

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini akan membahas tentang identifikasi masalah, analisis permasalahan, solusi permasalahan dan perancangan sistem dalam Rancang Bangun Sistem Informasi Penjualan Roti Menggunakan *Web-Services*. Sebelum melakukan identifikasi dan analisis permasalahan, telah dilakukan pengumpulan data dengan teknik wawancara dan observasi yang dilakukan di perusahaan. Adapun hasil dari wawancara dan observasi dapat dilihat pada lampiran 1.

3.1 Identifikasi Permasalahan

Identifikasi permasalahan dilakukan pada saat maupun setelah proses wawancara dan observasi pada Yulia Bakery. Untuk melakukan identifikasi masalah, maka dilakukan observasi oleh panelis pada Yulia Bakery yang dilakukan pada 29 oktober 2013 sampai 4 november 2013. Data-data yang diperlukan untuk dilakukan analisis dikumpulkan dengan melakukan wawancara pada tiga responden yaitu : pemilik Yulia Bakery, bagian produksi, dan bagian kasir untuk mengumpulkan data. Selain itu juga melakukan pengamatan langsung terhadap kegiatan yang sedang berjalan.

Kegiatan penjualan yang berjalan pada cabang Yulia Bakery sama seperti kegiatan penjualan pada umumnya. Pelanggan yang memilih roti yang akan dibeli, kemudian melakukan pembayaran pada kasir. Kekurangan yang dimiliki oleh Yulia Bakery ini terlihat pada pelayanan yang dilakukan di kasir terjadi cukup lama. Konsumen harus menunggu kasir untuk membuat nota dan mencatat transaksi penjualan ke dalam buku penjualan, data- data yang dicatat oleh kasir ke

dalam nota dan buku penjualan adalah nama roti, jumlah roti yang dibeli, dan total pembelian.

Selain melayani konsumen yang melakukan pembelian secara tunai, kasir juga melayani pesanan roti. Kasir akan mencatat pemesanan ke dalam buku pesanan, data-data yang akan dicatat dalam buku pesanan adalah nama konsumen, alamat, daerah, jenis roti yang dipesan, jumlah pesanan, dan tanggal pemesanan. Dalam pembayaran pemesanan yang diterima, konsumen bisa langsung membayar secara tunai atau dengan memberikan uang muka untuk tanda jadi. Pesanan yang telah diterima oleh kasir nantinya akan dicek setiap hari oleh bagian produksi untuk mengetahui apakah terdapat pesanan yang harus diproduksi.

Dari data-data rekapitulasi penjualan dan pemesanan yang diterima oleh kasir setiap minggunya akan diberikan kepada pemilik yang berada di pusat Yulia Bakery, yang nantinya data-data ini bisa digunakan pemilik untuk memberikan kebijakan. Data-data yang diterima oleh pemilik hanya sebatas transaksi yang berjalan pada Yulia Bakery setiap harinya, yang nantinya pemilik akan mengecek ulang dari penjualan tersebut.

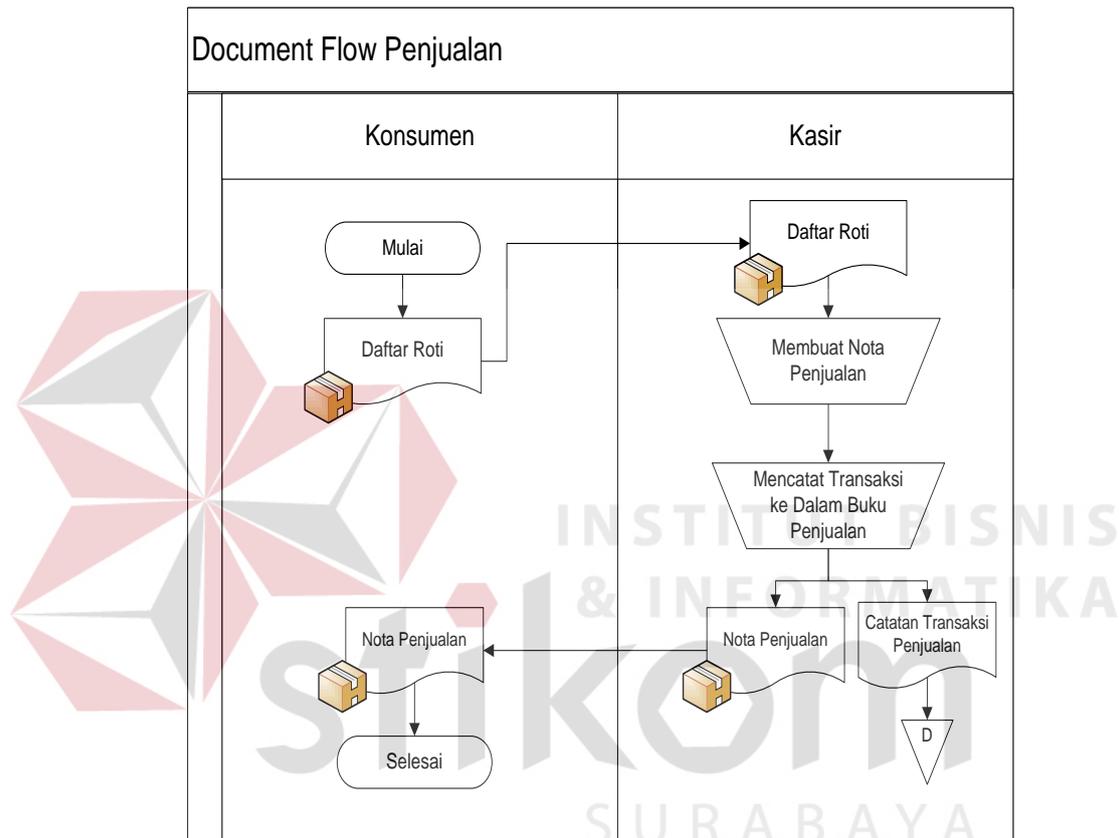
Dari hasil identifikasi yang telah dilakukan, Gambaran proses bisnis yang ada pada Yulia Bakery adalah sebagai berikut :

1. *Document Flow* Transaksi Penjualan Roti.
2. *Document Flow* Pemesanan Roti.
3. *Document Flow* Laporan penjualan dan pemesanan.

3.1.1 Document Flow Transaksi Penjualan Roti

Pada fungsionalitas ini, proses dimulai dari konsumen yang memilih roti yang ada di rak *display* Yulia Bakery kemudian membawa ke kasir dan

melakukan pembayaran. Kasir mencatat transaksi tersebut ke dalam buku penjualan dan membuat nota penjualan yang nantinya akan diberikan kepada konsumen. Data-data yang dimasukkan kedalam buku penjualan adalah nama roti, jumlah roti yang dibeli, dan total pembelian.

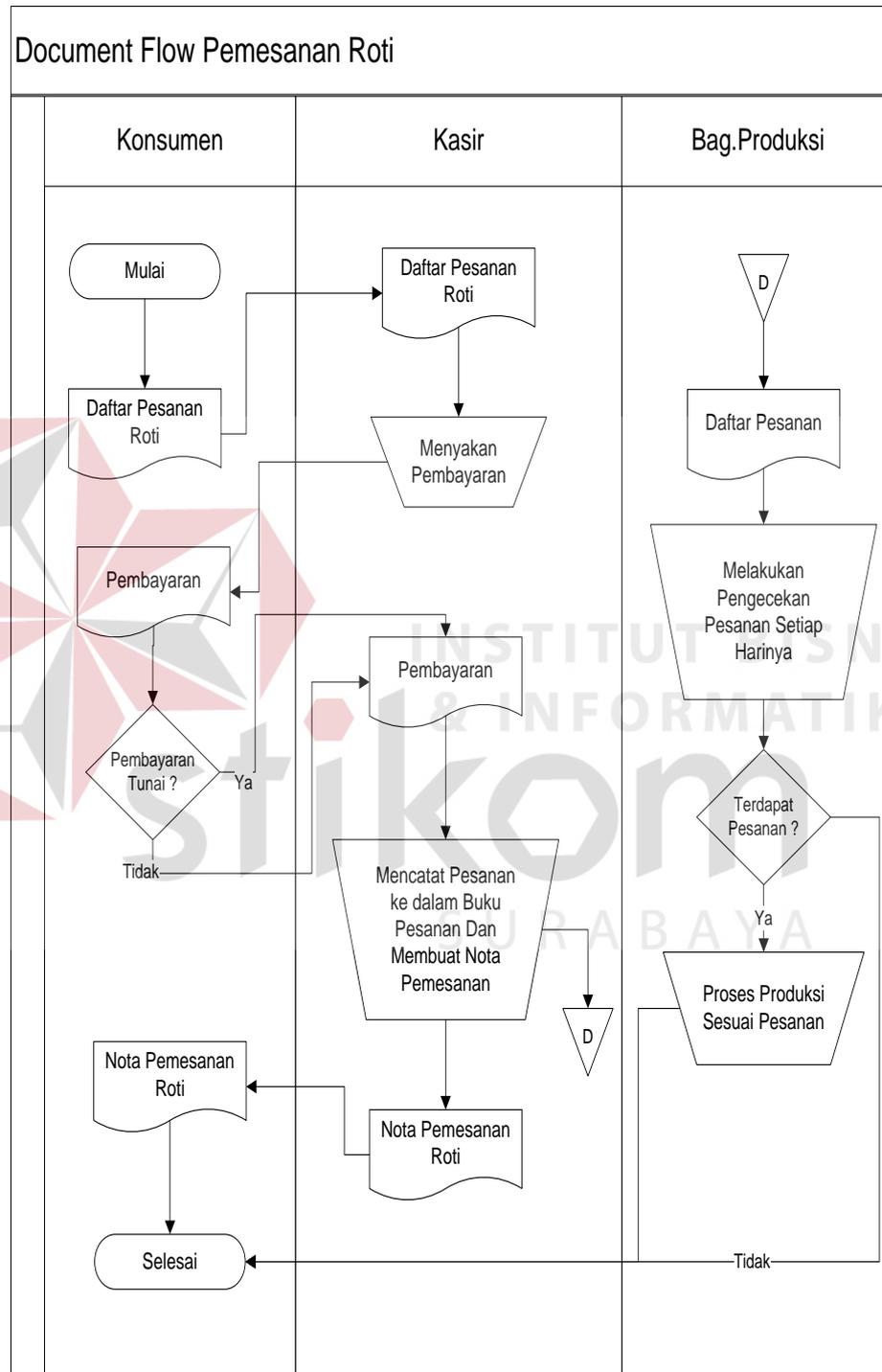


Gambar 3.1 *Document Flow* Penjualan Roti

3.1.2 *Document Flow* Pemesanan Roti

Pada fungsionalitas ini, proses dimulai dari konsumen yang akan memesan roti di Yulia Bakery diterima oleh kasir. Kasir akan mencatat dalam buku pesanan dan membuat nota pemesanan yang diberikan kepada konsumen sebagai tanda bukti ketika akan mengambil roti yang akan dipesan. Pembayaran pesanan dapat dilakukan secara tunai maupun uang muka sebagai tanda jadi.

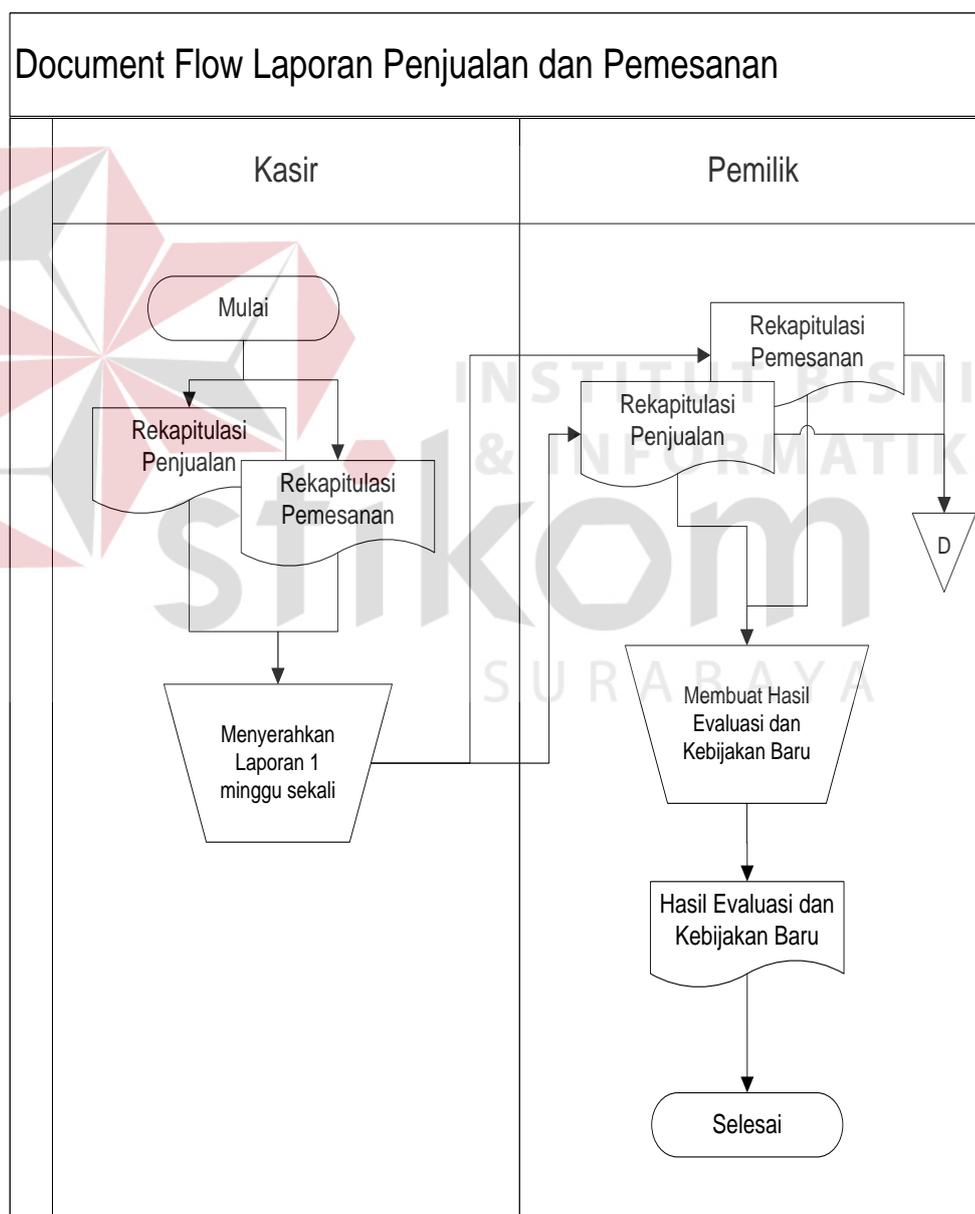
Setiap hari bagian produksi akan mengecek pesanan yang telah diterima oleh kasir untuk selanjutnya dilakukan produksi sesuai dengan pesanan.



Gambar 3.2 Document Flow Pemesanan Roti

3.1.3 Document Flow Laporan Penjualan dan Pemesanan

Pada fungsionalitas laporan penjualan ini proses diawali dari kasir yang memberikan hasil rekapitulasi penjualan dan pemesanan roti selama 7 hari kepada pemilik. Pemilik akan mengecek hasil rekapitulasi penjualan dan pemesanan tersebut dan akan melakukan evaluasi. Hasil dari evaluasi yang dilakukan oleh pemilik akan dijadikan acuan untuk kebijakan selanjutnya pada Yulia Bakery.



Gambar 3.3 Document Flow Laporan Penjualan dan Pemesanan

3.2 Permasalahan

Setelah diketahui proses alur dokumen atau *document flow* yang dilakukan oleh masing-masing pengguna, maka proses berikutnya adalah melakukan analisis kebutuhan yang sesuai dengan proses-proses tersebut. Analisis kebutuhan ini diperlukan untuk merancang kebutuhan masing-masing pengguna sistem. Analisis ini dilakukan dengan kebutuhan masing-masing pengguna sistem. Analisis ini dilakukan pada setiap pengguna yang secara langsung berinteraksi dengan sistem nantinya. Berikut ini hasil analisis kebutuhan untuk masing-masing pengguna :

3.2.1 Analisis pada Proses Staf Kasir

Dalam proses transaksi penjualan terjadi suatu antrian dan menunggunya konsumen yang disebabkan oleh kasir yang melakukan pembuatan nota penjualan dan pencatatan data penjualan ke dalam buku penjualan. Hal seperti ini tentu saja membuat konsumen kecewa dengan pelayanan yang lama.

3.2.2 Analisis pada Proses Bagian Produksi

Dalam proses pemesanan roti terjadi rawannya terlewatnya produksi oleh bagian produksi terhadap pemesanan yang diterima dikarenakan pada penerimaan pesanan jadwal pengambilan pesanan tanggal tidak urut.

3.2.3 Analisis pada Proses Pemilik

Dalam proses penerimaan laporan, pemilik mendapatkan laporan setiap satu minggu sekali. Laporan yang diterima terlalu lama sehingga pembuatan kebijakan hanya bisa dilakukan setelah laporan diterima dan dievaluasi oleh pemilik.

3.3 Solusi Permasalahan

Setelah dilakukan pengumpulan data melalui proses wawancara dan observasi, pengolahan data dari hasil observasi, dilanjutkan dengan melakukan identifikasi masalah, didapatkan suatu permasalahan yang harus diselesaikan dengan memberikan solusi terbaik yang sesuai dengan permasalahan yang ada. Dalam menyelesaikan permasalahan, solusi yang diberikan ialah dengan membangun Sistem Informasi Penjualan yang disesuaikan dengan kebutuhan pihak Yulia Bakery.

Dalam membangun sebuah sistem informasi sebagai solusi pada permasalahan yang ada diperusahaan, dikerjakan melalui beberapa tahapan. Tahapan pengembangan perangkat lunak terdiri dari :

3.3.1 Kebutuhan Perangkat Lunak(*Software Requirement*)

Kebutuhan perangkat lunak merupakan langkah awal dalam membangun sebuah sistem atau aplikasi, hal ini dilakukan agar aplikasi yang dibangun sesuai dengan kebutuhan pengguna. Dalam melakukan identifikasi kebutuhan perangkat lunak, ada beberapa tahapan yang harus dilalui, yaitu :

A. Elisitasi Kebutuhan (*Requirement Elicitation*)

Elisitasi atau pengumpulan kebutuhan merupakan aktivitas awal dalam proses rekayasa kebutuhan (*Requirement Engineering*). Proses elisitasi ini dilakukan dengan cara seperti pertama kali melakukan identifikasi permasalahan, yaitu wawancara dan observasi terhadap *stakeholder* yang terkait. Sebelum kebutuhan dapat dianalisis, dimodelkan, atau ditetapkan, kebutuhan harus dikumpulkan melalui proses elisitasi. Pada tahapan ini dilakukan penyeleksian data

yang diperoleh sehingga dapat diketahui data-data yang digunakan dan yang tidak digunakan terkait dengan pengembangan perangkat lunak.

Berikut ini data yang dikumpulkan melalui proses wawancara ataupun observasi pada perusahaan. Data tersebut meliputi :

a. Data Cabang

Data cabang digunakan untuk proses pengelolaan master cabang yang ada pada sistem nantinya.

b. Data Barang

Data barang digunakan untuk proses pemasukan data sebagai data master pada aplikasi, sehingga ketika aplikasi pada sistem digunakan sudah terdapat data barang yang sudah sesuai dengan barang yang sekarang dijual atau digunakan.

c. Data Karyawan

Data karyawan digunakan untuk pengelolaan terhadap hak akses pengguna atau *user* yang terlibat dalam sistem ini nantinya.

d. Data Konsumen

Data konsumen yang digunakan adalah data konsumen yang sudah pernah melakukan pemesanan roti sebelumnya.

e. Data Penjualan

Data jumlah penjualan yang dikumpulkan adalah data mulai bulan oktober tahun 2012 sampai bulan September tahun 2013. Data ini digunakan sebagai data pembandingan kebenaran antara sistem dengan proses manual.

f. Data Pemesanan

Data pemesanan yang diterima oleh kasir yang dikumpulkan sebagai data pemesanan yang diterima oleh bagian produksi.

B. Analisis Kebutuhan (*Requirement Analysis*)

Sesuai dengan hasil elisitasi data-data yang dibutuhkan untuk membangun perangkat lunak, dibutuhkan beberapa proses yang dapat digabungkan dan dibangun fungsi secara terkomputerisasi.

B.1 Analisis Kebutuhan Kasir

Membutuhkan suatu aplikasi yang dapat mendukung transaksi penjualan dan transaksi pelayanan yang dilakukan oleh kasir sehingga pelayanan dan rekap data penjualan dan data pemesanan menjadi lebih cepat.

B.2 Analisis Kebutuhan Bagian Produksi

Dari hasil analisis yang telah dilakukan pada tahap sebelumnya, bahwa bagian produksi membutuhkan peningkatan kinerja untuk beberapa proses, yaitu :

- a. Membuat suatu *reminder* atau pengingat untuk bagian produksi untuk pesanan yang diterima sehingga meminimalkan terlewatnya pesanan pelanggan
- b. Membuat suatu notifikasi untuk bagian produksi bahwa ada barang yang kadaluarsa untuk hari ini.

B.3 Analisis Kebutuhan Pemilik

Membuat suatu informasi yang disampaikan untuk pemilik yang bisa di akses dimana saja dengan data yang didapat atau *diupdate* setiap tiga jam sekali menggunakan *web-services*, sehingga pemilik bisa memantau transaksi dan dapat membuat kebijakan yang diperlukan dengan cepat.

C. Spesifikasi Kebutuhan Perangkat Lunak (*Requirement Specification*)

Dalam proses membangun dan mengembangkan perangkat lunak, diperlukan perancangan spesifikasi perangkat lunak yang tepat dan detil, dengan tujuan agar perangkat lunak yang akan dikembangkan tersebut memiliki deskripsi fungsi yang sesuai dengan apa yang dibutuhkan oleh masing-masing pengguna. Kebutuhan fungsi tersebut meliputi kebutuhan fungsional dan non-fungsional.

C.1 Kebutuhan Fungsional

Kebutuhan fungsional merupakan dasar dari fungsi penyusunan fungsi-fungsi yang akan dibangun didalam perangkat lunak. Fungsi-fungsi perangkat lunak tersebut telah melewati proses identifikasi kebutuhan setiap pengguna atau *user*. Adapun kebutuhan fungsional yang sudah disetujui oleh *stakeholder* tersebut meliputi staf kasir, bagian produksi, dan pemilik.

C.1.1 Staf Kasir

Kebutuhan fungsional beserta penjelasannya untuk kasir dapat dilihat pada tabel 3.1.

Tabel 3.1 Detail Kebutuhan Fungsi Melakukan Proses Transaksi Penjualan

Nama Fungsi	Melakukan Proses Transaksi Penjualan	
Stakeholder	Staf Kasir	
Deskripsi	Fungsi ini digunakan untuk melakukan proses transaksi penjualan.	
Kondisi Awal	<ol style="list-style-type: none"> 1. Data cabang sudah tersedia 2. Data pengguna sudah tersedia. 3. Data barang sudah tersedia 	
Alur Normal	Aksi Pengguna	Respon Sistem
	1. Pengguna Memasukkan <i>Username</i> dan <i>Password</i>	<ol style="list-style-type: none"> 1. a) Sistem Melakukan Otentikasi Masukkan Pengguna b) Sistem Menampilkan <i>Form</i> Utama sesuai dengan otorisasi

	Aksi Pengguna	Respon Sistem
Alur Normal	2. Pengguna memilih sub menu “Penjualan” yang terdapat pada menu “Transaksi”.	2.a) Sistem menampilkan <i>form</i> transaksi penjualan. b) Sistem sudah melakukan <i>generate</i> otomatis pada nomor nota penjualan. c) Nama kasir yang terdapat pada <i>form</i> transaksi penjualan sudah otomatis terisi sesuai dengan pengguna yang melakukan <i>login</i> ke dalam aplikasi.
	3. Pengguna melakukan <i>scan barcode</i> barang yang akan dibeli untuk dimasukkan ke dalam <i>field textbox</i> yang terdapat pada <i>form</i> transaksi penjualan.	3. a) Sistem menampilkan nama barang, jenis barang, harga barang dan <i>quantity</i> yang otomatis terisi sejumlah 1 <i>item</i> . b) Sistem menampilkan detail penjualan di dalam <i>gridview</i> detail penjualan.
	4. Pengguna melakukan tombol bayar selesai melakukan <i>entry</i> detail penjualan.	4. a) Sistem menampilkan <i>form</i> pembayaran penjualan. b) Sistem Otomatis Menampilkan Total Penjualan sesuai dengan penjualan.
	5. Pengguna Memasukkan pembayaran ke dalam <i>field</i> pembayaran.	5. Sistem Otomatis melakukan perhitungan antara uang pembayaran – total penjualan dan ditampilkan dalam <i>field</i> kembalian pembayaran.
	6. Pengguna menekan tombol bayar pada <i>form</i> pembayaran penjualan	6. a) Sistem akan Menampilkan <i>report</i> nota penjualan b) Sistem akan melakukan print nota penjualan
Alur Eksepsi	1. Pengguna salah memasukkan <i>username</i> atau <i>password</i> .	1. Sistem Menampilkan pesan kesalahan <i>input-an username</i> ataupun <i>password</i> maupun kedua-duanya.
	2. Pengguna tidak bisa masuk ke <i>form</i> utama dari aplikasi sistem	2. Sistem menampilkan pesan <i>database</i> tidak tersambung,
	3. Pengguna mengisi pembayaran kurang dari total penjualan	3. Tombol bayar pada <i>form</i> pembayaran penjualan tidak bisa digunakan atau <i>disable</i> .
Kondisi Akhir	1. Fungsi ini dapat menyimpan data transaksi penjualan ke dalam <i>database</i> penjualan dan detail penjualan. 2. Fungsi ini dapat mencetak nota penjualan.	

Kebutuhan Non-Fungsional	<i>Security</i>	Sistem hanya bisa diakses oleh pengguna yang memiliki hak akses melakukan transaksi penjualan
	<i>Correctness</i>	Perhitungan antara total penjualan dengan harga barang dan <i>quantity</i> harus sesuai.
	<i>Interface</i>	1. Menu tersedia dalam bahasa indonesia. 2. Warna tampilan <i>form</i> di-design tidak terlalu mencolok.
	<i>Performance</i>	Dalam proses pelayanan transaksi penjualan sistem harus mampu melakukan transaksi minimal 10 barang dengan catatan waktu di bawah 2 menit.
	<i>Operability</i>	1. Posisi menu <i>button</i> antara halaman sama. 2. Setiap <i>field</i> dalam <i>form</i> terdapat fungsi <i>tab</i> yang disesuaikan dengan alir proses bisnis

Kebutuhan fungsional kasir yang lainnya adalah melakukan transaksi Pemesanan Roti. Lebih jelasnya dapat dilihat pada tabel 3.2.

Tabel 3.2 Detail Kebutuhan Fungsi Melakukan Proses Transaksi Pemesanan Roti

Nama Fungsi	Melakukan Proses Transaksi Pemesanan Barang	
Stakeholder	Staf Kasir	
Deskripsi	Fungsi ini digunakan untuk melakukan proses transaksi pemesanan barang atau roti.	
Kondisi Awal	1. Data cabang sudah tersedia 2. Data pengguna sudah tersedia. 3. Data barang sudah tersedia 4. Data Konsumen sudah tersedia	
Alur Normal	Aksi Pengguna	Respon Sistem
	1. Pengguna Memasukkan <i>Username</i> dan <i>Password</i>	1. a) Sistem Melakukan Otentikasi Masukkan Pengguna b) Sistem Menampilkan <i>Form</i> utama sesuai dengan otorisasi

	Aksi Pengguna	Respon Sistem
<p>Alur Normal</p>	<p>2. Pengguna memilih sub menu “Pemesanan” yang terdapat pada menu “Transaksi”.</p>	<p>pengguna</p> <p>2.a) Sistem menampilkan <i>form</i> master konsumen untuk data konsumen yang akan melakukan pemesanan roti.</p> <p>b) Sistem sudah melakukan <i>generate</i> otomatis pada nomor “id_konsumen”</p>
	<p>3. Pengguna memilih tombol “Simpan dan Lanjut Pemesanan”</p>	<p>3.a) Sistem menampilkan <i>form</i> transaksi pemesanan.</p> <p>b) Sistem sudah melakukan <i>generate</i> otomatis pada nomor nota pemesanan.</p> <p>c) Nama kasir yang terdapat pada <i>form</i> transaksi penjualan sudah otomatis terisi sesuai dengan pengguna yang melakukan <i>login</i> ke dalam aplikasi.</p>
	<p>4. Pengguna memasukkan kode barang dari barang yang akan dibeli ke dalam <i>field</i> kode barang dan menekan tombol “enter” pada <i>keyboard</i>.</p>	<p>4. Sistem menampilkan nama barang, jenis barang, dan harga barang.</p>
	<p>5. Pengguna melakukan pengisian <i>quantity</i> pemesanam ke dalam <i>field quantity</i>. Dan menekan tombol “tambah”</p>	<p>5. a) Sistem menampilkan detail pemesanan di dalam <i>gridview</i> detail pemesanan.</p> <p>b) Sistem Menampilkan Total Pemesanan sesuai dengan pesanan,</p>
	<p>6. Pengguna Menekan tombol pembayaran pada <i>form</i> transaksi pemesanan.</p>	<p>6.a) Sistem akan menampilkan <i>form</i> pembayaran pemesanan.</p> <p>b) Sistem akan menampilkan total biaya pemesanan dan minimal pembayaran yang didapat dari 50% total biaya pemesanan.</p>
	<p>7. Pengguna menekan tombol “Bayar”</p>	<p>6. a) Sistem akan Menampilkan <i>report</i> nota pemesanan</p> <p>b) Sistem akan melakukan print nota pemesanan.</p>
		Aksi Pengguna
<p>Alur Eksepsi</p>	<p>1. Pengguna salah memasukkan <i>username</i> atau <i>password</i>.</p>	<p>1. Sistem Menampilkan pesan kesalahan <i>input-an username</i> ataupun <i>password</i> maupun kedua-duanya.</p>

	Aksi Pengguna	Respon Sistem
Alur Eksepsi	2. Pengguna tidak bisa masuk ke <i>form</i> utama dari aplikasi sistem	2. Sistem menampilkan pesan <i>database</i> tidak tersambung,
	4. Pengguna mengisi pembayaran kurang dari total pemesanan.	4. Tombol bayar pada <i>form</i> pembayaran penjualan tidak bisa digunakan atau <i>disable</i> .
Kondisi Akhir	3. Fungsi ini dapat menyimpan data transaksi penjualan ke dalam <i>database</i> pemesanan dan detail pemesanan. 4. Fungsi ini dapat mencetak nota pemesanan.	
Kebutuhan Non-Fungsional	<i>Security</i>	Sistem hanya bisa diakses oleh pengguna yang memiliki hak akses melakukan transaksi pemesanan
	<i>Correctness</i>	Perhitungan antara total pemesanan dengan harga barang dan <i>quantity</i> harus sesuai.
	<i>Interface</i>	1. Menu tersedia dalam bahasa indonesia. 2. Warna tampilan <i>form</i> di-design tidak terlalu mencolok.
	<i>Performance</i>	Dalam proses pelayanan transaksi penjualan sistem harus mampu melakukan transaksi minimal 10 barang dengan catatan waktu di bawah 2 menit.
	<i>Operability</i>	1. Posisi menu <i>button</i> antara halaman sama. 2. Setiap <i>field</i> dalam <i>form</i> terdapat fungsi <i>tab</i> yang disesuaikan dengan alir proses bisnis

Kebutuhan fungsional kasir yang lainnya adalah Melakukan transaksi Pengambilan Pemesanan Lebih jelasnya dapat dilihat pada tabel 3.3.

Tabel 3.3 Detail Kebutuhan Fungsi Melakukan Transaksi Pengambilan Pemesanan

Nama Fungsi	Melakukan Proses Transaksi pengambilan pemesanan	
Stakeholder	Staf Kasir	
Deskripsi	Fungsi ini digunakan untuk melakukan proses pengambilan pemesanan.	
Kondisi Awal	<ol style="list-style-type: none"> 1. Data cabang sudah tersedia 2. Data pengguna sudah tersedia. 3. Data barang sudah tersedia 4. Data Konsumen sudah tersedia 5. Data Karyawan sudah tersedia 6. Data transaksi pemesanan sudah tersedia 	
Alur Normal	Aksi Pengguna	Respon Sistem
	<ol style="list-style-type: none"> 1. Pengguna Memasukkan <i>Username</i> dan <i>Password</i> 2. Pengguna memilih sub menu “Pengambilan pemesanan” yang terdapat pada menu “Transaksi”. 3. Pengguna memasukkan Kode Nota Pemesanan 4. Pengguna melakukan <i>checklist</i> pada status pengambilan barang 5. Pengguna menekan tombol simpan. 	<ol style="list-style-type: none"> 1. a) Sistem melakukan otentikasi masukkan pengguna. b) Sistem menampilkan “Halaman Utama”, dan memberikan otorisasi kepada pengguna. 2. Sistem dapat Menampilkan “Halaman Pengambilan Pemesanan” 3. a) Sistem Menampilkan Detail Barang Pesanan b) Sistem Menampilkan Total Pembayaran yang Belum di Bayar 4. Button simpan menjadi <i>enable</i> 5. Data Pemesanan status berubah menjadi sudah diambil dan sudah hilang dari daftar pesanan
Alur Eksepsi	Aksi Pengguna	Respon Sistem
	<ol style="list-style-type: none"> 1. Pengguna salah memasukkan <i>username</i> atau <i>password</i>. 2. Pengguna tidak bisa masuk ke <i>form</i> utama dari aplikasi sistem 	<ol style="list-style-type: none"> 1. Sistem Menampilkan pesan kesalahan <i>input-an username</i> ataupun <i>password</i> maupun kedua-duanya. 2. Sistem menampilkan pesan <i>database</i> tidak tersambung,

	Aksi Pengguna	Respon Sistem
Alur Eksepsi	3. Pengguna salah memasukkan nota pesanan	3. Tombol bayar pada <i>form</i> pembayaran penjualan tidak bisa digunakan atau <i>disable</i> .
Kondisi Akhir	1. Fungsi ini akan mengubah data status pesanan. 2. Daftar pesanan yang belum selesai sudah berubah.	
Kebutuhan Non-Fungsional	<i>Security</i>	Sistem hanya bisa diakses oleh pengguna yang memiliki hak akses melakukan transaksi pemesanan
	<i>Correctness</i>	Detail pesanan dengan nota pemesanan yang dimasukkan sesuai.
	<i>Interface</i>	1. Menu tersedia dalam bahasa indonesia. 2. Warna tampilan <i>form</i> di- <i>design</i> tidak terlalu mencolok.
	<i>Operability</i>	1. Posisi menu <i>button</i> antara halaman sama. 2. Setiap <i>field</i> dalam <i>form</i> terdapat fungsi <i>tab</i> yang disesuaikan dengan alir proses bisnis

C.1.2 Bagian Produksi

Kebutuhan fungsional dan beserta penjelasan untuk bagian produksi dapat dilihat pada tabel 3.4.

Tabel 3.4 Detail Kebutuhan Fungsi *Reminder* Pesanan

Nama Fungsi	<i>Reminder</i> Pesanan	
Stakeholder	Bagian Produksi	
Deskripsi	Fungsi ini digunakan untuk <i>reminder</i> pesanan yang diterima untuk hari ini dan hari berikutnya.	
Kondisi Awal	Data pesanan sudah tersedia	
Alur Normal	Aksi Pengguna	Respon Sistem
	1. Pengguna Memasukkan <i>Username</i> dan <i>Password</i>	1. a) Sistem melakukan otentikasi masukkan pengguna. b) Sistem menampilkan "Halaman Utama", dan memberikan otorisasi kepada

	Aksi Pengguna	Respon Sistem
Alur Normal		pengguna. c) Sistem menampilkan Informasi Bahwa Ada Pesanan
	2. Pengguna menekan Tombol “Lihat Pesanan” pada Informasi Ada Pesanan.	2. Sistem dapat menampilkan halaman pesanan untuk hari ini dan besok secaraurut sesuai dengan tanggal dan jam pengamban
	3. Pengguna Merubah status pengerjaan pada halaman pesanan untuk hari ini dan hari esok dan menekan tombol simpan.	3. a) Sistem dapat mengubah status pengerjaan pada database. b) menampilkan informasi bahwa status pengerjaan sudah diubah
Alur Eksepsi	Aksi Pengguna	Respon Sistem
	1. Pengguna salah memasukkan <i>username</i> atau <i>password</i> .	1. Sistem Menampilkan pesan kesalahan <i>input-an username</i> ataupun <i>password</i> maupun kedua-duanya.
	2. Pengguna tidak bisa masuk ke <i>form</i> utama dari aplikasi sistem	2. Sistem menampilkan pesan <i>database</i> tidak tersambung,
Kondisi Akhir	Fungsi ini akan memberikan <i>reminder</i> pesanan untuk hari ini dan hari esok.	
Kebutuhan Non-Fungsional	<i>Security</i>	Sistem hanya bisa diakses oleh pengguna yang memiliki hak akses melakukan transaksi pemesanan.
	<i>Interface</i>	1. Menu tersedia dalam bahasa indonesia. 2. Warna tampilan <i>form</i> di- <i>design</i> tidak terlalu mencolok.
	<i>Operability</i>	1. Posisi menu <i>button</i> antara halaman sama. 2. Setiap <i>field</i> dalam <i>form</i> terdapat fungsi <i>tab</i> yang disesuaikan dengan alir proses bisnis setiap proses yang dijalankan, sehingga mudah dioperasikan.

C.1.3 Pemilik

Kebutuhan fungsional dan beserta penjelasan untuk pemilik dapat dilihat pada tabel 3.5.

Tabel 3.5 Detail Kebutuhan Fungsi Laporan Penjualan dan Pemesanan

Nama Fungsi	Laporan Penjualan dan Pemesanan	
Stakeholder	Pemilik	
Deskripsi	Fungsi ini digunakan untuk mencetak laporan penjualan dan pemesanan sesuai dengan parameter yang diinginkan	
Kondisi Awal	<ol style="list-style-type: none"> 1. Data cabang sudah tersedia 2. Data barang sudah tersedia 3. Data karyawan sudah tersedia 4. Data konsumen sudah tersedia 5. Data transaksi penjualan sudah tersedia 6. Data transaksi pemesanan sudah tersedia 	
Alur Normal	Aksi Pengguna	Respon Sistem
	<ol style="list-style-type: none"> 1. Pengguna memilih menu laporan pada tampilan <i>website</i>. 2. Pengguna memilih sub menu laporan yang dipilih 3. Pengguna memasukkan Kode Cabang ke dalam <i>field</i> kode cabang yang ada di tampilan laporan dan menekan tombol "Enter" pada <i>keyboard</i>. 4. Pengguna menginputkan parameter tanggal awal dan tanggal akhir untuk <i>range</i> tanggal laporan yang diinginkan dan menekan tombol "Lihat Laporan" 5. Pengguna Menekan gambar <i>pdf</i> pada laporan yang dihasilkan 	<ol style="list-style-type: none"> 1. Sistem menampilkan beberapa sub menu laporan untuk pemilik. 2. Sistem dapat menampilkan tampilan sub menu yang dipilih oleh pengguna 3. Sistem menampilkan nama cabang, dan alamat cabang berdasarkan <i>inputan</i> pada <i>field</i> kode cabang. 4. Sistem dapat menampilkan hasil laporan yang diinginkan oleh pengguna sesuai dengan sub menu dan sesuai dengan parameter. 5. Sistem akan melakukan download laporan ke dalam format <i>pdf</i>
Alur Eksepsi	<ol style="list-style-type: none"> 1. Pengguna salah menginputkan kode cabang. 	<ol style="list-style-type: none"> 1. Sistem menampilkan pesan kode cabang yang diinputkan salah.
	<ol style="list-style-type: none"> 2. Pengguna salah menginputkan parameter tanggal akhir lebih besar dari pada tanggal awal 	<ol style="list-style-type: none"> 2. Sistem menampilkan pesan <i>range</i> tanggal akhir harus lebih besar dari pada tanggal awal

Kondisi Akhir	Fungsi ini akan menghasilkan laporan penjualan dan pemesanan	
Kebutuhan Non-Fungsional	<i>Security</i>	Sistem hanya bisa diakses oleh pengguna yang memiliki hak akses melakukan melihat laporan penjualan dan pemesanan.
	<i>Interface</i>	1. Menu tersedia dalam bahasa indonesia. 2. Warna tampilan <i>form</i> di-design tidak terlalu mencolok.
	<i>Performance</i>	1. Sistem dapat mengupdate data setiap 3 jam sekali. 2. Laporan yang diterima sesuai dengan penjualan sampai batas waktu terakhir penarikan data dari <i>database</i> sementara ke dalam <i>database</i> server.

C.2 Kebutuhan Non-Fungsional

Dalam penerapan fungsi-fungsi tersebut dengan tujuan mendukung kinerja fungsi utama dari sistem, dibutuhkan juga fungsi non-fungsional. Adapun kebutuhan non-fungsional yang sudah disetujui oleh *stakeholder* tersebut dapat dilihat lebih detail pada tabel 3.6.

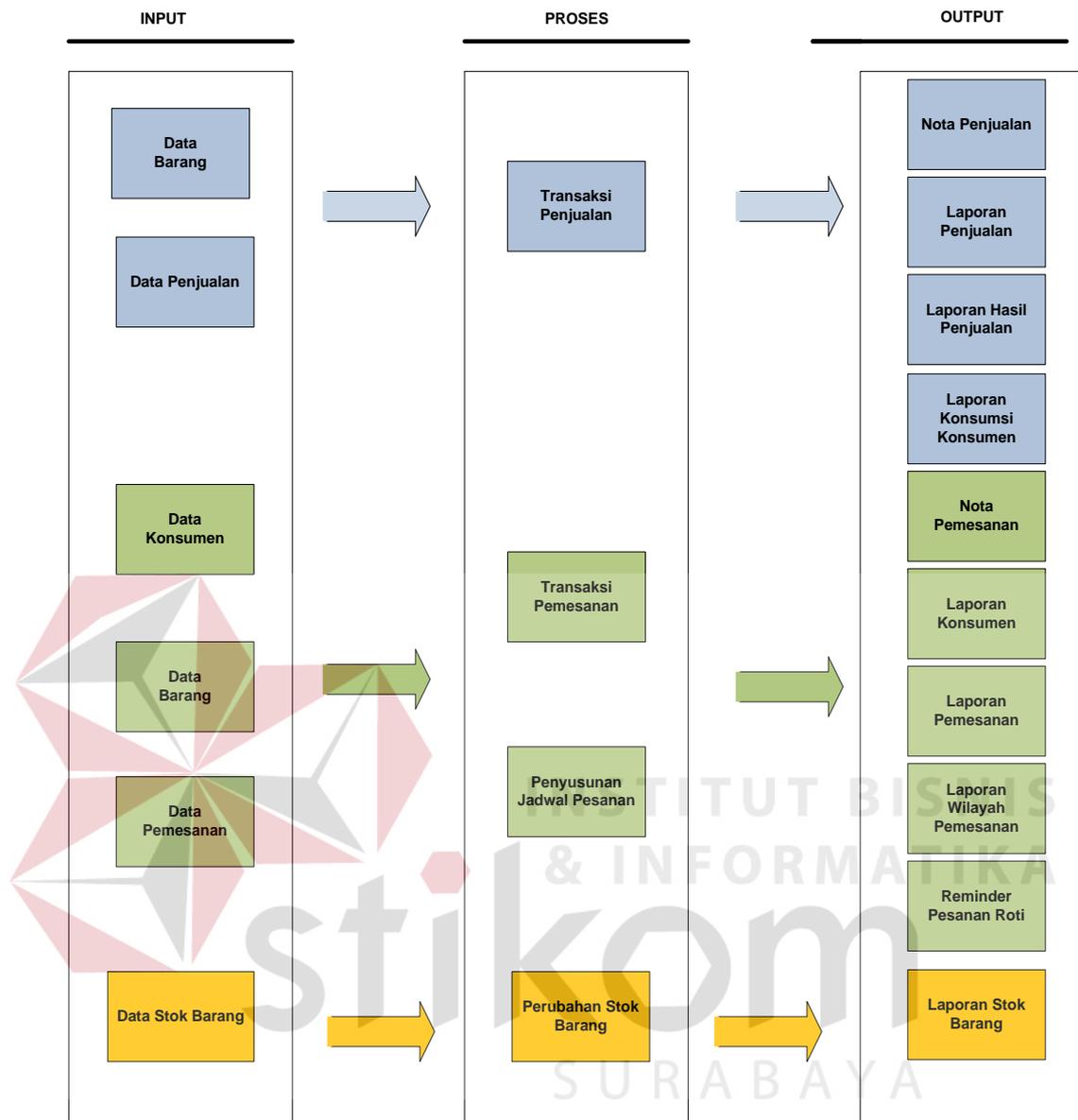
Tabel 3.6 Hubungan Fungsional dan Non-Fungsional Sistem

No.	Stakeholder	Fungsional Sistem	Non-Fungsional Sistem
1	Kasir	a. Melakukan transaksi penjualan	a. <i>Security</i> b. <i>Correctness</i> c. <i>Interface</i> d. <i>Performance</i> e. <i>Operability</i>
		b. Melakukan transaksi pemesanan	a. <i>Security</i> b. <i>Correctness</i> c. <i>Interface</i>

No.	Stakeholder	Fungsional Sistem	Non-Fungsional Sistem
			d. <i>Operability</i>
		c. Melakukan proses transaksi pengambilan pesanan	a. <i>Security</i> b. <i>Correctness</i> c. <i>Interface</i> d. <i>Operability</i>
2.	Bagian Produksi	a. Kebutuhan Fungsi <i>reminder</i> Pesanan	a. <i>Security</i> b. <i>Interface</i> c. <i>Operability</i>
3	Pemilik	a. Laporan Penjualan dan Pemesanan	a. <i>Security</i> b. <i>Interface</i> c. <i>Performance</i> d. <i>operability</i>

3.3.2 Desain Sistem (*Software Design*)

Perancangan sistem merupakan tahap pengembangan setelah analisis sistem dilakukan. Dalam perancangan sistem akan memberikan gambaran tentang langkah-langkah yang akan dilakukan dalam membangun sistem yang akan dibuat. Gambaran umum mengenai sistem yang akan dibangun dapat dilihat pada gambar 3.4 Blok Diagram pada halaman 29. Blok Diagram tersebut menjelaskan tentang apa saja yang menjadi *input* dan akan diproses oleh sistem sehingga menghasilkan *output* yang akan dihasilkan oleh sistem yang dibangun.

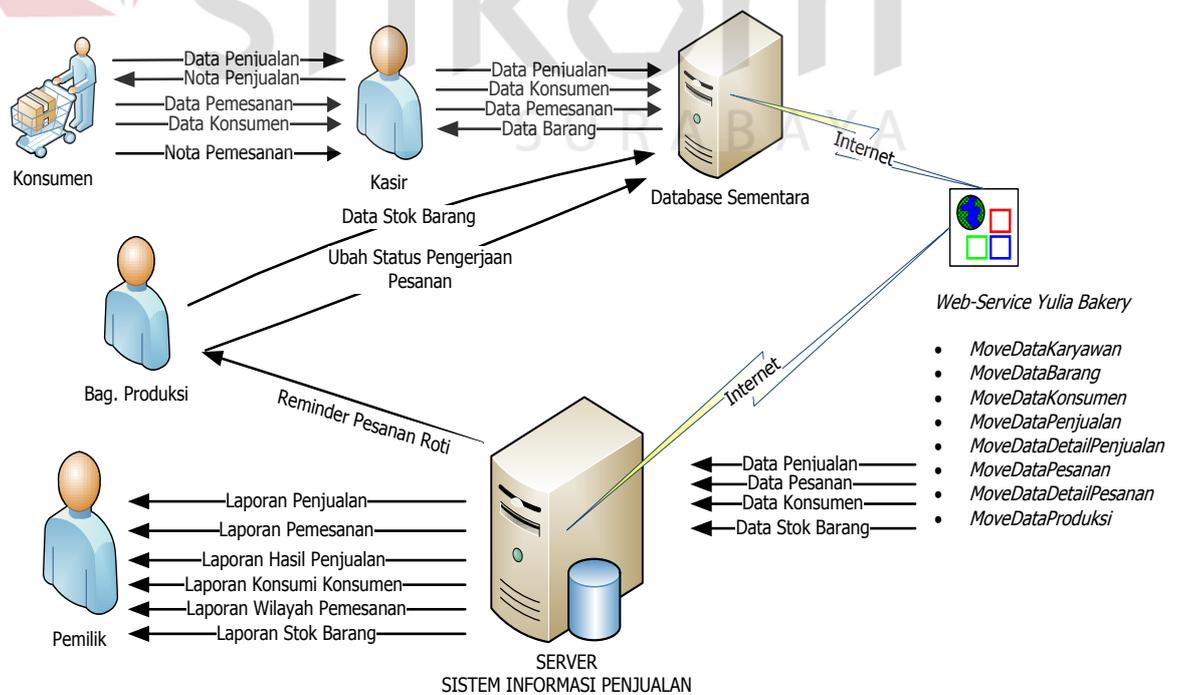


Gambar 3.4 Blok Diagram Perancangan Sistem

Pada gambar di atas menjelaskan bahwa data inputan yang dibutuhkan oleh sistem untuk mengolah transaksi penjualan adalah data barang dan data penjualan yang diperoleh dari transaksi penjualan setiap hari. Laporan yang nanti dihasilkan dari transaksi penjualan ini adalah laporan penjualan, laporan hasil penjualan, dan laporan jenis konsumsi konsumen. Sedangkan data yang

diperlukan untuk pemesanan roti adalah data konsumen, data barang, dan data pemesanan. Dari data pemesanan roti ini terjadi proses transaksi pemesanan dan penyusunan jadwal pesanan pelanggan. Laporan yang dihasilkan dari proses pemesanan ini adalah laporan konsumen, laporan pemesanan, dan laporan wilayah pemesanan. Dalam proses penyusunan jadwal pesanan, sistem akan memberikan *reminder* untuk bagian produksi yang dijadikan pengingat bahwa ada pesanan untuk hari ini yang harus diproduksi.

Model pengembangan yang digunakan dalam Tugas Akhir ini yaitu berupa rancang bangun sistem informasi penjualan roti menggunakan *web-service*. Dengan mengumpulkan data-data pada transaksi setiap harinya dan data-data pemesanan yang nantinya data tersebut digunakan sebagai inputan. Selanjutnya data tersebut akan diproses dan dianalisis sesuai dengan informasi yang dihasilkan. Secara garis besar dapat digambarkan dengan menggunakan arsitektur seperti pada gambar 3.5.



Gambar 3.5 Skema Diagram

Sistem informasi penjualan roti pada Yulia Bakery ini dilakukan untuk mengoptimalkan tingkat pelayanan konsumen oleh kasir. data penjualan yang akan diinputkan ke dalam sistem dengan database sementara adalah data barang, dan data jumlah barang yang dibeli oleh konsumen dari proses ini akan menghasilkan struk atau nota untuk konsumen. Sedangkan untuk pelayanan pesanan yang diterima oleh kasir, data yang diinputkan ke dalam sistem dengan database sementara adalah data pesanan, data konsumen, dan data barang dan sistem akan memberikan output berupa struk atau nota pesanan konsumen. Dari data yang telah disimpan pada database sementara, setiap tiga jam sekali data tersebut akan mengupdate data yang ada di dalam database pusat dengan menggunakan *web-service*. Data-data yang akan dipindahkan dari database sementara ke dalam database pusat adalah data penjualan, data pesanan, data stok barang, dan data konsumen.

Data pesanan yang sudah masuk pada database pusat server nantinya akan memberikan suatu *reminder* untuk bagian produksi bahwa terdapat pesanan yang harus di produksi. Dengan adanya *reminder* ini diharapkan akan mengurangi terlewatnya pesanan konsumen yang berdampak kerugian bagi Yulia Bakery. Untuk produksi setiap harinya bagian produksi akan menambahkan *stock* roti ke dalam sistem database sementara ketika setelah produksi dan akan dimasukkan ke dalam rak display.

Untuk laporan yang bisa diinformasikan kepada pemilik, data yang diambil adalah data yang sudah masuk dalam database pusat yang berada pada pusat Yulia Bakery. Informasi yang didapat oleh pemilik adalah laporan penjualan, laporan pemesanan, laporan hasil penjualan, laporan konsumsi

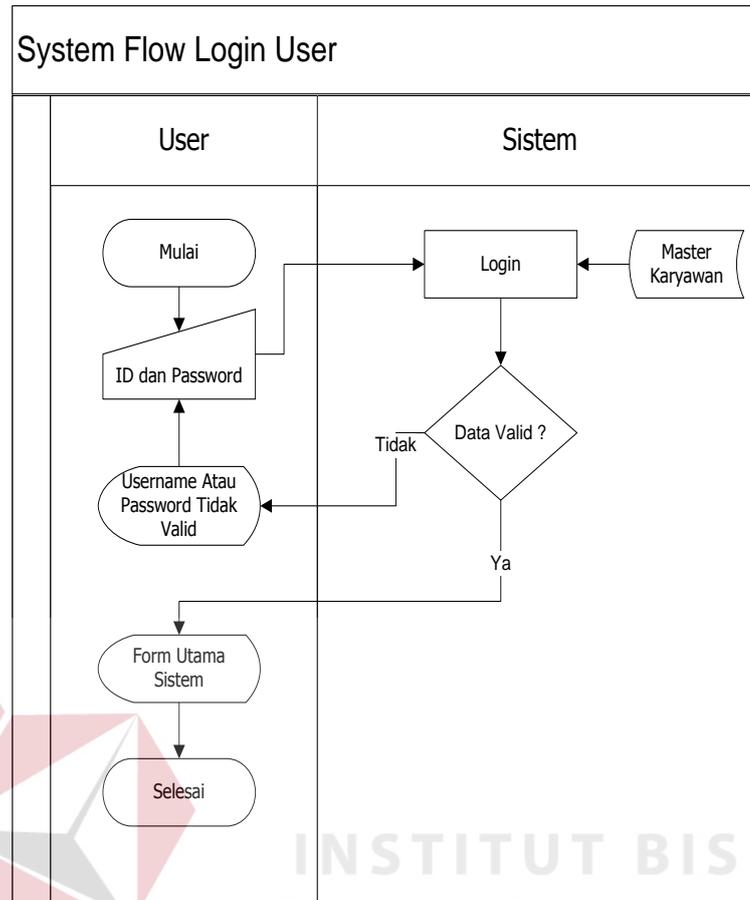
konsumen, laporan wilayah pemesanan, dan laporan stok barang. Laporan tersebut nantinya akan dijadikan pemilik sebagai acuan untuk membuat kebijakan.

Proses desain tahap selanjutnya dilakukan berdasarkan hasil analisis kebutuhan yang telah dilakukan sebelumnya. Beberapa model perancangan perangkat lunak tersebut adalah sebagai berikut :

1. *System Flow*
2. *Data Flow Diagram*
3. *Entity Relationship Diagram (ERD)*,
4. *Rancangan Database*
5. *Rancangan Tampilan Antar Muka(Interface)*

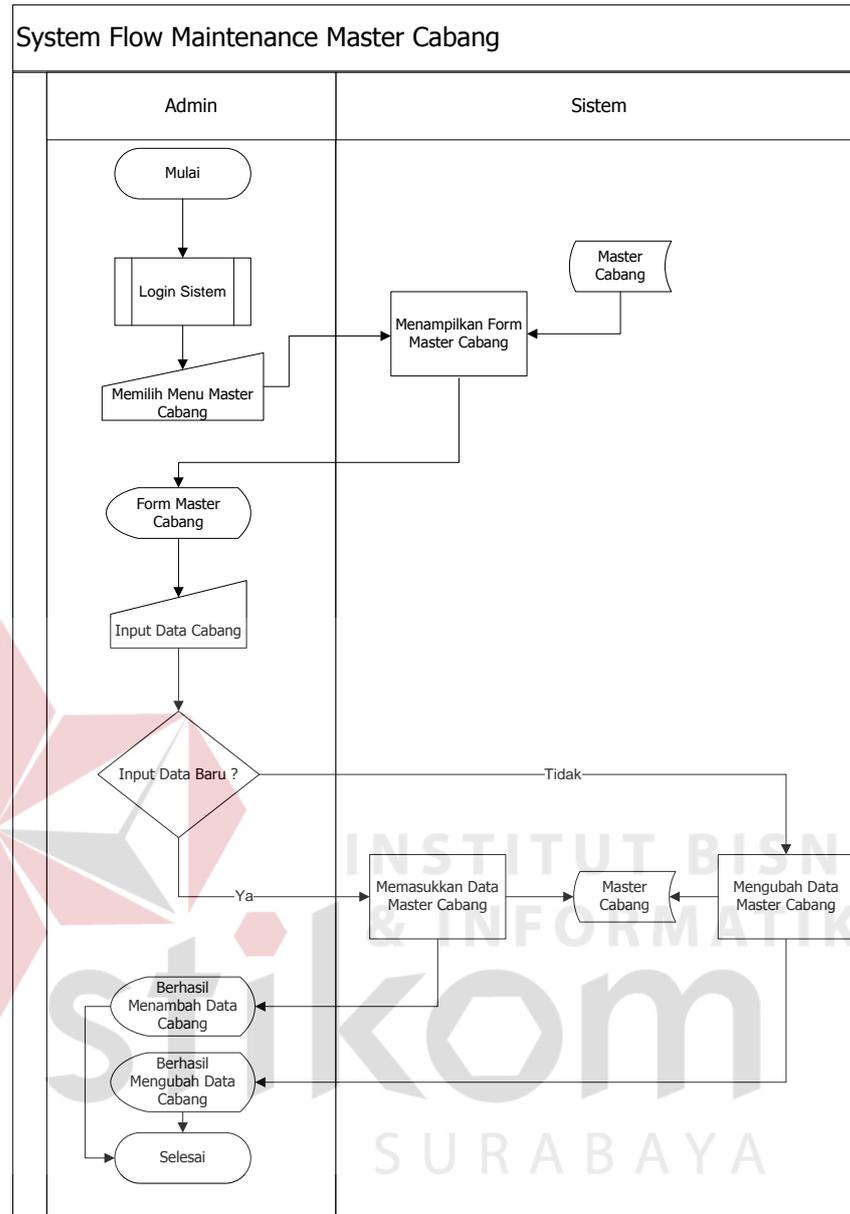
A. *System Flow*

Gambar 3.6 menjelaskan tentang alur sistem dalam melakukan *login* ke dalam sistem. Untuk dapat melakukan *login* ke dalam sistem, karyawan harus memiliki *username* dan *password* yang sudah didapatkan oleh setiap karyawan. Karyawan yang sudah memiliki *username* dan *password* memasukkan ke dalam *field* yang sudah terdapat di dalam aplikasi. Aplikasi akan melakukan pengecekan di dalam *database* master karyawan, jika *username* dan *password* yang diinputkan sudah sesuai dengan yang terdapat didalam *database* maka aplikasi akan menampilkan *form* utama aplikasi sesuai dengan jabatan dari karyawan tersebut. Sedangkan apabila *username* dan *password* yang diinputkan salah maka akan proses akan kembali ke dalam memasukkan *username* dan *password* kembali.



Gambar 3.6 System Flow Login Sistem

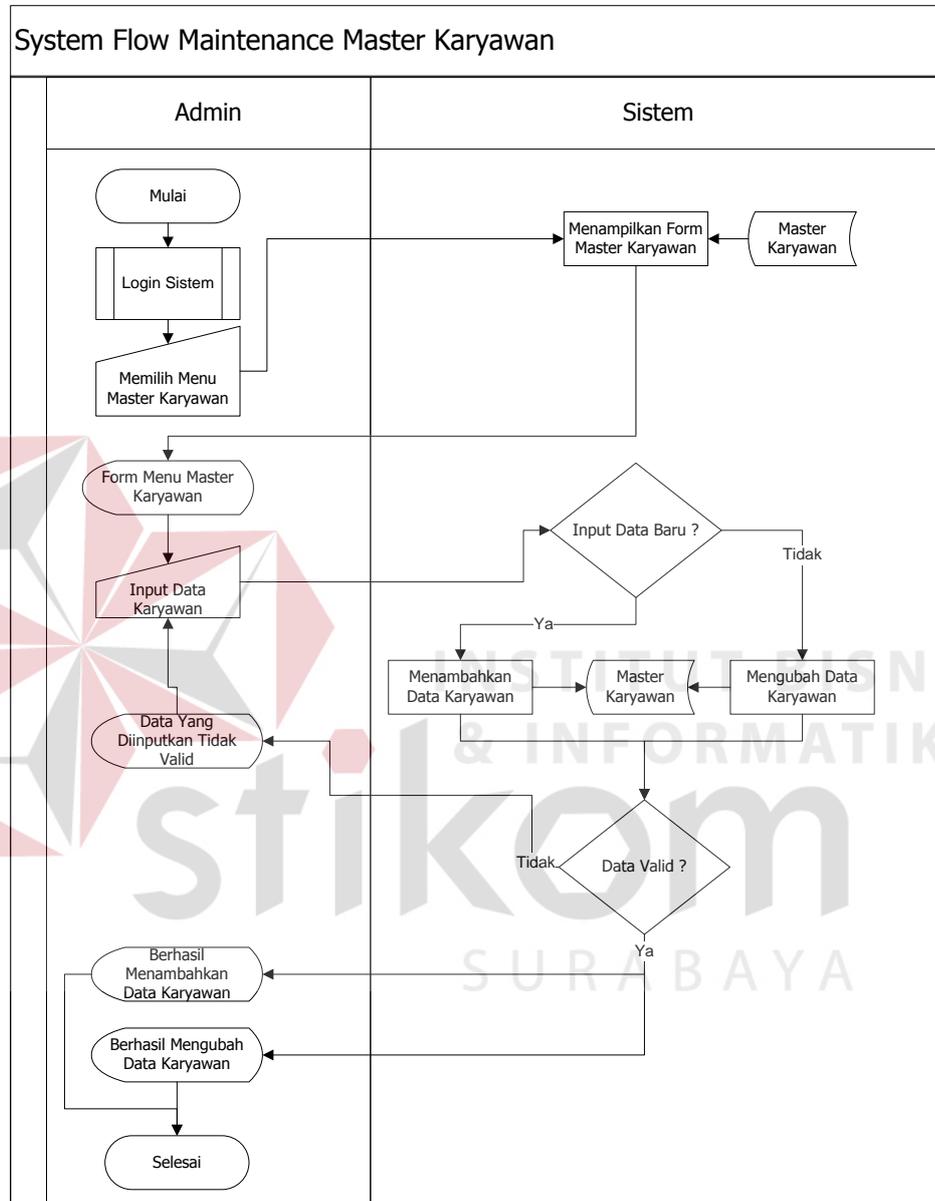
Gambar 3.7 menjelaskan tentang *maintenance* master cabang untuk melakukan penambahan maupun perubahan pada data setiap cabang Yulia Bakery. Dalam melakukan proses *maintenance* master cabang ini hanya bisa dilakukan oleh karyawan yang sudah mendapatkan otoritas sebagai admin, untuk melakukan inputan data cabang baru data akan diambil dari database server untuk mengecek kode cabang yang bisa digunakan untuk cabang dari cabang baru. Jika admin akan melakukan perubahan data cabang, data diambil dari database yang berada di cabang yang nantinya akan di update ke dalam database yang berada di server pusat.



Gambar 3.7 System Flow Maintenance Master Cabang

Gambar 3.8 menjelaskan alur proses dari *Maintenance* master karyawan. Sistem ini diawali dari admin yang *login* ke dalam sistem. Admin akan memilih menu master karyawan dan memilih menu untuk menginputkan data baru atau merubah data karyawan yang sudah ada. Sistem akan melakukan validasi terhadap data yang diinputkan oleh admin, jika inputan data valid sistem akan memberikan informasi inputan yang telah diinputkan oleh admin. Sedangkan jika inputan yang

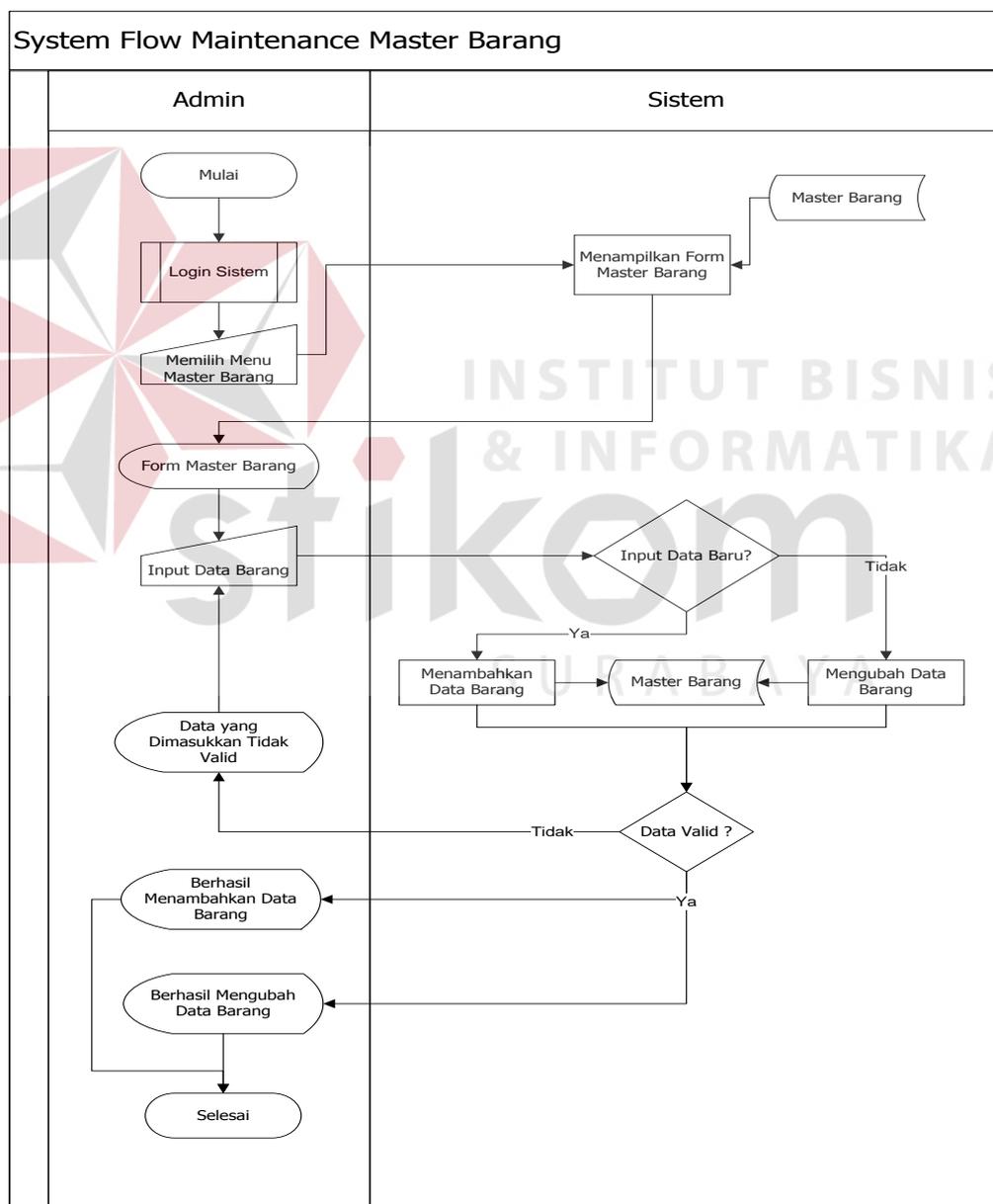
diinputkan oleh admin tidak valid maka sistem akan memberikan informasi inputan tidak valid.



Gambar 3.8 System Flow Maintenance Master Karyawan

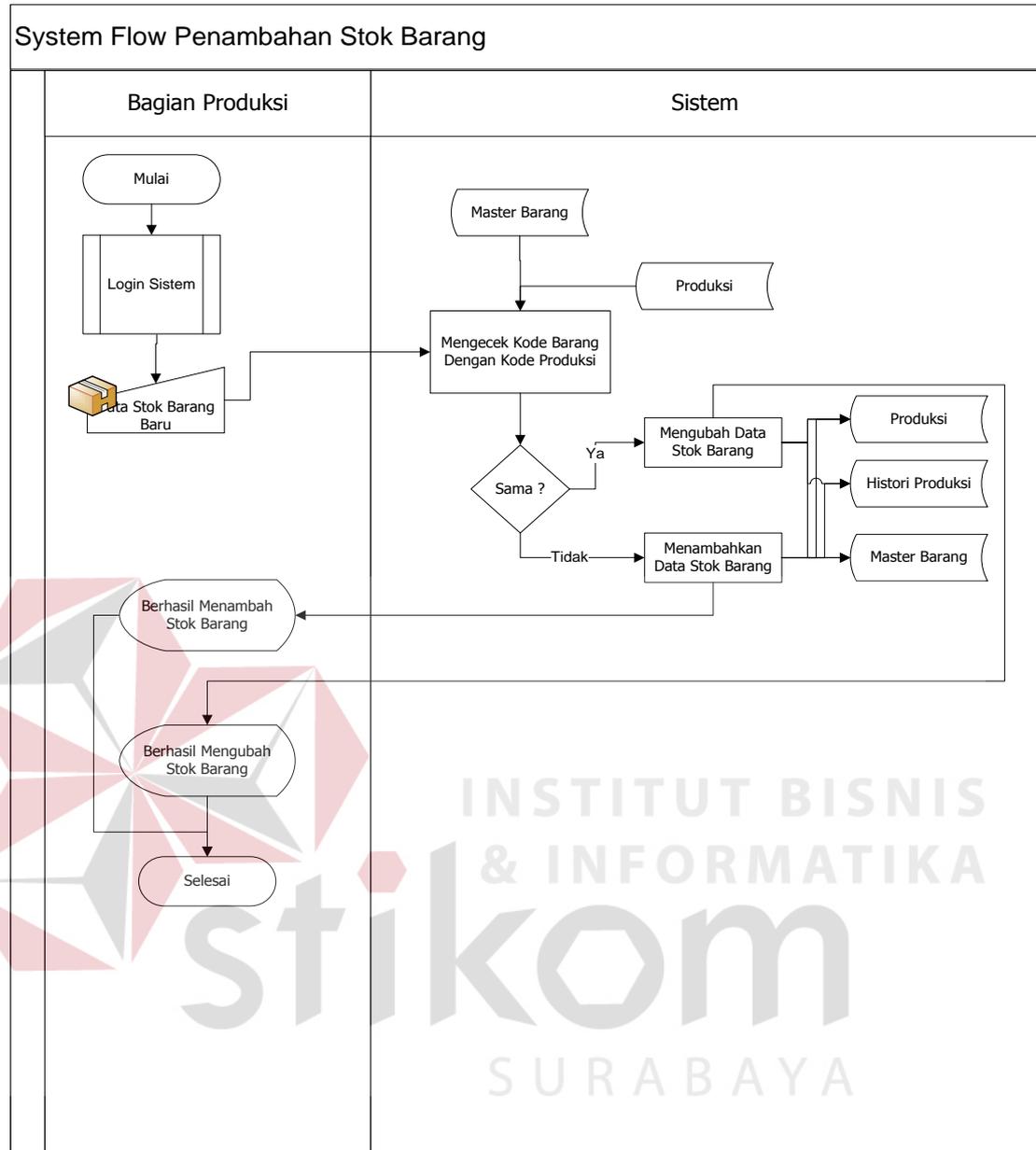
Gambar 3.9 menjelaskan tentang *Maintenance* master barang. Sistem ini dimulai oleh *user* yang sudah mempunyai otoritas dan telah sukses masuk ke dalam sistem. *User* akan memilih menu master barang pada sistem dan sistem

akan menampilkan *form* master barang, *user* dapat memilih menu untuk menginput data baru atau merubah data yang telah ada. Sistem akan melakukan validasi terhadap inputan yang dilakukan oleh *user*, sistem akan memberikan sebuah informasi untuk *user* ketika validasi terdapat kesalahan maupun ketika *user* telah berhasil menginputkan data master barang baru atau merubah data master barang yang telah ada.



Gambar 3.9 System Flow Maintenance Master Barang

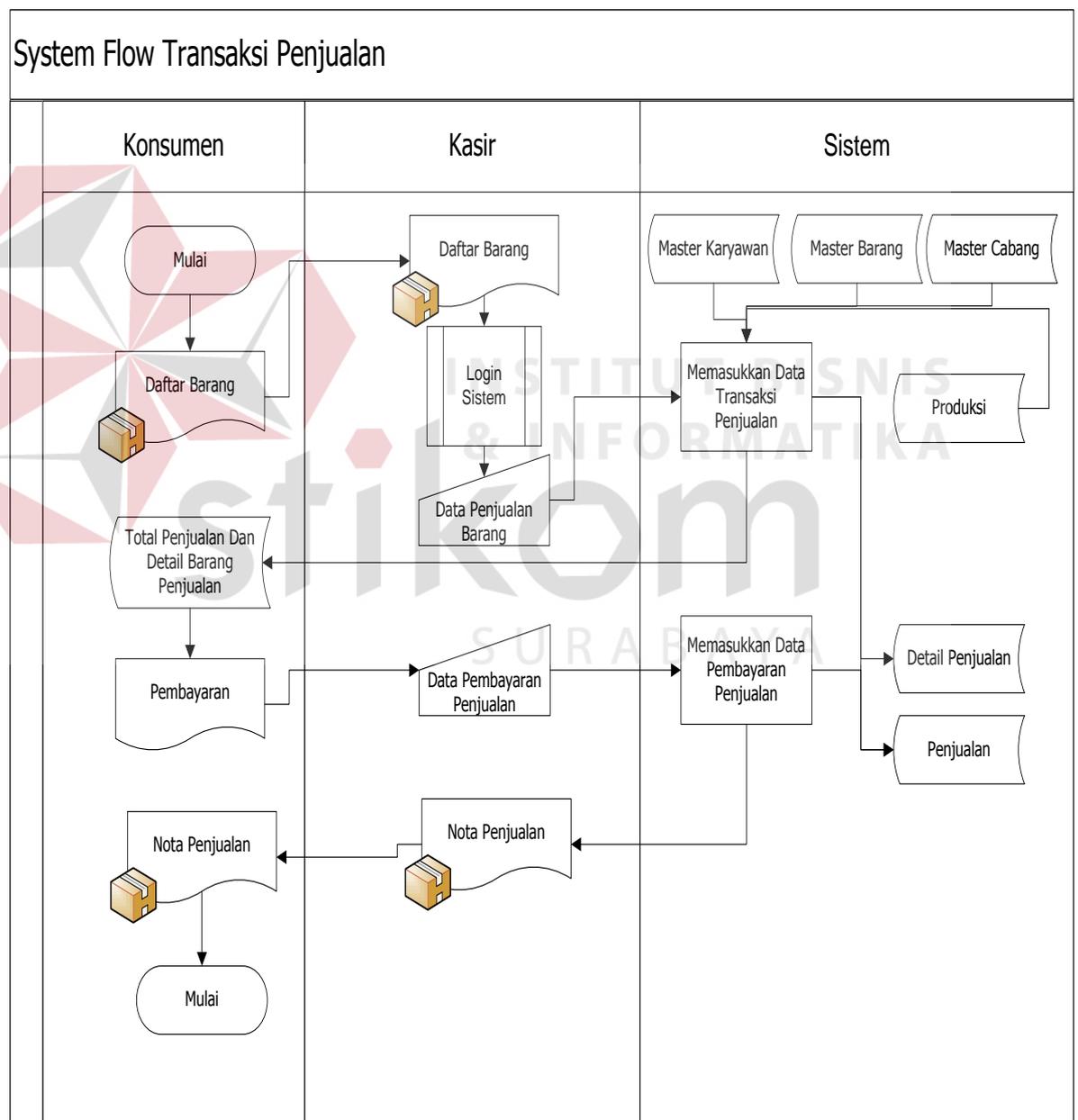
Gambar 3.10 menjelaskan tentang penambahan stok barang. Sistem ini dimulai dari *user* yang telah berhasil *login* dan mempunyai otoritas sebagai bagian produksi, *user* akan menginputkan data stok barang baru yang telah diproduksi. Sistem akan melakukan pengecekan terhadap barang yang diinputkan, jika barang yang diinputkan sudah memiliki kode produksi yang sama dengan data yang telah ada dalam *database*, maka akan ada notifikasi atau *message box* bahwa data stok barang dengan kode barang dan dengan kode produksi yang sama sudah ada pada *database*. *User* akan memilih pilihan tambahkan stok pada pilihan yang ada pada notifikasi tersebut dan stok barang otomatis akan ditambahkan sesuai dengan data yang dimasukkan oleh *user*. Jika barang belum memiliki data kode produksi, barang tersebut akan diberikan kode produksi yang sesuai dengan format tanggal, bulan dan tahun. Sistem akan memberikan suatu informasi bahwa *user* berhasil menambahkan stok barang dengan kode produksi yang sama atau menambahkan barang dengan kode produksi yang baru. Jika terdapat kesalahan pada *input* data penambahan stok barang, *user* bisa melakukan pengurangan stok barang dengan memasukkan kode barang dengan kode barang yang sama dan akan ada notifikasi barang dengan kode produksi yang sama sudah terdapat pada *database*. *User* bisa memilih tombol kurang produksi pada notifikasi tersebut, dan barang dengan kode produksi yang sama akan berkurang sesuai dengan *inputan* data yang dimasukkan oleh *user*.



Gambar 3.10 *System Flow* Penambahan Stok Barang

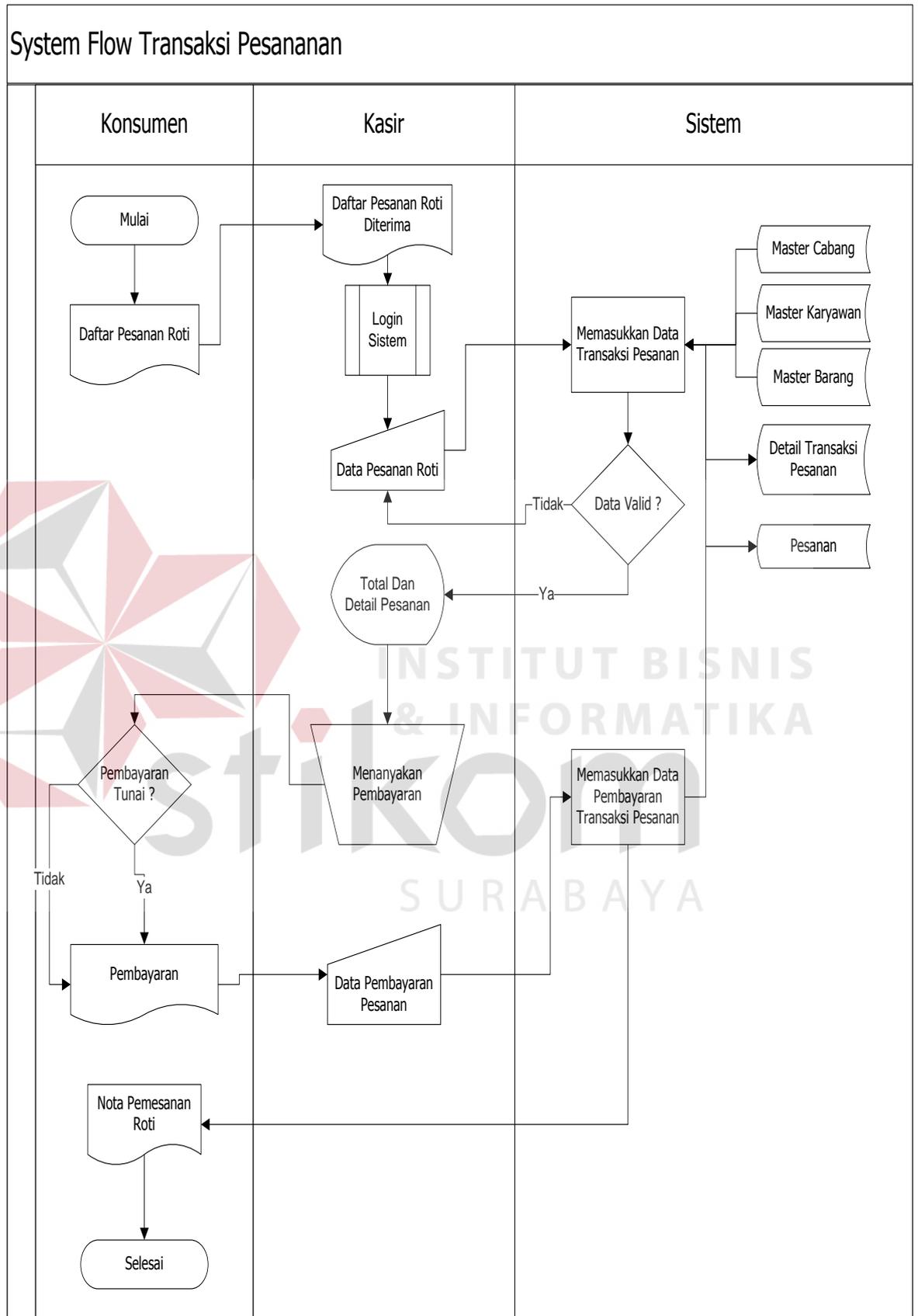
Gambar 3.11 menjelaskan tentang transaksi penjualan. Sistem ini dimulai dari konsumen yang mengambil roti yang terdapat pada rak *display* pada toko. Roti yang telah diambil oleh konsumen kemudian diberikan kepada kasir untuk dilakukan pencatatan transaksi penjualan. Kasir akan menginputkan data transaksi penjualan ke dalam sistem, informasi yang ditampilkan oleh sistem berupa detail

penjualan dan total penjualan kepada konsumen. Konsumen akan memberikan pembayaran sesuai dengan total penjualan yang telah diinformasikan dan kasir akan menginputkan data pembayaran yang telah diberikan oleh konsumen. Sistem akan memberikan nota penjualan berisi tentang detail penjualan dan total penjualan dari penjualan yang dilakukan oleh kasir.



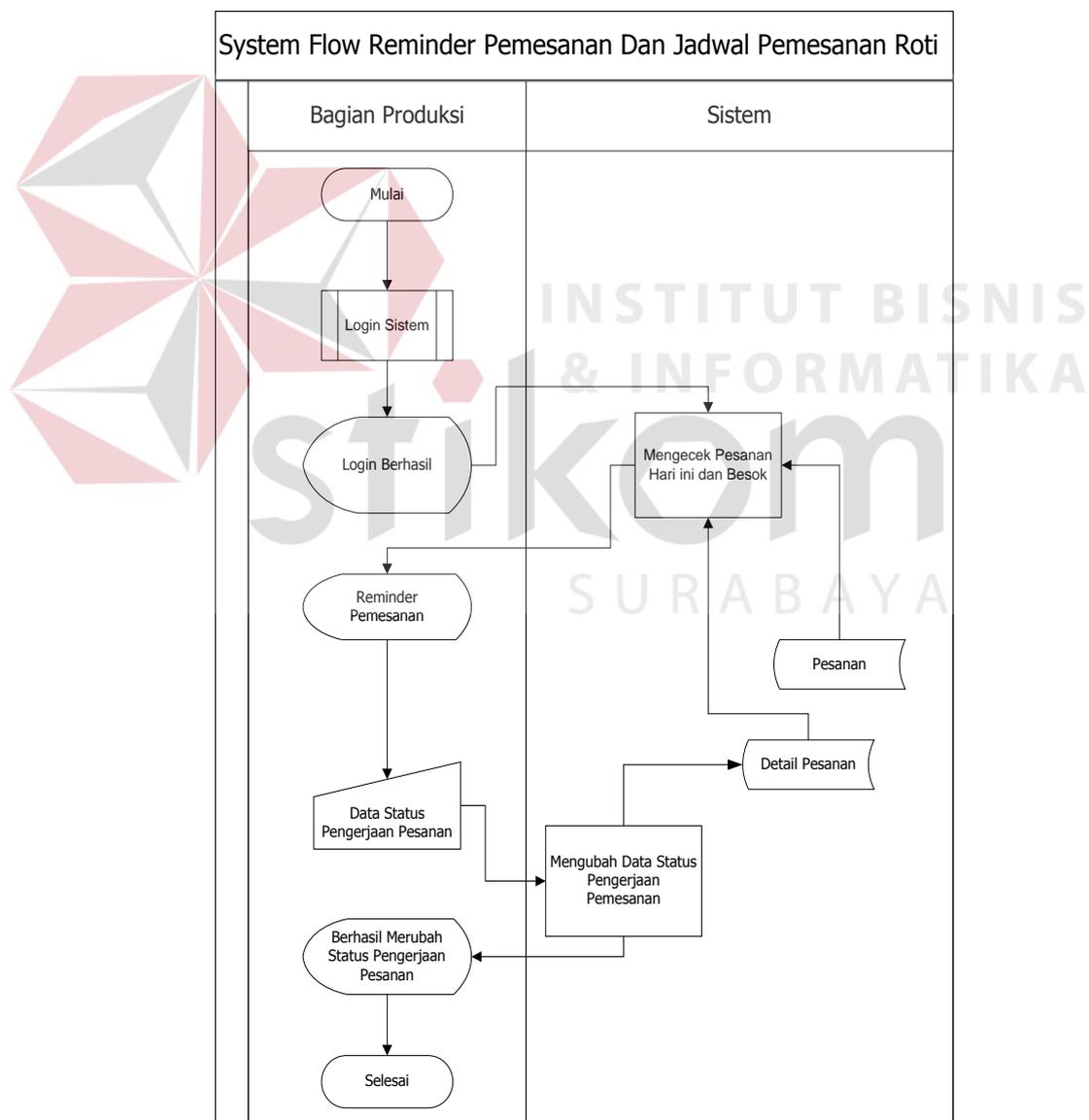
Gambar 3.11 System Flow Transaksi Penjualan

Gambar 3.12 menjelaskan tentang transaksi pemesanan. Sistem ini dimulai dari konsumen yang memberikan daftar pesanan roti yang akan dipesan, kasir akan menerima pesanan dan menginputkan nama konsumen yang melakukan pemesanan dan menginputkan data konsumen ke dalam sistem. Selesai melakukan *input* data konsumen maka kasir akan menginputkan data pesanan ke dalam sistem. Sistem akan memberikan tampilan informasi berupa detail pemesanan barang yang dipesan dan total pemesanan. Kasir juga harus memasukkan data pengambilan pemesanan untuk disimpan ke dalam *database*. Untuk pembayaran, Konsumen dapat memilih pembayaran yang akan dilakukan yaitu dengan cara pembayaran secara penuh ataupun dengan uang muka sebagai tanda jadi. Pembayaran yang dilakukan oleh konsumen akan diterima dan akan diinputkan data pembayaran ke dalam sistem oleh kasir. Jika konsumen melakukan pembayaran secara uang muka, maka sistem akan memberikan nilai pembayaran minimum untuk pemesanan yang telah dilakukan oleh konsumen. Setelah proses pemesanan dilakukan maka sistem akan mencetak nota pemesanan roti yang akan diterima oleh konsumen. Nota pemesanan roti tersebut nantinya akan digunakan oleh konsumen untuk pengambilan pemesanan sesuai dengan tanggal dan waktu pemesanan akan diambil.



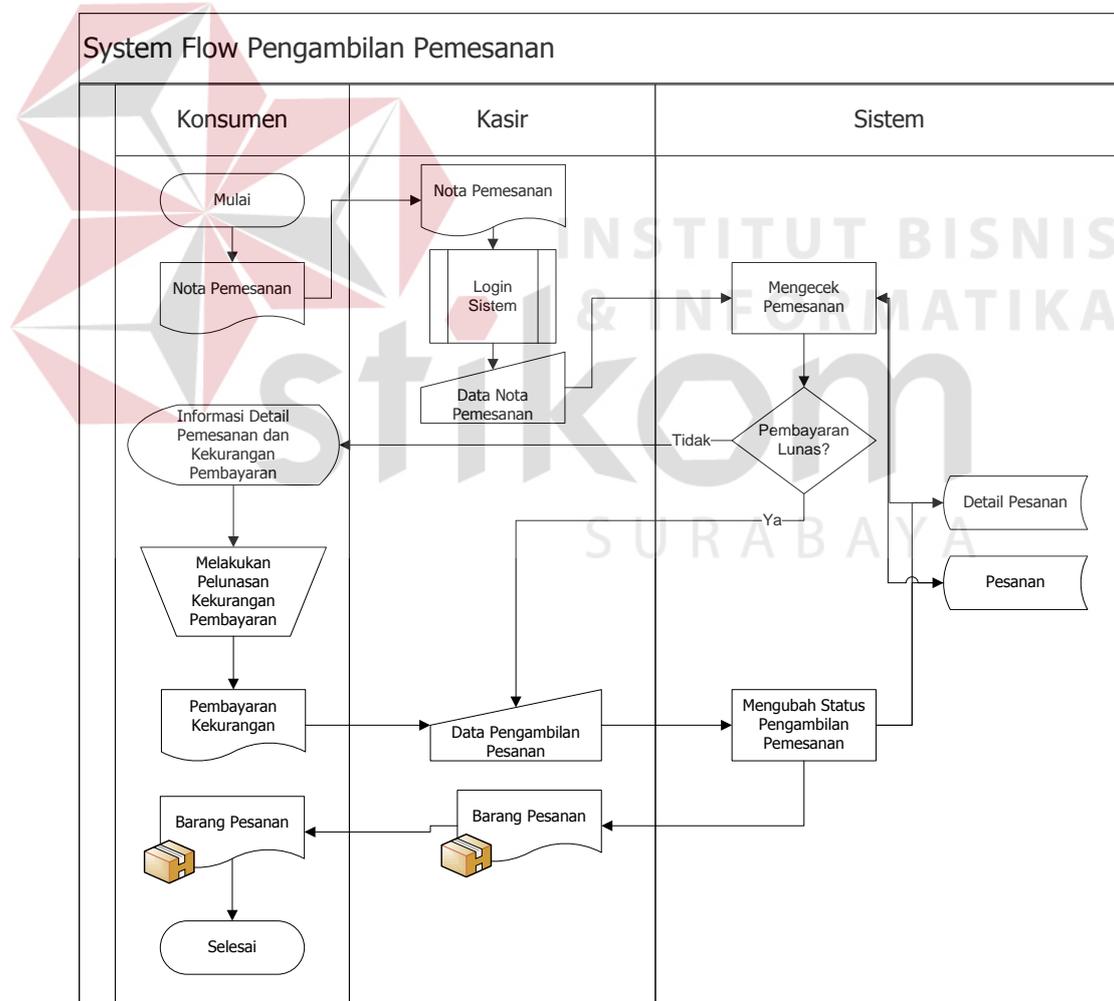
Gambar 3.12 System Flow Transaksi Pemesanan

Gambar 3.13 menjelaskan tentang *reminder* pemesanan yang diterima oleh bagian produksi. Bagian produksi yang telah memiliki otoritas untuk masuk ke dalam sistem dan telah berhasil *login* akan mendapatkan *reminder* pemesanan berupa *notifikasi* pesanan atau *message box* untuk pemesanan yang harus diproduksi untuk hari ini dan besok. Bagian produksi bisa merubah status pengerjaan menjadi “Dalam Pengerjaan” atau “Barang Sudah Jadi”, Status tersebut nantinya akan tersimpan ke dalam *database*.



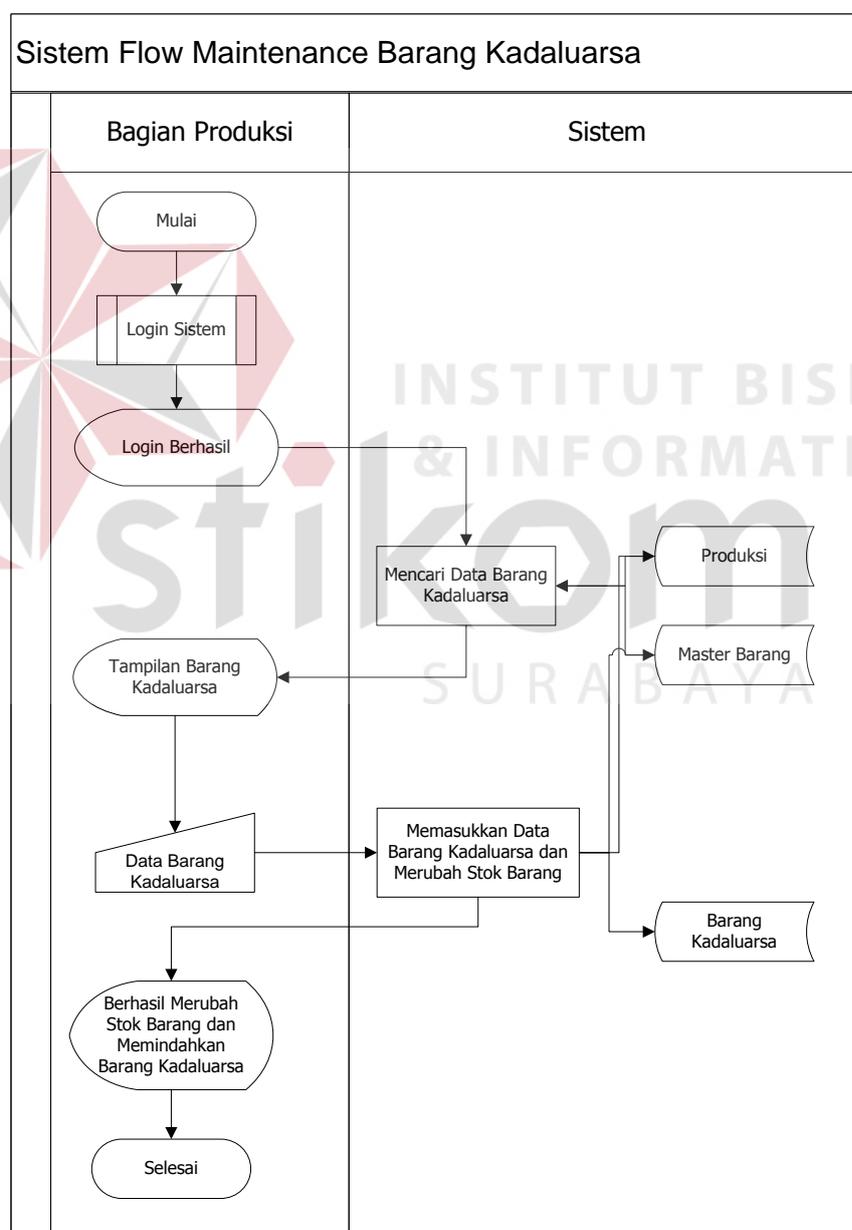
Gambar 3.13 System Flow Reminder Pemesanan dan Jadwal Pemesanan

Gambar 3.14 menjelaskan tentang pengambilan pemesanan yang dilakukan oleh bagian kasir. Bagian kasir akan menerima nota pemesanan yang diberikan oleh konsumen dan memasukkan data nota pesanan ke dalam sistem, sistem akan mengecek pemesanan dan pembayaran yang dilakukan oleh konsumen ketika melakukan pemesanan. Bagian kasir akan menginputkan data pembayaran jika pembayaran yang dilakukan oleh konsumen menggunakan uang muka. Pesanan akan diterima oleh konsumen dan sistem akan merubah status pengambilan pada pesanan yang sesuai dengan data yang dimasukkan oleh kasir.



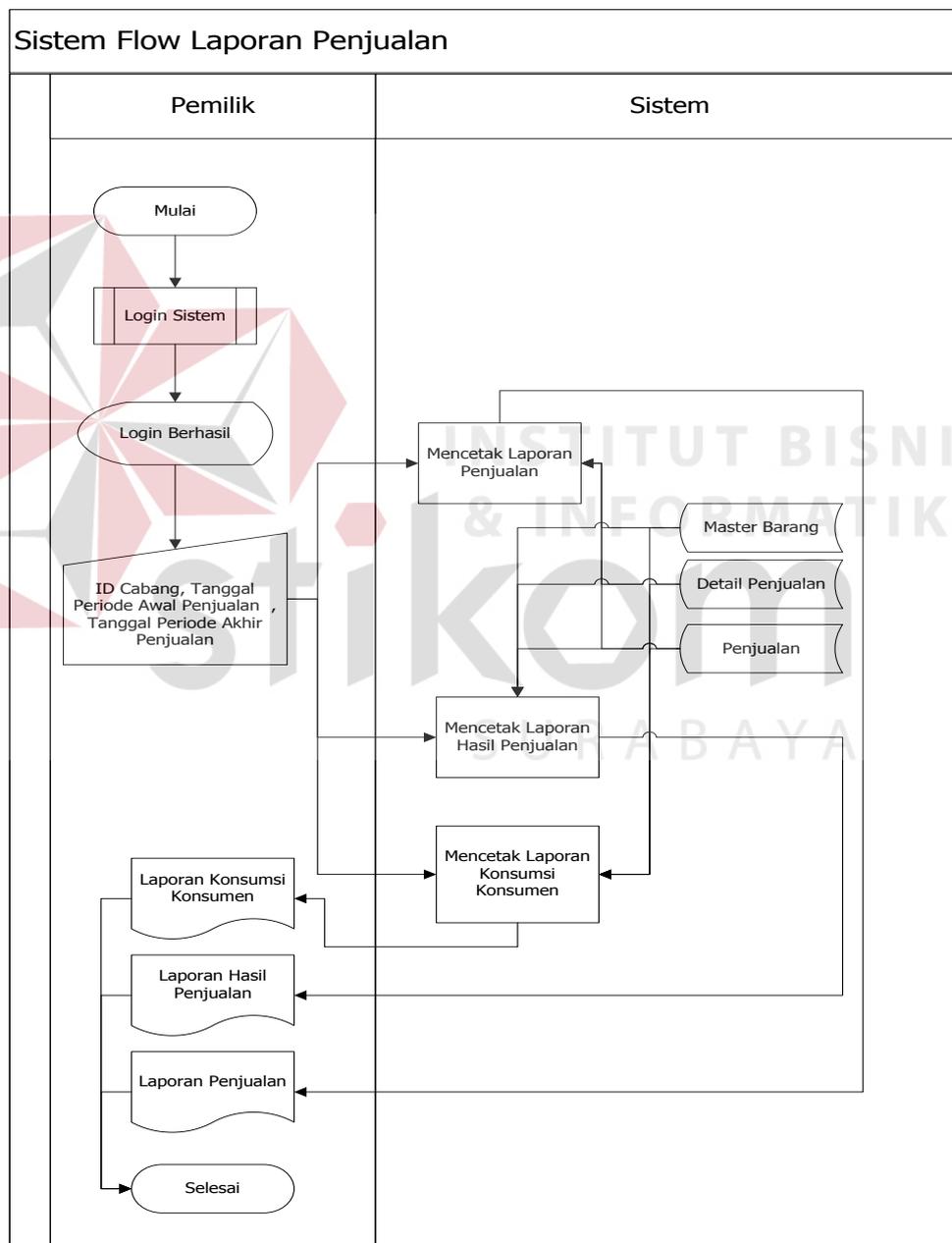
Gambar 3.14 *System Flow* Pengambilan Pemesanan

Gambar 3.15 menjelaskan tentang *maintenance* barang kadaluarsa yang diterima oleh bagian produksi. bagian produksi yang berhasil *login* akan mendapatkan notifikasi dari sistem ketika ada barang yang sudah kadaluarsa yang ada di rak penjualan. Barang yang telah kadaluarsa akan diambil oleh bagian produksi dan otomatis sistem akan mengurangi stok barang dan akan memindahkan data barang kadaluarsa ke dalam *table* barang kadaluarsa.



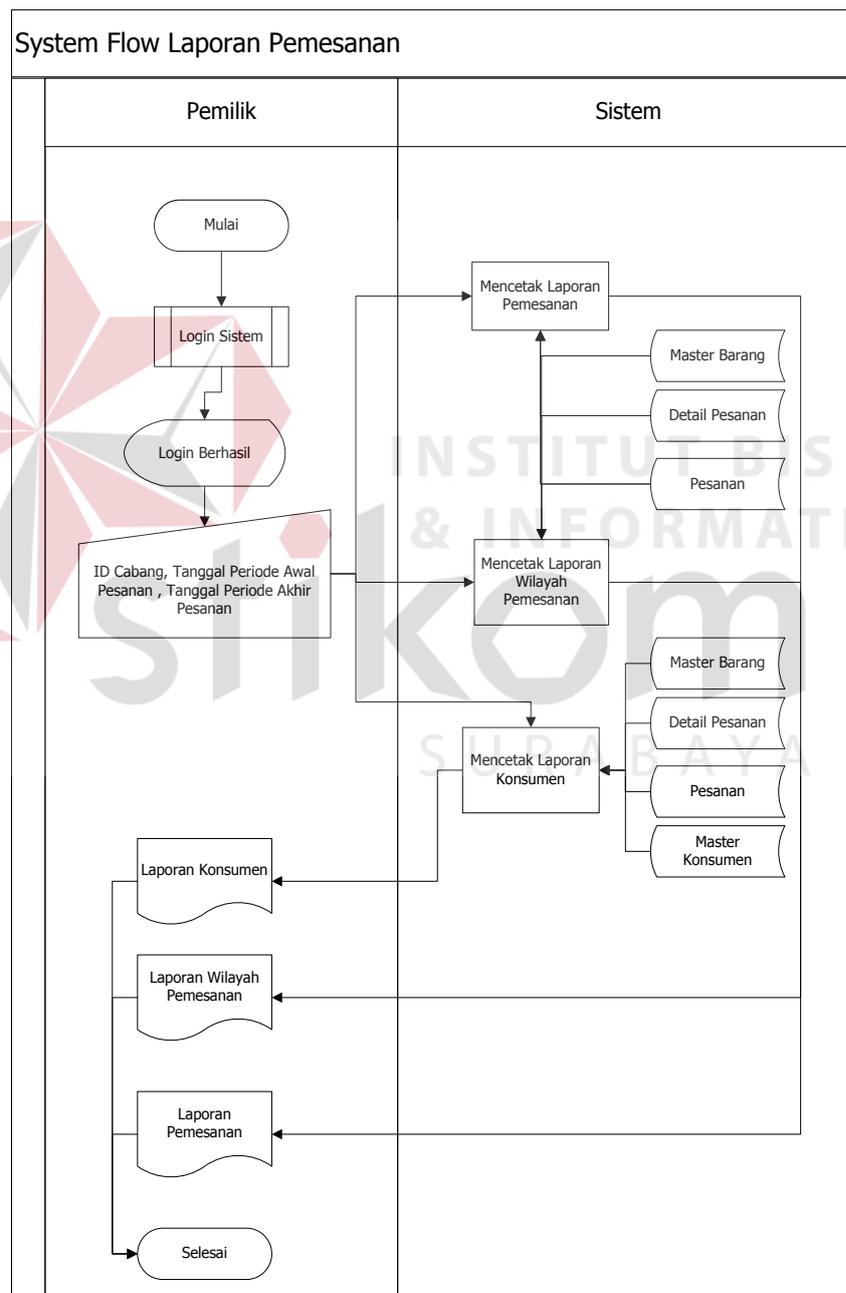
Gambar 3.15 *System Flow Maintenance* Barang Kadaluarsa

Gambar 3.16 menjelaskan tentang laporan penjualan yang dapat diakses oleh *owner* atau pemilik. Pemilik yang telah berhasil *login* memilih menu laporan penjualan yang ada pada sistem, laporan penjualan terdiri dari laporan penjualan, laporan hasil penjualan, dan laporan konsumsi konsumen secara periode yang sesuai dengan inputan yang diberikan oleh pemilik.



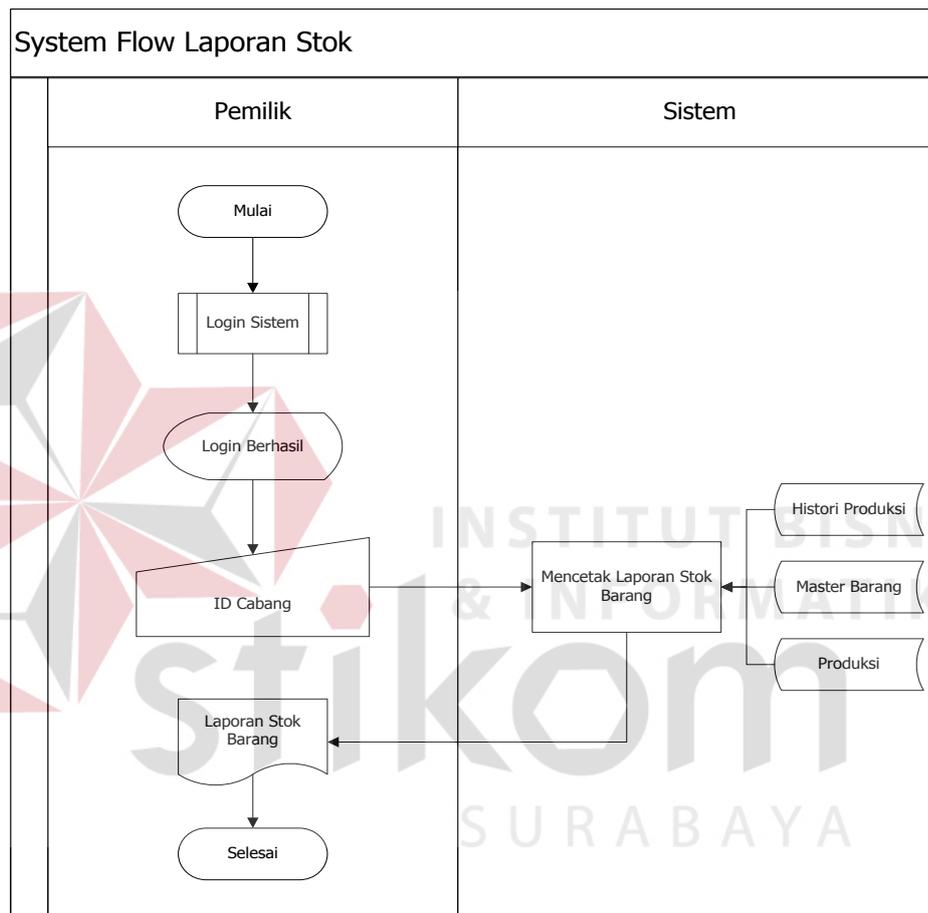
Gambar 3.16 *System Flow* Laporan Penjualan

Gambar 3.17 menjelaskan tentang laporan pemesanan yang dapat diakses oleh *owner* atau pemilik. Pemilik yang telah berhasil *login* memilih menu laporan pemesanan yang ada pada sistem, laporan pemesanan, laporan wilayah pemesanan, dan laporan konsumen secara periode yang sesuai dengan inputan yang diberikan oleh pemilik.



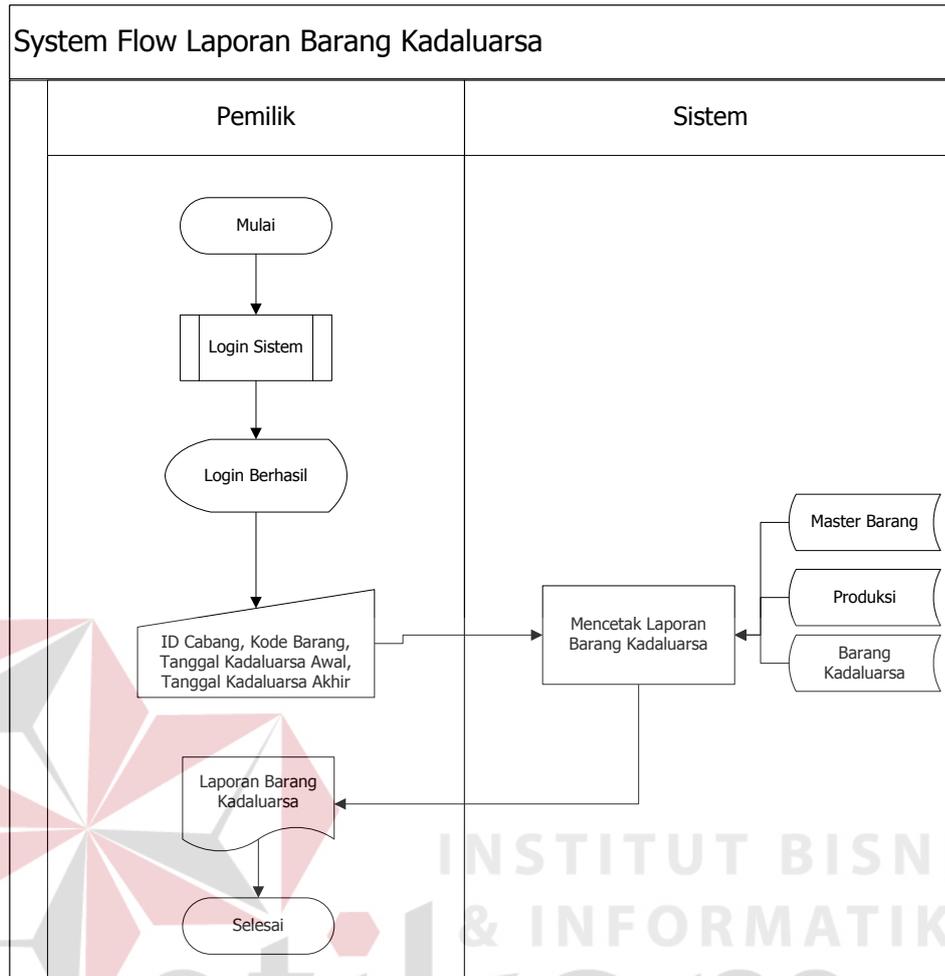
Gambar 3.17 System Flow Laporan Pemesanan

Gambar 3.18 menjelaskan tentang laporan stok barang yang dapat diakses oleh *owner* atau pemilik. Pemilik yang telah berhasil *login* memilih menu laporan stok barang yang ada pada sistem, laporan stok barang ini berfungsi untuk melihat stok barang yang ada pada masing masing cabang pada hari ini.



Gambar 3.18 *System Flow* Laporan Stok

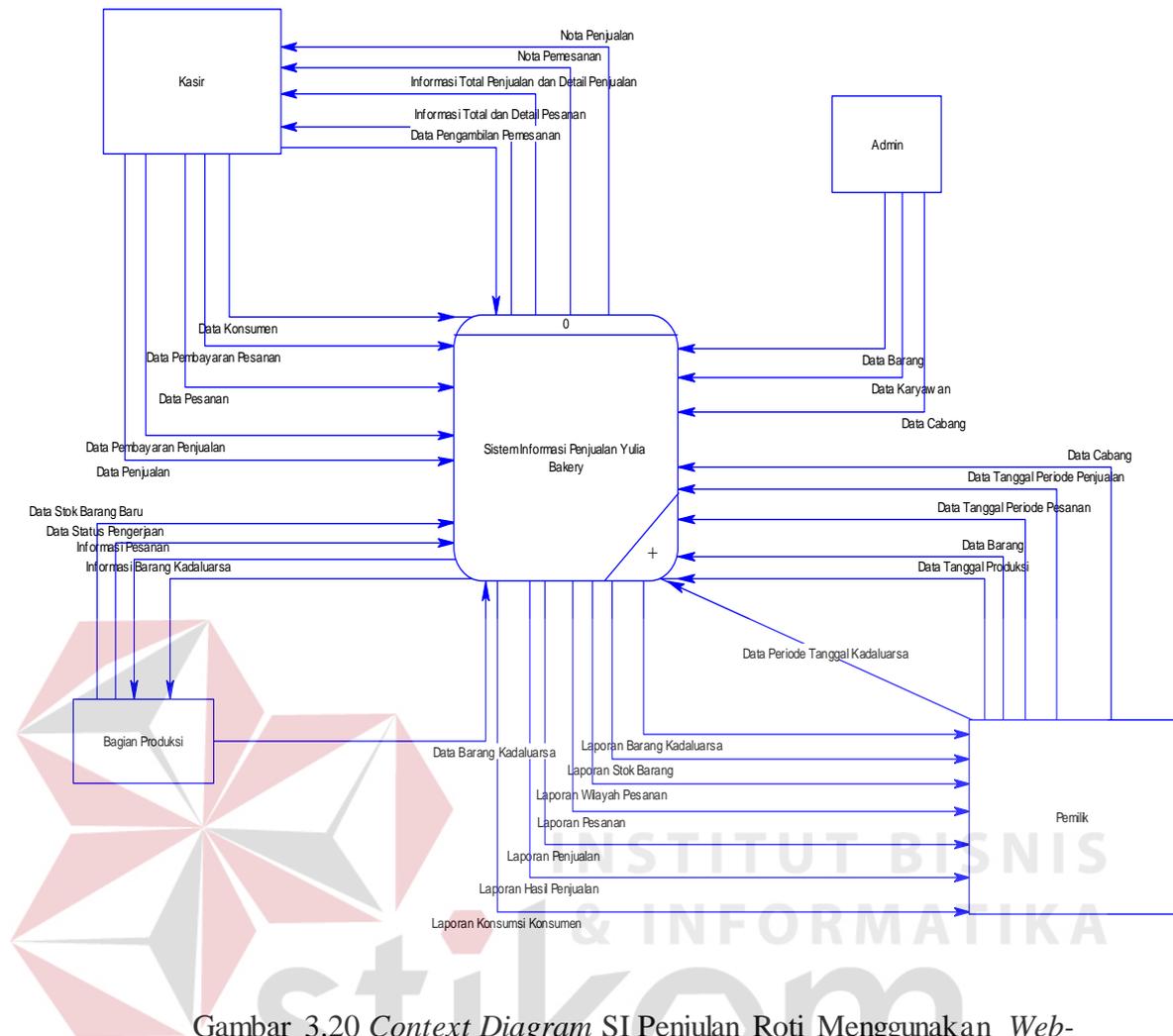
Gambar 3.19 menjelaskan tentang laporan barang kadaluarsa yang dapat diakses oleh pemilik atau *owner*. Pemilik yang telah berhasil *login* memilih menu laporan barang kadaluarsa yang ada pada sistem, laporan ini berfungsi untuk menampilkan barang kadaluarsa per periode sesuai dengan inputan yang dimasukkan oleh pemilik.



Gambar 3.19 System Flow Laporan Barang Kadaluarsa

B. Data Flow Diagram (DFD)

Pada Gambar 3.20 merupakan *data flow diagram* Sistem Informasi Penjualan dengan *Web-Services* pada Yulia Bakery level konteks yang menggambarkan alur data pada sistem secara umum. Pada *context diagram* ini terdapat 4 entitas (bagian), yaitu pemilik, admin, bagian produksi, dan kasir.

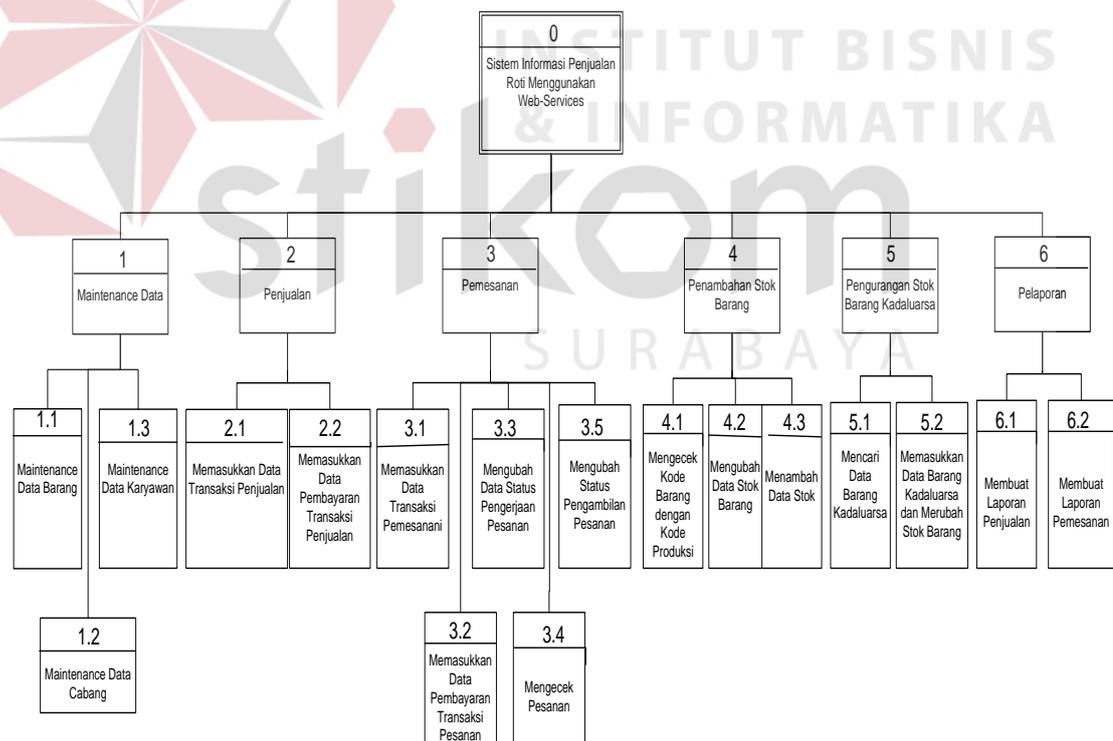


Gambar 3.20 *Context Diagram* SI Penjualan Roti Menggunakan *Web-Services*

B.1 Diagram Berjenjang

Setelah membuat *context diagram*, tahapan selanjutnya adalah membuat diagram berjenjang (HIPO). Diagram berjenjang yang digambarkan berupa hierarki seluruh proses yang ada pada sistem. Dengan adanya diagram berjenjang, alur proses yang ada pada sistem akan menjadi lebih teratur dan jelas. Proses yang ada pada sistem informasi penjualan roti ini terdiri dari 6 (enam) sub proses yang terdiri dari proses *maintenance* data, penjualan, pemesanan, penambahan stok barang, pengurangan stok barang kadaluarsa, dan pelaporan. Proses *maintenance* data dibagi menjadi 3 (tiga) proses yaitu *maintenance* data barang, *maintenance*

data cabang, *maintenance* data karyawan. Untuk proses penjualan dibagi menjadi 2 (dua) proses yaitu memproses permintaan barang, dan mencetak nota penjualan. Proses pemesanan dibagi menjadi 4(empat) proses yaitu memesan roti, menyimpan data, menjadwalkan pesanan, dan memeriksa kekurangan pembayaran. Proses penambahan stok barang terdiri dari 2 (dua) proses yaitu mengecek stok barang, dan menambahkan stok barang. Untuk pengurangan stok kadaluarsa terdiri dari 2 (dua) proses yaitu memeriksa barang kadaluarsa dan mengubah stok barang. Dalam pelaporan terdapat 2 (dua) proses yaitu membuat laporan penjualan dan membuat laporan pemesanan. Gambar 3.21 merupakan diagram berjenjang sistem informasi penjualan roti menggunakan *Web-Services* pada Yulia Bakery.



Gambar 3.21 Diagram Berjenjang Sistem Informasi Penjualan Roti Menggunakan *Web-Service* pada Yulia Bakery

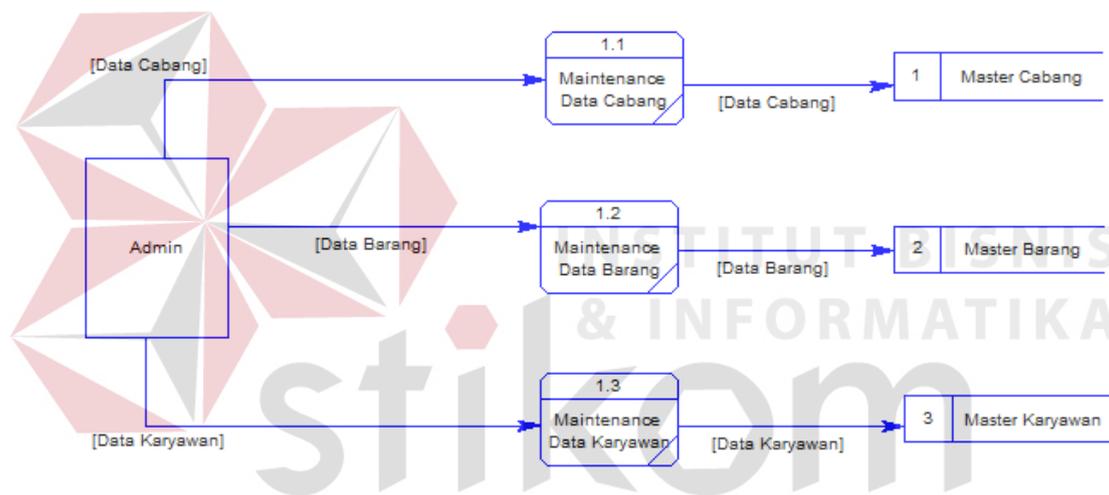
B.2 Data Flow Diagram Level 0

Setelah membuat *context diagram* dan diagram berjenjang, perancangan dilanjutkan dengan membagi *context diagram* menjadi proses-proses yang lebih kecil dan terinci sesuai dengan diagram berjenjang. Gambar 3.22 adalah *data flow diagram* level 0 sistem informasi penjualan roti menggunakan *web-services* pada Yulia Bakery.



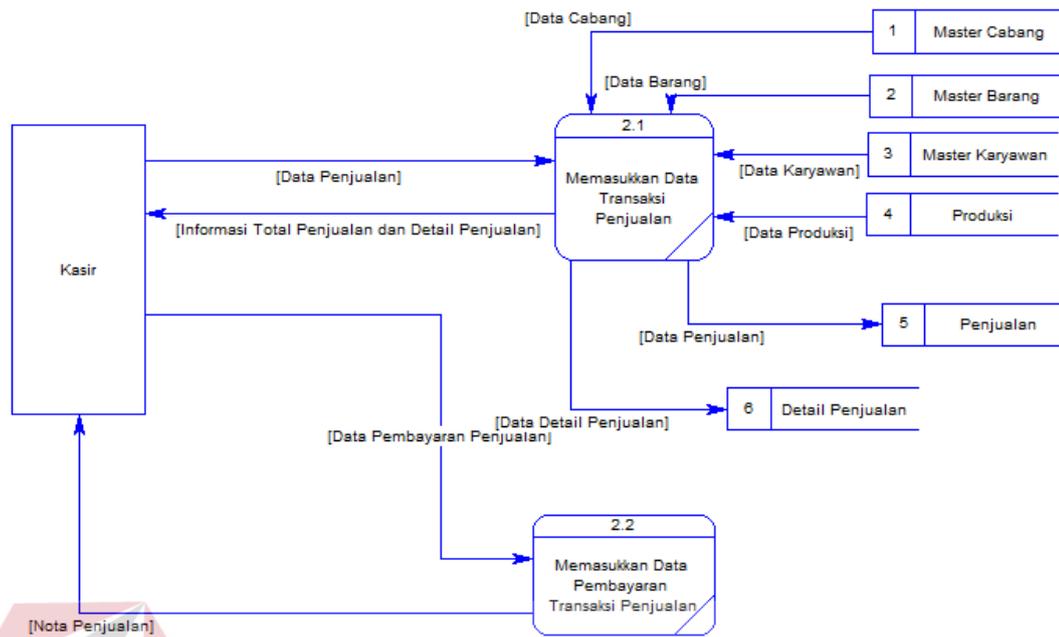
B.3 Data Flow Diagram Level 1

Gambar 3.23 data flow level 1 *maintenance* data adalah Data Flow Diagram level 1.1 proses *maintenance* data yang terdiri dari proses *maintenance* data barang, *maintenance* data cabang, *maintenance* data karyawan. Entitas yang terlibat dalam proses *maintenance* data adalah admin. Tabel yang digunakan pada DFD level 1 *maintenance* data terdapat 3 (tiga) table yang terdiri dari table *master_barang*, *master_cabang*, dan *master_karyawan*.



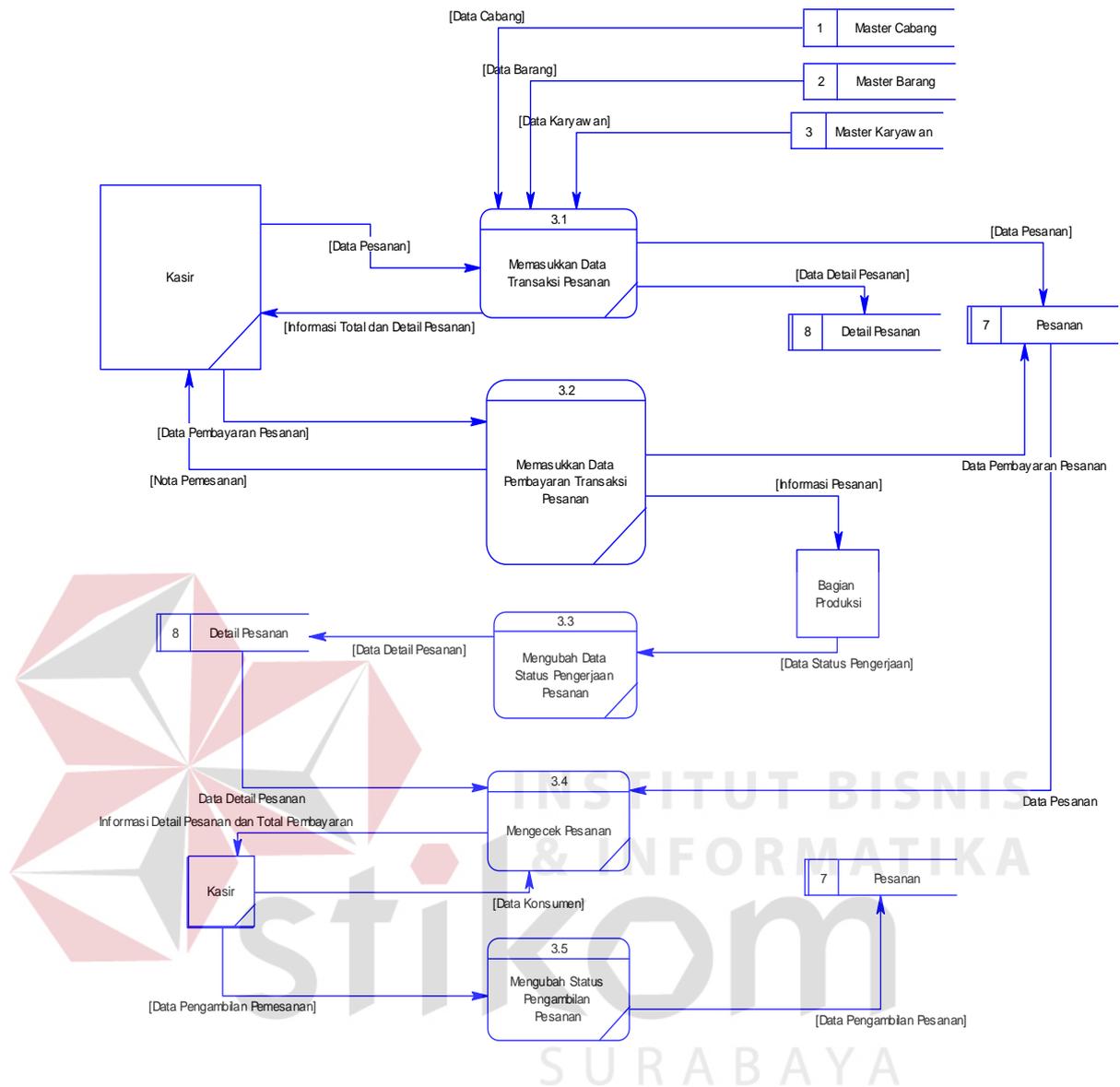
Gambar 3.23 Data flow diagram level 1 subsistem *maintenance* data

Gambar 3.24 menggambarkan subsistem level 1 subsistem penjualan. proses penjualan yang terdiri dari proses memproses permintaan barang dan mencetak nota penjualan. Entitas yang terlibat dalam proses penjualan adalah konsumen. Tabel yang digunakan pada DFD level 1 penjualan ada 6 (enam) tabel yaitu tabel *master_barang*, *produksi*, *master_cabang*, *master_karyawan*, *penjualan*, dan *detail_penjualan*.



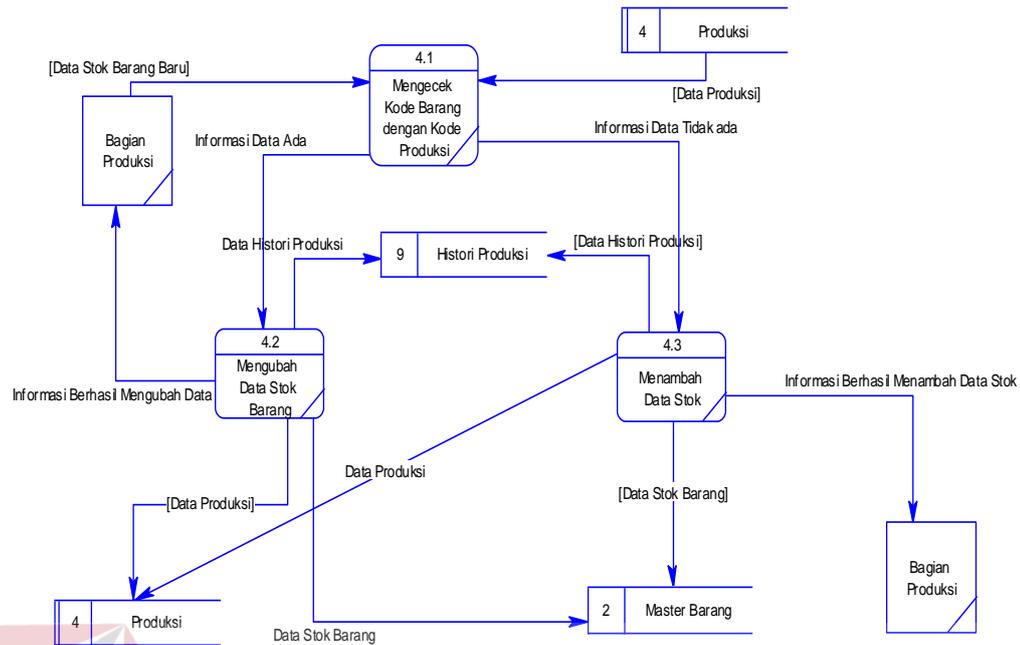
Gambar 3.24 Data flow diagram level 1 subsistem penjualan

Gambar 3.25 menggambarkan subsistem level 1 pemesanan proses pemesanan yang terdiri dari proses memasukkan data transaksi pemesanan, memasukkan data pembayaran transaksi pemesanan, mengubah data status pengerjaan pemesanan, mengecek pesanan, dan mengubah status pengambilan pesanan. Entitas yang terlibat dalam proses pemesanan adalah konsumen dan bagian produksi. Tabel yang digunakan pada DFD level 1 pemesanan ada 5 (lima) tabel yaitu tabel master_barang, master_cabang, pesanan, dan detail pesanan, dan master_karyawan.



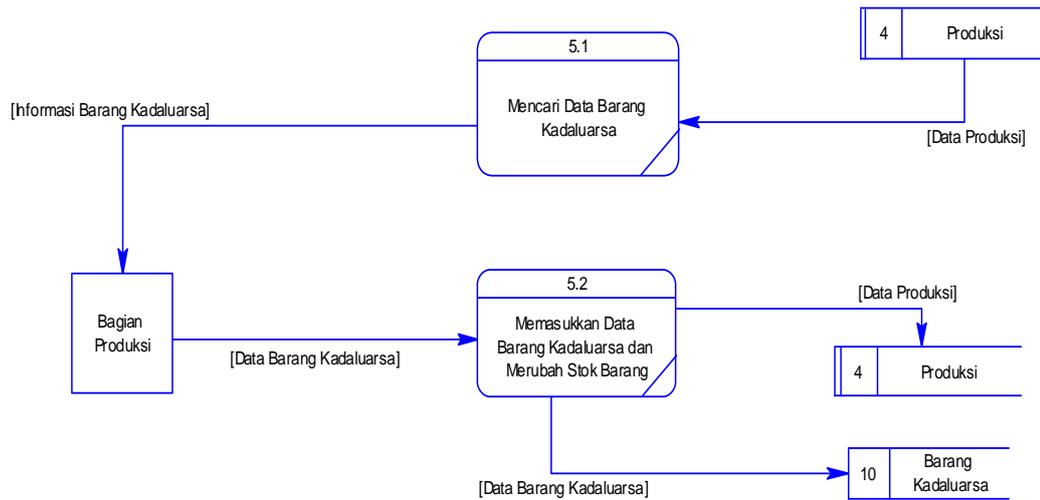
Gambar 3.25 Data flow diagram Level 1 subsistem pemesanan

Gambar 3.26 menggambarkan subsistem level 1 proses penambahan stok barang terdiri dari proses mengecek kode barang dengan kode produksi sama, dan mengubah data stok. Entitas yang terlibat dalam proses penambahan stok barang adalah bagian produksi. Tabel yang digunakan pada DFD level 1 penambahan stok barang ada 3 (tiga) tabel yaitu tabel master_barang, produksi, dan histori produksi.



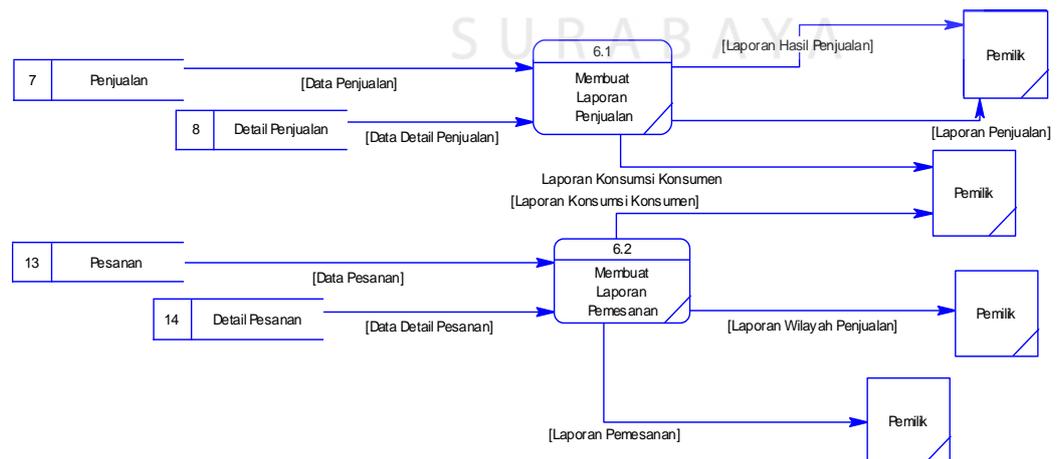
Gambar 3.26 Data flow diagram Level 1 subsistem Penambahan Stok Barang

Gambar 3.27 menggambarkan subsistem level 1 proses pengurangan stok barang kadaluarsa yang terdiri dari proses mencari data barang kadaluarsa, dan memasukkan data barang kadaluarsa dan merubah stok barang. Entitas yang terlibat dalam proses pengurangan stok barang kadaluarsa adalah bagian produksi. Tabel yang digunakan pada DFD level 1 pengurangan stok barang kadaluarsa ada 2 (dua) tabel yaitu tabel produksi, dan barang_kadaluarsa.



Gambar 3.27 Data flow diagram level 1 subsistem pengurangan stok barang kadaluarsa

Gambar 3.28 menggambarkan subsistem proses pelaporan yang terdiri dari membuat laporan penjualan dan membuat laporan pemesanan. Entitas yang terlibat dalam proses pelaporan adalah pemilik. Tabel yang digunakan pada DFD level 1 pelaporan ada 4 (empat) tabel yaitu penjualan, detail_penjualan, pemesanan, dan detail_pemesanan.



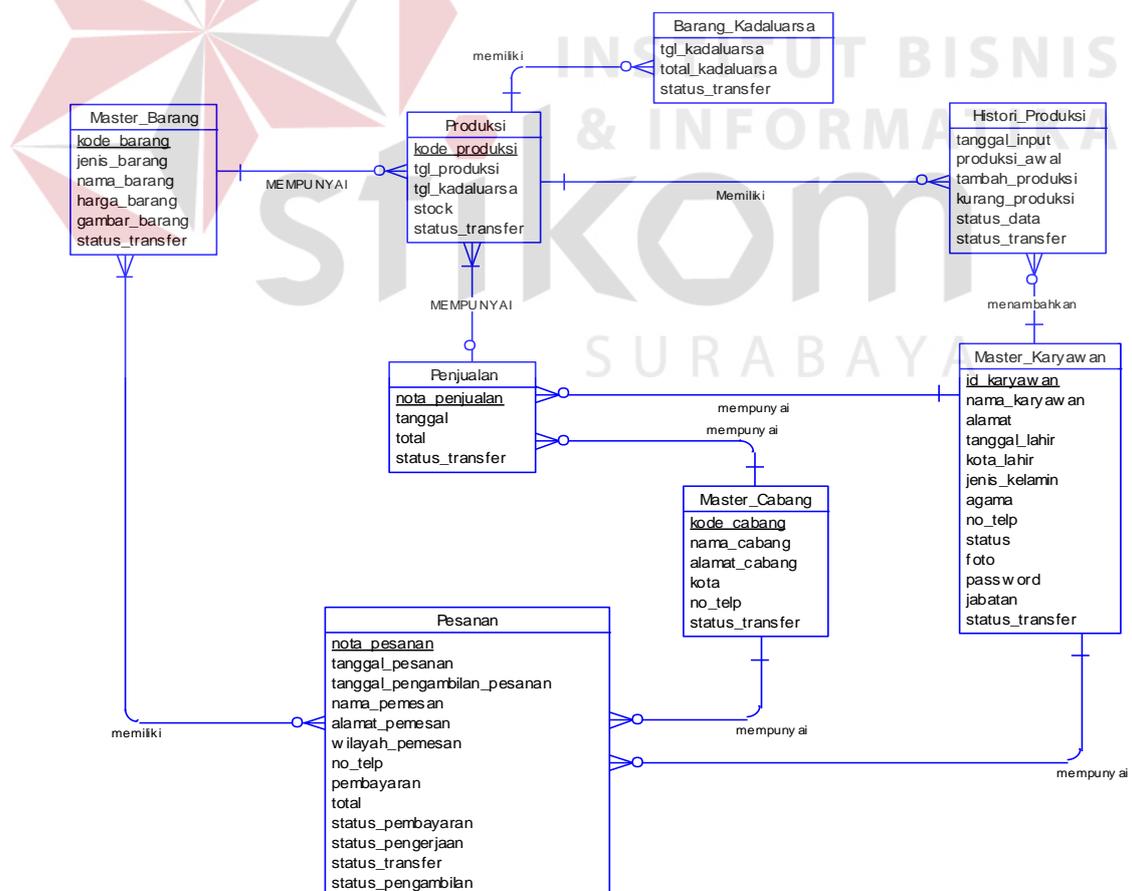
Gambar 3.28 Data flow diagram Level 1 Subsystem Pelaporan

C. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) digunakan untuk mengidentifikasi data yang akan diambil, disimpan, dan diambil kembali (*retrieve*) untuk keperluan-keperluan tertentu dalam mendukung kegiatan dalam sistem. ERD juga digunakan untuk mengidentifikasi asal data yang dibutuhkan dan akan dilaporkan. ERD dibagi menjadi dua, yaitu *Conceptual Data Model (CDM)* atau secara logic dan *Physical Data Model (PDM)* atau secara fisik.

C.1 Conceptual Data Model (CDM)

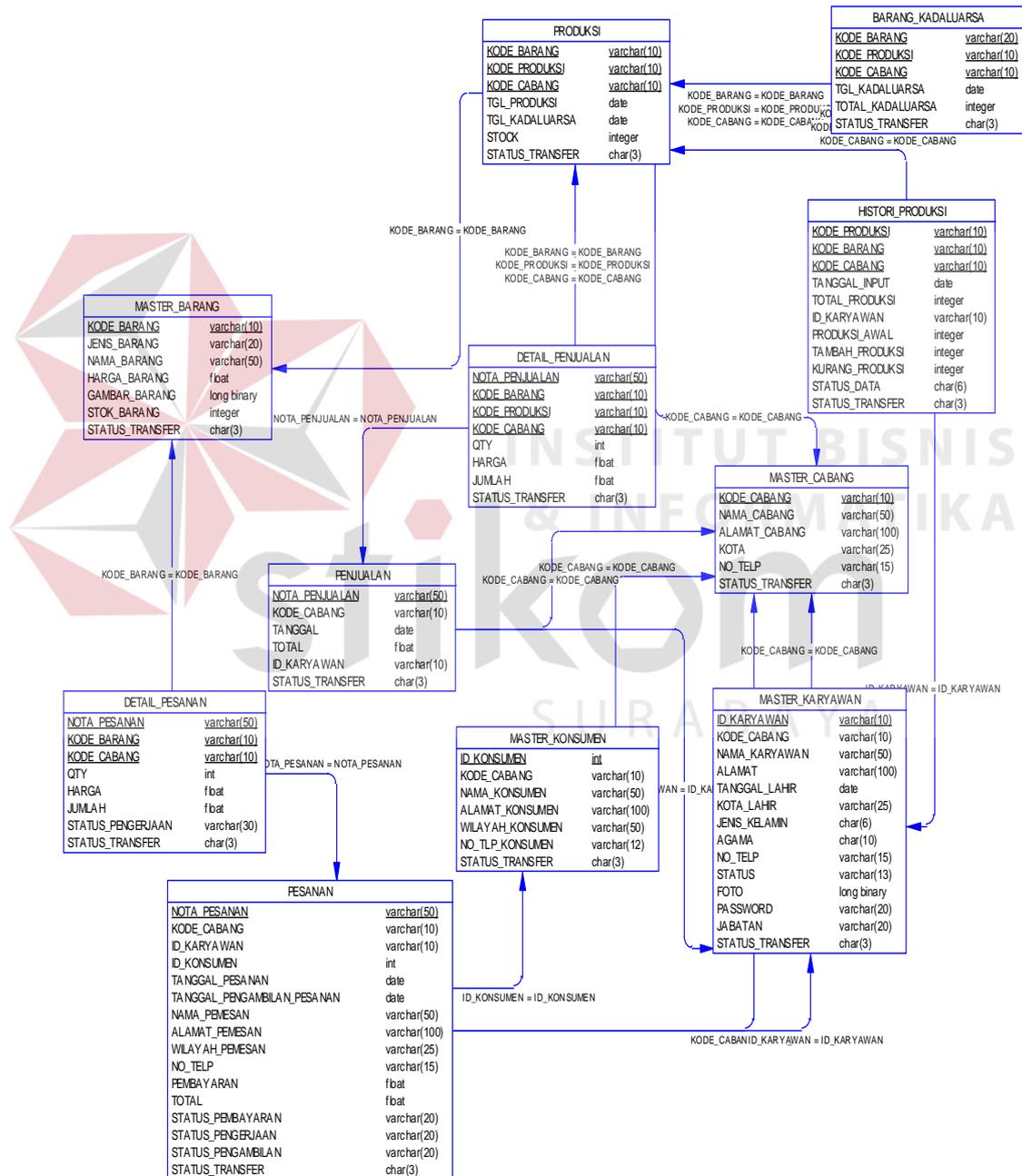
Pada *Conceptual Data Model*, terdapat delapan buah tabel yang saling terintegrasi secara logic.



Gambar 3.29 CDM Sistem Informasi Penjualan Roti Pada Yulia Bakery

C.2 Physical Data Model (PDM)

Pada *Physical Data Model* yang tertera dibawah, telah menunjukkan adanya relasi antar tabel. Terlihat bahwa antar tabel satu dengan yang lain saling memberikan informasi berupa identitas (kode) untuk mengenali tabel yang lain.



Gambar 3.30 PDM Sistem Informasi Penjualan Roti Pada Yulia Bakery

D. Rancangan Database

Tabel-tabel pada Entity Relationship Diagram tersebut memiliki struktur tabel yang saling terintegrasi dan memberikan informasi yang cukup lengkap bagi pengguna sistem. Berikut penjelasan struktur tabel beserta dengan tipe data dan kegunaan dari tiap tabel yang dibutuhkan dalam sistem informasi penjualan roti menggunakan *web-services* pada Yulia Bakery.

1. Tabel Master_Karyawan

Primary Key : id_karyawan

Foreign Key : kode_cabang

Fungsi : Menyimpan data karyawan

Tabel 3.7 Tabel Master_Karyawan

NO	Field Name	Type Data	Keterangan
1	Id_Karyawan	Varchar(10)	Primary key
2	Kode_Cabang	Varchar(10)	Foreign key
3	Nama_karyawan	Varchar(50)	
4	Alamat	Varchar(100)	
5	Tanggal_lahir	Date	
6	Kota_lahir	Varchar(25)	
7	Jenis_kelamin	Char(6)	
8	Agama	Char(10)	
9	No_telp	Varchar(15)	
10	Status	Varchar(13)	
11	Foto	Long Binary	
12	Password	Varchar(20)	
13	Jabatan	Varchar(20)	
14	Status_Transfer	Char(3)	

2. Tabel Master_Cabang

Primary Key : kode_cabang

Foreign Key :-

Fungsi : Menyimpan data cabang Yulia Bakery

Tabel 3.8 Tabel Master_Cabang

NO	Field Name	Type Data	Keterangan
1	Kode_cabang	Varchar(10)	Primary key
2	Nama_cabang	Varchar(50)	
3	Alamat_cabang	Varchar(100)	
4	Kota	Varchar(25)	
5	No_telp	Varchar(15)	
6	Status_Transfer	Char(3)	

3. Tabel Master_Barang

Primary Key : kode_barang

Foreign Key : -

Fungsi : Menyimpan data barang

Tabel 3.9 Tabel Master_Barang

NO	Field Name	Type Data	Keterangan
1	Kode_barang	Varchar(10)	Primary key
2	Jenis_barang	Varchar(20)	
3	Nama_barang	Varchar(50)	
4	Harga_barang	Float	
5	Gambar_barang	Long Binary	
6	Stok_barang	int	
7	Status_Transfer	Char(3)	

4. Tabel Produksi

Primary Key : kode_barang,kode_produksi,kode_cabang

Foreign Key : -

Fungsi : Menyimpan data barang dengan kode_produksi berbeda

Tabel 3.10 Tabel Produksi

NO	Field Name	Type Data	Keterangan
1	Kode_barang	Varchar(10)	Primary Key
2	Kode_produksi	Varchar(10)	Primary Key

NO	Field Name	Type Data	Keterangan
3	Kode_cabang	Varchar(10)	Primary Key
4	Tgl_produksi	Date	
5	Tgl_kadaluarsa	Date	
6	Stok	Int	
7	Status_Transfer	Char(3)	

5. Tabel Barang Kadaluarsa

Primary Key : kode_barang,kode_produksi,kode_cabang

Foreign Key : kode_barang,kode_produksi,kode_cabang

Fungsi : Menyimpan data yang telah kadaluarsa

Tabel 3.11 Tabel Barang Kadaluarsa

NO	Field Name	Type Data	Keterangan
1	Kode_barang	Varchar(10)	PK,FK
2	Kode_produksi	Varchar(10)	PK,FK
3	Kode_cabang	Varchar(10)	PK,FK
3	Tgl_kadaluarsa	Date	
4	Total_kadaluarsa	Int	
5	Status_Transfer	Char(3)	

6. Tabel Penjualan

Primary Key : nota_penjualan

Foreign Key : id_karyawan,kode_cabang

Fungsi : Menyimpan data yang transaksi penjualan

Tabel 3.12 Tabel Penjualan

NO	Field Name	Type Data	Keterangan
1	Nota_Penjualan	Varchar(50)	Primary Key
2	Kode_cabang	Varchar(10)	Foreign Key
3	Id_karyawan	Varchar(10)	Foreign Key
4	Total	Float	
5	Tanggal	Date	
6	Status_Transfer	Char(3)	

7. Tabel Detail Penjualan

Primary Key : nota_penjualan,kode_barang,kode_produksi,kode_cabang

Foreign Key : nota_penjualan,kode_barang,kode_produksi,kode_cabang

Fungsi : Menyimpan data detail penjualan dari setiap penjualan

Tabel 3.13 Tabel Detail Penjualan

NO	Field Name	Type Data	Keterangan
1	Nota_penjualan	varchar(50)	PK,FK
2	Kode_barang	Varchar(10)	PK,FK
3	Kode_produksi	Varchar(10)	PK,FK
4	Kode_cabang	Varchar(10)	PK,FK
5	Qty	Int	
6	Harga	Float	
7	Jumlah	Float	
8	Status_Transfer	Char(3)	

8. Tabel Pesanan

Primary Key : nota_pesanan,kode_pesanan,kode_cabang

Foreign Key : id_karyawan

Fungsi : Menyimpan data transaksi pesanan

Tabel 3.14 Tabel Pesanan

NO	Field Name	Type Data	Keterangan
1	Nota_pesanan	Varchar(50)	Primary Key
2	Kode_cabang	Varchar(10)	Foreign Key
3	Id_karyawan	Varchar(10)	Foreign Key
4	Id_konsumen	Int	Foreign Key
5	Tanggal_pesanan	Date	
6	Tanggal_Pengambilan_Pesanan	DateTime	
7	Nama_pemesan	Varchar(50)	
8	Alamat_pemesan	Varchar(100)	
9	Wilayah_pemesan	Varchar(50)	
10	No_telp	Varchar(15)	
11	Pembayaran	Float	
12	Total	Float	
13	Status_Pembayaran	Varchar(20)	
14	Status_Pengerjaan	Varchar(20)	

NO	Field Name	Type Data	Keterangan
15	Status_pengambilan	Varchar(20)	
16	Status_Transfer	Char(3)	

9. Tabel Detail_Pesanan

Primary Key : nota_pesanan,kode_barang,kode_cabang

Foreign Key : nota_pesanan,kode_barang,kode_cabang,Id_Karyawan

Fungsi : Menyimpan data detail transaksi pesanan

Tabel 3.15 Tabel Detail_Pesanan

NO	Field Name	Type Data	Keterangan
1	Nota_pesanan	Varchar(50)	PK,FK
2	Kode_barang	Varchar(10)	PK,FK
3	Kode_cabang	Varchar(10)	PK,FK
4	Qty	Int	
5	Harga	Float	
6	Jumlah	Float	
7	Status_Pengerjaan	Varchar(30)	
8	Status_Transfer	Char(3)	

10. Tabel Histori Produksi

Primary Key : kode_barang,kode produksi,kode_cabang

Foreign Key : kode_barang,kode produksi,kode_cabang,id_karyawan

Fungsi : Menyimpan data Histori Produksi

Tabel 3.16 Tabel Histori Produksi

NO	Field Name	Type Data	Keterangan
1	Kode_barang	Varchar(10)	PK,FK
2	Kode_produksi	Varchar(10)	PK,FK
3	Kode_cabang	Varchar(10)	PK,FK
4	Id_karyawan	Varchar(10)	FK
5	Tanggal_input	Datetime	
6	Produksi_awal	Int	
7	Tambah_produksi	Int	

NO	Field Name	Type Data	Keterangan
8	Kurang_produksi	Int	
9	Total_produksi	Int	
10	Status_data	Char(6)	
11	Status_Transfer	Char(3)	

11. Tabel Master_Konsumen

Primary Key : id_konsumen, kode_cabang

Foreign Key : kode_cabang

Fungsi : Menyimpan data Histori Produksi

Tabel 3.17 Tabel Histori Produksi

NO	Field Name	Type Data	Keterangan
1	ID_Konsumen	Varchar(10)	PK,FK
2	Kode_cabang	Varchar(10)	PK,FK
3	Nama_Konsumen	Varchar(50)	
4	Alamat_Konsumen	Varchar(100)	
5	Wilayah_Konsumen	Varchar(10)	
6	No_Tlp	Varchar(25)	
7	Status_Transfer	Char(3)	

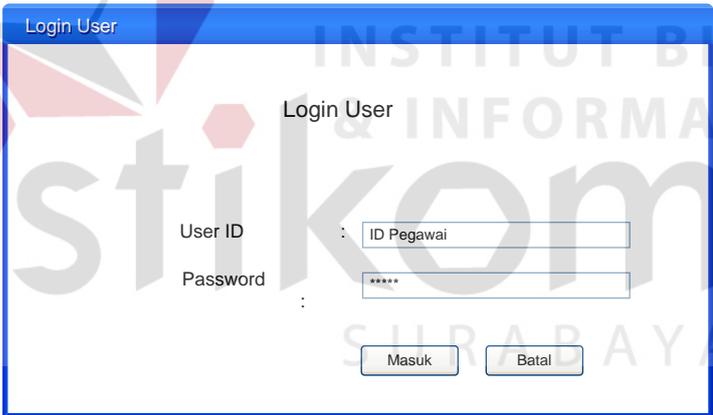
E. Rancangan Interface

Setelah melakukan perancangan sistem dan perancangan basis data, tahap selanjutnya adalah membuat desain *input/output*. Desain *input/output* menggambarkan interaksi antara pengguna dengan sistem yang akan dibuat. Desain *input/output* dibuat sebelum membuat *interface* yang sesungguhnya. desain ini dapat digunakan sebagai bahan dasar perancangan *interface* dari program yang sesuai dengan kebutuhan *user*. Apabila desain sudah cukup *user friendly* dengan *user* maka akan mudah untuk membuat *interface* program yang akan dibuat. Secara keseluruhan Sistem Informasi Penjualan Roti menggunakan *Web-Service* pada Yulia Bakery terdiri dari : Desain *Form Login*, Desain *Form*

Utama, Desain *Form* Master Cabang, Desain *Form* Master Barang, Desain *Form* Penjualan, Desain *Form* Pemesanan, Desain Pembayaran Penjualan, Desain Pengambilan Pemesanan, Desain Laporan Penjualan, Desain Laporan Hasil Penjualan, Desain Laporan Pemesanan, Desain Laporan Konsumsi Konsumen.

E.1 Desain *Form* Login

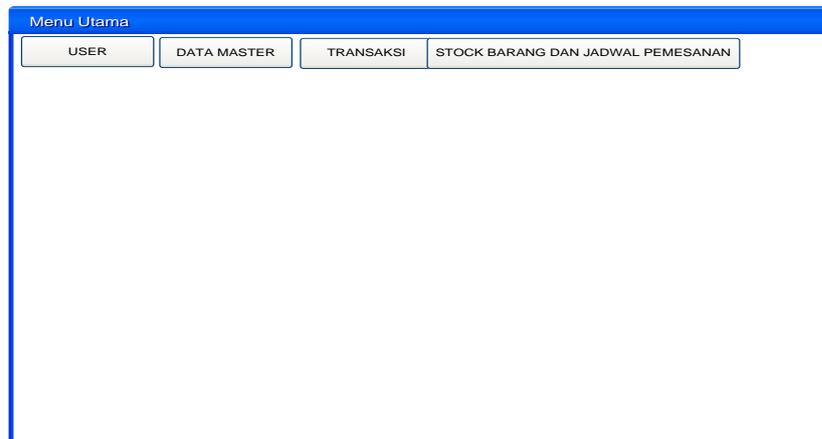
Pada *Form* login ini digunakan untuk melakukan validasi terhadap pengguna yang akan menggunakan aplikasi yang dibuat. Penggunaanya akan dibagi menjadi tiga pengguna , yaitu : kasir, Admin, dan Bagian Produksi. setiap pengguna nantinya akan mendapatkan menu yang berbeda sesuai dengan *otoritas* penggunaanya. Gambar 3.31 adalah desain halaman *login*.

The image shows a screenshot of a web-based login form. The form has a blue title bar at the top that says "Login User". Below the title bar, the text "Login User" is displayed. There are two input fields: one for "User ID" with the placeholder text "ID Pegawai" and another for "Password" with placeholder text "*****". Below the input fields, there are two buttons: "Masuk" (Login) and "Batal" (Cancel). The background of the screenshot is a watermark featuring a large red and white geometric logo on the left and the text "INSTITUT BISNIS & INFORMATIKA stikom SURABAYA" in a light gray font.

Gambar 3.31 Desain Halaman *Login*

E.2 Desain *Form* Utama

Setelah proses *login* berhasil, aplikasi akan menampilkan tampilan utama. Di tampilan utama ini terdapat beberapa navigasi menu. Setiap *user* yang *login* tampilan menunya tidak akan sama, untuk kasir hanya bisa memilih menu transaksi, dan untuk admin hanya bisa melihat menu data master, sedangkan untuk bagian produksi hanya bisa memilih menu stok barang dan jadwal pemesanan.



Gambar 3.32 Desain *Form* Menu Utama

E.3 Desain *Form* Master Cabang

Dalam desain *form* master cabang pada Gambar 3.33 digunakan untuk melakukan *maintenance* data master cabang. Disini terdapat nama cabang, alamat, kota, dan nomor telepon di cabang Yulia Bakery. untuk button yang tersedia ada simpan, ubah, dan batal, untuk simpan akan muncul jika di dalam *database* cabang belum terdapat datanya. Jika sudah terdapat datanya maka yang muncul hanya *button* ubah.

 A screenshot of a "Form Master Cabang" window. The title bar says "Form Master Cabang". The main content area has the title "Form Master Cabang" and five input fields: "Kode Cabang", "Nama Cabang", "Alamat", "Kota", and "No. Telp/Handphone". Each field is preceded by a colon. At the bottom right, there are two buttons: "Ubah" and "Batal".

Gambar 3.33 Desain *Form* Master Cabang

E.4 Desain *Form Master Barang*

Form pada Gambar 3.34 berfungsi untuk menyimpan data barang untuk dijadikan *list* dalam penjualan. Tombol simpan untuk menyimpan, sedangkan tombol ubah berguna untuk merubah nama, harga dan gambar barang. Untuk tombol hapus digunakan untuk menghapus data barang yang sudah ada, dan tombol *clear* digunakan untuk membersihkan semua *textbox*, *combobox*, dan *image*. Untuk *tabcontrol* cari data barang digunakan untuk melihat semua *list* barang yang ada di *database*.

Gambar 3.34 Desain Form Master Barang

E.5 Desain *Form Master Karyawan*

Form pada Gambar 3.35 digunakan oleh admin yang berfungsi untuk menyimpan, mengubah atau menghapus data karyawan dari *master_karyawan*.

Gambar 3.35 Desain *Form* Master Karyawan

E.6 Desain *Form* Penjualan

Form pada Gambar 3.36 akan tampil jika *user login* sebagai kasir. Kasir akan memasukkan data penjualan yang dilakukan oleh konsumen. Tombol tambah digunakan sebagai memasukkan data ke *list* penjualan, tombol pembatalan digunakan untuk membatalkan penjualan yang sudah masuk ke dalam *list* penjualan. Tombol simpan untuk melanjutkan ke *form* pembayaran.

Gambar 3.36 Desain *Form* Penjualan

E.7 Desain Form Pemesanan

Form pada Gambar 3.37 akan tampil jika *user login* sebagai kasir. Kasir akan memasukkan data pemesanan barang yang dipesan oleh konsumen. Fungsi tombol tambah adalah menambahkan barang ke dalam list pemesanan. Tombol *clear* adalah membersihkan *textbox* pemesanan jika terjadi salah memasukkan data. Tombol hapus adalah penghapusan pemesanan dari *list* pemesanan. Tombol ubah dilakukan jika ada perubahan pemesanan yang sebelumnya telah masuk ke dalam *list*. Untuk tombol simpan digunakan untuk masuk ke *form* pembayaran. Tombol *cancel* digunakan untuk membatalkan pemesanan.

The image shows a software form titled "Entry Data Pemesanan". It is divided into two main sections: "Entry Data Pemesanan" and "Detail Pemesan".

Entry Data Pemesanan Section:

- Nota Pemesanan :
- Tanggal Pemesanan :
- Kasir :
- Kode Barang :
- Nama Barang :
- Jenis Barang :
- Harga : Rp.
- Quantity :
- Picture :
- Buttons: Tambah, Clear, Ubah, Hapus
- Table: Detail Pemesanan Barang (with TOTAL PEMESANAN header)

Detail Pemesan Section:

- Nama Pemesan :
- Alamat Pemesan :
- Wilayah Pemesan :
- No Telp :
- Tanggal Pengambilan:
- Buttons: Simpan, Cancel

A large watermark "stikom SURABAYA" is visible across the center of the form.

Gambar 3.37 Desain Form Pemesanan

E.8 Desain Pembayaran Penjualan

Form pada Gambar 3.38 digunakan untuk melakukan pembayaran penjualan setelah melakukan pengisian data penjualan. Tombol simpan digunakan untuk menyimpan pembayaran dan akan mencetak nota penjualan.

Gambar 3.38 Desain *Form* Pembayaran Penjualan

E.9 Desain Pembayaran Pemesanan

Form pada Gambar 3.39 digunakan untuk melakukan pembayaran pemesanan setelah melakukan pengisian data pemesanan. Dalam pembayaran pemesanan bisa dilakukan secara tunai atau dengan memberikan uang muka sebagai jaminan tanda jadi. Tombol simpan digunakan untuk menyimpan pembayaran dan akan mencetak nota pemesanan.

Gambar 3.39 Desain *Form* Pembayaran Pemesanan

E.10 Desain Form Pengambilan Pemesanan

Form Gambar 3.40 digunakan kasir untuk pengambilan pemesanan konsumen. Kasir akan *inputkan* nota pemesanan dari konsumen, atau juga bisa mencari pesanan pada *tab* cari pesanan yang nantinya akan muncul pesanan yang akan diambil hari ini. Pada *form* ini petugas juga mengecek kekurangan pembayaran konsumen jika konsumen tersebut membayar menggunakan cara uang muka. Apabila barang pesanan sudah diambil konsumen maka kasir akan *update* status pengambilan pesanan dengan memberikan cek *list* pada status pengambilan.

Gambar 3.40 Desain *Form* Pengambilan Pemesanan

E.11 Desain Form Stok Barang

Form pada Gambar 3.41 digunakan oleh bagian produksi untuk menambahkan stok barang atau roti yang selesai produksi. setiap produksi roti memiliki kode barang yang sama tetapi dengan kode produksi sesuai dengan

tanggal produksi hari ini. Jika produksi hari ini dilakukan dua kali maka stock barang akan di tambah dengan stok yang selesai diproduksi. Tetapi jika ada barang dengan kode barang yang sama tetapi dengan kode produksi berbeda maka barang akan ditambahkan baru kedalam *database*.

Gambar 3.41 Desain *Form* Stock Barang

E.12 Desain *Form* Laporan Penjualan

Form pada Gambar 3.42 digunakan untuk memunculkan laporan penjualan barang per periode. *Form* ini digunakan pemilik untuk melihat barang yang terjual pada setiap cabang selama periode yang diinginkan. Tombol *view* digunakan untuk memunculkan laporan ketika pemilik memilih cabang dan periode yang diinginkan.

Laporan Penjualan Barang Per Periodik

Nama Cabang :

Kode Cabang :

Tanggal Awal :

Tanggal Akhir :

Kode Barang	Nama Barang	Qty	Harga Barang	Sub Total

Gambar 3.42 Desain Form Laporan Penjualan

E.13 Desain *Form* Laporan Hasil Penjualan

Form pada Gambar 3.43 digunakan oleh pemilik untuk melihat transaksi penjualan per periode. Tombol *view* digunakan untuk melihat laporan hasil penjualan per periode mulai dari tanggal awal dan tanggal akhir yang dipilih oleh pemilik. tombol *clear* digunakan untuk membersihkan *textbox*, *combobox*, dan tanggal yang sebelumnya telah diisi oleh pemilik.

Laporan Hasil Penjualan Harian

Nama Cabang :

Kode Cabang :

Tanggal Awal :

Tanggal Akhir :

Tanggal	Nota Penjualan	Total

Gambar 3.43 Desain *Form* Laporan Hasil Penjualan

E.14 Desain *Form* Laporan Pemesanan

Form pada Gambar 3.44 digunakan oleh pemilik untuk melihat transaksi pemesanan per periode. Tombol *view* digunakan untuk melihat laporan hasil pemesanan per periode mulai dari tanggal awal dan tanggal akhir yang dipilih oleh pemilik. tombol *clear* digunakan untuk membersihkan *textbox*, *combobox*, dan tanggal yang sebelumnya telah diisi oleh pemilik.

Laporan Hasil Pemesanan

Nama Cabang :

Kode Cabang :

Tanggal Awal :

Tanggal Akhir :

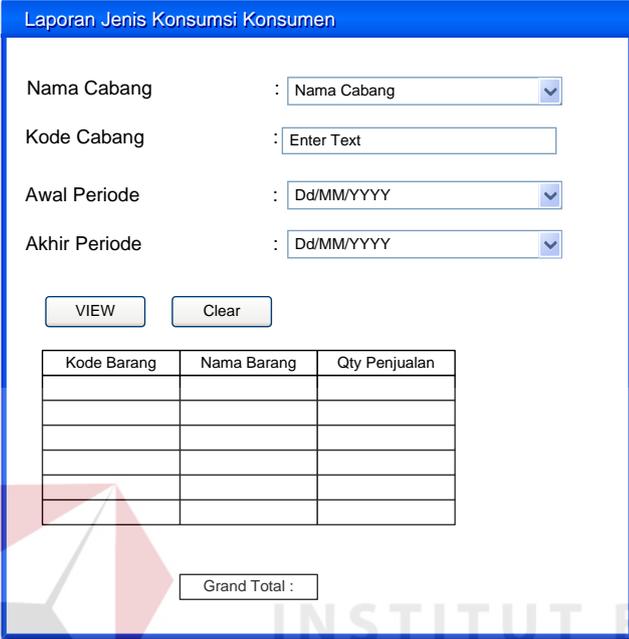
Tanggal	Nota Pemesanan	Total

Gambar 3.44 Desain *Form* Laporan Pemesanan

E.15 Desain *Form* Laporan Konsumsi Konsumen

Form pada Gambar 3.45 digunakan oleh pemilik untuk melihat laporan konsumsi konsumen setiap cabang per periode. Tombol *view* digunakan untuk melihat laporan konsumsi konsumen per periode mulai dari tanggal awal dan tanggal akhir yang dipilih oleh pemilik. tombol *clear* digunakan untuk

membersihkan *textbox*, *combobox*, dan tanggal yang sebelumnya telah diisi oleh pemilik.



Laporan Jenis Konsumsi Konsumen

Nama Cabang :

Kode Cabang :

Awal Periode :

Akhir Periode :

Kode Barang	Nama Barang	Qty Penjualan

Grand Total :

Gambar 3.45 Desain *Form* Laporan Konsumsi Konsumen

