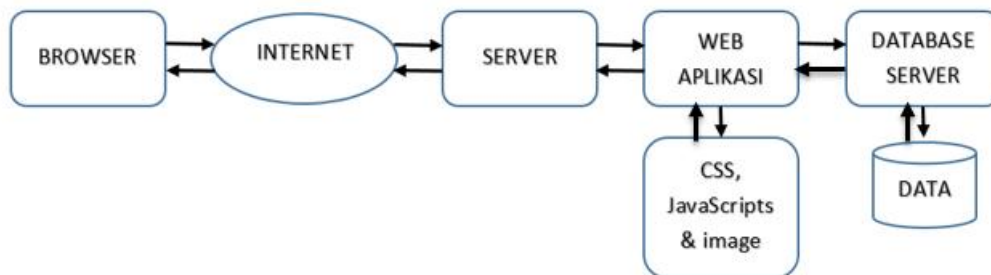


BAB 5 PERANCANGAN

5.1 Perancangan Arsitektur

Perancangan arsitektur sistem informasi Son Screen Printing dibangun dengan berbasis web agar dapat diakses oleh pada admin dimana saja dan kapan saja. Desain arsitektur sistem dapat dilihat pada Gambar 5.1 sebagai berikut:



Gambar 5.1 Desain Arsitektur Sistem Informasi Son Screen Printing

5.2 Perancangan Interaksi dan Kelas

Perancangan interaksi dan kelas ini merupakan kumpulan diagram yang sudah mendekati dengan bahasa pemrograman agar memudahkan implementasi. Pada sistem informasi ini terbagi menjadi 2 jenis pembuatan diagram yaitu (1) Membuat *sequence diagram* untuk mengetahui alur dan interaksi antar *class* dan dilengkapi dengan kejelasan mengenai *Model*, *View*, maupun *Controller* dari setiap *Use Case* pada *Use Case Diagram*; (2) Membuat *class diagram* untuk mengetahui relasi antar *class* yang terdiri dari *class* pada *controller* dan *class* pada *model* yang saling berinteraksi satu sama lain. Dengan adanya dua jenis diagram tersebut akan menjelaskan interaksi yang dilakukan antar *class* dan fungsi yang ada pada setiap *class*nya.

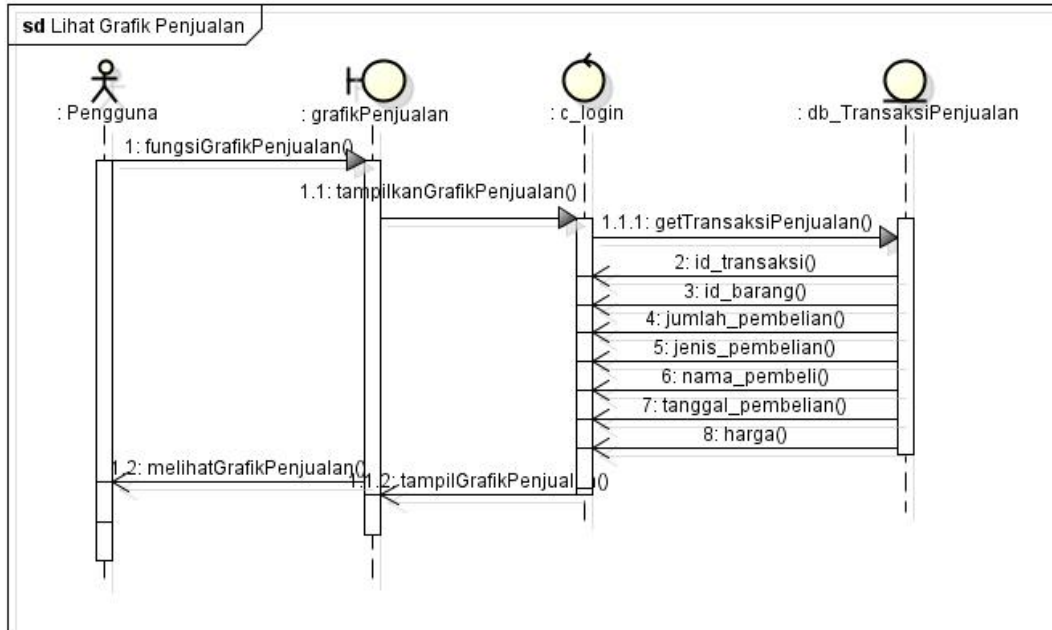
5.2.1 Sequence Diagram

Pada bagian ini menjelaskan mengenai gambaran perilaku pada sebuah *scenario* dan aktifitas detail dari setiap *use case* pada *use case diagram*. Hubungan antara aktor dan sistem akan jelas di samping itu akan diperlihatkan hubungan antar sistem yang terbagi menjadi *controller* dan *model* sehingga fungsi dari setiap *use case* akan terlihat disertai dengan langkah untuk menyelesaikan setiap *use case* pada *use case diagram*.

5.2.1.1 Sequence Diagram Lihat Grafik Penjualan

Pada *sequence diagram* lihat grafik penjualan dimulai ketika pengguna sistem menggunakan fungsi menu grafik penjualan dengan memanggil fungsi *tampilkanGrafikPenjualan* pada *controller c_login*. Kemudian pada *controller* dilakukan pengambilan data transaksi penjualan dari database melalui *model db_TransaksiPenjualan* dan *controller* akan menampilkan halaman grafik

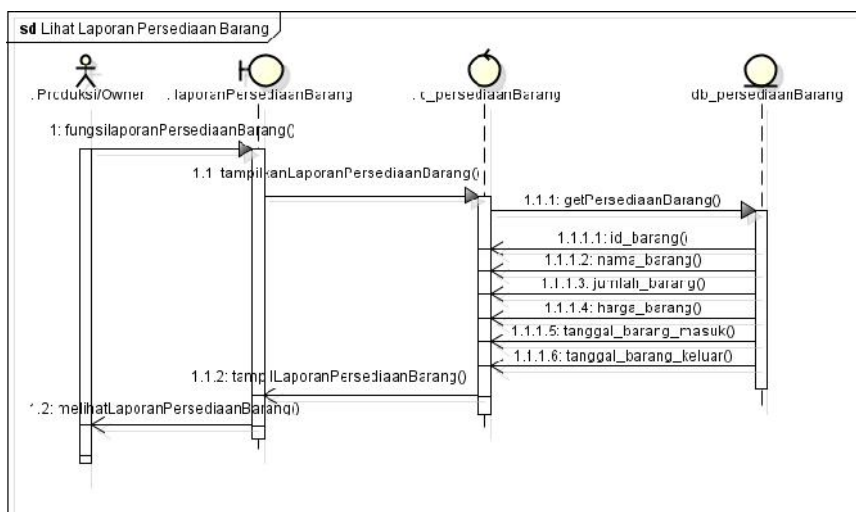
penjualan. Berikut ini merupakan *sequence diagram* lihat grafik penjualan yang ditunjukkan pada Gambar 5.3.



Gambar 5.2 Sequence Diagram Lihat Grafik Penjualan

5.2.1.2 Sequence Diagram Lihat Laporan Persediaan Barang

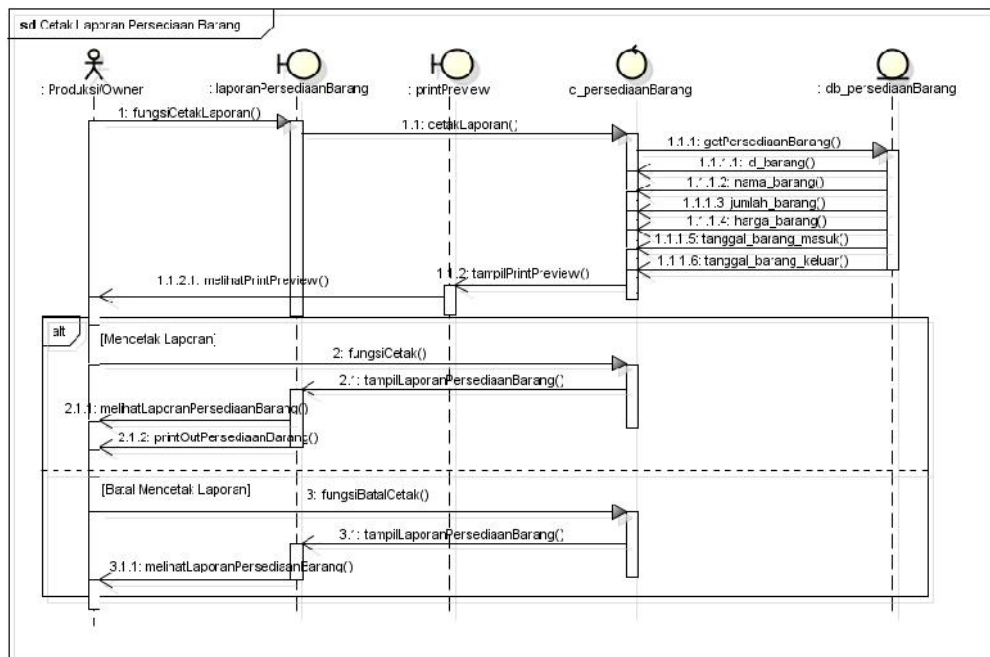
Sequence diagram lihat laporan persediaan barang akan dimulai ketika pengguna berhasil login sebagai Produksi atau Owner dan menggunakan menu lihat laporan persediaan barang. Kemudian akan dikirim ke *controller* *c_persediaanBarang* dan dilakukan pengambilan data persediaan barang dari database melalui *model* *db_persediaanBarang*, yang selanjutnya *controller* akan menampilkan halaman lihat laporan persediaan barang. Berikut ini merupakan *sequence diagram* lihat laporan persediaan barang yang ditunjukkan pada Gambar 5.4.



Gambar 5.3 Sequence Diagram Lihat Laporan Persediaan Barang

5.2.1.3 Sequence Diagram Cetak Laporan Persediaan Barang

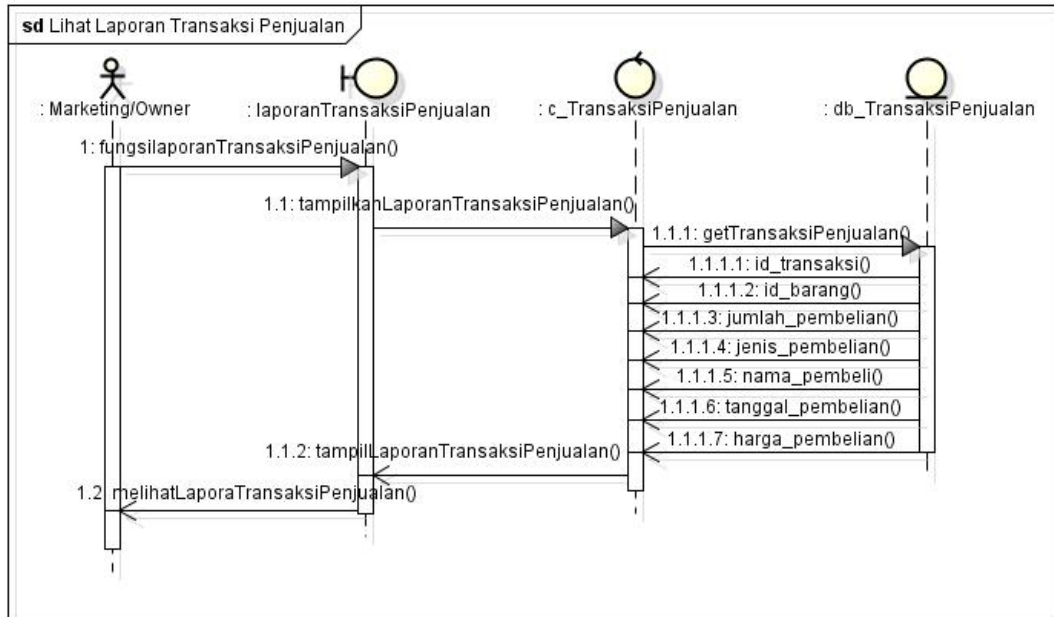
Sequence diagram cetak laporan persediaan barang dimulai ketika pengguna berhasil login sebagai Produksi atau Owner yang menggunakan fungsi cetak laporan pada halaman laporan persediaan barang. Kemudian *controller* *c_persediaanBarang* merespon *action* tersebut dan mengambil data persediaan barang dari database melalui *model* *db_persediaanBarang*. Setelah mendapatkan data persediaan barang *controller* akan menampilkan halaman *printPreview*. Jika pengguna memilih fungsi *cetak* maka pengguna akan mendapatkan *print out* laporan persediaan barang, namun jika pengguna memilih fungsi *batal* maka pengguna membatalkan cetak laporan. Berikut ini merupakan *sequence diagram cetak laporan persediaan barang* yang ditunjukkan pada Gambar 5.5.



Gambar 5.4 Sequence Diagram Cetak Laporan Persediaan Barang

5.2.1.4 Sequence Diagram Lihat Laporan Transaksi Penjualan

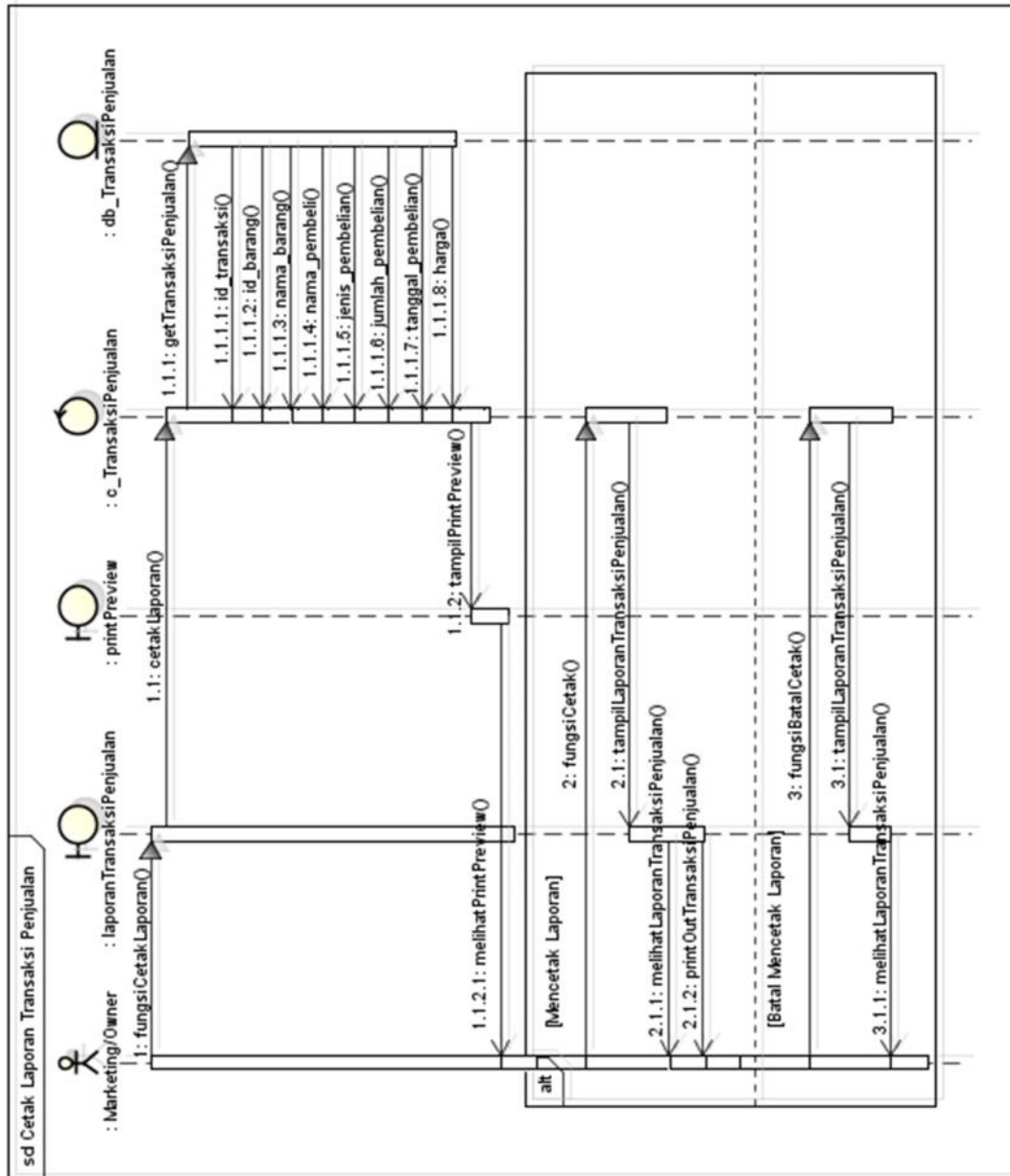
Sequence diagram lihat laporan transaksi penjualan akan dimulai ketika pengguna berhasil login sebagai Marketing atau Owner dan menggunakan menu lihat laporan transaksi penjualan. Kemudian akan dikirim ke *controller* *c_TransaksiPenjualan* dan dilakukan pengambilan data transaksi penjualan dari database melalui *model* *db_TransaksiPenjualan*, yang selanjutnya *controller* akan menampilkan halaman lihat laporan transaksi penjualan. Berikut ini merupakan *sequence diagram lihat laporan transaksi penjualan* yang ditunjukkan pada Gambar 5.6.



Gambar 5.5 Sequence Diagram Lihat Laporan Transaksi Penjualan

5.2.1.5 Sequence Diagram Cetak Laporan Transaksi Penjualan

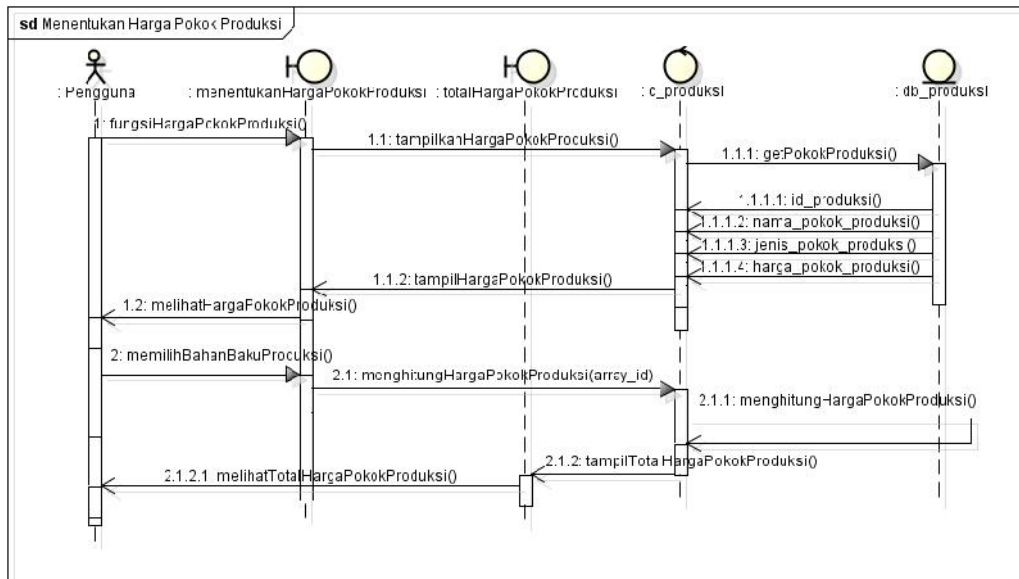
Sequence diagram cetak laporan transaksi penjualan dimulai ketika pengguna berhasil login sebagai Marketing atau Owner yang menggunakan fungsi cetak laporan pada halaman laporan transaksi penjualan. Kemudian *controller* *c_TransaksiPenjualan* merespon *action* tersebut dan mengambil data transaksi penjualan dari database melalui *model* *db_TransaksiPenjualan*. Setelah mendapatkan data transaksi penjualan *controller* akan menampilkan halaman *printPreview*. Jika pengguna memilih fungsi *cetak* maka pengguna akan mendapatkan *print out* laporan transaksi penjualan, namun jika pengguna memilih fungsi *batal* maka pengguna membatalkan cetak laporan yang selanjutnya *controller* menampilkan halaman laporan transaksi penjualan. Berikut ini merupakan *sequence diagram cetak laporan transaksi penjualan* yang ditunjukkan pada Gambar 5.7.



Gambar 5.6 Sequence Diagram Cetak Laporan Transaksi Penjualan

5.2.1.6 Sequence Diagram Menentukan Harga Pokok Produksi

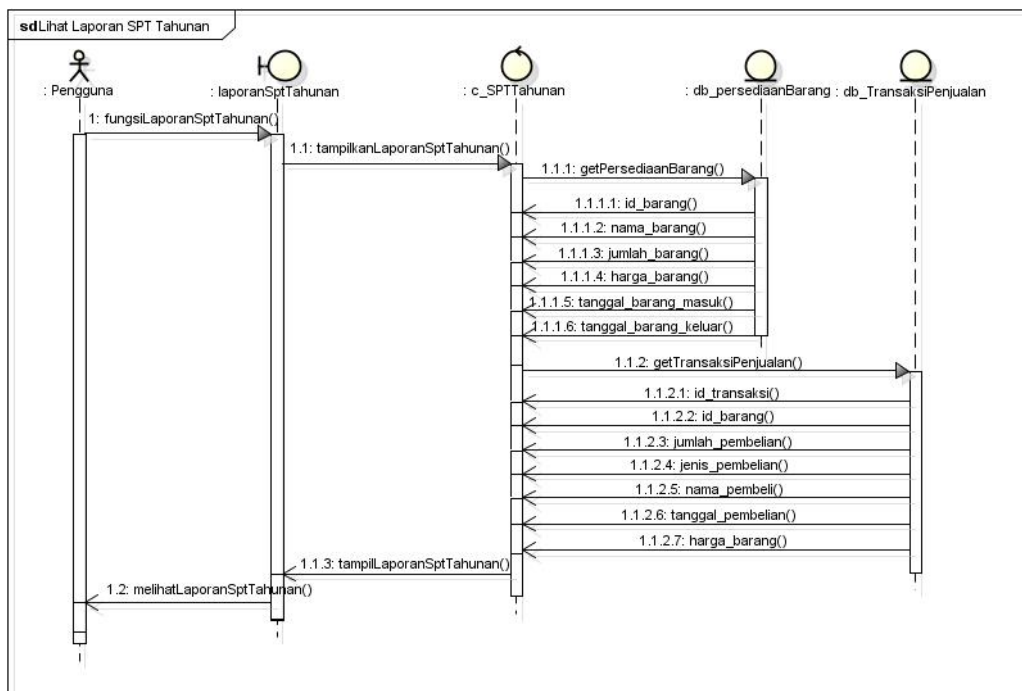
Sequence diagram menentukan harga pokok produksi dimulai ketika pengguna telah berhasil login dan menggunakan fungsi menentukan harga pokok produksi. Kemudian *controller* *c_produksi* mengambil data produksi dari database melalui *model* *db_produksi* dan menampilkan halaman harga pokok produksi. Selanjutnya pengguna memasukkan data produksi yang akan dihitung, kemudian *controller* menghitung harga pokok produksi berdasarkan masukan dari pengguna dan menampilkan halaman total harga pokok produksi. Berikut ini merupakan *sequence diagram* menentukan harga pokok produksi yang ditunjukkan pada Gambar 5.8.



Gambar 5.7 Sequence Diagram Menentukan Harga Pokok Produksi

5.2.1.7 Sequence Diagram Lihat Laporan pendukung SPT Tahunan

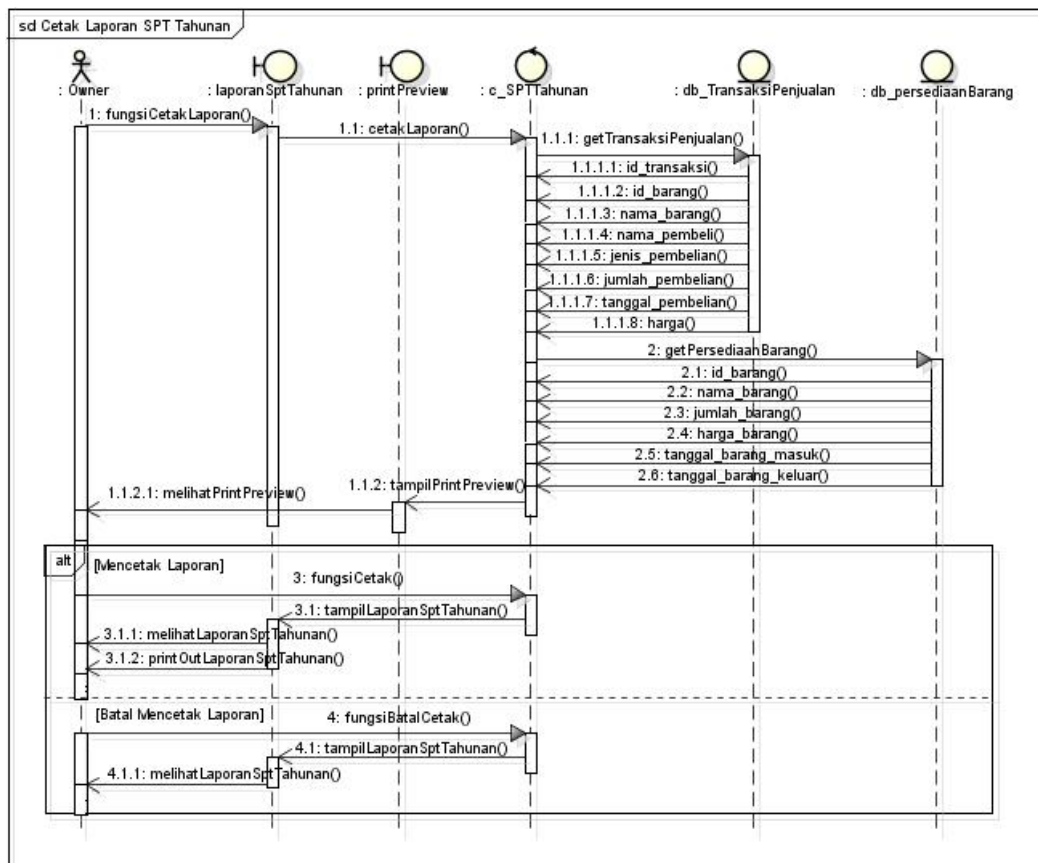
Sequence diagram lihat laporan pendukung SPT Tahunan akan dimulai ketika pengguna berhasil login dan menggunakan menu lihat laporan pendukung SPT Tahunan. Kemudian akan dikirim ke *controller c_SPTTahunan* dan dilakukan pengambilan data persediaan barang dan transaksi penjualan dari database melalui *model db_persediaanBarang* dan *db_TransaksiPenjualan*, yang selanjutnya *controller* akan menampilkan halaman lihat laporan pendukung SPT Tahunan. Berikut ini merupakan *sequence diagram lihat laporan pendukung SPT Tahunan* yang ditunjukkan pada Gambar 5.9.



Gambar 5.8 Sequence Diagram Lihat Laporan pendukung SPT Tahunan

5.2.1.8 Sequence Diagram Cetak Laporan pendukung SPT Tahunan

Sequence diagram cetak laporan pendukung SPT Tahunan dimulai ketika pengguna berhasil login sebagai Owner yang menggunakan fungsi cetak laporan pada halaman laporan pendukung SPT Tahunan. Kemudian *controller c_SPTTahunan* merespon *action* tersebut dan mengambil data persediaan barang dan transaksi penjualan dari database melalui *model db_persediaanBarang* dan *db_TransaksiPenjualan*. Setelah mendapatkan data persediaan barang dan transaksi penjualan *controller* akan menampilkan halaman *printPreview*. Jika pengguna memilih fungsi *cetak* maka pengguna akan mendapatkan *print out* laporan pendukung SPT Tahunan, namun jika pengguna memilih fungsi *batal* maka pengguna membatalkan cetak laporan. Berikut ini merupakan *sequence diagram cetak laporan pendukung SPT Tahunan* yang ditunjukkan pada Gambar 5.11.

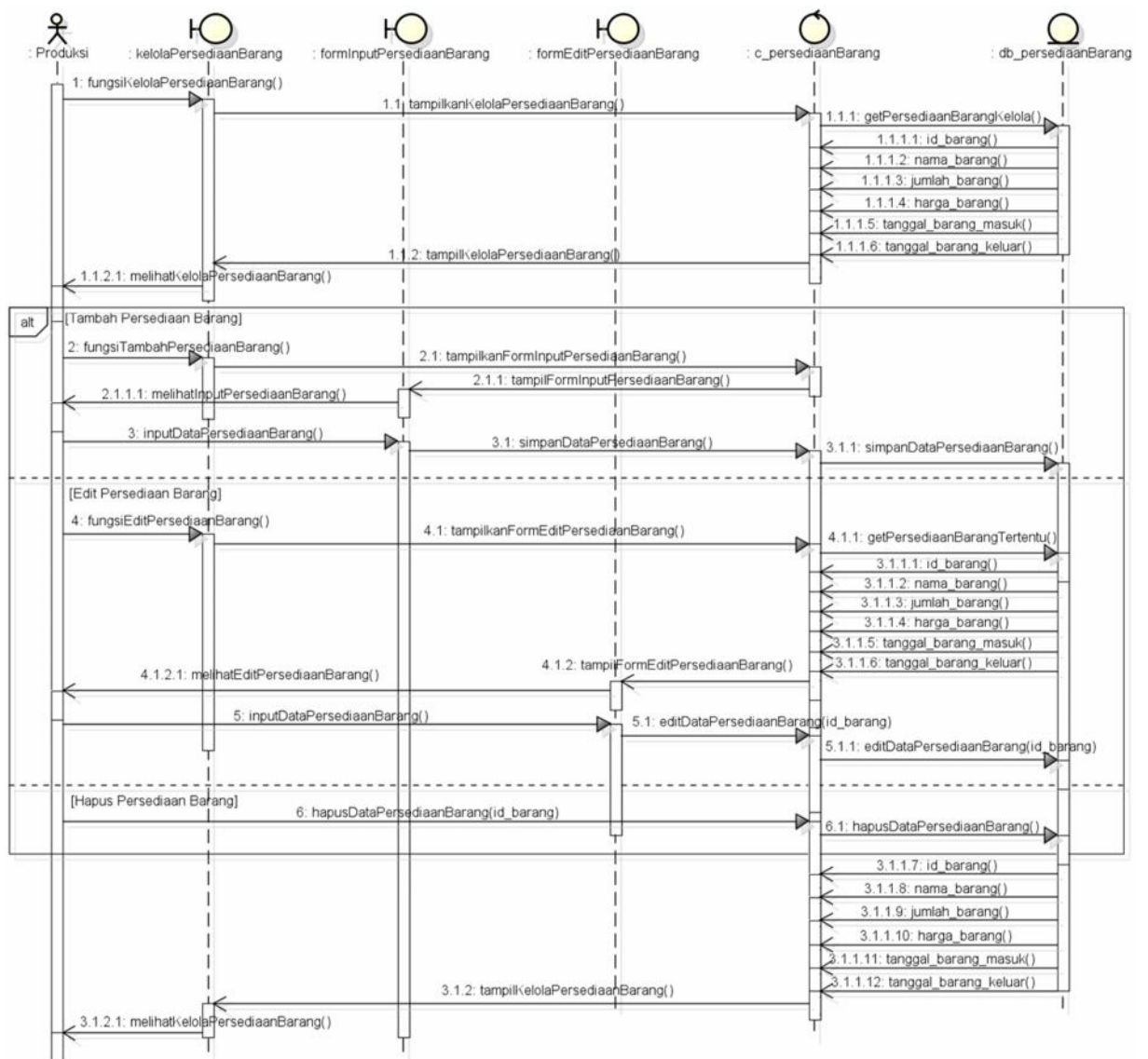


Gambar 5.9 Sequence Diagram Cetak Laporan pendukung SPT Tahunan

5.2.1.9 Sequence Diagram Kelola Persediaan Barang

Sequence diagram kelola persediaan barang dimulai ketika pengguna berhasil login sebagai Produksi dan menggunakan fungsi kelola persediaan barang yang direpson secara langsung oleh *controller c_persediaanBarang*. Kemudian *controller* mengambil data persediaan barang dari database melalui *model db_persediaanBarang* dan menampilkan halaman kelola persediaan barang.

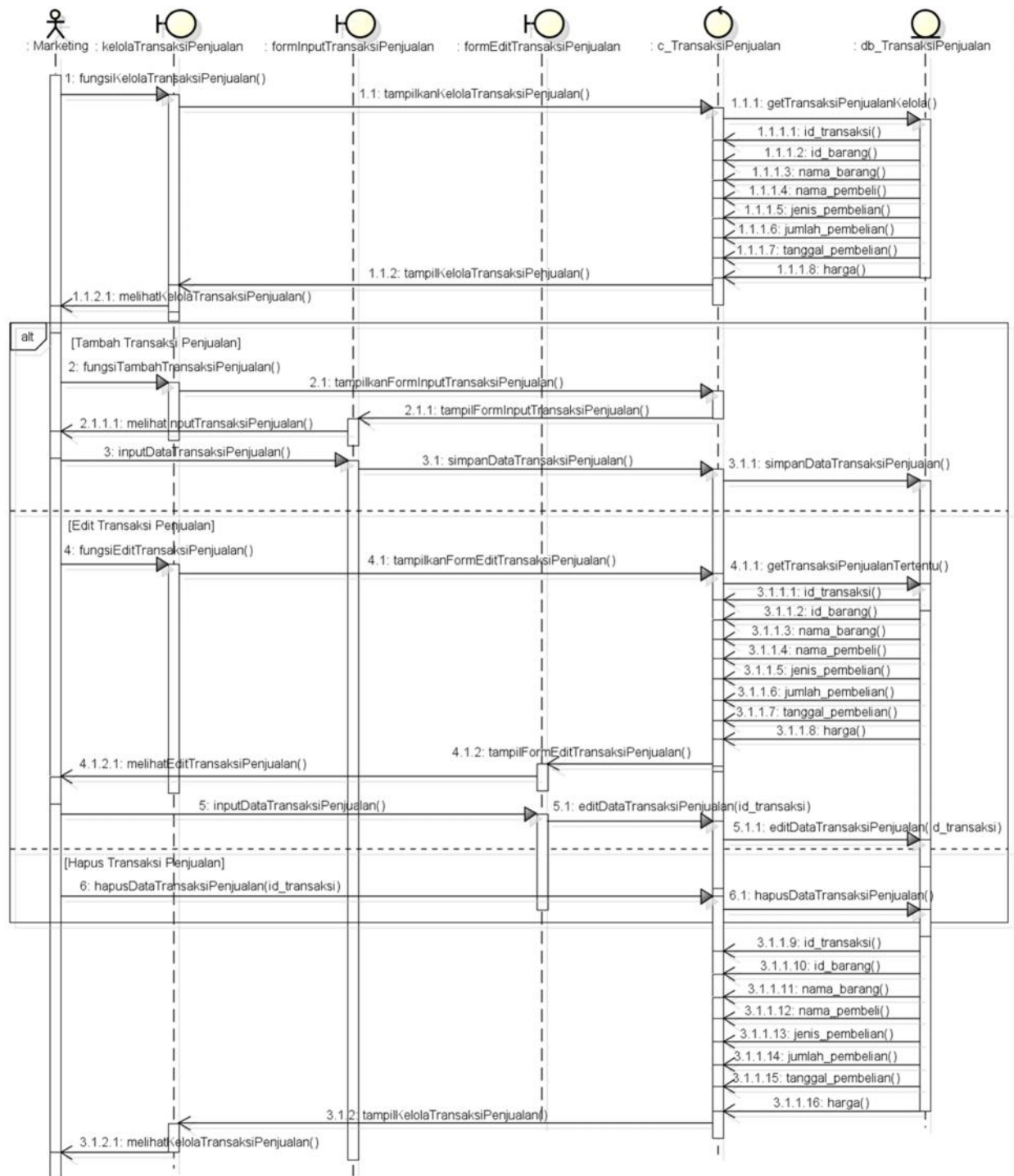
Terdapat 3 fungsi yang dapat dilakukan oleh pengguna pada halaman kelola persediaan barang yaitu menambahkan, mengubah, dan menghapus data persediaan barang. Jika pengguna memilih tambah persediaan barang, *controller* akan menampilkan halaman form input persediaan barang dan setelah data dimasukkan, *controller* akan menambahkan data persediaan barang ke database melalui *model db_persediaanBarang*. Jika pengguna memilih ubah persediaan barang, *controller* akan menampilkan halaman form edit persediaan barang dan setelah data yang baru dimasukkan, *controller* akan mengubah data persediaan barang dari database melalui *model db_persediaanBarang*. Dan jika pengguna memilih hapus persediaan barang, *controller* akan menghapus data persediaan barang dari database melalui *model db_persediaanBarang*. Berikut ini merupakan *sequence diagram kelola persediaan barang* yang ditunjukkan pada Gambar 5.12.



Gambar 5.10 Sequence Diagram Kelola Persediaan Barang

5.2.1.10 Sequence Diagram Kelola Transaksi Penjualan

Sequence diagram kelola transaksi penjualan dimulai ketika pengguna berhasil login sebagai Marketing dan menggunakan fungsi kelola transaksi penjualan yang direspon secara langsung oleh *controller c_TransaksiPenjualan*. Kemudian *controller* mengambil data transaksi penjualan dari database melalui *model db_TransaksiPenjualan* dan menampilkan halaman kelola transaksi penjualan. Terdapat 3 fungsi yang dapat dilakukan oleh pengguna pada halaman kelola transaksi penjualan yaitu menambahkan, mengubah, dan menghapus data transaksi penjualan. Jika pengguna memilih tambah transaksi penjualan, *controller* akan menampilkan halaman form input transaksi penjualan dan setelah data dimasukkan, *controller* akan menambahkan data transaksi penjualan ke database melalui *model db_TransaksiPenjualan*. Jika pengguna memilih ubah transaksi penjualan, *controller* akan menampilkan halaman form edit transaksi penjualan dan setelah data yang baru dimasukkan, *controller* akan mengubah data transaksi penjualan dari database melalui *model db_TransaksiPenjualan*. Dan jika pengguna memilih hapus transaksi penjualan, *controller* akan menghapus data transaksi penjualan dari database melalui *model db_TransaksiPenjualan*. Berikut ini merupakan *sequence diagram kelola transaksi penjualan* yang ditunjukkan pada Gambar 5.13.

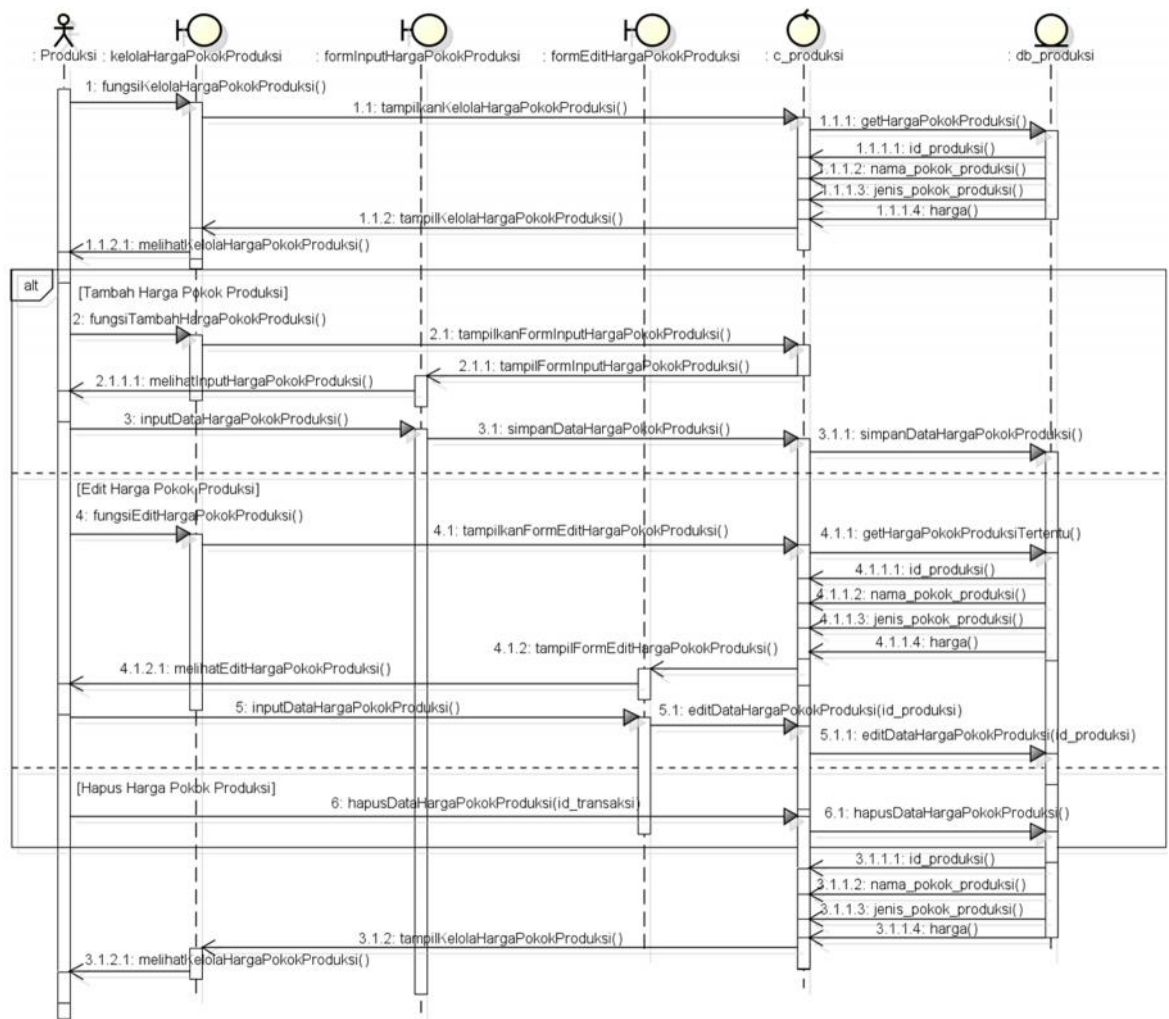


Gambar 5.11 Sequence Diagram Kelola Transaksi Penjualan

5.2.1.11 Sequence Diagram Kelola Harga Pokok Produksi

Sequence diagram kelola harga pokok produksi dimulai ketika pengguna berhasil login sebagai Produksi dan menggunakan fungsi kelola harga pokok produksi yang direspon secara langsung oleh *controller c_produksi*. Kemudian *controller* mengambil data produksi dari database melalui *model db_produksi* dan menampilkan halaman kelola harga pokok produksi. Terdapat 3 fungsi yang

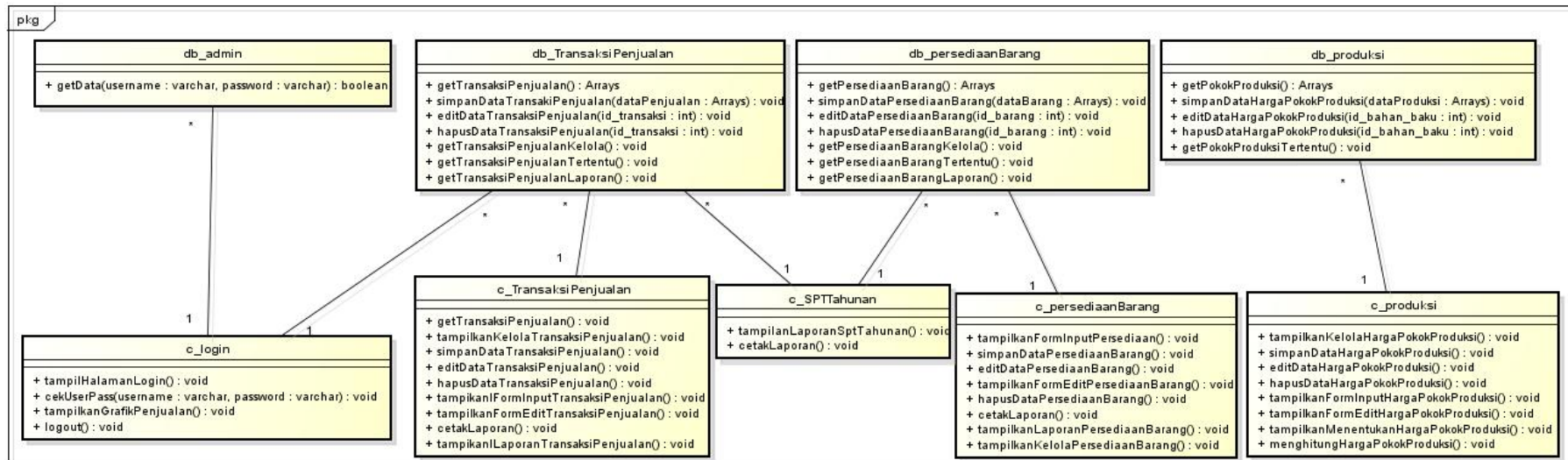
dapat dilakukan oleh pengguna pada halaman kelola harga pokok produksi yaitu menambahkan, mengubah, dan menghapus data produksi. Jika pengguna memilih tambah harga pokok produksi, *controller* akan menampilkan halaman form input harga pokok produksi dan setelah data dimasukkan, *controller* akan menambahkan data harga pokok produksi ke database melalui *model db_produksi*. Jika pengguna memilih ubah harga pokok produksi, *controller* akan menampilkan halaman form edit harga pokok produksi dan setelah data yang baru dimasukkan, *controller* akan mengubah data harga pokok produksi dari database melalui *model db_produksi*. Dan jika pengguna memilih hapus harga pokok produksi, *controller* akan menghapus data harga pokok produksi dari database melalui *model db_produksi*. Berikut ini merupakan *sequence diagram kelola harga pokok produksi* yang ditunjukkan pada Gambar 5.14.



Gambar 5.12 Sequence Diagram Kelola Harga Pokok Produksi

5.2.2 Pemodelan Class (Class Diagram)

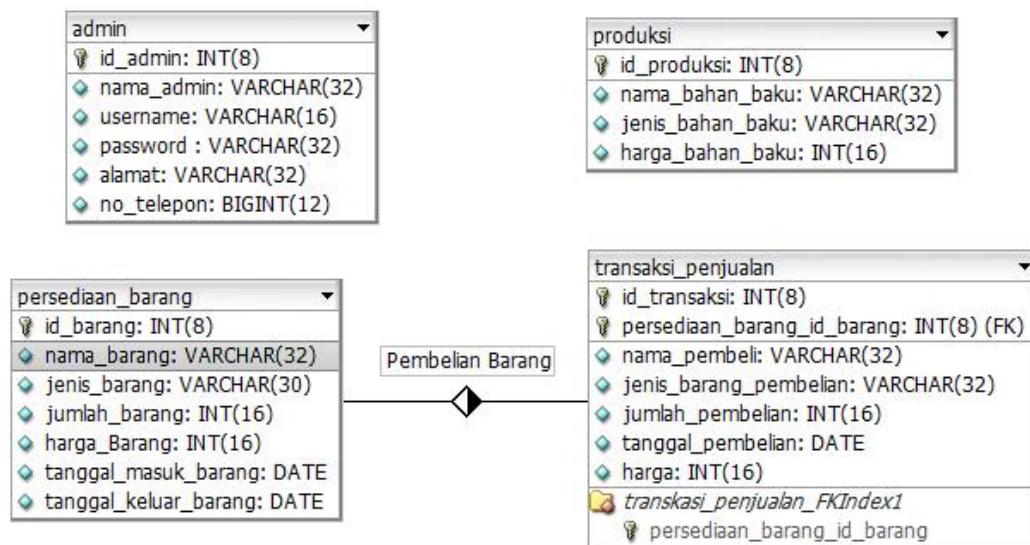
Setelah diketahui hubungan mendetail mengenai perilaku setiap *use case* yang telah dijelaskan pada sub bab 5.2.1 yaitu *sequence diagram* langkah berikutnya adalah mengetahui seluruh hubungan antar *class* yang disertai dengan *atribut* dan *operan(method/fungsi)* agar dalam pembuatan perangkat lunak sudah mengetahui class, atribut, dan fungsi apa yang akan dibuat dan disertai dengan hubungan antar class. Hal tersebut akan dijelaskan dengan membuat *class diagram* yang akan ditunjukkan pada Gambar 5.15 sebagai berikut.



Gambar 5.13 Class Diagram Sistem Informasi Son Screen Printing

5.3 Perancangan Database

Database (basis data) merupakan tempat penyimpanan data. Basis data yang dipakai pada sistem informasi ini menggunakan basis data Mysql. Perancangan basis data digunakan untuk merancang basis data yang akan dibuat agar masukan dan keluaran program sesuai dengan apa yang diharapkan. Perancangan basis data mengambil acuan dari proses analisis data yang dilakukan pada tahap analisis kebutuhan. Gambar rancangan basis data ditunjukkan pada Gambar 5.16.



Gambar 5.14 Physical Data Model Sistem Informasi Son Screen Printing

Berikut ini merupakan struktur tabel dan keterangan masing-masing tabel yang ada pada basis data. Tabel *db_persediaanBarang* merupakan penyimpanan data persediaan barang yang *primary key* dari tabel ini adalah *id_barang*. Struktur tabel *persediaan_barang* akan ditunjukkan pada Tabel 5.1

Tabel 5.1 Struktur Tabel *persediaan_barang*

No.	Nama Fields	Tipe	Lebar	Keterangan
1.	id_barang	integer(PK)	8	Id barang sebagai <i>primary key</i>
2.	nama_barang	varchar	32	Nama barang
3.	jenis_barang	varchar	30	Jenis barang apakah produksi, setengah jadi, kain, benang, pewarna, peralatan jahit, peralatan sablon, atau kancing

Tabel 5.1 Struktur Tabel persediaan_barang (lanjutan)

No.	Nama Fields	Tipe	Lebar	Keterangan
3.	jumlah_barang	integer	16	Jumlah barang
4.	harga_barang	integer	16	Harga barang
5.	tanggal_barang_masuk	date		Tanggal barang diterima perusahaan
6.	tanggal_barang_keluar	date		Tanggal barang keluar dari perusahaan

Tabel *produksi* merupakan penyimpanan data bahan-bahan untuk produksi yang *primary key* dari tabel ini adalah *id_bahan_baku*. Struktur tabel *produksi* akan ditunjukkan pada Tabel 5.2.

Tabel 5.2 Struktur Tabel *produksi*

No.	Nama Fields	Tipe	Lebar	Keterangan
1.	id_bahan_baku	integer(PK)	8	Id bahan baku produksi sebagai <i>primary key</i>
2.	nama_bahan_baku	varchar	32	Nama bahan baku
3.	jenis_bahan_baku	varchar	32	Jenis bahan baku
4.	harga_bahan_baku	integer	16	Harga bahan baku

Tabel *transaksi_penjualan* merupakan penyimpanan data transaksi yang dilakukan dengan pembeli yang *primary key* dari tabel ini adalah *id_transaksi*. Struktur tabel *db_TransaksiPenjualan* akan ditunjukkan pada Tabel 5.3 yang terdapat relasi antar tabel yaitu relasi dengan tabel *db_persediaanBarang* dengan *foreign key id_barang* yang nama relasinya adalah membuat laporan. Jadi setiap pembelian barang akan mempengaruhi persediaan barang.

Tabel 5.3 Struktur Tabel *transaksi_penjualan*

No.	Nama Fields	Tipe	Lebar	Keterangan
1.	id_transaksi	integer(PK)	8	Id transaksi penjualan sebagai <i>primary key</i>
2.	persediaan_barang_id_barang	integer(FK)	8	Sebagai relasi antara tabel transaksi penjualan dengan persediaan barang

Tabel 5.3 Struktur Tabel transaksi_penjualan (lanjutan)

No.	Nama Fields	Tipe	Lebar	Keterangan
3.	nama_pembeli	varchar	32	Nama pembeli
4.	jenis_barang_pembelian	varchar	32	Jenis barang seperti kaos, sweater, jaket, dan lain-lain
5.	jumlah_pembelian	Integer	16	Jumlah pembelian
6.	tanggal_pembelian	date		Tanggal pembelian
7.	harga	integer	16	Harga pembelian

Tabel *admin* merupakan penyimpanan data aktor yang memiliki hak akses untuk login. Struktur tabel *admin* akan ditunjukkan pada Tabel 5.4.

Tabel 5.4 Struktur Tabel *admin*

No.	Nama Fields	Tipe	Lebar	Keterangan
1.	id_admin	integer(PK)	8	Id bahan baku produksi sebagai <i>primary key</i>
2.	nama_admin	varchar	32	Nama admin
3.	username	varchar	16	Username admin
4.	password	varchar	32	Password admin
5.	alamat	varchar	32	Alamat admin
6.	no_telepon	big integer	12	Nomer telepon admin

5.4 Perancangan Komponen

Perancangan komponen mendefinisikan struktur data atau algoritma yang digunakan dalam suatu komponen perangkat lunak. Di dalam perancangan komponen ini hanya mencantumkan algoritma menghitung harga pokok produksi yang diletakkan pada *controller c_produksi*, simpan data transaksi penjualan yang terdapat pada *controller c_TransaksiPenjualan*, dan tampilkan laporan pendukung SPT Tahunan yang diletakkan pada *controller c_SPTTahunan*.

5.4.1 Controller Produksi

Disertakan pada Tabel 5.5 pseudocode dari *controller c_produksi* dengan *function* menghitungHargaPokokProduksi() sebagai berikut.

Tabel 5.5 Pseudocode Menghitung Harga Pokok Produksi

1.	biaya_bahan_baku = inputan_biaya_bahan_baku
2.	total_biaya_bahan_baku = 0
3.	for (index from 1 to biaya_bahan_baku)
4.	if (inputan_jumlah_biaya_bahan_baku != NULL inputan_jumlah_biaya_bahan_baku != 0)
5.	nama = inputan_nama_bahan_baku
6.	jumlah = inputan_jumlah_bahan_baku
7.	harga = inputan_harga_bahan_baku
8.	dataProduksi['nama_bahan_baku'][index] = nama
9.	dataProduksi['jumlah_bahan_baku'][index] = jumlah
10.	dataProduksi['harga_bahan_baku'][index] = harga
11.	total_biaya_bahan_baku += (jumlah * harga)
12.	else
13.	biaya_bahan_baku--
14.	end if
15.	end for
16.	dataProduksi['biaya_bahan_baku'] = total_biaya_bahan_baku
17.	dataProduksi['total_biaya_bahan_baku'] = total_biaya_bahan_baku
18.	biaya_tenaga_kerja_langsung = inputan_biaya_tenaga_kerja_langsung
19.	total_biaya_tenaga_kerja_langsung = 0
20.	for (index from 1 to biaya_tenaga_kerja_langsung)
21.	if (inputan_jumlah_biaya_tenaga_kerja_langsung != NULL inputan_jumlah_biaya_tenaga_kerja_langsung != 0)
22.	nama = inputan_nama_tenaga_kerja_langsung
23.	jumlah = inputan_jumlah_tenaga_kerja_langsung
24.	harga = inputan_harga_tenaga_kerja_langsung
25.	dataProduksi['nama_tenaga_kerja_langsung'][index] = nama
26.	dataProduksi['jumlah_tenaga_kerja_langsung'][index] = jumlah
27.	dataProduksi['harga_tenaga_kerja_langsung'][index] = harga
28.	total_biaya_tenaga_kerja_langsung += (jumlah * harga)
29.	else

30.	biaya_tenaga_kerja_langsung --
31.	end if
32.	end for
33.	dataProduksi['biaya_tenaga_kerja_langsung'] = biaya_tenaga_kerja_langsung
34.	dataProduksi['total_biaya_tenaga_kerja_langsung'] = total_biaya_tenaga_kerja_langsung
35.	biaya_overhead_variabel = inputan_biaya_overhead_variabel
36.	total_biaya_overhead_variabel = 0
37.	for (index from 1 to biaya_overhead_variabel)
38.	if (inputan_jumlah_biaya_overhead_variabel != NULL inputan_jumlah_biaya_overhead_variabel != 0)
39.	nama = inputan_nama_overhead_variabel
40.	jumlah = inputan_jumlah_overhead_variabel
41.	harga = inputan_harga_overhead_variabel
42.	dataProduksi['nama_overhead_variabel'][index] = nama
43.	dataProduksi['jumlah_overhead_variabel'][index] = jumlah
44.	dataProduksi['harga_overhead_variabel'][index] = harga
45.	total_biaya_overhead_variabel += (jumlah * harga)
46.	else
47.	biaya_overhead_variabel --
48.	end if
49.	end for
50.	dataProduksi['biaya_overhead_variabel'] = biaya_overhead_variabel
51.	dataProduksi['total_biaya_overhead_variabel'] = total_biaya_overhead_variabel
52.	dataProduksi['total_produksi'] = total_biaya_bahan_baku + total_biaya_tenaga_kerja_langsung + total_overhead_variabel
53.	view('totalHargaPokokProduksi', dataProduksi)

5.4.2 Controller Transaksi Penjualan

Disertakan pada Tabel 5.6 pseudocode dari *controller* c_TransaksiPenjualan dengan *function* simpanDataTransaksiPenjualan() sebagai berikut.

Tabel 5.6 Pseudocode Simpan Data Transaksi Penjualan

1.	dataStatus = userdata('logged_user')
2.	if (data = "Marketing")
3.	dataTransaksi = array (nama_pembeli = inputan_nama_pembeli, persediaan_barang_id_barang = inputan_pembelian, jenis_barang_pembelian = inputan_jenis_barang_pembelian, jumlah_pembelian = inputan_jumlah_pembelian, tanggal_pembelian = inputan_tanggal_pembelian, harga = inputan_harga)
4.	status_simpan_data = simpanDataTransaksiPenjualan(dataTransaksi)
5.	if (status_simpan_data == TRUE)
6.	dataTransaksi['notif'] = 'Data Berhasil Disimpan'
7.	dataTransaksi['transaksi'] = getTransaksiPenjualanKelola()
8.	view('kelolaTransaksiPenjualan', dataTransaksi)
9.	else
10.	dataTransaksi['notif'] = 'Data Gagal Disimpan'
11.	dataTransaksi['transaksi'] = getTransaksiPenjualanKelola()
12.	view('kelolaTransaksiPenjualan', dataTransaksi)
13.	end if
14.	else
15.	session = 'Waktu Session Habis'
16.	data['notif'] = session
17.	view('Login', data)
18.	end if

5.4.3 Controller SPT Tahunan

Disertakan pada Tabel 5.7 pseudocode dari *controller* c_SPTTahunan dengan *function* tampilkanLaporanSptTahunan() sebagai berikut.

Tabel 5.7 Pseudocode Tampilkan Laporan pendukung SPT Tahunan

1.	dataStatus = userdata('logged_user')
2.	if (data = "TRUE")
3.	dataTransaksi['transaksi'] = getTransaksiPenjualanLaporan()
4.	dataTransaksi['barang'] = getPersediaanBarang()
5.	dataTransaksi['omset'] = 0
6.	dataTransaksi['produksi'] = 0
7.	foreach (dataTransaksi['transaksi'] as transaksi)
8.	if (transaksi->tahun == date('Y'))
9.	dataTransaksi['omset'] += transaksi->harga
10.	dataTransaksi['produksi'] += transaksi->harga_barang
11.	else
12.	continue
13.	end if
14.	end for
15.	dataTransaksi['labaRugi'] = dataTransaksi['omset'] – dataTransaksi['produksi']
16.	dataTransaksi['neto'] = dataTransaksi['labaRugi']
17.	view('laporanSptTahunan', dataTransaksi)
18.	else
19.	session = 'Waktu Session Habis'
20.	data['notif'] = session
21.	view('Login', data)
22.	end if

5.5 Perancangan Antarmuka

5.5.1 Antarmuka Login

Halaman *Login* merupakan halaman pertama yang akan dilihat oleh pengguna sistem karena untuk menggunakan fungsi lain selain *Login* harus melakukan tahap keamanan yaitu *Login* yang dapat dilakukan pada halaman *Login*. Pada halaman ini terdapat 4 fungsi utama yaitu (1) notifikasi yang berguna untuk memberikan pemberitahuan kepada pengguna sistem seperti username atau password salah ataupun ucapan terima kasih; (2) memasukkan username yang berfungsi untuk masukan identitas username yang telah terdaftar; (3)

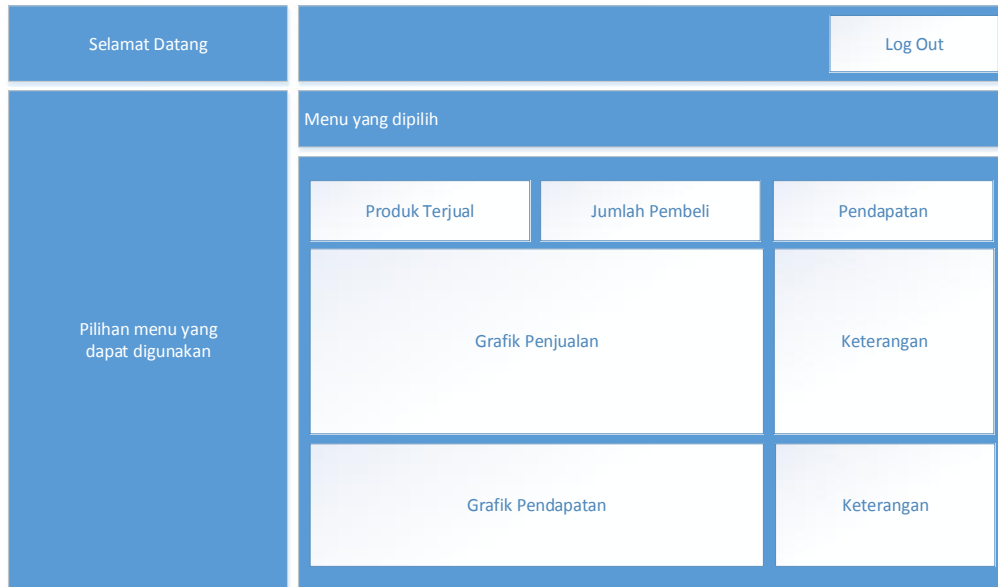
memasukkan password yang berguna untuk masukan password; dan (4) tombol button *Login* yang berfungsi untuk mengecek username dan password. Antarmuka dari halaman *Login* dapat dilihat pada Gambar 5.17.

The image shows a login form with a light blue header containing the text "Nama Perusahaan". Below the header, there is a main form area with a light blue background. On the left side of this area, there is a blue square labeled "Logo". To the right of the logo, there are three input fields stacked vertically: "Notifikasi", "Username", and "Password". Each field has a light blue label on its left side. At the bottom right of the form area, there is a blue button labeled "Login".

Gambar 5.15 Antarmuka Login

5.5.2 Antarmuka Grafik Penjualan

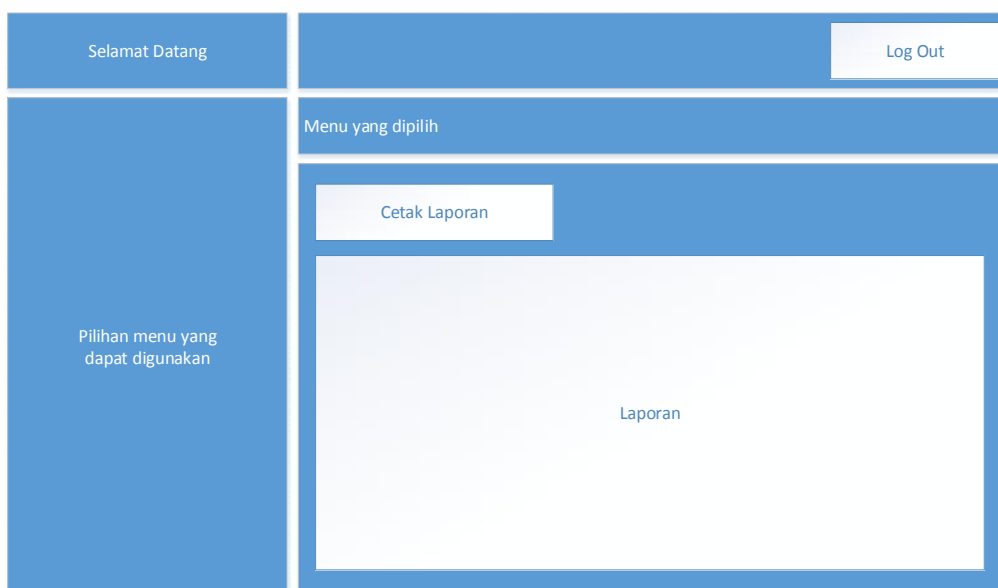
Halaman *Grafik Penjualan* merupakan halaman yang menampilkan perkembangan penjualan pada Son Screen Printing Sidoarjo. Pada halaman ini terdapat 7 fungsi utama yaitu (1) *Logout* merupakan salah satu fungsi yang berguna untuk keamanan sistem dari orang-orang yang tidak bertanggung jawab yaitu dengan cara keluar dan menghapus session dari user; (2) produk terjual berisi tentang total produk yang terjual selama satu bulan; (3) jumlah pembeli berisi mengenai total pembeli yang terjual selama satu bulan; (4) pendapatan berisi tentang total pendapatan yang terjual selama satu bulan; (5) grafik penjualan yang berisi mengenai grafik penjualan selama satu bulan dan satu tahun; (6) grafik pendapatan berisi mengenai pendapatan perusahaan selama satu tahun; dan (7) keterangan berisi mengenai penjelasan secara rinci mengenai grafik yang ditampilkan. Antarmuka dari halaman *Grafik Penjualan* dapat dilihat pada Gambar 5.18.



Gambar 5.16 Antarmuka Grafik Penjualan

5.5.3 Antarmuka Laporan

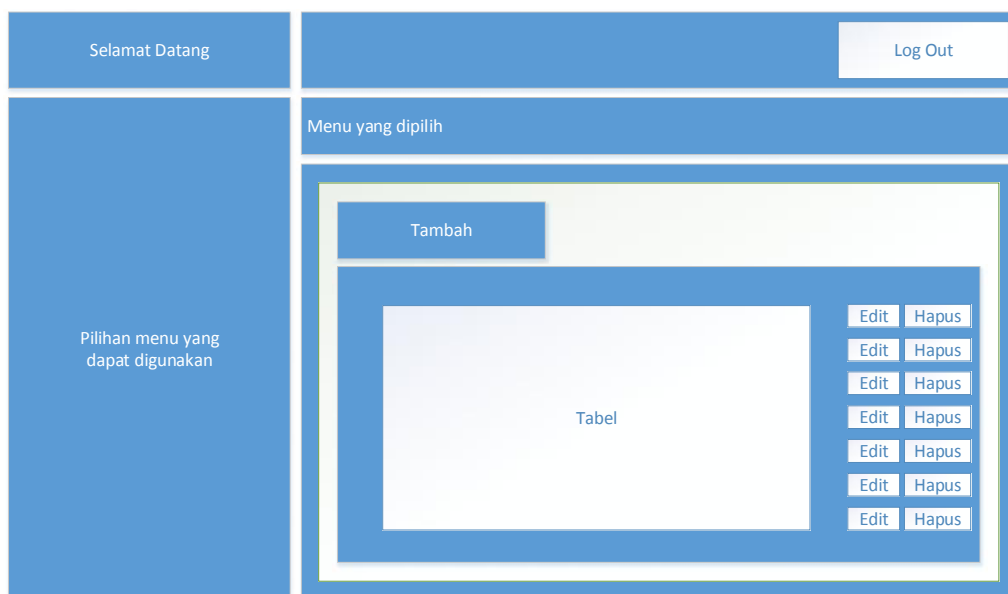
Pada antarmuka *Laporan* merupakan halaman pelaporan dari persediaan barang, transaksi penjualan, dan SPT tahunan. Halaman ini memiliki 3 fungsi utama yaitu (1) *Logout* merupakan salah satu fungsi yang berguna untuk keamanan sistem dari orang-orang yang tidak bertanggung jawab yaitu dengan cara keluar dan menghapus session dari user; (2) cetak laporan ditujukan untuk dokumen perusahaan berupa *print out* laporan perusahaan sehingga data yang terkumpul dapat diolah oleh perusahaan; dan (3) isi laporan merupakan penyampaian berita, keterangan, pemberitahuan ataupun pertanggungjawaban dari perusahaan. Antarmuka dari halaman *Laporan* dapat dilihat pada Gambar 5.19.



Gambar 5.17 Antarmuka Laporan

5.5.4 Antarmuka Kelola Data

Pada antarmuka *Kelola Data* merupakan halaman kelola data dari persediaan barang, transaksi penjualan, dan harga pokok produksi. Halaman ini memiliki 5 fungsi utama yaitu (1) *Logout* merupakan salah satu fungsi yang berguna untuk keamanan sistem dari orang-orang yang tidak bertanggung jawab yaitu dengan cara keluar dan menghapus session dari user; (2) tambah yaitu menambahkan data baru yang disimpan kedalam basis data agar datanya dapat diolah oleh sistem; (3) tabel berisi tentang data perusahaan yang diambil dari basis data; (4) edit merupakan fungsi yang digunakan untuk mengubah data yang ditampilkan pada tabel kelola data; dan (5) hapus merupakan fungsi yang digunakan untuk menghapus data yang ditampilkan pada tabel kelola data. Antarmuka dari halaman *Kelola Data* dapat dilihat pada Gambar 5.20.



Gambar 5.18 Antarmuka Kelola Data

5.5.5 Antarmuka Form Input atau Edit

Pada antarmuka *Form Input atau Edit* merupakan halaman yang berisi form input atau edit dari persediaan barang, transaksi penjualan, dan harga pokok produksi. Halaman ini memiliki 4 fungsi utama yaitu (1) *Logout* merupakan salah satu fungsi yang berguna untuk keamanan sistem dari orang-orang yang tidak bertanggung jawab yaitu dengan cara keluar dan menghapus session dari user; (2) nama fitur yang sedang dilakukan contoh edit data persediaan barang, tambah data persediaan barang, atau yang lainnya; (3) form inputan yang berisi tentang kumpulan inputan untuk diisi oleh pengguna; dan (4) submit yaitu fungsi yang digunakan untuk menyimpan atau mengubah data yang telah dimasukkan kedalam form. Antarmuka dari halaman *Form Input dan Edit* dapat dilihat pada Gambar 5.21.

Gambar 5.19 Antarmuka Form Input dan Edit

5.5.6 Antarmuka Menentukan Harga Pokok Produksi

Halaman *Menentukan Harga Pokok Produksi* merupakan halaman yang menampilkan bahan baku produksi, biaya tenaga kerja langsung, dan biaya overhead variabel yang disertai dengan harga dan jumlah untuk dihitung total harga pokok produksi. Pada halaman ini terdapat 4 fungsi utama yaitu (1) *Logout* merupakan salah satu fungsi yang berguna untuk keamanan sistem dari orang-orang yang tidak bertanggung jawab yaitu dengan cara keluar dan menghapus session dari user; (2) bahan baku produksi berisi mengenai kumpulan bahan baku produksi yang dapat dipilih oleh pengguna untuk menentukan harga pokok produksi; (3) biaya tenaga kerja langsung berisi tentang kumpulan tenaga kerja yang secara langsung berpengaruh pada harga pokok produksi dan dapat dipilih oleh pengguna; dan (4) biaya overhead variabel yang berisi mengenai kumpulan kelebihan atau over biaya dari perusahaan seperti biaya penyusutan, biaya pemeliharaan, atau yang lain sebagainya yang dapat dipilih oleh pengguna untuk menentukan harga pokok produksi. Antarmuka dari halaman *Menentukan Harga Pokok Produksi* dapat dilihat pada Gambar 5.22.



Gambar 5.20 Antarmuka Menentukan Harga Pokok Produksi

5.5.7 Antarmuka Total Harga Pokok Produksi

Halaman *Total Harga Pokok Produksi* merupakan halaman yang menampilkan hasil perhitungan dari harga pokok produksi yang telah didefinisikan pada halaman menentukan harga pokok produksi. Pada halaman ini terdapat 3 fungsi utama yaitu (1) *Logout* merupakan salah satu fungsi yang berguna untuk keamanan sistem dari orang-orang yang tidak bertanggung jawab yaitu dengan cara keluar dan menghapus session dari user; (2) hasil perhitungan berisi tentang rincian pilihan biaya yang berpengaruh dalam produksi dan total harga pokok produksi; dan (3) kembali yaitu fungsi yang berguna untuk kembali ke halaman menentukan harga pokok produksi. Antarmuka dari halaman *Total Harga Pokok Produksi* dapat dilihat pada Gambar 5.23.



Gambar 5.21 Antarmuka Total Harga Pokok Produksi