

## BAB III

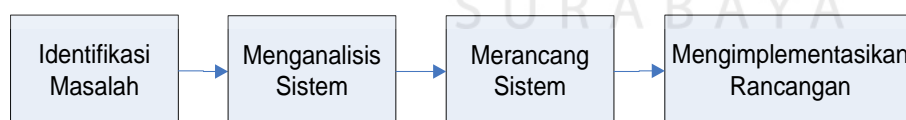
### ANALISIS DAN IMPLEMENTASI

Pada bab ini membahas tentang langkah-langkah yang digunakan dalam penelitian, diantaranya jenis penelitian, sumber data, tahap analisis, blok diagram, dan *system flow*, *context diagram*, *data flow diagram*, *entity relationship diagram*, serta pengolahan data.

#### 3.1 Analisis Sistem

Analisis sistem merupakan tahap pertama dalam mengembangkan perangkat lunak dengan metode *waterfall*. Pada tahap ini dilakukan pengumpulan data, menganalisis data-data tersebut, kemudian mengkomunikasikannya melalui laporan analisis sistem. Temuan-temuan tersebut digambarkan menggunakan diagram-diagram tertentu untuk memperjelas alur atau proses yang ada.

Langkah-langkah yang dilakukan dalam menyelesaikan masalah dapat dilihat pada Gambar 3..



Gambar 3.1 Tahap-Tahap Analisis Sistem

Tahap analisis dimulai dari identifikasi masalah yang ada kemudian dilakukan usaha dalam menemukan kelemahan atau kekurangan yang ada pada sistem yang lama. Setelah itu menentukan dan merumuskan kebutuhan informasi pengguna sistem sehingga dapat ditentukan solusi untuk memecahkan permasalahan tersebut.

### 3.1.1 Identifikasi Masalah

ASTRANS Logistics saat ini secara keseluruhan masih bersifat manual dan belum menggunakan standar akuntansi yang baik karena semua pencatatan dilakukan secara manual berdasarkan hasil dari observasi dan wawancara ASTRANS Logistics masih menggunakan *tools* Microsoft Excell sebagai alat hitung arus kas dan masuk maupun piutang pembayaran yang dilakukan oleh pihak pengirim maupun pihak penerima yang kurang optimal sehingga sering terjadinya selisih saldo kas kecil maupun saldo kas bank.

Kebutuhan perhitungan kas dan laporan laba rugi dilakukan oleh bagian admin, salah satu masalah yang sering terjadi adalah lambatnya proses perhitungan dan yang sering terjadi adalah sistem tidak dapat langsung menjurnalkan setiap transaksi yang masuk maupun transaksi yang keluar, adapun proses perhitungan laba rugi yang sangat lambat karena admin harus menjurnalkan transaksi secara manual untuk dapat menghasilkan laporan laba rugi, ditambah closing harian secara manual untuk mencocokkan transaksi antara pembukuan dan saldo.

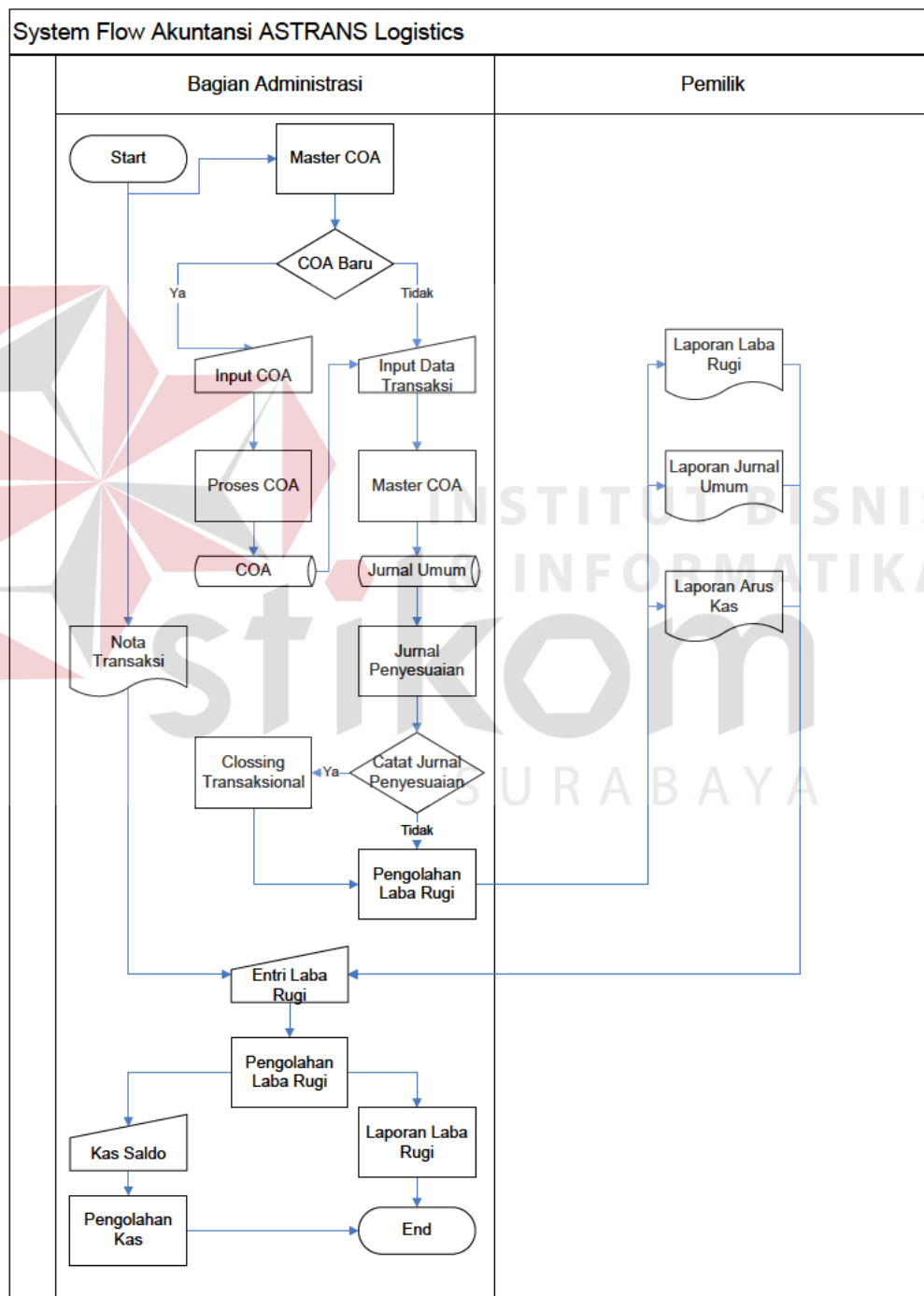
Untuk mengetahui sistem yang saat ini diterapkan, maka dibuatlah aplikasi laporan laba rugi yang dapat membantu memudahkan ASTRANS Logistics untuk mengetahui laba rugi dari setiap transaksi.

### 3.1.2 Analisis dan Pemecahan Masalah

Berdasarkan permasalahan diatas, maka rancang bangun aplikasi akuntansi laba rugi ini akan dapat dengan mudah membantu admin untuk mempercepat perhitungan laporan keuangan dan laporan laba rugi di ASTRANS Logistics.

### 3.1.3 System Flow Akuntansi ASTRANS Logistics

Adapun *system flow* proses akuntansi pada ASTRANS Logistics. Secara umum gambaran sistem yang akan dibangun untuk laporan laba rugi dapat dilihat pada Gambar 3.2.



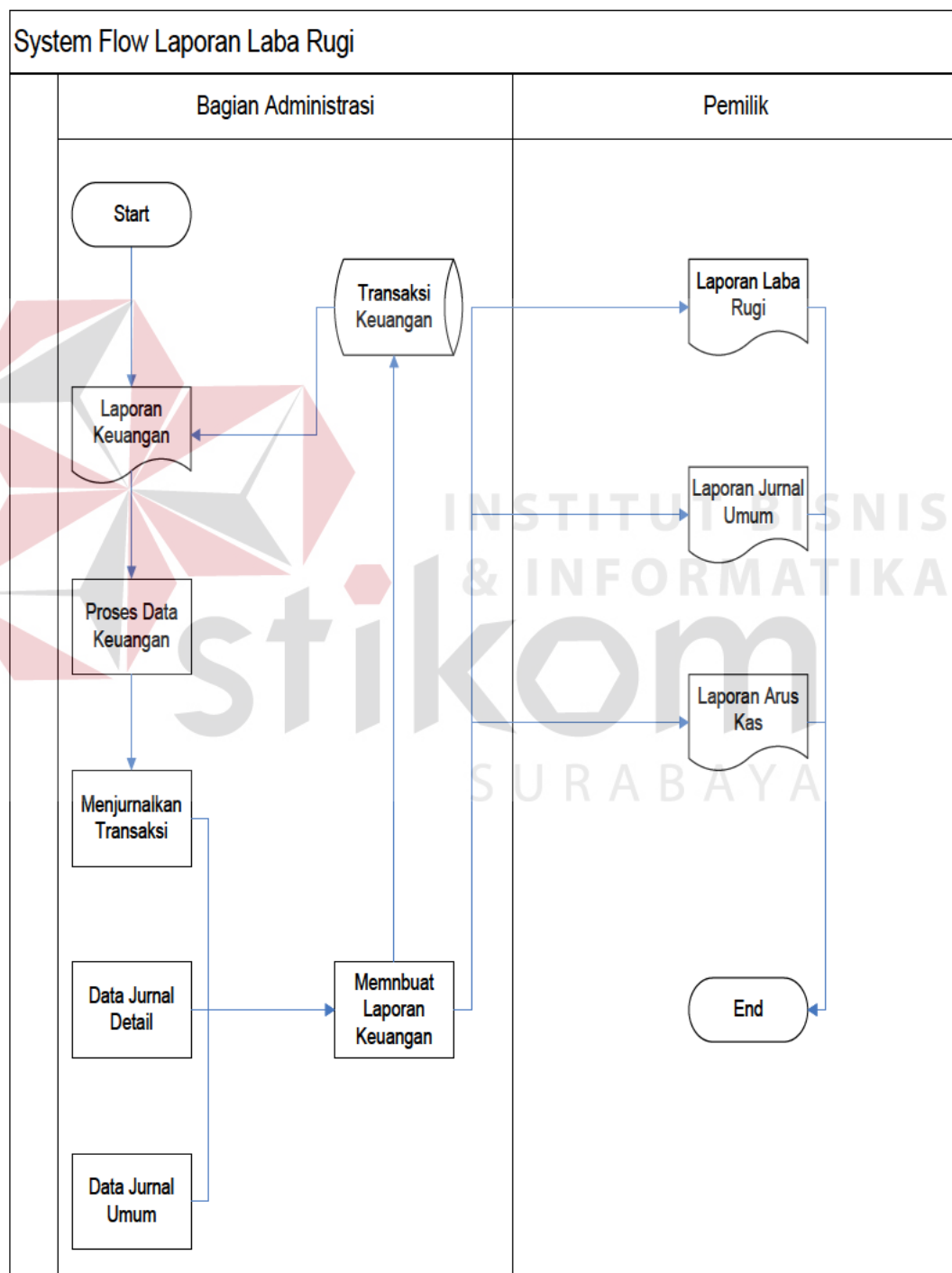
Gambar 3.2 System Flow Proses Akuntansi ASTRANS Logistics

Proses akuntansi yang dilakukan oleh bagian admin di ASTRANS Logistics adalah sebagai berikut:

1. Dimulai dengan bagian administrasi menentukan nota transaksi yang telah dibuat. Kemudian melakukan pengecekan *Chart of Account* (COA). Jika COA yang dibutuhkan ada, maka data akan di-*entry* ke aplikasi akuntansi dalam menu jurnal umum. Jika tidak ada, maka akan di-*entry* kode akun kedalam menu COA.
2. Setelah memasukan data ke jurnal umum, sistem akan memproses ke laporan jurnal umum dan jurnal penyesuaian secara otomatis, sehingga menghasilkan laporan laba rugi, laporan arus kas, dan laporan jurnal umum. Nota transaksi dan laporan arus kas digunakan sebagai acuan penginputan jurnal untuk memasukan transaksi kedalam pelaporan laba rugi.
3. Buku kas saldo dicatat secara manual, setelah data dicatat maka data akan diproses dan disimpan. Kemudian buku kas saldo digunakan untuk memasukan data transaksi maupun data jurnal yang nantinya data akan diolah sebagai laporan laba rugi perusahaan.
4. Agar tidak terjadi ketidaksesuaian kas saldo perusahaan maka akan dilakukan proses *closing*, sebelum proses *closing* dilakukan maka jurnal harus disesuaikan melalui proses jurnal penyesuaian yang nantinya akan berfungsi untuk melihat data arus kas perusahaan secara keseluruhan melalui laporan laba rugi perusahaan.
5. Data yang telah di-*closing* tidak dapat disesuaikan, karena sesuai definisi *closing* tidak dapat melakukan *backdate* transaksi, kesalahan perhitungan akan ditambahkan/dikurangi pada tanggal berikutnya.

### 3.1.4 System Flow Pembuatan Laporan Laba Rugi

Adapun *system flow* proses pembuatan laporan laba rugi pada ASTRANS Logistics. Secara umum gambaran sistem yang akan dibangun untuk laporan laba rugi dapat dilihat pada Gambar 3.3.



Gambar 3.3 System Flow Proses Pembuatan Laporan Laba Rugi

Proses pembuatan laporan laba rugi di ASTRANS Logistics adalah sebagai berikut:

1. Dimulai dari data keuangan dan *master* COA maka akan dilakukan proses penjurnalan transaksi keuangan yang akan menghasilkan proses jurnal dan detail jurnal.
2. Dari proses jurnal dan detail jurnal maka akan dibuatkan laporan keuangan yang akan diberikan kepada pemilik dalam bentuk laporan jurnal umum, laporan arus kas dan laporan laba rugi.
3. Proses pembuatan laporan keuangan didapat dari proses penjurnalan transaksi, data jurnal detail, dan data jurnal umum yang akan diteruskan untuk proses pembuatan laporan laba rugi.
4. Laporan jurnal dan laporan rugi laba yang telah dibuat akan dilaporkan kepada pemilik.

### **3.2 Perancangan Sistem**

Perancangan sistem dibuat agar aplikasi yang akan dibuat dapat berfungsi seperti yang diharapkan. Dalam perancangan sistem ini ada tahapan-tahapan yang harus dilakukan, yaitu pembuatan *Data Flow Diagram*, *Entity Relationship Diagram*, Struktur Tabel dan *Design Input/Output*.

#### **3.2.1 Data Flow Diagram (DFD)**

*Data Flow Diagram* (DFD) menggambarkan aliran data yang terjadi di dalam sistem, sehingga dengan dibuatnya DFD ini akan terlihat arus data yang mengalir dalam *system*.

### A. *Context Diagram*

Berikut ini adalah *desain context diagram* untuk proses laporan laba rugi pada ASTRANS Logistics. Pada *context diagram* terlihat ada dua *entitas* pengguna yang nantinya akan berinteraksi dengan system. Pada *context diagram* ini terdapat dua *entitas* eksternal, Lebih lengkapnya akan dijelaskan pada Gambar 3.4.

#### 1. Administrasi

*Entitas* Bagian Administrasi akan memberikan data periode laba rugi, data periode jurnal, data jurnal, data *closing*, dan data coa semua data tersebut akan diproses oleh sistem.

#### 2. Pemilik

*Entitas* Pemilik akan mendapatkan laporan jurnal dan laporan laba rugi dari hasil proses aplikasi laba rugi.

#### 3. Data COA (*Chart of Account*)

Data COA yang dimaksud diproses ini meliputi pembuatan coa, pengkodean coa dan penentuan status coa.

#### 4. Data Jurnal

Data jurnal dari semua hasil transaksional antara pembukuan dan saldo dari perusahaan, meliputi jurnal penyesuaian dan laporan transaksi.

#### 5. Data *Closing*

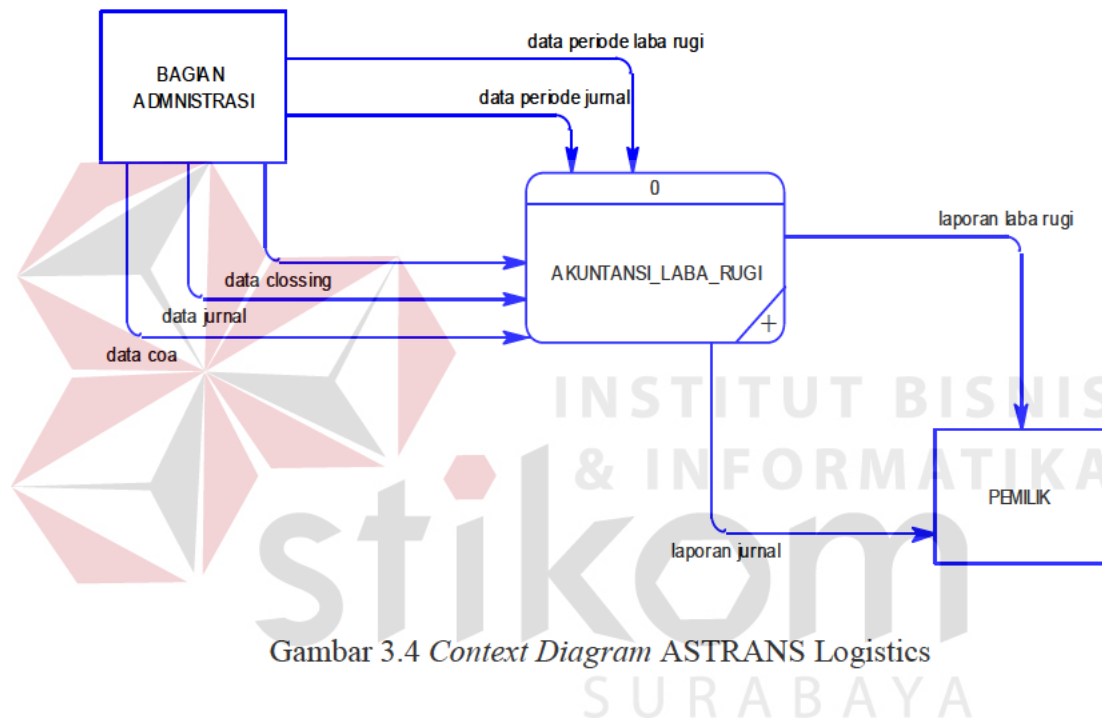
Data *closing* ini meliputi penutupan transaksional perusahaan berdasarkan periode permintaan dari pemilik.

6. Data periode laporan laba rugi

Data laporan laba rugi yang dimaksud adalah laporan dari semua jenis transaksi yang ada didalam perusahaan.

7. Data periode jurnal

Data periode jurnal yang dimaksud adalah laporan dari jurnal transaksi berdasarkan periode yang ditentukan sebelumnya.

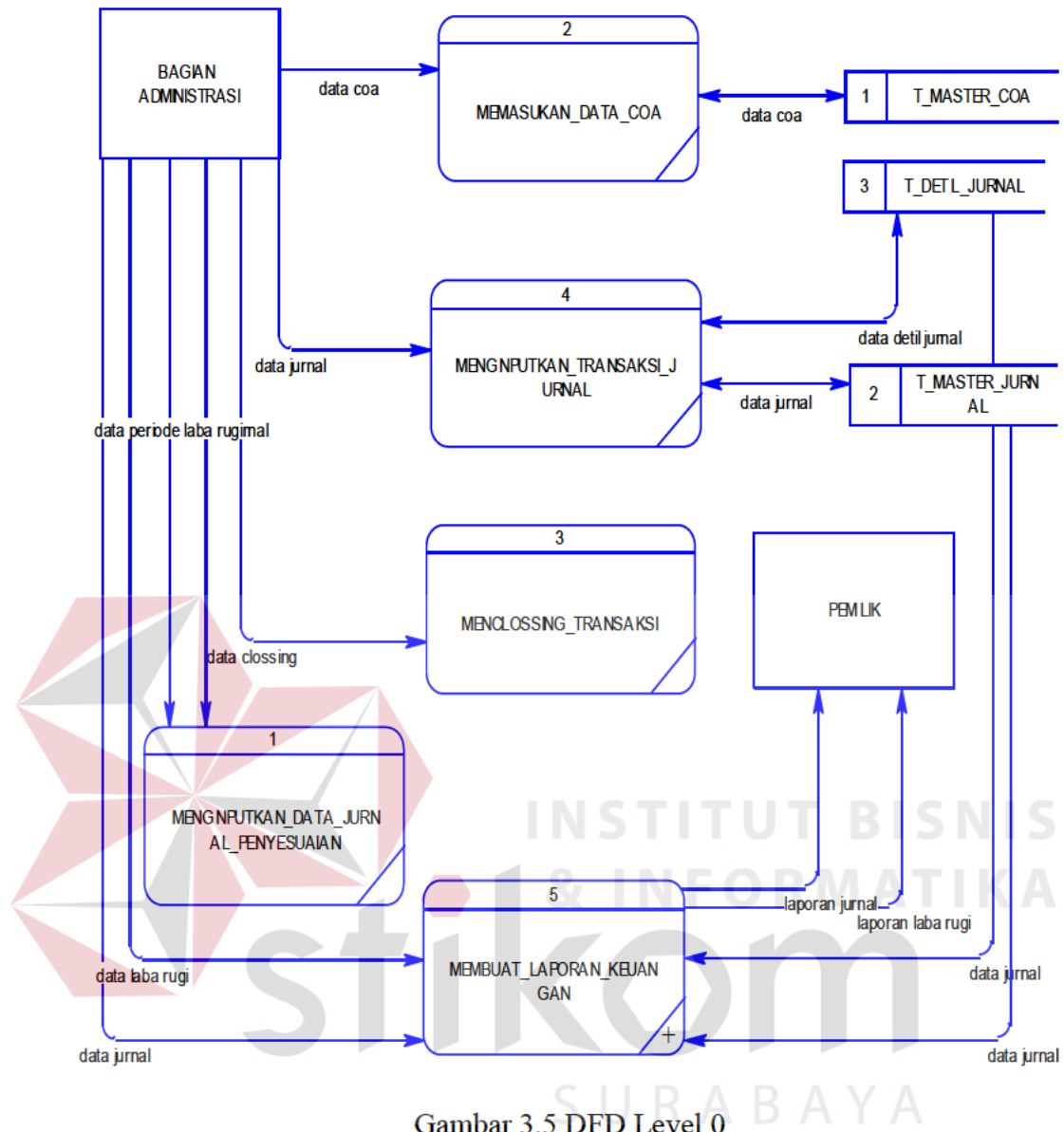


Gambar 3.4 Context Diagram ASTRANS Logistics

### B. DFD Level 0

DFD Level 0 ini merupakan *breakdown* dari *context diagram* yang terdiri dari proses memasukan data coa, proses menginputkan transaksi jurnal, proses *mclosing* transaksi, proses menginputkan data jurnal penyesuaian, dan proses membuat laporan keuangan. DFD Level 0 ini terdapat dua *entitas*, yaitu bagian administrasi dan pemilik. Selain itu juga mulai menampilkan tiga *datastore*. Lebih lengkapnya akan dijelaskan pada Gambar 3.5.

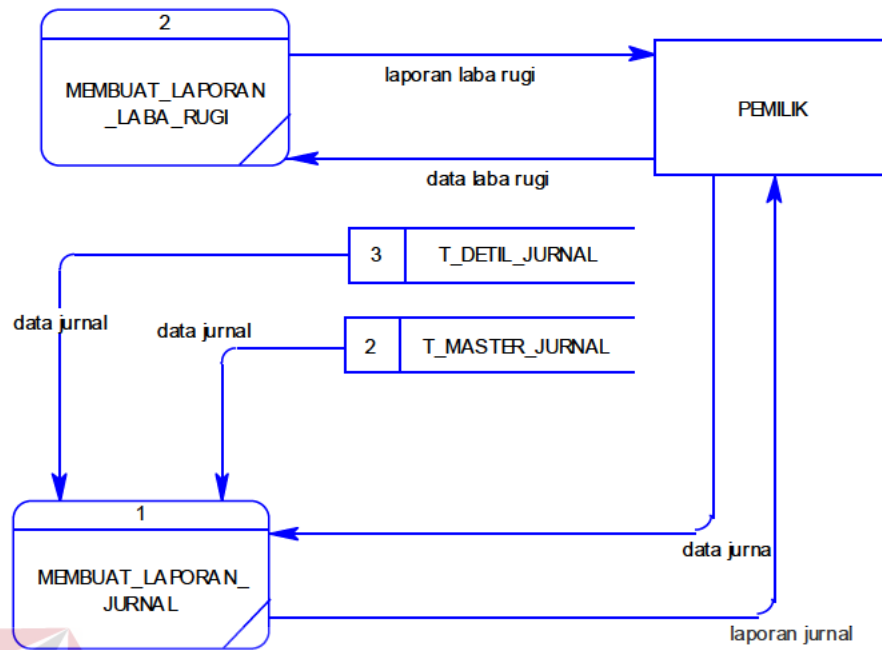




Gambar 3.5 DFD Level 0

### C. DFD Level 1 Proses Membuat Laporan Keuangan

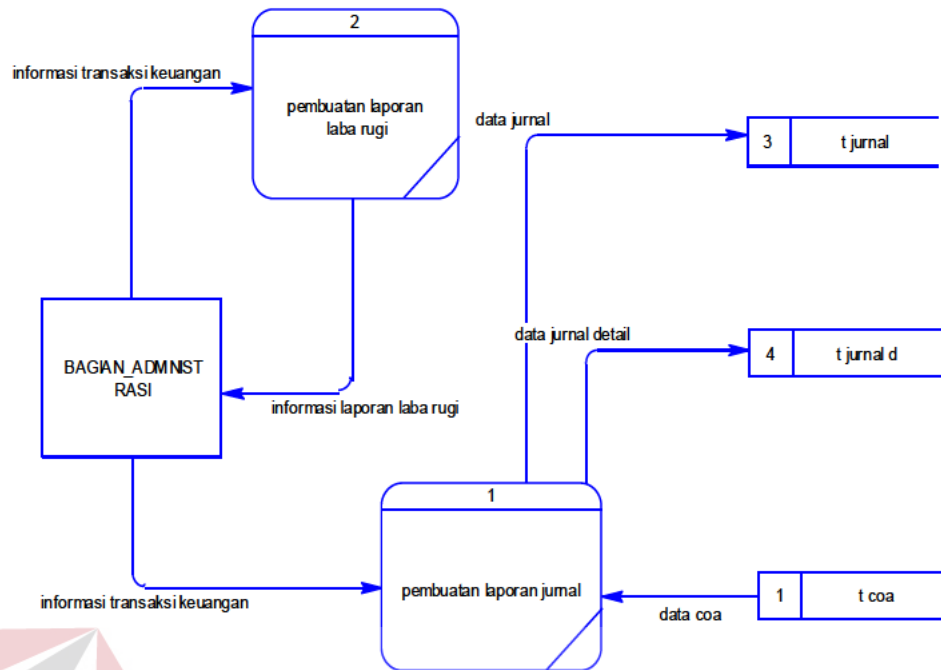
DFD Level 1 proses membuat laporan keuangan memiliki dua proses yaitu, proses membuat laporan laba rugi dan proses membuat laporan jurnal. Terdapat satu *entitas* yaitu pemilik. Arus data seperti laporan laba rugi, data laba rugi, data jurnal yang didapat dari tabel detail jurnal dan data jurnal yang didapat dari tabel master jurnal. Lebih lengkapnya akan dijelaskan pada Gambar 3.6.



Gambar 3.6 DFD Level 1 Proses Master

#### D. DFD Level 1 Proses Transaksi Keuangan

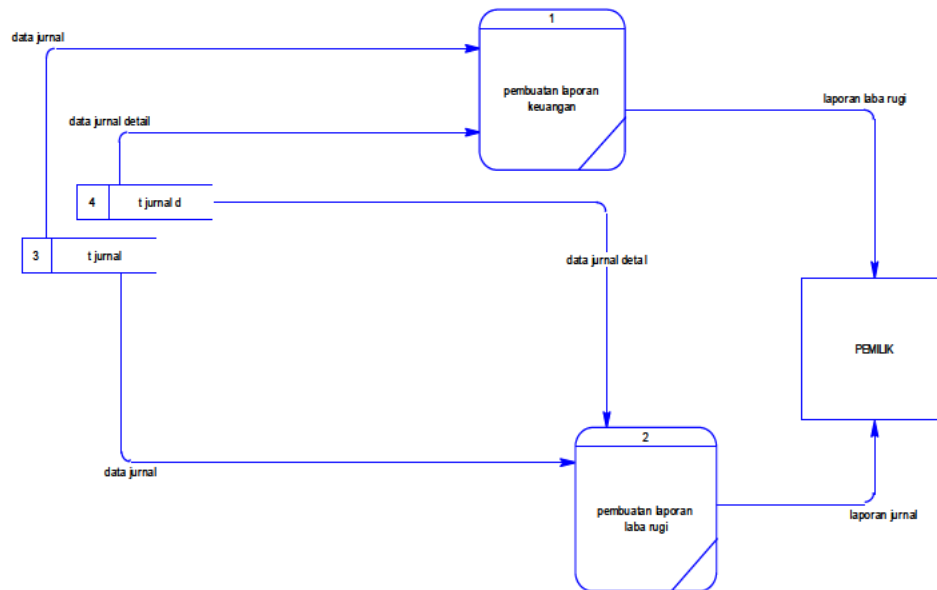
DFD Level 1 proses transaksi keuangan memiliki dua proses yaitu, pembuatan laporan laba rugi, dan pembuatan laporan jurnal. DFD Level 1 proses transaksi melibatkan satu *entitas* yaitu bagian administrasi dan pemilik serta melibatkan tiga *datastore*. *Entitas* bagian administrasi akan mengirimkan informasi transaksi keuangan yang akan diproses untuk pembuatan laporan laba rugi, bagian administrasi juga mengirimkan informasi transaksi keuangan yang akan diproses untuk pembuatan laporan jurnal yang dimana mengambil data dari tabel coa, setelah semua informasi diproses maka akan menghasilkan data jurnal, data jurnal detail yang akan dimasukkan ketabel jurnal dan jurnal detail. Lebih lengkapnya akan dijelaskan pada Gambar 3.7.



Gambar 3.7 DFD Level 1 Proses Transaksi Keuangan

#### E. DFD Level 1 Proses Mencetak Laporan Keuangan

DFD Level 1 proses mencetak laporan keuangan memiliki dua proses yaitu, pembuatan laporan laba rugi, dan pembuatan laporan keuangan. DFD Level 1 proses transaksi melibatkan satu *entitas* yaitu bagian pemilik serta melibatkan dua *datastore*. tabel yang didapatkan dari data jurnal, data jurnal detail akan diproses ke pembuatan laporan laba rugi, pembuatan laporan keuangan agar menghasilkan informasi laporan jurnal yang selanjutnya akan dikirimkan ke *entitas* pemilik. Lebih lengkapnya akan dijelaskan pada Gambar 3.8.



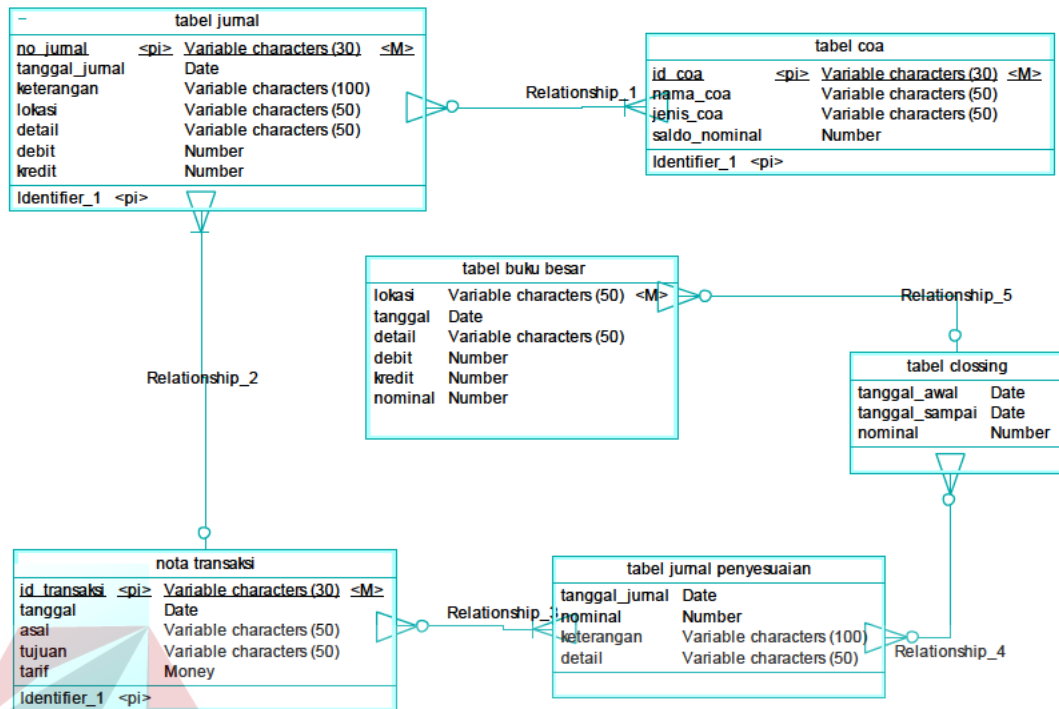
Gambar 3.8 DFD Level 1 Proses Mencetak Laporan Keuangan

### 3.2.2 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) digunakan untuk mempresentasikan, menentukan dan mendokumentasikan kebutuhan sistem kedalam suatu bentuk dengan tujuan untuk menunjukan struktur keseluruhan dari data pemakai. Hubungan antar tabel yang terdapat dalam sistem ERD disajikan dalam bentuk *Conceptual Data Model* (CDM) dan *Physical Data Model* (PDM).

#### A. Conceptual Data Model (CDM)

Sebuah CDM menggambarkan secara keseluruhan konsep struktur basis data yang dirancang untuk suatu program atau aplikasi. Pada CDM belum tergambar jelas tabel-tabel penyusun basis data beserta *field-field* yang terdapat pada setiap tabel. Tabel-tabel tersebut sudah mengalami *relationship* tetapi tidak terlihat pada kolom mana hubungan antar tabel tersebut. Pada CDM telah didefinisikan kolom mana yang menjadi *primary key*. CDM untuk Aplikasi Laba Rugi ASTRANS Logistics dapat dilihat pada Gambar 3.9.

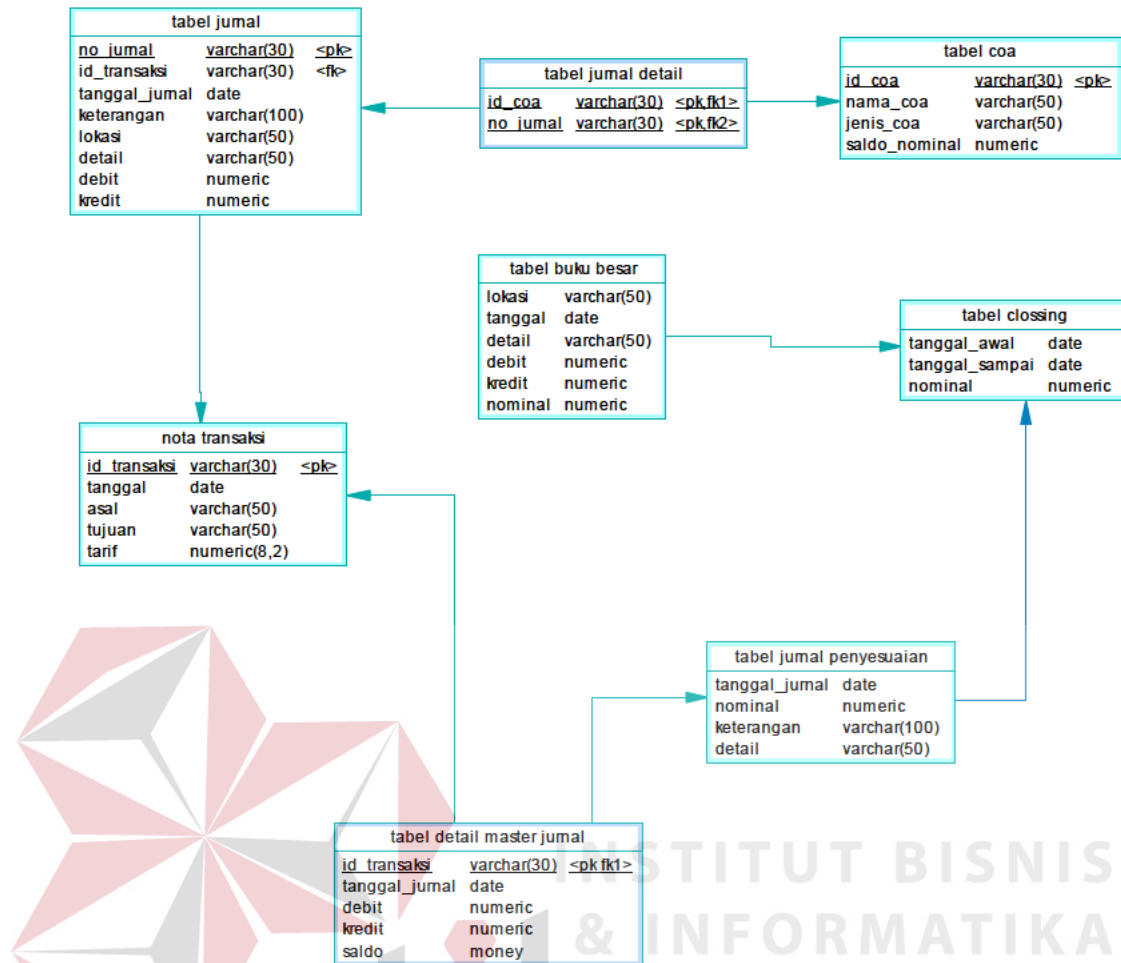


Gambar 3.9 CDM Aplikasi Laporan Laba Rugi ASTRANS Logistics.

Gambar 3.9 merupakan CDM Aplikasi Laporan Laba Rugi ASTRANS Logistics yang terdiri dari enam *entity*, yaitu *entity* tabel coa, *entity* tabel jurnal, *entity* tabel buku besar, *entity* tabel nota transaksi, *entity* tabel jurnal penyesuaian dan *entity* tabel *closing*.

### B. Physical Data Model (PDM)

Sebuah PDM menggambarkan secara detail konsep rancangan struktur basis data yang dirancang untuk suatu program aplikasi. PDM merupakan hasil *generate* dari CDM. Pada PDM tergambar jelas tabel-tabel penyusun basis data beserta *field-field* yang terdapat pada setiap tabel. PDM untuk Aplikasi Laporan Laba Rugi ASTRANS Logistics dapat dilihat pada Gambar 3.10.



Gambar 3.10 PDM Aplikasi Laporan Laba Rugi ASTRANS Logistics.

Gambar 3.10 merupakan PDM Aplikasi Laporan Laba Rugi ASTRANS Logistics yang terdiri dari delapan *entity*, yaitu *entity* tabel coa, *entity* tabel jurnal, *entity* tabel buku besar, *entity* tabel nota transaksi, *entity* tabel jurnal penyesuaian, *entity* tabel *closing*, *entity* tabel detail master jurnal dan *entity* tabel jurnal detail.

### 3.2.3 Struktur Database

Struktur *database* akan menjelaskan tentang fungsi tabel, relasi antar tabel, *constraint*, dan *item-item* yang terdapat dalam sebuah tabel yang digunakan sebagai gambaran *database* yang terbentuk.

### A. Tabel Jurnal Penyesuaian

Nama Tabel : tabel jurnal penyesuaian

*Primary Key* : id jurnal

*Foreign Key* : -

Fungsi : Menyimpan data jurnal

Tabel 3.1 Tabel Jurnal Penyesuaian

No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	id jurnal	Varchar	30	Primary Key	
2.	tanggal jurnal dibuat	Date			
3.	nominal	Numeric			
4.	debit	Money			
5.	kredit	Money			
6.	saldo	Money			
7.	keterangan	Varchar	100		

### B. Tabel Nota Transaksi

Nama Tabel : tabel nota transaksi

*Primary Key* : id transaksi

*Foreign Key* : -

Fungsi : Menyimpan data transaksi

Tabel 3.2 Tabel Nota Transaksi

No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	id transaksi	Varchar	30	Primary Key	
2.	id invoice	Varchar	30	Foreign Key	tabel invoice
3.	no jurnal	Varchar	30	Foreign Key	tabel jurnal
4.	id pelanggan	Varchar	30	Foreign Key	tabel pelanggan
5.	tanggal	Date			
6.	asal	Varchar	100		
7.	tujuan	Varchar	100		
8.	tarif dasar	Money			
9.	panjang	Decimal	10		
10.	lebar	Decimal	10		
11.	tinggi	Decimal	10		
12.	nilai convert	Decimal	10		
13.	nilai kilo	Decimal	10		
14.	jenis ukuran	Varchar	10		
15.	nominal transaksi	Money			
16.	nominal asuransi	Money			



No.	Coloumn	Data Type	Length	Constraint	FK Table
17.	nominal <i>packing</i>	Money			
18.	nominal potongan	Money			
19.	nominal <i>total</i>	Money			

### C. Tabel Buku Besar

Nama Tabel : tabel buku besar

Primary Key :

Foreign Key : no\_jurnal

Fungsi : Menyimpan data buku besar

Tabel 3.3 Tabel Buku Besar

No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	no_jurnal	Varchar	30	Foreign Key	
2.	tanggal jurnal	Date	100		
3.	nominal	Money	100		
4.	keterangan	Varchar	100		

### D. Tabel COA

Nama Tabel : tabel coa

Primary Key : id coa

Foreign Key : -

Fungsi : Menyimpan data coa

Tabel 3.4 Tabel COA

No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	id coa	Varchar	30	Primary Key	
2.	nama coa	Varchar	50		
3.	jenis coa	Varchar	50		
4.	saldo normal	Number			

### E. Tabel Jurnal

Nama Tabel : tabel jurnal

Primary Key : id jurnal

Foreign Key : id invoice, id transaksi

Fungsi : Menyimpan data jurnal

Tabel 3.5 Tabel Jurnal

No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	no jurnal	Varchar	30	Primary Key	
2.	id invoice	Varchar	30	Foreign Key	tabel invoice
3.	id transaksi	Varchar	30	Foreign Key	tabel transaksi
4.	tanggal jurnal	Date			

No.	Coloumn	Data Type	Length	Constraint	FK Table
5.	Keterangan	Varchar	10		

### F. Tabel Invoice

Nama Tabel : tabel *invoice*

Primary Key : id *invoice*

Foreign Key : no jurnal, id transaksi

Fungsi : Menyimpan data *invoice*

Tabel 3.6 Tabel *Invoice*

No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	id <i>invoice</i>	Varchar	30	Primary Key	
2.	no jurnal	Varchar	30	Foreign Key	tabel jurnal
3.	id transaksi	Varchar	30	Foreign Key	tabel transaksi
4.	tanggal <i>invoice</i>	Date			
5.	tanggal jatuh tempo	Date			
6.	tanggal dibayar	Date			
7.	nominal <i>invoice</i>	Number			
8.	nominal dibayar	Number			

### G. Tabel Laporan Laba Rugi

Nama Tabel : tabel laba rugi

*Primary Key* : id transaksi

*Foreign Key* : id transaksi

Fungsi : Menyimpan data laporan laba rugi

Tabel 3.7 Tabel Laporan Laba Rugi

No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	id nota transaksi	Varchar	30	Primary Key, Foreign Key	tabel nota transaksi
2.	id jurnal	Varchar	30	Primary Key, Foreign Key	tabel jurnal
3.	keterangan	Varchar	100		

### H. Tabel Jurnal Detail

Nama Tabel : tabel jurnal *detail*

*Primary Key* : no jurnal, id coa

*Foreign Key* : no jurnal, id coa

Fungsi : Menyimpan data jurnal *detail*

Tabel 3.8 Tabel Jurnal *Detail*

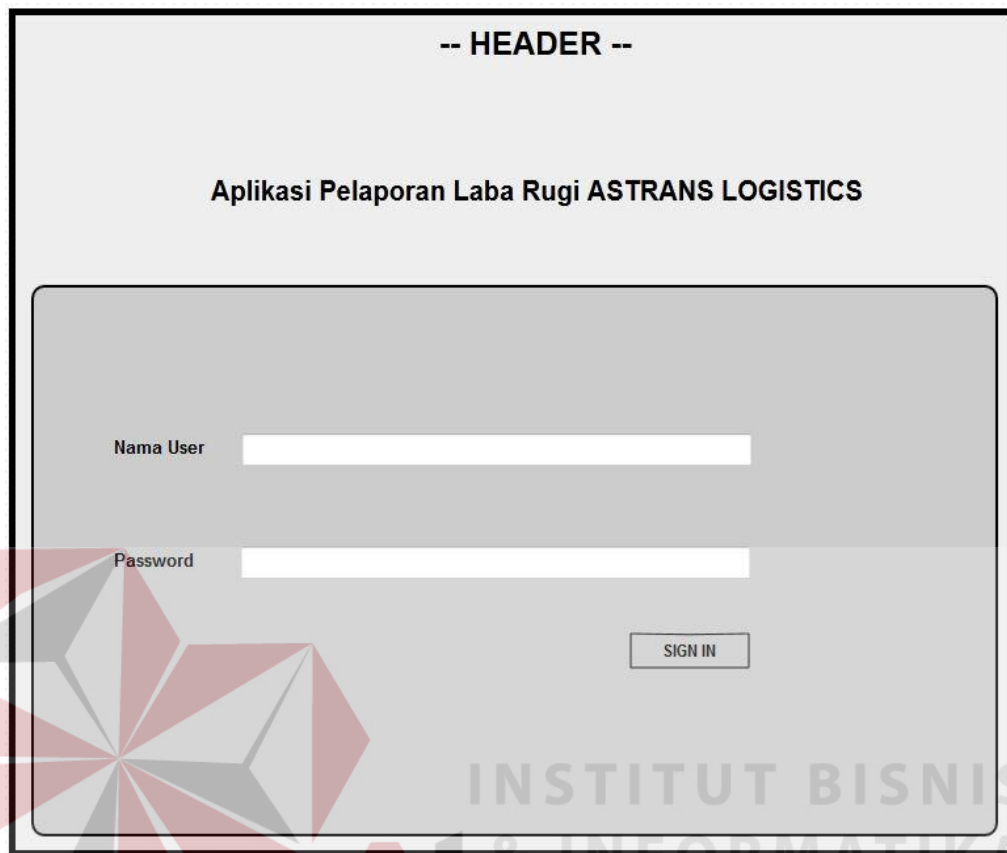
No.	Coloumn	Data Type	Length	Constraint	FK Table
1.	no jurnal	Varchar	30	Primary Key, Foreign Key	tabel jurnal
2.	id coa	Varchar	30	Primary Key, Foreign Key	tabel coa
3.	debit kredit	Varchar	1		
4.	Nominal	Double			

### 3.2.4 Desain *Input/Output*

Desain *input/output* merupakan rancangan *input/output* yang berupa *form* untuk memasukan data dan laporan sebagai informasi yang dihasilkan dari pengolahan data. Desain *input/output* juga merupakan acuan pembuatan aplikasi dalam merancang dan membangun sistem. Melalui desain *input/output* pengembang dapat mengetahui tampilan sistem berbasis *web* yang akan dibangun.

Masukkan atau input merupakan awal dimulainya proses informasi. Bahan mentah dari informasi adalah data yang terjadi dari transaksi yang dilakukan oleh Organisasi, Dinas atau Instansi. Data hasil dari transaksi merupakan dasar untuk memasukkan suatu sistem informasi. Dokumen dasar biasanya berbentuk formulir yang digunakan untuk menangkap (*Capture*) data yang terjadi.

### A. Form Login



-- HEADER --

Aplikasi Pelaporan Laba Rugi ASTRANS LOGISTICS

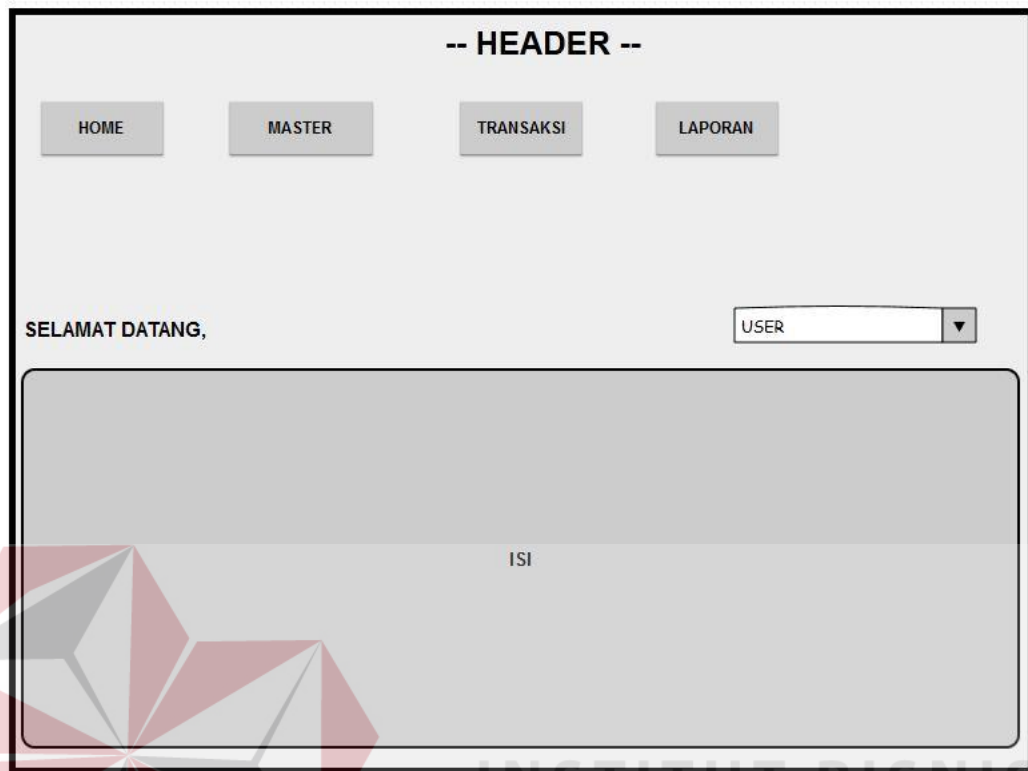
Nama User

Password

Gambar 3.11 *Form Login*

*Form Login* digunakan oleh pengguna untuk masuk ke dalam sistem dengan memasukkan *username* dan *password*. Pada halaman ini juga akan memberikan informasi apabila data pengguna yang dimasukan tidak benar atau tidak terdaftar. Disini tidak terdapat fitur *sign up* karena pembuatan *username* dan *password* akan dilakukan oleh pemilik. Rancangan untuk halaman login dapat dilihat pada Gambar 3.11.

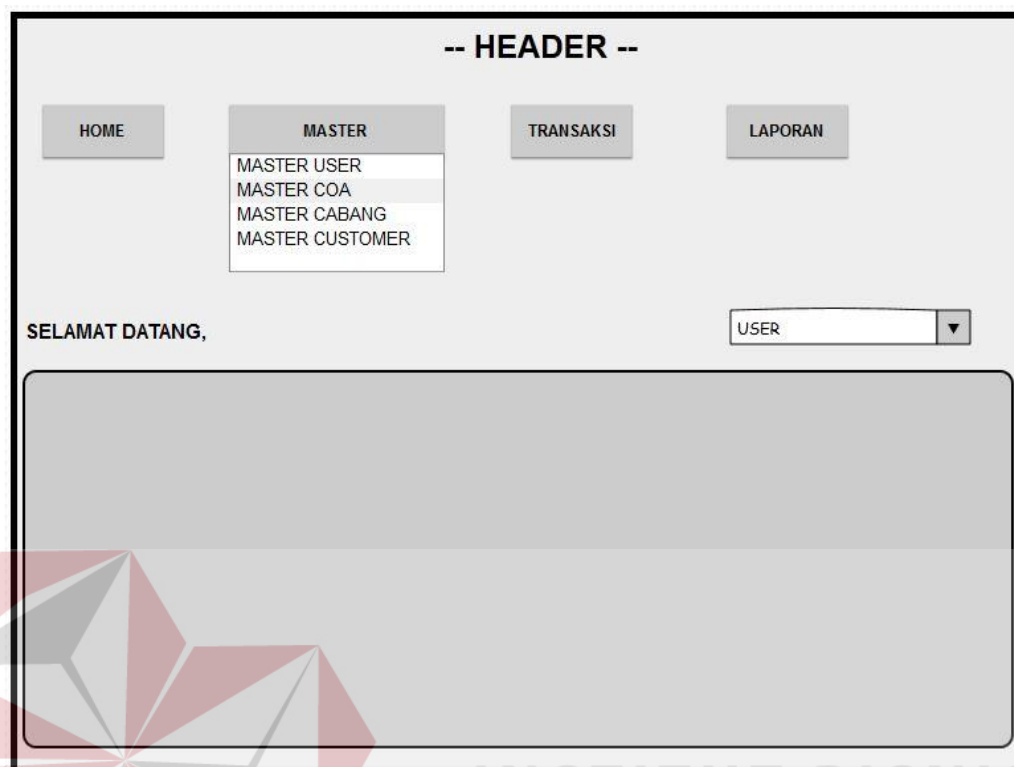
## B. Halaman Utama



Gambar 3.12 Halaman Utama

Halaman utama adalah halaman yang ditampilkan setelah pengguna berhasil melakukan *login*. Pada halaman ini menu yang ditampilkan di bedakan menurut hak akses dari pengguna. Halaman ini juga menampilkan data pengguna yang *login* dan ucapan selamat datang. Rancangan halaman utama dapat dilihat pada Gambar 3.12

### C. Menu *Master*



Gambar 3.13 Menu Master

Pada halaman menu *master* pengguna dapat melihat *master* apa saja yang terdapat dalam menu tersebut. Apabila pengguna memilih salah satu dari *master* tersebut maka pengguna akan dipindahkan ke halaman *master* yang telah dipilih oleh pengguna. *master* ini hanya dapat dipilih oleh *user* berdasarkan hak akses yang diberikan oleh pemilik, hak akses *user* diberikan pada saat pembuatan *user* baru yang dilakukan oleh pemilik. Rancangan menu master dapat dilihat pada Gambar 3.13.



#### D. Menu *Master User*

**-- HEADER --**

HOME    MASTER    TRANSAKSI    LAPORAN    USER

Cari User     + Data Baru

No	Nama Karyawan	User Name	Nama Lokasi	Status	Aksi
1	Mas Syaiful	Admin	Sidoarjo	Aktif	Edit/Tambah
2	Mbak Rikha	Rikha	Sidoarjo	Aktif	Edit/Tambah
3	Mas Samsul	SulSam	Banyuwangi	Non Aktif	Edit/Tambah
4	Mbak Dewi	Dedew	Bali	Aktif	Edit/Tambah

Gambar 3.14 Menu *Master User*

Pada halaman menu *master user* pengguna dapat melihat *user* yang sudah mendaftar dan terdapat fitur pencarian *user*, dihalaman ini pengguna juga dapat mengedit atau menambah dari masing-masing *user* yang sudah terdaftar, juga terdapat fitur untuk menambahkan *user* baru dan apabila pengguna menekan tombol tambah data baru maka pengguna akan dipindahkan kehalaman daftar baru *user*. Rancangan menu *master user* dapat dilihat pada Gambar 3.14.

### E. Daftar Baru *Master User*

-- HEADER --

HOME MASTER TRANSAKSI LAPORAN USER

Cari User + Daftar User

Nama Cabang --- Pilih Cabang ---  
BALI (BALI)  
Banyuwangi (BNYW)  
Mataram (MTR)  
Sidoarjo (SDA)

Nama Karyawan  
Nama User  
Password  
Valid Password

SIMPAN BATAL

Gambar 3.15 Daftar Baru *Master User*

Pada halaman daftar baru *master user* pengguna dapat memasukan data baru *user*. Semua *field* pada halaman ini harus diisi nama karyawan tidak dapat diisi angka, *password* minimal diisi empat karakter dan apabila sebuah *field* tidak diisi maka akan muncul peringatan data harus diisi. *Field* nama cabang dipilih berdasarkan cabang daftar calon *user*. Rancangan daftar baru *master user* dapat dilihat pada Gambar 3.15.

## F. Menu Daftar Baru *Master* Cabang

The screenshot shows a web application interface for adding a new branch. At the top, there is a header labeled "-- HEADER --" with navigation buttons for HOME, MASTER, TRANSAKSI, and LAPORAN, and a user selection dropdown menu showing "USER". Below the header, there is a search bar labeled "Cari COA" and a link "+ Daftar COA". The main form area contains input fields for "Nama Cabang", "Kode", "Alamat", and "No. Telpon", along with "SIMPAN" and "BATAL" buttons.

Gambar 3.16 Menu Daftar Baru *Master* Cabang

Pada halaman daftar baru *master* cabang pengguna dapat memasukan data baru cabang. Semua *field* pada halaman ini harus diisi apabila sebuah *field* tidak diisi maka akan muncul peringatan data harus diisi. *Field* nomer telepon harus diisi berupa angka, apabila tidak diisi dengan angka maka akan muncul peringatan data yang diisi harus berupa angka. Rancangan daftar baru *master* cabang dapat dilihat pada Gambar 3.16

### G. Menu *Master COA*

**-- HEADER --**

HOME    MASTER    TRANSAKSI    LAPORAN    USER

Cari COA     + Data Baru

No	Jenis	Nama COA	Normal Saldo	Status	Aksi
1	Biaya Umum	BBM GrandMax B 6607 CG	Debit	Aktif	Edit/Tambah
2	Pendapatan Jasa	TEST	Kredit	Non Aktif	Edit/Tambah
3	Biaya Umum	Bekal Luar Kota	Debit	Non Aktif	Edit/Tambah
4	Biaya Umum	BBM Operasional	Debit	Aktif	Edit/Tambah
5	Biaya Umum	Biaya Vendor	Debit	Aktif	Edit/Tambah
6	Biaya Umum	Biaya Packing	Debit	Aktif	Edit/Tambah
7	Biaya Umum	Makan Supir Luar Kota	Debit	Non Aktif	Edit/Tambah
8	Biaya Umum	Hutang Kendaraan	Debit	Aktif	Edit/Tambah
9	Biaya Umum	Prive Pak Bayu	Debit	Non Aktif	Edit/Tambah
10	Biaya Umum	Prive Bu Suharti	Debit	Aktif	Edit/Tambah

Gambar 3.17 Menu *Master COA*

Pada halaman menu *master coa* pengguna dapat melihat jenis coa, nama coa, normal *saldo* yang berkurang antara *debit* dan *kredit* dan juga status dari coa tersebut, pengguna juga dapat *edit* atau sekedar menambahkan daftar coa. Pada halaman ini terdapat fitur untuk pencarian data coa dan juga fitur untuk menambahkan data baru coa. Rancangan menu *master coa* dapat dilihat pada Gambar 3.17.

## H. Menu Daftar Baru *Master COA*

**-- HEADER --**

HOME    MASTER    TRANSAKSI    LAPORAN    USER

Cari COA    + Daftar COA

Type: --- Pilih Jenis COA ---

- Aset Lancar (D)
- Bank (D)
- Biaya Umum (D)
- Hutang (K)
- Invoice Bayar Tujuan (K)
- Modal (K)
- Pendapatan Jasa (K)
- Piutang (D)
- Rebusment (D)
- Saldo Laba (K)

Nama COA

Aktif:  Aktif     Non Aktif

SIMPAN    BATAL

Gambar 3.18 Menu Daftar Baru *Master COA*

Pada halaman menu daftar baru *master coa* pengguna dapat mengisi daftar coa baru. Semua *field* dihalaman ini harus diisi, apabila data tidak diisi maka akan muncul peringatan data harus diisi. *Field type* akan memunculkan *list box* untuk memudahkan pengguna dalam memilih tipe coa yang akan diisi, dan juga menggunakan *radio button* untuk memilih status coa. Rancangan menu daftar baru *master coa* dapat dilihat pada Gambar 3.18.

## I. Menu *Master Customer*

**-- HEADER --**

HOME    MASTER    TRANSAKSI    LAPORAN    USER

Cari Customer     + Data Baru Customer

No	Nama Customer	Alamat Customer	CP Customer	No. Telepon	Tarif	Aksi
1	PT INDOFOOD	IFOOD	Surabaya	0123456789	10.000.000	Edit/Tambah
2	PT WINGS	WINGS	Sidoarjo	7894561230	22.500.000	Edit/Tambah
3	PT SEMANGAT BARU	SEBA	Banyuwangi	7539514682	8.600.000	Edit/Tambah
4	PT PERMATA JAYA	PEJA	Bali	8195746389	7.482.00	Edit/Tambah

Gambar 3.19 Menu *Master Customer*

Pada halaman menu *master customer* pengguna dapat melihat siapa saja yang telah menjadi *customer* selama ini, daftar *menu customer* berupa informasi nama *customer*, alamat, *contact person*, nomer telepon beserta tarif, pengguna juga dapat mengedit atau sekedar menambahkan daftar untuk data baru *customer*. Pada halaman ini terdapat fitur untuk pencarian data *customer* dan juga fitur untuk menambahkan data baru *customer*. Rancangan menu *master customer* dapat dilihat pada Gambar 3.19.

## J. Menu *Master* Daftar Baru *Customer*

The screenshot shows a web application interface for adding a new customer. At the top, there is a header with the text "-- HEADER --" and four navigation buttons: HOME, MASTER, TRANSAKSI, and LAPORAN. To the right of these buttons is a dropdown menu currently showing "USER". Below the header, there is a search bar labeled "Cari Customer" and a button labeled "+ Daftar Customer". The main content area is a form with the following fields: "Nama Customer", "Alamat Customer", "CP Customer", "Nomer Telpon", and "Tarif". At the bottom of the form, there are two buttons: "SIMPAN" and "BATAL". A large, semi-transparent watermark for "STIKOM SUPABAYA INSTITUT BISNIS & INFORMATIKA" is overlaid on the form.

Gambar 3.20 Menu *Master* Daftar Baru *Customer*

Pada halaman menu *master* daftar baru *customer* pengguna dapat mengisi daftar *customer* baru. Semua *field* dihalaman ini harus diisi, apabila data tidak diisi maka akan muncul peringatan data harus diisi. *Field* nomer telepon harus diisi berupa angka, apabila tidak diisi dengan angka maka akan muncul peringatan data yang diisi harus berupa angka, pada *field* tarif akan langsung terisi secara otomatis *default* minimal tarif. Rancangan menu *master* daftar baru *customer* dapat dilihat pada Gambar 3.20.

## K. Menu Transaksi Nota Transaksi

**-- HEADER --**

HOME    MASTER    TRANSAKSI    LAPORAN    USER ▾

Tanggal  s/d     + Data Baru

No	Tanggal	No Transaksi	Nama Customer	Asal Tujuan	Tarif KG	Keterangan Isi	Aksi

Gambar 3.21 Menu Nota Transaksi

Pada halaman menu nota transaksi pengguna dapat melihat informasi transaksi yang berisi tanggal transaksi, nomer transaksi, nama *customer*, asal tujuan transaksi, tarif perkilogram, keterangan isi, pengguna juga dapat mengedit atau sekedar menambahkan daftar transaksi baru. Pada halaman ini terdapat fitur untuk pencarian transaksi dapat melakukan pencarian berdasarkan tanggal transaksi. Apabila pengguna menekan tombol tambah *data* transaksi maka pengguna akan dipindahkan ke *form* pengisian transaksi. Rancangan menu nota transaksi dapat dilihat pada Gambar 3.21.



## L. Menu Daftar Baru Transaksi

**-- HEADER --**

HOME
MASTER
TRANSAKSI
LAPORAN

USER ▼

ADD / Edit Resi
View Daftar Resi

No. Transaksi  \*Akan terisi otomatis jika baru

---

Tgl Terima

Customer

Tarif

---

Asal

Tujuan

Jarak

Ket. Penerima

Ket. Isi

Ket. Koli

Type Kirim  KG  VOLUME

Panjang

Lebar

Tinggi

Berat KG

---

Biaya Kirim

Biaya Asuransi

Diskon

Total

---

Bayar

Sisa Kredit

Gambar 3.22 Menu Daftar Baru Transaksi

Pada halaman menu daftar baru transaksi pengguna dapat mengisi daftar transaksi baru. Semua *field* dihalaman ini harus diisi, apabila data tidak diisi maka akan muncul peringatan data harus diisi. *Field* nomer transaksi akan diurutkan secara otomatis berdasarkan jumlah transaksi yang telah dilakukan, tanggal terima transaksi juga akan terisi secara otomatis berdasarkan tanggal transaksi itu dibuat, pada *field customer* akan memunculkan *list box* berdasarkan data *customer* yang sudah diisi pada halaman customer, data tarif akan secara otomatis terisi berdasarkan jumlah minimal tarif pengiriman, *field* asal dan tujuan akan

memunculkan *list box* untuk memudahkan pengguna menentukan asal dan tujuan pengiriman barang, *field* jarak akan terisi secara otomatis berdasarkan asal dan tujuan pengiriman. *Type* kirim menggunakan *radio button* yang terdiri dari dua opsi yaitu kilogram dan *volume*, pada *field* panjang, lebar, tinggi akan terisi otomatis ketika pengguna menentukan tipe kirim pada *field* sebelumnya. Pada *field* biaya kirim akan terisi otomatis berdasarkan tipe kirim yang ditentukan oleh pengguna, pada *field* biaya asuransi dan diskon akan diisi sesuai kebijakan perusahaan dan pengguna, pada *field total* akan otomatis melakukan perhitungan jumlah biaya berdasarkan tipe kirim, diskon, dan biaya asuransi. Pada *field* bayar secara otomatis akan langsung menghitung jumlah yang dibayar dan sisa *kredit* yang harus dilunasi. *Field* panjang, lebar, tinggi, berat kilogram, biaya asuransi, jumlah biaya dan diskon harus diisi berupa angka, apabila tidak diisi dengan angka maka akan muncul peringatan data yang diisi harus berupa angka, pada *field* tarif akan langsung terisi secara otomatis *default* minimal tarif. Rancangan menu daftar baru transaksi dapat dilihat pada Gambar 3.22.

#### M. Menu Daftar Baru Nota Transaksi

The screenshot shows a web application interface with a header section titled "-- HEADER --". The header contains four main navigation buttons: HOME, MASTER, TRANSAKSI, and LAPORAN. The TRANSAKSI button is currently selected, and a dropdown menu is open below it, listing the following options: RESI, MANIFEST, PENERIMAAN MANIFEST, INVOICE, PENERIMAAN INVOICE, JURNAL UMUM, and CLOSING. Below the header, there is a greeting "SELAMAT DATANG," and a user selection dropdown menu labeled "USER". The main content area below the header is currently empty.

Gambar 3.23 Menu *Master* Transaksi

Pada halaman menu *master* transaksi pengguna dapat melihat *master* apa saja yang terdapat dalam menu tersebut. Di halaman menu transaksi ini pengguna dapat melihat transaksi, *manifest*, penerimaan *manifest invoice*, penerimaan *invoice*, jurnal umum dan *closing* harian. Apabila pengguna memilih salah satu dari *master* tersebut maka pengguna akan dipindahkan ke halaman *master* yang telah dipilih oleh pengguna. Rancangan menu *master* transaksi dapat dilihat pada Gambar 3.23.

## N. Menu Laporan Data Jurnal

-- HEADER --

HOME    MASTER    TRANSAKSI    LAPORAN    USER

Data Jurnal

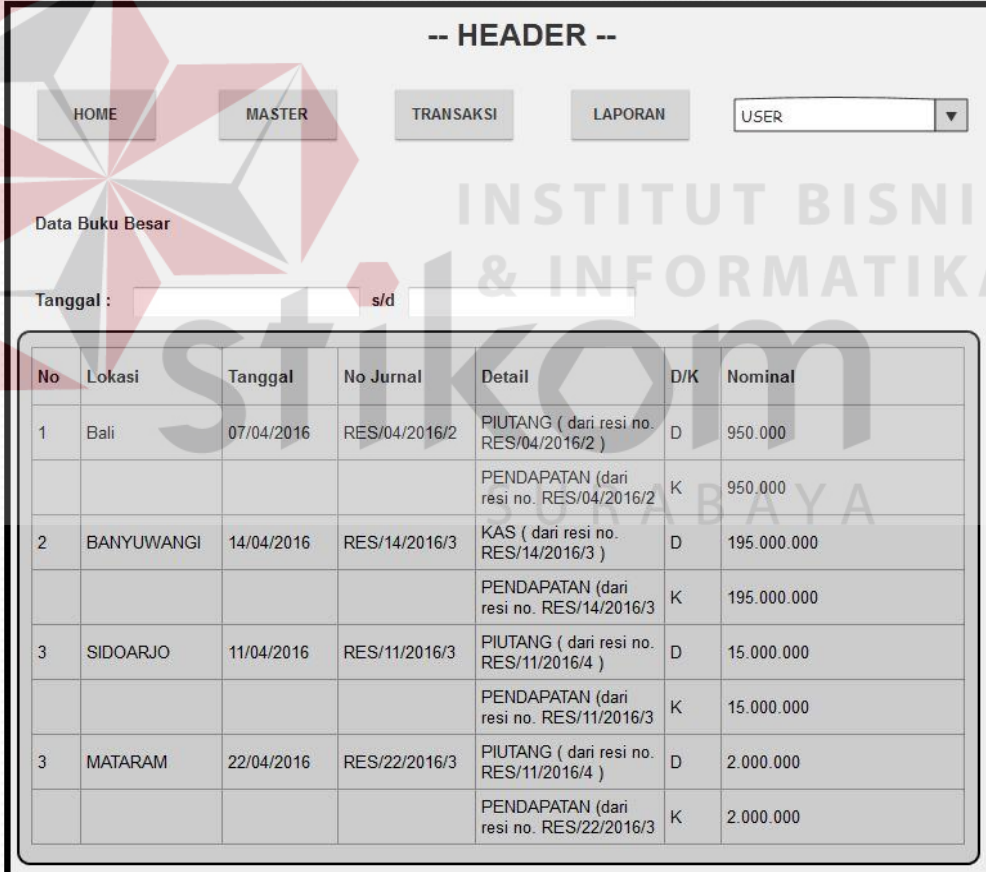
Tanggal :  s/d

No	Lokasi	Tanggal	No Jurnal	Detail	D/K	Debet	Kredit
1	Bali	07/04/2016	RES/04/2016/2	PIUTANG ( dari resi no. RES/04/2016/2 )	D	950.000	
				PENDAPATAN (dari resi no. RES/04/2016/2)	K		950.000
2	BANYUWANGI	14/04/2016	RES/14/2016/3	KAS ( dari resi no. RES/14/2016/3 )	D	195.000.000	
				PENDAPATAN (dari resi no. RES/14/2016/3)	K		195.000.000
3	SIDOARJO	11/04/2016	RES/11/2016/3	PIUTANG ( dari resi no. RES/11/2016/4 )	D	15.000.000	
				PENDAPATAN (dari resi no. RES/11/2016/3)	K		15.000.000
3	MATARAM	22/04/2016	RES/22/2016/3	PIUTANG ( dari resi no. RES/11/2016/4 )	D	2.000.000	
				PENDAPATAN (dari resi no. RES/22/2016/3)	K		2.000.000

Gambar 3.24 Menu Laporan Data Jurnal

Pada halaman laporan data jurnal pengguna dapat melihat data jurnal berdasarkan transaksi-transaksi yang ada, data jurnal berupa informasi lokasi, tanggal, no jurnal, *detail* jurnal, *debit* dan *kredit*. Pengguna juga dapat melihat data jurnal berdasarkan tanggal yang diinginkan pengguna, pada *field* tanggal akan muncul secara otomatis ketika dipilih oleh pengguna, ini dibuat agar memudahkan pengguna untuk memilih dan menentukan tanggal. Rancangan menu laporan data jurnal dapat dilihat pada Gambar 3.24.

### O. Menu Laporan Laba Rugi



No	Lokasi	Tanggal	No Jurnal	Detail	D/K	Nominal
1	Bali	07/04/2016	RES/04/2016/2	PIUTANG ( dari resi no. RES/04/2016/2 )	D	950.000
				PENDAPATAN (dari resi no. RES/04/2016/2)	K	950.000
2	BANYUWANGI	14/04/2016	RES/14/2016/3	KAS ( dari resi no. RES/14/2016/3 )	D	195.000.000
				PENDAPATAN (dari resi no. RES/14/2016/3)	K	195.000.000
3	SIDOARJO	11/04/2016	RES/11/2016/3	PIUTANG ( dari resi no. RES/11/2016/4 )	D	15.000.000
				PENDAPATAN (dari resi no. RES/11/2016/3)	K	15.000.000
3	MATARAM	22/04/2016	RES/22/2016/3	PIUTANG ( dari resi no. RES/11/2016/4 )	D	2.000.000
				PENDAPATAN (dari resi no. RES/22/2016/3)	K	2.000.000

Gambar 3.25 Menu Laporan Laba Rugi

Pada halaman laporan laba rugi pengguna dapat melihat data laba rugi berdasarkan transaksi-transaksi yang ada, laporan laba rugi berupa informasi

lokasi, tanggal, no jurnal, detail jurnal, *debit* dan *kredit*. Pengguna juga dapat melihat data jurnal berdasarkan tanggal yang diinginkan pengguna, pada *field* tanggal akan muncul secara otomatis ketika dipilih oleh pengguna, ini dibuat agar memudahkan pengguna untuk memilih dan menentukan tanggal. Rancangan menu laporan laba rugi dapat dilihat pada Gambar 3.25.

