

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Dalam pengembangan sistem ini menggunakan konsep *Systems Development Life Cycle* (SDLC) yang berfungsi untuk menggambarkan urutan alur langkah-langkah dari setiap tahapan. Langkah-langkah yang akan dilakukan dalam pembuatan sistem informasi kepegawaian yaitu: identifikasi masalah, analisis sistem, perancangan sistem, implementasi sistem, perawatan dan pengembangan sistem.

3.1 Analisis Masalah

3.1.1 Identifikasi Masalah

Pada langkah identifikasi masalah dilakukan tahapan-tahapan untuk mengetahui permasalahan yang ada yaitu dengan melakukan pengamatan kegiatan pada perusahaan, wawancara langsung dengan pihak terkait khususnya bagian *Human Resource Departmen* (HRD) di PKIS Sekar Tanjung, mencatat kegiatan saat ini, serta mempelajari sistem yang berjalan di PKIS Sekar Tanjung. sehingga bisa dilakukan tindakan solusi yang tepat untuk menyelesaikan permasalahan tersebut.

Hasil observasi dan wawancara dengan Bapak Wanto selaku bagian HRD di PKIS Sekar Tanjung, menceritakan bahwa bagian HRD selama tidak memiliki sistem yang baku untuk mengolah data-data pegawai yang ada. Semua dilakukan

secara manual sehingga sangat menyita banyak waktu untuk pembuatan laporan yang dibutuhkan.

HRD di PKIS Sekar Tanjung merupakan bagian yang memiliki peran penting pada perusahaan karena bagian HRD merupakan sebuah bagian yang mengelola seluruh data tentang pegawai mulai dari masuk dalam perusahaan sampai pegawai tersebut keluar dari perusahaan. Untuk menjalankan perannya tersebut, bagian HRD melakukan beberapa aktivitas, antara lain: menangani pelanggaran, menangani kontrak kerja, menangani mutasi atau perpindahan posisi jabatan dan bagian, menangani *training* kerja, dan memantau kehadiran pegawai, serta memantau prestasi pegawai dengan melakukan penilaian kinerja.

Saat ini, bagian HRD dalam melakukan pencatatan data dan kegiatan pegawai masih dilakukan manual dengan menggunakan *Microsoft Excel*, data-data pegawai tersebut kebanyakan hanya diarsipkan kedalam file dan dokumen. Penggunaan *Microsoft excel* dalam pengolahan data pegawai di PKIS Sekar Tanjung masih kurang membantu, khususnya dalam penyajian informasi kepegawaian karena *Microsoft excel* tidak bisa menampilkan histori data pegawai secara lengkap untuk masing-masing pegawai seperti informasi habis masa kontrak kerja, histori pelanggaran yang dilakukan, *training* yang telah diikuti, perpindahan jabatan atau bagian (mutasi), serta kehadiran saat bekerja untuk setiap pegawai. Hal ini dikarenakan banyaknya jumlah data pegawai yang ada, serta masih terpisah-pisahannya antara satu data dengan data lainnya, seperti data pegawai, data pelanggaran, data mutasi, data *training*, data kontrak kerja. Maka dari itu manajemen membutuhkan sebuah sistem yang mampu menyimpan dan

mengolah data pegawai menjadi sebuah laporan yang bermanfaat bagi perusahaan.

Dengan adanya sistem yang dibangun untuk mendukung kerja HRD pada PKIS Sekar Tanjung, maka akan membantu kerja HRD dalam menyajikan informasi, pencarian data maupun penyusunan laporan. Data-data yang dibutuhkan dalam sistem ini terdiri dari data kontrak kerja, data pindah jabatan, data pelanggaran, data pegawai, data *training*, dan data absensi. Data-data tersebut nantinya akan dijadikan sebagai sebuah informasi yang berguna bagi manajemen. Laporan-laporan yang dibutuhkan yaitu: laporan habis masa kontrak kerja, laporan pelanggaran yang telah dilakukan, laporan *training* yang telah diikuti, laporan penilaian kinerja, serta laporan kehadiran saat bekerja untuk setiap pegawai.

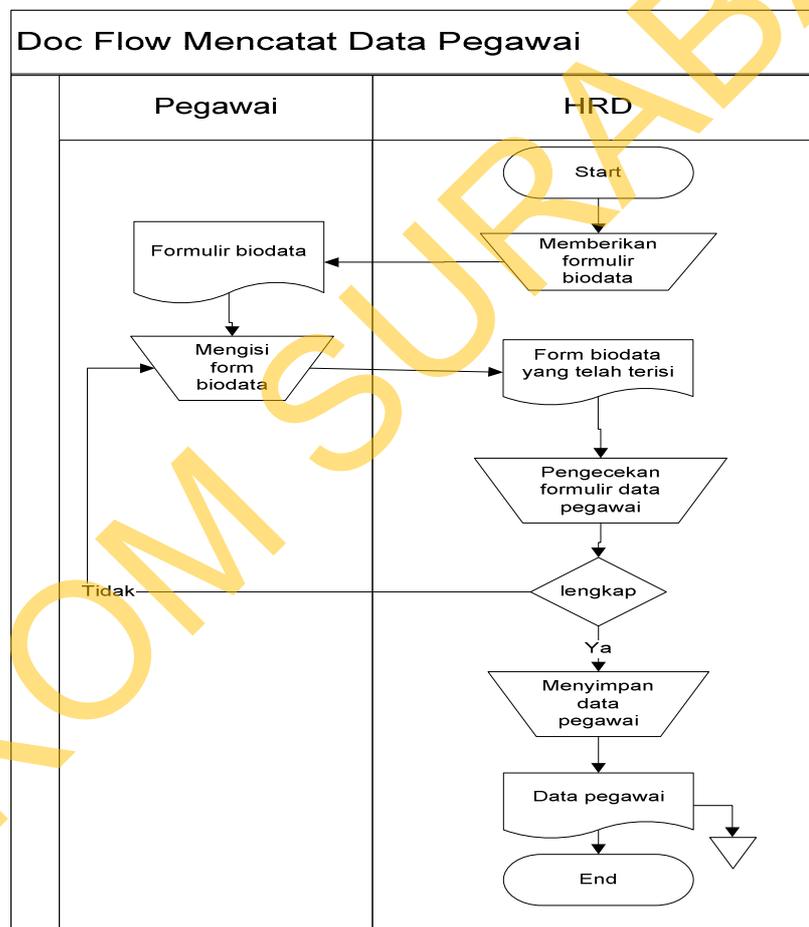
Berdasarkan permasalahan di atas, bagian HRD sangat membutuhkan aplikasi yang mampu mendukung dalam pengembangan kepegawaian di PKIS Sekar Tanjung dengan mengaplikasikan sistem informasi kepegawaian sebagai pendukung kegiatan di PKIS Sekar Tanjung.

3.1.2 Dokumen *Flow* Sistem Yang Berjalan

Berikut ini adalah gambaran proses bisnis dari identifikasi yang sedang berjalan pada PKIS Sekar Tanjung, yang di gambarkan dalam dokumen *flow* dibawah ini.

1. Mencatat Pegawai

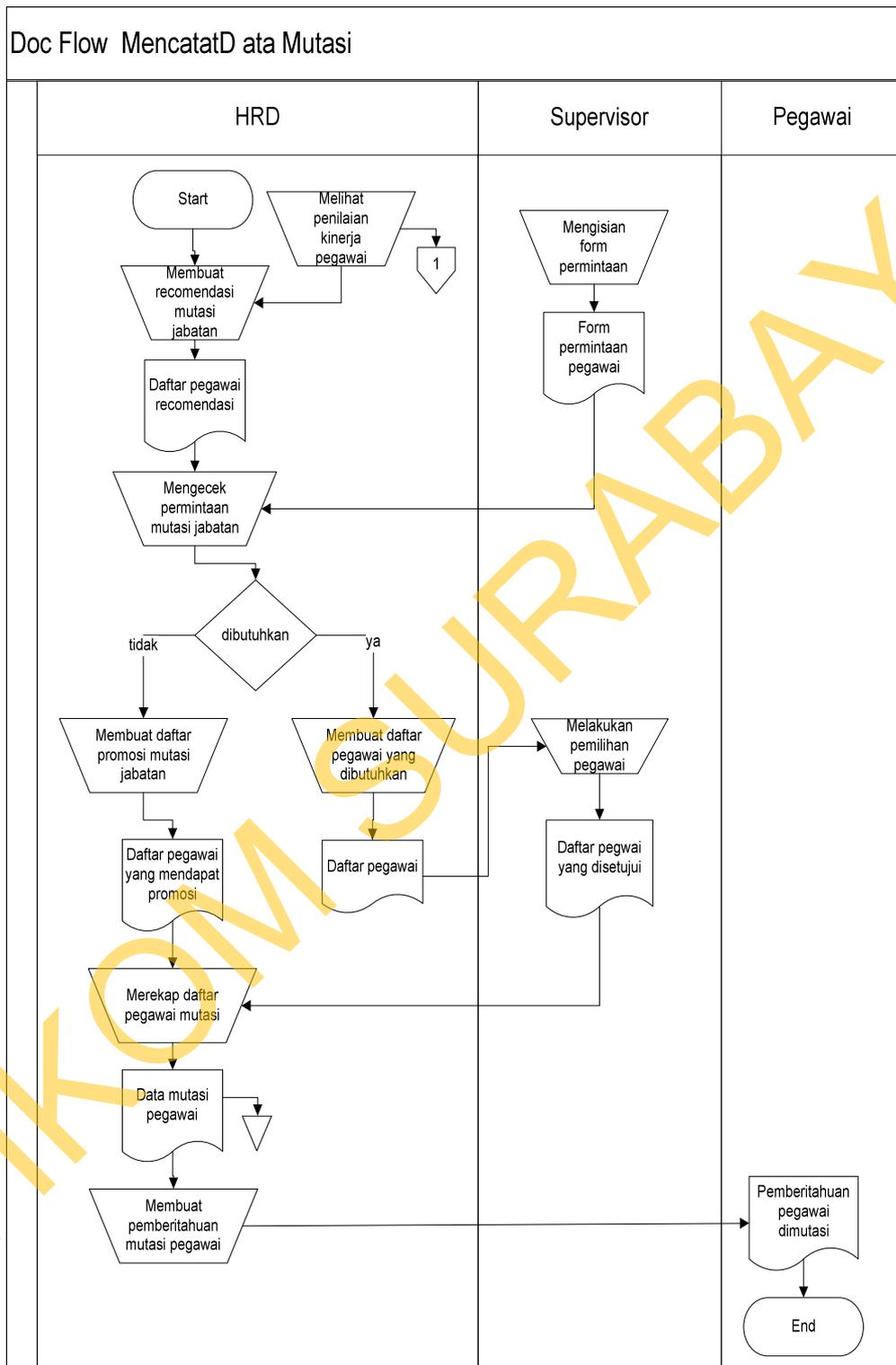
Pada gambar 3.1 merupakan proses untuk mencatat pegawai yang sedang berjalan, menjelaskan bahwa HRD menyerahkan formulir data pegawai kosong kepada pegawai, kemudian pegawai mengisi data diri pegawai, setelah di isi kemudian dikembalikan kembali kepada HRD, selanjutnya HRD mengecek kelengkapan data kemudian data tersebut disimpan dan diarsipkan.



Gambar 3.1 Diagram Alir Mencatat Data Pegawai.

2. Mencatat Mutasi Jabatan

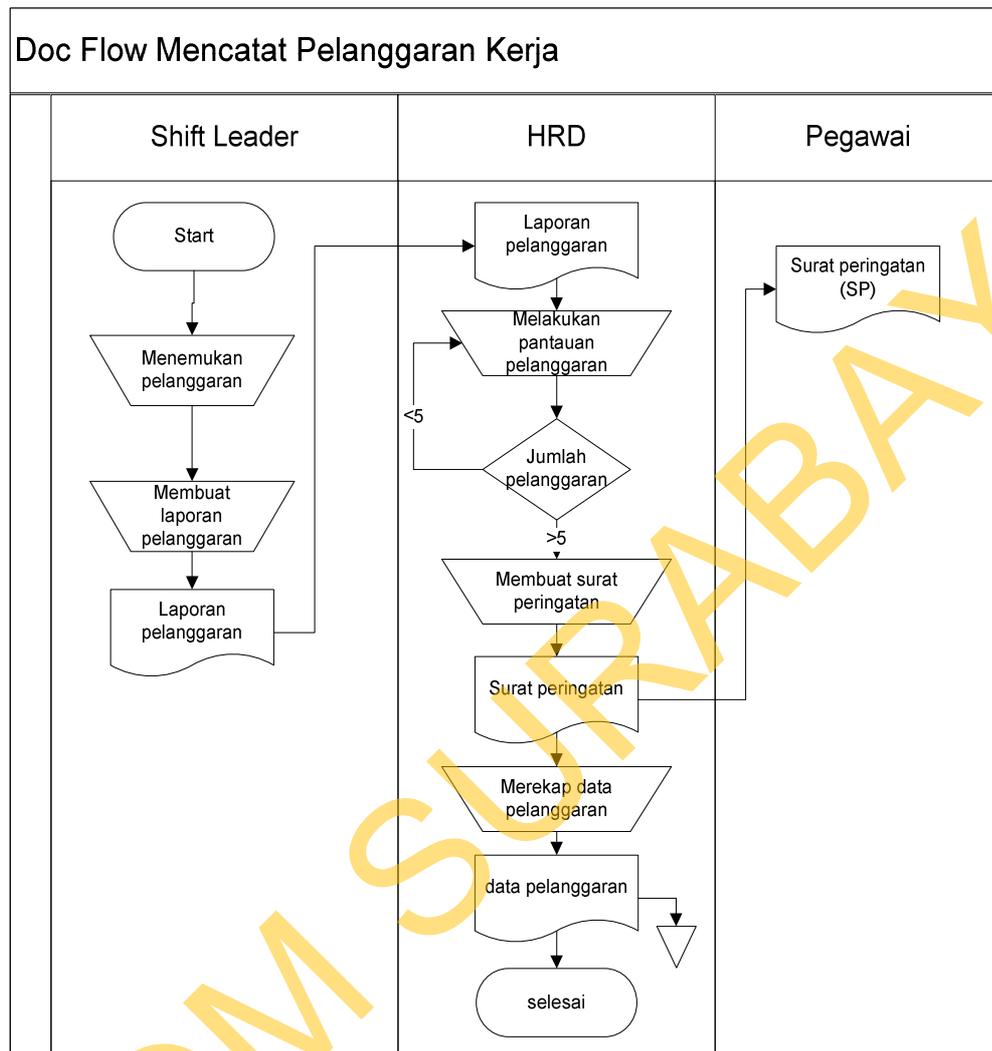
Merupakan proses untuk mencatat mutasi jabatan yang dapat dilihat pada gambar 3.2 yang menjelaskan mengenai proses pencatatan manual histori mutasi kemana saja pernah dipindahkan selama bekerja di PKIS Sekar Tanjung. Pada *document flow* mencatat mutasi jabatan, HRD membuat *recomendasi* pegawai yang mendapatkan mutasi. Setelah membuat rekomendasi HRD mengecek apakah ada bagian yang membutuhkan pegawai. Setelah itu HRD memberikan daftar pegawai kepada *supervisor* bagian. Selanjutnya HRD membuat surat pemberitahuan mutasi untuk pegawai yang mendapatkan mutasi jabatan.



Gambar 3.2 Diagram Alir Mutasi

3. Mencatat Pelanggaran

Gambar 3.3 merupakan proses untuk mencatat pelanggaran terhadap tata tertib yang sedang berjalan di PKIS Sekar Tanjung. Pada *document flow* pelanggaran, dimulai *supervisor/shift leader* bagian membuat laporan pelanggaran pegawai, kemudian laporan pelanggaran tersebut diberikan HRD untuk dibuat surat peringatan. Surat peringatan itu diberikan kepada pegawai setelah pegawai melakukan sebanyak lima (5) kali kesalahan secara berterus-menerus. Setelah pegawai mendapatkan surat peringatan selanjutnya HRD merekap seluruh laporan pelanggaran.

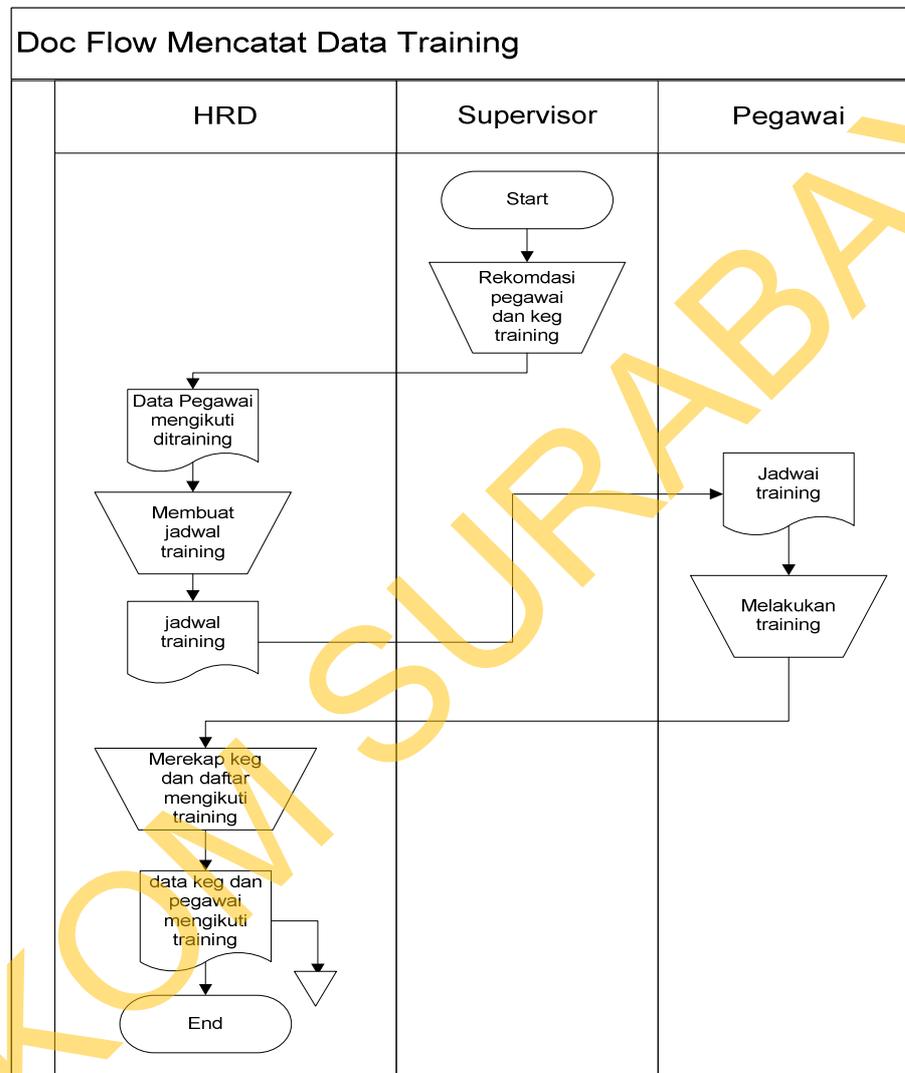


Gambar 3.3 Diagram Alir Mencatat Pelanggaran

4. Mencatat *Training* Kerja

Merupakan proses untuk mencatat *training* kerja yang pernah diikuti pegawai dapat dilihat pada gambar 3.4. Histori *training* ini digunakan untuk mengetahui *training* apa saja yang telah dilakukan agar dapat melatih disiplin kerja dan meningkatkan kinerja pegawai. Pada *document flow* mencatat data *training*, *supervisor* memberikan daftar rekomendasi pegawai yang mengikuti *training*,

kemudian HRD membuat jadwal. Setelah pegawai melakukan *training* HRD mencatat *training* tersebut untuk menjadi histori rekapan.

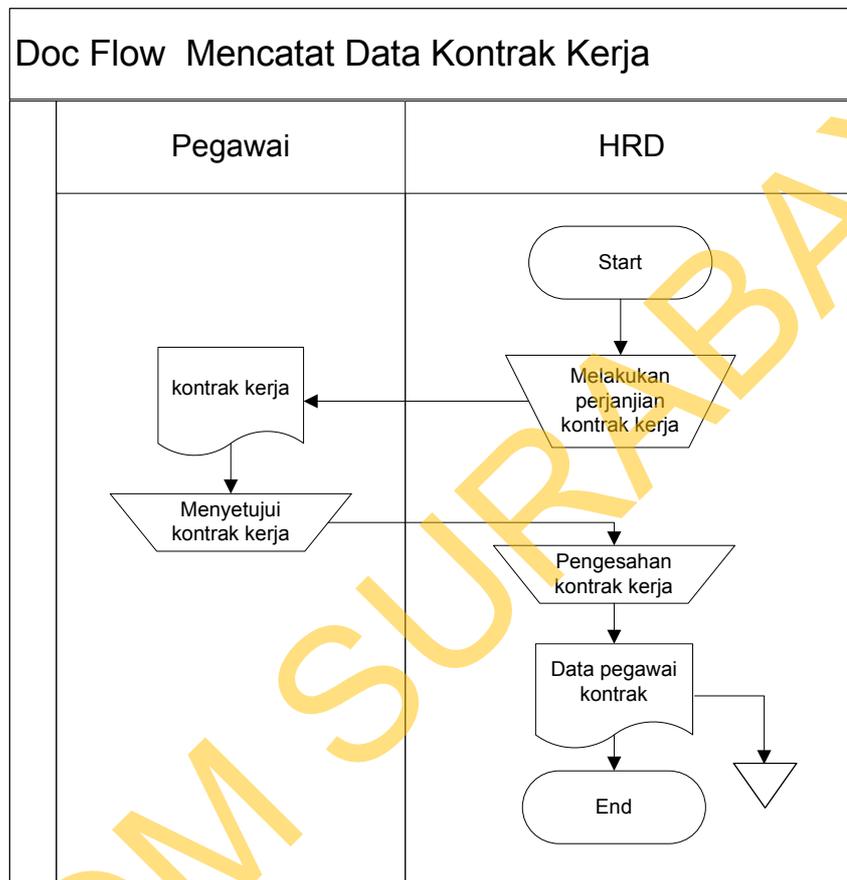


Gambar 3.4 Diagram Alir *Training* Kerja

5. Mencatat Kontrak Kerja

Merupakan proses untuk mencatat kontrak kerja untuk pegawai lama yang dapat dilihat pada gambar 3.5. HRD dan pegawai baru melakukan kontrak kerja, HRD memberikan pernyataan persetujuan kontrak kerja kepada pegawai. Jika

pegawai tersebut menyetujui semua yang ada pada kontrak kerja maka bagian HRD akan mencatat kontrak sebagai histori.

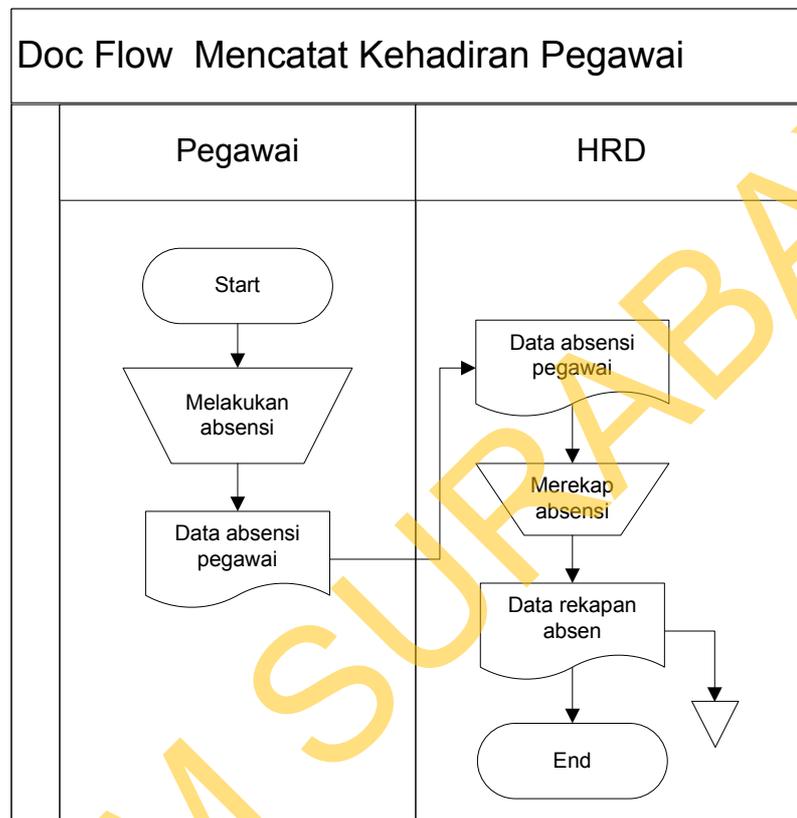


Gambar 3.5 Diagram Alir Kontrak Kerja

6. Proses Absensi

Merupakan proses untuk mencatat kehadiran pegawai saat akan masuk kerja di PKIS Sekar Tanjung dapat dilihat pada gambar 3.6. Selama ini proses absensi yang dilakukan di PKIS Sekar Tanjung masih sederhana, yaitu menggunakan mesin absensi *check clock*. Aktivitas pegawai dimulai dengan

melakukan *check clock*, kemudian setelah pegawai *check clock* pada mesin, HRD mengambil kartu absensi tersebut dan merekap kehadiran pegawai.

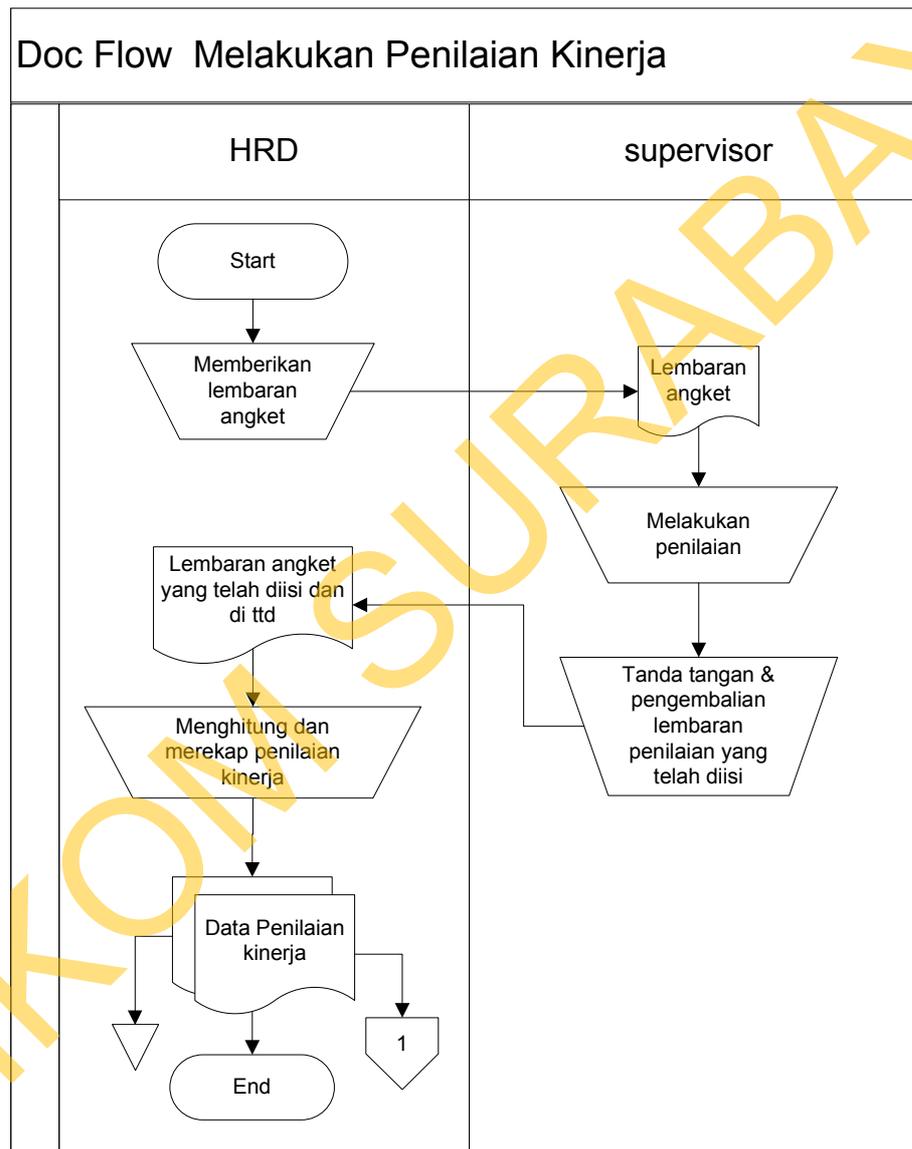


Gambar 3.6 Diagram Alir Absensi

7. Proses Penilaian Kinerja

Merupakan proses untuk menilai kinerja pegawai yang dapat dilihat pada gambar 3.7. *Supervisor* pada masing-masing bagian melakukan penilaian kinerja pegawai. Penilaian kinerja pegawai ini dilakukan *supervisor* setiap 1 bulan sekali. Langkah awal dalam melakukan penilaian kinerja adalah *supervisor* masing-masing bagian mendapatkan *form* penilaian pegawai dari HRD yang selanjutnya diisi oleh *shift leader*. Setelah *form* itu selesai diisi dapat langsung dikembalikan

pada HRD yang selanjutnya diinputkan ke dalam *Microsoft excel* untuk dihitung. Terdapat sembilan (9) aspek penilaian yang dapat dinilai oleh supervisor, 9 aspek penilaian tersebut dapat dilihat pada lampiran 2.



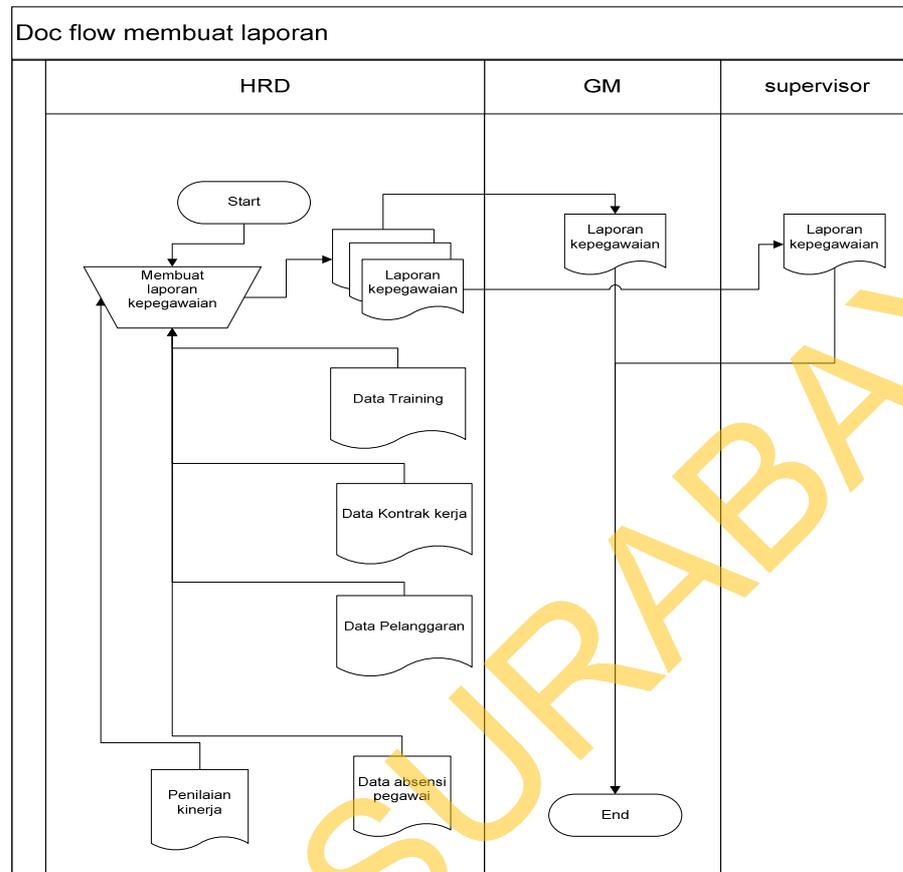
Gambar 3.7 Diagram Alir Penilaian Kinerja

Setelah dilakukan penilaian oleh *shift leader* bagian selanjutnya adalah pemberian nilai untuk karyawan. Karyawan bisa mendapatkan nilai A atau nilai B

tergantung penilaian yang telah dilakukan oleh *shift leader*. Pegawai mendapatkan nilai “A” jika jumlah poin yang bersifat negatif kurang dari tiga (3), sedangkan nilai “B” jika jumlah poin yang bersifat negatif lebih dari sama dengan 3. Untuk penjelasan nilai yang bersifat negatif dan positif dapat dilihat pada bab landasan teori.

8. Membuat Laporan

Untuk proses pembuatan laporan, bagian HRD mengumpulkan data-data yang dibutuhkan secara manual dengan melihat data-data pegawai pada *file-file* berkas yang ada, kemudian bagian HRD membuat laporan secara manual kedalam komputer yang hasilnya diberikan kepada GM maupun. Laporan-laporan yang sering dibutuhkan yaitu laporan pegawai, laporan pelanggaran, laporan training, laporan absensi, laporan penilaian kinerja. Dapat dilihat pada gambar 3.8 doc flow membuat laporan.



Gambar 3.8 Diagram Alir Prosedur Pembuatan Laporan

3.1.3 Hasil Analisis Sistem

Dari identifikasi masalah di atas, didapatkan kelemahan-kelemahan sistem yang lama dan untuk memperbaiki kelemahan tersebut sistem yang dibuat disesuaikan dengan kebutuhan pihak manajemen. Hasil identifikasi masalah yang didapat sebagai berikut:

1. Kelemahan dari sistem yang lama antara lain:
 - a. Data-data kepegawaian yang tersimpan dalam berkas-berkas *file* belum dapat menjadi informasi yang dapat mendukung manajemen dalam mengambil keputusan.

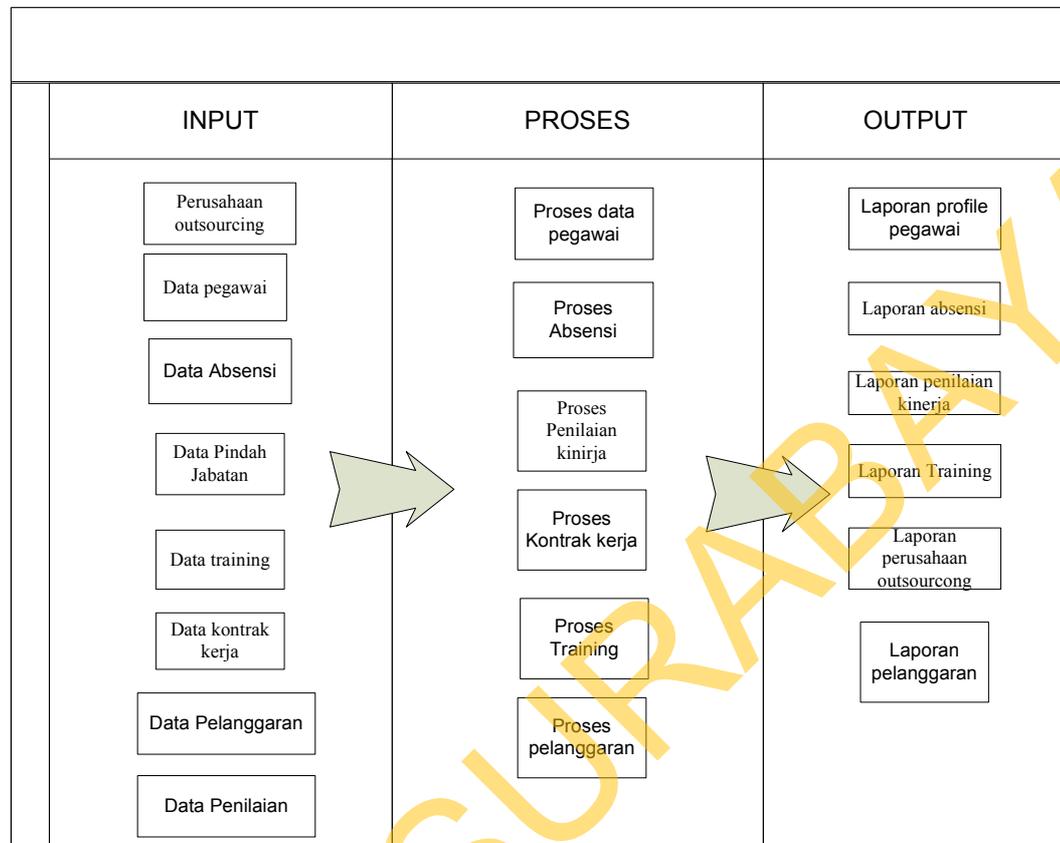
b. Masih memungkinkan terjadinya kesalahan saat pencatatan data pegawai dan pembuatan laporan dikarenakan *database* dan aplikasi belum mendukung.

2. Kebutuhan informasi pemakai atau manajemen

1. Informasi untuk mendukung manajemen dalam pengambilan keputusan seperti informasi pelanggaran, informasi *training* yang diikuti, informasi *outsourcing*, informasi absensi, dan informasi penilaian kinerja.
2. Membutuhkan sebuah aplikasi yang dapat mendukung informasi agar kegiatan kepegawaian dapat berjalan sebagai mana mestinya.

3.2 Perancangan Sistem

Perancangan sistem akan memberikan gambaran tentang langkah-langkah dalam membangun sistem yang akan dibuat. Gambaran umum mengenai sistem yang akan dibangun dapat dilihat pada gambar 3.9. Blok diagram tersebut menjelaskan tentang apa saja yang menjadi *input*, proses dan *output* yang akan diperlukan sistem.



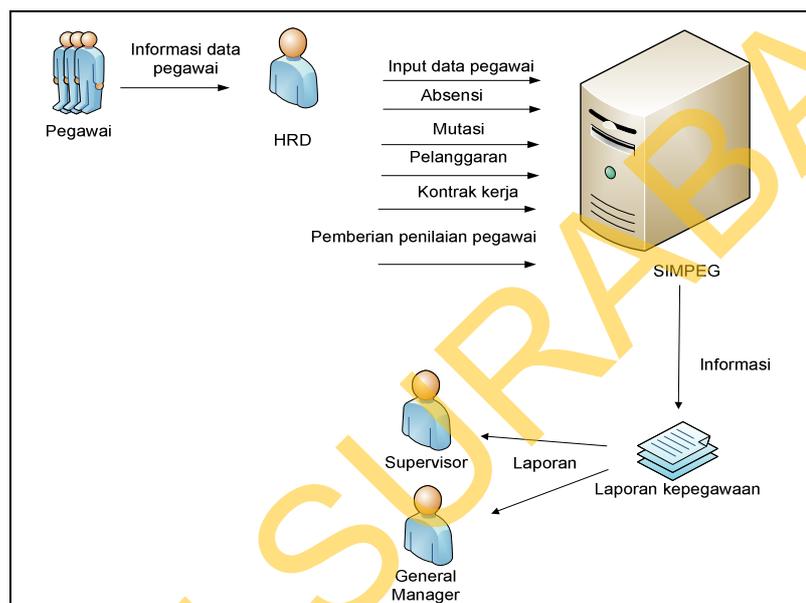
Gambar 3.9 Blok Diagram Sistem Informasi Kepegawaian

Pada gambar 3.9 menjelaskan bahwa masukan yang diperlukan oleh sistem antara lain adalah data absensi, data pelanggaran, data mutasi, data kontrak kerja, data penilaian dan data pegawai. Data-data tersebut selanjutnya akan diproses pada sistem yang nanti keluarannya yang didapat berupa: laporan pegawai, lama masa kontrak kerja, pelanggaran yang dilakukan, absensi, penilaian kinerja.

3.2.1 Model Pengembangan

Model pengembangan yang disarankan dalam tugas akhir ini yaitu berupa rancang bangun dengan mengumpulkan data mutasi, data pelanggaran, data

kontrak, data *training*, data absensi dan penilaian kinerja sebagai *input* dalam rancang bangun ini. Selanjutnya data tersebut akan diproses dan dianalisa sesuai dengan informasi yang dihasilkan. Secara garis besar dapat digambarkan dengan menggunakan arsitektur seperti pada gambar 3.10.

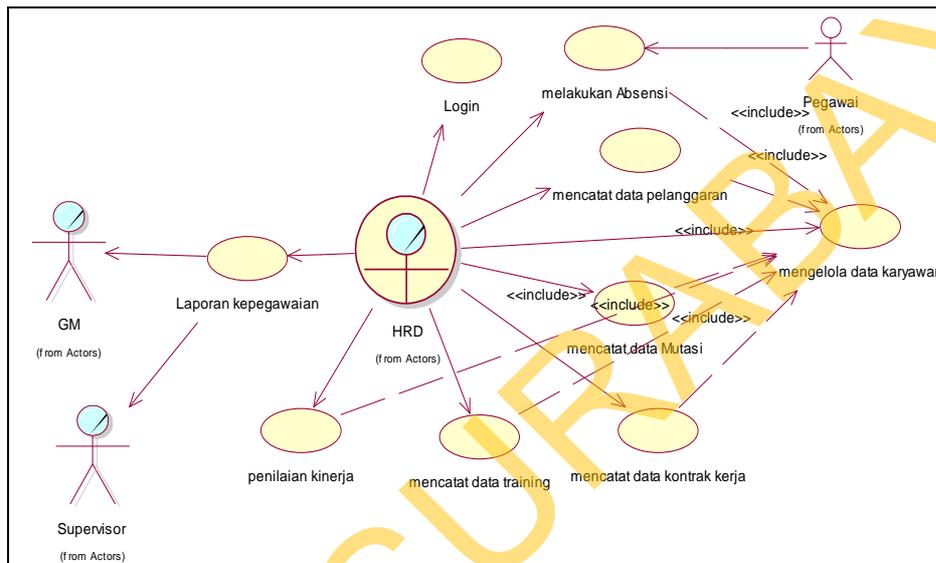


Gambar 3.10 Arsitektur Sistem Informasi Kepegawaian

3.2.2 Use case Diagram Sistem Informasi Kepegawaian

Diagram *use case* memperlihatkan hubungan-hubungan yang terjadi antara aktor-aktor dengan *use case* dalam sistem. Pada *use case* diagram 3.11 terdapat tiga (3) pekerja bisnis untuk Sistem Informasi Kepegawaian. Pekerja bisnis atau *business worker* adalah peranan di dalam organisasi, bukan posisi. Seseorang bisa memainkan banyak peran tetapi hanya memegang satu posisi, sedangkan aktor bisnis atau *business actor* adalah seseorang atau sesuatu yang ada di luar

organisasi dan berinteraksi dengan organisasi yang terlibat dalam kegiatan bisnis organisasi (Sholih, 2006).



Gambar 3.11 Use case Diagram Sistem informasi kepegawaian

Pada gambar 3.11 terdapat tiga pekerja bisnis yaitu HRD, Supervisor dan *General Manager* (GM) sedangkan pegawai berada di luar organisasi sehingga disebut sebagai aktor bisnis. Menerima seluruh laporan kepegawaian yang diolah oleh HRD diperankan oleh GM dan *supervisor* bagian, selain itu *supervisor* bagian berperan dalam menilai pegawai yang dibawahinya. Sedangkan HRD berperan dalam menangani kepegawaian data pegawai di dalam sistem, data yang dapat diolah yaitu: data *training*, data kontrak kerja, data pelanggaran, data mutasi, absensi dan data penilaian kinerja.

3.2.3 Flow Of Events

Flow of events bertujuan untuk mendokumentasikan alur logika dalam *use case* yang menjelaskan secara lengkap apa yang akan pemakai lakukan dan apa yang sistem itu sendiri lakukan (Sholiq, 2010). *Flow of events login* disajikan pada tabel 3.1 dengan kondisi akhir yang diinginkan atau kondisi akhir gagal, serta alur alternatif untuk menangani kondisi salah.

1. Flow of events Use Case Login

Langkah awal dalam kegiatan ini adalah proses *login*. Proses *login* dilakukan untuk memulai seluruh aktivitas aplikasi kepegawaian pada PKIS Sekar Tanjung. *Flow of events* dari *use case login* disajikan dalam tabel 3.1.

Tabel 3.1 *Flow Of Event Login*

Nama Use Case	Login
Kebutuhan terkait	Salah satu proses untuk memulai sistem informasi kepegawaian adalah <i>login</i> pada sistem yang dilakukan oleh HRD. HRD memasukkan <i>username</i> dan <i>password</i> yang harus diverifikasi oleh sistem agar bagian HRD dapat melakukan proses bisnis pada sistem kepegawaian.
Tujuan	Menangani proses <i>login</i> yang dilakukan oleh bagian HRD.
Prasyarat	Tidak ada.
Kondisi akhir sukses	Pengguna yang berhasil <i>login</i> dapat mengakses menu-menu yang telah disediakan menurut peran pengguna tersebut.
Kondisi akhir gagal	Apabila <i>login</i> yang dilakukan oleh pengguna gagal, maka pengguna harus menginputkan <i>username</i> dan <i>password</i> yang mereka miliki dengan benar.

Nama Use Case	<i>Login</i>	
Aktor utama	HRD.	
Aktor sekunder	Tidak ada.	
Pemicu	Aktor memilih menu “ <i>login</i> ” untuk dapat mengakses seluruh menu pada sistem.	
Alur utama	Langkah	Aksi
	1.	Membuka halaman utama <i>website</i> sistem informasi kepegawaian.
	2.	Menampilkan halaman <i>login</i> dengan inputan yang diperlukan adalah <i>username</i> dan <i>password</i> .
	3.	Mengisi <i>username</i> dan <i>password</i> kemudian menekan tombol <i>login</i> .
	4.	Melakukan koneksi ke <i>database</i> .
	5.	Melakukan validasi inputan <i>login</i> .
	6.	Mengambil <i>password</i> berdasarkan <i>username</i> yang telah diinputkan ke <i>database</i>
	7.	Menampilkan pesan <i>login</i> sukses/berhasil.
	8.	Mengarahkan pada <i>web</i> halaman utama
9.	<i>Use case</i> berakhir.	
Alur Perluasan	Langkah	Aksi Percabangan
	5.1.	Menampilkan pesan <i>login</i> gagal, kemudian aktor kembali ke langkah 3.

2. Flow Of Events Use Case Data Pegawai

Berikut ini disajikan *flow of events* dari *use case* diagram data pegawai. Dimulai dari pegawai memberikan *form* biodata yg telah lengkap kepada HRD, kemudian bagian HRD memasukkan biodata tersebut ke dalam sistem. *Flow of events* dari *use case* data pegawai disajikan dalam tabel 3.2.

Tabel 3.2 *Flow Of Events* Data Pegawai

Nama Use Case	Mencatat Data Pegawai	
Kebutuhan terkait	Pegawai mengisi <i>form</i> biodata pegawai terlebih dahulu kemudian HRD memasukkan data pegawai tersebut kedalam sistem.	
Tujuan	Menangani data pegawai.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	HRD yang benar mengisi data pegawai dapat disimpan dan dapat memasukkan data pegawai lainnya kedalam tabel pegawai.	
Kondisi akhir gagal	<ul style="list-style-type: none"> • Jika kegagalan ada pada: <i>database</i>, maka <i>input</i> data pegawai tidak bisa dilakukan. • Jika kegagalan ada pada saat <i>input</i> data maka data pegawai tidak bisa disimpan. 	
Aktor utama	HRD.	
Aktor sekunder	Tidak ada.	
Pemicu	Aktor HRD memilih menu “Master Pegawai” untuk memasukkan data pegawai baru.	
Alur utama	Langkah	Aksi
	1.	Memilih menu master pegawai.
	2.	Mengarahkan aktor ke halaman master pegawai.
	3.	Aktor mengisikan data-data yang diperlukan untuk pegawai baru seperti nama, alamat, TTL, jabatan awal,

Nama Use Case	Mencatat Data Pegawai	
		pendidikan, dan lain-lain
	4.	Aktor menekan tombol simpan
	5.	Melakukan validasi inputan aktor.
	6.	Mengirimkan data-data pegawai yang telah divalidasi sistem ke <i>server</i> .
	7.	Menyimpan data pegawai ke dalam <i>database</i>
	8.	Menampilkan pesan simpan pegawai baru berhasil.
	9.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	2.1.	Menekan tombol <i>edit</i> untuk mengubah data pegawai yang ada di dalam <i>database</i> .
	2.2.	Mengarahkan pengguna kepada halaman untuk mengedit data pegawai.
	2.3.	Membaca data pegawai yang ada di dalam <i>database</i> .
	2.4.	Mengubah data-data yang diperlukan.
	2.5	Menekan tombol simpan.
	5.1	Memeriksa seluruh inputan yang diperlukan, apabila ada yang kosong, maka menampilkan pesan bahwa ada data yang kosong kemudian kembali ke langkah yang ke-3

3. Flow Of Events Use Case Data Pelanggaran

Berikut ini disajikan *flow of events* dari *use case* diagram mencatat data pelanggaran. HRD memasukan data pelanggaran apabila terjadi sebuah pelanggaran yang dilakukan pegawai melanggar tata tertib aturan yang ada pada PKIS Sekar Tanjung. *Flow of events* dari *use case* mencatat data pelanggaran disajikan dalam tabel 3.3.

Tabel 3.3 *Flow Of Events* Data Pelanggaran

Nama Use Case	Mencatat Pelanggaran	
Kebutuhan terkait	Pegawai yang melakukan pelanggaran tata tertib saat bekerja akan diberi surat peringatan(SP) oleh HRD. HRD akan memasukkan data pelanggaran tersebut kedalam sistem untuk menjadi histori.	
Tujuan	Menangani pelanggaran yang dilakukan pegawai.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	HRD yang benar menginputkan data pelanggaran maka dapat disimpan di dalam tabel pelanggaran.	
Kondisi akhir gagal	Jika kegagalan tidak ditemukan NIK pegawai pada tabel pegawai, maka pencatatan pelanggaran tidak dapat dilakukan.	
Aktor utama	HRD.	
Aktor sekunder	Tidak Ada	
Pemicu	Aktor HRD memilih menu “Pelanggaran” untuk mencatat pelanggaran yang dilakukan.	
Alur utama	Langkah	Aksi
	1.	Memilih menu pelanggaran.
	2.	Mengarahkan aktor ke halaman pelanggaran.
	3.	Aktor memasukkan dalam sistem data-data pelanggaran pegawai seperti NIK,

Nama Use Case	Mencatat Pelanggaran	
		nama pegawai, tanggal kejadian, keterangan.
	4.	Aktor menekan tombol simpan.
	5.	Melakukan validasi inputan aktor.
	6.	Mengirimkan data-data pelanggaran yang telah divalidasi ke <i>server</i> .
	7.	Menyimpan data pelanggaran ke dalam <i>database</i>
	8.	Menampilkan pesan simpan pelanggaran berhasil.
	9.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	2.1.	Menekan tombol <i>edit</i> pelanggaran untuk mengubah data pelanggaran yang ada di dalam <i>database</i> .
	2.2.	Mengarahkan pengguna kepada halaman untuk mengedit pelanggaran.
	2.3.	Membaca data pelanggaran yang ada di dalam <i>database</i> .
	2.4.	Mengubah data-data yang diperlukan.
	2.5.	Menekan tombol simpan.
	5.1.	Memeriksa seluruh inputan yang diperlukan, apabila ada yang kosong, maka menampilkan pesan bahwa ada data yang kosong kemudian kembali ke langkah yang ke-3

4. Flow Of Events Use Case Data Mutasi

Berikut ini disajikan *flow of events* dari *use case* diagram mencatat data mutasi. Pencatatan data mutasi dilakukan setelah pegawai mendapatkan surat pindah ke bagian lain di PKIS Sekar Tanjung setelah itu HRD memasukkan data mutasi tersebut kedalam *database* sebagai histori. *Flow of events* dari *use case* mencatat data mutasi disajikan dalam tabel 3.4.

Tabel 3.4 *Flow Of Events* Data Mutasi

Nama Use Case	Mencatat Mutasi	
Kebutuhan terkait	HRD mengelola perpindahan pegawai atas permintaan <i>supervisor</i> yang membutuhkan tambahan pegawai pada bagiannya.	
Tujuan	Menangani perpindahan pegawai.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	HRD yang benar menginputkan data mutasi maka dapat disimpan di dalam tabel mutasi.	
Kondisi akhir gagal	Jika kegagalan tidak ditemukan NIK pegawai pada tabel pegawai, maka pencatatan mutasi tidak dapat dilakukan.	
Aktor utama	HRD.	
Aktor sekunder	Tidak ada	
Pemicu	Aktor HRD memilih menu “mutasi” untuk mencatat perpindahan bagian yang baru terjadi.	
Alur utama	Langkah	Aksi
	1.	Memilih menu mutasi.
	2.	Mengarahkan aktor ke halaman mutasi.
	3.	Aktor memasukkan dalam sistem data-data mutasi pegawai seperti NIK, nama

Nama Use Case	Mencatat Mutasi	
		pegawai, departemen baru, jabatan baru, lama jabatan awal.
	4.	Aktor menekan tombol simpan.
	5.	Melakukan validasi inputan aktor.
	6.	Mengirimkan data-data mutasi yang telah divalidasi sistem ke <i>server</i> .
	7.	Menyimpan data mutasi ke dalam <i>database</i>
	8.	Menampilkan pesan simpan mutasi berhasil.
	9.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	2.1.	Menekan tombol <i>edit</i> mutasi untuk mengubah data mutasi yang ada di dalam <i>database</i> .
	2.2.	Mengarahkan pengguna kepada halaman untuk mengedit mutasi.
	2.3.	Membaca data mutasi yang ada di dalam <i>database</i> .
	2.4.	Mengubah data-data yang diperlukan.
	2.5.	Menekan tombol simpan.
	5.1	Memeriksa seluruh inputan yang diperlukan, apabila ada yang kosong, maka menampilkan pesan bahwa ada data yang kosong kemudian kembali ke langkah yang ke-3

5. Flow of events Use Case Data Kontrak Kerja

Berikut ini disajikan *flow of events* dari *use case* diagram mencatat data kontrak untuk pegawai lama. Sebelum pegawai masuk, HRD membuat perjanjian atau kontrak kerja terlebih dahulu kepada pegawai. *Flow of events* dari *use case* mencatat data kontrak kerja disajikan dalam tabel 3.5.

Tabel 3.5 *Flow Of Events* Data Kontrak

Nama Use Case	Mencatat Kontrak kerja	
Kebutuhan terkait	Pegawai yang telah melakukan kontrak kerja atau perjanjian kerja, HRD akan memasukkan data kontrak tersebut ke dalam sistem untuk menjadi histori.	
Tujuan	Menangani kontrak kerja untuk pegawai.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	HRD yang benar menginputkan data kontrak maka dapat disimpan di dalam tabel kontrak	
Kondisi akhir gagal	Jika kegagalan tidak ditemukan NIK pegawai pada tabel pegawai, maka pencatatan kontrak tidak dapat dilakukan.	
Aktor utama	HRD.	
Aktor sekunder	Tidak ada.	
Pemicu	Aktor HRD memilih menu “kontrak kerja” untuk mencatat kontrak pegawai.	
Alur utama	Langkah	Aksi
	1.	Memilih menu kontrak kerja.
	2.	Mengarahkan aktor ke halaman kontrak kerja.
	3.	Aktor memasukkan dalam sistem data-data kontrak pegawai seperti NIK, nama pegawai, tanggal awal masuk, tanggal

Nama Use Case	Mencatat Kontrak kerja	
		akhir kerja, keterangan
	4.	Aktor menekan tombol simpan.
	5.	Melakukan validasi inputan aktor.
	6.	Mengirimkan data-data kontrak kerja yang telah divalidasi ke <i>server</i> .
	7.	Menyimpan data kontrak kerja ke dalam <i>database</i>
	8.	Menampilkan pesan simpan kontrak kerja pegawai berhasil.
	9.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	2.1.	Menekan tombol <i>edit</i> kontrak kerja untuk mengubah data kontrak kerja yang ada di dalam <i>database</i> .
	2.2.	Mengarahkan pengguna kepada halaman untuk mengedit kontrak kerja.
	2.3.	Membaca data kontrak kerja yang ada di dalam <i>database</i> .
	2.4.	Mengubah data-data yang diperlukan.
	2.5.	Menekan tombol simpan.
	5.1.	Memeriksa seluruh inputan yang diperlukan, apabila ada yang kosong, maka menampilkan pesan bahwa ada data yang kosong kemudian kembali ke langkah yang ke-3

6. Flow Of Events Use Case Training Kerja

Berikut ini disajikan *flow of events* dari *use case* diagram mencatat data *training* kerja pegawai. HRD memasukkan data *training* pegawai setelah pegawai tersebut melakukan pelatihan yang diselenggarakan oleh HRD PKIS Sekar Tanjung. *Flow of events* dari *use case* mencatat data *training* kerja disajikan dalam tabel 3.6.

Tabel 3.6 *Flow Of Events Training Kerja*

Nama Use Case	Mencatat <i>Training</i>	
Kebutuhan terkait	Pegawai yang telah melakukan pelatihan atau <i>training</i> kerja. HRD memasukkan data <i>training</i> tersebut kedalam sistem untuk menjadi histori.	
Tujuan	Menangani pelatihan yang dilakukan pegawai.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	HRD yang benar menginputkan data <i>training</i> maka dapat disimpan di dalam tabel <i>training</i> .	
Kondisi akhir gagal	Jika kegagalan tidak ditemukan NIK pegawai pada tabel pegawai, maka pencatatan <i>training</i> tidak dapat dilakukan.	
Aktor utama	HRD.	
Aktor sekunder	Tidak ada.	
Pemicu	Aktor HRD memilih menu " <i>training</i> " untuk mencatat pelatihan yang dilakukan.	
Alur utama	Langkah	Aksi
	1.	Memilih menu <i>training</i> .
	2.	Mengarahkan aktor ke halaman <i>training</i> .
	3.	Aktor memasukkan dalam sistem data-data <i>training</i> pegawai seperti nama <i>event</i>

Nama Use Case	Mencatat Training	
		<i>training</i> , keterangan <i>training</i> , tanggal, nama <i>trainer</i> , biaya, lama <i>training</i> .
	4	Aktor memasukkan nama-nama pegawai yang ikut dalam <i>training</i> berupa NIK dan nama.
	5.	Aktor menekan tombol simpan.
	6.	Melakukan validasi inputan aktor.
	7.	Mengirimkan data-data <i>training</i> yang telah divalidasi ke <i>server</i> .
	8.	Menyimpan data <i>training</i> ke dalam <i>database</i>
	8.	Menampilkan pesan simpan <i>training</i> berhasil.
	10.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	2.1.	Menekan tombol <i>edit training</i> untuk mengubah data <i>training</i> yang ada di dalam <i>database</i> .
	2.2.	Mengarahkan pengguna kepada halaman untuk mengedit <i>training</i> .
	2.3.	Membaca data <i>training</i> yang ada di dalam <i>database</i> .
	2.4.	Mengubah data-data yang diperlukan.
	2.5	Menekan tombol simpan.
	5.1.	Memeriksa seluruh inputan yang diperlukan, apabila ada yang kosong, maka menampilkan pesan bahwa ada data

Nama Use Case	Mencatat Training	
		yang kosong kemudian kembali ke langkah yang ke-3

7. Flow Of Events Use Case Penilaian Kinerja

Berikut ini disajikan *flow of events* dari *use case* diagram penilaian kinerja. *Supervisor* mendapatkan *form* penilaian dari HRD untuk menilai pegawai. *Flow of events* dari *use case* menangani penilaian kinerja disajikan dalam tabel 3.7.

Tabel 3.7 Flow Of Events Penilaian Kinerja

Nama Use Case	Mencatat Penilaian	
Kebutuhan terkait	Penilaian kinerja merupakan sebuah sarana untuk menilai kinerja pegawai saat bekerja yang hasil dari penilaian ini menghasilkan informasi bagi manajemen.	
Tujuan	Menangani penilaian kinerja pegawai.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	Jika <i>form</i> penilaian tersedia maka <i>supervisor</i> dapat melakukan penilaian kinerja.	
Kondisi akhir gagal	Jika kegagalan ada pada: data pegawai yang tidak ada pada tabel pegawai, maka penilaian tidak biasa dilakukan.	
Aktor utama	<i>Supervisor</i> .	
Aktor sekunder	HRD.	
Pemicu	Aktor <i>supervisor</i> memilih menu “penilaian pegawai” untuk melakukan penilaian.	
Alur utama	Langkah	Aksi

Nama Use Case	Mencatat Penilaian	
	1.	Memilih menu penilaian kinerja.
	2.	Mengarahkan aktor ke halaman penilaian kinerja
	3.	Menampilkan departemen dan data pegawai yang harus dinilai. Aktor menekan tombol lakukan penilaian.
	4.	Aktor mengisi <i>form</i> kuisioner penilaian yang tampil.
	5.	Aktor menekan tombol simpan setelah mengisi.
	6.	Melakukan penghitungan penilaian yang telah diinputkan oleh aktor ke sistem
	7.	Menyimpan data penilaian ke dalam <i>database</i>
	8.	Menampilkan simpan penilaian berhasil
	9.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	3.1.	Apabila pada langkah 3 tidak ada nama pegawai yang keluar maka aktor <i>shift leader</i> tidak dapat melakukan penilaian.

8. Flow Of Events Use Case Absensi

Berikut ini disajikan *flow of events* dari *use case* diagram aktivitas absensi. Pegawai memasukkan NIK dan *password* untuk melakukan proses absensi di PKIS Sekar Tanjung. *Flow of events* dari *use case* menangani absensi disajikan dalam tabel 3.8.

Tabel 3.8 *Flow Of Events* Absensi

Nama Use Case	Absensi	
Kebutuhan terkait	Absensi merupakan sarana untuk melakukan pencatatan kehadiran ketika pegawai tersebut datang dan bekerja di PKIS Sekar Tanjung	
Tujuan	Menangani absensi pegawai.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	Proses absensi pegawai.	
Kondisi akhir gagal	Jika kegagalan ada pada: data pegawai yang tidak ada pada <i>database</i> , maka absensi tidak bisa dilakukan.	
Aktor utama	Pegawai.	
Aktor sekunder	Tidak Ada.	
Pemicu	Aktor pegawai memasukkan data kehadiran ketika datang.	
Alur utama	Langkah	Aksi
	1.	Menampilkan halaman absensi dengan inputan yang diperlukan adalah <i>username</i> dan <i>password</i> .
	2.	Mengisi <i>username</i> dan <i>password</i> kemudian menekan tombol proses.
	3.	Melakukan validasi <i>password</i>
	4.	Mencocokkan <i>password</i> berdasarkan <i>username</i> yang telah diinputkan.
	5.	Menampilkan absen berhasil pada layar.
	6.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	4.1.	Menampilkan pesan: data masukkan salah, kemudian aktor harus mengulang

Nama Use Case	Absensi	
		kembali ke langkah 2.

9. Flow Of Events Use Case Membuat Laporan

Berikut ini disajikan *flow of events* dari *use case* diagram membuat laporan. Laporan yang dihasilkan dari sistem ini ada dua jenis, yaitu laporan bulanan dan laporan tahunan. *Flow of events* membuat laporan seperti disajikan dalam tabel 3.9 berikut:

Tabel 3.9 *Flow Of Events* Membuat Laporan

Nama Use Case	Membuat Laporan	
Kebutuhan terkait	Laporan kepegawaian adalah salah satu <i>output</i> dari sistem informasi kepegawaian yang menghasilkan informasi penting bagi manajemen untuk membantu dalam pengambilan keputusan.	
Tujuan	Menangani pembuatan laporan bulanan dan tahunan.	
Prasyarat	Tidak ada	
Kondisi akhir sukses	Proses membuat laporan sukses.	
Kondisi akhir gagal	<ul style="list-style-type: none"> ➤ Jika kegagalan ada pada : <i>query</i> pembuatan laporan tidak berhasil dieksekusi oleh sistem. ➤ Jika kegagalan ada pada : tidak bisa mencetak laporan, maka HRD membatalkan proses mencetak laporan. 	
Aktor utama	Bagian HRD.	
Aktor sekunder	GM dan <i>supervisor</i> .	
Pemicu	Aktor HRD memilih menu “laporan” untuk mulai membuat laporan.	
Alur utama	Langkah	Aksi

Nama Use Case	Membuat Laporan	
	1.	Memilih menu laporan.
	2.	Menampilkan menu jenis laporan yang akan dibuat.
	3.	Aktor memilih salah satu jenis laporan yang ditampilkan.
	4.	Menekan tombol tampilkan.
	5.	Mengirimkan pilihan jenis laporan yang ditampilkan
	6.	Melakukan koneksi ke <i>database</i> .
	7.	Menjalankan kueri berdasarkan pilihan jenis laporan.
	7.	Menampilkan hasil <i>query</i> laporan dan pilihan untuk mencetak laporan.
	8.	Memilih menu cetak laporan.
	9.	Menampilkan pesan sedang mencetak laporan.
	10.	<i>Use case</i> berakhir.
Alur Perluasan	Langkah	Aksi Percabangan
	4.1.	Apabila aktor yang tidak memilih jenis laporan, maka laporan tidak dapat di tampilkan dan di cetak, kembali ke langkah k-3.

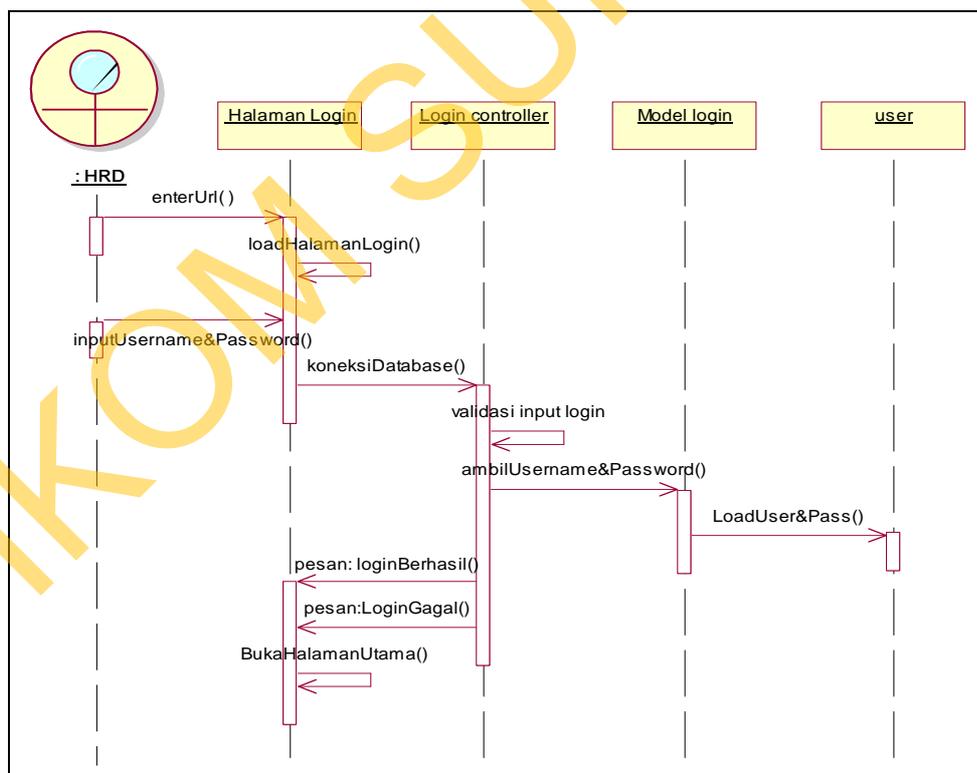
3.2.4 Sequence Diagram

Menurut Nugroho (2005) *sequence* diagram adalah *interaction* diagram yang memperlihatkan *event-event* yang berurutan sepanjang berjalannya waktu.

Diagram sekuensial menggambarkan interaksi yang dilakukan oleh objek-objek dalam sistem. Kerja sama antar objek-objek dilaksanakan dengan saling mengirimkan pesan yang membentuk sebuah alur kerja sama.

1. Diagram Sekuensial *Login*

Alur yang digambarkan pada diagram sekuensial tak ubahnya adalah *flow of events* yang sudah dibuat sebelumnya. Pada diagram sekuensial ini digambarkan proses-proses yang terjadi sewajarnya. Berikut ini adalah gambar 3.12 merupakan diagram sekuensial dari *use case login*.



Gambar 3.12 Diagram Sekuensial *Login*

Pada diagram sekuensial di atas terlihat proses *login* dimulai dari alur kegiatan yang dilakukan oleh bagian HRD membuka aplikasi Sistem Informasi Kepegawaian. Sebelum melakukan *login* bagian HRD tidak dapat mengakses halaman menu yang lain. Halaman utama keluar setelah HRD melakukan *login*, kemudian akan ditampilkan pilihan menu yang dapat dilakukan untuk menjalankan kepegawaian pada PKIS Sekar Tanjung.

HRD melakukan menginputan *username* dan *password* yang dimiliki pada sistem, kemudian *username* dan *password* tersebut divalidasi di dalam *controller login*. Pada saat memvalidasi *username* dan *password* yang dimasukkan oleh pengguna, *controller login* meminta data *username* dan *password* dari *login* model. *Login* model mengambil data dari tabel *user*, kemudian mengirimkan kepada *controller login*. Apabila *username* dan *password* sesuai dengan yang ada di *database* maka pengguna tersebut telah berhasil *login*. Apabila tidak berhasil, maka akan menampilkan pesan kepada pengguna bahwa *login* gagal.

2. Diagram Sekuensial Master Pegawai

Diagram sekuensial di gambar 3.13 merealisasikan *use case* “mencatat data pegawai”. Diagram sekuensial mencatat data pegawai juga menggambarkan *flow of events* dari proses mencatat data pegawai yang merupakan proses bisnis dari kepegawaian.

Alur mencatat data pegawai dimulai ketika HRD memilih halaman master pegawai untuk mencatat data pegawai. Halaman ini akan menampilkan seluruh

data pegawai yang ada pada PKIS Sekar Tanjung. Pada bagian HRD jika ingin menambah data pegawai baru dengan menekan tombol tambah pegawai. Kemudian *controller* pegawai mengarahkan HRD kepada halaman yang berisi *form* untuk mengisi data-data pegawai. Kemudian HRD mengisi data pegawai seperti nama, alamat, TTL, jabatan awal, pendidikan, dan lain-lain. Ketika HRD menekan tombol simpan, maka sistem akan memeriksa data-data yang telah dimasukkan. Apabila data tersebut sesuai dengan ketentuan validasi, maka *controller* pegawai dapat menyimpan data-data tersebut ke dalam tabel pegawai di dalam *database*. Apabila data yang dimasukkan HRD tidak sesuai dengan ketentuan validasi maka sistem akan menampilkan pesan data masih kosong atau salah pengisian.

HRD dapat melakukan pencarian pegawai dengan memasukkan inputan berupa nama pegawai pada kotak yang telah disediakan. Kemudian sistem akan mencarikan inputan tersebut dan akan menampilkan data tersebut pada halaman pegawai.

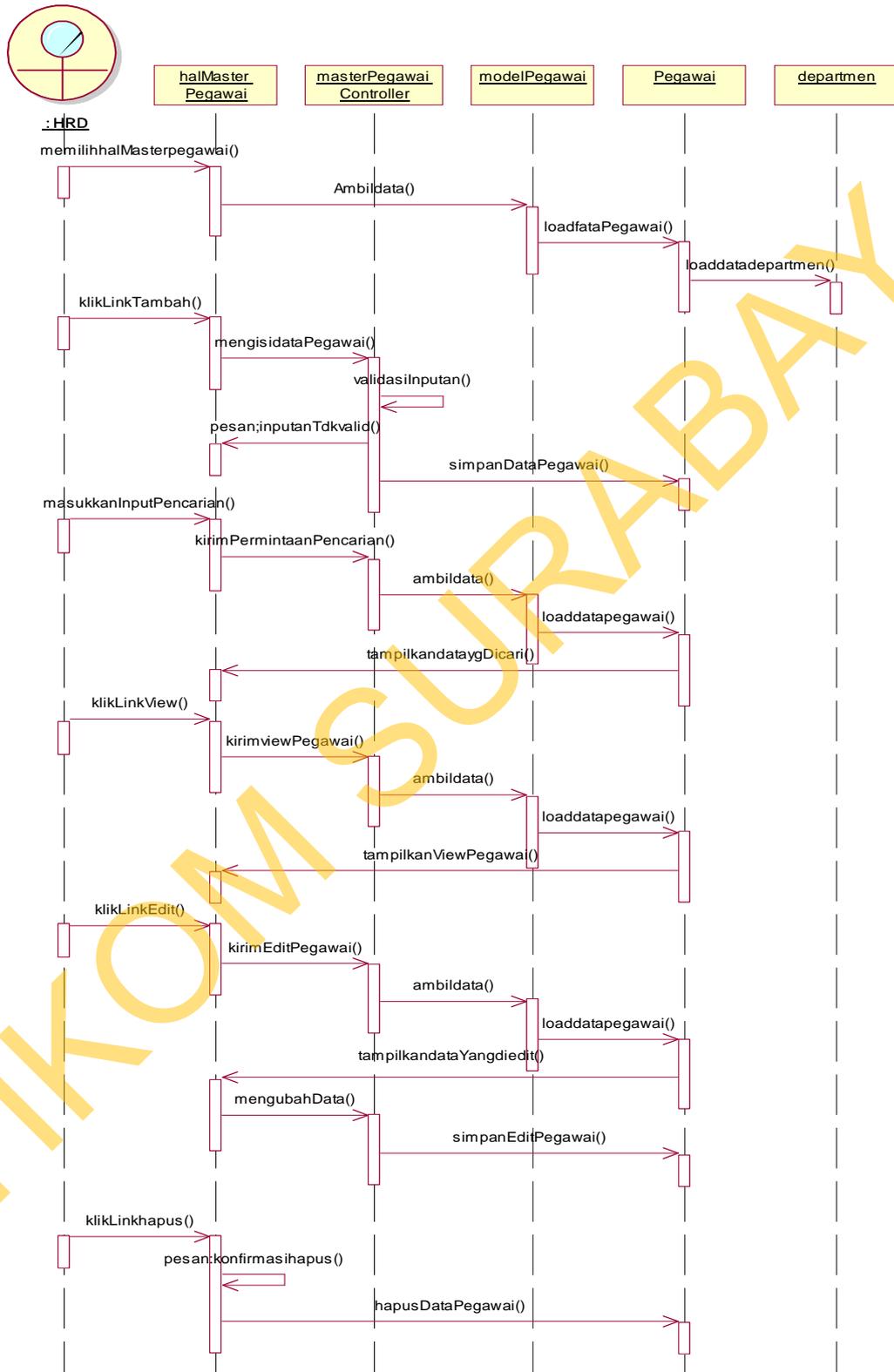
Pada halaman ini HRD juga dapat menampilkan secara detail *profile* pegawai tersebut dengan menekan tombol *view* yang ada pada sebelah nama pegawai di halaman pegawai.

HRD dapat mengubah data pegawai dengan menekan tombol *edit* yang ada pada halaman master pegawai. Kemudian sistem akan mengarahkan HRD kepada halaman yang berisi *form* untuk mengubah data pegawai. Setelah HRD selesai

mengedit yang dibutuhkan langsung menekan tombol simpan, maka sistem akan mengupdate data tersebut pada *database*.

HRD dapat menghapus data pegawai yang ada pada tabel pegawai dengan menekan tombol hapus. Ketika HRD menekan tombol hapus, sistem akan menampilkan pesan konfirmasi penghapusan data pegawai. Apabila HRD mengkonfirmasi penghapusan data tersebut, maka *controller* pegawai akan menghapus data pegawai dari dalam *database*.

STIKOM SURABAYA



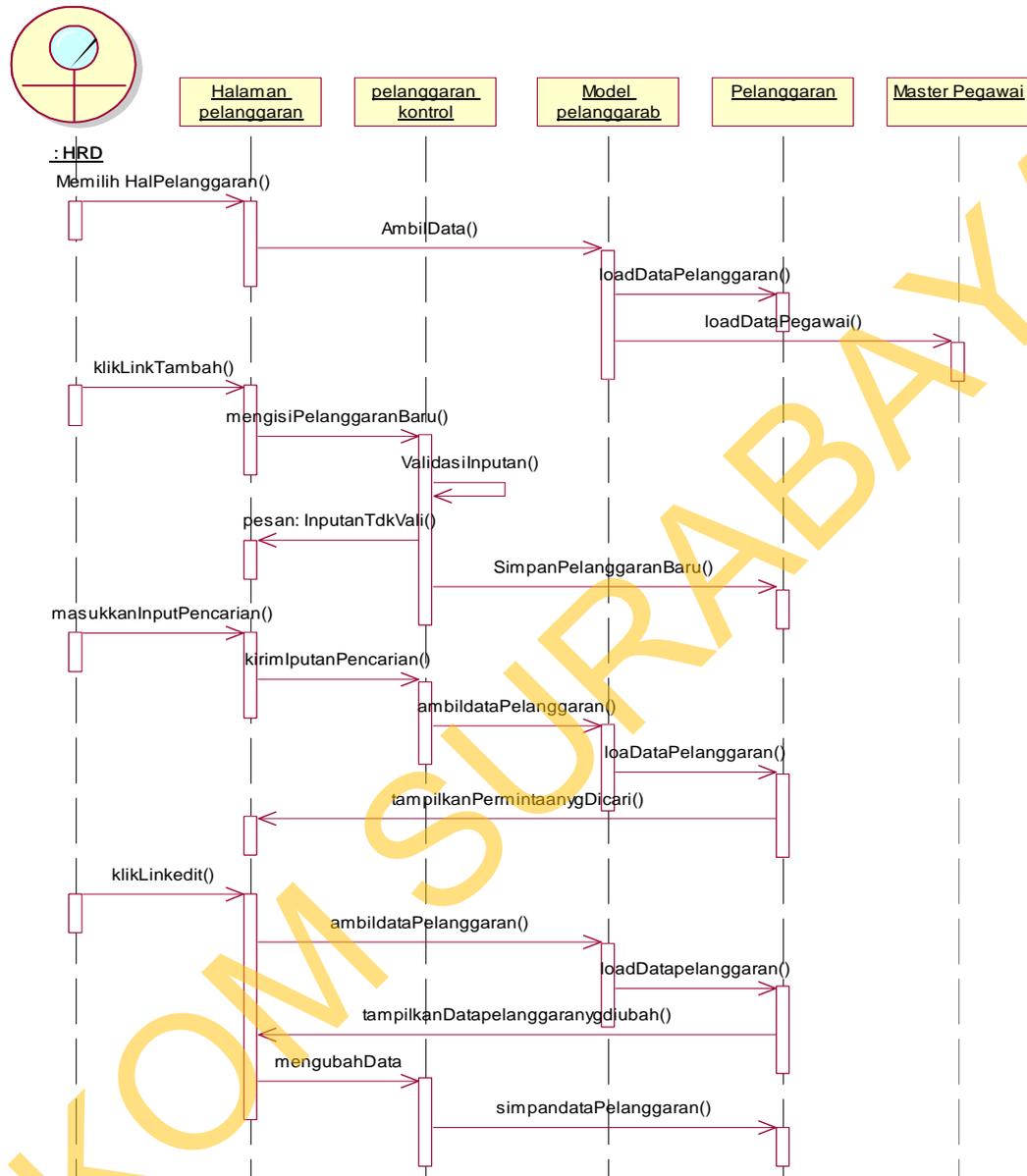
Gambar 3.13 Diagram Sekuensial Master Pegawai

3. Diagram Sekuensial Mencatat Pelanggaran

Diagram sekuensial di gambar 3.14 merealisasikan *use case* “mencatat pelanggaran”. Diagram sekuensial mencatat pelanggaran juga menggambarkan *flow of events* dari proses mencatat pelanggaran yang merupakan proses bisnis dari kepegawaian.

Alur mencatat pelanggaran dimulai ketika HRD memilih halaman pelanggaran untuk mencatat data pelanggaran. Halaman ini akan menampilkan pelanggaran apa saja yang telah terjadi setiap harinya. Jika HRD ingin menambah pelanggaran baru dengan menekan tombol tambah pelanggaran. Kemudian *controller* pelanggaran mengarahkan HRD kepada halaman yang berisi *form* untuk mengisi data-data pelanggaran. Kemudian HRD mengisi data pelanggaran seperti NIK, nama pegawai, departemen, tanggal kejadian, sanksi, keterangan pelanggaran. Ketika HRD menekan tombol simpan, maka sistem akan memeriksa data-data yang telah dimasukkan. Apabila data tersebut sesuai dengan ketentuan validasi, maka *controller* pelanggaran dapat menyimpan data-data tersebut ke dalam tabel pelanggaran di dalam *database*. Apabila data yang dimasukkan HRD tidak sesuai dengan ketentuan validasi maka sistem akan menampilkan pesan data masih kosong atau salah pengisian.

HRD dapat melakukan pencarian data pelanggaran dengan melakukan inputan berupa nama pegawai. Kemudian sistem akan mencarikan inputan tersebut sesuai inputan yang telah dimasukkan dan akan menampilkan data tersebut pada halaman pelanggaran.



Gambar 3.14 Diagram Sekuensial Mencatat Pelanggaran

HRD juga dapat mengubah data pelanggaran apabila dalam menginputan terjadi kesalahan dengan menekan tombol *edit* yang ada pada sebelah nama di halaman pelanggaran. Kemudian sistem akan mengarahkan HRD kepada halaman yang berisi *form* untuk mengubah data pelanggaran. Setelah HRD selesai

mengedit yang dibutuhkan langsung menekan tombol simpan, maka sistem akan langsung mengupdate data pelanggaran pada *database*.

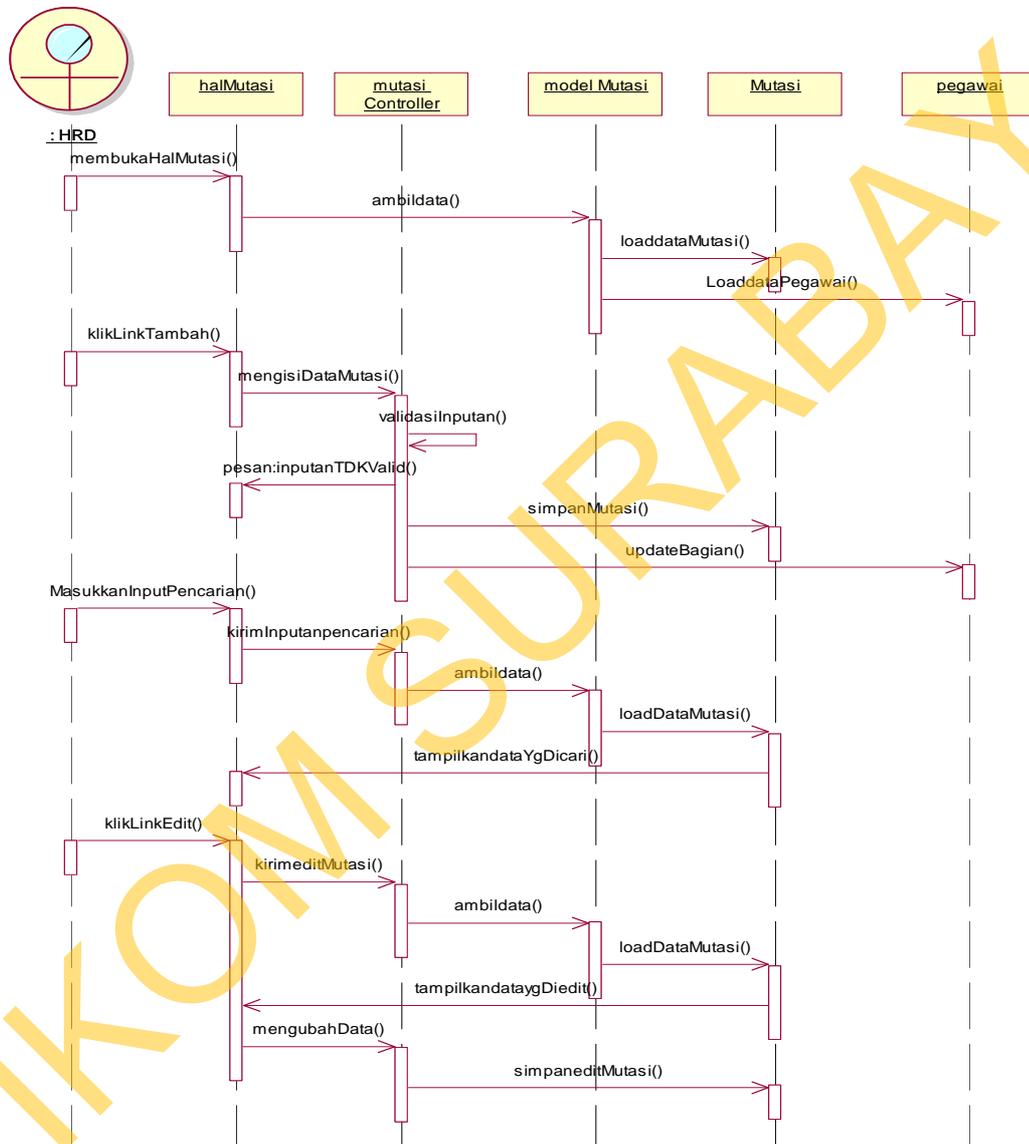
4. Diagram Sekuensial Mencatat Mutasi

Diagram sekuensial di gambar 3.15 merealisasikan *use case* “mencatat mutasi”. Diagram sekuensial mencatat mutasi juga menggambarkan *flow of events* dari proses mencatat mutasi yang merupakan proses bisnis dari kepegawaian.

Alur mencatat mutasi dimulai ketika HRD memilih halaman mutasi untuk mencatat data mutasi. Halaman ini akan menampilkan perpindahan posisi sebelum dan sesudah dipindahkan. Melalui halaman mutasi ini jika ingin menambah mutasi baru dengan menekan tombol tambah mutasi. Kemudian *controller* mutasi mengarahkan HRD kepada halaman yang berisi *form* untuk mengisi data-data mutasi. Kemudian HRD mengisi data mutasi seperti NIK, nama pegawai, posisi jabatan awal, posisi jabatan baru, lama jabatan awal. Ketika HRD menekan tombol simpan, maka sistem akan memeriksa data-data yang telah dimasukkan. Apabila data tersebut sesuai dengan ketentuan validasi, maka *controller* mutasi dapat menyimpan data-data tersebut ke dalam tabel mutasi dan mengubah posisi departemen yang baru pada *profile* pegawai. Apabila data yang dimasukkan HRD tidak sesuai dengan ketentuan validasi maka sistem akan menampilkan pesan data masih kosong atau salah pengisian.

HRD dapat melakukan pencarian data mutasi dengan melakukan inputan berupa nama pegawai. Kemudian sistem akan mencari inputan tersebut sesuai

masukkan yang telah dimasukkan dan akan menampilkan data tersebut pada halaman mutasi.



Gambar 3.15 Diagram Sekuensial Mencatat Mutasi

HRD dapat mengubah data mutasi dengan menekan tombol *edit* yang ada pada halaman mutasi. Kemudian sistem akan mengarahkan HRD kepada halaman

yang berisi *form* untuk mengubah data mutasi. Setelah HRD selesai mengedit yang dibutuhkan langsung menekan tombol simpan, maka sistem akan memeriksa data mutasi yang dimasukkan dan mengupdate langsung pada *database*.

5. Diagram Sekuensial Mencatat *Training*

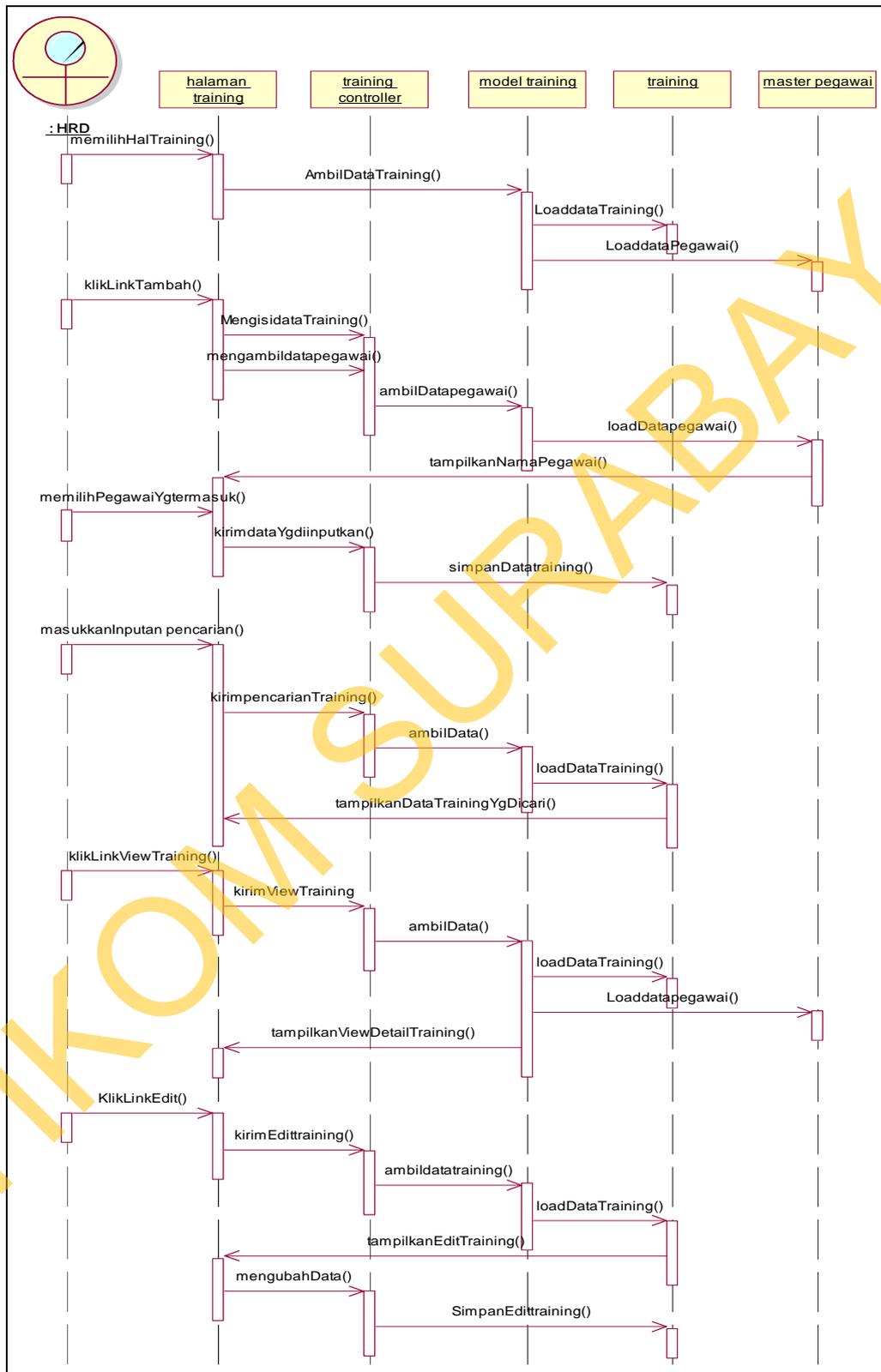
Diagram sekuensial di gambar 3.16 merealisasikan *use case* “mencatat *training*”. Diagram sekuensial mencatat *training* juga menggambarkan *flow of events* dari proses mencatat *training* yang merupakan proses bisnis dari kepegawaian.

Alur mencatat *training* dimulai ketika HRD memilih halaman *training* untuk mencatat data *training*. Halaman ini akan menampilkan *training* apa saja yang telah diikuti pegawai di PKIS Sekar Tanjung. Melalui halaman ini HRD dapat menambah data kegiatan *training* yang baru saja yang dilakukan. Kemudian *training controller* mengarahkan HRD kepada halaman yang berisi *form* untuk mengisi data-data *training*. Kemudian HRD mengisi data *training* seperti nama *training*, tanggal *training*, nama *trainer*, keterangan *training*, biaya. Setelah HRD membuat *event training* selanjutnya HRD memasukkan nama-nama pegawai yang mengikutinya. Ketika HRD menekan tombol simpan, maka sistem akan memeriksa data-data yang telah dimasukkan. Apabila data tersebut sesuai dengan ketentuan validasi, maka *controller training* dapat menyimpan data-data tersebut ke dalam tabel *training* di dalam *database*. Apabila data yang dimasukkan HRD

tidak sesuai dengan ketentuan validasi maka sistem akan menampilkan pesan data masih kosong atau salah pengisian.

HRD dapat melakukan pencarian data *training* dengan melakukan inputan berupa nama pegawai. Kemudian sistem akan mencari inputan tersebut sesuai masukkan yang telah dimasukkan dan akan menampilkan data tersebut pada halaman *training*.

STIKOM SURABAYA



Gambar 3.16 Diagram Sekuensial Mencatat *Training*

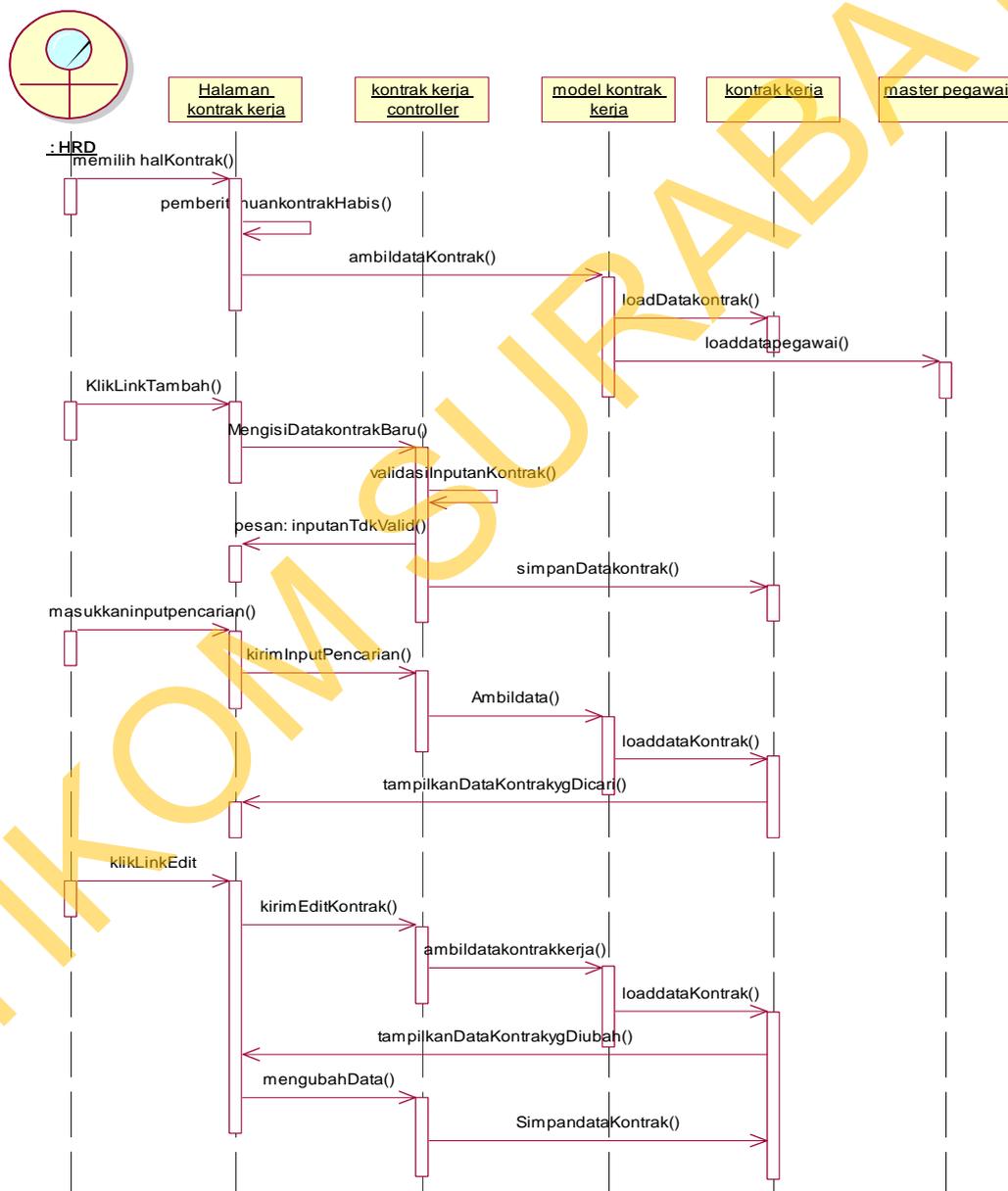
HRD juga dapat mengubah data *training* apabila terjadi kesalahan pemasukan data dengan menekan tombol *edit* yang ada pada halaman *training*. Kemudian sistem akan mengarahkan HRD kepada halaman yang berisi *form* untuk mengubah data *training*. Setelah HRD selesai mengedit yang dibutuhkan langsung menekan tombol simpan *edit training*, maka sistem akan mengupdate data tersebut pada *database*.

6. Diagram Sekuensial Mencatat Kontrak Kerja

Diagram sekuensial di gambar 3.17 merealisasikan *use case* “mencatat kontrak”. Diagram sekuensial mencatat kontrak juga menggambarkan *flow of events* dari proses mencatat kontrak yang merupakan proses bisnis dari kepegawaian.

Alur mencatat kontrak kerja dimulai ketika HRD memilih halaman kontrak kerja untuk mencatat data kontrak kerja. Pada saat membuka halaman ini akan menampilkan kontrak kerja yang akan berakhir masa kontrak kerja. Melalui halaman ini jika HRD ingin menambah kontrak kerja baru dengan menekan tombol tambah kontrak kerja. Kemudian *controller* kontrak kerja mengarahkan HRD kepada halaman yang berisi *form* untuk mengisi data-data kontrak kerja. Kemudian HRD mengisi data kontrak kerja seperti NIK, nama pegawai, tanggal awal masuk, tanggal akhir kerja, dan keterangan. Ketika HRD menekan tombol simpan, maka sistem akan memeriksa data-data yang telah dimasukkan. Apabila data tersebut sesuai dengan ketentuan validasi, maka *controller* kontrak kerja

dapat menyimpan data-data tersebut ke dalam tabel kontrak kerja di dalam *database*. Apabila data yang dimasukkan HRD tidak sesuai dengan ketentuan validasi maka sistem akan menampilkan pesan data masih kosong atau salah pengisian.



Gambar 3.17 Diagram Sekuensial Mencatat Kontrak Kerja

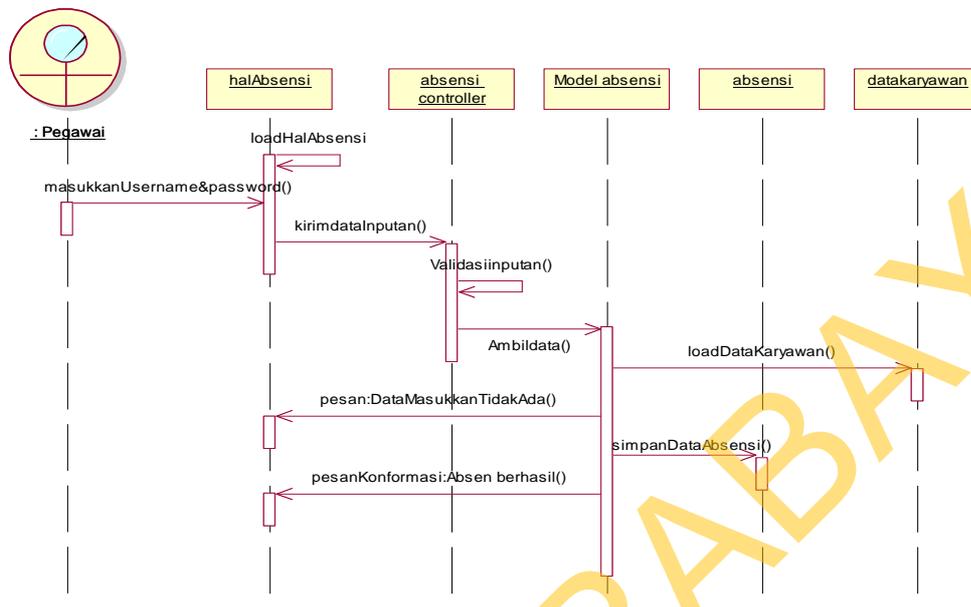
HRD dapat melakukan pencarian data kontrak dengan melakukan inputan berupa nama pegawai. Kemudian sistem akan mencari data tersebut sesuai masukkan yang telah dimasukkan dan akan menampilkan data tersebut pada halaman kontrak kerja.

Selain itu jika HRD ingin mengubah data kontrak kerja dengan menekan tombol *edit* yang ada pada halaman kontrak kerja. Kemudian sistem akan mengarahkan HRD kepada halaman yang berisi *form* untuk mengubah data kontrak kerja. Setelah HRD selesai mengedit yang dibutuhkan langsung menekan tombol simpan, maka sistem akan mengupdate data tersebut pada *database*.

7. Diagram Sekuensial Absensi

Diagram sekuensial di gambar 3.18 merealisasikan *use case* “Absensi”. Diagram sekuensial absensi juga menggambarkan *flow of events* dari proses absensi yang merupakan proses bisnis dari kepegawaian.

Alur proses absensi ini dilakukan pada halaman absensi pegawai. Dimulai dari pegawai memasukkan *input* berupa *username* dan *password* ke dalam sistem, kemudian sistem mengecek validasi *username* dan *password* tersebut ke dalam *database* lalu setelah data benar langsung dapat menyimpan data absensi. Pegawai akan mendapatkan pemberitahuan absen berhasil.



Gambar 3.18 Diagram Sekuensial Absensi

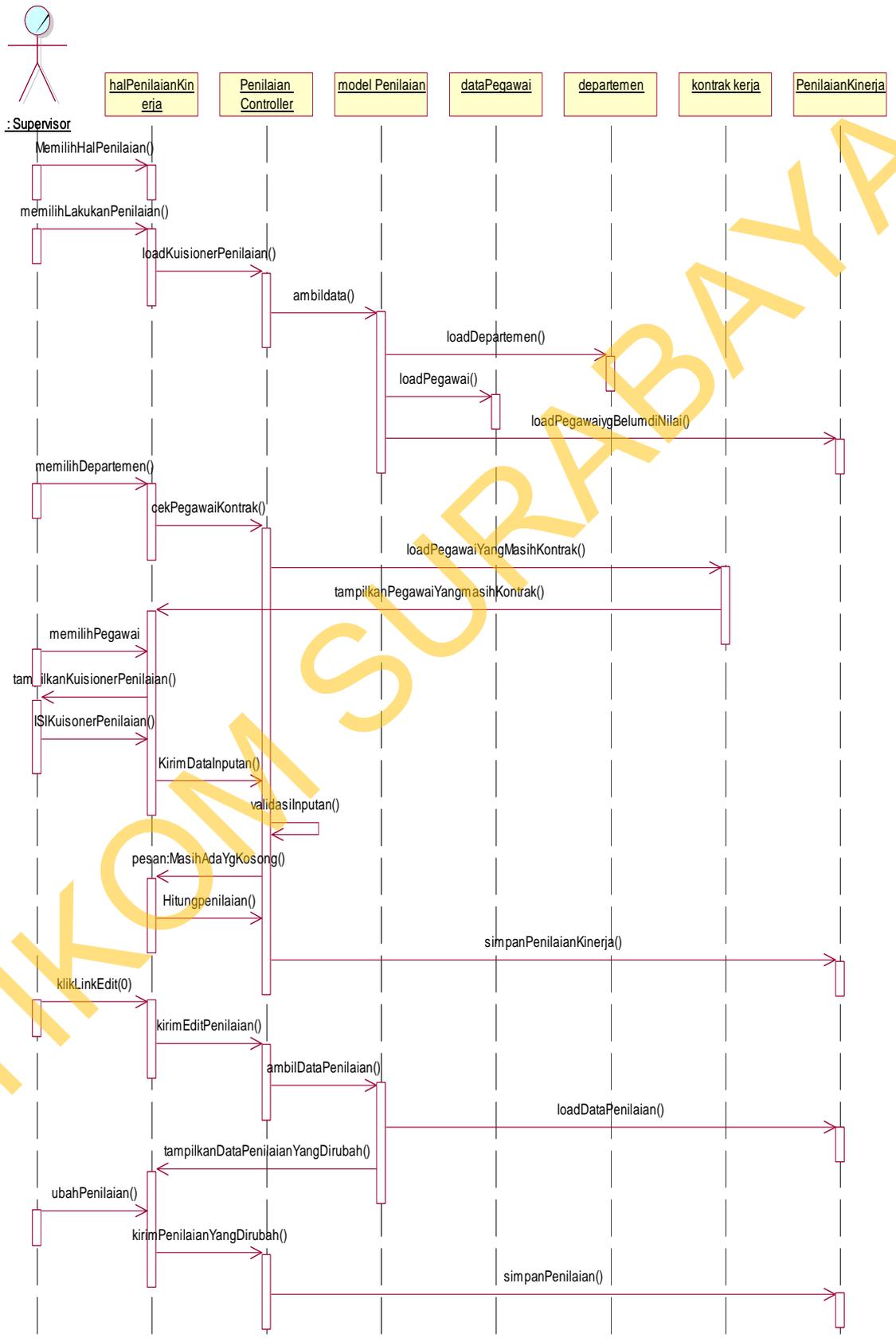
8. Diagram Sekuensial Melakukan Penilaian Kinerja

Diagram sekuensial di gambar 3.19 merealisasikan *use case* “Penilaian Kinerja”. Diagram sekuensial penilaian kinerja juga menggambarkan *flow of events* dari proses penilaian kinerja yang merupakan proses bisnis dari kepegawaian.

Penilaian pegawai dilakukan oleh *shift leader*. Kemudian *controller penilaian* akan mengambil data pegawai yang akan dinilai, departemennya. Setelah *shift leader* mendapatkan data-data tersebut, kemudian *shift leader* memberikan penilaian berupa “Ya” dan “Tidak” pada poin *form* yang tampil kemudian tekan tombol simpan untuk langsung disimpan ke dalam *database* penilaian kinerja. *Shift leader* / HRD jg dapat mengubah penilaian tersebut apabila

terjadi kesalahan input data penilaian. *Shift leader* / HRD menekan tautan edit sebelah nama pegawai setelah selesai tekan simpan.

STIKOM SURABAYA

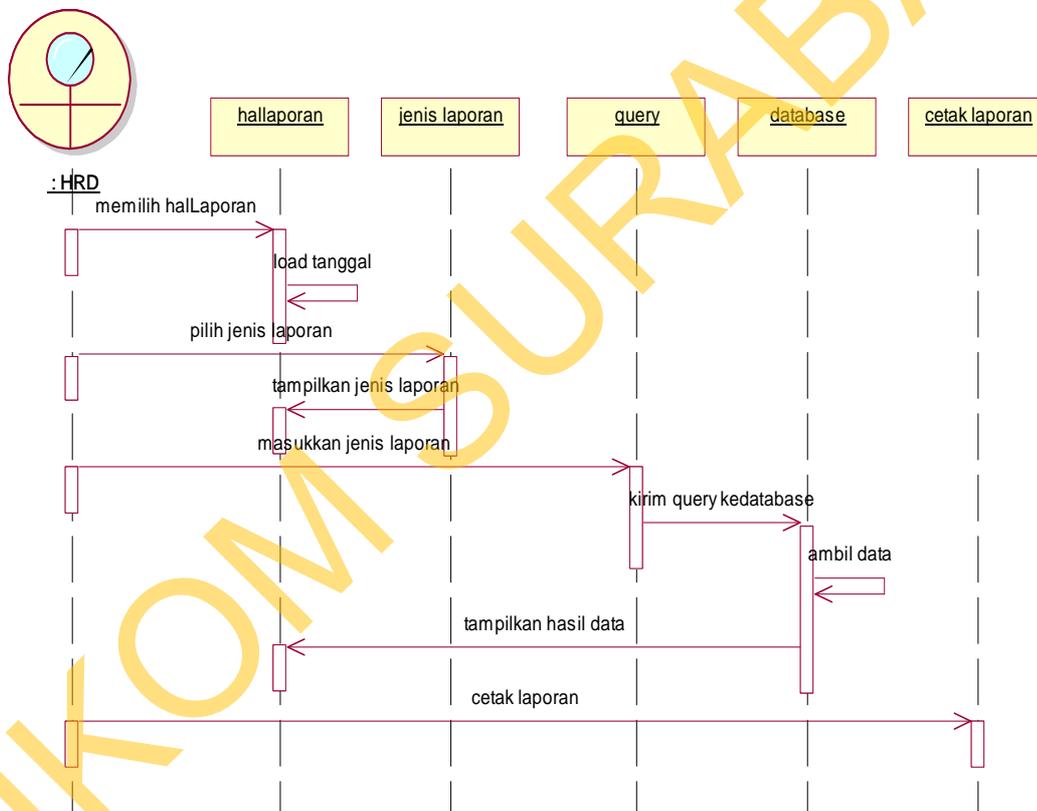


STIKOM SURABAYA

Gambar 3.19 Diagram Sekuensial Penilaian Kinerja

9. Diagram Sekuensial Membuat Laporan

Diagram sekuensial di gambar 3.20 merealisasikan *use case* “Laporan”. Diagram sekuensial laporan kinerja juga menggambarkan *flow of events* dari proses laporan yang merupakan proses bisnis akhir dari sistem informasi kepegawaian.



Gambar 3.20 Diagram Sekuensial Laporan

Aliran proses membuat laporan dimulai dari petugas yang memilih halaman laporan pada halaman utama *web*. Setelah itu akan menampilkan pilihan jenis laporan yang ingin dibuat. Setelah HRD memilih jenis laporan, halaman

laporan akan mengirimkan jenis pilihan pada *query* yang telah disiapkan di sistem kemudian dikirim ke *database* untuk diproses. Setelah *query* berhasil dijalankan, hasilnya akan ditampilkan pada halaman laporan yang diminta dan dapat dicetak.

3.2.5 Class Diagram

Class diagram digunakan untuk menampilkan *class* atau paket-paket dalam sistem dan relasi antar mereka. Biasanya, dibuat beberapa diagram *class* untuk satu sistem. Satu diagram *class* menampilkan subset dari *class* dan relasinya. Diagram *class* lainnya mungkin menampilkan *class* termasuk atribut dan operasi dari *class* pembentuk diagram.

Menurut Sholiq (2010) diagram *class* adalah alat perancangan terbaik untuk tim pengembang perangkat lunak. Diagram *class* membantu tim pengembang mendapatkan pola *class* dalam sistem, struktur sistem sebelum menuliskan kode program, dan membantu untuk memastikan bahwa sistem adalah rancangan terbaik dari beberapa alternatif rancangan.

Berdasarkan perencanaan sistem pada *use case* diagram, dibutuhkan *class* untuk membangun dan mendukung jalannya aplikasi. Hubungan antar *class* tersebut dapat digambarkan dalam sebuah *class* diagram. *Class* diagram dari sistem yang dibangun tidak ditampilkan secara keseluruhan, melainkan ditampilkan secara terpisah sesuai relasi-relasi yang ada untuk mempermudah pembacaan.

Class yang dibuat untuk aplikasi ini menggunakan pemodelan UML dengan konsep pemodelan *Model-View-Controller* (MVC). *Class* model

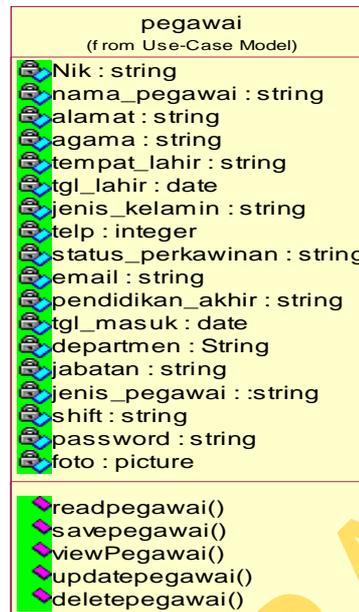
merupakan *class* yang akan menangani segala sesuatu yang berhubungan dengan entitas. *Class view* merupakan *class* yang akan menangani segala sesuatu yang berhubungan dengan tampilan *user interface*. *Class controller* merupakan *class* yang menangani segala proses seperti proses pencarian data dan segala proses yang berhubungan dengan penyimpanan data pada *SQL server*. *Class* yang dapat digunakan dalam *class* diagram pada sistem informasi kepegawaian yaitu:

1. *Class Model*

Class Model atau *class* entitas adalah *class* yang digunakan menangani informasi yang mungkin disimpan secara permanen di dalam *database*. Berikut ini merupakan *class* model yang ada dalam sistem informasi kepegawaian beserta atribut dan operasi dalam *class* tersebut.

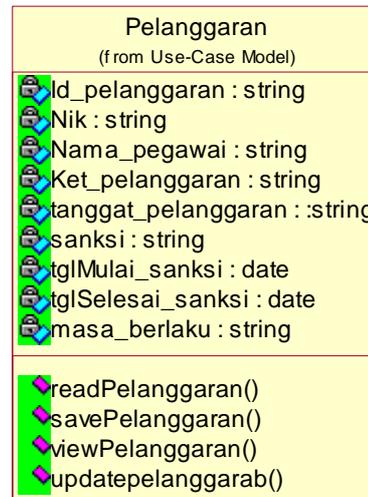
A. *Class Pegawai*

Class pegawai berfungsi menangani data pegawai yang digunakan untuk melakukan seluruh proses kepegawaian. *Class* pegawai mempunyai atribut berupa NIK, nama, alamat, tgl_lahir, tgl_masuk, agama, gender, telp, email, sts_perkawinan, status_kerja, departmen, jabatan, pendidikan_akhir. Pada gambar 3.21 berikut ini adalah notasi untuk *class* pegawai.

Gambar 3.21 Notasi *Class* Pegawai

B. *Class* Pelanggaran

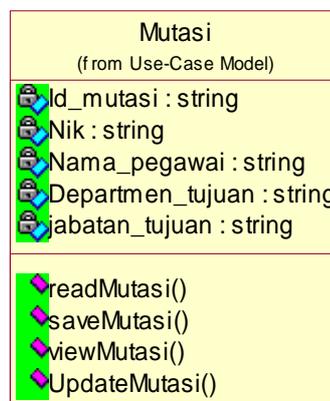
Class pelanggaran berfungsi untuk menangani data pelanggaran. Setiap pelanggaran yang dilakukan oleh pegawai saat bekerja semuanya dimasukkan ke dalam data ini. Atribut yang dimiliki *class* pelanggaran adalah id_pelanggaran, tanggal, pelanggaran, sanksi, dari, sampai. Notasi *class* pelanggaran dapat dilihat pada gambar 3.22.



Gambar 3.22 Notasi *Class* Pelanggaran

C. *Class* Mutasi

Class mutasi berfungsi menangani data mutasi pegawai yang dipindahkan. Bagian HRD mencatat agar dapat diketahui histori jabatan pegawai sebelum dan sesudah dipindahkan. Atribut yang dimiliki *class* mutasi adalah id_mutasi, NIK, nama, jabatan awal, jabatan akhir, dari, sampai, lama_menjabat. Notasi *class* golongan dapat dilihat pada gambar 3.23.



Gambar 3.23 Notasi *Class* Mutasi

D. Class Kontrak Kerja

Class kontrak kerja berfungsi menangani data kontrak kerja pegawai. HRD mencatat kontrak kerja agar diketahui lama masa kontrak. Atribut yang dimiliki *class* kontrak kerja adalah Notasi *class* golongan dapat dilihat pada gambar 3.24.



Gambar 3.24 Notasi *Class* Kontrak Kerja

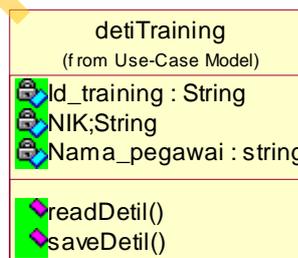
E. Class Training

Class training berfungsi menangani data *training* yang digunakan untuk mengetahui *training* apa saja yang telah pegawai terima selama bekerja di PKIS Sekar Tanjung. Atribut yang dimiliki *class training* adalah *id_training*, *nama_training*, *ket_training*, *nama_trainer*, *tgl_training*, *lama_training*, *biaya*. Notasi *class training* dapat dilihat pada gambar 3.25.

Gambar 3.25 Notasi *Class Training*

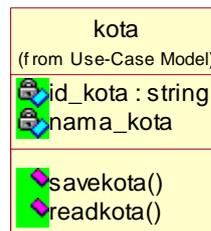
F. *Class DetilTraining*

Class detilTraining berfungsi untuk menangani seluruh data mengenai detiltraining. Data-data mengenai detiltraining disimpan ke dalam atribut yang dimiliki *class* detilTraining yaitu Id_training, NIK, nama pegawai. Notasi *class* detiltraining dapat dilihat pada gambar 3.26.

Gambar 3.26 Notasi *Class DetilTraining*

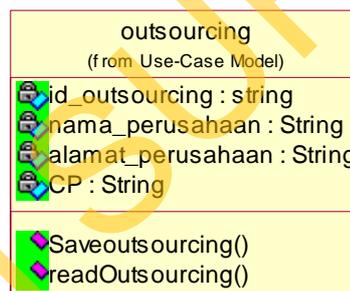
G. *Class Kota*

Class kota memiliki atribut yaitu id_kota dan nama_kota. Notasi *class* kota dapat dilihat pada gambar 3.27.

Gambar 3.27 Notasi *Class Kota*

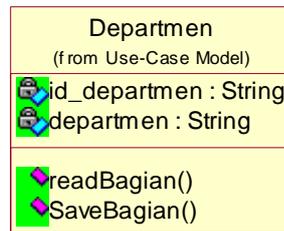
H. *Class Outsourcing*

Class outsourcing memiliki atribut yaitu `id_outsourcing`, `nama_perusahaan`, `alamat_perusahaan`, `alamat` dan `contact person`. Notasi *class outsourcing* dapat dilihat pada gambar 3.28.

Gambar 3.28 Notasi *Class Outsourcing*

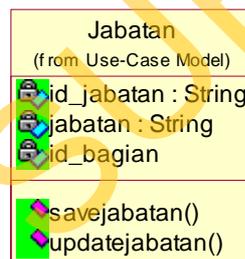
I. *Class Department*

Class department berfungsi menangani data mengenai bagian yang ada pada PKIS Sekar Tanjung. Atribut yang dimiliki *class department* yaitu `id_departmen` dan `departmen`. Notasi *class departmen* dapat dilihat pada gambar 3.29.

Gambar 3.29 Notasi *Class Department*

J. *Class Jabatan*

Class jabatan berfungsi menangani data mengenai jabatan yang ada pada PKIS Sekar Tanjung. Atribut yang dimiliki *class bagian* yaitu `id_jabatan` dan `jabatan`. Notasi *class jabatan* dapat dilihat pada gambar 3.30.

Gambar 3.30 Notasi *Class Jabatan*

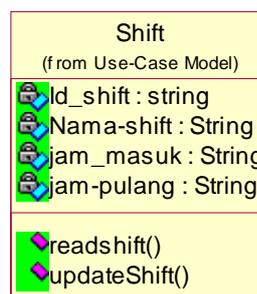
K. *Class Absensi*

Class absensi berfungsi menangani data absensi yang digunakan untuk mengetahui kehadiran pegawai selama bekerja di PKIS Sekar Tanjung. Atribut yang dimiliki *class absensi* adalah `id_absen`, `NIK`, `nama`, `jammasuk`, `jampulang`, `keterangan`. Notasi *class absensi* dapat dilihat pada gambar 3.31.

Gambar 3.31 Notasi *Class Absensi*

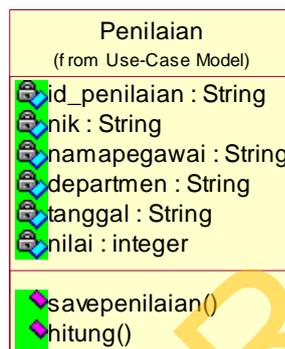
L. *Class Shift*

Class shift berfungsi menangani data *shift* yang digunakan untuk membagi beberapa bagian pada saat masuk kerja. Atribut yang dimiliki *class shift* adalah `id_shift`, `nama_shift`, `jam_masuk`, `jam_pulang`. Notasi *class shift* dapat dilihat pada gambar 3.32.

Gambar 3.32 Notasi *Class Shift*

M. *Class Penilaian Kinerja*

Class penilaian kinerja berfungsi menangani data penilaian yang digunakan untuk menilai pegawai selama bekerja di PKIS Sekar Tanjung. Atribut yang dimiliki *class* penilaian kinerja adalah *id_penilaian*, *NIK*, *nama*, *department*, *nilai*. Notasi *class* penilaian kinerja dapat dilihat pada gambar 3.33.



Gambar 3.33 Notasi *Class* Penilaian

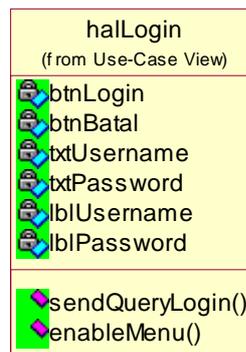
2. *Class View*

Pengertian dari *class view* sebenarnya hampir sama dengan *class* pembatas (*boundary class*). Yaitu *class* yang terletak diantara sistem dengan sekelilingnya. Semua *form*, laporan, *user interface*, termasuk dalam kategori *class view*. Berikut ini *class view* pada sistem informasi kepegawaian.

A. *Class HalLogin*

Class halLogin digunakan untuk menangani fungsi-fungsi pada halaman *login* sebagai tampilan awal pada aplikasi sistem informasi kepegawaian. Halaman *login* ini digunakan juga untuk verifikasi data pegawai sebelum mengakses fungsi lain pada aplikasi. Atribut yang digunakan pada halLogin yaitu

btnLogin, btnBatal, txtUsername, txtPassword, lblUsername dan lblPassword. Operasi pada *class* halLogin yaitu sendQueryLogin, dan enableMenu. Notasi *class* halLogin dapat dilihat pada gambar 3.34.



Gambar 3.34 Notasi *Class* HalLogin

B. *Class* HalUtama

Class halUtama digunakan untuk menangani fungsi-fungsi pada halaman utama sebagai tampilan setelah melakukan *login* pada aplikasi sistem informasi kepegawaian. Atribut yang digunakan pada halUtama yaitu btnMasterPegawai, btnPelanggaran, btnMutasi, btnKontrak, btnTraining, btnAbsen, btnPenilaianKinerja dan btnLogout. Operasi yang dimiliki *class* halUtama yaitu openMasterPegawai, openPelanggaran, openMutasi, openKontrak, openTraining, openAbsen, openPenilaianKinerja dan openLogout. Notasi *class* HalUtama dapat dilihat pada gambar 3.35.



Gambar 3.35 Notasi *Class* HalUtama

C. *Class* Halaman_Manage_Pegawai

Class Halaman_Manage_Pegawai digunakan untuk mengelola data pegawai. Melalui halaman ini bagian HRD dapat menampilkan profil pegawai, menambahkan data pegawai baru, mengubah data pegawai, serta menghapus data pegawai di dalam *database*.

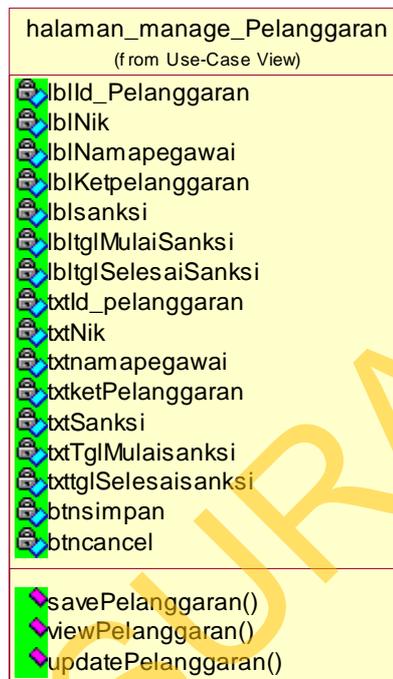


Gambar 3.36 Notasi Class Halaman_Manage_Pegawai

D. Class Halaman_Manage_Pelanggaran

Class `Halaman_Manage_Pelanggaran` digunakan untuk mengelola pelanggaran yang dilakukan pegawai. Melalui halaman ini bagian HRD dapat

menambahkan data pelanggaran baru, mengubah data pelanggaran, serta menghapus data pelanggaran pegawai di dalam *database*.

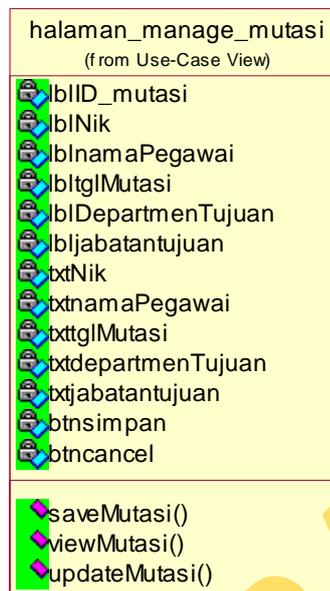


Gambar 3.37 Notasi *Class* Halaman_Manage_Pelanggaran

E. *Class* Halaman_Manage_Mutasi

Class Halaman_Manage_Mutasi digunakan untuk mengelola data mutasi.

Melalui halaman ini bagian HRD dapat menambahkan data mutasi baru, mengubah data mutasi, serta menghapus data mutasi pegawai di dalam *database*.



Gambar 3.38 Notasi *Class* Halaman *Manage_Mutasi*.

F. *Class* Halaman *Manage Kontrak*

Class `Halaman_Manage_Kontrak` digunakan untuk mengelola data kontrak kerja. Melalui halaman ini bagian HRD dapat menambahkan data kontrak kerja baru, mengubah data kontrak kerja, serta menghapus data kontrak kerja pegawai di dalam *database*.



Gambar 3.39 Notasi *Class* Halaman *Manage_Kontrak*

G. *Class* Halaman *Manage_Training*

Class Halaman *Manage_Training* digunakan untuk mengelola *training* yang telah dilakukan pegawai. Melalui halaman ini bagian HRD dapat menambahkan data *training* baru dan mengubah data *training*.



Gambar 3.40 Notasi *Class* Halaman_Manage_Training

H. Class Halaman_Absensi

Class Halaman_Absensi digunakan untuk mengelola absensi yang dilakukan pegawai. Melalui halaman ini pegawai dapat menginputkan waktu masuk kerja dan pulang kerja sesuai dengan *shift* yang telah ditentukan sebelumnya.

Gambar 3.41 Notasi *Class* Halaman_Absensi

I. *Class* Halaman Outsourcing

Class halaman *outsourcing* digunakan untuk mencatat perusahaan *vendor* yang melakukan kerja sama dengan PKIS Sekar Tanjung. Halaman *outsourcing* juga digunakan untuk mengetahui perusahaan *vendor* tersebut memberikan pegawai untuk PKIS Sekar Tanjung.

Gambar 3.42 Notasi *Class* Halaman_Outsourcing

J. *Class* Halaman_Penilaian Kinerja

Class Halaman_Penilaian kinerja digunakan untuk mencatat penilaian kerja yang dilakukan pegawai. Notasi *class* Halaman_penilaian kinerja dapat dilihat pada gambar 3.43.



Gambar 3.43 Notasi *Class* Halaman_penilaian kinerja

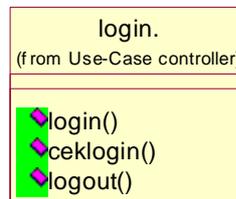
3. *Class Controller*

Class controller bertanggung jawab untuk mengkoordinasikan kegiatan-kegiatan terhadap *class* lainnya. *Class* ini bersifat *optional*, tetapi jika *class controller* ini diputuskan untuk digunakan dalam sistem, maka lazimnya satu *class controller* untuk satu *use case*. *Class controller* digunakan untuk mengatur urutan kejadian dalam *use case* tersebut. *Class controller* pada sistem informasi kepegawaian ini yaitu:

A. *Class Login*

Class login digunakan untuk mengatur jalannya *login* yang dilakukan oleh pegawai HRD pada PKIS Sekar Tanjung. *Username* dan *password* yang dimasukkan oleh pengguna *web* dikirimkan ke fungsi *cekLogin*. Kemudian fungsi tersebut memvalidasi masukkan pengguna. Apabila masukkan yang dimasukkan

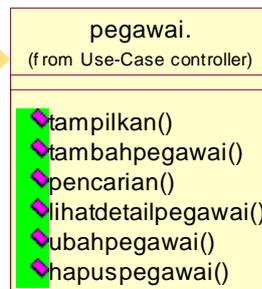
sesuai maka pengguna dapat masuk ke dalam sistem informasi kepegawaian dan menjalankan sesuai fungsinya.



Gambar 3.44 Notasi *Class Login*

B. *Class Master Pegawai*

Class master pegawai berfungsi untuk mengolah data pegawai. Melalui *class* ini HRD dapat menambah, mengubah, pencarian, melihat detail pegawai, dan menghapus data pegawai. *Class* pegawai ini memiliki operasi-operasi antara lain tampilkan, tambahPegawai, ubahPegawai, lihatDetailPegawai, HapusPegawai.

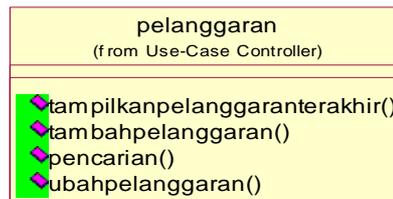


Gambar 3.45 Notasi *Class Master Pegawai*

C. *Class Pelanggaran*

Class pelanggaran berfungsi untuk mengolah data pelanggaran. Melalui *class* ini HRD dapat menambah, mengubah, pencarian data, dan melihat detail

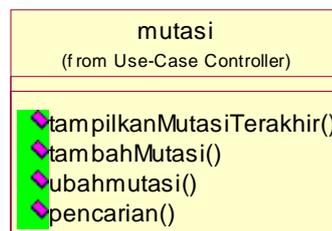
pelanggaran pegawai. Fungsi tampilkan data pelanggaran akhir adalah untuk menampilkan data pelanggaran yang terakhir diinputkan oleh pengguna. *Class* pelanggaran ini memiliki operasi-operasi antara lain tampilkanpelanggaran, tambahpelanggaran, ubahpelanggaran, lihatDetailpelanggaran.



Gambar 3.46 Notasi *Class* Pelanggaran

D. *Class* Mutasi

Class mutasi berfungsi untuk mengolah data mutasi. Melalui *class* ini HRD dapat menambah, mengubah, pencarian data, dan melihat mutasi pegawai. Fungsi tampilkan data mutasi adalah untuk menampilkan data mutasi terakhir diinputkan oleh pengguna. *Class* mutasi ini memiliki operasi-operasi antara lain tampilkan, tambahmutasi, ubahmutasi.



Gambar 3.47 Notasi *Class* Mutasi

E. *Class* Training

Class training berfungsi untuk mengolah data *training*. Melalui *class* ini HRD dapat menambah, mengubah, pencarian data, dan melihat detail *training* pegawai. Fungsi tampilkan data *training* adalah untuk menampilkan data *training* yang terakhir diinputkan oleh pengguna. *Class training* ini memiliki operasi-operasi antara lain tampilkan, tambahTraining, ubahTraining.



Gambar 3.48 Notasi *Class Training*

F. *Class Kontrak Kerja*

Class kontrak kerja berfungsi untuk mengatur pengolah data kontrak kerja. Melalui *class* ini HRD dapat menambah, mengubah, pencarian data, dan melihat detail kontrak kerja pegawai. Fungsi tampilkan data kontrak akhir adalah untuk menampilkan data kontrak kerja yang terakhir diinputkan oleh pengguna, sedangkan tampilan akhir pegawai kontrak memiliki fungsi menampilkan pegawai yang akan berakhir masa kerja. *Class* kontrak kerja ini memiliki operasi-operasi antara lain tambahkontrak, ubahkontrak.

Gambar 3.49 Notasi *Class* Kontrak Kerja

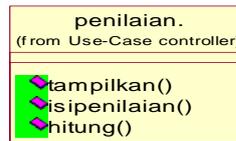
G. *Class* Absensi

Class login digunakan untuk mengatur jalannya absensi yang dilakukan oleh pegawai pada PKIS Sekar Tanjung. *Username* dan *password* yang dimasukkan oleh pengguna *web* dikirimkan ke fungsi *cekAbsensi*. Kemudian fungsi tersebut memvalidasi masukkan pengguna. Apabila masukkan yang dimasukkan sesuai maka pengguna telah melakukan absen dan mendapat pemberitahuan absen berhasil.

Gambar 3.50 Notasi *Class* Absensi

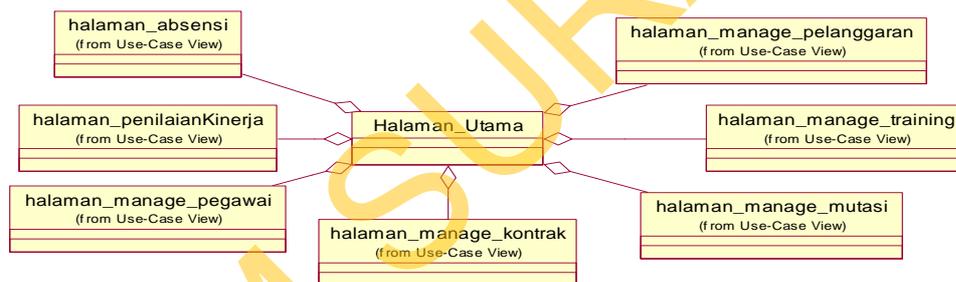
H. *Class* Penilaian Kinerja

Class penilaian kinerja untuk mengatur jalannya penilaian yang dilakukan oleh *supervisor*. Mulai *class* ini *supervisor* dapat memasukkan penilaian untuk bawahannya. *Class* penilaian kinerja ini memiliki operasi-operasi antara lain isipenilaian, hitungpenilaian.

Gambar 3.51 Notasi *Class* Penilaian Kinerja

4. Relasi *Class* Diagram Tampilan

Relasi *class* diagram tampilan menjelaskan tentang relasi antar *class view*. Pada gambar tersebut dijelaskan keterkaitan antara *class view* yang satu dengan *class view* yang lain di dalam *website* sistem informasi kepegawaian.

Gambar 3.52 Relasi *Class* Diagram Tampilan

5. Relasi *Class* Diagram Proses Dan Entitas

Relasi *class* diagram proses dan entitas menjelaskan tentang interaksi antara proses-proses yang terjadi di dalam sistem terhadap entitas tertentu. Entitas menggambarkan tabel-tabel yang di dalam *database*. Berikut ini adalah diagram-diagram yang menggambarkan relasi antar proses dan entitas yang terjadi di dalam sistem.

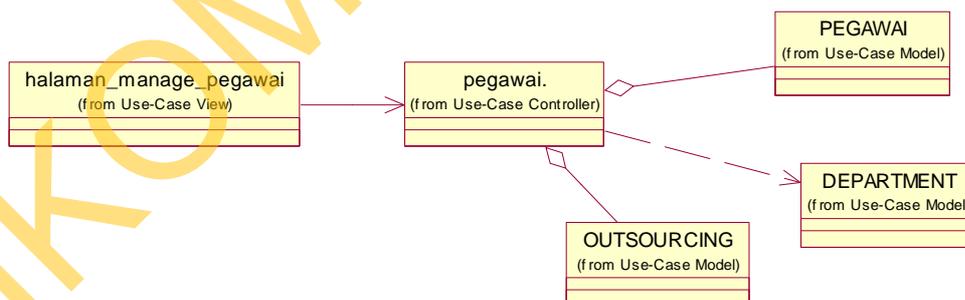
Pada gambar 3.53 dijelaskan bahwa *class* HalLogin akan memanggil fungsi yang ada pada *class* login dengan parameter *username* dan *password*.

Kemudian *class login* meminta *password* yang ada di dalam *class pegawai* sesuai dengan *username* yang diberikan. *Password* yang didapat dari *class pegawai* kemudian dicocokkan dengan *password* yang dimasukkan oleh HRD.



Gambar 3.53 Relasi Class Diagram Login

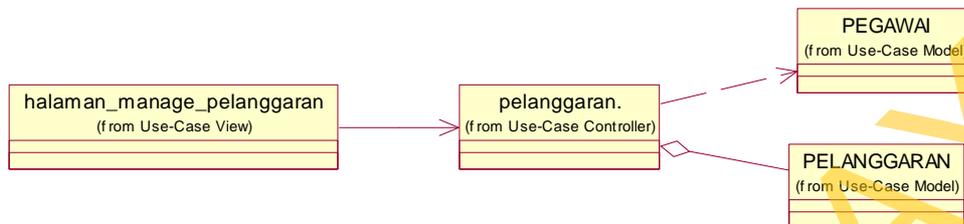
Gambar 3.54 menjelaskan tentang *class* yang terlihat pada proses *manage pegawai*. *Class* yang terlibat adalah *Halaman_manage_pegawai*, *class controller pegawai*, *pegawai*, *department*. *Class controller* artikel menggunakan seluruh atribut yang ada pada *class pegawai* sehingga hubungan antar *class* tersebut adalah agregasi.



Gambar 3.54 Relasi Class Diagram Pegawai

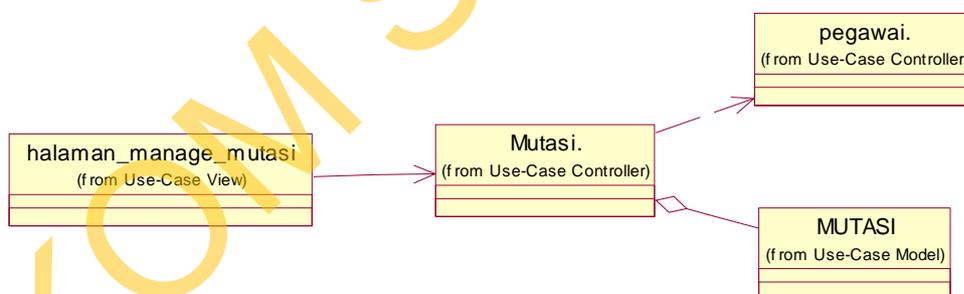
Pada gambar 3.55 menjelaskan tentang hubungan antara *class model*, *class view* dan *controller*. Entitas-entitas yang dilibatkan pada proses pelanggaran kerja

adalah Halaman_manage_pelanggaran, *class controller* pelanggaran, pelanggaran dan pegawai.



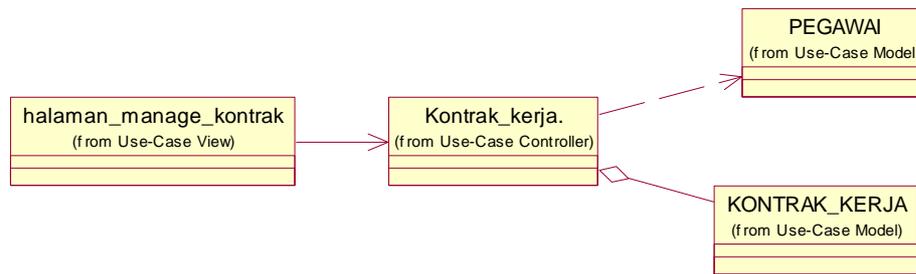
Gambar 3.55 Relasi Class Diagram Pelanggaran

Pada gambar 3.56 menjelaskan bahwa *class* yang terlibat antara lain halaman_manage_mutasi, *class controller* mutasi, pegawai dan mutasi. Seluruh atribut yang ada pada *class* model mutasi digunakan pada *class controller* mutasi sehingga hubungan antara *class* tersebut adalah agregasi.



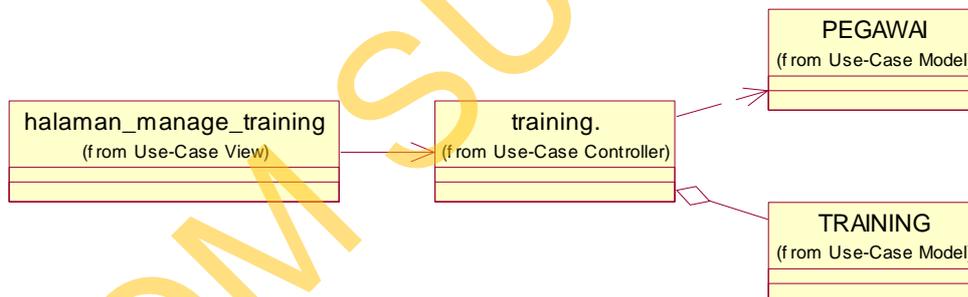
Gambar 3.56 Relasi Class Diagram Mutasi

Pada gambar 3.57 menjelaskan tentang *class* yang terlibat dalam proses menyimpan data. *Class* yang terlibat adalah halaman_manage_kontrakkerja, *class controller* kontrak, pegawai dan kontrak kerja. *Class controller* kontrak kerja menggunakan seluruh atribut yang ada pada *class* kontrak kerja sehingga hubungan antar *class* tersebut adalah agregasi.



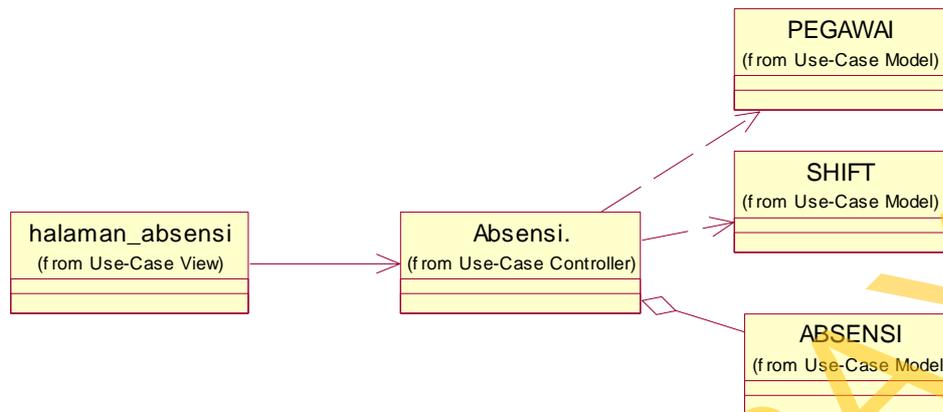
Gambar 3.57 Relasi *Class Diagram* Kontrak Kerja

Gambar 3.58 menjelaskan tentang *class* yang terlibat pada proses *manage training*. *Class* yang terlibat adalah *halaman_manage_training*, *class controller training*, *Pegawai*, *training*. *Class controller training* menggunakan seluruh atribut yang ada pada *class training* sehingga hubungan antar *class* tersebut adalah agregasi.



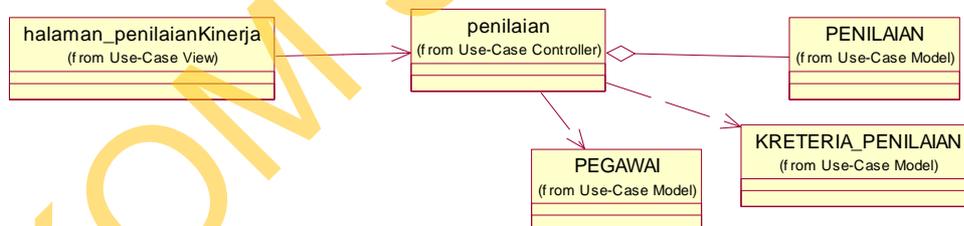
Gambar 3.58 Relasi *Class Diagram* Training

Gambar 3.59 menjelaskan tentang *class* yang terlibat di dalam proses absensi. *Class* yang terlibat antara lain: *halaman_absensi*, *class controller absensi*, *pegawai*, *shift*, dan *absensi*. *Class Absensi* dibutuhkan, karena melalui halaman ini pegawai dapat melakukan absen masuk maupun pulang untuk mengetahui ketepatan waktu kehadiran.



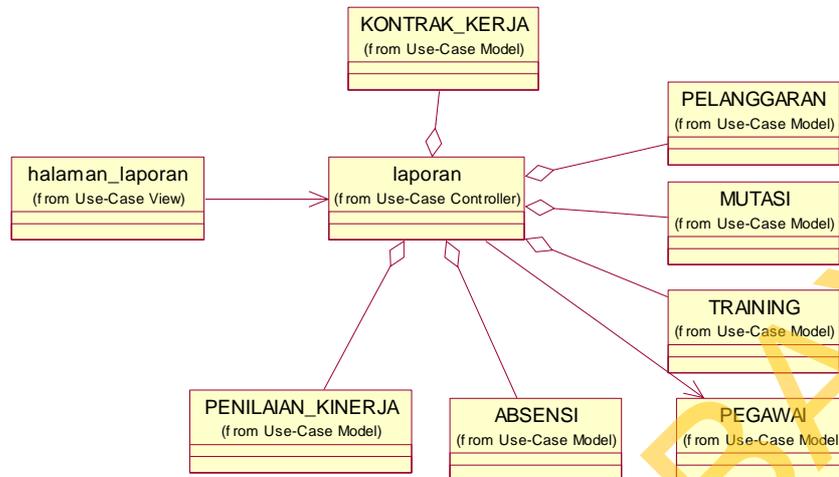
Gambar 3.59 Relasi *Class Diagram* Absensi

Gambar 3.60 menjelaskan tentang *class* yang terlibat di dalam proses penilaian kinerja. *Class* yang terlibat antara lain: halaman_penilaian, *controller* penilaian, pegawai, kriteria. *Class* ini dibutuhkan karena melalui halaman ini dapat menilai kinerja pegawai.



Gambar 3.60 Relasi *Class Diagram* Penilaian Kinerja

Class diagram selanjutnya adalah untuk proses membuat laporan. Pada gambar 3.61 merupakan tahap membuat laporan *class model* yang terlibat adalah *class* pelanggaran, *class* pegawai, *class* mutasi, *class* kontrak kerja, *class* training, *class* absensi dan *class* penilaian kinerja. Seluruh atribut yang ada pada *class model* penilaian digunakan pada *class controller* penilaian sehingga hubungan antara *class* tersebut adalah dependensi.

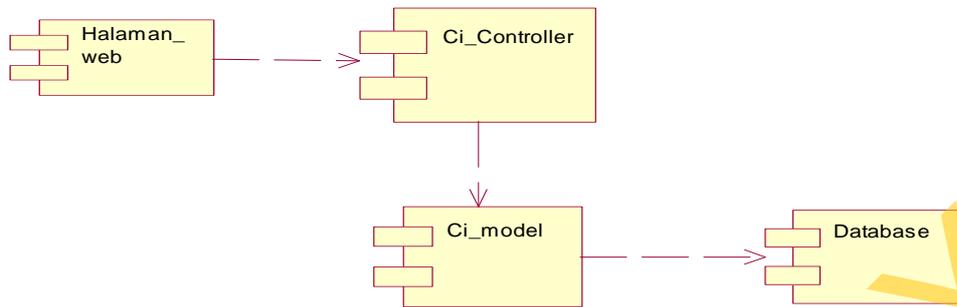


Gambar 3.61 Relasi *Class Diagram* Laporan

3.2.6 Diagram Komponen

Menurut Sholiq (2010) diagram komponen adalah modul fisik dari kode, komponen bisa mencantumkan pustaka kode program dan berkas-berkas *runtime* sekaligus. Garis-garis yang menghubungkan antara komponen yang satu dengan komponen yang lainnya menunjukkan dependensi komponen tersebut. Setiap komponen diperlukan agar program yang dibangun dapat berjalan dengan baik.

Halaman *web* berkomunikasi secara langsung dengan *server* melalui *controller*. Komponen *controller* dengan mendapatkan atau menyimpan data dalam *database* melalui model. Sedangkan untuk dapat mengakses data yang ada di dalam *database* membutuhkan komponen model yang secara langsung berkomunikasi dengan *database*. Berikut ini adalah komponen-komponen yang ada pada *website* sistem informasi kepegawaian.



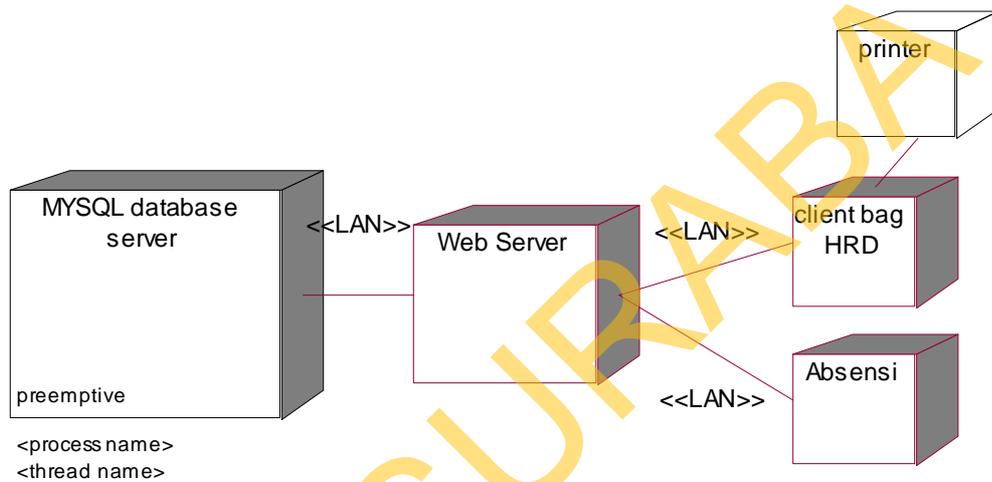
Gambar 3.62 Komponen Diagram Sistem Informasi Kepegawaian

Gambar di atas menjelaskan bahwa komponen *halaman web* bergantung kepada *class controller*. *Class controller* bergantung kepada *class model* dalam meminta data yang diperlukan dari dalam *database*. *Class model* mengirimkan *queri* yang bertanggung jawab dalam komunikasi dengan *database*.

3.2.7 Diagram Deployment

Diagram *deployment* menampilkan *layout* fisik aplikasi. Diagram ini membantu tim pengembang untuk merencanakan *deployment* yang akan ditawarkan. Diagram *deployment* berisi prosesor-prosesor, peralatan-peralatan, proses-proses dan hubungan antara prosesor atau peralatan tersebut. Gambar 3.63 menyajikan diagram *deployment* untuk sistem informasi kepegawaian pada PKIS Sekar Tanjung.

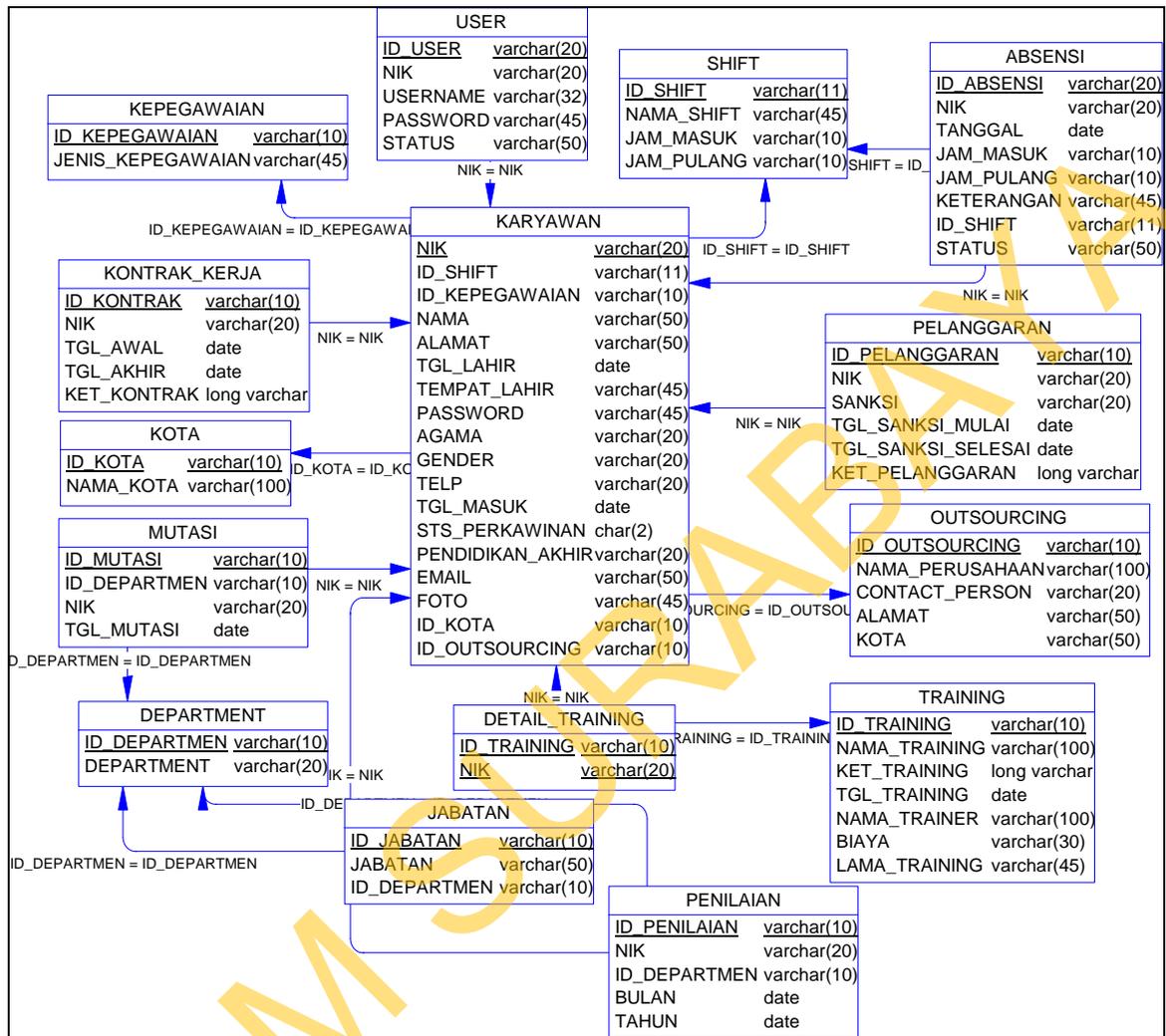
Aktor bagian HRD mengakses sistem informasi kepegawaian menggunakan aplikasi *web* yang terkoneksi menggunakan jaringan *local area network* (LAN). Basis data yang digunakan oleh kedua *client* adalah basis data yang sama yang tersimpan pada *database server*.



Gambar 3.63 *Deployment Diagram* Sistem Informasi kepegawaian

3.2.8 *Physical Data Model*

Physical data model menggambarkan relasi *database* yang berada pada sistem. Desain *database* ini didapatkan dari struktur *database* yang sudah ada di perusahaan dengan sistem lama, sehingga untuk sistem yang baru dapat mengambil struktur *database* yang sama. Desain *database* dari sistem informasi kepegawaian di PKIS Sekar Tanjung dapat dilihat pada gambar 3.64 .

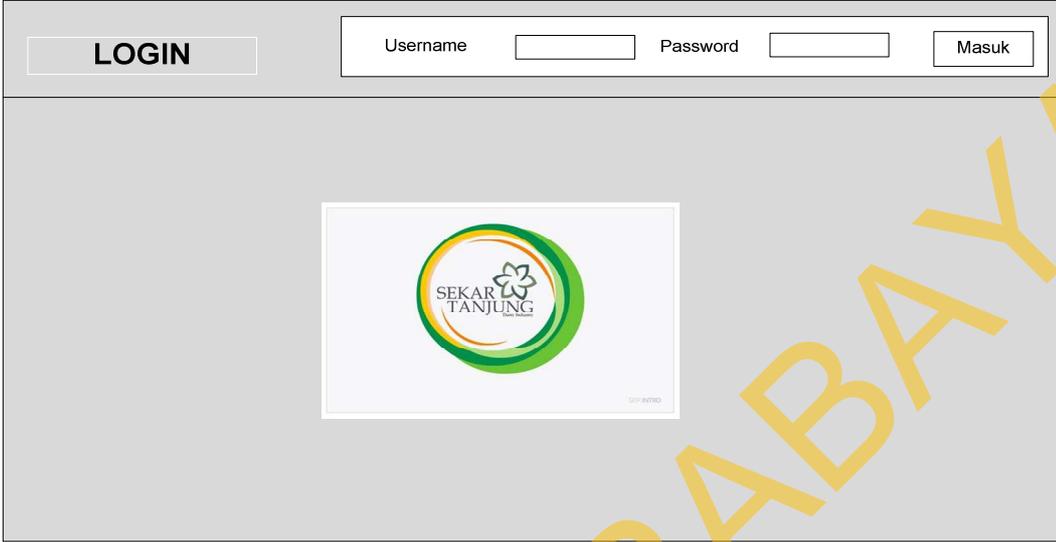


Gambar 3.64 Physical Data Model Database Sistem Informasi Kepegawaian

3.2.9 Desain Input/Output

Setelah melakukan perancangan struktur basis data, tahap selanjutnya adalah membuat desain *input/output*. Desain ini berguna untuk menggambarkan interaksi antara pengguna pada aplikasi yang dibuat. Dari hasil analisis yang telah dilakukan sebelumnya, aplikasi yang dibuat ini akan digunakan oleh HRD PKIS Sekar Tanjung.

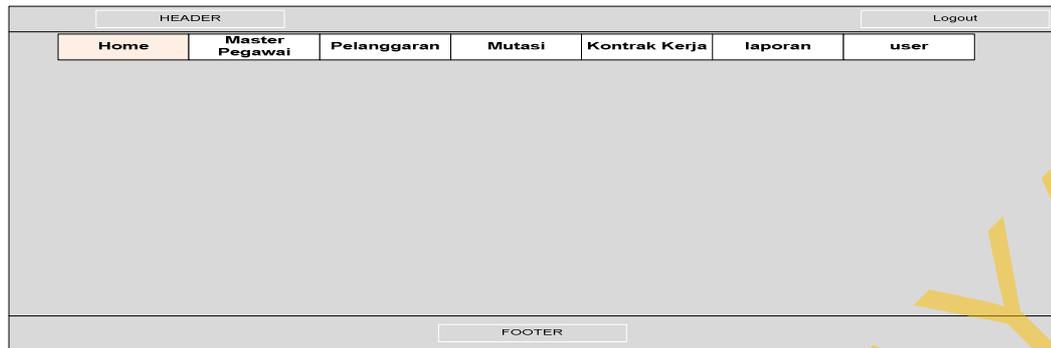
1. Tampilan Halaman *login*



Gambar 3.65 Halaman *Login*

Pada gambar 3.65 menampilkan halaman *login* untuk bagian HRD di Sekar Tanjung untuk masuk ke dalam sistem informasi kepegawaian. *Textbox username* diisi sesuai dengan *username* dan *textbox password* diisi sesuai dengan *password* yang diberi hak akses untuk membuka halaman utama. Pada *form* terdapat *button masuk* yang berfungsi untuk mengoneksikan aplikasi dengan *database* dan melakukan *verifikasi password*, apabila *username* dan *password* sesuai maka sistem akan mengarahkan pada menu utama mengelola data pegawai.

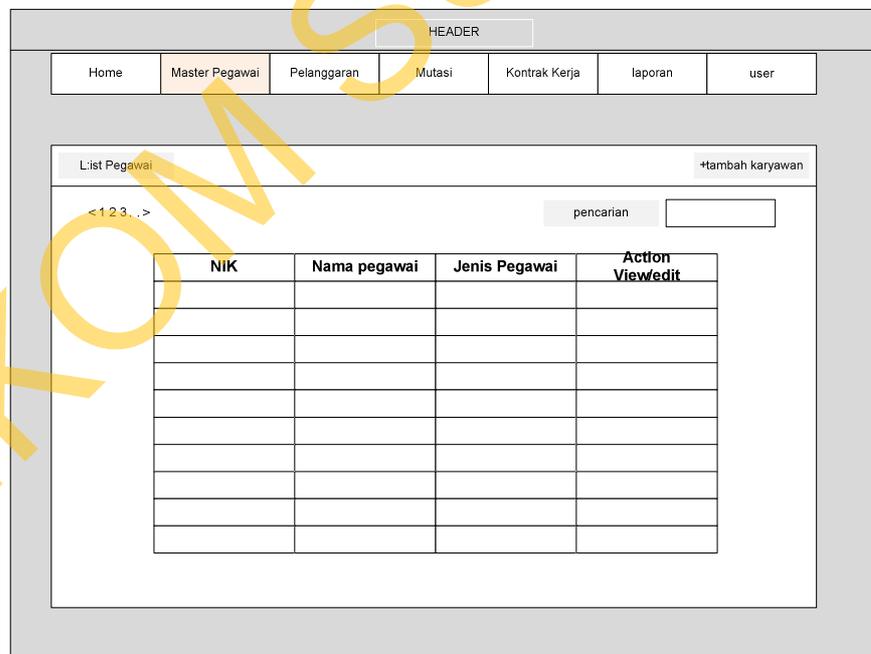
2. Tampilan Halaman Utama



Gambar 3.66 Halaman Utama

Pada halaman utama terlihat pada gambar 3.66 terdapat beberapa tautan yang dapat diakses untuk melakukan proses mengolah data pegawai antara lain master pegawai, pelanggaran, mutasi, *training*, kontrak kerja laporan dan *logout*.

3. Tampilan Halaman Master Pegawai

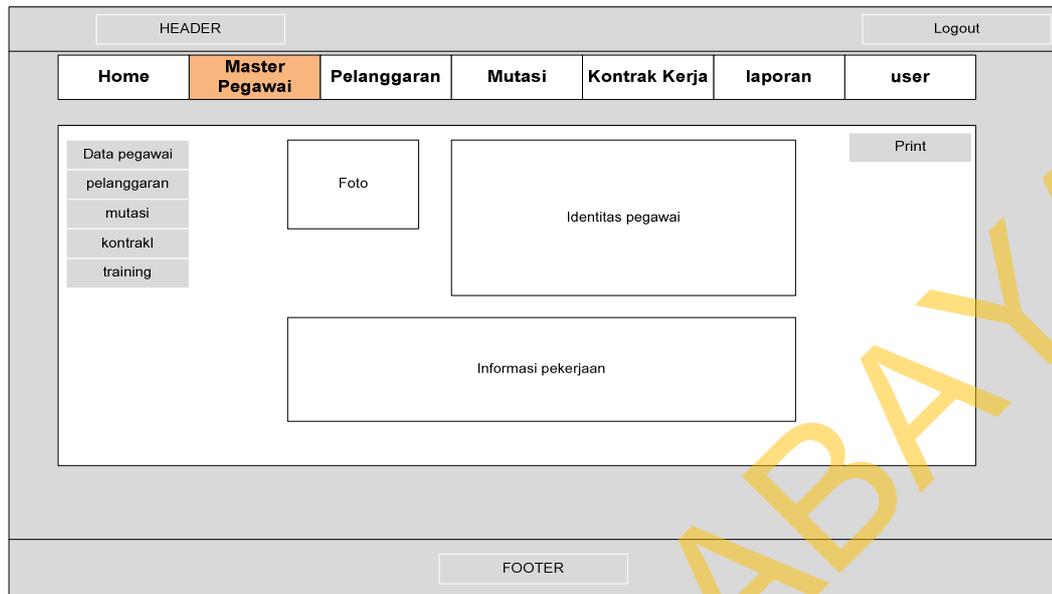


Gambar 3.67 Halaman Master Pegawai

Pada halaman master pegawai pada gambar 3.67 ini merupakan halaman untuk memasukkan data pegawai baru atau melihat detail pegawai. HRD dapat menambah data pegawai baru dengan menekan tombol (+) tambah pegawai yang ada pada kanan atas. Sedangkan untuk melihat detail pegawai dan mengedit data pegawai klik tautan *view* dan *edit* yang ada pada kanan nama pegawai, untuk lebih jelasnya dapat dilihat pada gambar 3.68 dan 3.69.

The image shows a web application interface for adding a new employee. At the top, there is a 'HEADER' section with a navigation menu containing 'Home', 'Master Pegawai' (which is highlighted), 'Pelanggaran', 'Mutasi', 'Kontrak Kerja', 'laporan', and 'user'. Below the header is a main content area titled 'Tambah pegawai baru'. This area contains a form with the following fields: 'Nik', 'Tgl Lahir', 'Agama', 'Nama', 'Pendidikan akhir', 'Jenis pegawai', and 'Alamat'. At the bottom of the form are two buttons: 'simpan' and 'kembali'. The entire form is enclosed in a 'FOOTER' section.

Gambar 3.68 Halaman *Input* Data Pegawai Baru



Gambar 3.69 Halaman View Detail Data Pegawai Dan Pekerjaan

4. Tampilan Halaman Pelanggaran

Halaman ini berfungsi untuk mengelola data pelanggaran yang dilakukan pegawai saat bekerja di PKIS Sekar Tanjung. HRD dapat memasukkan data-data pelanggaran setelah mendapatkan data pelanggaran itu dari *supervisor* masing-masing bagian. Halaman ini HRD dapat menampilkan data pelanggaran yang ada pada *database*. HRD dapat melakukan pencarian pelanggaran yang dilakukan pegawai dan juga dapat merubah data pelanggaran tersebut, untuk lebih jelasnya lihat gambar 3.70 dan 3.71.

ID pelanggaran	Nama pegawai	Ket. Pelanggaran	Action View/edit

Gambar 3.70 Halaman Pelanggaran

Gambar 3.71 Halaman *Input* Data Pelanggaran

5. Tampilan Halaman Mutasi

Halaman ini berfungsi untuk mengelola data perpindahan bagian atau jabatan yang dilakukan pegawai. Pada halaman ini HRD dapat menampilkan data mutasi pegawai sebelum dan sesudah dipindahkan. HRD dapat melihat dan

melakukan pencarian data mutasi pegawai dan juga dapat merubah data mutasi tersebut, untuk lebih jelasnya lihat gambar 3.72 dan 3.73 dibawah ini.

The screenshot shows a web application interface for managing transfers. At the top, there is a navigation menu with options: Home, Master Pegawai, Pelanggaran, Mutasi (highlighted), Kontrak Kerja, laporan, and user. Below the menu, there is a section titled 'List Mutasi' with a '+tambah Mutasi' button. A search bar labeled 'pencarian' is present. Below the search bar is a table with the following columns: ID Mutasi, Nama pegawai, Bagian, Jabatan, and Action View/edit. The table is currently empty.

ID Mutasi	Nama pegawai	Bagian	Jabatan	Action View/edit

Gambar 3.72 Halaman Mutasi

The screenshot shows the 'Tambah Mutasi' (Add New Transfer) form. It features a navigation menu at the top with options: Home, Master Pegawai, Pelanggaran, Mutasi (highlighted), Kontrak Kerja, laporan, and user. The form contains the following fields and buttons:

- Id mutasi:
- Departemen tujuan:
- Nik:
- jabatan:
- Tgl mutasi:
- simpan:
- kembali:

At the bottom of the page, there is a 'FOOTER' section.

Gambar 3.73 Halaman *Input* Data Mutasi Baru

6. Tampilan Halaman *Training*

Halaman *training* ini merupakan halaman yang digunakan HRD untuk mengelola sebuah data *training*. Setelah HRD membuat *event training* baru maka HRD dapat memasukkan nama-nama pegawai yang mengikuti *training* tersebut. Melalui halaman ini HRD dapat merubah data *training* tersebut. untuk lebih jelasnya lihat gambar 3.74 dan 3.75.

The screenshot shows a web application interface for adding training data. The header includes a 'Logout' button and a navigation menu with 'Home', 'Master Pegawai', 'Pelanggaran', 'Mutasi', 'Training' (selected), 'Kontrak Kerja', 'laporan', and 'user'. The main content area is titled 'Tambah training' and contains several input fields: 'Id_training', 'Tgl_training', 'Lama_training', 'Event training', 'Nama trainer', 'Biaya training', and 'Ket_training'. Below these is a section for 'Peserta training' with a '+tambah Mutasi' button and a search box labeled 'pencarian'. A table with columns 'Nik', 'Nama Peserta', and 'Action View/edit' is present, with two empty rows. At the bottom of the form are 'simpan' and 'kembali' buttons. The footer contains the word 'FOOTER'.

Gambar 3.74 Halaman *Input Tambah Training*

HEADER

Logout

Home Master Pegawai Pelanggaran Mutasi Training Kontrak Kerja laporan user

List pegawai

Silahkan masukkan pegawai yang diikuti dalam training pencarian

Nik	Nama Peserta	Pilih karwayan
		v
		v

simpan kembali

FOOTER

Gambar 3.75 Halaman Tambah Data Pegawai Pada Training

7. Tampilan Halaman Absensi

Halaman absensi ini digunakan pegawai untuk mengetahui kehadiran waktu pegawai tersebut datang maupun pulang. Pegawai dapat absen dengan memasukkan *username* dan *password* sesuai dengan NIK dan *password* yang mereka miliki ke dalam sistem, gambar 3.76 akan menjelaskan *form* absensi yang nantinya digunakan untuk *input* data absensi.

The image shows a web interface for entering absence data. At the top, there is a grey header bar with the word 'HEADER' centered. Below the header, there are two buttons: 'Home' and 'Absen'. The main content area is a white box with a grey border, titled 'Absensi' in the top left corner. Inside this box, there are two input fields. The first is labeled 'nik' and the second is labeled 'password'. Below these input fields are two buttons: 'simpan' and 'kembali'. At the bottom of the page, there is a grey footer bar with the word 'FOOTER' centered.

Gambar 3.76 Halaman *Input Absensi*

8. Tampilan Halaman Laporan *Profile Pegawai*

Pada halaman 3.77 merupakan halaman laporan *profile* pegawai, laporan ini berisikan data pelanggaran, data mutasi, data *training*. Laporan ini digunakan untuk melihat dengan detail data pegawai.

Gambar 3.77 Laporan Kepegawaian

3.2.10 Desain Uji Coba

Aplikasi yang dirancang dan diimplementasikan harus diuji coba untuk mengetahui tingkat keberhasilan dari pemakai aplikasi tersebut. Untuk melakukan uji coba pada aplikasi yang telah dibuat maka digunakan beberapa data transaksi yang dimasukkan sebagai inputan. Data transaksi yang dimasukkan bertujuan untuk menganalisa hasil *output* yang ditampilkan oleh sistem, apakah telah sesuai dengan tujuan pembuatan sistem informasi kepegawaian atau masih memerlukan perbaikan.

Untuk aplikasi sistem informasi kepegawaian, pengujian akan dilakukan dengan pendekatan metode *black box testing*. Pengujian *black box* adalah pengujian aspek *fundamental* sistem tanpa memperhatikan struktur logika internal

perangkat lunak. Metode ini digunakan karena aplikasi sistem informasi kepegawaian memiliki beberapa fungsi yang berjalan secara otomatis dibalik layar. Pengujian *black box* merupakan metode perancangan uji coba yang didasarkan pada spesifikasi perangkat lunak yang dibuat. Adapun hal-hal yang akan disajikan adalah sebagai berikut.

Tabel 3.10 Rencana Pengujian Sistem Informasi kepegawaian

Requirement yang diuji	Fungsi yang diuji
<i>Login</i>	Melakukan <i>login</i>
Halaman master pegawai	Mengelola data pegawai
Halaman pelanggaran	Mengelola data pelanggaran
Halaman mutasi	Mengelola data mutasi
Halaman kontrak	Mengelola data kontrak
Halaman <i>training</i> kerja	Mengelola data <i>training</i> kerja
Halaman penilaian kerja	Mengelola data hasil penilaian
Halaman absensi	Memasukkan data absensi
Halaman jabatan & bagian	Mengelola bagian dan jabatan
Halaman laporan	Menampilkan laporan kepegawaian

1. Desain Uji Coba *Login*

Pada desain uji coba proses *login* bertujuan untuk mengetahui apakah proses *login* dapat berfungsi dengan baik. Uji coba ini dilakukan dengan memasukkan *username* dan *password* yang valid dan tidak valid. Desain uji coba ini dijelaskan pada tabel 3.11.

Tabel 3.11 Desain Uji Coba Melakukan *Login* Pada Sistem

<i>Test case ID</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
1	Pengujian validasi <i>username</i> dan <i>password</i> untuk jenis pengguna HRD	<i>Username</i> = admin; <i>Password</i> =admin, kemudian klik tombol <i>login</i>	Sistem langsung mengarahkan kepada halaman awal dengan menu-menu yang disediakan untuk mengelola kepegawaian	
2	Pengujian terhadap <i>input username</i> dan <i>password</i> yang salah	<i>Username</i> = admin; <i>Password</i> =qwerty, kemudian klik tombol <i>login</i>	Terdapat pesan kesalahan di bawah <i>form login</i>	

2. Desain Uji Coba Halaman Master Pegawai

Halaman master pegawai perlu diuji untuk mengetahui apakah halaman tersebut dapat berfungsi dengan baik. Pengujian halaman tersebut dilakukan dengan menggunakan tombol yang telah disediakan. Tabel 3.12 berikut ini menjelaskan uji coba yang akan dilakukan pada halaman master pegawai:

Tabel 3.12 Desain Uji Coba Halaman Master pegawai

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
3	Pengujian terhadap fungsi tautan judul master pegawai	Klik pada tautan judul master pegawai	Sistem mengarahkan pengguna kepada halaman master pegawai yang dimaksud.	

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
4	Menguji fungsi tautan tambah master pegawai	Klik pada tautan tambah pegawai	Sistem mengarahkan pengguna kepada halaman tambah pegawai yang dimaksud	
5	Menguji fungsi tautan <i>edit</i> master pegawai	Klik pada tautan <i>edit</i> pegawai	Sistem mengarahkan pengguna kepada halaman <i>edit</i> pegawai yang dimaksud	
6	Menguji hasil pencarian data pegawai yang telah ada	Nama pegawai=budi, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Tabel pegawai hasil pencarian yang berisi nama dan jenis pegawai	
7	Menguji hasil pencarian data pegawai yang tidak ada	Nama pegawai=qwerty, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Pesan bahwa data pegawai tidak ditemukan	
8	Menguji menampilkan riwayat pegawai lengkap	Klik <i>view</i> setelah nama pegawai	Sistem menampilkan data pegawai yang dimaksud	

3. Desain Uji Coba Halaman Pelanggaran

Halaman pelanggaran perlu diuji untuk mengetahui apakah halaman tersebut dapat berfungsi dengan baik. Pengujian halaman tersebut dilakukan dengan menggunakan tombol yang telah disediakan. Tabel 3.13 berikut ini menjelaskan uji coba yang akan dilakukan pada halaman pelanggaran:

Tabel 3.13 Desain Uji Coba Halaman Pelanggaran

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
9	Pengujian terhadap fungsi tautan judul pelanggaran	Klik pada tautan judul pelanggaran	Sistem mengarahkan pengguna kepada halaman pelanggaran yang dimaksud.	
10	Menguji fungsi tautan tambah pelanggaran	Klik pada tautan tambah pelanggaran	Sistem mengarahkan pengguna kepada halaman tambah pelanggaran yang dimaksud	
11	Menguji fungsi tautan <i>edit</i> pelanggaran	Klik pada tautan <i>edit</i> pelanggaran	Sistem mengarahkan pengguna kepada halaman <i>edit</i> pelanggaran yang dimaksud	
12	Menguji hasil pencarian data pelanggaran yang telah ada	Nama pegawai= budi, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Tabel pelanggaran hasil pencarian yang berisi nama dan keterangan pelanggaran	
13	Menguji hasil pencarian data	Nama pegawai=qwerty,	Pesan bahwa data pelanggaran tidak	

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
	pegawai yang tidak ada	kemudian tunggu pencarian sampai data keluar dengan sendirinya	ditemukan	

4. Desain Uji Coba Halaman Mutasi

Uji coba ini dilakukan untuk mengetahui apakah fungsi-fungsi yang terdapat pada halaman mutasi dapat digunakan dengan baik. Uji coba dilakukan dengan menginputkan beberapa data mutasi baru. Uji coba halaman mutasi dijelaskan pada tabel 3.14.

Tabel 3.14 Desain Uji Coba Halaman Mutasi

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
14	Pengujian terhadap fungsi tautan judul mutasi	Klik pada tautan judul mutasi	Sistem mengarahkan pengguna kepada halaman mutasi yang dimaksud.	
15	Menguji fungsi tautan tambah mutasi	Klik pada tautan tambah mutasi	Sistem mengarahkan pengguna kepada halaman tambah mutasi yang	

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
			dimaksud	
16	Menguji fungsi tautan <i>edit</i> mutasi	Klik pada tautan <i>edit</i> mutasi	Sistem mengarahkan pengguna kepada halaman <i>edit</i> mutasi yang dimaksud	
17	Menguji hasil pencarian data pegawai yang telah ada	Nama pegawai= budi, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Tabel pegawai hasil pencarian yang berisi nama, bagian dan jabatan	
18	Menguji hasil pencarian data pegawai yang tidak ada	Nama pegawai=qwerty, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Pesan bahwa data pegawai tidak ditemukan	
19	Menguji hasil <i>update</i> mutasi baru yang telah dilakukan pada master pegawai	Memasukkan jabatan yang baru pada <i>form</i> mutasi, kemudian tekan simpan	Sistem akan mengupdate master pegawai jabatan dan bagian yang baru sesuai mutasi yang telah dilakukan	

5. Desain Uji Coba Halaman Kontrak Kerja

Halaman kontrak kerja perlu diuji untuk mengetahui apakah halaman tersebut dapat berfungsi dengan baik. Pengujian halaman tersebut dilakukan

dengan menggunakan tombol yang telah disediakan. Tabel 3.15 berikut ini menjelaskan uji coba yang akan dilakukan pada halaman kontrak kerja.

Tabel 3.15 Desain Uji Coba Halaman Kontrak Kerja

<i>Test case id</i>	Tujuan	<i>Input</i>	Output yang diharapkan	Status
20	Pengujian terhadap fungsi tautan judul kontrak kerja	Klik pada tautan judul kontrak kerja	Sistem mengarahkan pengguna kepada halaman kontrak kerja yang dimaksud.	
21	Menguji fungsi tautan tambah kontrak kerja	Klik pada tautan tambah kontrak kerja	Sistem mengarahkan pengguna kepada halaman kontrak kerja yang dimaksud	
22	Menguji fungsi tautan <i>edit</i> kontrak kerja	Klik pada tautan <i>edit</i> kontrak kerja	Sistem mengarahkan pengguna kepada halaman kontrak kerja yang dimaksud	
23	Menguji hasil pencarian data pegawai yang telah ada	Nama pegawai= budi, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Tabel pegawai hasil pencarian yang berisi nama dan tanggal akhir kontrak kerja	
24	Menguji hasil pencarian data pegawai yang tidak ada	Nama pegawai=qwerty, kemudian tunggu pencarian sampai data keluar dengan	Pesan bahwa data pegawai tidak ditemukan	

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
		sendirinya		

6. Desain Uji Coba Halaman *Training*

Halaman *training* perlu diuji untuk mengetahui apakah halaman tersebut dapat berfungsi dengan baik. Pengujian halaman tersebut dilakukan dengan menggunakan tombol yang telah disediakan. Tabel 3.16 berikut ini menjelaskan uji coba yang akan dilakukan pada halaman *training*.

Tabel 3.16 Desain Uji Coba Halaman *Training*

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
25	Pengujian terhadap fungsi tautan judul <i>training</i>	Klik pada tautan judul <i>training</i>	Sistem mengarahkan pengguna kepada halaman <i>training</i> yang dimaksud.	
26	Menguji fungsi tautan tambah <i>training</i>	Klik pada tautan tambah <i>training</i>	Sistem mengarahkan pengguna kepada halaman tambah <i>training</i> yang dimaksud	
27	Menguji fungsi tautan <i>edit training</i>	Klik pada tautan <i>edit training</i>	Sistem mengarahkan pengguna kepada halaman <i>edit training</i> yang	

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
			dimaksud	
28	Menguji hasil pencarian data pegawai yang telah ada	Nama pegawai= budi, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Tabel pegawai hasil pencarian yang berisi nama dan jenis pegawai	
29	Menguji hasil pencarian data pegawai yang tidak ada	Nama pegawai=qwerty, kemudian tunggu pencarian sampai data keluar dengan sendirinya	Pesan bahwa data pegawai tidak ditemukan	
30	Menguji menampilkan pegawai yang telah mengikuti <i>training</i>	Memilih kegiatan <i>training</i> yang ada	Sistem akan menampilkan pegawai yang telah mengikuti <i>training</i> sesuai <i>training</i> yang diikuti	

7. Desain Uji Coba Halaman Absensi

Halaman master pegawai perlu diuji untuk mengetahui apakah halaman tersebut dapat berfungsi dengan baik. Pengujian halaman tersebut dilakukan dengan menggunakan tombol yang telah disediakan. Tabel 3.17 berikut ini menjelaskan uji coba yang akan dilakukan pada halaman master pegawai.

Tabel 3.17 Desain Uji Coba Halaman Absensi

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
31	Pengujian validasi <i>username</i> dan <i>password</i> untuk absensi harian	<i>Username</i> =k02080001; <i>password</i> = k02080001, kemudian klik masuk	Sistem akan menampilkan pesan sukses dan menyimpan waktu masuk kerja	
32	Pengujian validasi <i>username</i> dan <i>password</i> yang salah	<i>Username</i> =andi; <i>password</i> =12345, kemudian klik masuk	Sistem akan menampilkan kesalahan absensi.	

8. Desain Uji Coba Halaman Bagian dan Jabatan

Test case ini guna memenuhi fitur departemen dan jabatan. Fitur yang harus dipenuhi adalah memberikan jabatan berdasarkan tiap departemen.

Tabel 3.18 Desain Uji Coba halaman Departemen dan Jabatan

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output Diharapkan</i>	Status
33	Menampilkan Departemen	Tekan tautan tambah.	Tampil Nama departemen	
34	Mengedit departemen	Tekan tautan <i>edit</i> di dalam kolom aksi, melakukan	Perubahan dapat berubah seperti yang diinginkan	

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output Diharapkan</i>	Status
		perubahan		
35	Menampilkan jabatan	Tekan tombol tautan tambah.	Tampil departemen, nama jabatan	
36	Mengedit jabatan	Tekan tautan <i>edit</i> dalam kolom aksi, melakukan perubahan.	Perubahan dapat berubah seperti yang diinginkan	

9. Desain Uji Coba Halaman laporan

Halaman mengakses seluruh laporan digunakan untuk menampilkan data yang telah diolah. Pengujian halaman ini dilakukan pada fungsi penyaringan data apakah dapat berfungsi dengan baik. Uji coba yang akan dilakukan dijelaskan pada *tabel 3.19*.

Tabel 3.19 Desain Uji Coba Halaman Laporan

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
37	Menampilkan laporan kehadiran, kemudian dicetak	Klik tautan absensi, kemudian generate laporan. Masukkan tanggal 08-07-2013	Dapat mencetak laporan kehadiran pegawai keseluruhan dalam bentuk PDF	
38	Menampilkan laporan detail	Klik tautan absensi, kemudian generate	Dapat mencetak laporan	

<i>Test case id</i>	Tujuan	<i>Input</i>	<i>Output yang diharapkan</i>	Status
	kehadiran, kemudian dicetak	laporan. Kemudian centang lap rekapitulasi dan masukkan tanggal 01-07-2013 s/d 01-07-2013	rekapitulasi kehadiran pegawai keseluruhan dalam bentuk PDF	
39	Menampilkan laporan penilaian tiap bulan, kemudian dicetak	Klik tautan penilaian kinerja kemudian klik laporan hasil penilaian	Dapat mencetak laporan penilaian kinerja pegawai keseluruhan dalam bentuk pdf	
40	Menampilkan laporan riwayat pegawai, kemudian dicetak	Klik tautan master pegawai, kemudian pilih nama pegawai, selanjutnya klik view dan cetak riwayat pegawai.	Dapat mencetak laporan riwayat pegawai dalam bentuk pdf	
41	Menampilkan laporan perusahaan outsourcing, kemudian dicetak	Klik tautan laporan kemudian pilih laporan outsourcing dan cetak	Dapat mencetak laporan perusahaan outsourcing dan nama pegawai outsourcing dalam bentuk pdf	
42	Menampilkan laporan pelanggaran, kemudian dicetak	Klik tautan laporan kemudian pilih laporan pelanggaran dan cetak sesuai tanggal yang dimasukkan	Dapat mencetak laporan pelanggaran dalam bentuk pdf	

STIKOM SURABAYA