

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Untuk mendapatkan gambaran umum dalam membangun sistem yang baik, maka diperlukan perancangan analisis sistem yang mampu mengatasi permasalahan yang ada dengan cara membuat perancangan baru yang nantinya akan diimplementasikan pada CV AGUNG JAYA.

CV AGUNG JAYA Sukoharjo merupakan sebuah tempat usaha yang bergerak dalam bidang penjualan beras yang bertempat di Desa Karangwuni, Kecamatan Polokarto, Kabupaten Sukoharjo. Adapun masalah yang dihadapi yaitu proses *inventory* yang sedang berjalan saat ini masih manual sehingga sering terjadi ketidakakuratan data dan waktu yang diperlukan cukup lama, karena proses pencatatannya masih dilakukan secara manual dan berulang-ulang serta dapat membuat proses jual beli terlambat dikarenakan *update* datanya terlambat.

3.1.1 Analisis Sistem Yang Berjalan saat ini

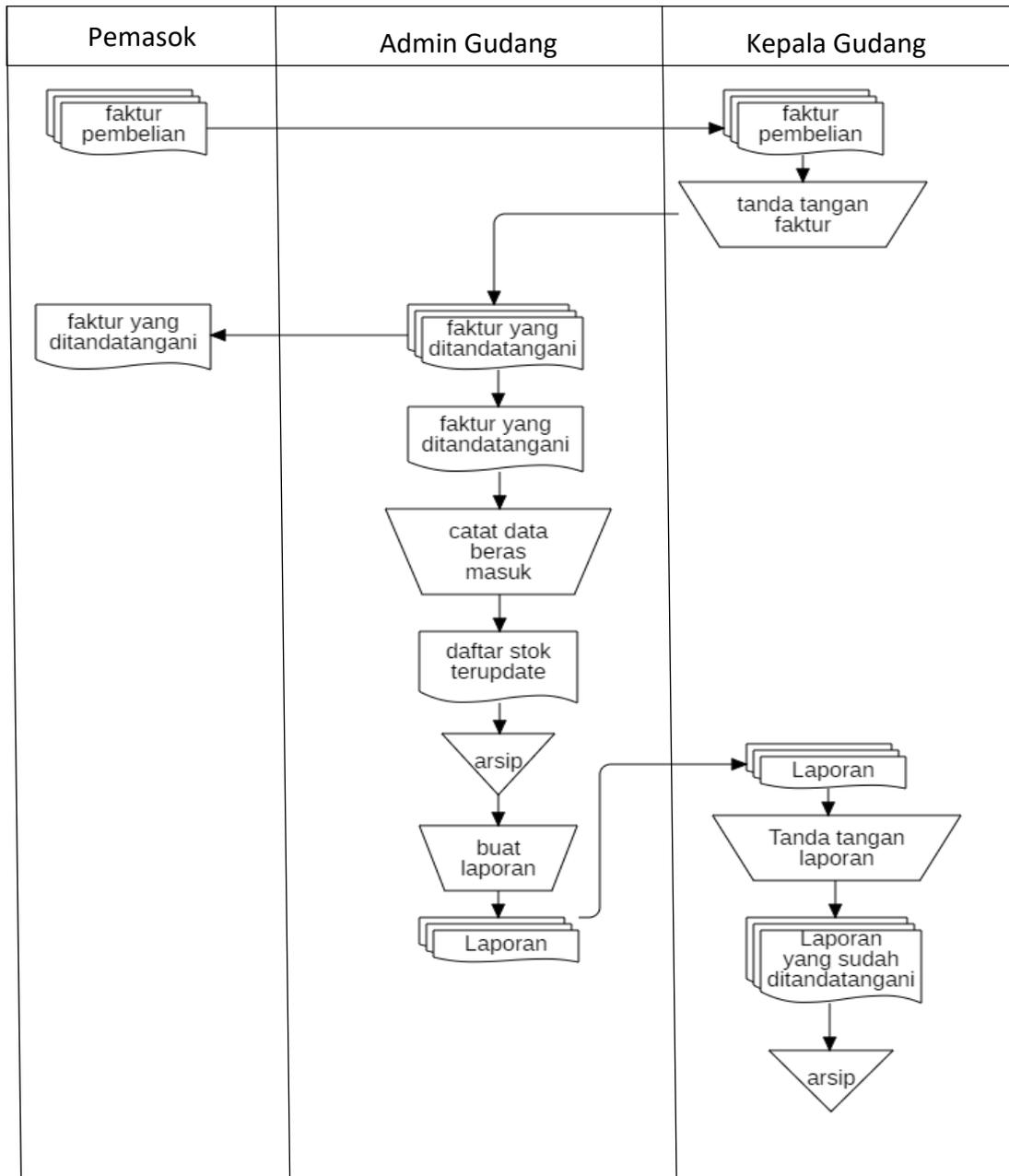
Setelah melakukan observasi dan menganalisa pada CV AGUNG JAYA dapat diketahui sistem yang sedang berjalan selama ini masih kurang efektif, dikarenakan pendataan untuk pengolahan data barang dan laporan jual beli masih dicatat secara manual.

Hasil dari menganalisa prosedur yang sedang berjalan menguraikan secara sistematis aktifitas-aktifitas yang terjadi dalam proses pendataan persediaan stok barang pada CV AGUNG JAYA Sukoharjo. Agar tahapan proses pembangunan sistem dapat berjalan dengan baik dan lancar, akan lebih jelas digambarkan dengan menggunakan pemodelan *flowmap* di bawah ini.

1.1.1.1 Flowmap Pendataan Beras Masuk

Flowmap pendataan beras masuk merupakan proses dimana stok beras yang masuk dari pemasok yang nantinya akan ditambahkan di dalam gudang penyimpanan

seperti yang dapat dilihat pada Gambar 3.3 dibawah ini



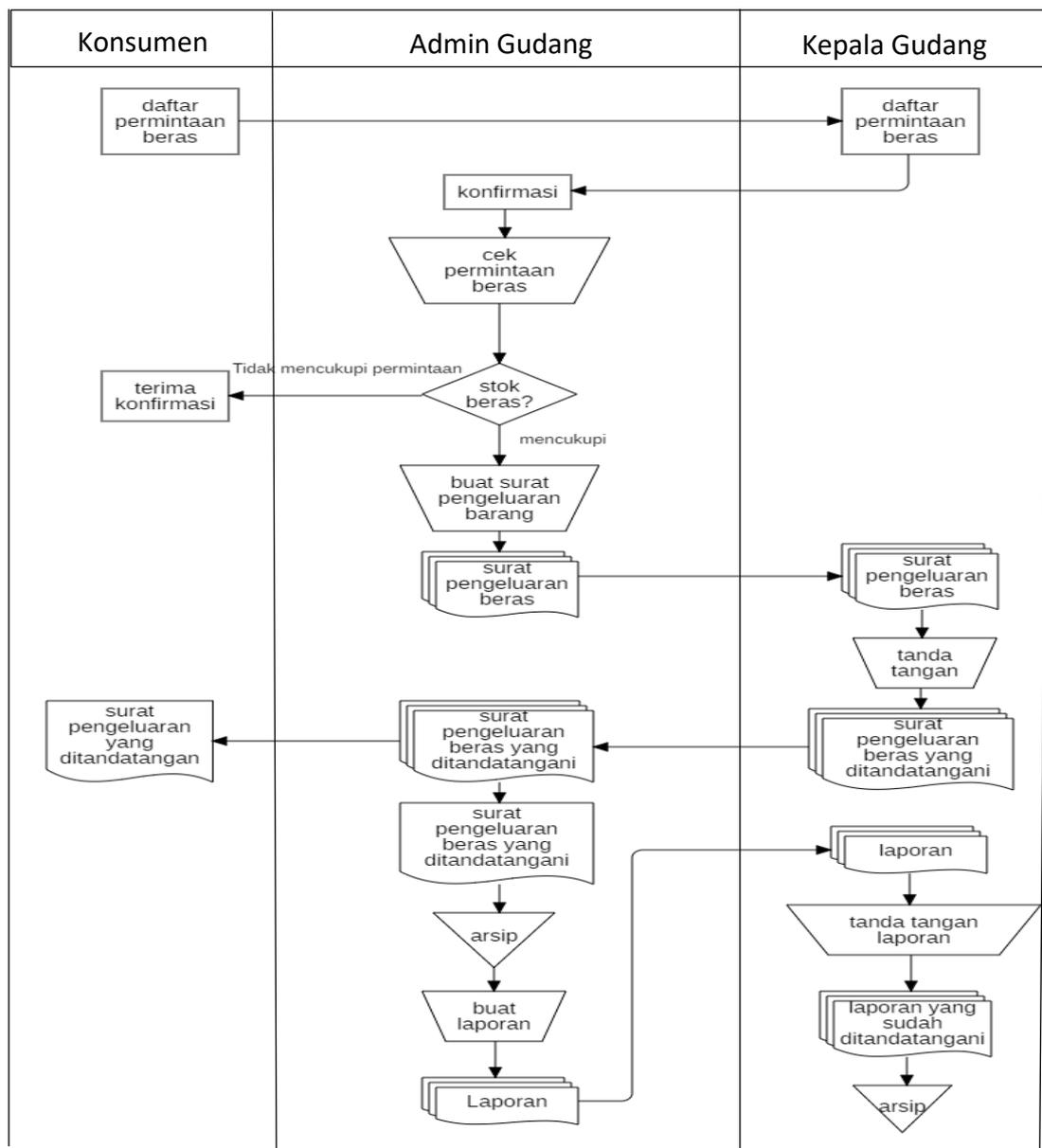
Gambar 3.1 *Flowmap* Pendataan Barang Masuk Yang Berjalan

Gambar diatas menjelaskan bahwa setiap beras yang masuk faktur pembelian dari pemasok akan dicatat data beras yang dibeli dan total harga pembelian yang keluar, setelah stok *terupdate* maka data pencatatan akan dimasukkan kedalam arsip. Setelah arsip terkumpul maka admin gudang membuat laporan pembelian yang sudah dilakukan dan laporan diserahkan ke kepala gudang dan ditandatangani lalu

kembali diarsipkan.

1.1.1.2 Flowmap Pendataan Beras Keluar

Flowmap pendataan beras keluar merupakan proses pencatatan laporan beras dijual dan laporan pendapatan saat beras kembali dijual serta pencatatan *update* stok beras terkini seperti yang dilihat pada Gambar 3.4 dibawah ini.



Gambar 3.2 *Flowmap* Pendataan Beras Keluar

Gambar 3.4 diatas menjelaskan dimana daftar permintaan beras dari pembeli akan dicek oleh admin gudang, jika stok yang dibutuhkan tercukupi lalu buat surat

pengeluaran beras dan diserahkan di kepala gudang. Jika sudah tertandatangani lalu akan dibuatkan notifikasi nota yang diserahkan kepada pembeli. Setelah proses transaksi maka proses selanjutnya adalah proses pembuatan laporan stok beras yang telah berkurang dan pencatatan pendapatan setelah proses transaksi. Setelah laporan jadi maka diserahkan kepada kepala gudang dan ditandatangani lalu dimasukkan kedalam arsip.

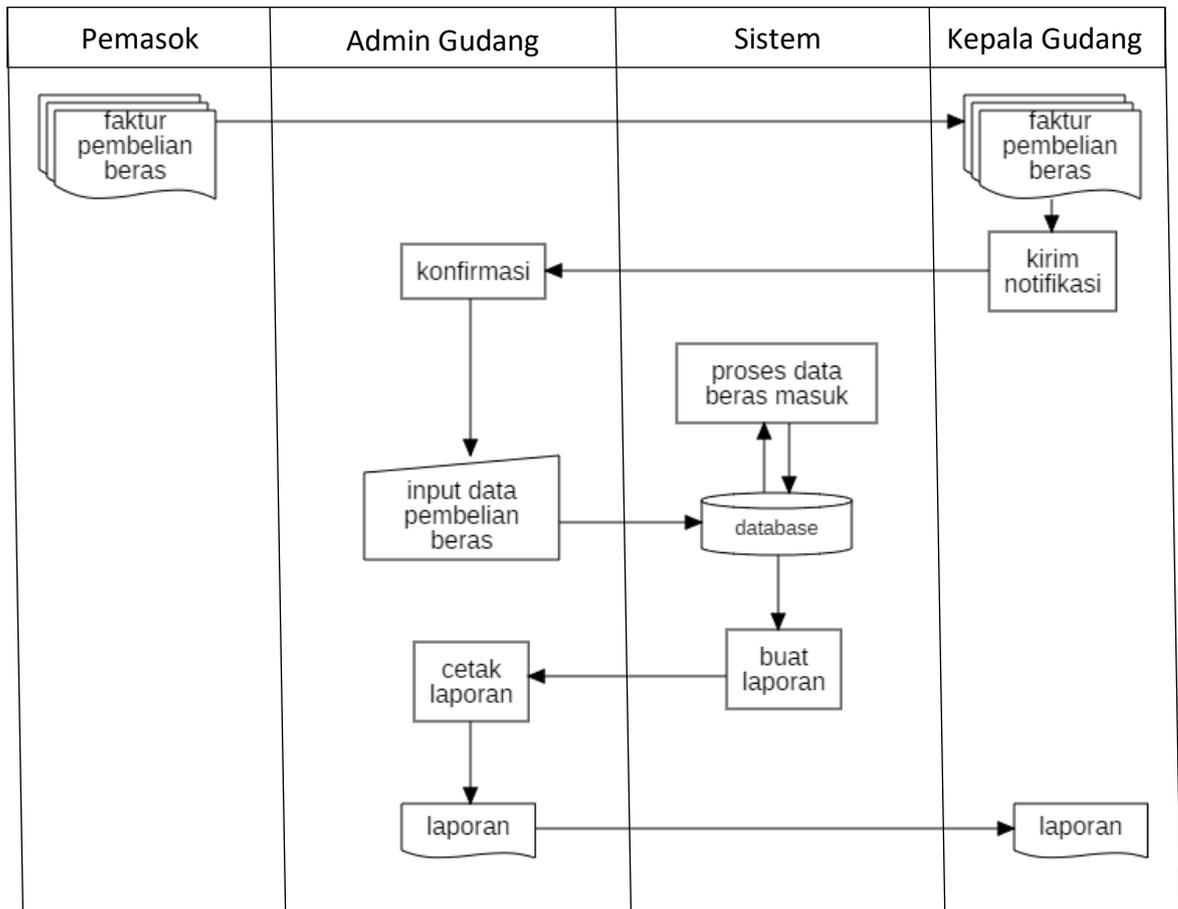
3.1.2 Analisis Yang Baru

Berdasarkan hasil pengamatan dan penelitian dari sitem informasi *Inventory* yang sedang berjalan pada CV AGUNG JAYA Sukoharjo, dimana selama ini masih menggunakan sistem pencatatan persediaan dan pelaporan beras secara manual yang sering menyebabkan data yang didapat kurang akurat serta waktu yang diperlukan cukup lama yang dapat mengakibatkan kendala dalam memproduksi barang sehingga sering terjadi kesalahan atau tidak teliti dalam pencatatan laporan, Seiring dengan berkembangnya CV AGUNG JAYA permintaan akan kebutuhan pelanggan dalam pemesanan barang juga meningkat sehingga sangat berpengaruh pada *Inventory*.

Untuk menanggulangi masalah yang terdapat pada sistem *inventory* stok beras pada CV AGUNG JAYA Sukoharjo maka diperlukan perubahan - perubahan dalam sistem. Adapun perubahan - perubahan yang dilakukan pada usulan pemecahan masalah yang diuraikan pada sub bab sistem yang berjalan saat ini sebelumnya dengan mengimplementasikan bahasa pemograman *web* dimana aplikasi ini dapat memudahkan *user* dalam pengolahan data baik dalam stok beras dan juga pelaporan serta *user* dapat mengetahui stok beras dan profit penghasilan yang diterima di CV AGUNG JAYA Sukoharjo.

3.1.2.1 Flowmap Pendataan Beras Masuk Yang Baru

Flowmap pendataan beras masuk yang baru merupakan sistem yang akan berjalan kedepannya seperti yang dideskripsikan pada Gambar 3.5 dibawah ini.

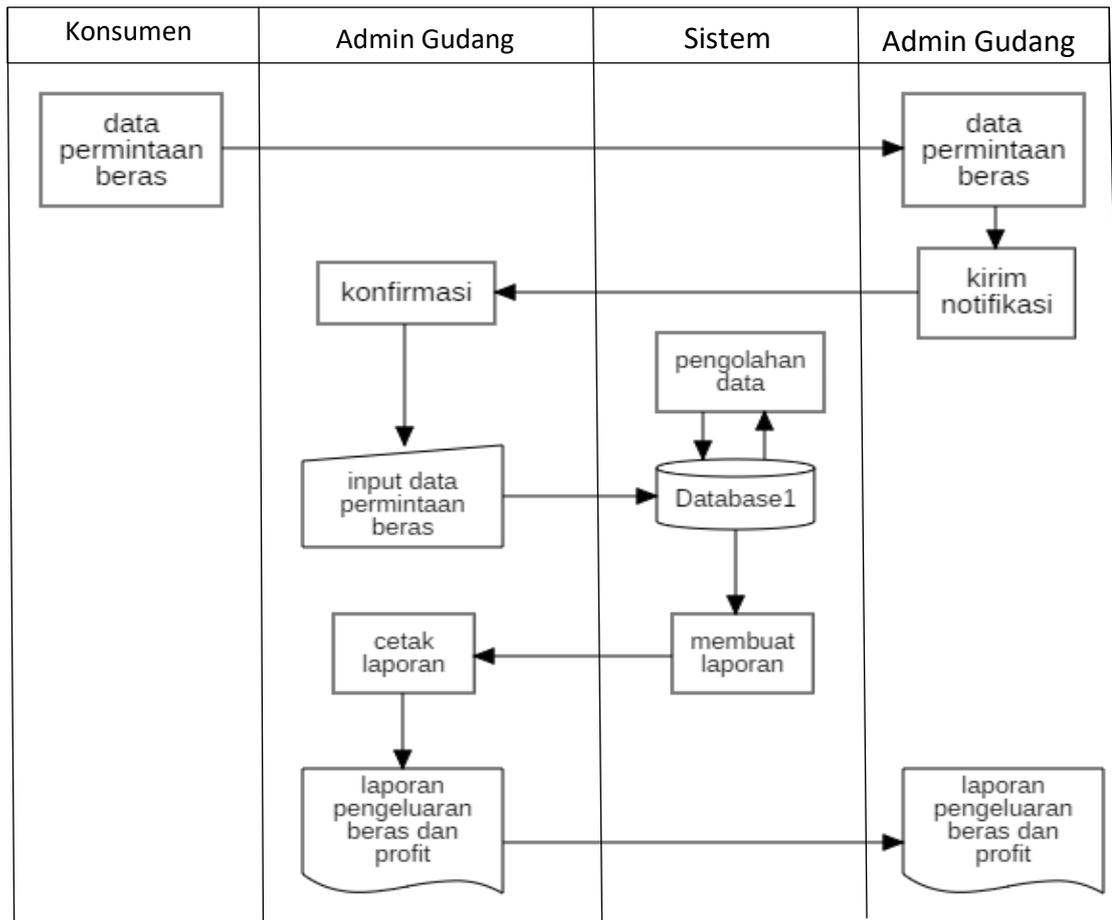


Gambar 3.3 *Flowmap* Pendataan Beras Masuk Yang Baru

Gambar 3.5 diatas menjelaskan saat faktur pembelian diterima oleh admin, lalu admin melakukan input data pembelian beras yang dibutuhkan. Setelah diinputkan lalu simpan dan akan ditampung dalam *database*. Setelah itu sistem akan membuat laporan rekap data pembelian beras pada *supplier*. Setelah itu laporan bisa dicetak dan hasil laporan akan diserahkan kepada kepala gudang.

3.1.2.2 Flowmap Pendataan Beras Keluar Yang Baru

Flowmap ini merupakan proses beras keluar yang akan diimplementasikan kedepannya seperti yang diperlihatkan pada Gambar 3.6 dibawah ini.



Gambar 3.4 *Flowmap* Pendataan Beras Keluar Yang Baru

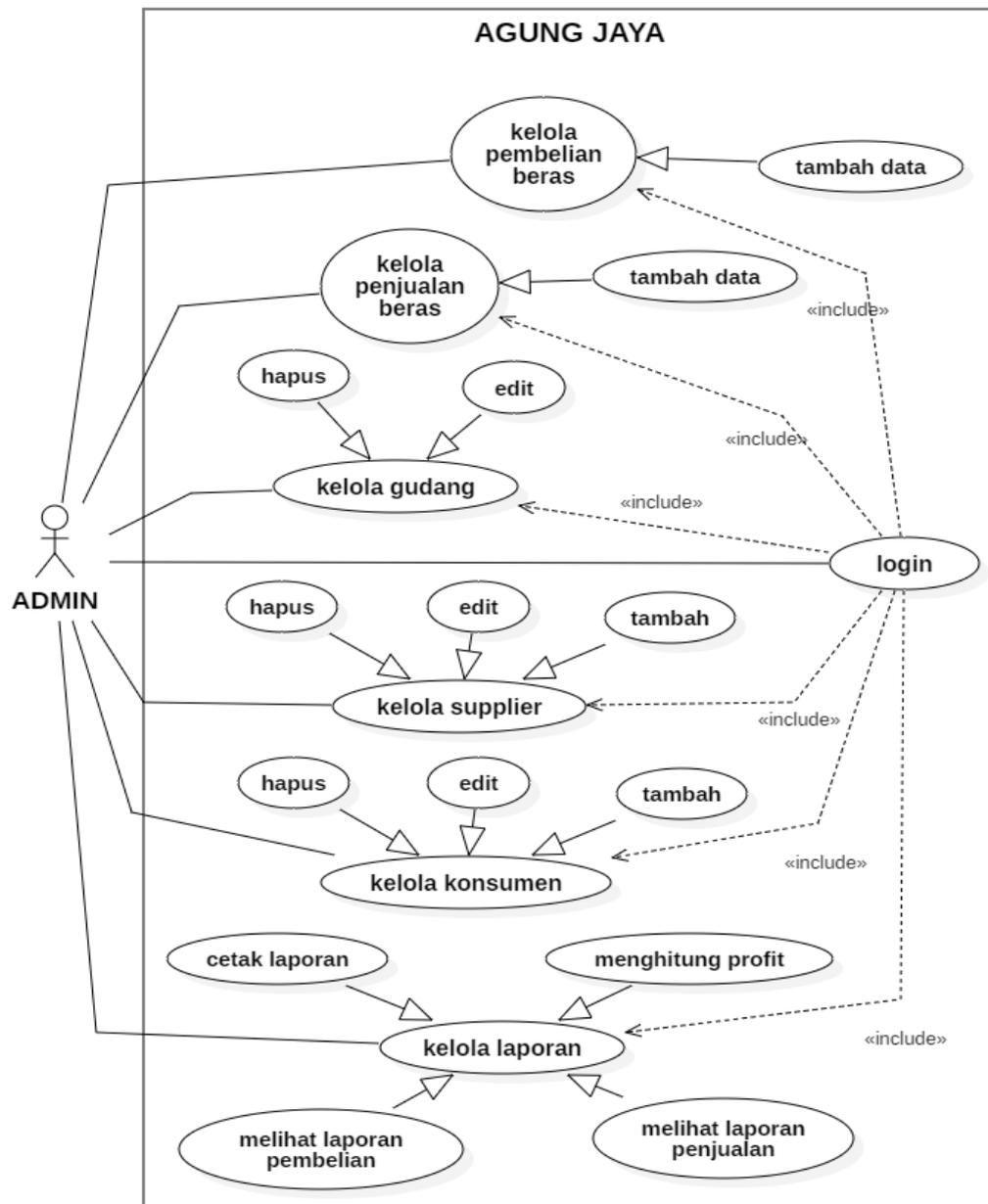
Gambar 3.6 diatas menjelaskan dimana saat data permintaan beras dari konsumen diterima admin gudang lalu admin gudang menginputkan data permintaan beras yang dibutuhkan konsumen. Setelah diinputkan datanya lalu disimpan dan akan ditampung didalam *database*. Setelah itu sistem akan membuat rekap laporan penjualan dan profit lalu rekapan tersebut dicetak dan siserahkan kepada kepala gudang untuk ditindak lanjuti.

3.2 Rancangan Sistem

Perancangan sistem selanjutnya berisikan penggambaran, perencanaan, dan pembuatan sketsa dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi. Perancangan sistem *inventory* beras CV AGUNG JAYA yang dibangun ini bersifat *object oriented* (berorientasi objek) dengan menggunakan metode *Unified Modelling Language* (UML).

3.2.1 Use Case Diagram

Pada langkah ini yang pertama kali akan dilakukan adalah memodelkan sistem atau perangkat lunak yang akan dibuat dengan menggunakan *use case diagram*. *Use case diagram* ini menggambarkan hubungan antara aktor dan kegiatan yang dapat dilakukannya terhadap aplikasi pada CV AGUNG JAYA seperti yang terlihat pada Gambar 3.7 dibawah ini.



Gambar 3.5 Use Case Diagram

3.2.1.1 Use Case Login

Use case login merupakan langkah kerja yang dilakukan oleh *user* (pengguna) sebelum memulai menggunakan sistem *inventory* beras. *Use case login* ini dibuat untuk menjelaskan apa dan siapa yang dapat mengakses situs jika *form* pengisian yang diisikan benar, maka sistem akan menampilkan halaman menu utama sesuai

hak akses. Berikut merupakan penjelasan *use case login* pada Tabel 3.8.

Tabel 3.1 *Use Case Login*

<i>Use Case</i>	<i>Login</i>
Deskripsi	<i>Use case</i> ini menangani verifikasi dan hak akses pengguna sistem aplikasi <i>inventory</i> beras CV AGUNG JAYA
<i>Actor</i>	<i>Admin</i>
<i>Condition</i>	<i>Login</i> merupakan langkah <i>user</i> sistem untuk menjalankan aplikasi atau <i>sistem</i> dan menjadi pintu untuk hak akses pengguna sistem

Keterangan :

1. *Use case* ini dimulai ketika admin melakukan *login*
2. *Admin* meng-input nama *username* dan *password*
3. *Admin* menekan tombol *login*
4. Sistem memeriksa *username* dan *password* pengguna dan menampilkan *form* sesuai dengan hak akses

3.2.1.2 *Use Case* Kelola Pembelian Beras

Use case kelola pembelian beras adalah langkah kerja ketika aktor ingin membeli beras dari *supplier*. *Admin* membuka menu pembelian dan sistem akan menampilkan menu *form* pembelian. Berikut merupakan penjelasan *use case* mengelola pembelian pada Tabel 3.9 dibawah ini.

Tabel 3.2 *Use Case* Mengelola Pembelian Beras

<i>Use Case</i>	<i>Login</i>
Deskripsi	<i>Use Case</i> ini melakukan tambah data pembelian
<i>Actor</i>	<i>Admin</i>
<i>Condition</i>	<i>Admin</i> memilih menu pembelian dan sistem menyimpan data pembelian

Keterangan :

1. *Use case* ini dimulai ketika *admin* telah melakukan proses *login*
2. *Admin* masuk ke menu pembelian
3. Sistem menampilkan form pembelian
4. *Admin* melakukan penginputan pada *form* yang telah disediakan
5. Sistem menyimpan data pembelian

3.2.1.3 Use Case Kelola Penjualan Beras

Use case kelola penjualan beras adalah langkah kerja ketika aktor ingin menjual beras ke konsumen. *Admin* membuka menu penjualan dan sistem akan menampilkan menu *form* penjualan. Berikut merupakan penjelasan *use case* mengelola penjualan pada Tabel 3.10 dibawah ini.

Tabel 3.3 *Use Case* Mengelola Penjualan Beras

<i>Use Case</i>	<i>Login</i>
Deskripsi	<i>Use Case</i> ini melakukan tambah data pembelian
<i>Actor</i>	<i>Admin</i>
<i>Condition</i>	<i>Admin</i> memilih menu penjualan dan sistem menyimpan data penjualan

Keterangan :

1. *Use case* ini dimulai ketika *admin* telah melakukan proses *login*
2. *Admin* masuk ke menu pembelian
3. Sistem menampilkan form pembelian
4. *Admin* melakukan penginputan pada *form* yang telah disediakan
5. Sistem menyimpan data pembelian

3.2.1.4 Use Case Kelola Gudang

Use case kelola gudang adalah langkah kerja ketika aktor ingin menyimpan data

stok beras. *Admin* membuka menu gudang dan sistem akan menampilkan menu *data* stok beras. Berikut merupakan penjelasan *use case* mengelola gudang pada Tabel 3.11 dibawah ini.

Tabel 3.4 Use Case Mengelola Gudang

<i>Use Case</i>	<i>Login</i>
Deskripsi	<i>Use Case</i> ini melakukan penyimpanan stok beras
<i>Actor</i>	<i>Admin</i>
<i>Condition</i>	<i>Admin</i> memilih menu gudang dan sistem akan memperlihatkan stok beras

Keterangan :

1. *Use case* ini dimulai ketika *admin* telah melakukan proses *login*
2. *Admin* masuk ke menu gudang
3. Sistem menampilkan halamn gudang
4. *Admin* dapat melihat stok gudang *terupdate*
5. *Admin* dapat mengedit dan menghapus sesuai keinginan
6. Sistem menyimpan data perubahan

3.2.1.5 Kelola Supplier

Use case kelola *supplier* adalah langkah kerja ketika *admin* ingin menambahkan *supplier*. *Admin* membuka menu *supplier* dan sistem akan menampilkan menu *form* kelola *supplier*. Berikut merupakan penjelasan *use case* mengelola *supplier* pada Tabel 3.12 dibawah ini.

Tabel 3.5 Use Case Mengelola Supplier

<i>Use Case</i>	<i>Login</i>
Deskripsi	<i>Use Case</i> ini melakukan tambah data pembelian
<i>Actor</i>	<i>Admin</i>
<i>Condition</i>	<i>Admin</i> memilih menu pembelian dan sistem menyimpan data pembelian

Keterangan :

1. *Use case* ini dimulai ketika *admin* telah melakukan proses *login*
2. *Admin* masuk ke menu *supplier*
3. Sistem menampilkan menu *supplier*
4. *Admin* dapat mengelola *supplier* seperti menambah, edit dan hapus
5. Sistem menyimpan data perubahan

3.2.1.6 Kelola Konsumen

Use case kelola konsumen adalah langkah kerja ketika *admin* ingin menambahkan konsumen. *Admin* membuka menu konsumen dan sistem akan menampilkan menu *form* kelola konsumen. Berikut merupakan penjelasan *use case* mengelola konsumen pada Tabel 3.13 dibawah ini.

Tabel 3.6 Mengelola Konsumen

<i>Use Case</i>	<i>Login</i>
Deskripsi	<i>Use Case</i> ini melakukan tambah data pembelian
<i>Actor</i>	<i>Admin</i>
<i>Condition</i>	<i>Admin</i> memilih menu pembelian dan sistem menyimpan data pembelian

Keterangan :

1. *Use case* ini dimulai ketika *admin* telah melakukan proses *login*
2. *Admin* masuk ke menu konsumen
3. Sistem menampilkan menu konsumen
4. *Admin* dapat mengelola konsumen seperti menambah, edit dan hapus
5. Sistem menyimpan data perubahan

3.2.1.7 Kelola Laporan

Use case kelola laporan adalah langkah kerja ketika aktor ingin melihat laporan data penjualan dan pembelian serta profit keuntungan. *Admin* membuka menu laporan dan sistem akan menampilkan menu laporan penjualan, pembelian, dan profit. Berikut merupakan penjelasan *use case* mengelola laporan pada Tabel 3.13

dibawah ini.

Tabel 3.7 Use Case Mengelola Laporan

<i>Use Case</i>	<i>Login</i>
Deskripsi	<i>Use Case</i> ini melakukan tambah data pembelian
<i>Actor</i>	<i>Admin</i>
<i>Condition</i>	<i>Admin</i> memilih menu laporan dan sistem akan mengkalkulasikan laporan penjualan, pembelian dan profit.

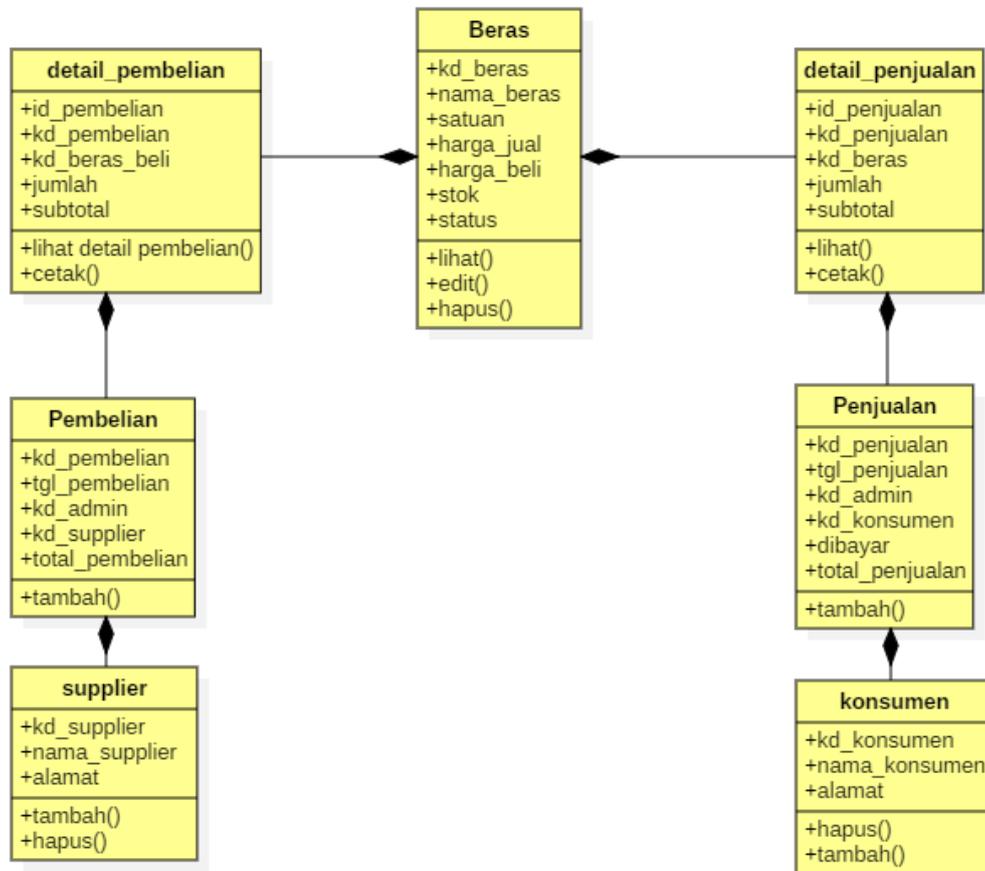
Keterangan :

1. *Use case* ini dimulai ketika *admin* telah melakukan proses *login*
2. *Admin* masuk ke menu laporan
3. Sistem menampilkan menu laporan
4. *Admin* dapat melihat laporan penjualan, pembelian dan profit yang dapat diatur menurut tanggalnya

3.2.2 Class Diagram

Class biasanya digunakan untuk menggambarkan obyek-obyek bisnis. *Class-class* seperti ini biasanya mendefinisikan model *database* dari suatu sistem. Atas dasar itulah *class* seperti ini sering disebut dengan *class entity* karena mewakili obyek *database*, seperti yang terlihat pada Gambar 3.21.

Class diagram menjelaskan hubungan antar *class* yang bertugas memproses *inputan* dari *user*, mengolah, dan menampilkan informasi ke *user*.



Gambar 3.6 Class Diagram

Tabel *supplier* digunakan untuk menyimpan data supplier yang memasok saat pembelian barang dilakukan untuk mempermudah proses laporan yang akan datang. Untuk tabel konsumen digunakan untuk menyimpan data konsumen yang membeli saat penjualan barang dilakukan untuk mempermudah proses laporan yang akan datang, untuk tabel beras untuk menyimpan stok-stok beras *terupdate*, tabel Pembelian untuk menyimpan data-data pembelian, tabel penjualan untuk menyimpan data-data penjualan, tabel detail pembelian digunakan untuk menyimpan detail dari data pembelian, tabel detail penjualan digunakan untuk menyimpan detail dari data penjualan

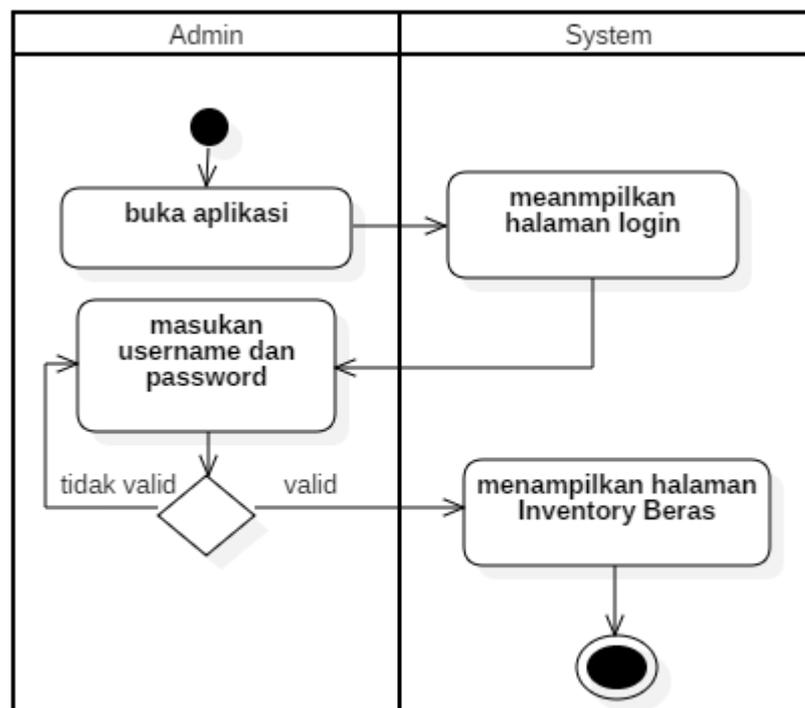
3.2.3 Activity Diagram

Activity diagram digunakan untuk menggambarkan aktifitas sebuah proses bisnis atau sistem. Berikut dibawah ini merupakan *activity diagram* dari sistem *inventory* CV AGUNG JAYA.

3.2.2.1. Activity Diagram Login

Untuk memasuki aplikasi sistem *inventory* CV AGUNG JAYA dibutuhkan *login* agar dapat mengakses aplikasi sistem *inventory* ini.

Sebelum memasuki halaman sistem *inventory* beras diwajibkan untuk *login* terlebih dahulu. Sistem akan otomatis mengarahkan halaman *login* terlebih dahulu. Di halaman *login* ini *admin* diwajibkan memasukan *email* dan *password* yang telah dibuat sebelumnya untuk dapat mengakses sistem *inventory* beras CV AGUNG JAYA. Seperti yang terlihat pada Gambar 3.8.

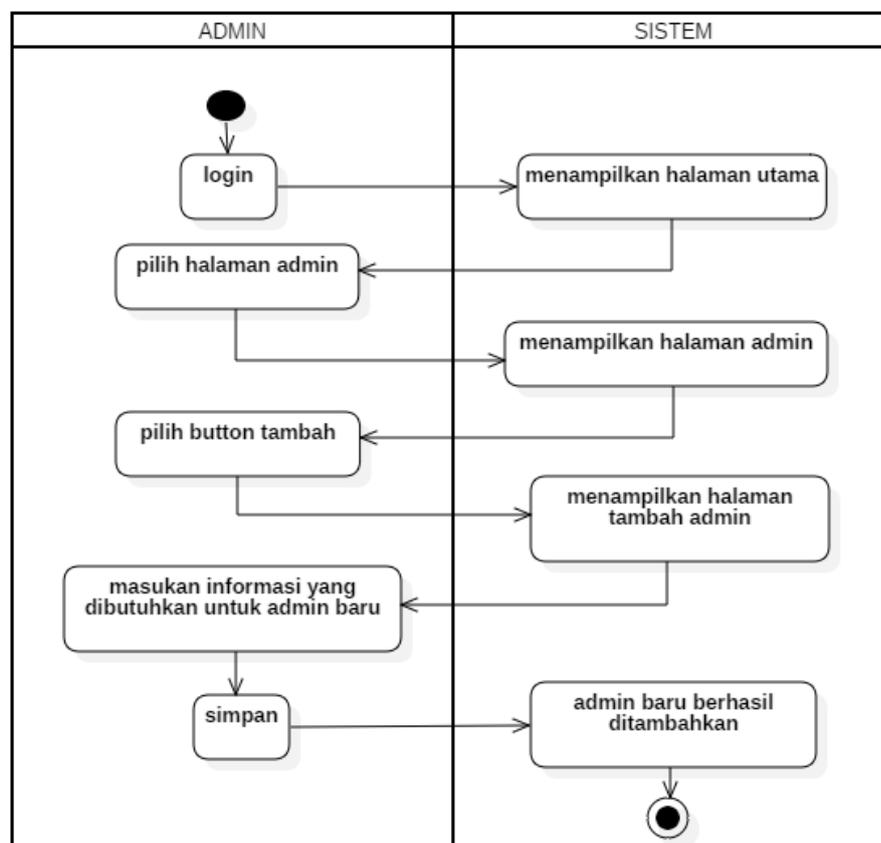


Gambar 3.7 Activity Diagram Login

3.2.2.2. Activity Diagram Tambah Admin

Activity Diagram tambah *admin* merupakan langkah dimana ingin menambah *admin* baru untuk mengelola sistem *inventory* pada CV AGUNG JAYA.

Diawali dengan *login* terlebih dahulu setelah itu akan tampil menu halaman utama. Langkah selanjutya yakni pilih halaman *admin*. Setelah itu pilih *button* tambah *admin*. Lalu masukan informasi mengenai *admin* yang baru agar dapat mengakses sistem *inventory* ketika *login* dan simpan. Seperti yang disuguhkan pada Gambar 3.9.

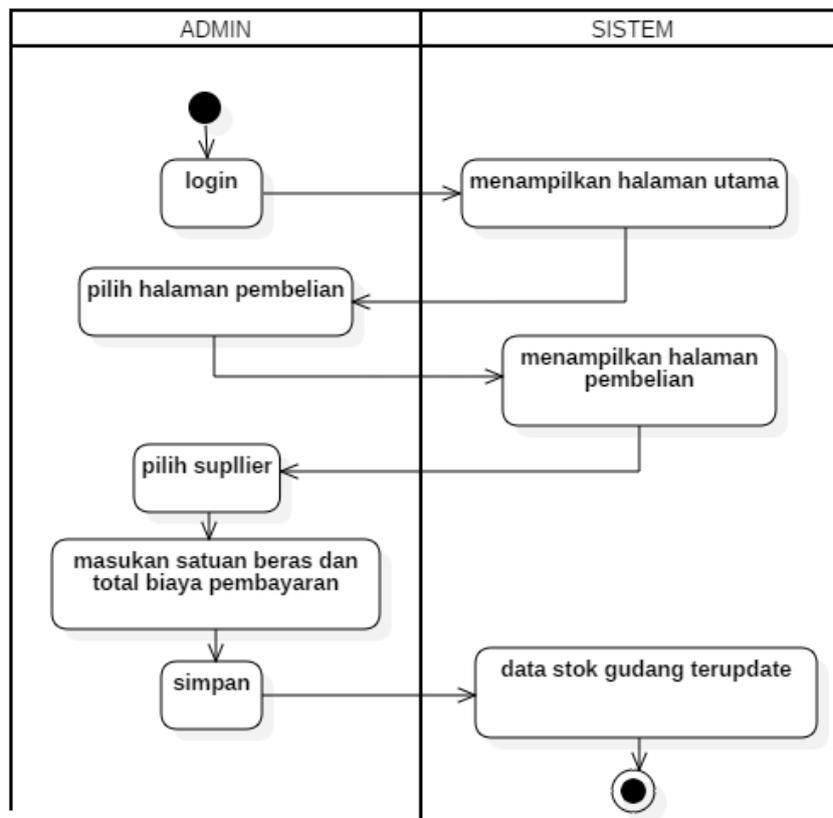


Gambar 3.8 Activity Diagram Tambah Admin

3.2.2.3. Activity Diagram Pembelian Beras

Activity Diagram Pembelian merupakan langkah dimana *admin* akan memasukan informasi pembelian stok beras pada *Supplier* yang telah ditentukan.

Diawali dengan *login* terlebih dahulu setelah itu akan tampil menu halaman utama. Langkah selanjutnya yakni pilih halaman pembelian lalu pilih *supplier* yang telah ditentukan sebelumnya. Lalu masukan informasi mengenai pembelian yang dilakukan seperti jumlah beras yang dibeli dan total biaya yang keluar untuk membeli stok beras pada *supplier*. Lalu simpan informasi pembelian. Setelah itu stok gudang akan *terupdate* otomatis. Seperti yang disuguhkan pada Gambar 3.10.



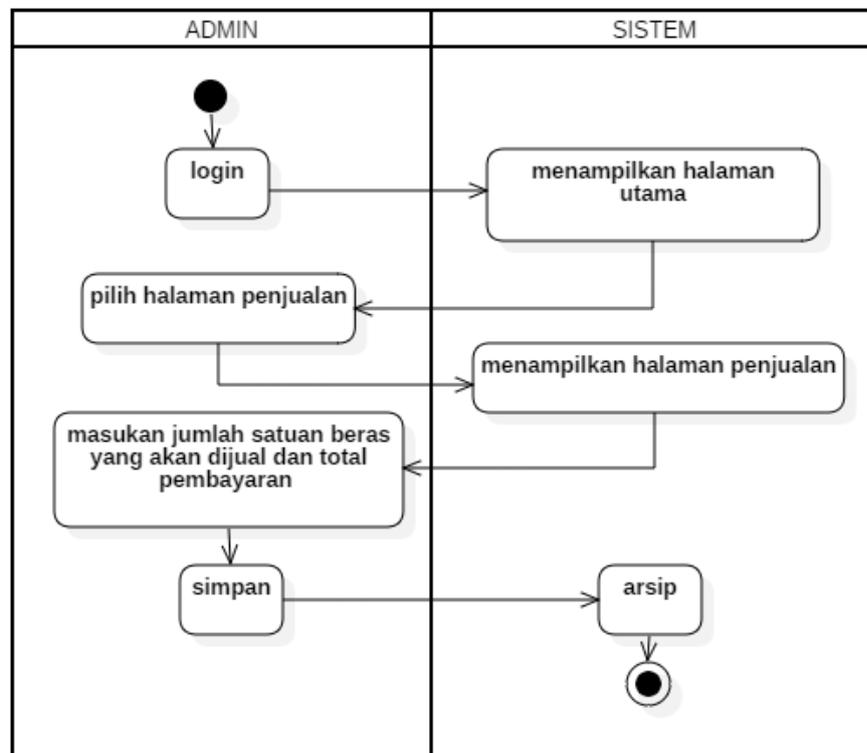
Gambar 3.9 Activity Diagram Pembelian Beras

3.2.2.4. Activity Diagram Penjualan Beras

Activity Diagram Penjualan beras merupakan langkah dimana *admin* akan memasukan informasi penjualan beras pada pembeli.

Diawali dengan *login* terlebih dahulu setelah itu akan tampil menu halaman utama. Langkah selanjutnya yakni pilih halaman penjualan. Lalu masukan informasi mengenai penjualan yang dilakukan seperti jumlah beras yang dijual dan totalan

biaya dari pembeli yang masuk. Lalu simpan informasi penjualan. Setelah itu stok gudang akan *terupdate* otomatis. Seperti yang disuguhkan pada Gambar 3.11.

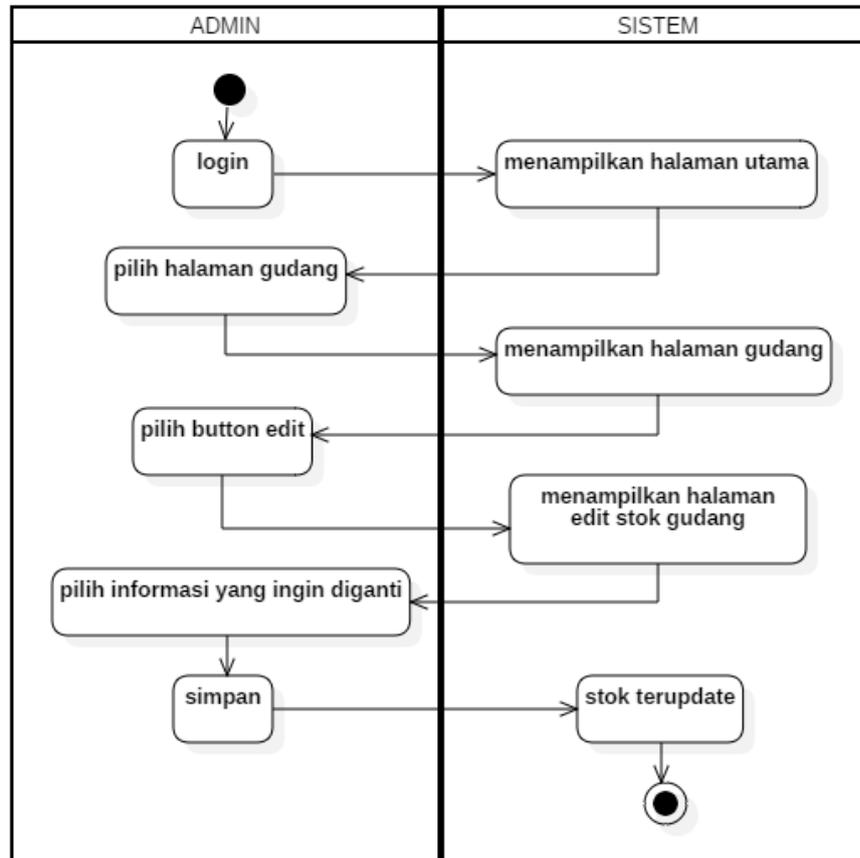


Gambar 3.10 Activity Diagram Penjualan Beras

3.2.2.5. Activiy Diagram Kelola Gudang (Edit)

Activity Diagram Kelola Gudang (Edit) merupakan langkah dimana *admin* ingin mengedit informasi mengenai stok yang sekiranya kurang sesuai dengan informasi yang diberikan.

Langkah pertama yakni *login* untuk memasukii halaman utama. Setelah itu pilih halaman gudang. Setelah memasuki halaman gudang pilih stok yang akan diedit. Masukan informasi yang akan diganti lalu simpan. Stok gudang akan *terupdate* seperti yang disuguhkan pada Gambar 3.12.

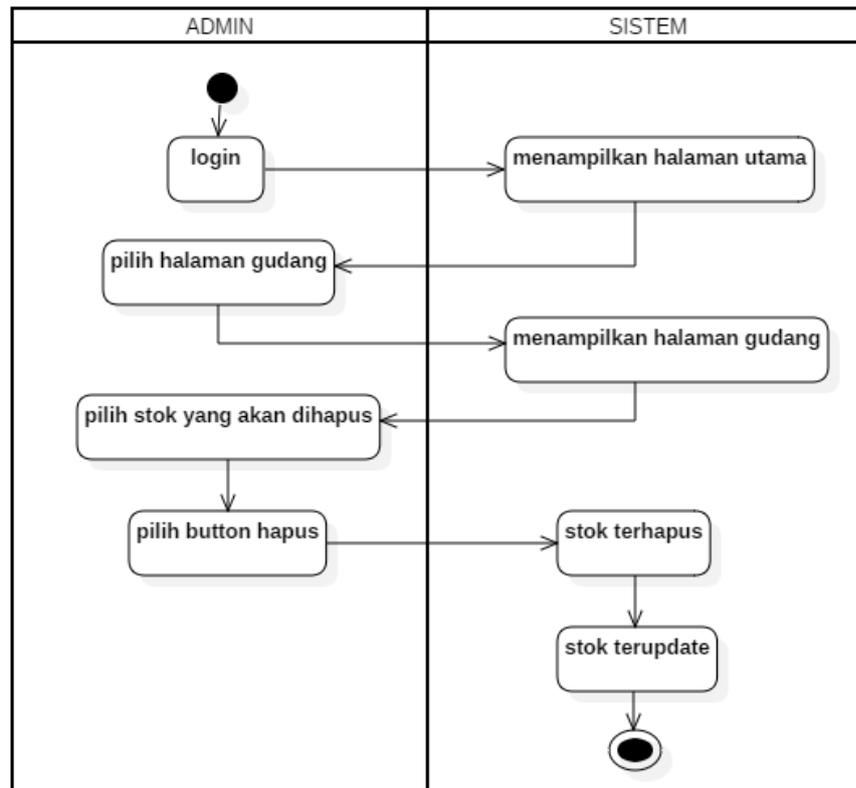


Gambar 3.11 *Activity Diagram* Kelola Gudang (*Edit*)

3.2.2.6. *Activity Diagram* Kelola Gudang (*Hapus*)

Activity Diagram Kelola Gudang (*Hapus*) merupakan langkah dimana *admin* ingin menghapus stok yang sekiranya ingin dihapus.

Langkah pertama yakni *login* untuk memasukii halaman utama. Setelah itu pilih halaman gudang. Setelah memasuki halaman gudang pilih stok yang akan dihapus. Pilih *button* hapus dan otomatis stok akan terhapus. Seperti yang disuguhkan pada Gambar 3.13.

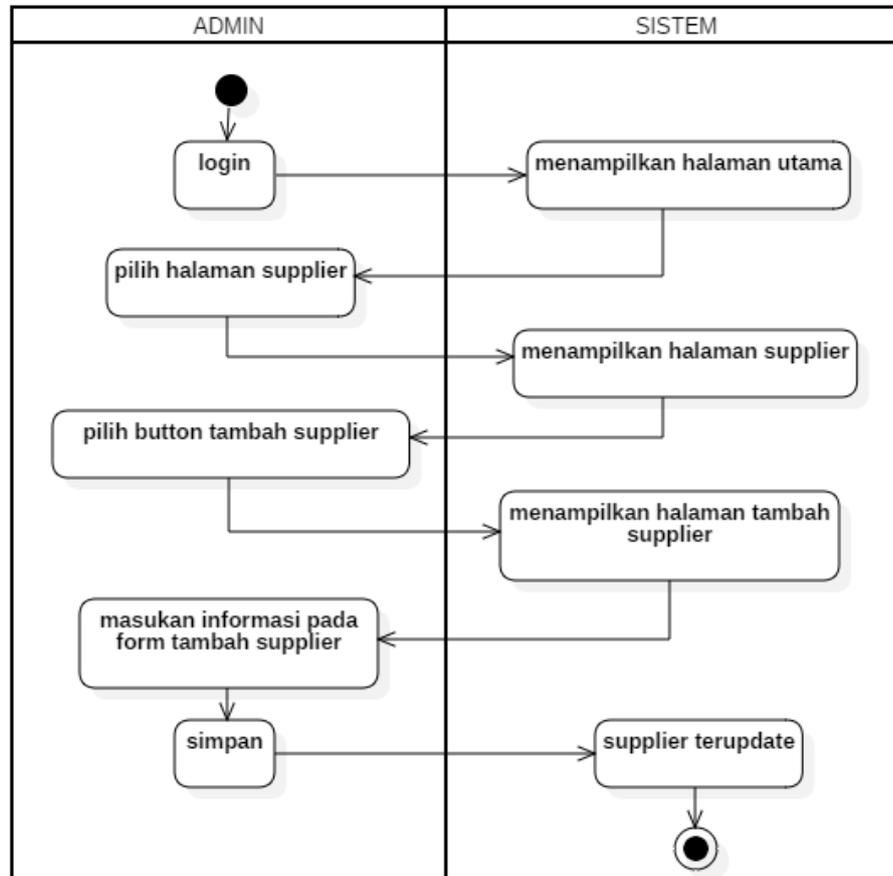


Gambar 3.12 Activity Diagram Kelola Gudang (Hapus)

3.2.2.7. Activity Diagram Kelola Supplier (Tambah)

Activity diagram kelola *supplier* (tambah) merupakan langkah dimana *admin* ingin menambahkan *supplier* yang akan ditambahkan untuk proses pembelian nantinya.

Langkah pertama yakni login untuk memasukii halaman utama. Setelah itu pilih halaman *supplier*. Setelah masuk dihalam *supplier* lalu pilih *button* tambah. Maka sistem akan mengalihkan di halaman tambah *supplier*. Masukan informasi mengenai *supplier* yang akan ditambah lalu simpan. *Supplier* akan *terupdate*. Seperti yang disuguhkan pada Gambar 3.14.

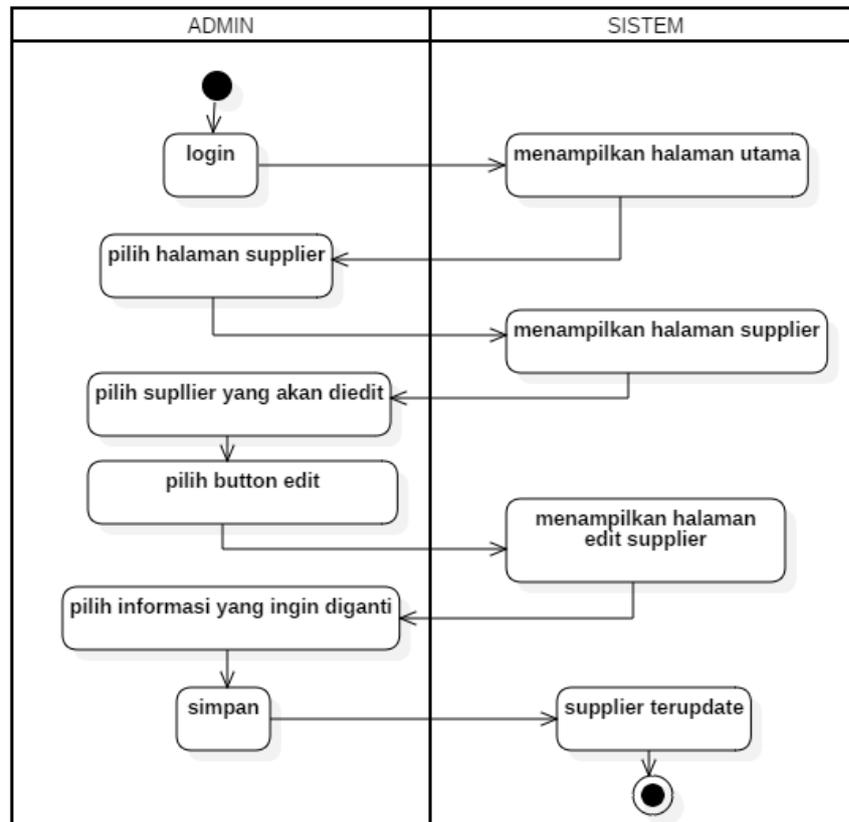


Gambar 3.13 *Activity Diagram* Kelola *Supplier* (Tambah)

3.2.2.8. *Activity Diagram* Kelola *Supplier* (Edit)

Activity diagram kelola *supplier* (edit) merupakan langkah dimana *admin* ingin mengedit *supplier* yang akan *diedit*.

Langkah pertama yakni *login* untuk memasukii halaman utama. Setelah itu pilih halaman *supplier*. Setelah masuk dihalaman *supplier* lalu pilih *supplier* yang akan diedit lalu pilih *button edit*. Maka sistem akan mengalihkan di halaman *edit supplier*. Masukan informasi mengenai *supplier* yang akan diganti lalu simpan. *Supplier* akan *terupdate*. Seperti yang disuguhkan pada Gambar 3.15.

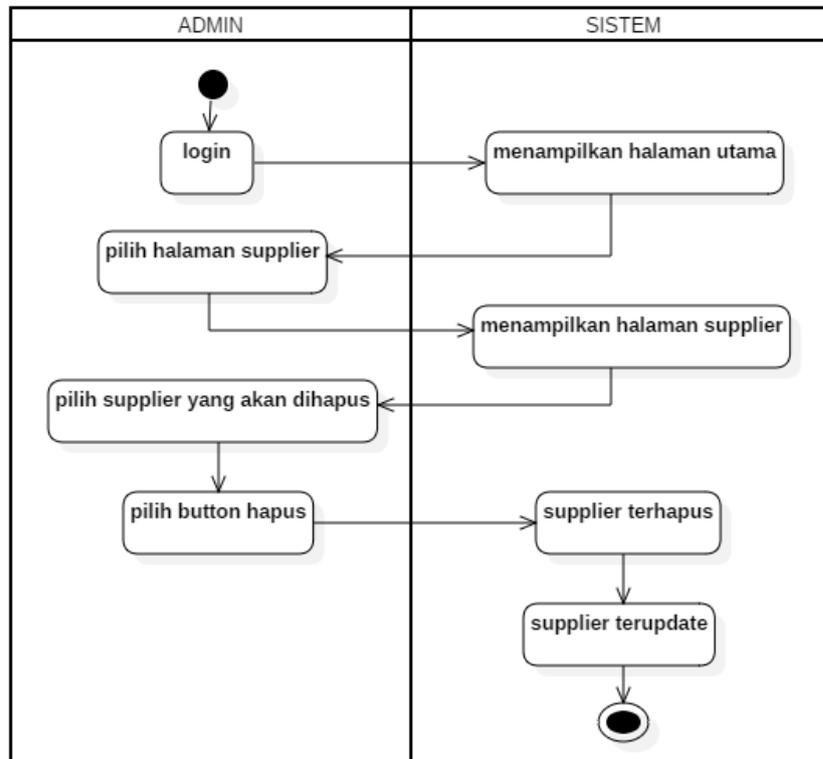


Gambar 3.14 Activity Diagram Kelola Supplier (Edit)

3.2.2.9. Activity Diagram Kelola Supplier (Hapus)

Activity diagram kelola supplier (hapus) merupakan langkah dimana *admin* ingin menghapus *supplier*.

Langkah pertama yakni *login* untuk memasuki halaman utama. Setelah itu pilih halaman *supplier*. Setelah masuk dihalaman *supplier* lalu pilih *supplier* yang akan hapus setelah itu pilih *button* hapus. Maka sistem akan otomatis menghapus *supplier* yang telah ditentukan. Seperti yang disuguhkan pada Gambar 3.16.

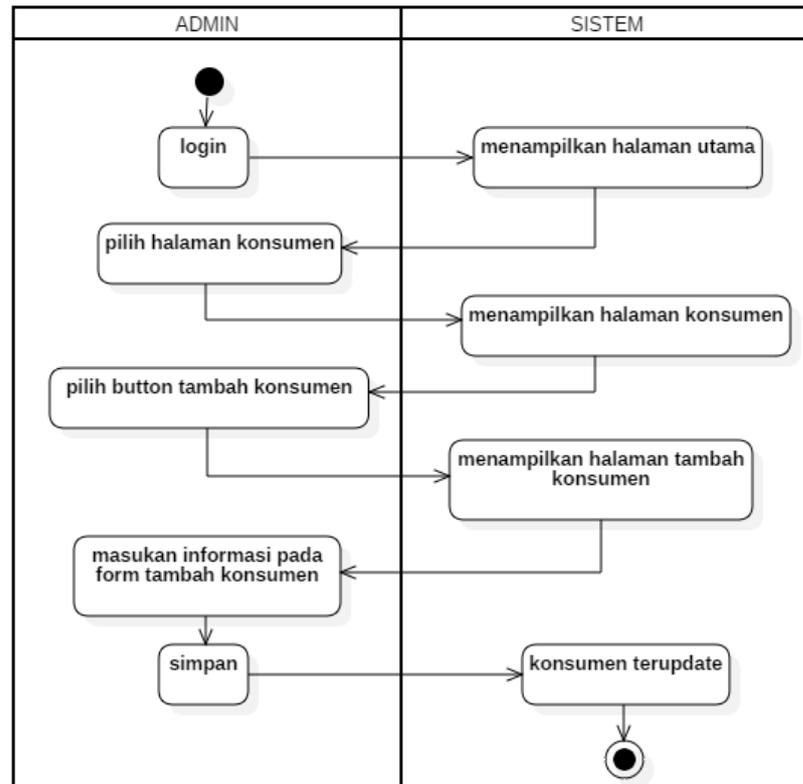


Gambar 3.15 *Activity Diagram* Kelola *Supplier* (Hapus)

3.2.2.10. *Activity Diagram* Kelola Konsumen (Tambah)

Activity diagram kelola konsumen (tambah) merupakan langkah dimana *admin* ingin menambahkan konsumen yang akan ditambahkan untuk proses penjualan nantinya.

Langkah pertama yakni *login* untuk memasukii halaman utama. Setelah itu pilih halaman konsumen. Setelah masuk dihalam konsumen lalu pilih *button* tambah. Maka sistem akan mengalihkan di halaman tambah konsumen. Masukan informasi mengenai konsumen yang akan ditambah lalu simpan. konsumen akan *terupdate*. Seperti yang disuguhkan pada Gambar 3.17.

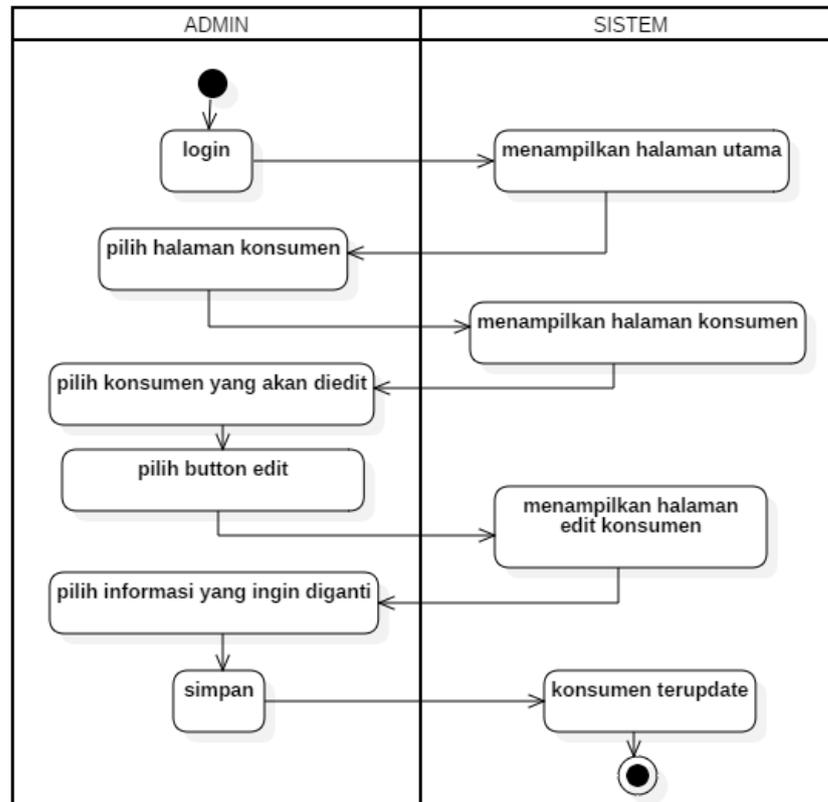


Gambar 3.16 *Activity Diagram* Kelola Konsumen (Tambah)

3.2.2.11. *Activity Diagram Kelola Konsumen (Edit)*

Activity diagram kelola konsumen (*edit*) merupakan langkah dimana *admin* ingin mengedit konsumen yang akan *diedit*.

Langkah pertama yakni login untuk memasukii halaman utama. Setelah itu pilih halaman konsumen. Setelah masuk dihalaman konsumen lalu pilih konsumen yang akan diedit lalu pilih *button edit*. Maka sistem akan mengalihkan di halaman *edit* konsumen. Masukan informasi mengenai konsumen yang akan diganti lalu simpan, *konsumen* akan *terupdate*. Seperti yang disuguhkan pada Gambar 3.18.

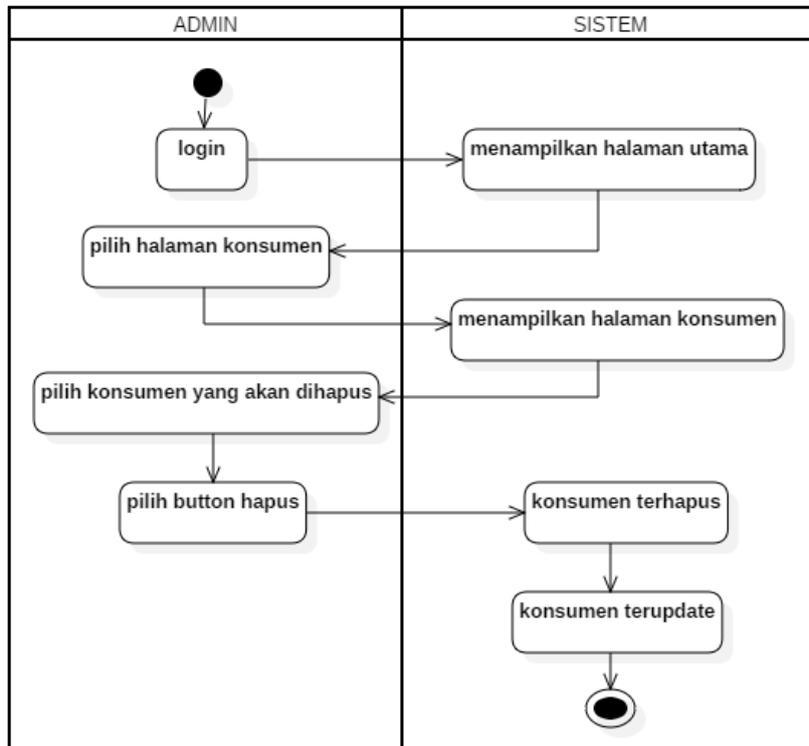


Gambar 3.17 Activity Diagram Kelola Konsumen (*Edit*)

3.2.2.12. Activity Diagram Kelola Konsumen (*Hapus*)

Activity diagram kelola konsumen (*hapus*) merupakan langkah dimana *admin* ingin menghapus konsumen.

Langkah pertama yakni *login* untuk memasuki halaman utama. Setelah itu pilih halaman konsumen. Setelah masuk dihalaman konsumen lalu pilih konsumen yang akan hapus setelah itu pilih *button* hapus. Maka sistem akan otomatis menghapus konsumen yang telah ditentukan. Seperti yang disuguhkan pada Gambar 3.19.

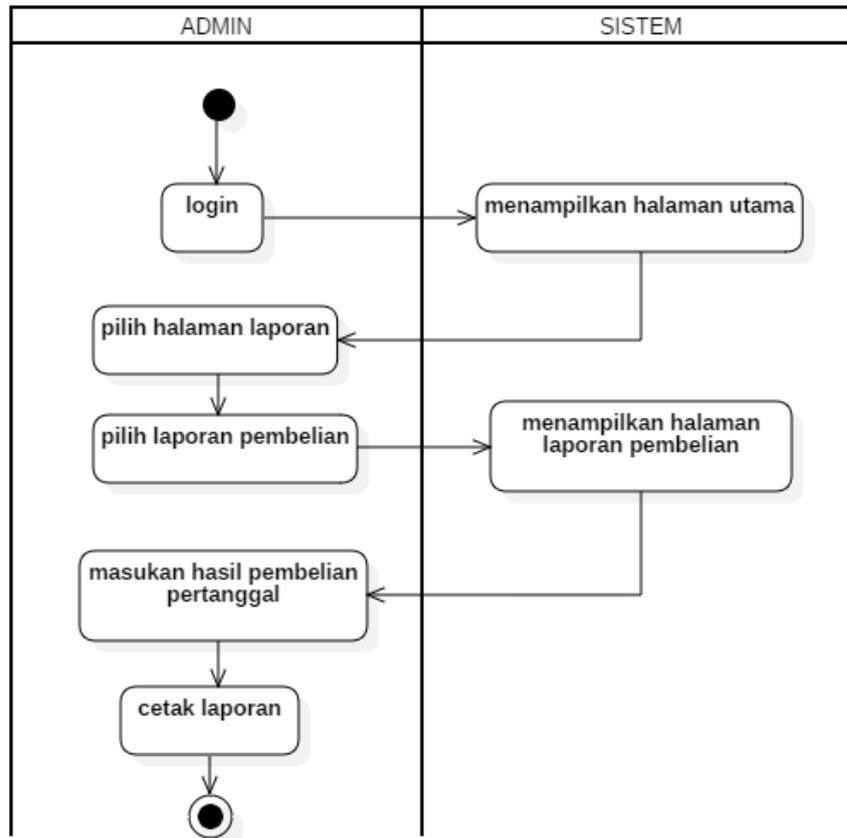


Gambar 3.18 *Activity Diagram* Kelola Konsumen (Hapus)

3.2.2.13. *Activity Diagram* Laporan Pembelian Beras

Activity diagram laporan pembelian beras merupakan langkah dimana *admin* ingin mengetahui laporan pembelian yang telah dilakukan sebelumnya dengan tanggal yang telah ditentukan.

Langkah pertama yakni login untuk memasuki halaman utama. Setelah itu pilih halaman laporan. Setelah masuk di halaman laporan pilih laporan pembelian beras. *Setting* tanggal yang akan ditentukan untuk mengetahui laporan pembelian pada tanggal yang ingin ditentukan. Setelah muncul hasilnya lalu cetak dan hasil *output* pun diserahkan pada kepala gudang. Seperti yang disuguhkan pada Gambar 3.20.

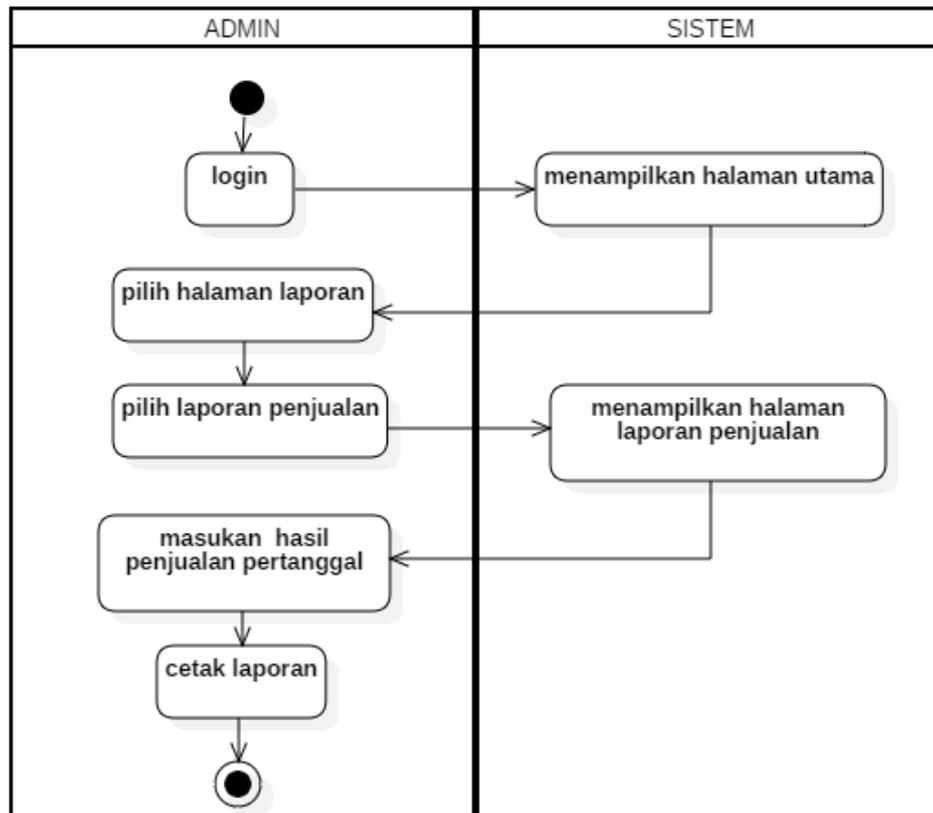


Gambar 3.19 Activity Diagram Laporan Pembelian Beras

3.2.2.14. Activity Diagram Laporan Penjualan Beras

Activity diagram laporan penjualan beras merupakan langkah dimana *admin* ingin mengetahui laporan pembelian yang telah dilakukan sebelumnya dengan tanggal yang telah ditentukan.

Langkah pertama yakni *login* untuk memasukii halaman utama. Setelah itu pilih halaman laporan. Setelah masuk dihalaman laporan pilih laporan penjualan beras. *Setting* tanggal yang akan ditentukan untuk mengetahui laporan penjualan pada tanggal yang ingin ditentukan. Setelah muncul hasilnya lalu cetak dan hasil *output* pun diserahkan pada kepala gudang. Seperti yang disuguhkan pada Gambar 3.21.

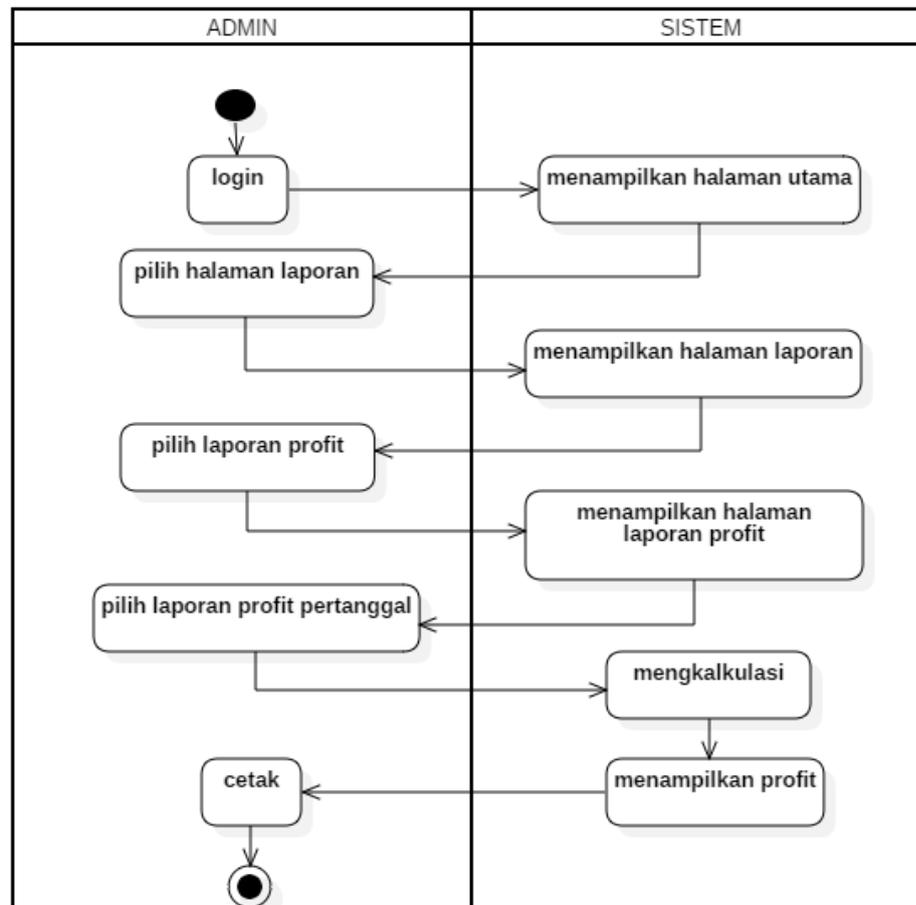


Gambar 3.20 Activity Diagram Laporan Penjualan Beras

3.2.2.15. Activity Diagram Laporan Profit Pendapatan

Activity diagram laporan pembelian beras merupakan langkah dimana *admin* ingin mengetahui laporan profit pendapatan yang didapatkan dalam jangka waktu yang ditentukan.

Langkah pertama yakni *login* untuk memasukii halaman utama. Setelah itu pilih halaman laporan. Setelah masuk dihalaman laporan pilih laporan profit . *Setting* tanggal yang akan ditentukan untuk mengetahui laporan profit pada tanggal yang ingin ditentukan. Setelah muncul hasilnya lalu cetak dan hasil *output* pun diserahkan pada kepala gudang. Seperti yang disuguhkan pada Gambar 3.22



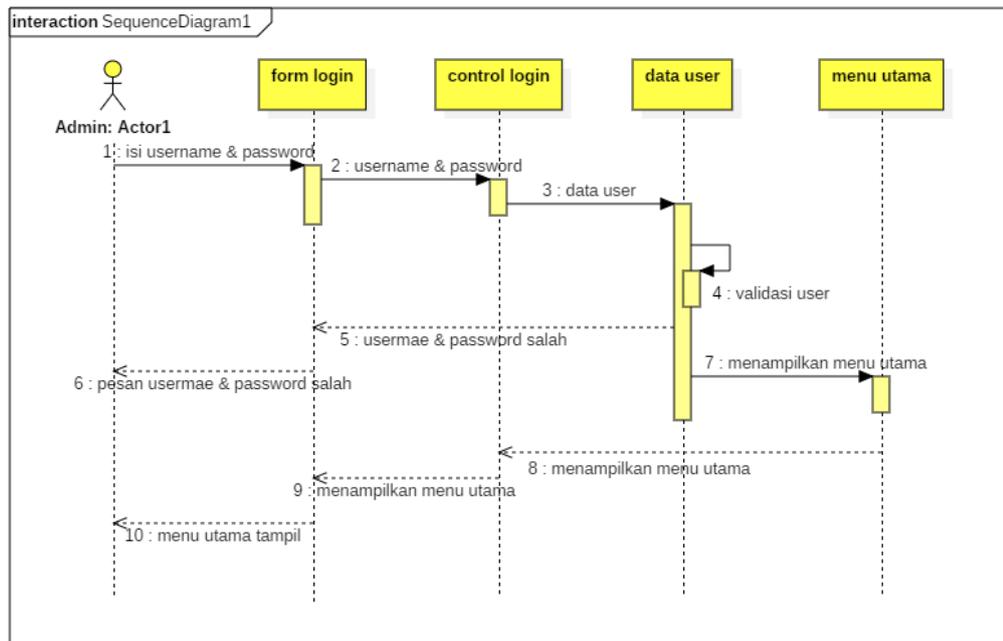
Gambar 3.21 *Activity Diagram* Laporan Profit Pendapatan

3.2.4 *Sequence Diagram*

Sequence Diagram secara grafis menggambarkan bagaimana objek berinteraksi dengan satu sama lain melalui pesan pada eksekusi sebuah *use case* atau operasi, seperti menggambarkan bagaimana pesan terkirim dan diterima diantara objek.

3.2.3.1. *Sequence Diagram Login*

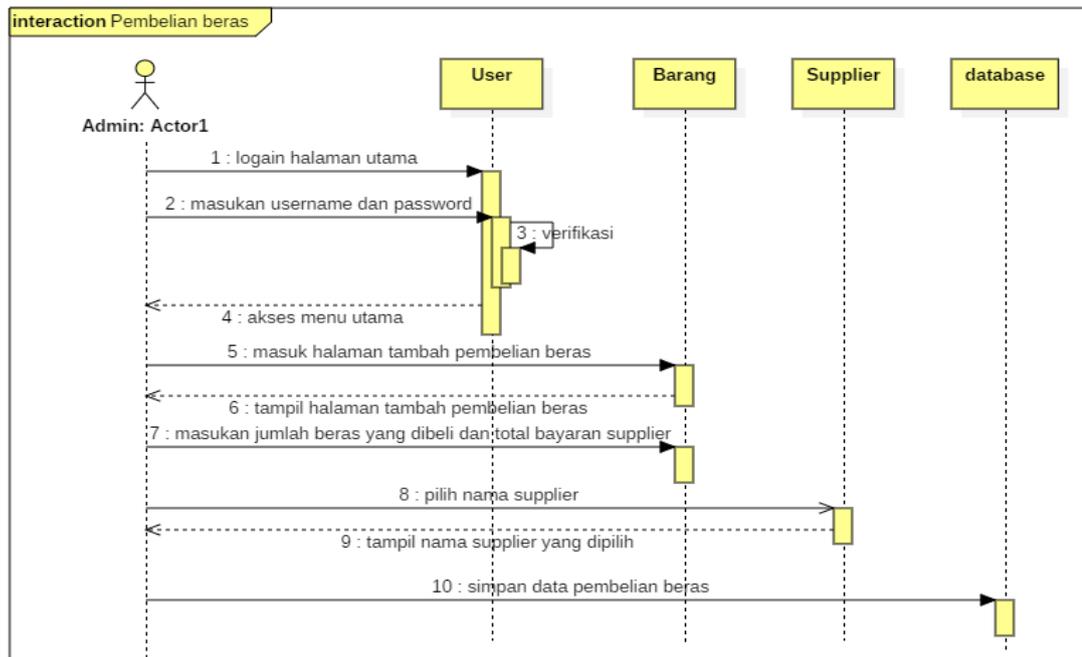
Sequence diagram ini menerangkan tentang hubungan yang terjadi antara *admin* gudang dengan object dalam proses *login*. Aktor melakukan *login user* terhadap sistem, kemudian sistem memverifikasi *username* dan *password* tersebut kemudian masuk ke halaman utama. Seperti yang terlihat pada Gambar 3.23.



Gambar 3.22 *Sequence Diagram Login*

3.2.3.2. *Sequence Diagram Pembelian Beras*

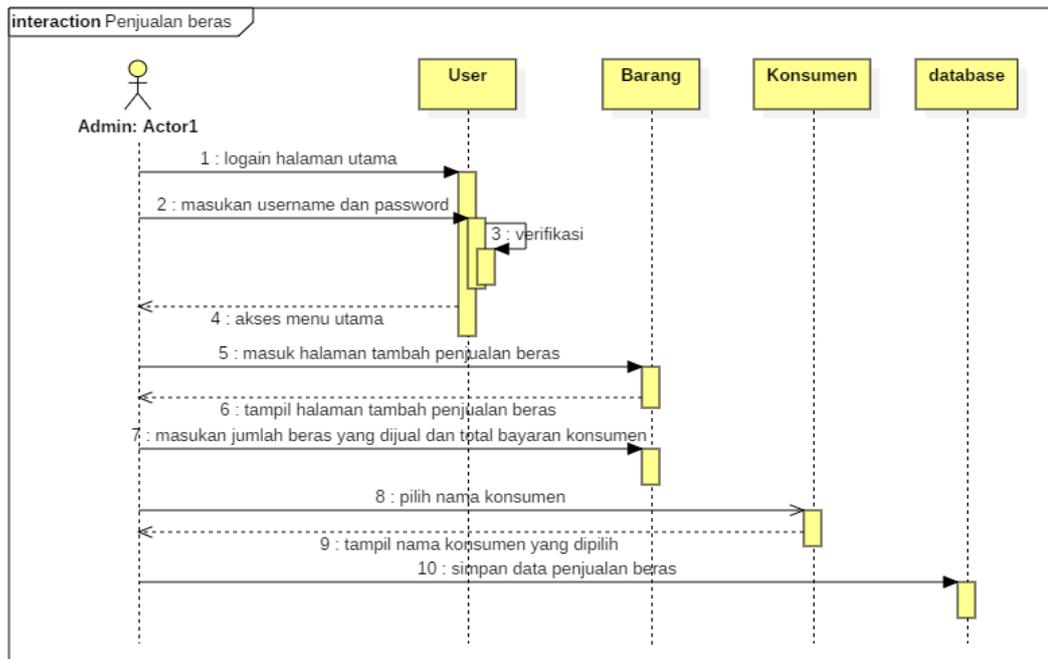
Sequence Diagram ini menerangkan tentang hubungan yang terjadi antara *admin* gudang dengan *object* dalam proses *input* data pembelian beras. Aktor melakukan *login user* terhadap sistem, kemudian sistem memverifikasi *username* dan *password* tersebut kemudian masuk ke halaman utama. Kemudian aktor memilih *input* data barang masuk yang kemudian dilanjutkan dengan memasukkan data-data beras masuk, totalan pembayaran dan data *supplier* yang menyuplai beras tersebut. Kemudian sistem menyimpan data beras tersebut kedalam *database*. Seperti yang dilihat pada Gambar 3.24



Gambar 3.23 *Sequence Diagram* Pembelian Beras

3.2.3.3. *Sequence Diagram* Penjualan Beras

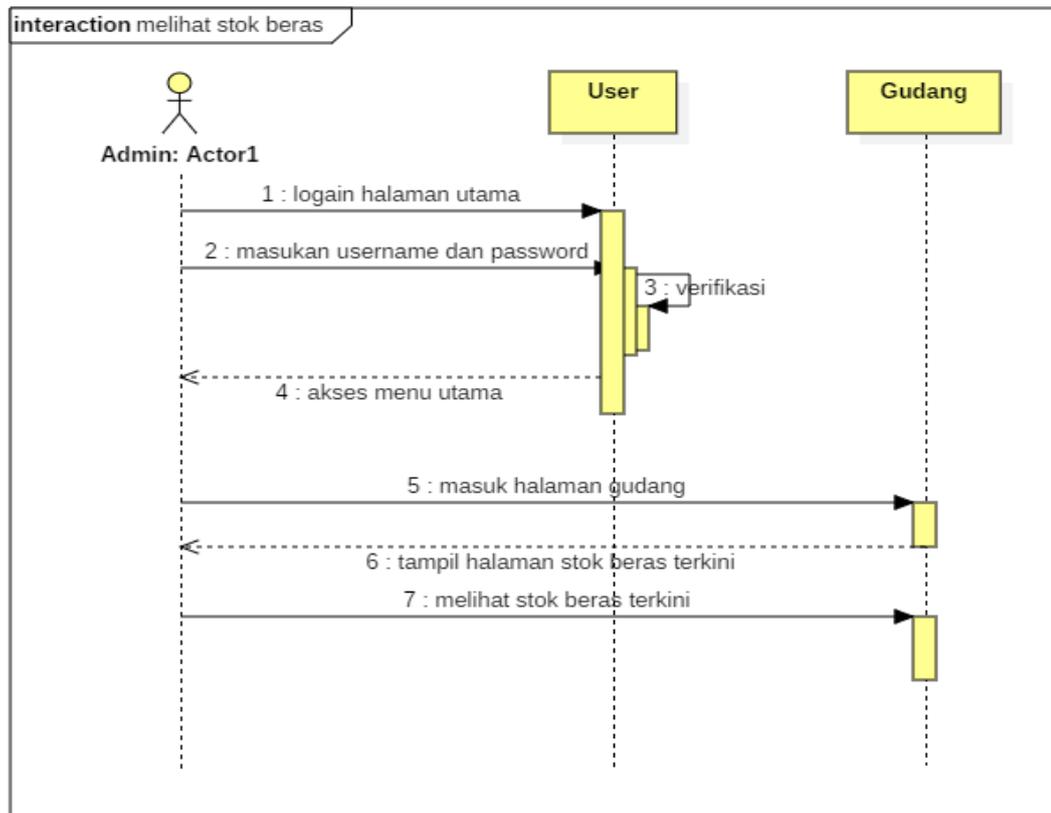
Sequence diagram ini menerangkan tentang hubungan yang terjadi antara *admin* gudang dengan *object* dalam proses *input* data penjualan beras. Aktor melakukan *login user* terhadap sistem, kemudian sistem memverifikasi *username* dan *password* tersebut kemudian masuk ke halaman utama. Kemudian aktor memilih input data barang masuk yang kemudian dilanjutkan dengan memasukkan data-data beras keluar, totalan pembayaran dan data konsumen yang membeli beras tersebut. Kemudian sistem menyimpan data beras tersebut kedalam *database*. Seperti yang terlihat pada Gambar 3.25.



Gambar 3.24 *Sequence Diagram* Penjualan Beras

3.2.3.4. *Sequence Diagram* Cek Stok Beras

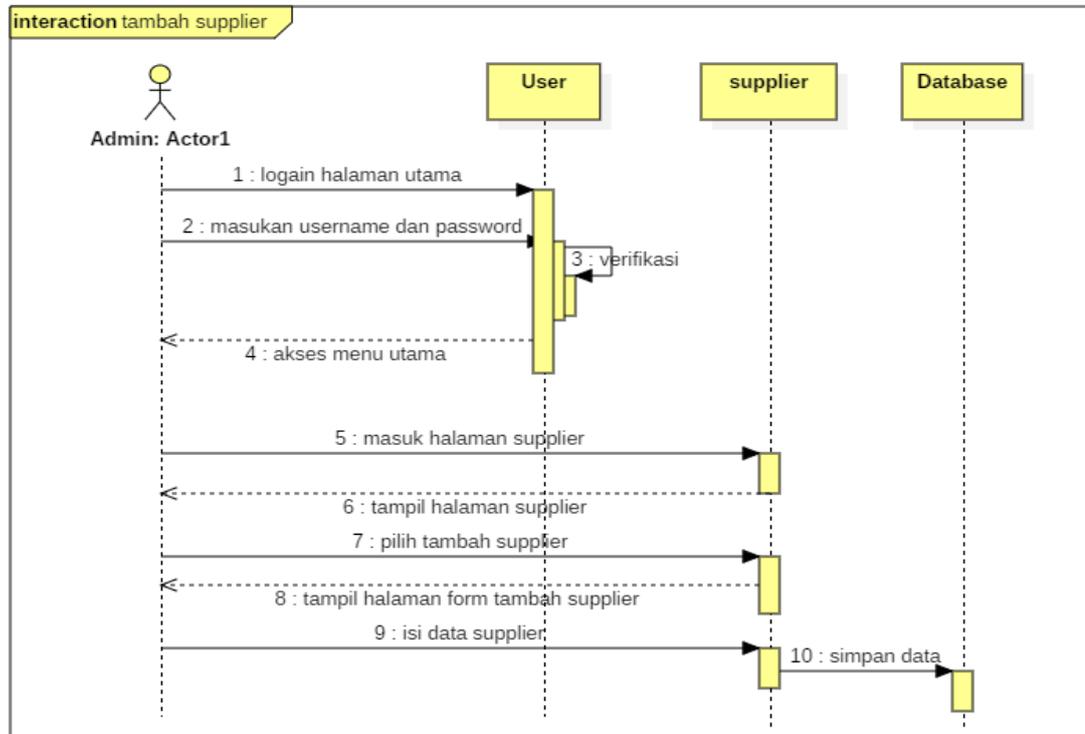
Sequence diagram ini menerangkan hubungan antara aktor *admin* gudang dengan *object* yang terkait dalam proses cek beras. Aktor melakukan *login* terhadap sistem yang kemudian sistem akan memverifikasi *username* dan *password* tersebut. Kemudian di halaman utama aktor memilih halaman gudang dan kemudian sistem akan menampilkan ketersediaan beras yang dimiliki CV AGUNG JAYA. Seperti yang terlihat pada Gambar 3.26.



Gambar 3.25 *Sequence Diagram* Cek Stok Beras

3.2.3.5. *Sequence Diagram Input Supplier*

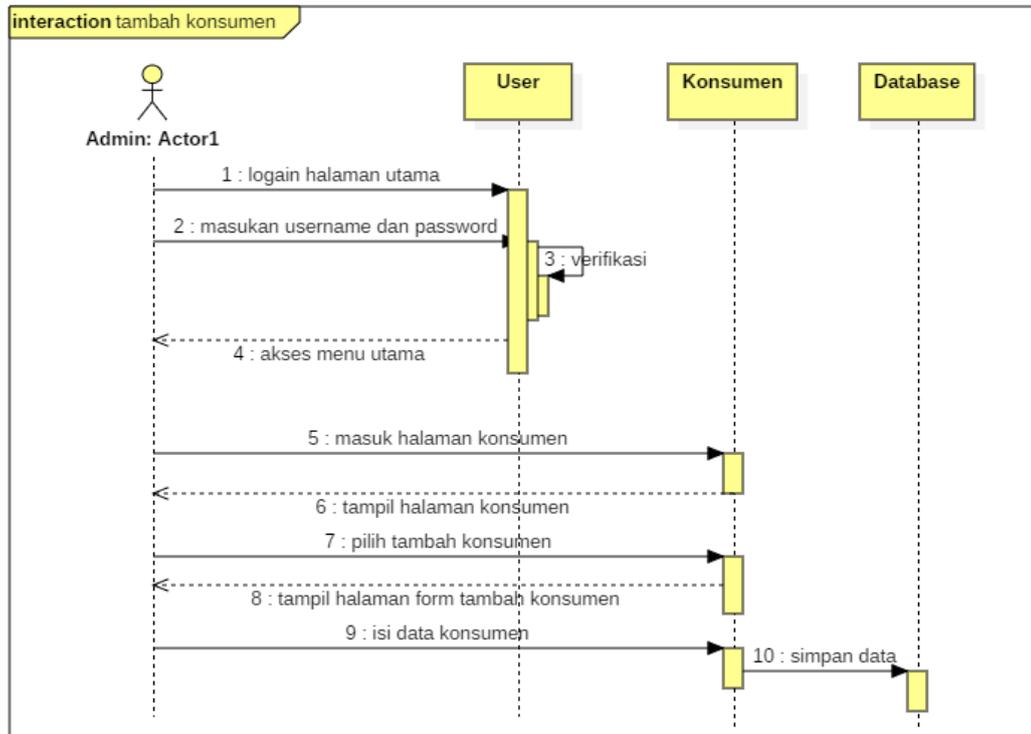
Sequence diagram ini menerangkan tentang hubungan yang terjadi antara *admin* gudang dengan *object* dalam menambah data *supplier*. Aktor melakukan *login user* terhadap sistem, kemudian sistem memverifikasi *username* dan *password* tersebut kemudian masuk ke halaman utama. Didalam halaman *supplier*, aktor memilih *button* tambah data *supplier*, kemudian aktor memasukan data *supplier* yang ingin ditambahkan lalu mengklik tombol simpan yang kemudian data *supplier* tersebut tersimpan dalam *database supplier*. Seperti yang terlihat pada Gambar 3.27.



Gambar 3.26 *Sequence Diagram* Tambah Supplier

3.2.3.6. *Sequence Diagram* Input Konsumen

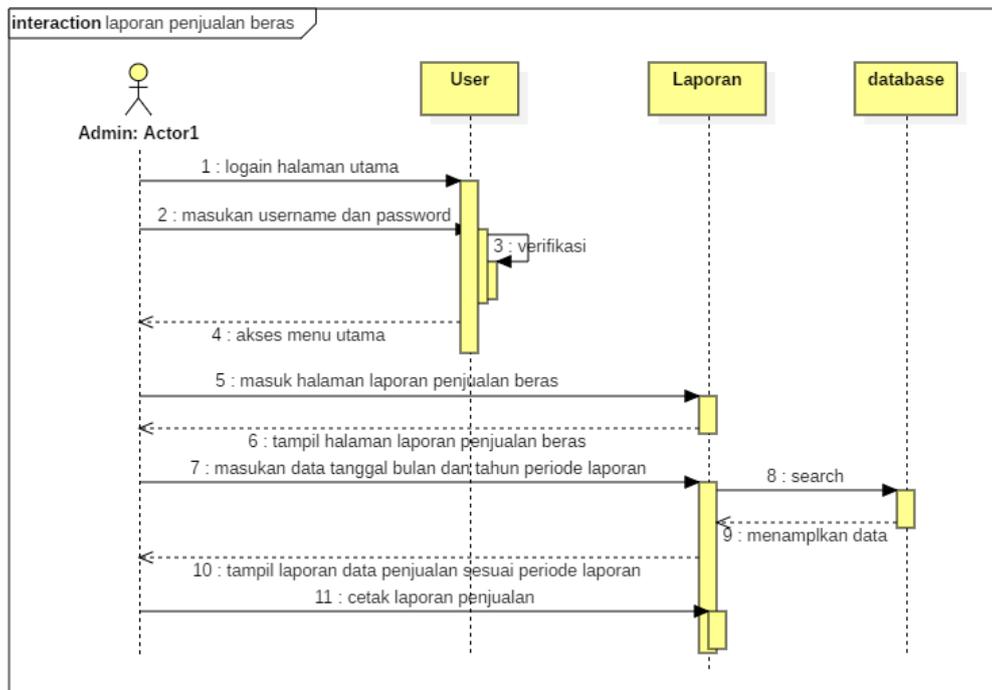
Sequence diagram ini menerangkan tentang hubungan yang terjadi antara *admin* gudang dengan *object* dalam menambah data konsumen. Aktor melakukan *login user* terhadap sistem, kemudian sistem memverifikasi *username* dan *password* tersebut kemudian masuk ke halaman utama. Didalam halaman konsumen, aktor memilih *button* tambah data konsumen, kemudian aktor memasukkan data konsumen yang ingin ditambahkan lalu mengklik tombol simpan yang kemudian data konsumen tersebut tersimpan dalam *database* konsumen. Seperti yang terlihat pada Gambar 3.28.



Gambar 3.27 *Sequence Diagram* Tambah Konsumen

3.2.3.7. *Sequence Diagram* Laporan Penjualan

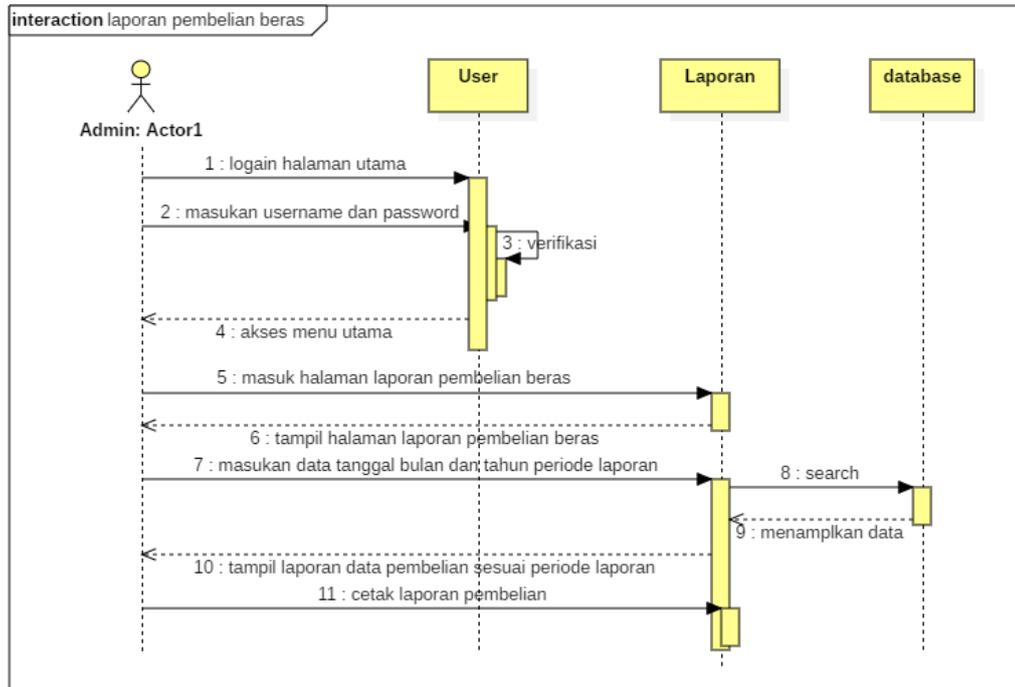
Sequence diagram ini menerangkan hubungan antara aktor *admin* gudang dengan *object* yang terkait dalam proses laporan data penjualan beras. Aktor melakukan *login* terhadap sistem yang kemudian sistem akan memverifikasi *username* dan *password* tersebut. Kemudian di halaman utama aktor memilih halaman laporan penjualan beras. Kemudian aktor memasukkan tanggal periode laporan yang kemudian dilanjutkan dengan mengklik tombol *search*. Sistem akan menampilkan laporan data penjualan beras sesuai periode yang diinputkan oleh aktor. Kemudian aktor dapat mengklik tombol *print* apabila ingin mencetak hasil laporan terkait. Seperti yang terlihat pada Gambar 3.29.



Gambar 3.28 *Sequence Diagram* Laporan Penjualan Beras

3.2.3.8. *Sequence Diagram* Laporan Pembelian

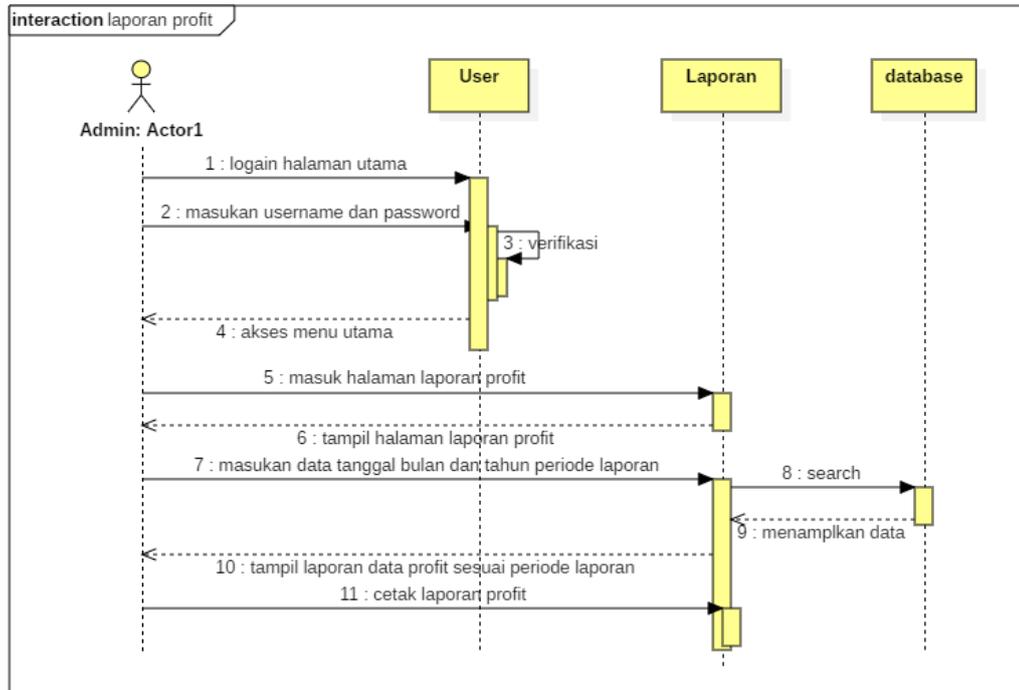
Sequence diagram ini menerangkan hubungan antara aktor *admin* gudang dengan *object* yang terkait dalam proses laporan data pembelian beras. Aktor melakukan *login* terhadap sistem yang kemudian sistem akan memverifikasi *username* dan *password* tersebut. Kemudian di halaman utama aktor memilih halaman laporan pembelian beras. Kemudian aktor memasukkan tanggal periode laporan yang kemudian dilanjutkan dengan mengklik tombol *search*. Sistem akan menampilkan laporan data pembelian beras sesuai periode yang *diinputkan* oleh aktor. Kemudian aktor dapat mengklik tombol *print* apabila ingin mencetak hasil laporan terkait. Seperti yang terlihat pada Gambar 3.30.



Gambar 3.29 *Sequence Diagram* Laporan Pembelian Beras

3.2.3.9. *Sequence Diagram* Profit

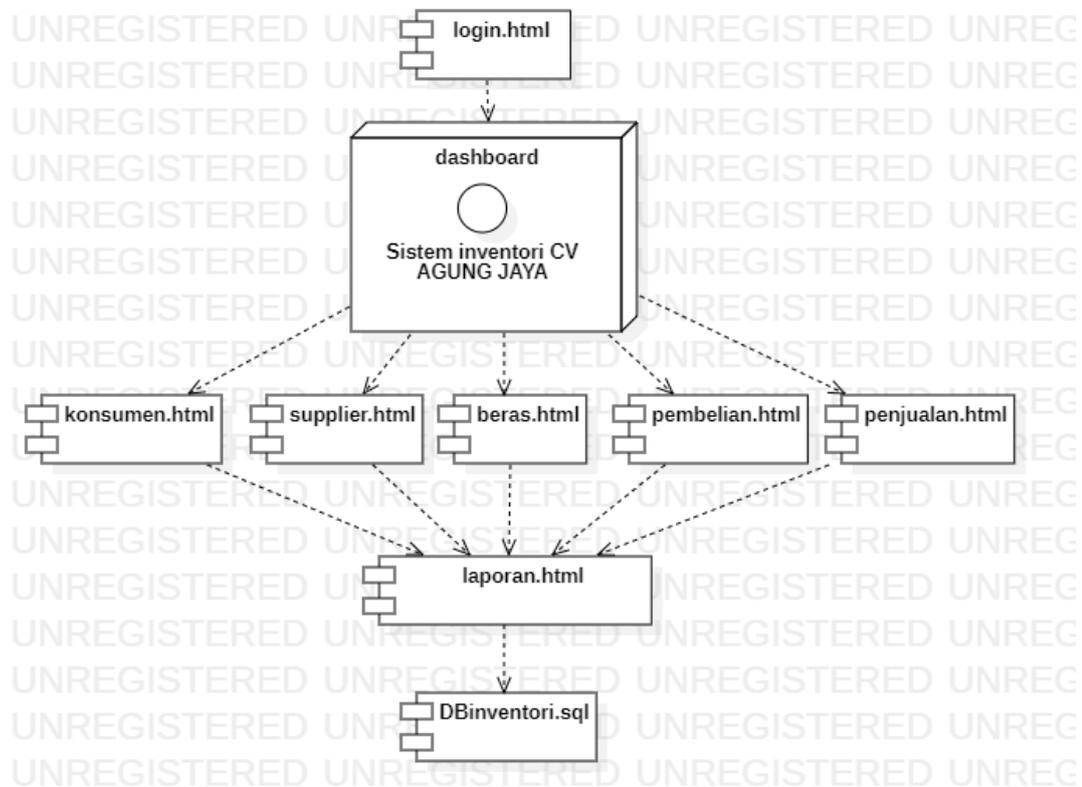
Sequence diagram ini menerangkan hubungan antara aktor *admin* gudang dengan *object* yang terkait dalam proses laporan profit keuntungan. Aktor melakukan *login* terhadap sistem yang kemudian sistem akan memverifikasi *username* dan *password* tersebut. Kemudian di halaman utama aktor memilih halaman laporan profit. Kemudian aktor memasukkan tanggal periode laporan yang kemudian dilanjutkan dengan mengklik tombol *search*. Sistem akan menampilkan laporan data profit keuntungan sesuai periode yang diinputkan oleh aktor. Kemudian aktor dapat mengklik tombol *print* apabila ingin mencetak hasil laporan terkait. Seperti yang terlihat pada Gambar 3.31.



Gambar 3.30 *Sequence Diagram* Laporan Profit

3.2.4 *Component Diagram*

Component Diagram merupakan pemecahan suatu sistem menjadi komponen-komponen serta menampilkan hubungan dengan antarmuka atau pemecahan komponen menjadi struktur yang lebih renah. *Component diagram* untuk perancangan sistem yang diusulkan dapat dilihat pada Gambar 3.32 dibawah ini.

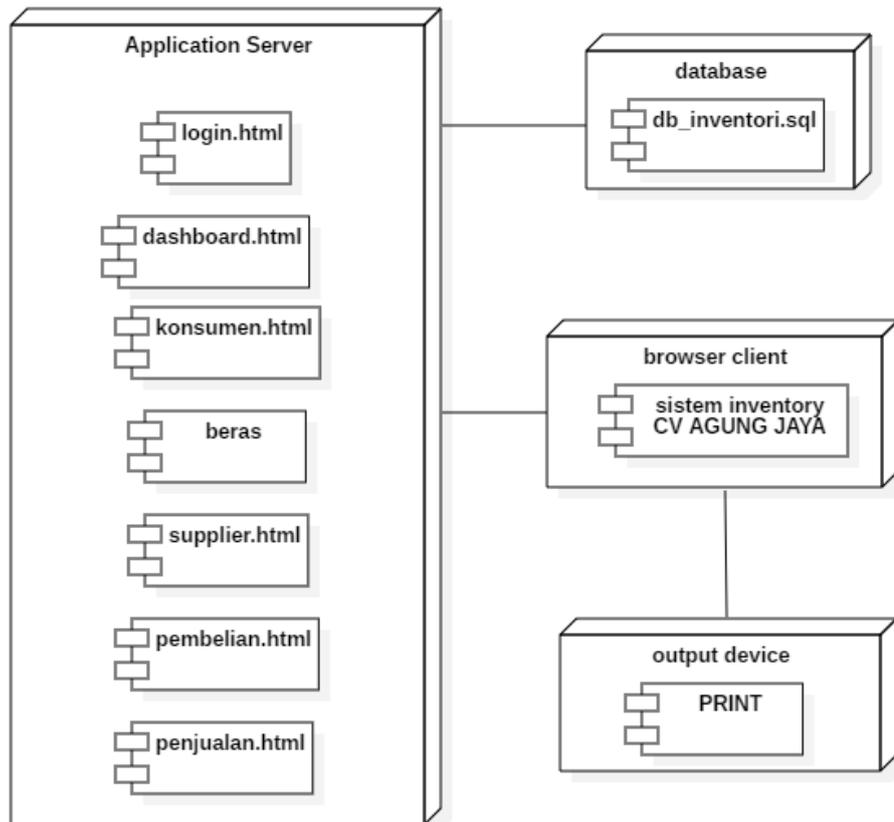


Gambar 3.31 Component Diagram

Nama Komponen	Deskripsi
login	Komponen untuk antarmuka form data login.
dashboard	Komponen untuk antarmuka form data menu utama
konsumen	Komponen untuk antarmuka form data konsumen
supplier	Komponen untuk antarmuka form data supplier
beras	Komponen untuk antarmuka form data beras
penjualan	Komponen untuk antarmuka form data penjualan
pembelian	Komponen untuk antarmuka form data pembelian
laporan	Komponen untuk antarmuka form data pengolahan laporan
DBinventori	Komponen untuk penyimpanan dan koneksi data ke database.

3.2.5 Deployment Diagram

Deployment diagram dibawah ini menunjukkan tata letak sebuah sistem secara fisik, menampakan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware* pada sistem *inventory* CV AGUNG JAYA yang disajikan pada Gambar 3.32 dibawah ini.



Gambar 3.32 *Deployment Diagram*

Tabel 3.8 Deskripsi Deployment Diagram

Node	Deskripsi
<<Application Server>> Sistem inventori Beras CV AGUNG JAYA	Node yang berisi komponen komponen sistem yang akan dirancang .
<<Database Inventori>>	Node yang berisi database yang digunakan pada sistem Inventori.
<<Browser Client>> Sistem inventori Beras CV AGUNG JAYA	Node yang berisi aplikasi sistem inventori beras yang dikonfigurasi pada komputer client melalui mesin pencari browser engine.
<<Output Device>> Printer	Node yang berisi perangkat keras yang digunakan untuk mencetak laporan data inventori barang.

3.3 Rancangan Interface

Perancangan antarmuka atau *interface* bertujuan untuk memberikan gambaran tentang sistem *inventory* CV AGUNG JAYA yang akan dibangun sehingga akan mempermudah dalam mengimplementasikan serta akan memudahkan dalam pembuatan aplikasi ini.

3.3.1. Desain Halaman Login

Halaman *login* merupakan halaman yang digunakan oleh *admin* untuk masuk ke halaman sistem *inventory* beras . Jika *admin* memasukan *username* dan *password* yang benar maka *admin* dapat masuk ke halaman *dashboard* seperti yang terlihat pada Gambar 3.35.



HALAMAN LOGIN

Username

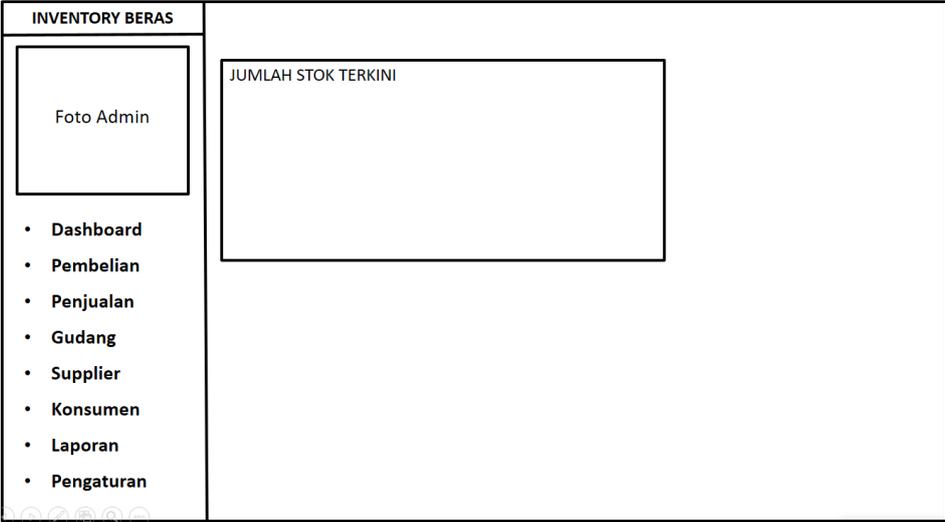
Password

LOGIN

Gambar 3.33 Desain Halaman *Login*

3.3.2. Desain Halaman Utama

Setelah admin melakukan login maka akan diarahkan pada halaman *dashboard* dimana halaman *dashboard* ini berisi jumlah stok terkini seperti yang dilihat pada Gambar 3.36.



INVENTORY BERAS

Foto Admin

- Dashboard
- Pembelian
- Penjualan
- Gudang
- Supplier
- Konsumen
- Laporan
- Pengaturan

JUMLAH STOK TERKINI

Gambar 3.34 Desain Halaman Utama

3.3.3. Desain Halaman Tambah Pembelian

Halaman tambah pembelian ini berfungsi untuk *admin* menambahkan data pembelian disaat *supplier* memasok beras pada CV AGUNG JAYA. Didalam

tambah data pembelian ini *admin* mengisi *form* informasi mengenai pembelian yang sedang dilakukan seperti yang dilihat pada Gambar 3.37.

Gambar 3.35 Desain Halaman Tambah Pembelian

3.3.4. Desain Halaman Data Pembelian

Halaman data pembelian berisikan data-data mengenai pembelian yang dilakukan CV.AGUNG JAYA. Di dalam menu ini *admin* dapat mencetak detail pembelian yang nantinya juga dapat dijadikan sebagai laporan pembelian seperti yang dilihat pada Gambar 3.38.

No	Kd_Pembelian	Tgl_pembelian	Kd_supplier	Nama supplier	Jumlah pembelian	Total pembelian	aksi
1	PEM00001	Hh/bb/tt	9999999	Xxxxxxxxx	999999	999999	Detail
2	PEM00002	Hh/bb/tt	9999999	xxxxxxxxx	999999	999999	detail

Gambar 3.36 Desain Halaman Data Pembelian

3.3.5. Desain Halaman Tambah Penjualan

Desain halaman tambah penjualan berfungsi ketika *admin* ingin menambahkan data informasi mengenai penjualan yang dilakukan CV AGUNG JAYA, seperti yang dilihat pada Gambar 3.39.

INVENTORY BERAS	HALAMAN TAMBAH DATA PENJUALAN	
Foto Admin	<input type="text" value="Kd_beras"/>	<input type="text" value="Kode penjualan"/>
<ul style="list-style-type: none">• Dashboard• Pembelian• Penjualan• Gudang• Supplier• Konsumen• Laporan• Pengaturan	<input type="text" value="Nama beras"/>	<input type="text" value="Tanggal penjualan (Hh/bb/tt)"/>
	<input type="text" value="Jumlah item"/>	<input type="text" value="Pilih pembeli"/>
	<input type="button" value="TAMBAH"/>	<input type="text" value="Total bayar"/>

Gambar 3.37 Desain Halaman Tambah Penjualan

3.3.6. Desain Halaman Data Penjualan

Halaman data penjualan berisikan data-data mengenai penjualan yang dilakukan CV.AGUNG JAYA. Di dalam menu ini *admin* dapat mencetak detail penjualan yang nantinya juga dapat dijadikan sebagai laporan penjualan seperti yang dilihat pada Gambar 3.40.

INVENTORY BERAS																									
Foto Admin	<p>HALAMAN DATA PENJUALAN</p> <table border="1"> <thead> <tr> <th>No</th> <th>Kd_Penjualan</th> <th>Tgl_penjualan</th> <th>Kd_konsumen</th> <th>Nama konsumen</th> <th>item</th> <th>Totalpenjualan</th> <th>aksi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>PEN00001</td> <td>Hh/bb/tt</td> <td>9999999</td> <td>Xxxxxxxxx</td> <td>99999</td> <td>Rp.999999</td> <td>Detail</td> </tr> <tr> <td>2</td> <td>PEN00002</td> <td>Hh/bb/tt</td> <td>9999999</td> <td>xxxxxxxxx</td> <td>99999</td> <td>Rp.999999</td> <td>detail</td> </tr> </tbody> </table>	No	Kd_Penjualan	Tgl_penjualan	Kd_konsumen	Nama konsumen	item	Totalpenjualan	aksi	1	PEN00001	Hh/bb/tt	9999999	Xxxxxxxxx	99999	Rp.999999	Detail	2	PEN00002	Hh/bb/tt	9999999	xxxxxxxxx	99999	Rp.999999	detail
No	Kd_Penjualan	Tgl_penjualan	Kd_konsumen	Nama konsumen	item	Totalpenjualan	aksi																		
1	PEN00001	Hh/bb/tt	9999999	Xxxxxxxxx	99999	Rp.999999	Detail																		
2	PEN00002	Hh/bb/tt	9999999	xxxxxxxxx	99999	Rp.999999	detail																		
<ul style="list-style-type: none"> • Dashboard • Pembelian • Penjualan • Gudang • Supplier • Konsumen • Laporan • Pengaturan 																									

Gambar 3.38 Desain Halaman Data Penjualan

3.3.7. Desain Halaman Gudang Beras

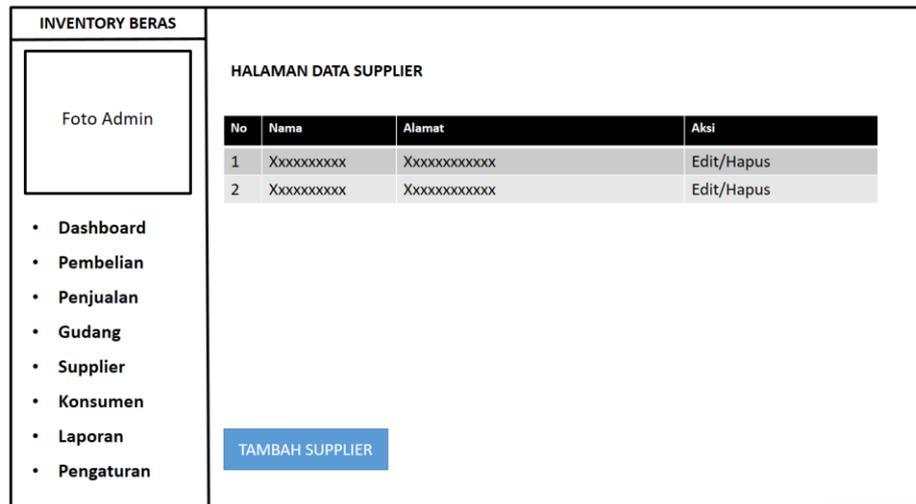
Halaman gudang beras ini merupakan halaman dimana *admin* dapat mengetahui stok beras yang ada di gudang. Setiap pembelian yang dilakukan akan otomatis ditambahkan didalam gudang ini dan saat melakukan penjualan stok di gudang juga otomatis berkurang seperti yang dilihat pada Gambar 3.41.

INVENTORY BERAS																									
Foto Admin	<p>HALAMAN DATA GUDANG BERAS</p> <table border="1"> <thead> <tr> <th>No</th> <th>Kode Barang</th> <th>Nama</th> <th>Satuan</th> <th>Harga Jual</th> <th>Harga Bell</th> <th>Stok</th> <th>aksi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>BRS00001</td> <td>Beras xxx</td> <td>Kg</td> <td>Rp.9999</td> <td>Rp.9999</td> <td>99</td> <td>Edit/hapus</td> </tr> <tr> <td>2</td> <td>BRS00002</td> <td>Beras xxx</td> <td>kg</td> <td>Rp.9999</td> <td>Rp.9999</td> <td>99</td> <td>Edit/hapus</td> </tr> </tbody> </table>	No	Kode Barang	Nama	Satuan	Harga Jual	Harga Bell	Stok	aksi	1	BRS00001	Beras xxx	Kg	Rp.9999	Rp.9999	99	Edit/hapus	2	BRS00002	Beras xxx	kg	Rp.9999	Rp.9999	99	Edit/hapus
No	Kode Barang	Nama	Satuan	Harga Jual	Harga Bell	Stok	aksi																		
1	BRS00001	Beras xxx	Kg	Rp.9999	Rp.9999	99	Edit/hapus																		
2	BRS00002	Beras xxx	kg	Rp.9999	Rp.9999	99	Edit/hapus																		
<ul style="list-style-type: none"> • Dashboard • Pembelian • Penjualan • Gudang • Supplier • Konsumen • Laporan • Pengaturan 																									

Gambar 3.39 Desain Halaman Gudang Beras

3.3.8. Desain Halaman *Supplier*

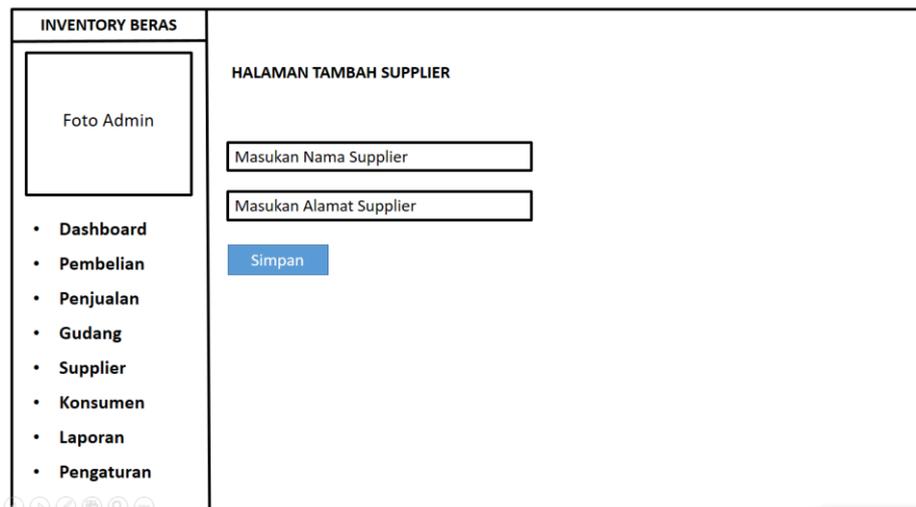
Halaman *supplier* berisi data-data *supplier* yang menjual beras pada CV AGUNG JAYA, seperti yang dilihat pada Gambar 3.42.



Gambar 3.40 Desain Halaman *Supplier*

3.3.9. Desain Halaman Tambah *Supplier*

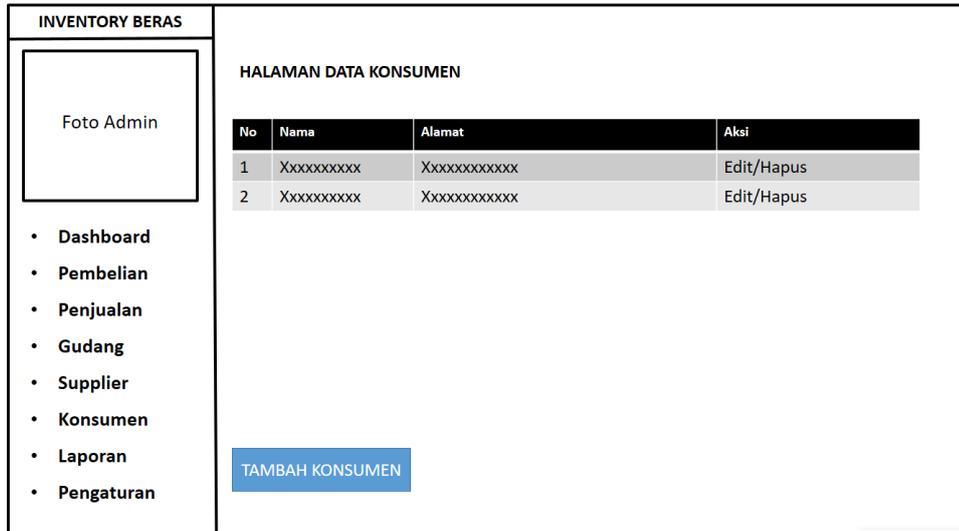
Desain Halaman Tambah *supplier* merupakan langkah admin dalam upaya menambahkan data *supplier* baru seperti yang dilihat pada Gambar 3.43.



Gambar 3.41 Desain Halaman Tambah *Supplier*

3.3.10. Desain Halaman Konsumen

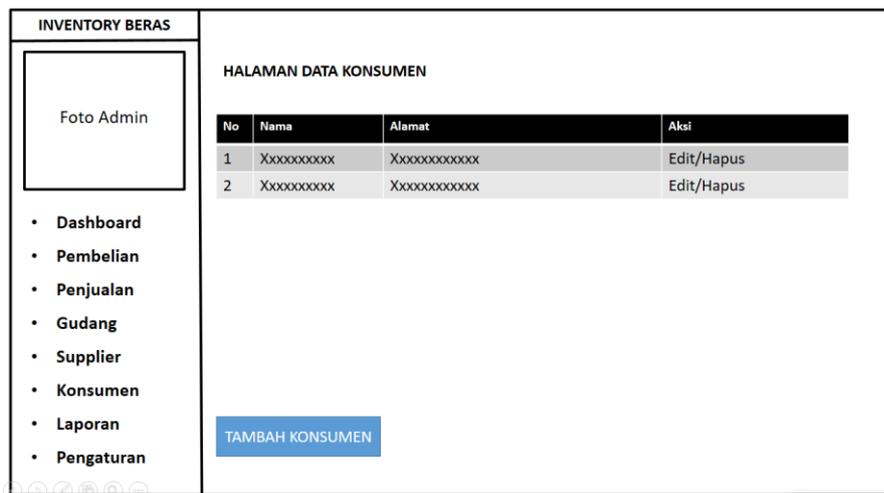
Halaman konsumen berisi data-data konsumen yang berlangganan pada CV AGUNG JAYA, seperti yang dilihat pada Gambar 3.44.



Gambar 3.42 Desain Halaman Konsumen

3.3.11. Desain Halaman Tambah Konsumen

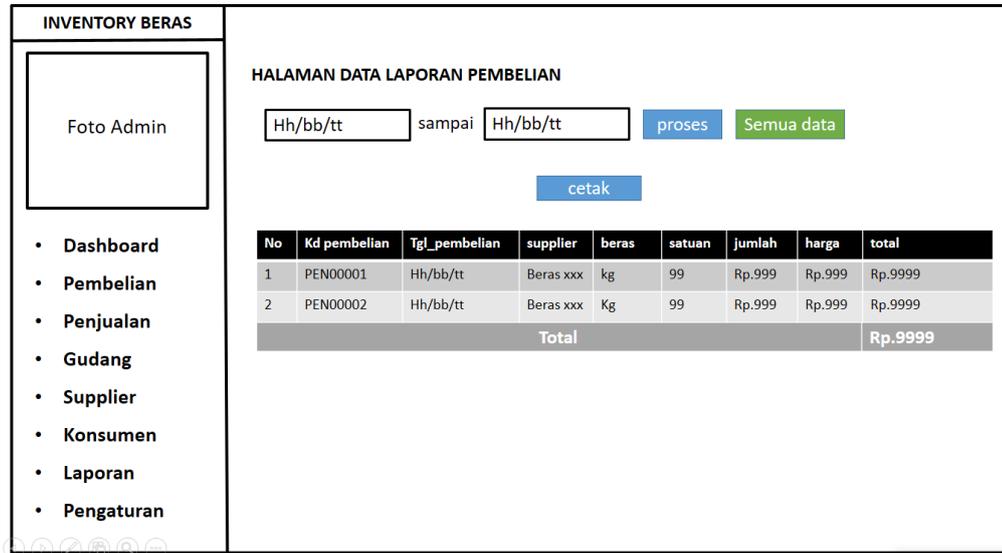
Desain halaman tambah konsumen merupakan langkah admin dalam upaya menambahkan data konsumen baru seperti yang dilihat pada Gambar 3.45.



Gambar 3.43 Desain Halaman Tambah Konsumen

3.3.12. Desain Halaman Laporan Pembelian

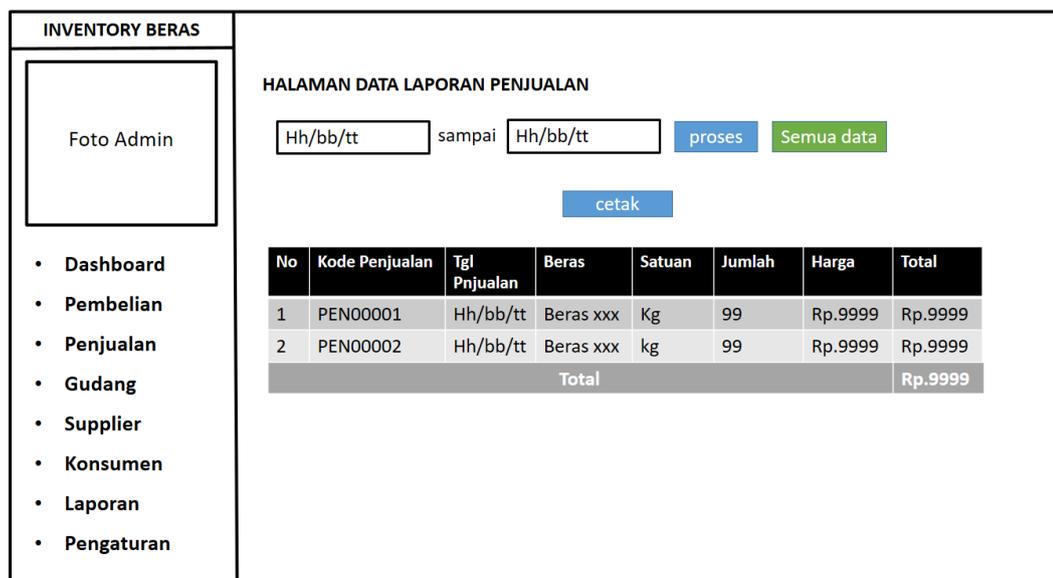
Didalam halaman laporan pembelian terdapat informasi mengenai laporan pembelian yang dilakukan CV AGUNG JAYA. Terdapat informasi pembelian yang dilakukan pertanggal dan dapat dicetak sebagai bukti laporan pembelian seperti yang dilihat pada Gambar 3.46.



Gambar 3.44 Desain Halaman Laporan Pembelian

3.3.13. Desain Halaman Laporan Penjualan

Didalam halaman laporan penjualan terdapat informasi mengenai laporan penjualan yang dilakukan CV AGUNG JAYA. Terdapat informasi penjualan yang dilakukan pertanggal dan dapat dicetak sebagai bukti laporan penjualan seperti yang dilihat pada Gambar 3.47.



Gambar 3.45 Desain Halaman Laporan Penjualan

3.3.14. Desain Halaman Laporan Stok

Laporan stok merupakan laporan terkini mengenai stok beras masuk dan keluar selama melakukan pembelian dan penjualan di CV AGUNG JAYA. Seperti yang terlihat pada Gambar 3.48.

Gambar 3.46 Desain Halaman Laporan Stok

3.3.15. Desain Halaman Laporan Profit

Laporan profit merupakan laporan mengenai keuntungan yang diperoleh CV AGUNG JAYA selama melakukan pembelian stok beras dan penjualan. Laporan profit ini juga bisa dicetak sebagai bukti laporan keuntungan CV AGUNG JAYA. Seperti yang dilihat pada Gambar 3.48.

No	Kd penjualan	Tgl_penjualan	beras	satuan	jumlah	Harga beli	Harga jual	profit
1	PEN00001	Hh/bb/tt	Beras xxx	kg	99	Rp.999	Rp.999	Rp.9999
2	PEN00002	Hh/bb/tt	Beras xxx	Kg	99	Rp.999	Rp.999	Rp.9999
Total								Rp.9999

Gambar 3.47 Desain Halaman Laporan Profit

3.3.16. Desain Halaman *Admin*

Halaman *admin* berisikan data *admin* yang bisa mengakses sistem inventori CV AGUNG JAYA seperti yang dilihat pada Gambar 3.49.

INVENTORY BERAS																			
Foto Admin	<p>HALAMAN DATA ADMIN</p> <table border="1"> <thead> <tr> <th>No</th> <th>Nama</th> <th>Email</th> <th>Password</th> <th>Foto</th> <th>Aksi</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>XXXXX</td> <td>XXXX@</td> <td>XXXX</td> <td><input type="text"/></td> <td>Edit/hapus</td> </tr> <tr> <td>2</td> <td>XXXXX</td> <td>XXXX@</td> <td>XXXX</td> <td><input type="text"/></td> <td>Edit/hapus</td> </tr> </tbody> </table> <p>TAMBAH ADMIN</p>	No	Nama	Email	Password	Foto	Aksi	1	XXXXX	XXXX@	XXXX	<input type="text"/>	Edit/hapus	2	XXXXX	XXXX@	XXXX	<input type="text"/>	Edit/hapus
No	Nama	Email	Password	Foto	Aksi														
1	XXXXX	XXXX@	XXXX	<input type="text"/>	Edit/hapus														
2	XXXXX	XXXX@	XXXX	<input type="text"/>	Edit/hapus														
<ul style="list-style-type: none"> • Dashboard • Pembelian • Penjualan • Gudang • Supplier • Konsumen • Laporan • Pengaturan 																			

Gambar 3.48 Desain Halaman *Admin*

3.3.17. Desain Halaman Tambah *Admin*

Halaman tambah *admin* merupakan halaman dimana *admin* ingin menambah *admin* baru untuk mengakses sistem CV AGUNG JAYA seperti yang dilihat pada Gambar 3.50.

INVENTORY BERAS	
Foto Admin	<p>HALAMAN TAMBAH ADMIN</p> <p>Email <input type="text"/></p> <p>Password <input type="text"/></p> <p>Nama <input type="text"/></p> <p>Pilih gambar <input type="text"/></p> <p>Simpan</p>
<ul style="list-style-type: none"> • Dashboard • Pembelian • Penjualan • Gudang • Supplier • Konsumen • Laporan • Pengaturan 	

Gambar 3.49 Desain Halaman Tambah *Admin*

3.3.18. Desain Halaman Perusahaan

Desain halaman perusahaan merupakan halaman data lengkap profil CV AGUNG JAYA seperti yang dilihat pada Gambar 3.51.

INVENTORY BERAS	
Foto Admin	HALAMAN PERUSAHAAN
<ul style="list-style-type: none">• Dashboard• Pembelian• Penjualan• Gudang• Supplier• Konsumen• Laporan• Pengaturan	<input type="text" value="Nama Perusahaan"/>
	<input type="text" value="Alamat"/>
	<input type="text" value="Nama Pemilik"/>
	<input type="text" value="Kota"/>
	<input type="button" value="Simpan"/>

Gambar 3.50 Desain Halaman Perusahaan