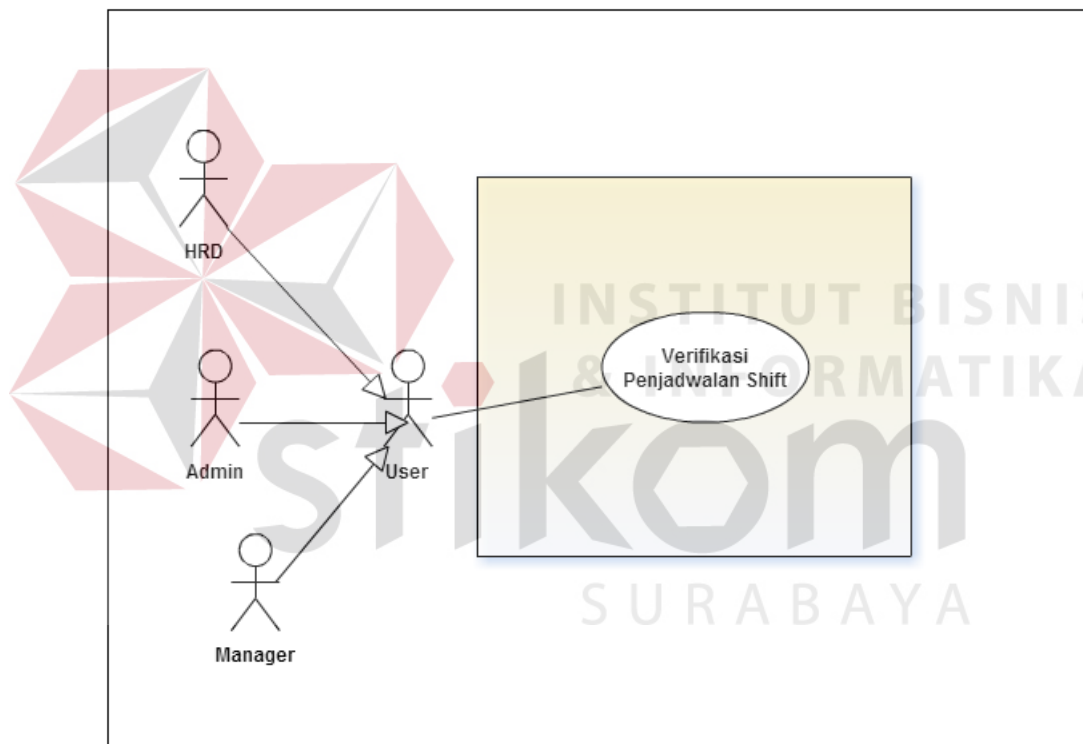
Proses verifikasi penjadwalan adalah proses dimana para manager unit dan HRD melihat dan memverifikasi perencanaan jadwal yang dibuat oleh supervisor unit. Apabila ada hal yang tidak sesuai dengan keinginan manajer, manajer bisa meminta supervisor untuk melakukan kroscek kembali untuk melakukan revisi dengan cara menggunakan fitur edit yang telah tersedia dalam aplikasi. Pada tabel 3.18 akan ditunjukkan hak akses dari modul ini.

Tabel 3. 17 Hak Akses Modul Verifikasi Penjadwalan Shift Pegawai

	HRD	Supervisor	Manager	Accounting	Admin
Verifikasi Penjadwalan			√		√

#### a. Diagram Use Case Modul Verifikasi Penjadwalan Shift

Berikut ini adalah diagram *use case* dari modul verifikasi penjadwalan shift yang dapat dilihat pada gambar 3.103.



Gambar 3.103 Diagram Use Case pada Modul Verifikasi Penjadwalan Shift

#### b. Flow of Events Modul Verifikasi Penjadwalan

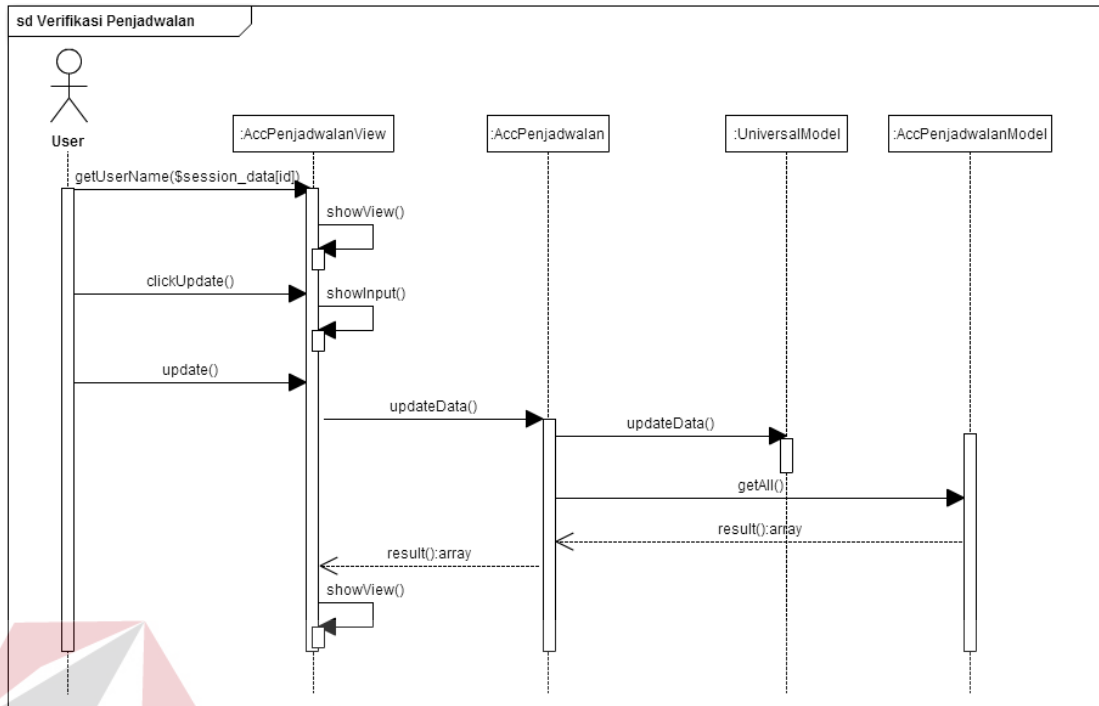
*Flow of event* untuk modul *verifikasi penjadwalan shift* dapat dilihat pada tabel 3.19.

Tabel 3.18 Flow of Event Verifikasi Penjadwalan

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>update data</i>	
<b>Tujuan</b>	<i>User</i> bisa mencentang <i>check box</i> dalam aplikasi dan menyimpannya dalam <i>database</i> sebagai status verifikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa meng-update data	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka data tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman Penjadwalan.	
<b>Alur Utama</b>		
	1	Use case dimulai ketika <i>user</i> berada pada form Verifikasi Penjadwalan.
	2	<i>Use case</i> dimulai ketika <i>user</i> melakukan click pada tombol <input type="checkbox"/>
	3	<i>User</i> melakukan <i>approval</i> dengan mencentang <i>check box</i> yang telah tersedia.
	4	Sistem akan melakukan koneksi <i>database</i> dan memasukkan data yang telah di- <i>input</i> . jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

### c. Sequence Diagram Modul Master

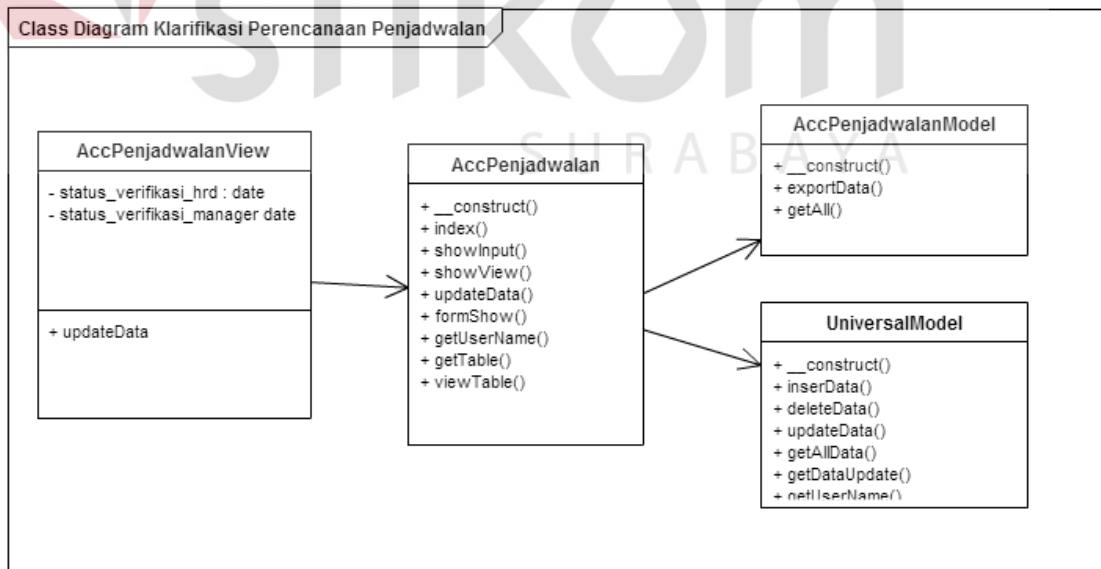
*Sequence diagram* untuk *use case* verifikasi penjadwalan dapat dilihat pada gambar 3.104.



Gambar 3.104 Sequence Diagram pada Modul Verifikasi Penjadwalan

**c. Class Diagram Modul Verifikasi Penjadwalan**

Class diagram untuk *use case* verifikasi penjadwalan dapat dilihat pada gambar 3.105



Gambar 3.105 Class Diagram pada Use Case Verifikasi Penjadwalan

#### d. Desain Interface Modul Verifikasi Penjadwalan

Desain interface untuk modul penjadwalan shift pegawai dapat dilihat pada gambar 3.106

Gambar 3.106 Desain *Interface* pada *Form* Verifikasi Penjadwalan

#### e. Unit Testing Modul Verifikasi Penjadwalan

Uji coba dalam modul verifikasi penjadwalan ini untuk menguji apakah berbagai fungsi yang terdapat berjalan sesuai dengan yang diharapkan. Desain uji coba modul verifikasi penjadwalan dapat dilihat pada tabel 3.19.

Tabel 3.19 Uji Coba pada Modul Verifikasi Penjadwalan

Test Case ID	Tujuan	Input	Output yang diharapkan	User stories
1	Menambah data	Field yang tersedia pada tiap form	Data dalam <i>database</i> dapat di tambahkan sesuai dengan inputan.	U16

## 5. Iterasi Export Log Pegawai

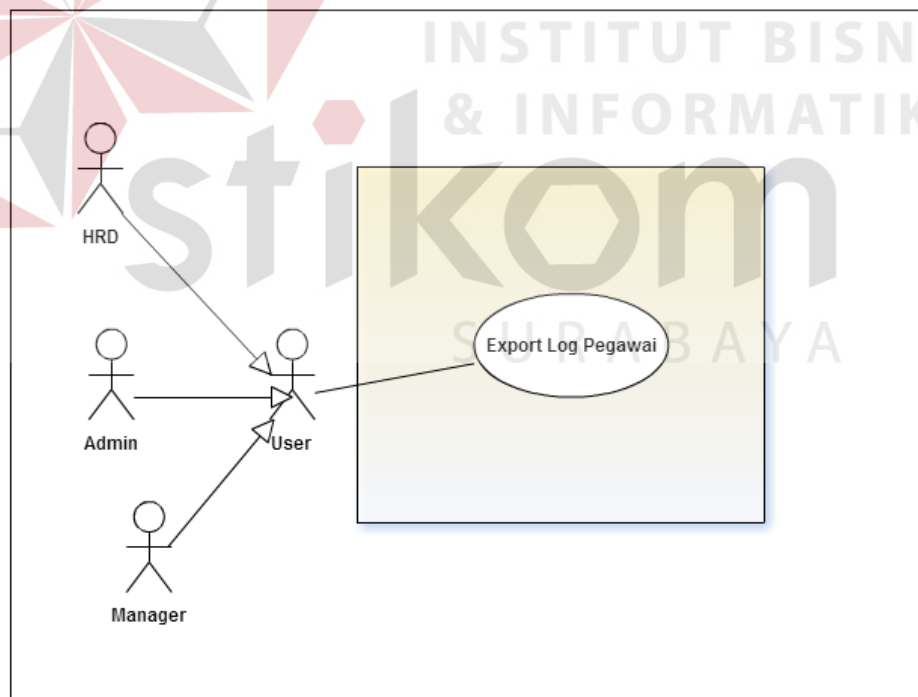
Modul export log pegawai adalah modul yang akan paling sering dipakai, dimana petugas HRD akan melakukan aktivitas *retrieve* realisasi absensi pegawai setiap harinya dengan mengambil data absensi dari *database* mesin *fingerprint* ke *database* aplikasi. Pada tabel 3.20 akan ditunjukkan hak akses dari modul ini.

Tabel 3. 20 Hak Akses Modul Export Log Pegawai

	HRD	Supervisor	Manager	Accounting	Admin
Export Log	√				√

### a. Diagram Use Case Modul Export Log Pegawai

Berikut ini adalah digram *use case* dari modul *export log* pegawai yang dapat dilihat pada gambar 3.107



Gambar 3.107 Diagram Use Case Modul Export Log Pegawai

### b. Flow of Events Modul Export Log Pegawai

Flow of event untuk modul *export log* pegawai dapat dilihat pada tabel 3.21.

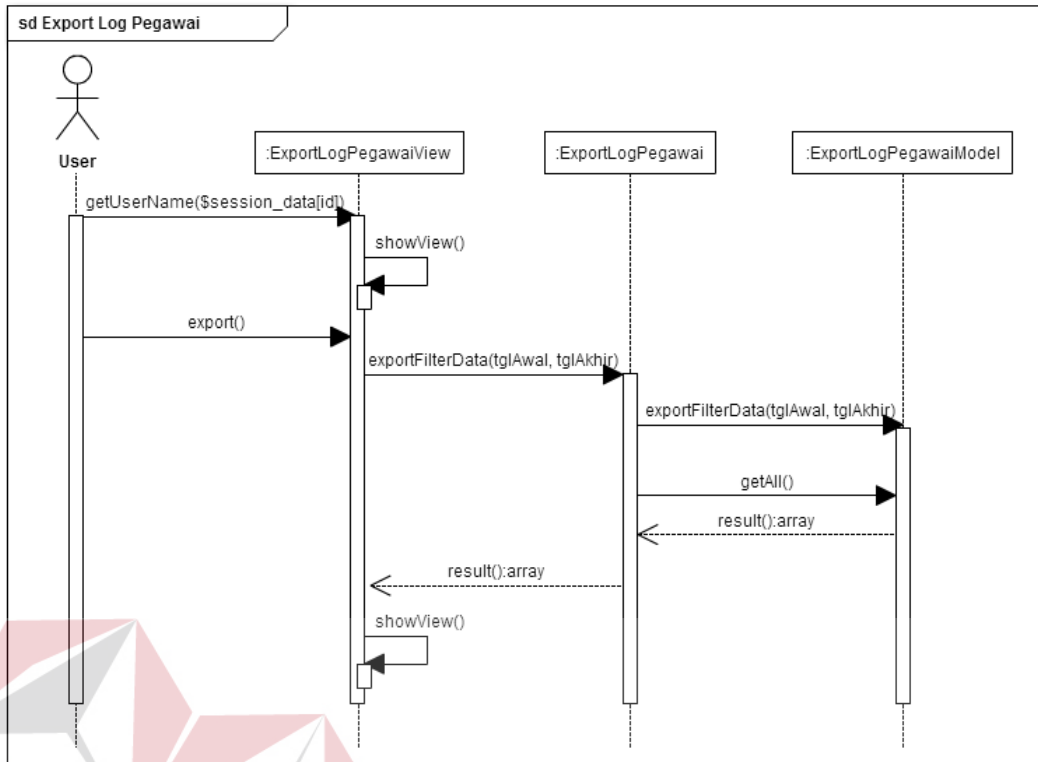
Tabel 3.21 Flow of Event pada Modul Export Log Pegawai

<b>Deskripsi Use case Detail</b>	<b>Keterangan</b>	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>insert data</i>	
<b>Tujuan</b>	<i>User</i> bisa menambahkan data ke <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa menambahkan data.	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka data tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman <i>Log Pegawai</i> dari data yang akan ditambahkan	
<b>Alur Utama</b>		
	1	<i>Use case</i> dimulai ketika <i>user</i> berada pada <i>form log pegawai</i> .
	2	<i>Use case</i> dimulai ketika <i>user</i> mengisi tanggal awal dan tanggal akhir
	3	<i>User</i> menekan tombol "export"
	4	Sistem akan melakukan koneksi <i>database</i> dan mengambil data pada <i>database fingerprint</i> dan meng-input-kannya ke dalam <i>database</i> aplikasi. Jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

### c. Sequence Diagram Modul Export Log Pegawai

Sequence diagram untuk *use case export log pegawai* terdapat pada gambar 3.108.

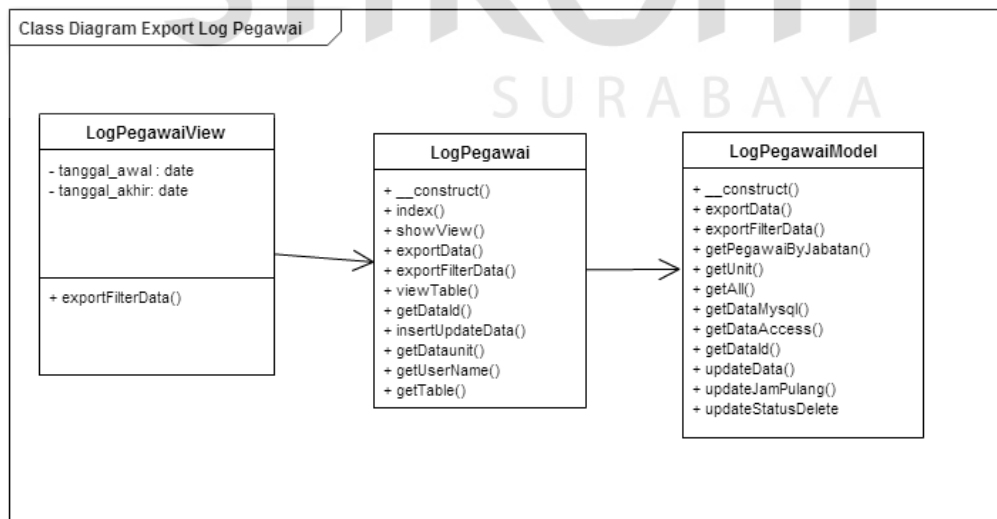




Gambar 3.108 Sequence Diagram pada modul Export Log Pegawai

**d. Class Diagram Modul Export Log Pegawai**

Relasi antar *class* pada modul export log pegawai dapat dilihat pada gambar 3.109.



Gambar 3.109 Class Diagram pada Modul Export Log Pegawai

### e. Desain Interface Modul Export Log Pegawai

Desain interface untuk modul *export log* pegawai dapat dilihat pada gambar 3.110.

Gambar 3.110 Desain User Interface pada Modul Export Log Pegawai

### f. Unit Testing Modul Export Log Pegawai

Uji coba dalam modul master ini untuk menguji apakah berbagai fungsi yang terdapat dalam modul master dapat berjalan sesuai dengan yang diharapkan. Desain uji coba modul pertemanan dapat dilihat pada tabel 3.22.

Tabel 3.22 Uji Coba pada Modul Export Log Pegawai

Test Case ID	Tujuan	Input	Output yang diharapkan	User stories
1	Menambah data	Tanggal awal, tanggal akhir, tombol export data	Data dalam <i>database</i> dapat ditambahkan sesuai dengan parameter	U17

## 6. Iterasi Verifikasi Overtime

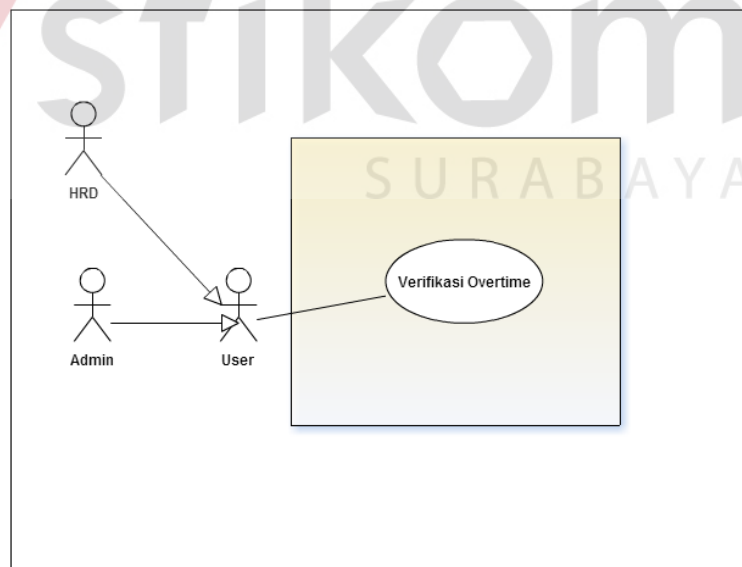
Overtime atau lembur merupakan sebuah apresiasi untuk karyawan yang bekerja lebih dari jam kerja yang biasanya diwujudkan dalam bentuk uang tambahan. Pada realitanya, tidak semua pegawai yang jam pulang jauh diatas jam kerja akan dihitung lembur. Modul ini menjembatani hal tersebut yaitu untuk melakukan verifikasi apakah seorang pegawai memang resmi diperintahkan untuk lembur atau hanya terlambat pulang atas kemauannya sendiri. Pada tabel 3.23 akan ditunjukkan hak akses dari modul ini

Tabel 3.23 Hak Akses Modul Verifikasi Overtime

	HRD	Supervisor	Manager	Accounting	Admin
Penjadwalan		√			√

### A. Diagram Use Case Modul Verifikasi Overtime

Berikut ini adalah diagram *use case* dari modul verifikasi *overtime* yang dapat dilihat pada gambar 3.111.



Gambar 3.111 Diagram Use Case pada Modul Verifikasi Overtime

## B. Flow of Events Modul Verifikasi Overtime

Flow of Events pada modul verifikasi *overtime* dapat dilihat pada tabel

3.24.

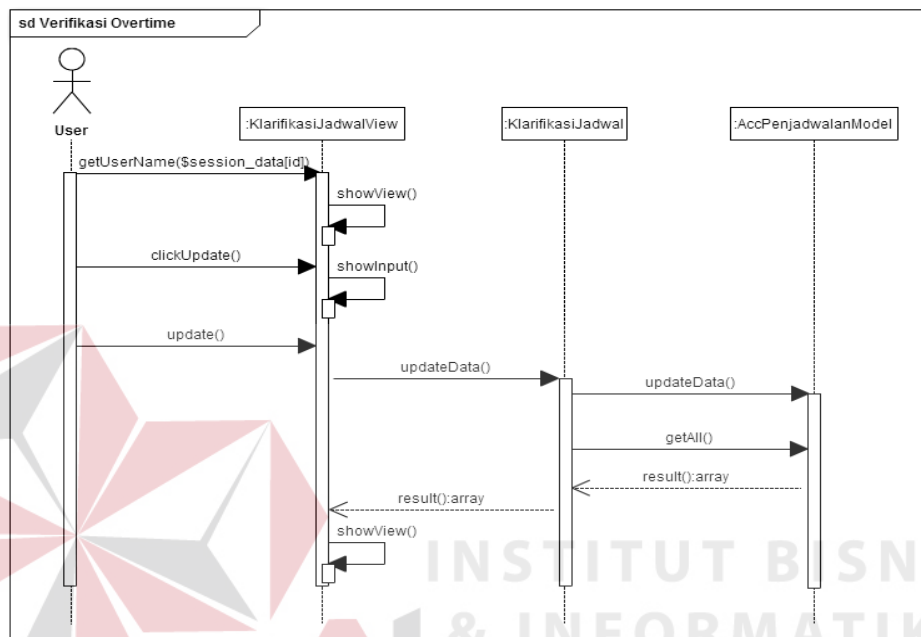
Tabel 3.24 Flow of Event pada Modul Verifikasi Overtime

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses <i>update data</i>	
<b>Tujuan</b>	<i>User</i> bisa mengisi data realisasi overtime ke <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa mengubah data.	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka data tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman Penjadwalan.	
<b>Alur Utama</b>		
	1	Use case dimulai ketika <i>user</i> berada pada form Klarifikasi Jadwal.
	2	<i>Use case</i> dimulai ketika <i>user</i> menekan tombol <input type="checkbox"/>
	3	<i>User</i> menekan mengisi form realisasi overtime lalu klik submit.
	4	Sistem akan melakukan koneksi <i>database</i> dan memasukkan data yang telah di- <i>input</i> . jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

## C. Sequence Diagram Modul Verifikasi Overtime

Sequence diagram untuk use case verifikasi overtime dapat dilihat pada gambar 3.112. Pada proses ini user melakukan verifikasi berapa jam lembur yang sebenarnya telah disetujui oleh supervisor dari bagian tersebut yang nantinya

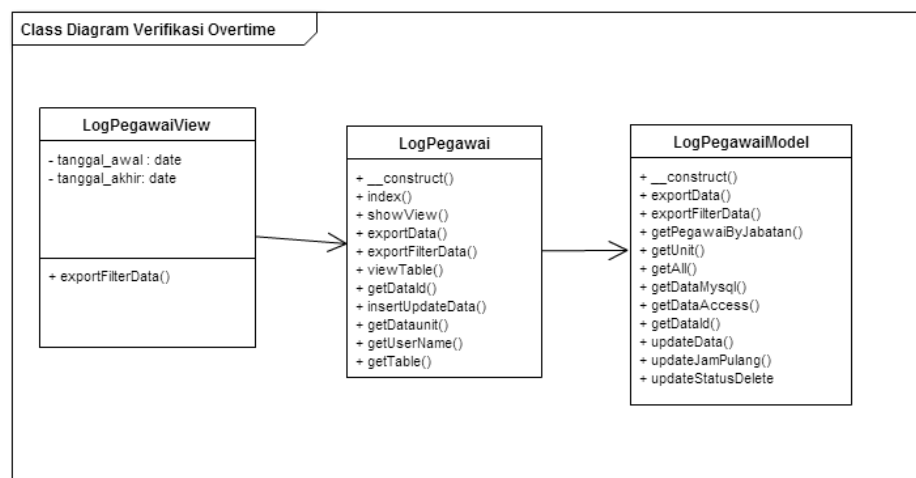
datanya akan disimpan sebagai bahan penghitungan gaji pada akhir bulan. Function `updateData()` pada class tersebut berfungsi untuk melakukan update apabila terdapat lembur karena sebelumnya, kolom lembur yang terdapat pada tabel penjadwalan di *set default null*.



Gambar 3.112 Sequence Diagram pada Modul Verifikasi Overtime

#### D. Class Diagram Modul Verifikasi Overtime

Class diagram untuk use case Verifikasi Overtime dapat dilihat pada gambar 3.113.



Gambar 3.113 Class Diagram pada Use Case Verifikasi Overtime

## E. Desain Interface Modul Verifikasi Overtime

Desain interface untuk modul verifikasi overtime dapat dilihat pada gambar 3.114.

Gambar 3.114 Desain Interface pada Modul Verifikasi Overtime

## F. Unit Testing Modul Verifikasi Overtime

Uji coba dalam modul Verifikasi Overtime ini untuk menguji apakah berbagai fungsi yang terdapat dalam modul master dapat berjalan sesuai dengan yang diharapkan. Desain uji coba modul verifikasi *overtime* dapat dilihat pada tabel 3.25.

Tabel 3.25 Uji Coba Pada Modul Verifikasi Overtime

Test Case ID	Tujuan	Input	Output yang diharapkan	User stories
1	Mengubah data	Field yang tersedia pada tiap form	Data dalam <i>database</i> dapat diubah sesuai dengan inputan.	U18

## 7. Iterasi Penghitungan Gaji

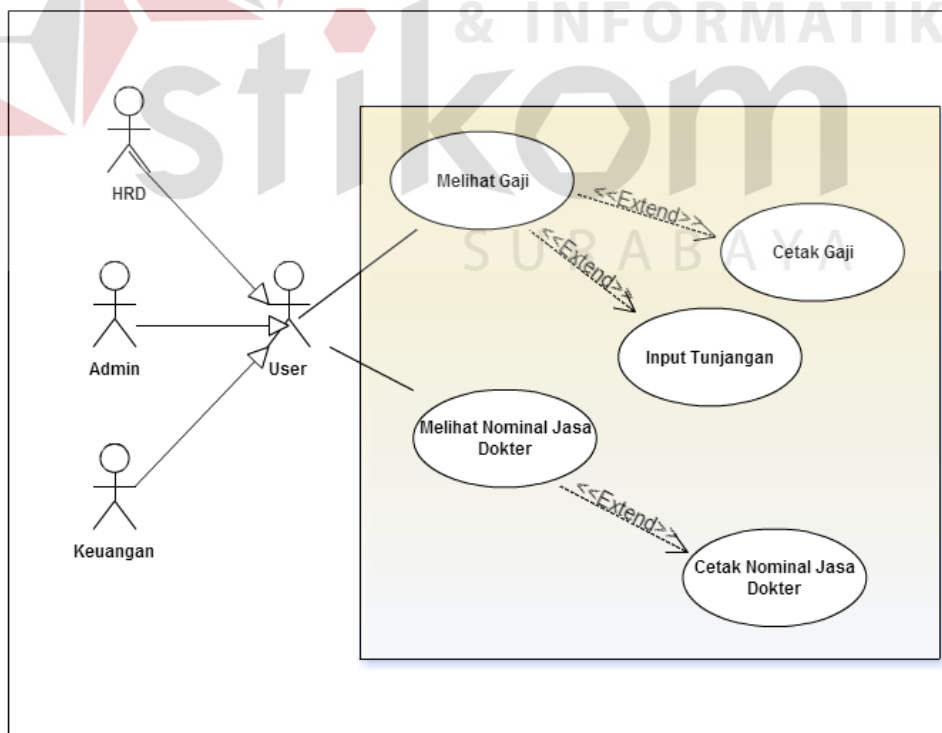
Penghitungan gaji adalah output utama dari aplikasi ini dimana untuk menentukan gaji dibutuhkan bermacam variabel yang telah disediakan untuk diinput pada fitur sebelumnya. Pada tabel 3.27 akan ditampilkan mengenai hak akses dari setiap pengguna dalam modul penghitungan gaji.

Tabel 3. 26 Hak Akses Modul Penghitungan Gaji

	HRD	Supervisor	Manager	Accounting	Admin
Penghitungan Gaji	√				√
View Nominal Jasa Medis				√	√

### A. Diagram Use Case Modul Penghitungan Gaji

Berikut ini adalah diagram *use case* dari modul master yang dapat dilihat pada gambar 3.115.



Gambar 3.115 Diagram Use Case pada Modul Penghitungan Gaji

## B. Flow of Events Modul Penghitungan Gaji

Terdapat 3 proses dalam modul penghitungan gaji yaitu input tunjangan, hitung gaji dan cetak report. Modul ini juga mencakup proses melihat nominal remunerasi medis serta mencetak nominal remunerasi medis. Berikut ini adalah *flow of event* pada modul penghitungan gaji.

### a. Flow of Event Insert Tunjangan

*Flow of event* untuk *insert* dapat dilihat pada tabel 3.28.

Tabel 3.27 Flow of Event Input Tunjangan

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan proses input Tunjangan	
<b>Tujuan</b>	<i>User</i> bisa menambahkan data ke <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	<i>User</i> bisa menambahkan data.	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka data tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman Laporan Gaji	
<b>Alur Utama</b>		
	1	<i>Use case</i> dimulai ketika <i>user</i> berada pada form Laporan Gaji.
	2	<i>Use case</i> dimulai ketika <i>user</i> mencari tanggal gaji dan unit gaji.
	3	<i>User</i> mengisi nilai tunjangan pada tabel yang telah tersedia
	4	Sistem akan melakukan koneksi <i>database</i> dan memasukkan data yang telah di- <i>input</i> . jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.



b. Flow of Event Hitung Gaji pada Modul Penghitungan Gaji

*Flow of event* untuk hitung gaji pada modul penghitungan gaji dapat dilihat pada tabel 3.29.

Tabel 3.28 Flow of Event Hitung Gaji

Deskripsi Use case Detail	Keterangan	
<b>Deskripsi</b>	<i>Use case</i> untuk melakukan penghitungan gaji	
<b>Tujuan</b>	<i>User</i> bisa menghitung dan menambahkan hasilnya sebagai data ke <i>database</i> melalui aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	Gaji bisa dihitung secara otomatis dari sistem.	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka data tidak akan ditampilkan di <i>View</i>	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman Laporan Gaji	
<b>Alur Utama</b>		
	1	<i>Use case</i> dimulai ketika <i>user</i> berada pada form Laporan Gaji.
	2	<i>Use case</i> dimulai ketika <i>user</i> mencari tanggal gaji dan unit gaji.
	3	<i>User</i> menekan tombol "Generate Gaji"
	4	Sistem akan melakukan koneksi <i>database</i> dan menghitung gaji lalu memasukkan data yang telah dihitung pada <i>database</i> . Jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

c. Flow of Event Cetak Report pada Modul Penghitungan Gaji

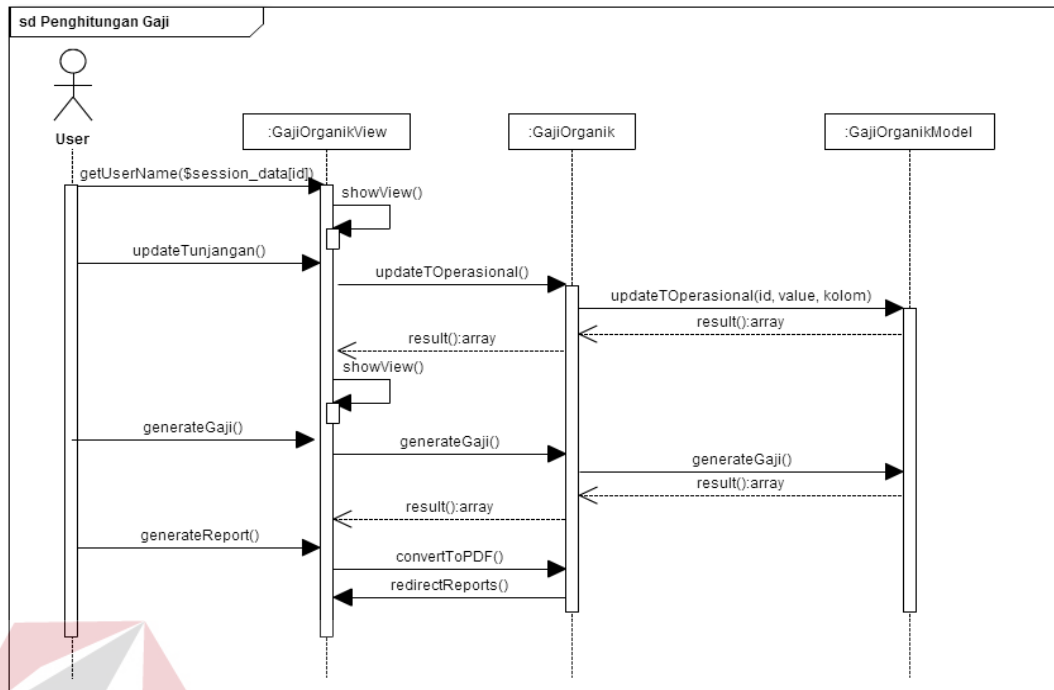
*Flow of event* cetak report pada modul penghitungan gaji dapat dilihat pada tabel 3.30.

Tabel 3. 29 Flow of Event Cetak Report

<b>Deskripsi Use case Detail</b>	<b>Keterangan</b>	
<b>Deskripsi</b>	<i>Use case</i> untuk mencetak <i>report</i>	
<b>Tujuan</b>	<i>User</i> bisa membuat laporan slip gaji yang otomatis ter-generate dari aplikasi	
<b>Prasyarat</b>	Login	
<b>Kondisi Akhir Sukses</b>	Muncul tampilan report dengan data yang benar dalam bentuk PDF.	
<b>Kondisi akhir gagal</b>	Jika terjadi kesalahan dalam <i>query</i> dan koneksi <i>database</i> maka <i>report</i> tidak akan ditampilkan	
<b>Aktor utama</b>	<i>User</i>	
<b>Aktor sekunder</b>	Tidak ada	
<b>Pemicu</b>	<i>User</i> berada pada halaman Laporan Gaji atau <i>user</i> berada pada halaman laporan jasa dokter	
<b>Alur Utama</b>		
	1	<i>Use case</i> dimulai ketika <i>user</i> berada pada form Laporan Gaji atau form jasa dokter.
	2	<i>Use case</i> dimulai ketika <i>user</i> mencari tanggal gaji dan unit gaji (Gaji Organik). <i>Use case</i> dimulai ketika <i>user</i> memilih tanggal (Jasa Dokter)
	3	<i>User</i> menekan tombol "Generate Report"
	4	Sistem akan melakukan koneksi <i>database</i> , mengambil data yang dibutuhkan lalu menampilkan dalam bentuk report. Jika terjadi kesalahan koneksi pada <i>database</i> , maka akan dilakukan langkah 4.1
<b>Alur Alternatif</b>	<b>Langkah</b>	<b>Aksi</b>
	-	-
<b>Alur Error</b>	<b>Langkah</b>	<b>Aksi</b>
	4.1	Jika terjadi kesalahan dalam koneksi <i>database</i> maka alur proses akan kembali pada langkah 1.

### C. Sequence Diagram Modul Penghitungan Gaji

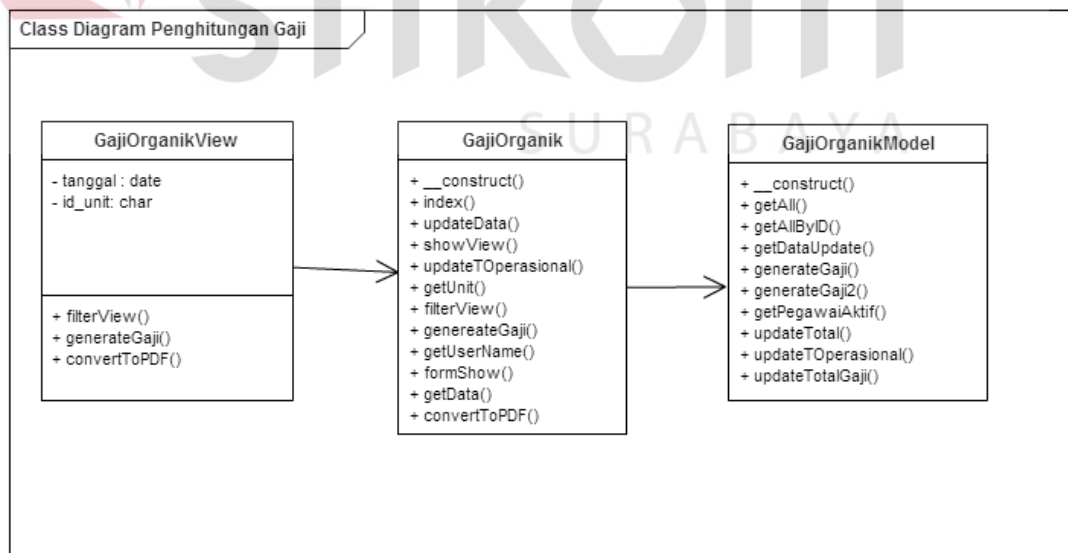
*Sequence diagram* pada modul penghitungan gaji dapat dilihat pada gambar 3.116.



Gambar 3.116 Sequence Diagram pada Modul Penghitungan Gaji

#### D. Class Diagram Modul Penghitungan Gaji

Relasi antar *class* pada modul penghitungan gaji dapat dilihat pada gambar 3.117.



Gambar 3.117 Class Diagram pada modul Penghitungan Gaji

## E. Desain Interface Modul Penghitungan Gaji

Desain *user interface* pada modul penghitungan gaji dapat dilihat pada gambar 3.118

Gambar 3.118 Desain User Interface pada Modul Penghitungan Gaji

## F. Unit Testing Modul Penghitungan Gaji

Uji coba dalam modul master ini untuk menguji apakah berbagai fungsi yang terdapat dalam modul penghitungan gaji dapat berjalan sesuai dengan yang diharapkan. Desain uji coba modul penghitungan gaji dapat dilihat pada tabel 3.31.

Tabel 3.30 Uji Coba Modul Penghitungan Gaji

Test Case ID	Tujuan	Input	Output yang diharapkan	User stories
1	Menghitung Gaji	Field yang tersedia pada tiap form	Data dalam <i>database</i> dapat ditambahkan sesuai dengan inputan.	U19, U20, U21, U22, U23