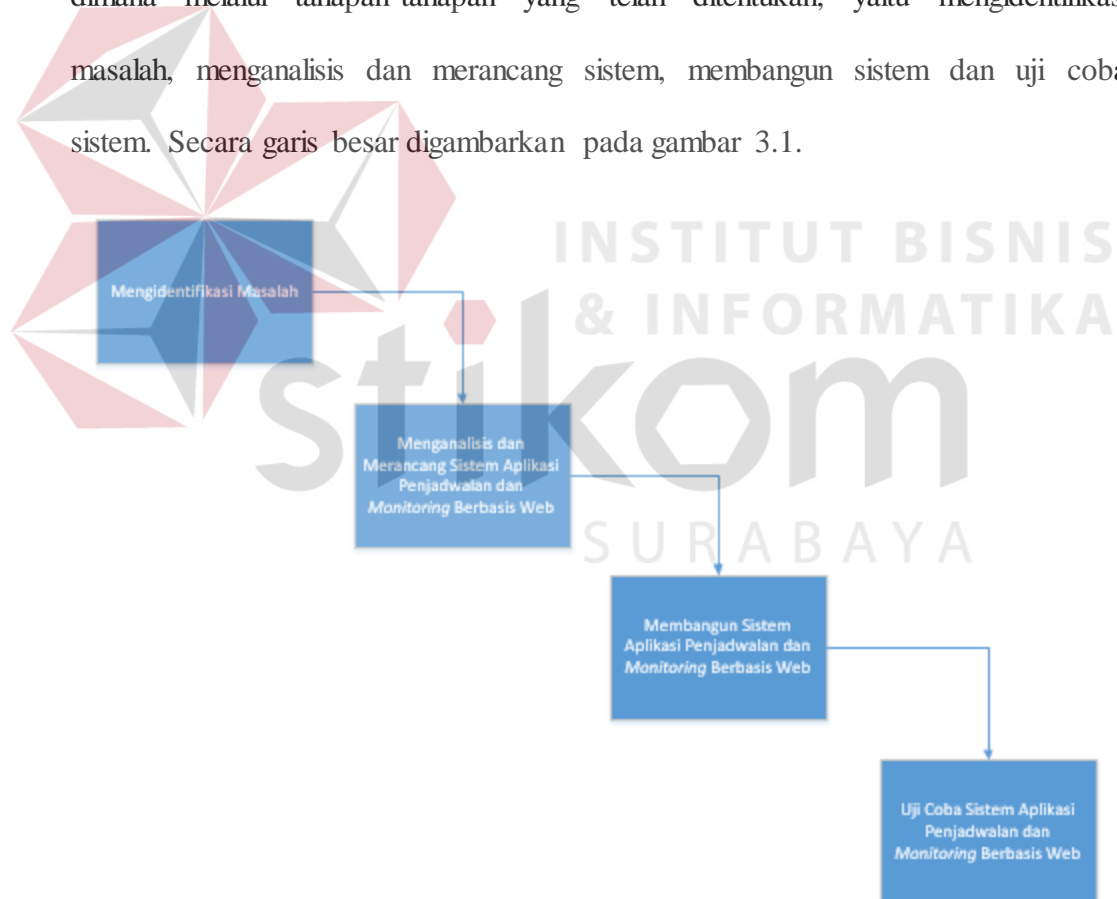


## BAB III

### ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini akan dibahas tentang identifikasi masalah, analisis dan perancangan sistem, rancangan pengujian, dan evaluasi sistem dalam rancang bangun aplikasi penjadwalan kunjungan dan *monitoring* pelanggan berbasis web pada PT. Royal Inti Mandiri Abadi. Model pengembangan yang digunakan pada penelitian ini menggunakan teori Pressman (2015) yaitu dengan model *waterfall* dimana melalui tahapan-tahapan yang telah ditentukan, yaitu mengidentifikasi masalah, menganalisis dan merancang sistem, membangun sistem dan uji coba sistem. Secara garis besar digambarkan pada gambar 3.1.



Gambar 3.1 Langkah-Langkah Analisis Sistem

### 3.1 Identifikasi Permasalahan

PT. Royal Inti Mandiri Abadi Abadi, atau disebut PT. RIMA, adalah perusahaan yang bergerak dalam bidang distribusi semen. PT. Royal Inti Mandiri Abadi adalah distributor utama Semen Tiga Roda wilayah Jawa Timur perwakilan dari PT Indocement Tunggal Prakarsa, Tbk selaku produsen Semen Tiga Roda.

Sebagai sebuah perusahaan yang bergerak dalam bidang distribusi semen PT. Royal Inti Mandiri Abadi mempunyai banyak cabang yang terdapat di provinsi Jawa Timur yang kemudian disebut sebagai zona.

Dalam setiap zona terdapat *sales* yang bertugas untuk memasarkan produk Semen Tiga Roda. Selain memasarkan produk tersebut *sales* mempunyai tugas untuk mengumpulkan informasi dari toko bangunan atau pelanggan.

Pada PT. Royal Inti Mandiri Abadi *sales* mempunyai tugas untuk mengumpulkan informasi dari toko pelanggan seperti menanggapi keluhan dari toko terkait dengan produk yang dipasarkan, besarnya kapasitas gudang toko, harga kompetitor, promo program dari kompetitor dan lain sebagainya. Untuk memperoleh informasi tersebut *sales* dapat melakukan kunjungan sesuai dengan jadwal yang telah dibuat sebelumnya oleh divisi *Customer Service*. Dari hasil kunjungan tersebut *sales* wajib membuat laporan dengan format yang telah ditentukan untuk kemudian diberikan kepada Kepala Perwalian atau Zona dan Kepala Perwalian akan mengevaluasi laporan kunjungan dan kemudian diberikan kepada divisi *Customer Service*.

Dalam kunjungan tersebut terdapat beberapa masalah yang seringkali harus dihadapi oleh divisi *Customer Service* seperti:

1. *Sales* melakukan kunjungan di luar batas waktu yang telah ditentukan.

2. Laporan yang diberikan kepada divisi *Customer Service* tidak *valid* atau tidak sesuai dengan yang diharapkan.
3. Jadwal kunjungan yang dibuat seringkali tidak tersampaikan dengan baik di masing-masing zona.
4. PT. Royal Inti Mandiri Abadi tidak bisa memonitoring kinerja dari masing-masing zona karena pembuatan jadwal kunjungan dan laporan yang dihasilkan masih manual dengan menggunakan aplikasi *excel*.

Masalah lain muncul dengan banyaknya jumlah pelanggan yang harus ditangani oleh *sales* di PT. Royal Inti Mandiri Abadi yang berdampak pada seringnya terjadi kesalahan pada saat sebelum dan sesudah dilakukan penjadwalan kunjungan ke pelanggan karena proses masih dilakukan secara *manual* seperti pembuatan jadwal kunjungan yang masih menggunakan *email* dan hasil laporan disimpan dengan menggunakan aplikasi *excel*. Oleh karena itu dibutuhkan adanya suatu aplikasi yang dapat mengelola penjadwalan kunjungan, *monitoring* kinerja pelanggan, dan *monitoring* kinerja *sales* sehingga aktivitas kunjungan yang terdapat pada tiap-tiap zona dapat dimonitor dengan baik dan dapat menghasilkan laporan yang informatif. Sehingga dapat meningkatkan penjualan dan pelayanan kepada pelanggan PT. Royal Inti Mandiri Abadi itu sendiri.

## **3.2 Analisis**

### **3.2.1 Analisis Kebutuhan Sistem**

Untuk mengetahui permasalahan dalam sistem penjadwalan dan *monitoring* pelanggan yang terdapat pada PT. Royal Inti Mandiri Abadi (RIMA), diperlukan langkah-langkah dalam menganalisis kebutuhan aplikasi, yaitu:

1. Pengamatan/*Observasi*

Dalam menganalisa kebutuhan, dilakukanlah observasi terhadap pabrik. Observasi tersebut untuk mengamati proses yang dilakukan pihak divisi *Customer Service* dalam pembuatan jadwal kunjungan dan *monitoring* pelanggan. Langkah ini dilakukan untuk melihat dan mengidentifikasi sistem penjadwalan yang sedang digunakan pada saat ini. Apakah dalam sistem yang mereka gunakan pada saat ini sudah mencukupi atau masih memiliki kekurangan-kekurangan yang bisa menyebabkan masalah bagi divisi *customer service* dalam mengelola laporan kunjungan pelanggan. Dengan adanya observasi diharapkan mampu menyelesaikan masalah yang telah diuraikan di dalam latar belakang yang telah disusun dalam penelitian ini.

## 2. Wawancara

Metode wawancara bertujuan mengumpulkan informasi. Metode ini dilakukan dengan melakukan tanya jawab kepada Kepala bagian divisi *Customer Service* PT. Royal Inti Mandiri Abadi yang dalam penelitian ini dijadikan sebagai studi kasus pembuatan tugas akhir ini. Dalam tahap wawancara ini, peneliti menggali informasi mengenai seperti apa visi dan misi perusahaan, bagaimana proses bisnis dalam penjadwalan kunjungan dan *monitoring* pelanggan.

## 3. Studi *Literatur* dan Penentuan Solusi

Setelah selesai dalam melakukan pengamatan dan wawancara, maka langkah selanjutnya adalah studi literatur dan penentuan solusi. Pada tahapan ini, dilakukan studi literatur yang berguna untuk memperoleh informasi yang diperlukan dalam pembuatan rancang bangun aplikasi penjadwalan kunjungan dan *monitoring* pelanggan pada PT. Royal Inti Mandiri Abadi. Informasi yang didapat tidak hanya melalui buku saja, tetapi juga berupa jurnal ilmiah dan skripsi mahasiswa.

Informasi literature berupa proses penjadwalan dan proses *monitoring* pelanggan yang telah dijalankan di PT. Royal Inti Mandiri Abadi. Pengumpulan berkas ini bertujuan untuk mengidentifikasi masalah yang ada pada PT. Royal Inti Mandiri Abadi.

### 3.2.2 *User Requirement*

Guna mengetahui kebutuhan pengguna yang terkait beserta tugas-tugasnya dilakukan analisis kebutuhan pengguna (*user requirement*) pada tabel 3.1 di bawah yang telah disesuaikan yang terkait dengan aplikasi penjadwalan kunjungan dan *monitoring* pelanggan pada PT. Royal Inti Mandiri Abadi. Setelah dilakukan analisis kebutuhan pengguna selanjutnya akan dilakukan kebutuhan fungsional pada table 3.2. Berikut dicantumkan tugas-tugas pengguna berdasarkan observasi pada PT. Royal Inti Mandiri Abadi.

Table 3.1 Kebutuhan Informasi Pengguna

No.	Pengguna	Tugas	<i>User Requirement</i>
1.	Customer Service	1.1. Melakukan input data pelanggan 1.2. Melakukan input data <i>sales</i> 1.3. Melakukan input data masalah 1.4. Melakukan input data harapan 1.5. Melakukan input data Kepala Perwalian atau Zona 1.6. Membuat jadwal kunjungan pelanggan 1.7. Melakukan Evaluasi laporan hasil kunjungan pelanggan	1.1. Dapat melakukan <i>input</i> data pelanggan (T1) 1.2. Dapat melakukan <i>input</i> data <i>sales</i> (T2) 1.3. Dapat melakukan <i>input</i> data masalah 1.4. Dapat melakukan <i>input</i> data harapan 1.5. Dapat melakukan input data Kepala Perwalian atau Zona (T3) 1.6. Mampu membuat jadwal kunjungan pelanggan (T4) 1.7. Dapat memberikan keputusan dari hasil

No.	Pengguna	Tugas	User Requirement
		1.8. Melakukan <i>monitoring</i> kinerja <i>sales</i> dan pelanggan	laporan kunjungan (T5) 1.8. Mampu <i>memonitor</i> kinerja <i>sales</i> dan pelanggan (T6)
2.	Kepala Perwalian atau Zona	2.1 Mendistribusikan jadwal kunjungan ke <i>sales</i> 2.2 Membuat laporan kunjungan pelanggan	2.1 Dapat mendistribusikan jadwal kunjungan pelanggan ke <i>sales</i> (T1) 2.2 Dapat membuat laporan kunjungan pelanggan (T2)

### 3.2.3 Functional Requirement

Table 3.2 Kebutuhan Fungsional

No.	Pengguna	User Requirement	Functional Requirement
1.	Customer Service	1.1. Dapat melakukan input data pelanggan (T1) 1.2. Dapat melakukan input data <i>sales</i> (T2) 1.3. Dapat melakukan input data Kepala Perwalian atau Zona (T3) 1.4. Dapat melakukan input data Masalah (T4) 1.5. Dapat melakukan input data harapan (T5) 1.6. Mampu membuat jadwal kunjungan pelanggan (T4) 1.7. Dapat memberikan keputusan dari hasil laporan kunjungan (T5)	1.1. Fungsi menyimpan data pelanggan (U1) 1.2. Fungsi menyimpan data <i>sales</i> (U2) 1.3. Fungsi menyimpan data kepala perwalian atau zona (U3) 1.4. Fungsi menyimpan data masalah (U4) 1.5. Fungsi menyimpan data harapan (U5) 1.6. Fungsi menyimpan data jadwal kunjungan (U4) 1.7. Fungsi proses pemberian keputusan dari hasil laporan kunjungan (T6)

No.	Pengguna	User Requirement	Functional Requirement
		1.8. Mampu <i>memonitor</i> kinerja <i>sales</i> dan pelanggan (T6)	1.8. Fungsi proses <i>monitoring</i> kinerja <i>sales</i> dan pelanggan
2.	Kepala Perwalian atau Zona	2.1 Dapat mendistribusikan jadwal kunjungan pelanggan ke <i>sales</i> (T1) 2.2 Dapat membuat laporan kunjungan pelanggan (T2)	2.1 Fungsi cetak jadwal kunjungan (U1) 2.2 Fungsi proses pencatatan laporan kunjungan pelanggan (U2)

### 3.3 Perancangan Sistem

Perancangan sistem ini bertujuan untuk mendefinisikan kebutuhan-kebutuhan fungsional yang menggambarkan aliran data dan alur sistem, dan sebagai tahap persiapan sebelum implementasi sistem. Perancangan sistem ini diharapkan dapat merancang dan mendesain sistem dengan baik, yang isinya meliputi langkah-langkah operasi dalam proses pengolahan data dan prosedur untuk mendukung operasi sistem.

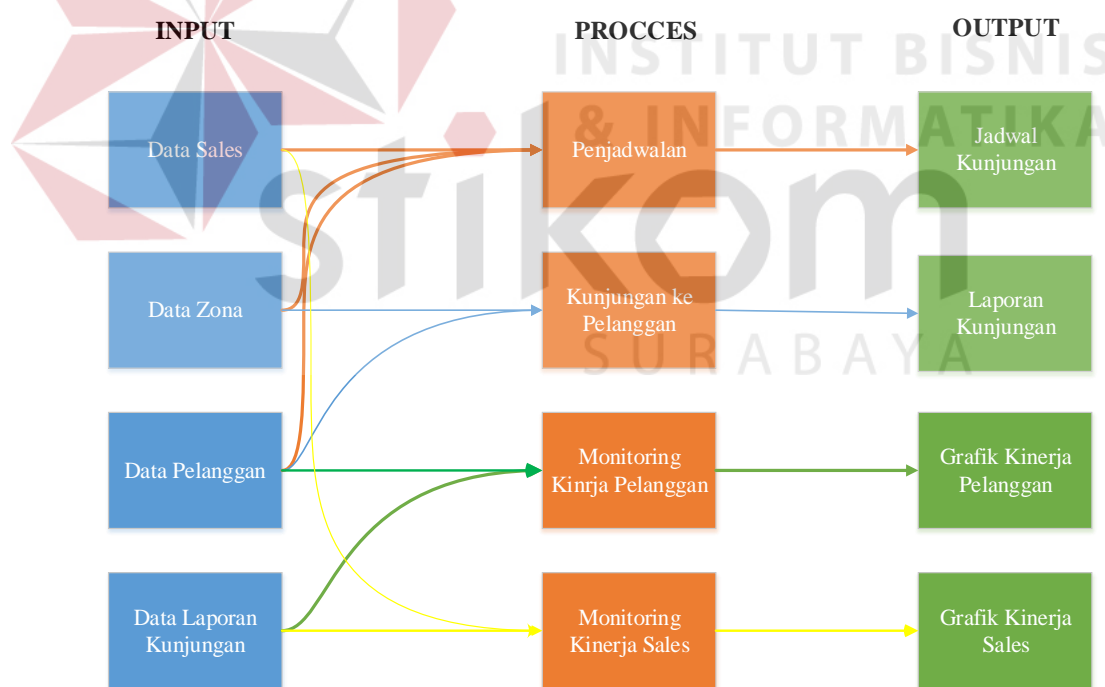
Langkah-langkah operasi dalam perancangan dan pengembangan sistem ini adalah sebagai berikut:

- a. Diagram *Block*
- b. Arsitektur Sistem
- c. *System Flow*
- d. Diagram Jenjang
- e. Data *Flow* Diagram
- f. *Conceptual Data Model* (CDM)

- g. *Physical Data Model (PDM)*
- h. Struktur Basis Data dan Tabel
- i. *Sitemap*
- j. Desain *User Interface*
- k. Desain *Input/Output*
- l. Rancangan Pengujian dan Evaluasi Sistem

### 3.3.1 Diagram Block

Aplikasi sistem penjadwalan dan *monitoring* yang dibangun mampu untuk membuat jadwal kunjungan, membuat laporan hasil kunjungan dan *memonitoring* kinerja *sales* dan pelanggan. Berikut merupakan bentuk diagram *block* aplikasi sistem penjadwalan dan *monitoring* yang akan dibuat sebagai berikut:



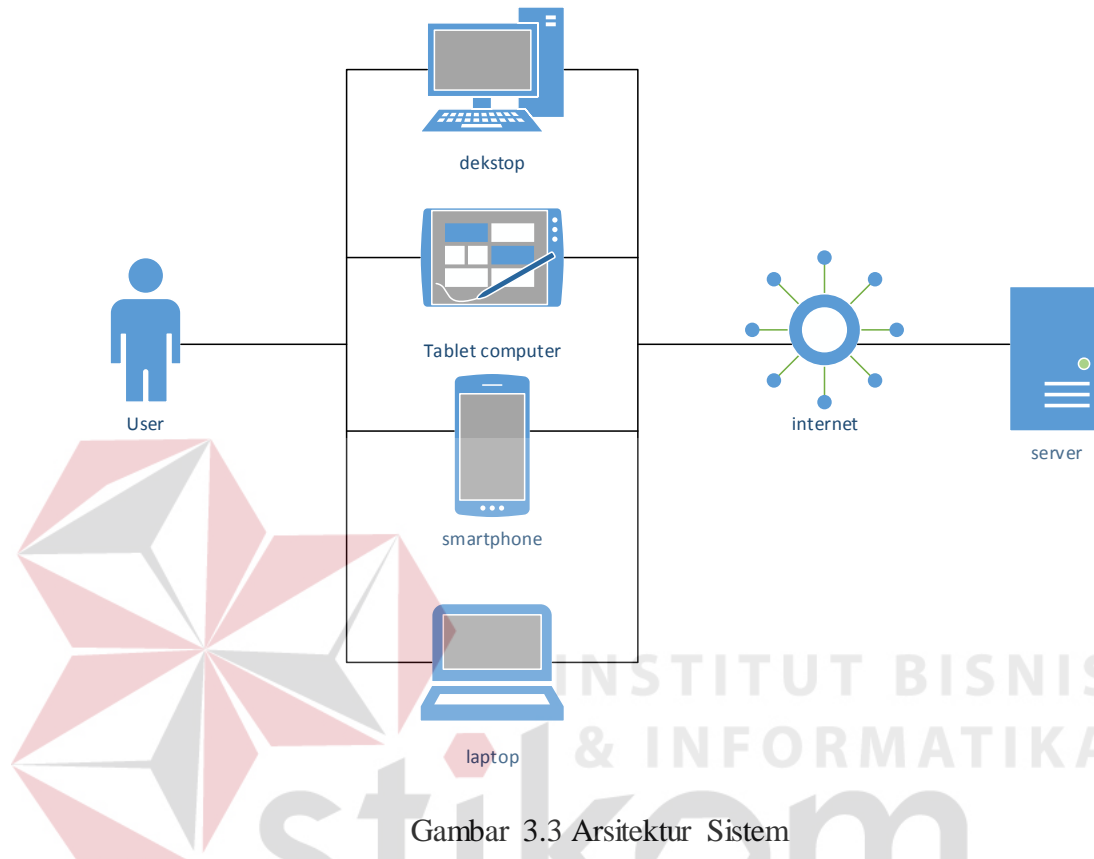
Gambar 3.2 Diagram *Block*

### 3.3.2 Arsitektur Sistem

Aplikasi Penjadwalan kunjungan dan monitoring pelanggan yang dibangun merupakan sistem berbasis web sehingga sistem dan data disimpan di data *center*



milik penyedia layanan. Arsitektur sistem yang akan dibangun adalah seperti pada gambar 3.3



Gambar 3.3 Arsitektur Sistem

Dengan arsitektur sistem berbasis web, pengguna tidak perlu melakukan pengelolaan dan perawatan infrastruktur sistem karena hal tersebut merupakan tanggung jawab dari penyedia layanan. Pengguna cukup mendaftarkan diri melalui halaman web penyedia layanan maka sistem informasi penjadwalan kunjungan dan *monitoring* pelanggan bisa langsung digunakan. Untuk mengakses sistem, pengguna perlu menyediakan perangkat keras seperti komputer, laptop, *tablet* atau *smartphone* yang tersambung dengan koneksi internet.

Model pengguna utama dari sistem didefinisikan sesuai fungsi dan tanggung jawab dapat dilihat pada table 3.3

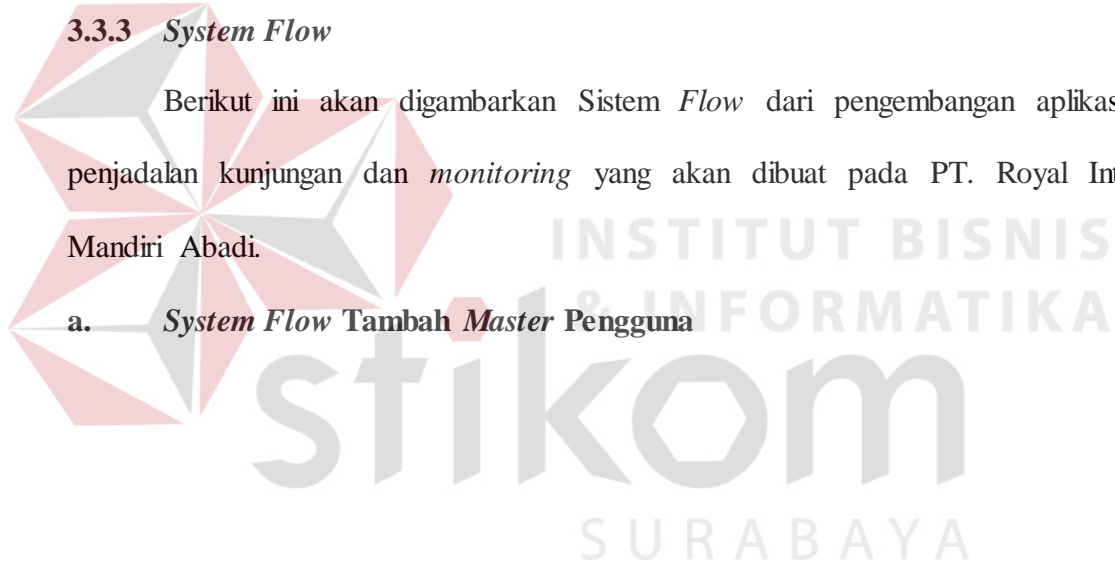
Tabel 3.3 Aktor Utama

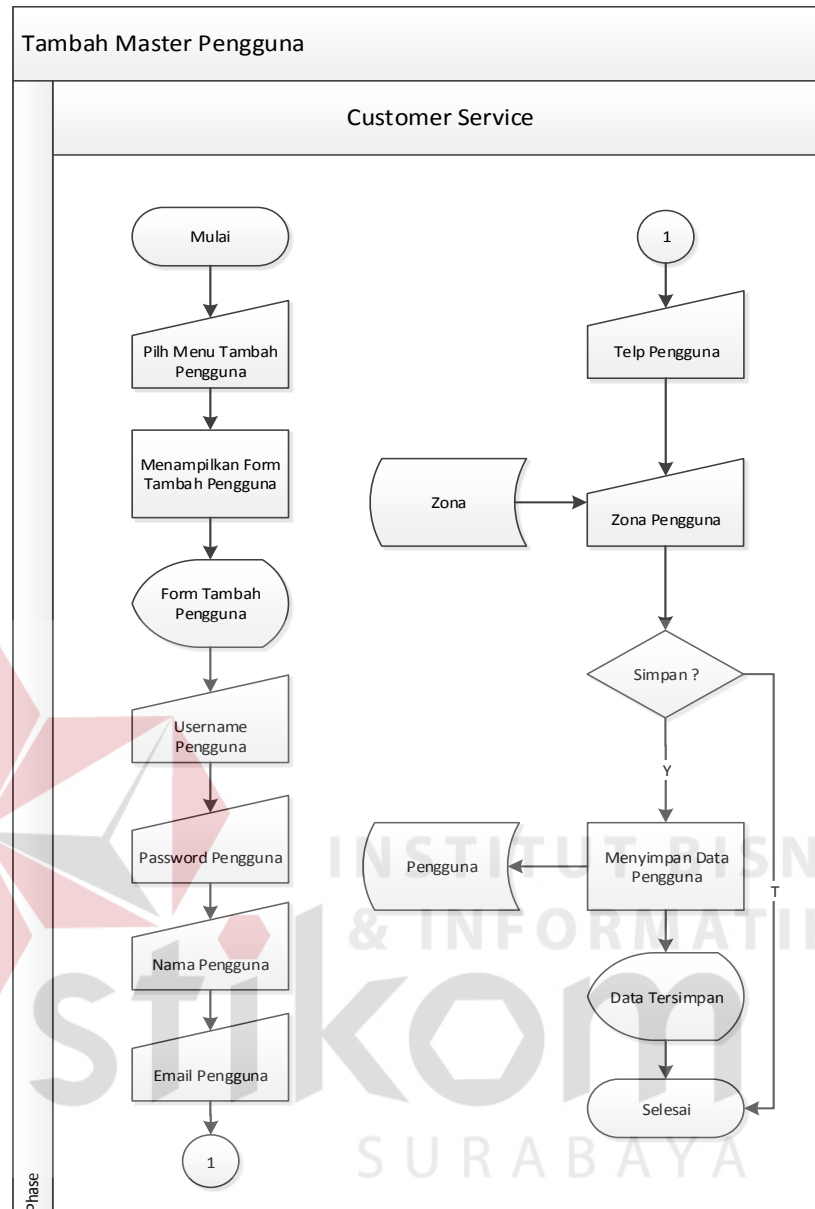
Pengguna	Fungsi
<b>Customer Service</b>	Mengelola jadwal kunjungan
	Mengelola data Sales beserta hak aksesnya
	Mengelola data master zona
	Mengelola data pelanggan
	Mengelola data masalah
	Mengelola data harapan
	Melakukan Monitoring Pelanggan dan <i>sales</i>
<b>Kepala Perwalian</b>	Membuat keputusan berdasarkan <i>history</i> pelanggan

### 3.3.3 System Flow

Berikut ini akan digambarkan Sistem *Flow* dari pengembangan aplikasi penjadalan kunjungan dan *monitoring* yang akan dibuat pada PT. Royal Inti Mandiri Abadi.

#### a. System Flow Tambah Master Pengguna

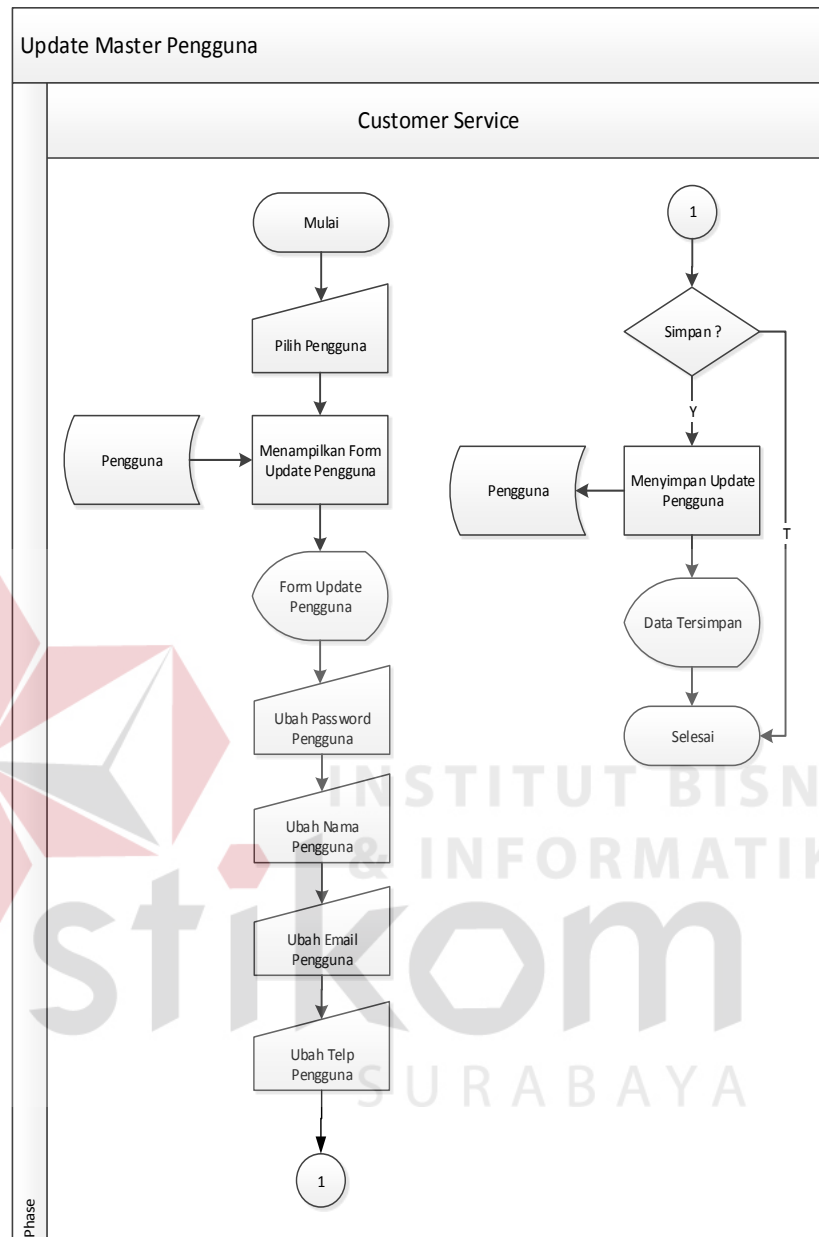




Gambar 3.4 Sistem Flow Tambah Master Pengguna

Pada sistem flow tambah master pengguna, alur dimulai dari customer service memilih menu tambah pengguna dan Sistem akan menampilkan form tambah pengguna setelah itu customer service mengisi inputan username pengguna, password, email, telepon, zona pengguna (diambil dari table zona yang ada didatabase). Data yang sudah diinputkan akan disimpan di dalam database table pengguna.

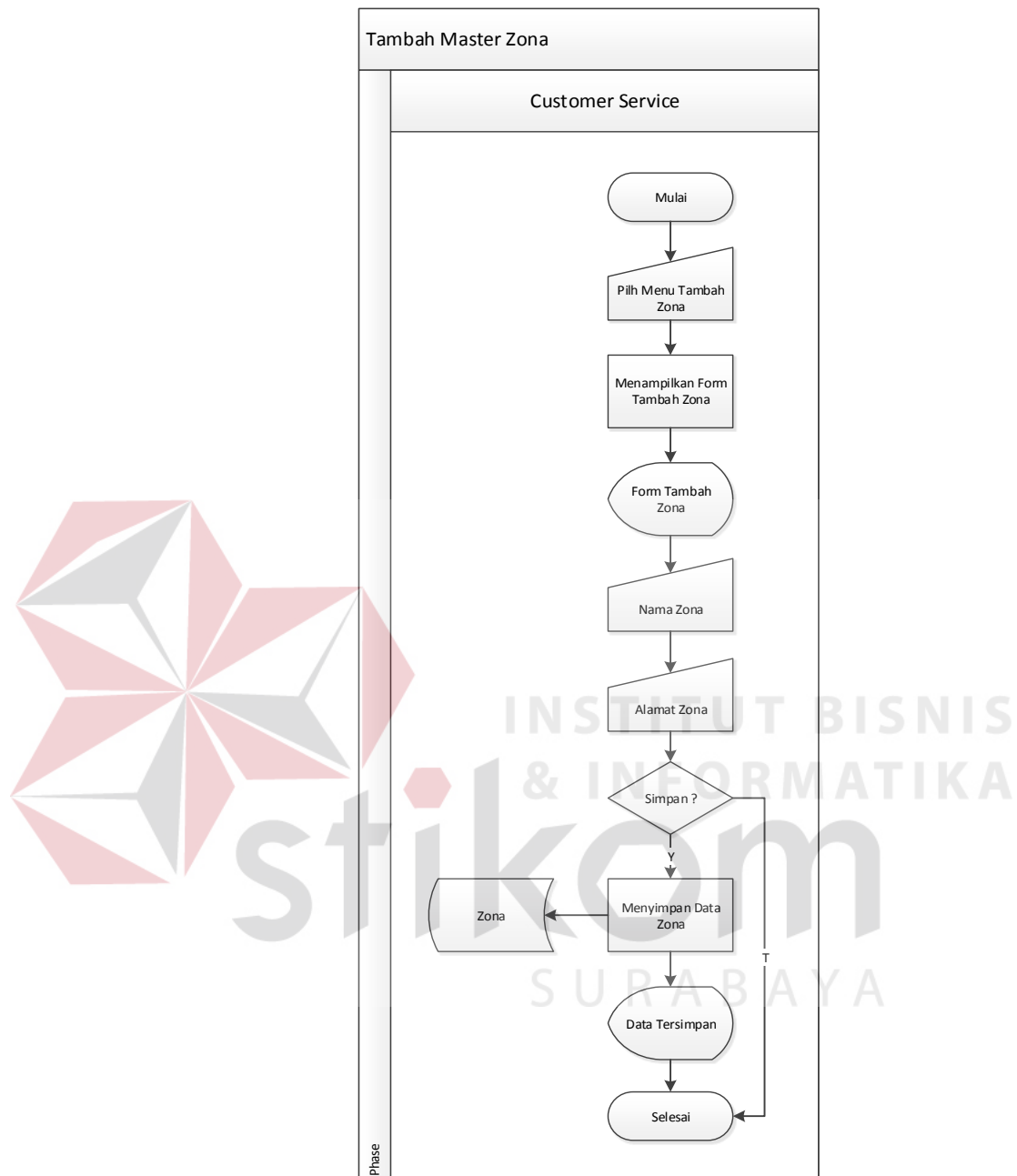
b. *System Flow Update Master Pengguna*



Gambar 3.5 *Sistem Flow Update Master Pengguna*

Pada *sistem flow update master pengguna*, alur dimulai dari customer service memilih menu *update* pengguna (diambil dari table pengguna yang ada didatabase) dan Sistem akan menampilkan form *update* pengguna setelah itu customer service mengubah inputan nama pengguna, *password*, *email*, telepon. Data yang sudah diupdate akan disimpan di dalam *database* table pengguna.

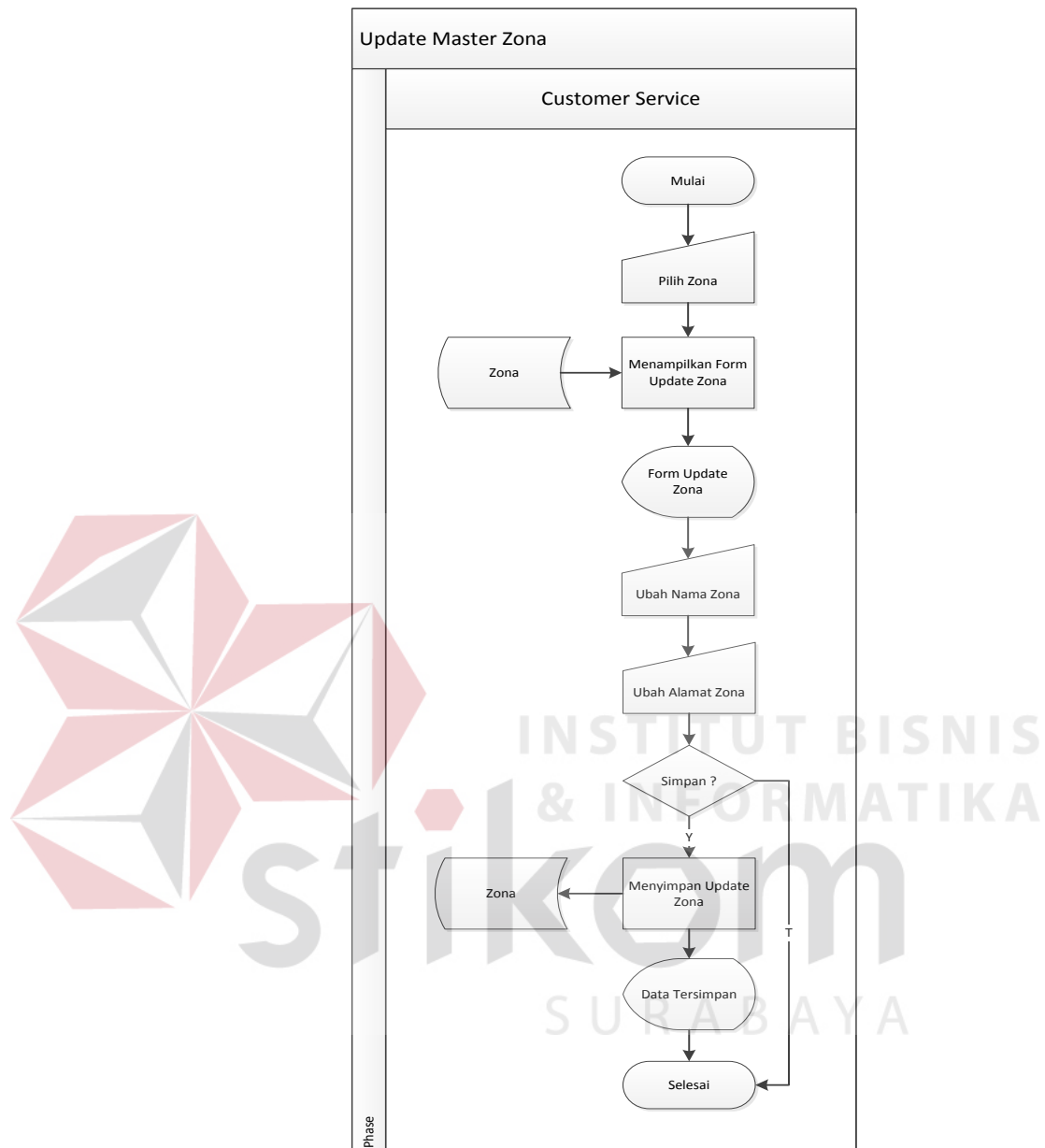
c. *System Flow Tambah Master Zona*



Gambar 3.6 *Sistem Flow Tambah Master Zona*

Pada sistem *flow* tambah *master zona*, alur dimulai dari *customer service* memilih menu tambah zona dan Sistem akan menampilkan *form* tambah zona setelah itu *customer service* mengisi inputan nama zona, alamat zona. Data yang sudah diisi akan disimpan di dalam *database table* zona.

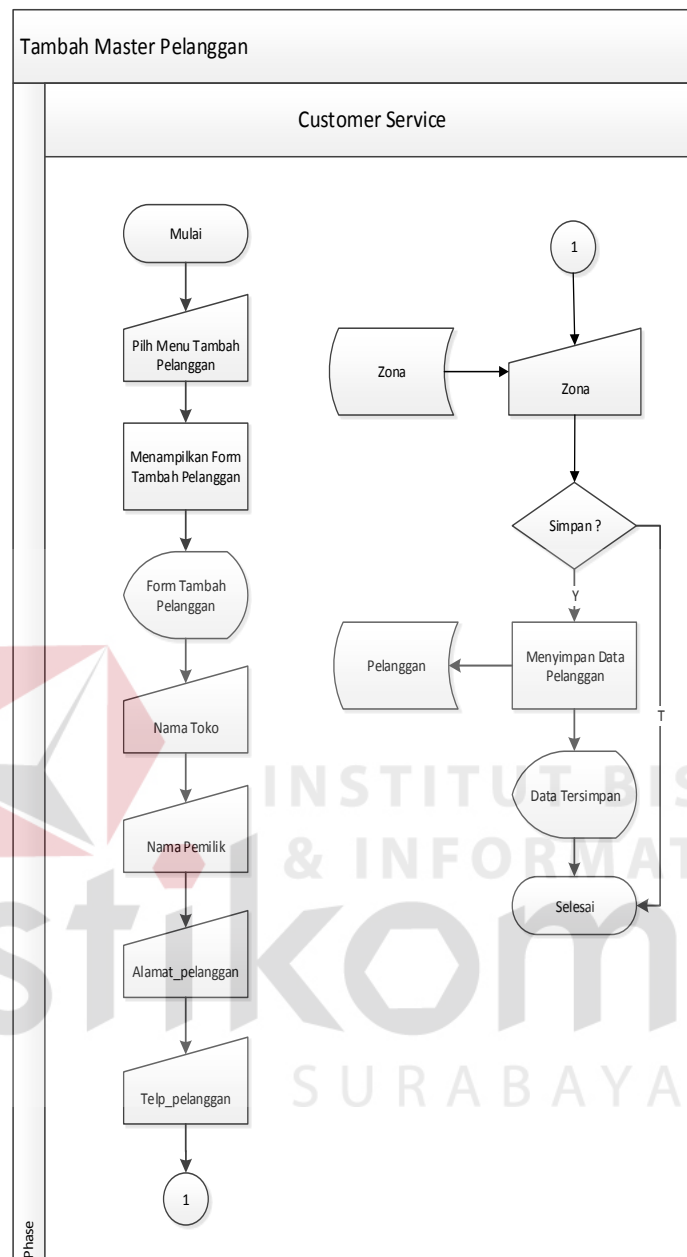
d. *System Flow Update Master Zona*



Gambar 3.7 *Sistem Flow Update Master Zona*

Pada *sistem flow update master zona*, alur dimulai dari *customer service* memilih menu *update zona* (diambil dari *table zona* yang ada *didatabase*) dan Sistem akan menampilkan form *update zona* setelah itu *customer service* mengubah inputan nama zona, alamat zona. Data yang sudah diupdate akan disimpan di dalam *database table zona*.

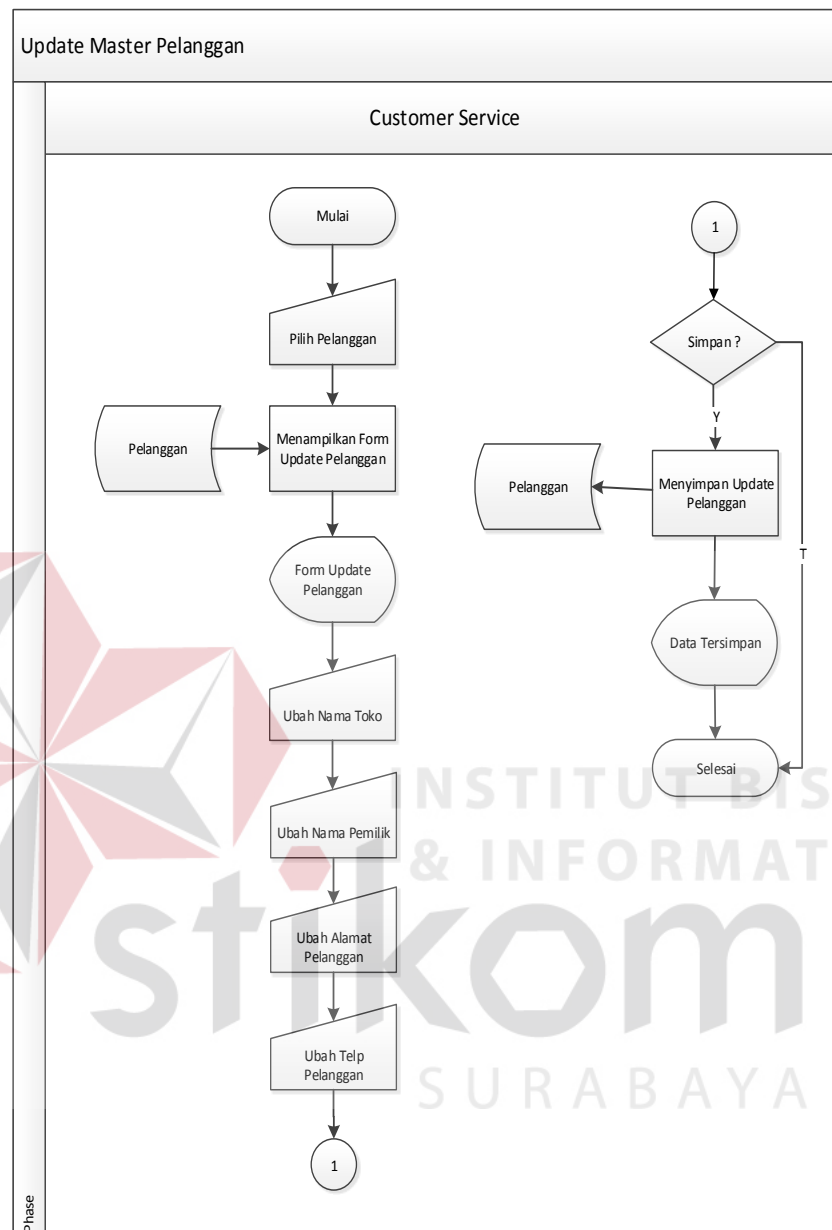
e. **System Flow Tambah Master Pelanggan**



Gambar 3.8 Sistem Flow Tambah Master Pelanggan

Pada sistem *flow* tambah *master* pelanggan, alur dimulai dari *customer service* memilih menu tambah pelanggan dan Sistem akan menampilkan *form* tambah pelanggan setelah itu *customer service* mengisi inputan nama toko, nama pemilik, alamat, telepon, zona (diambil dari *table* zona yang ada didatabase). Data yang sudah diinputkan akan disimpan di dalam *database table* pelanggan.

f. *System Flow Update Master Pelanggan*

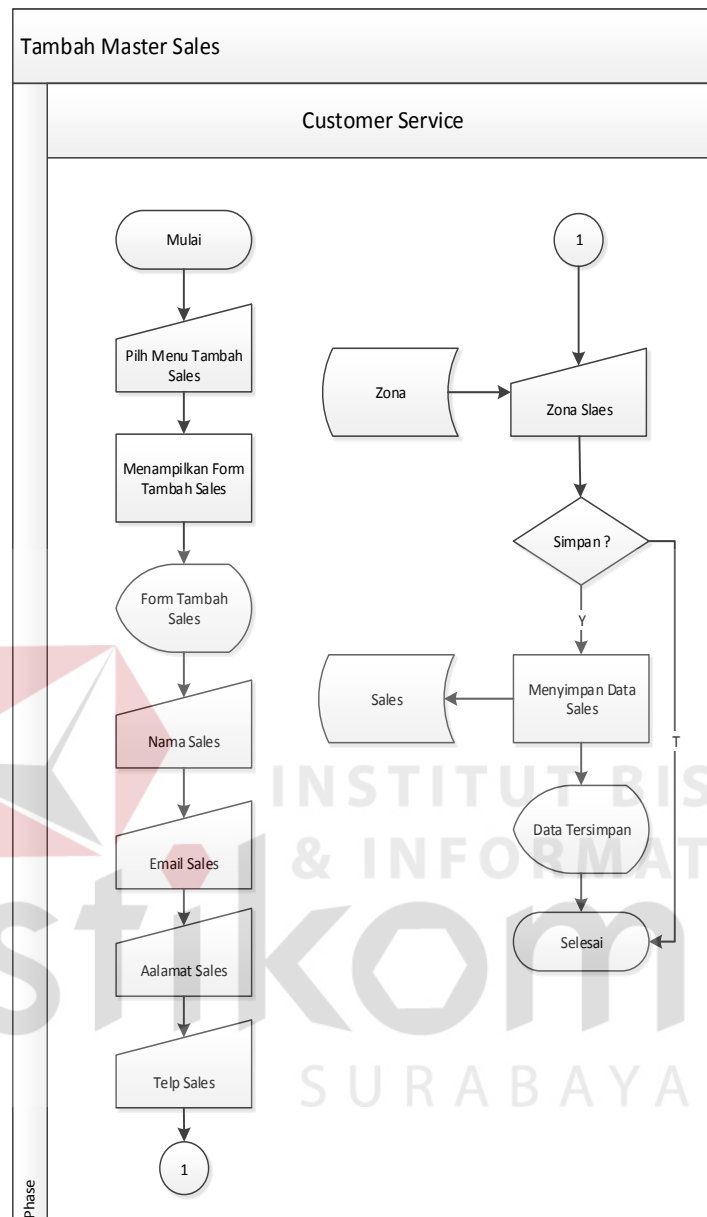


Gambar 3.9 *Sistem Flow Update Master Pelanggan*

Pada sistem *flow update* master pelanggan, alur dimulai dari *customer service* memilih menu *update* pelanggan (diambil dari *table* pelanggan yang ada *didatabase*) dan Sistem akan menampilkan *form update* pelanggan setelah itu *customer service* mengubah inputan nama toko, nama pemilik, alamat, telepon. Data yang sudah *diupdate* akan disimpan di dalam *database table* pelanggan.



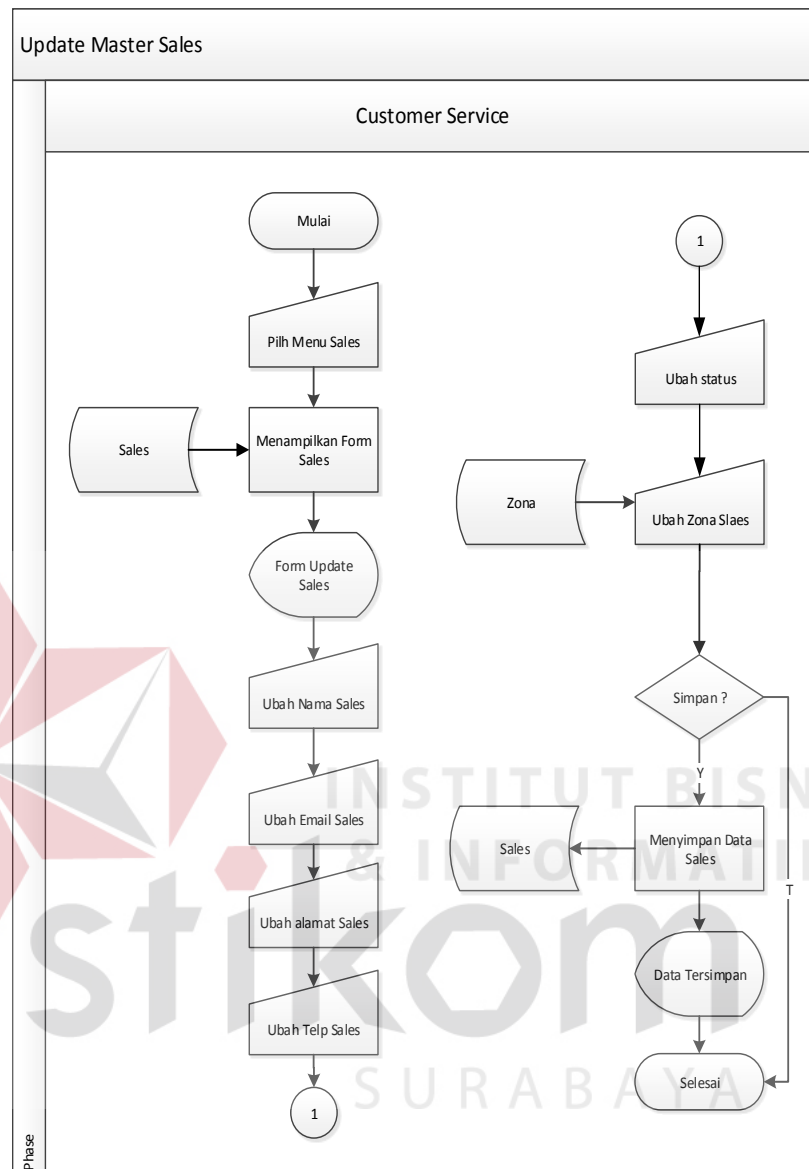
g. *System Flow Tambah Master Sales*



Gambar 3.10 *Sistem Flow Tambah Master Sales*

Pada sistem *flow* tambah *master sales*, alur dimulai dari *customer service* memilih menu tambah *sales* dan Sistem akan menampilkan form tambah *sales* setelah itu *customer service* mengisi inputan nama *sales*, *email*, alamat, telepon, zona (diambil dari *table* zona yang ada *didatabase*. Data yang sudah diinputkan akan disimpan di dalam *database table sales*.

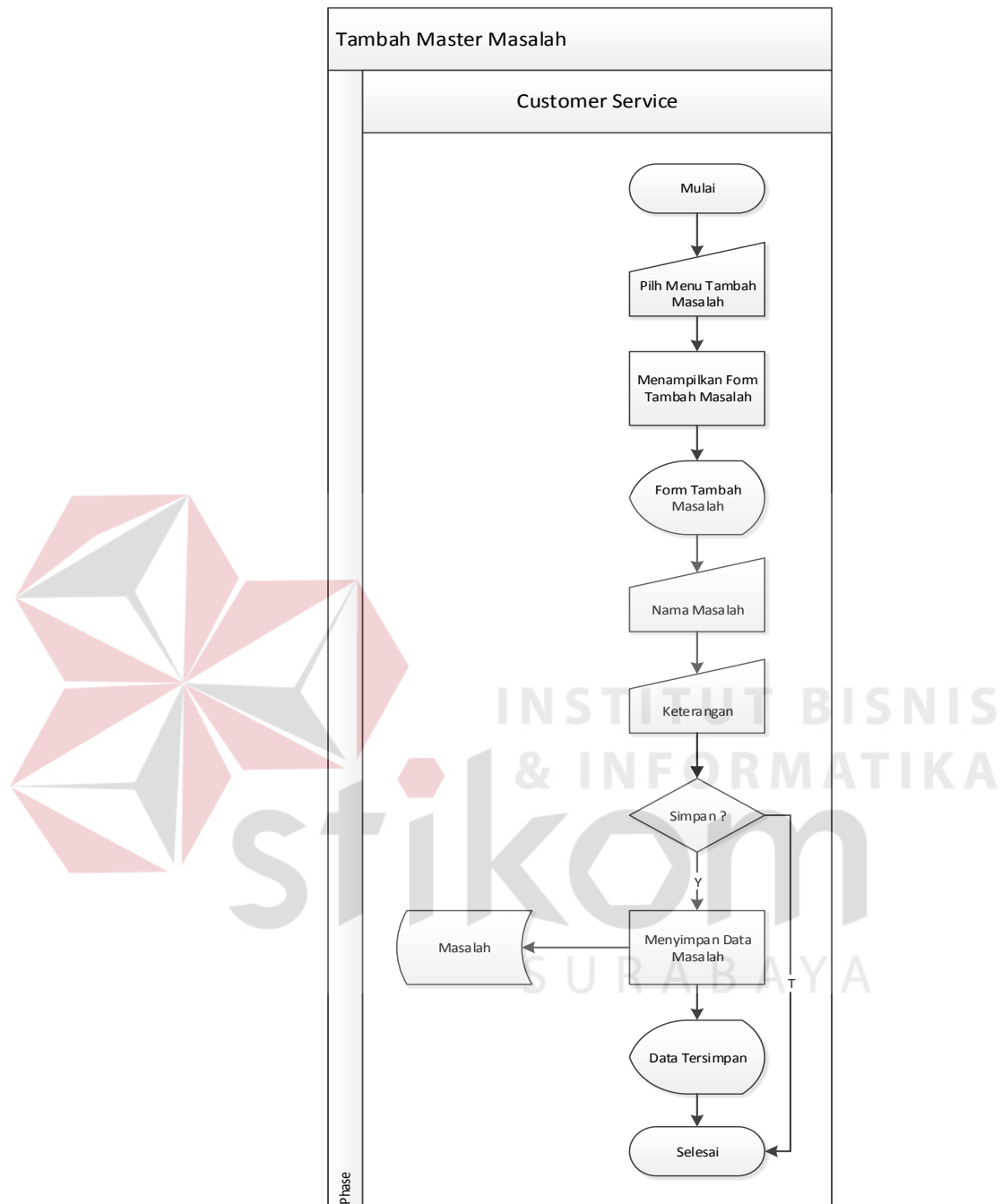
### h. System Flow Update Master Sales



Gambar 3.11 Sistem Flow Update Master Sales

Pada sistem *flow update master sales*, alur dimulai dari *customer service* memilih menu *update sales* (diambil dari *table sales* yang ada *didatabase*) dan Sistem akan menampilkan *form update sales* setelah itu *customer service* mengubah inputan nama *sales*, *email*, alamat, telepon, zona (diambil dari *table zona* yang ada *didatabase*), status. Data yang sudah *diupdate* akan disimpan di dalam database *table sales*.

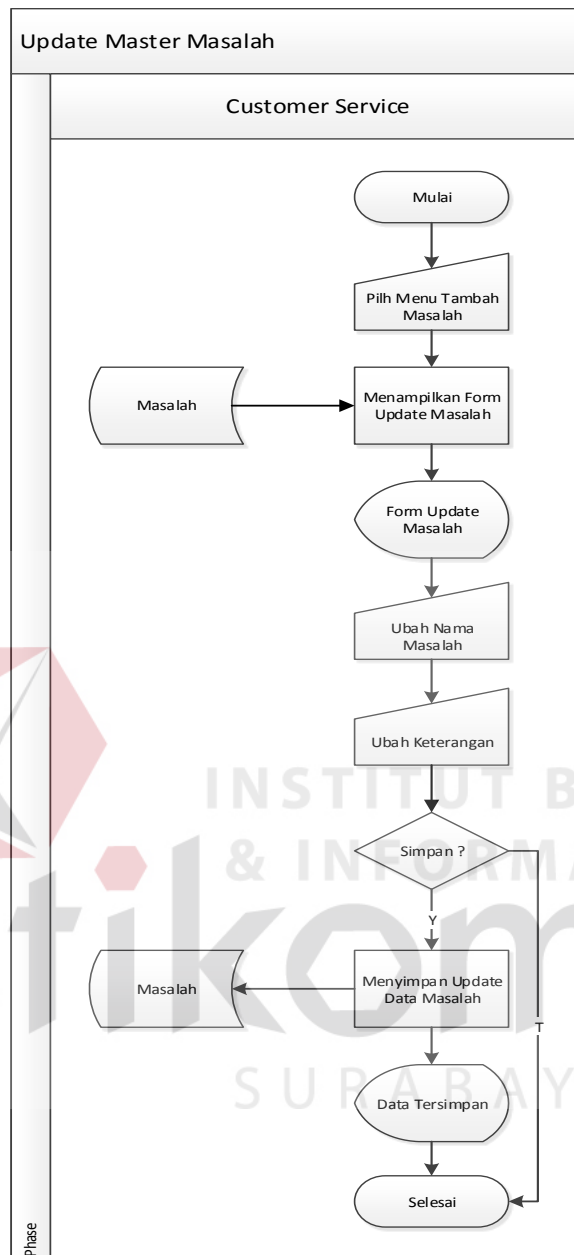
i. *System Flow Tambah Master Masalah*



Gambar 3.12 *Sistem Flow Tambah Master Zona*

Pada sistem *flow* tambah *master* Masalah, alur dimulai dari *customer service* memilih menu tambah masalah dan Sistem akan menampilkan *form* tambah masalah setelah itu *customer service* mengisi inputan nama masalah, keterangan. Data yang sudah diisi akan disimpan di dalam *database table* masalah.

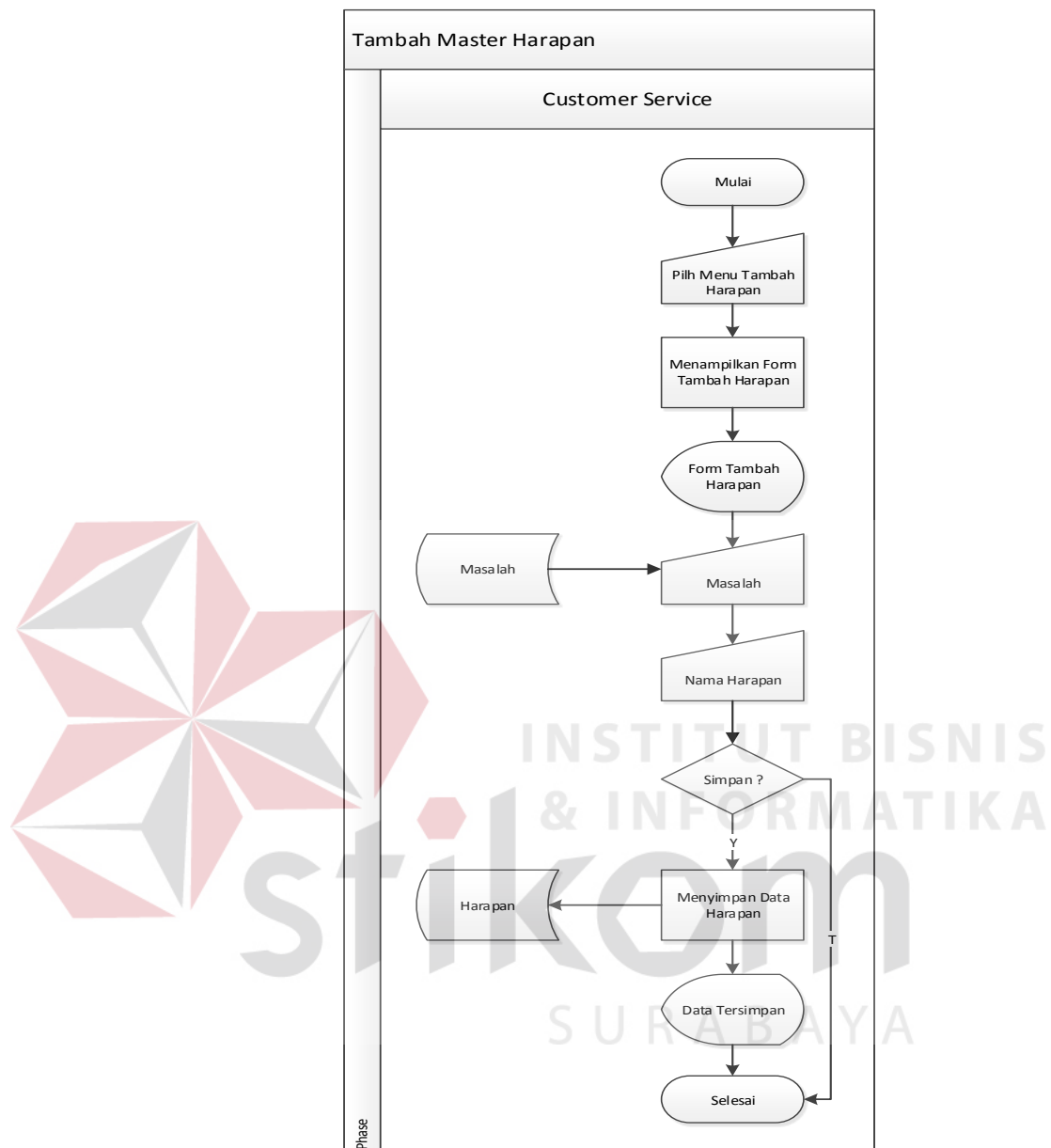
j. *System Flow Update Master Masalah*



Gambar 3.13 Sistem Flow Update Master Masalah

Pada sistem flow update master masalah, alur dimulai dari customer service memilih menu *update* masalah (diambil dari *table* masalah yang ada *didatabase*) dan Sistem akan menampilkan form *update* masalah setelah itu customer service mengubah inputan nama masalah dan keterangan. Data yang sudah diupdate akan disimpan di dalam *database table* masalah.

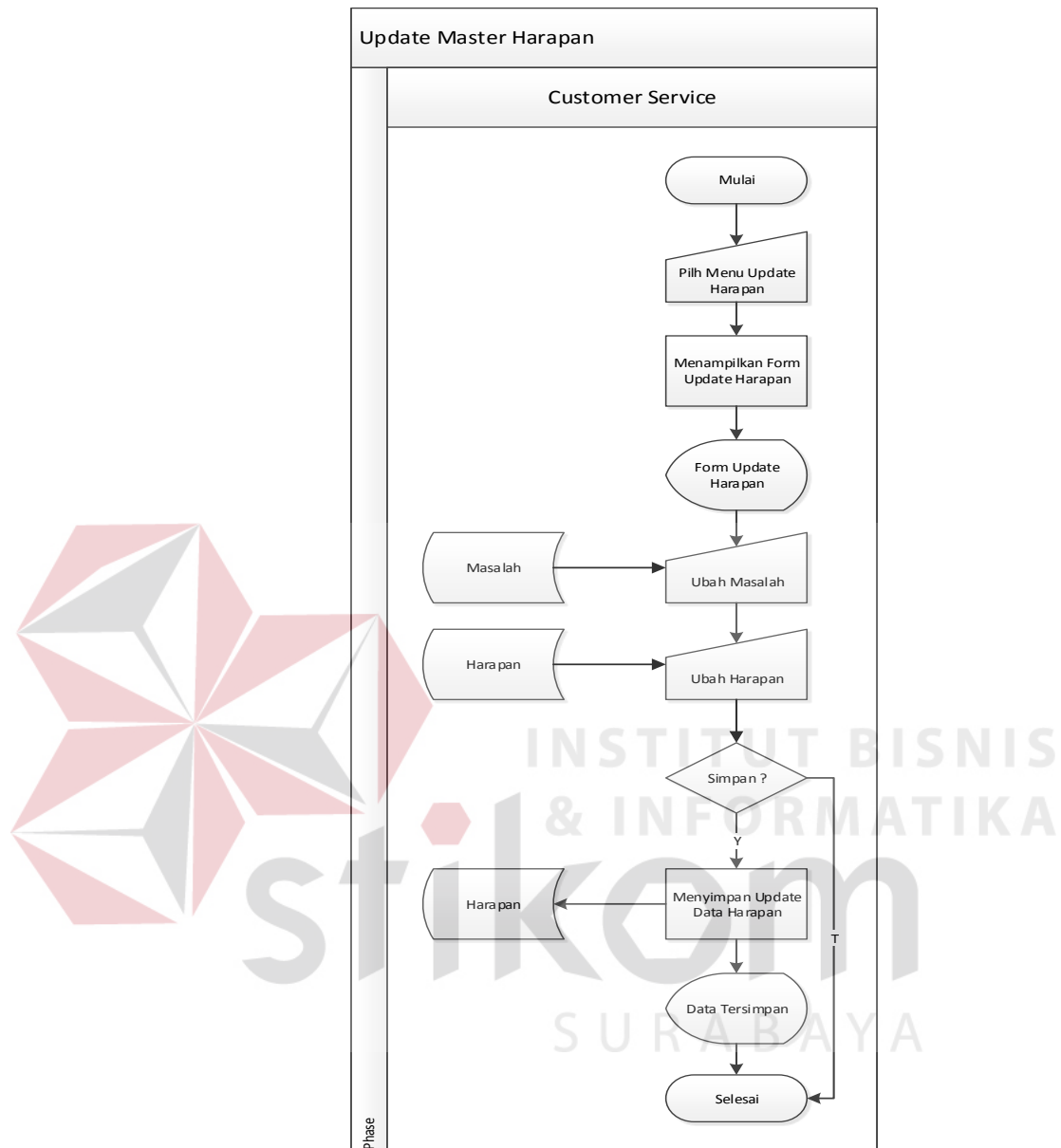
k. *System Flow Tambah Master Harapan*



Gambar 3.14 *Sistem Flow Tambah Master Harapan*

Pada sistem *flow* tambah *master* harapan, alur dimulai dari *customer service* memilih menu tambah harapan dan sistem akan menampilkan *form* tambah harapan setelah itu *customer service* mengisi *inputan* nama masalah (diambil dari *table* masalah yang ada *didatabase*) dan nama harapan. Data yang sudah diisi akan disimpan di dalam *database table* harapan.

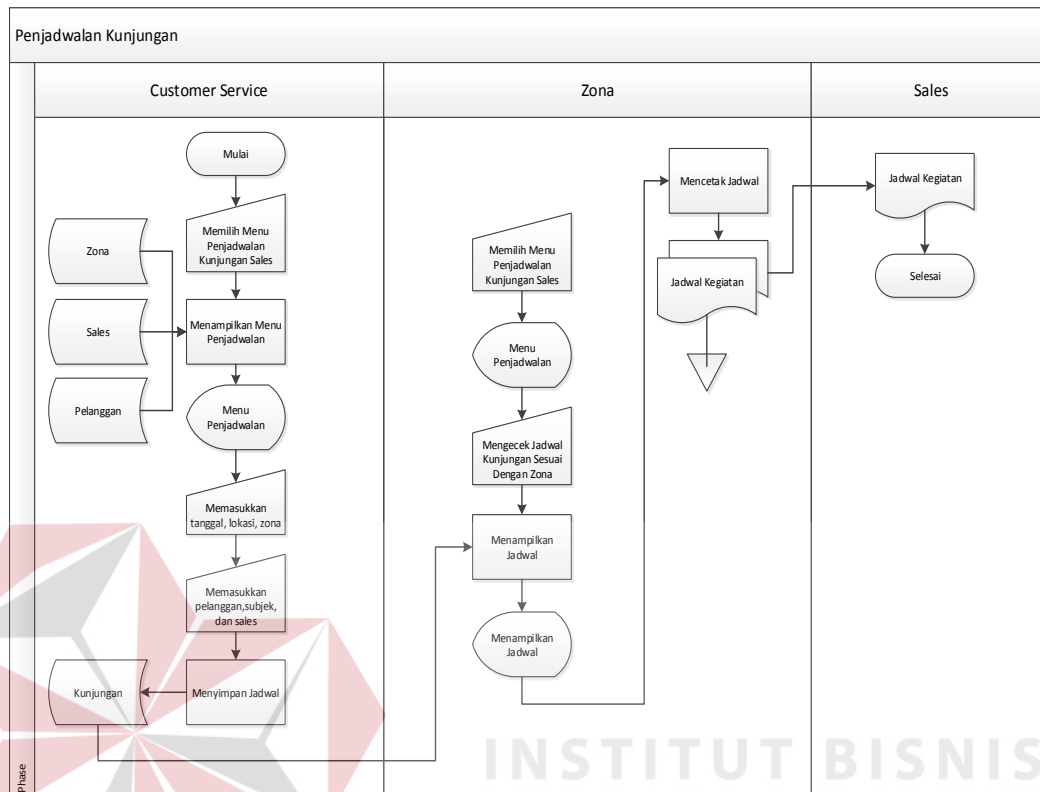
### I. *System Flow Update Master Harapan*



Gambar 3.15 *Sistem Flow Update Master Harapan*

Pada *sistem flow update master harapan*, alur dimulai dari *customer service* memilih menu *update* harapan (diambil dari *table* harapan yang ada *didatabase*) dan sistem akan menampilkan form *update* harapan setelah itu *customer service* mengubah masalah dan harapan. Data yang sudah diupdate akan disimpan di dalam *database table* harapan.

### m. System Flow Penjadwalan Kunjungan

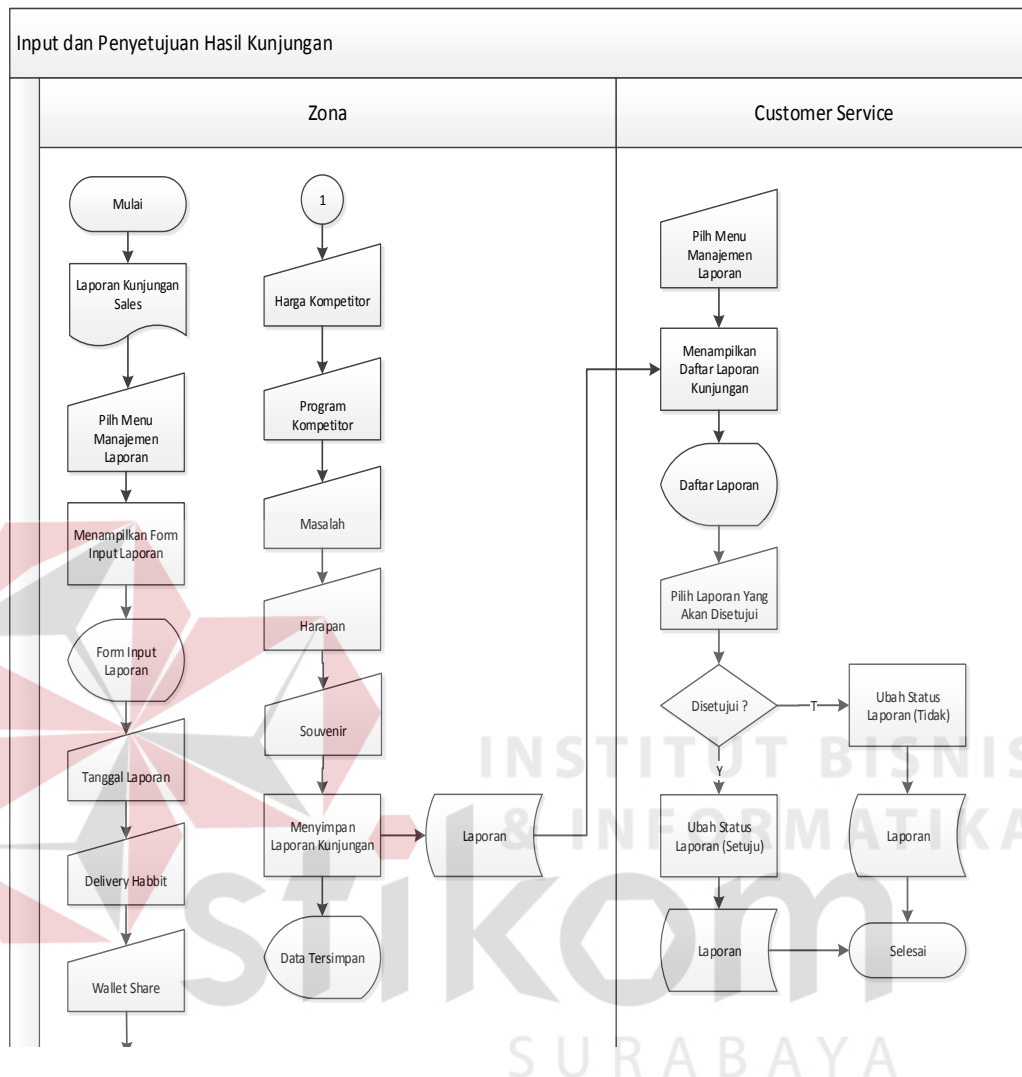


Gambar 3.16 Sistem Flow Penjadwalan Kunjungan

Pada sistem *flow* penjadwalan kunjungan, alur dimulai dari *customer service* memilih menu penjadwalan kunjungan (mengambil data dari *table* zona, *sales*, pelanggan yang ada *didatabase*) dan sistem akan menampilkan *form* penjadwalan setelah itu *customer service* mengisi *inputan* tanggal, lokasi, zona, pelanggan, agenda, *sales*. Data yang sudah diinputkan akan disimpan di dalam database *table* kunjungan.

Jadwal kunjungan yang sudah dibuat akan didistribusikan kepada kepala perwalian atau Zona. Kepala perwalian akan mengecek jadwal kunjungan dan mencetak dua rangkap jadwal kunjungan setelah itu jadwal kunjungan yang sudah dicetak akan dibagikan kepada *sales* yang ada di masing-masing zona.

n. **System Flow Input dan Penyetujuan Hasil Kunjungan**



Gambar 3.17 Sistem Flow Input dan Penyetujuan Hasil Kunjungan

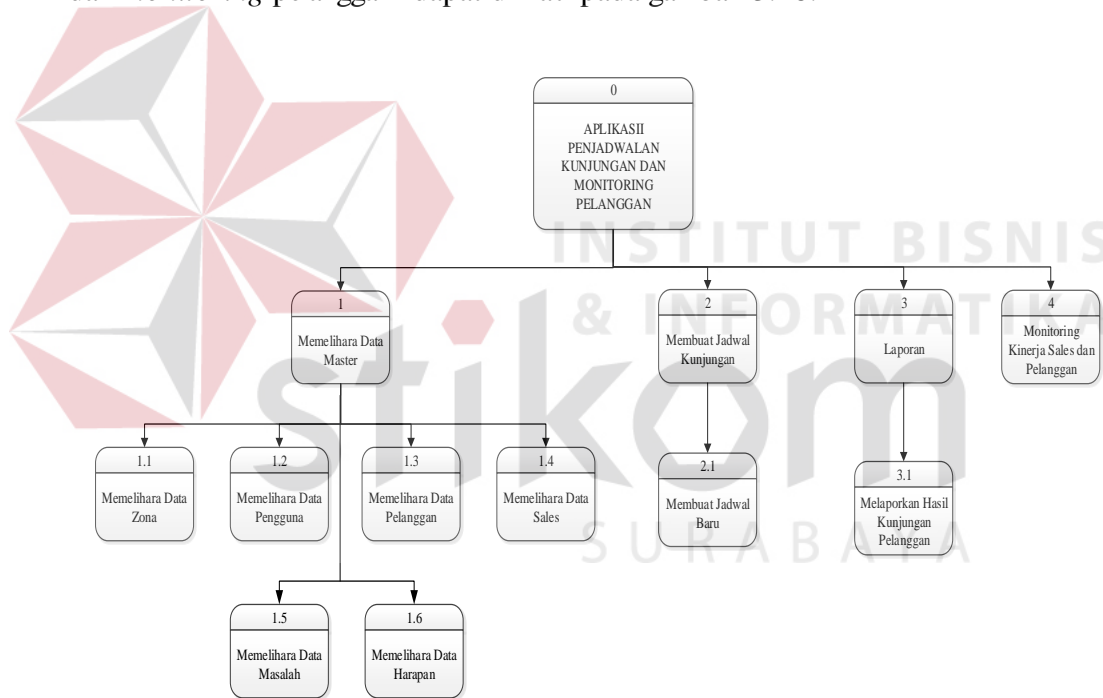
Pada sistem *flow input* dan penyetujuan hasil kunjungan, alur dimulai dari Kepala Pewalian atau zona mendapatkan dokumen laporan kunjungan sales setelah itu kepala perwalian memilih menu manajemen laporan dan Sistem akan menampilkan form manajemen laporan setelah itu kepala perwalian mengisi inputan tanggal laporan, delivery habit, wallet share, harga kompetitor, program kompetitor, masalah, harapan, *souvenir*. Data yang sudah diinputkan akan disimpan di dalam database table laporan.



Laporan kunjungan yang sudah dibuat akan didistribusikan kepada Customer Service. Customer Service akan mengecek daftar laporan kunjungan dan memilih laporan yang akan disetujui atau tidak disetujui. Hasil laporan akan disimpan kedalam tabel laporan yang ada di *database*

### 3.3.4 Diagram Jenjang

Diagram berjenjang merupakan alat perancangan sistem yang dapat menampilkan seluruh proses yang terdapat pada suatu aplikasi tertentu dengan jelas dan terstruktur. Secara garis besar diagram jenjang aplikasi penjadwalan kunjungan dan *monitoring* pelanggan dapat dilihat pada gambar 3.18.

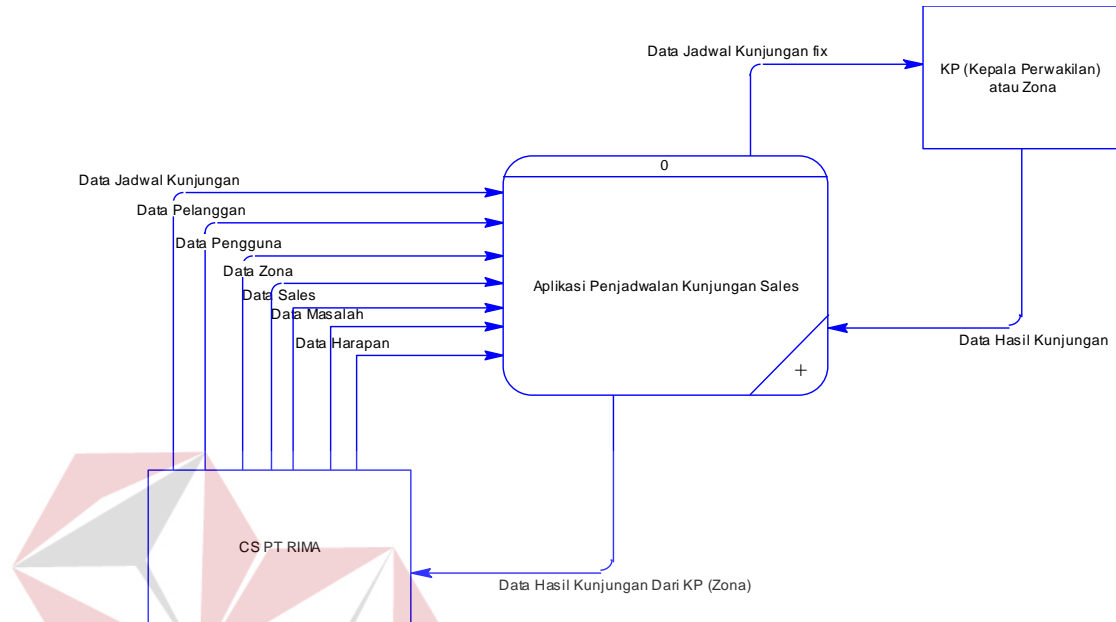


Gambar 3.18 Diagram Jenjang Aplikasi Penjadwalan Kunjungan dan *Monitoring* Pelanggan

### 3.3.5 Context Diagram

*Context* diagram ini menggambarkan proses sistem secara umum pada sistem informasi penanganan keluhan pelanggan pada PT. Royal Inti Mandiri

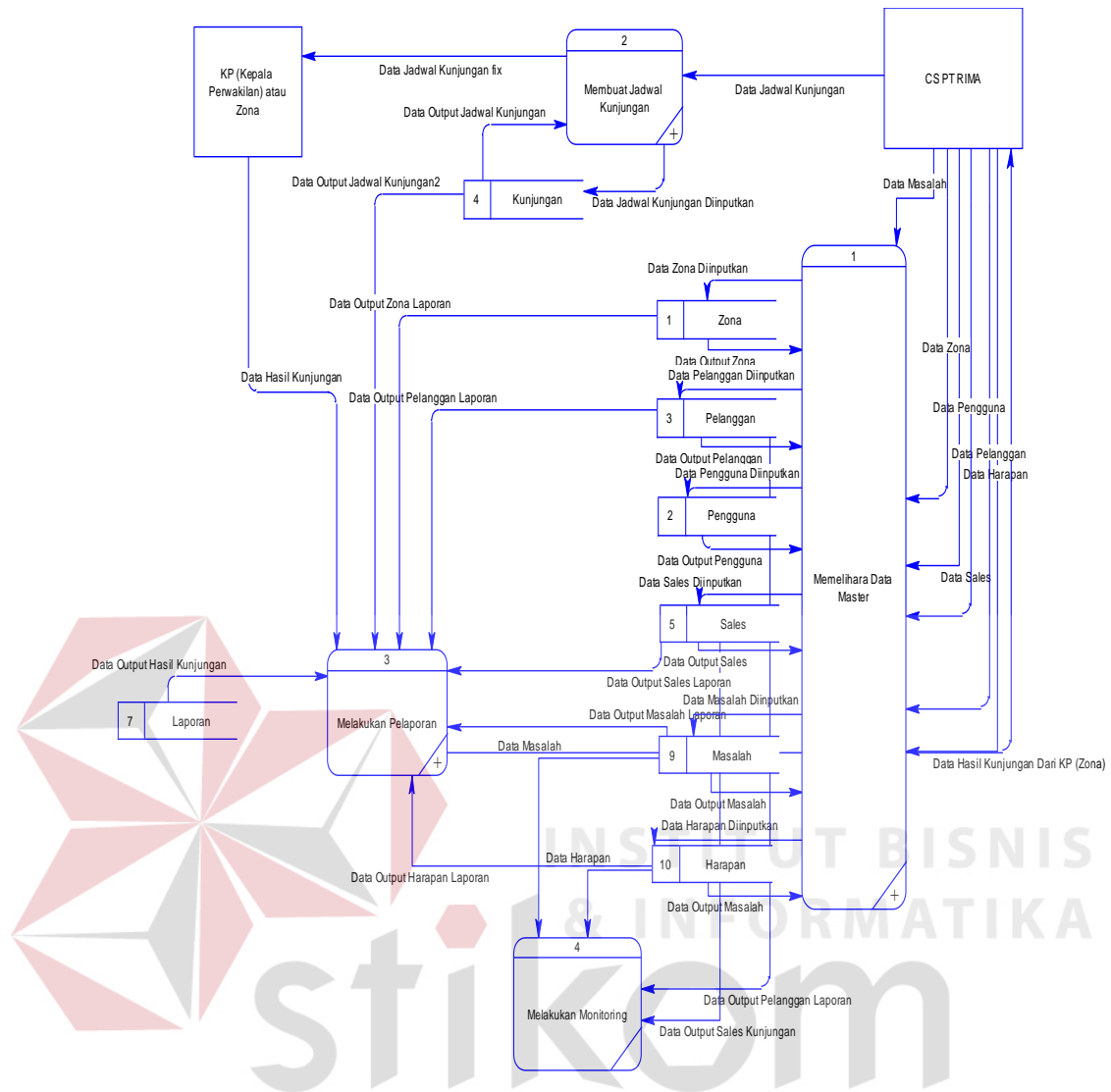
Abadi. Dalam *context* diagram ini melibatkan 2 entity yaitu: Customer Service dan KP (Kepala Perwakilan) atau zona.



Gambar 3.19 *Context* Diagram Aplikasi Penjadwalan Kunjungan dan *Monitoring* Pelanggan

### 3.3.6 DFD-Level 0

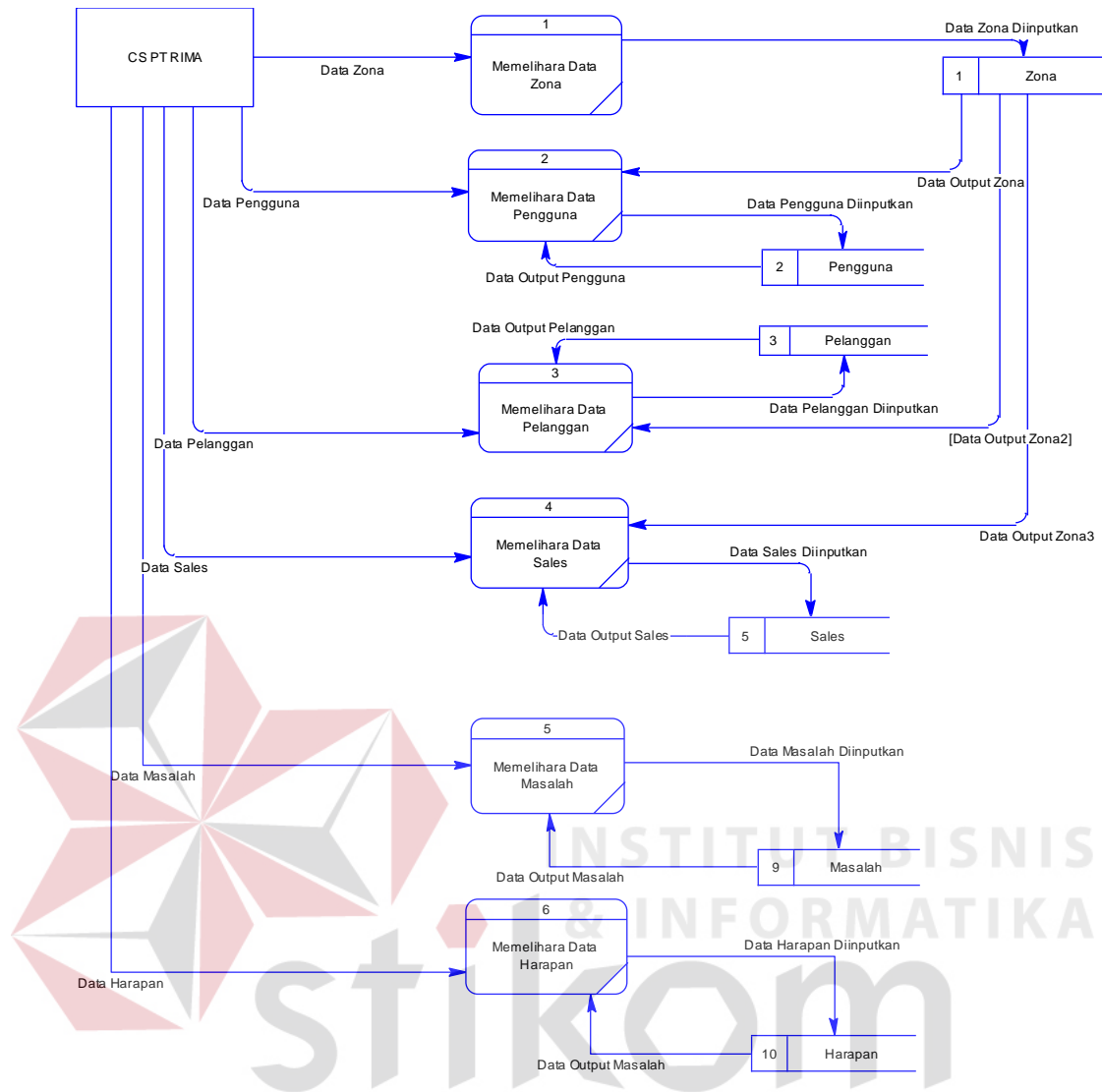
Gambaran sistem pada DFD level 0 merupakan hasil *decompose* dari *context diagram*, pada saat pembuatan DFD Level 0 terdapat pengembangan-pengembangan dari *context diagram*, berikut DFD level 0 pada gambar 3.20



Gambar 3.20 Diagram Level 0 Aplikasi Penjadwalan Kunjungan dan *Monitoring*

### 3.3.7 DFD Level 1-Proses Memelihara Data Master

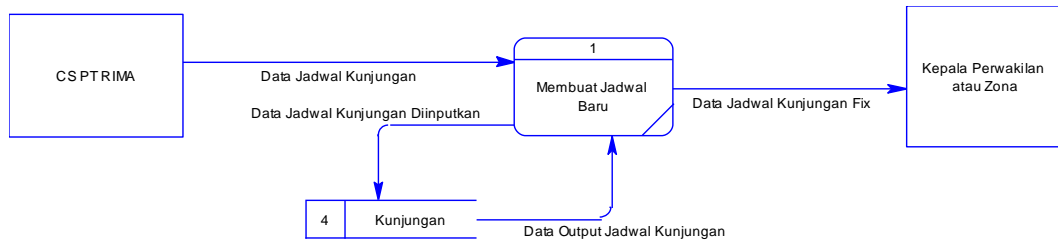
Gambaran sistem pada DFD level 1 merupakan pengembangan dari DFD level 0 dalam proses memelihara data master. Berikut ini DFD Level 1 Proses memelihara data master pada gambar 3.21.



Gambar 3.21 Diagram Level 1 Proses Memelihara Data Master

### 3.3.8 DFD Level 1-Proses Membuat Jadwal Kunjungan

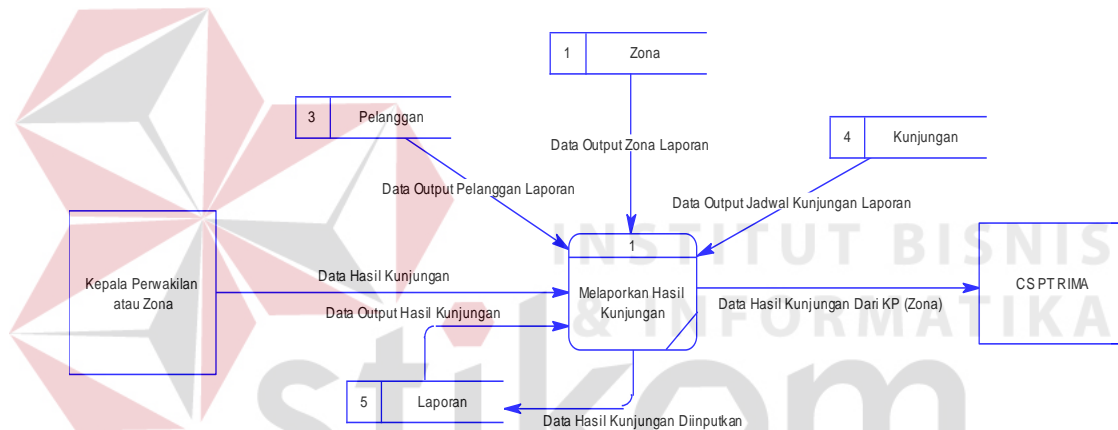
Gambaran sistem pada DFD level 1, DFD level 1 merupakan pengembangan dari DFD Level 0 tentang proses membuat jadwal kunjungan, berikut DFD level 1 tentang proses membuat jadwal kunjungan dapat dilihat pada gambar 3.22



Gambar 3.22 Diagram Level 1 Proses Membuat Jadwal Kunjungan

### 3.3.9 DFD Level 1-Proses Melakukan Pelaporan

Gambaran sistem pada DFD level 1, DFD level 1 merupakan pengembangan dari DFD Level 0 Proses melakukan pelaporan, berikut DFD level 1 tentang proses melakukan pelaporan dapat dilihat pada gambar 3.23.

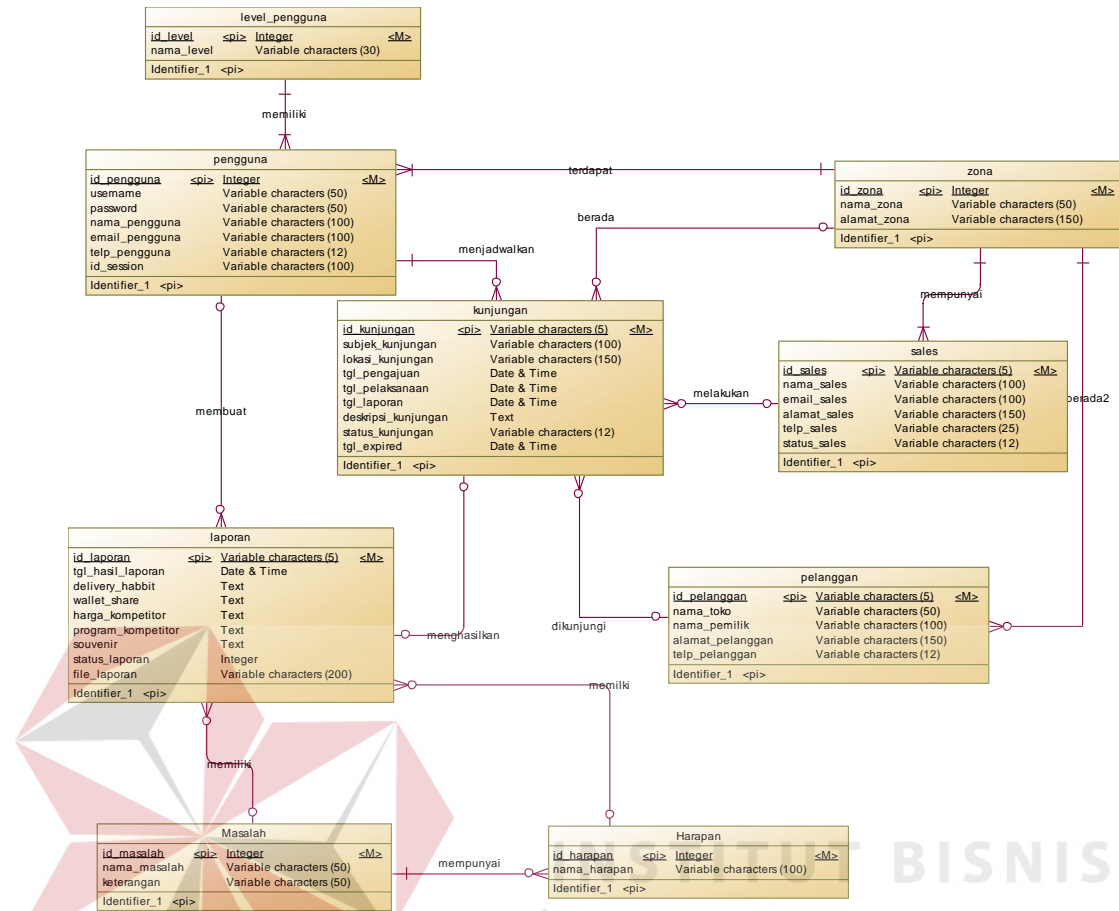


Gambar 3.23 Diagram Level 1 Proses Melakukan Pelaporan

### 3.3.10 Conceptual Data Model (CDM)

*Conceptual Data Model (CDM)* adalah gambaran secara keseluruhan struktur aplikasi. Dengan CDM kita bisa membangun desain awal sistem dan tidak perlu khawatir dengan detail implementasinya secara fisik. Dan melalui prosedur *generation* yang mudah, kita bisa melakukan *generate* CDM ke *Physical Data Model (PDM)*.

Bentuk *Conceptual Data Model* dari Aplikasi penjadwalan kunjungan dan *monitoring* pelanggan pada PT. Royal Inti Mandiri Abadi adalah sebagai berikut

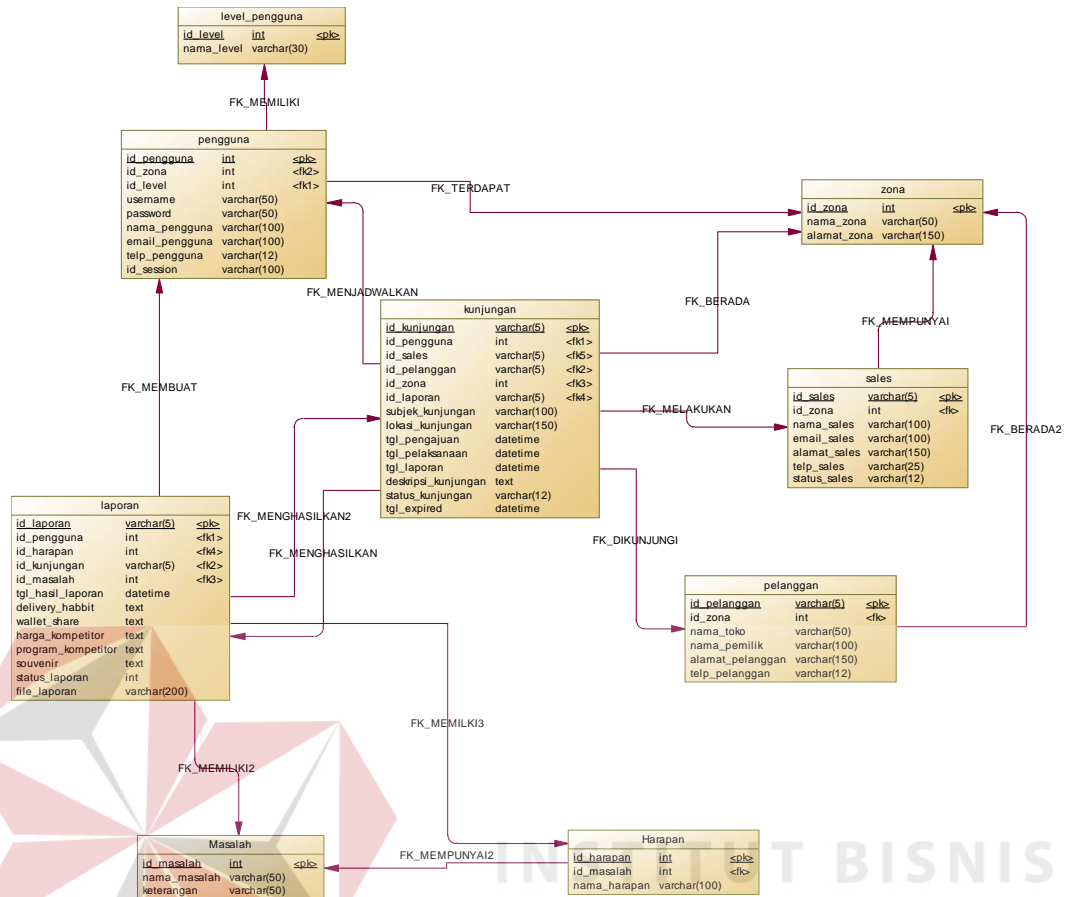


Gambar 3.24 Conceptual Data Model

### 3.3.11 Physical Data Modeling

*Physical Data Model* menggambarkan struktur data sebagaimana akan di implementasikan oleh DBMS. Dalam PDM kita bisa mengoptimalkan *database* dengan memodifikasi tabel, kolom, *index*, *refrential integrity*, *view*, *physical storage*, *trigger* and *stored procedure*. *Procedure database generation* menerapkan hal itu dengan cara menyesuaikan dengan DBMS yang kita pilih.

Bentuk *Physical data model* dari *generate conceptual data model* untuk Aplikasi penjadwalan kunjungan dan *monitoring* pelanggan pada PT. Royal Inti Mandiri Abadi adalah sebagai berikut:



Gambar 3.25 Physical Data Modeling

### 3.3.12 Struktur Basis Data dan Tabel

Struktur tabel digunakan dalam pembuatan sistem informasi penjadwalan kunjungan. Data-data dibawah ini akan menjelaskan satu per satu secara detil dari struktur tabel sistem.

#### a. Nama Tabel : Level Pengguna

Primary Key : ID\_Pengguna

Foreign Key : -

Fungsi : Untuk menyimpan level pengguna yang akan mengakses ke sistem informasi manajemen keluhan.

Tabel 3.4 Level Pengguna

No.	Name	Type	Size	Keterangan
1.	ID_Jabatan	Integer	-	Primary Key
2.	Nama_Level	Varchar	30	-

**b. Nama Tabel : Pengguna**

Primary Key : ID\_pengguna

Foreign Key : ID\_Level, ID\_Zona

Fungsi : Untuk menyimpan data pengguna yang akan mengakses ke dalam sistem.

Tabel 3.5 Pengguna

No.	Name	Type	Size	Keterangan
1.	ID_Pengguna	Integer	-	Primary Key
2.	ID_Level	Integer	-	Foreign Key
3.	ID_Zona	Integer	-	Foreign Key
4.	Username	Varchar	50	-
3.	Password	Varchar	50	-
4.	Nama_Pengguna	Varchar	100	-
5.	Email_Pengguna	Varchar	100	-
6.	Telp_Pengguna	Varchar	12	-
7.	ID_Sesion	Varchar	100	-

**c. Nama Tabel : Kunjungan**

Primary Key : ID\_Kunjungan

Foreign Key : ID\_Pengguna, ID\_Pelanggan, ID\_Zona, ID\_Laporan

Fungsi : Untuk menyimpan jadwal kunjungan yang ditujukan kepada masing-masing zona.

Tabel 3.6 Kunjungan

No.	Name	Type	Size	Keterangan
1.	ID_Kunjungan	Varchar	5	Primary Key
2.	ID_Pengguna	Integer	11	Foreign Key
3.	ID_Sales	Varchar	5	Foreign Key
4.	ID_Pelanggan	Varchar	5	Foreign Key
5.	ID_Zona	Integer	11	Foreign Key



No.	Name	Type	Size	Keterangan
6.	ID_Laporan	Varchar	5	Foreign Key
7.	Subjek_Kunjungan	Varchar	100	-
8.	Lokasi_Kunjungan	Varchar	150	-
9.	Tgl_Pengajuan	Date & Time	-	-
10.	Tgl_Pelaksanaan	Date & Time	-	-
11.	Tgl_Laporan	Date & Time	-	-
12.	Deskripsi_Kunjungan	Text	-	-
13.	Status_Kunjungan	Integer	-	-
14.	Tgl_Expired	Date & Time	-	-

**d. Nama Tabel : Laporan**

Primary Key : ID\_Laporan

Foreign Key : ID\_Pengguna, ID\_Kunjungan

Fungsi : Untuk menyimpan hasil kunjungan yang dihasilkan oleh masing-masing zona

Tabel 3.7 Laporan

No.	Name	Type	Size	Keterangan
1.	ID_Laporan	Varchar	5	Primary Key
2.	ID_Pengguna	Integer	-	Foreign Key
3.	ID_Kunjungan	Varchar	5	Foreign Key
4.	Tgl_Hasil_Laporan	Date & Time	-	-
5.	Delivery_Habbit	Text	-	-
6.	Wallet_Kompetitor	Text	-	-
7.	Harga_Kompetitor	Text	-	-
8.	Program_Kompetitor	Text	-	-
9.	Masalah	Text	-	-
10.	Harapan	Text	-	-
11.	Souvenir	Text	-	-
12.	Status_Laporan	Integer	-	-
13.	File_Laporan	Varchar	200	-

**e. Nama Tabel : Zona**

Primary Key : ID\_Zona

Foreign Key : -

Fungsi : Untuk Menyimpan data zona yang dimiliki oleh PT. Royal

Inti Mandiri Abadi.

Tabel 3.8 Zona

No.	Name	Type	Size	Keterangan
1.	ID_Zona	Integer	-	Primary Key
2.	Nama_Zona	Varchar	50	-
3.	Alamat_Zona	Varchar	150	-

f. Nama Tabel : Pelanggan

Primary Key : ID\_Pelanggan

Foreign Key : ID\_Zona

Fungsi : Untuk menyimpan data pelanggan dalam tiap-tiap zona.

Tabel 3.9 Pelanggan

No.	Name	Type	Size	Keterangan
1.	ID_Pelanggan	Varchar	5	Primary Key
2.	ID_Zona	Integer	-	Foreign Key
3.	Nama_Toko	Varchar	50	-
4.	Nama_Penilik	Varchar	100	-
5.	Alamat_Pelanggan	Varchar	150	-
6.	Telp_Pelanggan	Varchar	12	-

g. Nama Tabel : Sales

Primary Key : ID\_Sales

Foreign Key : ID\_Zona

Fungsi : Untuk menyimpan data sales dalam tiap-tiap zona.

Tabel 3.10 Sales

No.	Name	Type	Size	Keterangan
1.	ID_Sales	Varchar	5	Primary Key
2.	ID_Zona	Integer	-	Foreign Key
3.	Nama_Sales	Varchar	100	-
4.	Alamat_Sales	Varchar	150	-
5.	Telp_Sales	Varchar	12	-
6.	Email_Sales	Varchar	100	-
7.	Status_Sales	Varchar	12	-

**h. Nama Tabel : Masalah**

Primary Key : id\_masalah

Foreign Key : -

Fungsi : Untuk Menyimpan data masalah yang dimiliki oleh PT.

Royal Inti Mandiri Abadi.

Tabel 3.11 Masalah

No.	Name	Type	Size	Keterangan
1.	Id_masalah	Integer	-	Primary Key
2.	Nama_Masalah	Varchar	50	-
3.	Keterangan	Varchar	150	-

**i. Nama Tabel : Harapan**

Primary Key : id\_harapan

Foreign Key : id\_masalah

Fungsi : Untuk Menyimpan data harapan yang dimiliki oleh PT.

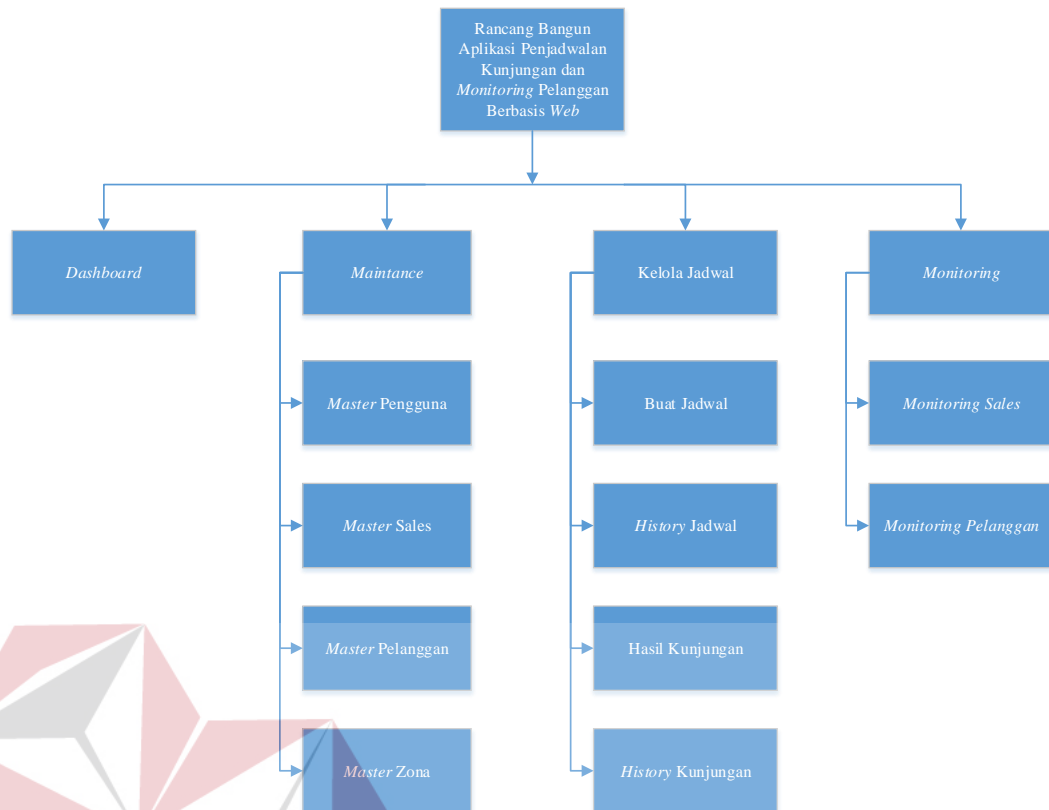
Royal Inti Mandiri Abadi.

Tabel 3.12 Harapan

No.	Name	Type	Size	Keterangan
1.	Id_harapan	Integer	-	Primary Key
2.	Id_masalah	Integer	-	Foreign Key
3.	Keterangan	Varchar	100	-

**3.3.13 Sitemap**

*Sitemap* merupakan sebuah *file* yang berisi daftar *link* yang terdapat pada sebuah *web/blog* atau dengan kata lain merupakan salah satu alat bantu yang dapat mempermudah dalam pengenalan peta situs dalam suatu *website*. Desain *sitemap* aplikasi Penjadwalan dan *Monitoring* dapat dilihat pada gambar 3.26.



Gambar 3.26 Desain Sitemap

### 3.3.14 Desain User Interface

Desain *user interface* adalah rancangan *form* yang akan diimplementasikan kedalam sistem dan berfungsi sebagai antarmuka pengguna dengan sistem. Rancangan ini akan menerima masukan data dari pengguna yang dimasukkan dalam *database* dan hasilnya akan berupa laporan.

#### a. Desain User Interface Halaman Login

Halaman *Login* ini digunakan untuk keamanan sistem. Tujuannya adalah supaya sistem digunakan oleh *user* yang berhak memakai dan berjalan sesuai hak aksesnya masing-masing. Adapun *field* yang harus diisi dalam form ini adalah *username* dan *password*. Selanjutnya klik tombol login, secara otomatis sistem akan bekerja untuk validasi pengguna tersebut dan mencari hak aksesnya. *User* juga dapat memilih opsi untuk mengingat *username* dan *password*. Desain *user interface*

halaman ini dapat dilihat pada gambar 3.27



Username

Password

Ingat kode user dan kata sandi

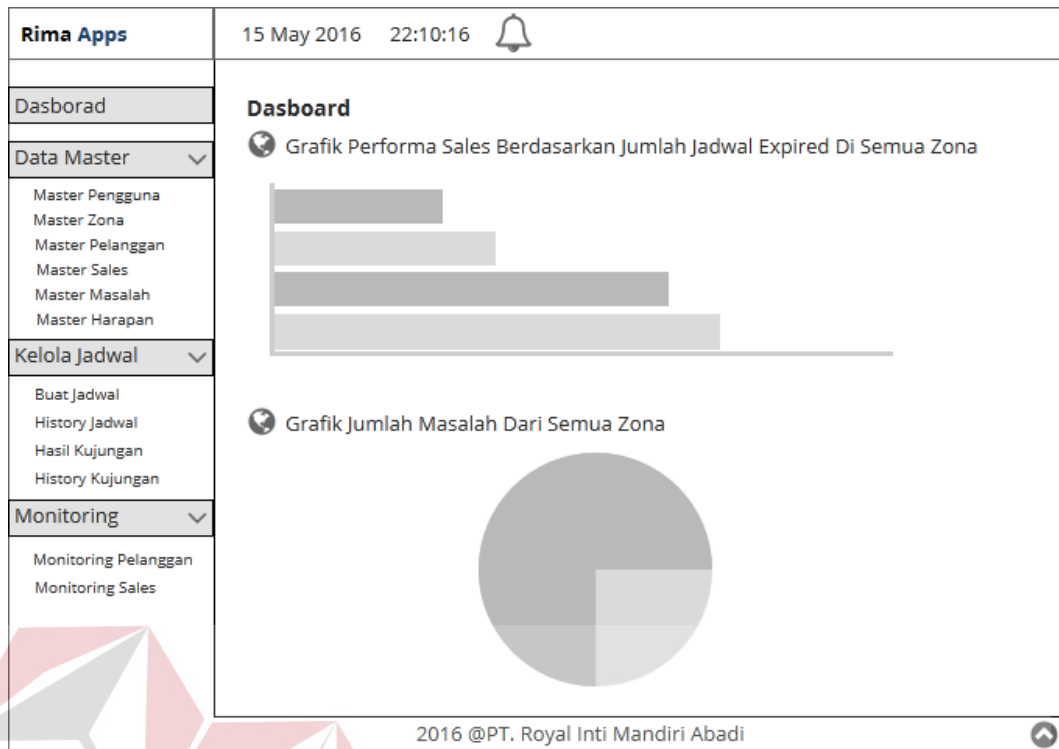
[Lupa kata sandi?](#)

Gambar 3.27 Desain *User Interface* Halaman *Login*

**b. Desain *User Interface* Halaman *Dashboard***

Halaman *dashboard* rancang bangun aplikasi penjadwalan kunjungan dan *monitoring* berbasis web dibagi menjadi 2 yang dibedakan berdasarkan hak akses masing-masing *user*. Halaman *dashboard* dibawah merupakan halaman *dashboard* yang dapat diakses oleh *user* yang memiliki hak akses sebagai *administrator*. Desain *user interface* halaman ini dapat dilihat pada gambar 3.28. Pada halaman ini terdapat beberapa grafik yang berfungsi untuk membantu *user* melihat performa *sales* dan pelanggan. Grafik yang ditampilkan pada halaman *dashboard* antara lain:

- a. Grafik Performa *sales* berdasarkan jumlah jadwal *expired* di semua zona
- b. Grafik jumlah masalah dari semua zona



Gambar 3.28 Desain *User Interface* Halaman *Dashboard Administrator*

Halaman *dashboard* dibawah merupakan halaman *dashboard* yang dapat diakses oleh *user* yang memiliki hak akses sebagai Kepala Perwalian. Desain *user interface* halaman *dashboard* bagian kepala perwalian dapat dilihat pada gambar 3.29. Pada halaman ini, terdapat beberapa *shortcut* data jadwal kunjungan yang langsung bisa diakses oleh *user*.

**Rima Apps** 15 May 2016 22:10:16

**Dasborad**

**Data Master** ▾

**Kelola Jadwal** ▾

**Dashboard**

**Data Jadwal Kunjungan**

10 ▾ Records Per Page Search:

No	ID	Pelaksanaan	Subjek	Status Kunjungan	Aksi
					View  Status

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.29 Desain *User Interface* Halaman *Dashboard* Bagian Kepala Perwalian

c. **Desain *User Interface* Halaman *Master Pengguna***

Halaman ini berfungsi untuk *maintenance* data pengguna, yang bertugas untuk melakukan *maintenance* data pengguna ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *insert* dan *update* data pengguna. Desain *user interface* halaman ini dapat dilihat pada gambar 3.30.







The screenshot shows the 'Master Pengguna' page in the Rima Apps system. On the left is a sidebar menu with items: 'Rima Apps', 'Dasborad', 'Data Master' (with a dropdown showing 'Master Pengguna', 'Master Zona', 'Master Pelanggan', 'Master Sales'), 'Kelola Jadwal', and 'Monitoring'. The main area has a header with the date '15 May 2016' and time '22:10:16'. Below the header, there's a '+ Tambah Pengguna' button and a 'Daftar Pengguna' section. This section includes a '10' dropdown for 'Records Per Page' and a 'Search:' input field. A table with 7 columns is shown below: 'No', 'Nama Zona', 'Nama Pengguna', 'Telp Pengguna', 'Email Pengguna', 'Username Pengguna', and 'Aksi'. The 'Aksi' column contains a 'Edit' button. At the bottom, it says '2016 @PT. Royal Inti Mandiri Abadi'.

Gambar 3.30 Desain *User Interface* Halaman *Master Pengguna*

**d. Desain *User Interface* Halaman Tambah Pengguna**

Halaman ini berfungsi untuk menambah data pengguna, yang bertugas untuk melakukan *maintenance* data pengguna ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat menambahkan data pengguna baru agar dapat mengakses aplikasi. Desain *user interface* halaman ini dapat dilihat pada gambar 3.31.









<b>Rima Apps</b>	15 May 2016 22:10:16 
Dasborad	 <b>Tambah Data Pengguna</b>
Data Master 	Nama Zona <input type="text"/>
Master Pengguna	Nama Pengguna <input type="text"/>
Master Zona	Telepon Pengguna <input type="text"/>
Master Pelanggan	Email Pengguna <input type="text"/>
Master Sales	Username <input type="text"/> <i>Username Tidak Dapat Diubah</i>
Kelola Jadwal 	Password <input type="text"/>
Monitoring 	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
	2016 @PT. Royal Inti Mandiri Abadi 

Gambar 3.31 Desain *User Interface* Halaman Tambah Pengguna

e. **Desain *User Interface* Halaman Update Pengguna**

Halaman ini berfungsi untuk *update* data pengguna, yang bertugas untuk melakukan *maintenance* data pengguna ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat merubah data pengguna baru agar dapat mengakses aplikasi. Desain *user interface* halaman ini dapat dilihat pada gambar 3.32. Dalam aplikasi ini nama zona dan username tidak bisa dirubah.

<b>Rima Apps</b>	15 May 2016 22:10:16 
Dasborad	 <b>Edit Data Pengguna</b>
Data Master 	Nama Zona <input type="text"/> <i>Zona Tidak Dapat Diubah</i>
Master Pengguna	Nama Pengguna <input type="text"/>
Master Zona	Telepon Pengguna <input type="text"/>
Master Pelanggan	Email Pengguna <input type="text"/>
Master Sales	Username <input type="text"/> <i>Username Tidak Dapat Diubah</i>
Kelola Jadwal 	Password <input type="text"/> <i>Apabila password tidak diubah, dikosongkan saja</i>
Monitoring 	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
	2016 @PT. Royal Inti Mandiri Abadi 

Gambar 3.32 Desain *User Interface* Halaman *Update* Pengguna

f. **Desain *User Interface* Halaman Master Zona**

Halaman ini berfungsi untuk *maintenance* data Zona, yang bertugas untuk melakukan *maintenance* data zona ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *insert*, *update* data zona. Desain *user interface* halaman ini dapat dilihat pada gambar 3.33.

**Rima Apps** 15 May 2016 22:10:16

**Master Zona**

**+ Tambah Zona**

**Daftar Zona**

10 Records Per Page Search:







No	Nama Zona	Alamat	Aksi
			Edit

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.33 Desain *User Interface* Halaman *Master Zona*

**g. Desain *User Interface* Halaman *Tambah Zona***







Halaman ini berfungsi untuk menambah data zona, yang bertugas untuk melakukan *maintenance* data zona ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat menambahkan data zona baru apabila mereka memiliki zona yang baru saja mereka buka. Desain *user interface* halaman ini dapat dilihat pada gambar 3.34.

<b>Rima Apps</b>	15 May 2016 22:10:16 
Dasborad	 <b>Tambah Data Zona</b>
Data Master 	Nama Zona <input type="text"/>
<ul style="list-style-type: none"> <li>Master Pengguna</li> <li>Master Zona</li> <li>Master Pelanggan</li> <li>Master Sales</li> </ul>	Alamat Zona <input type="text"/>
Kelola Jadwal 	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
Monitoring 	
	2016 @PT. Royal Inti Mandiri Abadi 

Gambar 3.34 Desain *User Interface* Halaman Tambah Zona

#### h. Desain *User Interface* Halaman *Update Master Zona*

Halaman ini berfungsi untuk *update* data zona, yang bertugas untuk melakukan *maintenance* data zona ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat merubah data zona agar dapat mengakses aplikasi. Desain *user interface* halaman ini dapat dilihat pada gambar 3.35.

<b>Rima Apps</b>	15 May 2016 22:10:16 
Dasborad	 <b>Edit Data Zona</b>
Data Master 	Nama Zona <input type="text"/>
Master Pengguna Master Zona Master Pelanggan Master Sales	Alamat Zona <input type="text"/>
Kelola Jadwal 	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
Monitoring 	
	2016 @PT. Royal Inti Mandiri Abadi 

Gambar 3.35 Desain *User Interface* Halaman *Update Master Zona*

**i. Desain *User Interface* Halaman *Master Sales***

Halaman ini berfungsi untuk *maintenance* data *sales*, yang bertugas untuk melakukan *maintenance* data *sales* ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *insert* dan *update* data *sales*. Desain *user interface* halaman ini dapat dilihat pada gambar 3.36.

**Rima Apps** 15 May 2016 22:10:16

**Master Sales**

[+ Tambah Sales](#)

**Daftar Sales**

10 Records Per Page Search:

ID	Nama Zona	Nama Sales	Email	Alamat	Telp	Status	Aksi
							<a href="#">Edit</a>

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.36 Desain *User Interface* Halaman *Admin Master Sales*

Halaman *dashboard* dibawah merupakan halaman *master sales* yang dapat diakses oleh *user* yang memiliki hak akses sebagai Kepala Perwalian. Desain *user interface* halaman *master sales* bagian kepala perwalian dapat dilihat pada gambar 3.37. Pada halaman ini, *user* dapat melihat daftar *sales* yang aktif dan tidak aktif.

The screenshot displays the 'Master Sales' interface. On the left is a sidebar menu for 'Rima Apps' with items: 'Dasborad', 'Data Master' (with a dropdown arrow), 'Master Pelanggan', 'Master Sales', 'Kelola Jadwal' (with a right arrow), and 'Monitoring' (with a right arrow). The main content area is titled 'Master Sales' and includes a 'Daftar Sales' section. This section features a 'Records Per Page' dropdown set to '10' and a search input field. Below these is a table with the following structure:

ID	Nama Sales	Email	Alamat	Teleponn	Status

The footer of the page contains the text '2016 @PT. Royal Inti Mandiri Abadi' and a small circular icon.

Gambar 3.37 Desain *User Interface* Halaman KP Master Sales

**j. Desain *User Interface* Halaman Tambah Sales**

Halaman ini berfungsi untuk menambah data *sales*, yang bertugas untuk melakukan *maintenance* data *sales* ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat menambahkan data *sales* baru pada masing-masing zona. Desain *user interface* halaman ini dapat dilihat pada gambar 3.38.

Rima Apps	15 May 2016 22:10:16
Dasborad	<b>Tambah Sales</b>
Data Master	ID Sales <input type="text"/>
Master Pengguna Master Zona Master Pelanggan Master Sales	Nama Zona <input type="text"/>
Kelola Jadwal	Nama Sales <input type="text"/>
Monitoring	Email Sales <input type="text"/>
	Alamat Sales <input type="text"/>
	Telepon Sales <input type="text"/>
	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
2016 @PT. Royal Inti Mandiri Abadi	

Gambar 3.38. Desain *User Interface* Halaman *Tambah Sales*

k. **Desain *User Interface* Halaman *Update Master Sales***

Halaman ini berfungsi untuk *update* data *sales*, yang bertugas untuk melakukan *maintenance* data *sales* ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat merubah data *sales* sesuai kebutuhan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.39. Dalam halaman ini *user* dapat merubah status *sales* menjadi aktif atau tidak aktif



<b>Rima Apps</b>	15 May 2016 22:10:16
Dasborad	<b>Edit Data Sales</b>
Data Master	Nama Zona <input type="text"/>
Master Pengguna	Nama Sales <input type="text"/>
Master Zona	Email Sales <input type="text"/>
Master Pelanggan	Alamat Sales <input type="text"/>
Master Sales	Telepon Sales <input type="text"/>
Kelola Jadwal	Status <input type="text"/>
Monitoring	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
	2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.39. Desain *User Interface* Halaman *Update Master Sales*

### 1. Desain *User Interface* Halaman *Master Pelanggan*

Halaman ini berfungsi untuk *maintenance* data pelanggan, yang bertugas untuk melakukan *maintenance* data pelanggan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *insert* dan *update* data pelanggan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.40.

**Rima Apps** 15 May 2016 22:10:16

Dasborad

Data Master

- Master Pengguna
- Master Zona
- Master Pelanggan
- Master Sales

Kelola Jadwal

Monitoring

### Master Pelanggan

Tambah Pelanggan

#### Daftar Pelanggan

10 Records Per Page Search:

ID	Nama Zona	Nama Toko	Nama Pemilik	Alamat Toko	Telp Toko	Aksi
						Edit

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.40 Desain *User Interface* Halaman *admin Master Pelanggan*

Halaman dibawah ini berfungsi untuk menampilkan data pelanggan, yang bertugas untuk menampilkan halaman data pelanggan ini adalah Kepala Perwalian. Desain *user interface* halaman ini dapat dilihat pada gambar 3.41.

**Rima Apps** 15 May 2016 22:10:16

**Master Pelanggan**

**Daftar Pelanggan**

10 | ▾ Records Per Page Search:








ID	Nama Zona	Nama Toko	Nama Pemilik	Alamat Toko	Telp Toko	Aksi

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.41 Desain *User Interface* Halaman KP Master Pelanggan

**m. Desain *User Interface* Halaman Tambah Master Pelanggan**

Halaman ini berfungsi untuk menambah data pelanggan, yang bertugas untuk melakukan *maintenance* data pelanggan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat menambahkan data pelanggan baru pada masing-masing zona. Desain *user interface* halaman ini dapat dilihat pada gambar 3.42

<b>Rima Apps</b>	15 May 2016 22:10:16 
Dasborad	 <b>Tambah Data Pelanggan</b>
Data Master 	ID Toko <input type="text"/>
Master Pengguna	Nama Zona <input type="text"/> 
Master Zona	Nama Toko <input type="text"/>
Master Pelanggan	Nama Pemilik <input type="text"/>
Master Sales	Alamat Toko <input type="text"/>
Kelola Jadwal 	Telepon Toko <input type="text"/>
Monitoring 	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
	2016 @PT. Royal Inti Mandiri Abadi 

Gambar 3.42 Desain *User Interface* Halaman Tambah Pelanggan

n. **Desain *User Interface* Halaman *Update Master Pelanggan***

Halaman ini berfungsi untuk *update* data pelanggan, yang bertugas untuk melakukan *maintenance* data pelanggan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat merubah data pelanggan sesuai kebutuhan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.43. Dalam halaman zona tidak dapat diubah.

Rima Apps	15 May 2016 22:10:16
Dasborad	<b>Edit Data Pelanggan</b>
Data Master	Nama Zona <input type="text" value="Zona Tidak Dapat Dirubah"/> <i>Zona Tidak Dapat Dirubah</i>
<ul style="list-style-type: none"> <li>Master Pengguna</li> <li>Master Zona</li> <li>Master Pelanggan</li> <li>Master Sales</li> </ul>	Nama Toko <input type="text"/>
Kelola Jadwal	Nama Pemilik <input type="text"/>
Monitoring	Alamat Toko <input type="text"/>
	Telepon Toko <input type="text"/>
	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
2016 @PT. Royal Inti Mandiri Abadi	

Gambar 3.43 Desain *User Interface* Halaman *Update* Pelanggan

**o. Desain *User Interface* Halaman *Master* Masalah**

Halaman ini berfungsi untuk *maintenance* data masalah, yang bertugas untuk melakukan *maintenance* data masalah ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *insert* dan *update* data masalah. Desain *user interface* halaman ini dapat dilihat pada gambar 3.44.

The screenshot displays the 'Master Masalah' page in the Rima Apps system. On the left is a sidebar with a 'Data Master' dropdown menu containing 'Master Pengguna', 'Master Zona', 'Master Pelanggan', and 'Master Sales'. The main area features a '+ Tambah Masalah' button, a 'Daftar Masalah' section with a search input and a 'Records Per Page' dropdown set to 10, and a table with the following structure:







No	Nama Masalah	Keterangan	Aksi
			<a href="#">Edit</a>

The footer of the page reads '2016 @PT. Royal Inti Mandiri Abadi'.

Gambar 3.44 Desain *User Interface* Halaman *admin Master Masalah*

p. **Desain *User Interface* Halaman Tambah Master Masalah**

Halaman ini berfungsi untuk menambah data masalah, yang bertugas untuk melakukan *maintenance* data masalah ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat menambahkan data masalah baru. Desain *user interface* halaman ini dapat dilihat pada gambar 3.45

<b>Rima Apps</b>	15 May 2016 22:10:16 
Dasborad	 <b>Tambah Data Masalah</b>
Data Master 	Nama Masalah <input type="text"/>
<ul style="list-style-type: none"> <li>Master Pengguna</li> <li>Master Zona</li> <li>Master Pelanggan</li> <li>Master Sales</li> </ul>	Keterangan <input type="text"/>
Kelola Jadwal 	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
Monitoring 	
	2016 @PT. Royal Inti Mandiri Abadi 

Gambar 3.45 Desain *User Interface* Halaman Tambah Masalah

q. **Desain *User Interface* Halaman *Update Master* Masalah**

Halaman ini berfungsi untuk *update* data masalah, yang bertugas untuk melakukan *maintenance* data masalah ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat merubah data masalah sesuai kebutuhan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.46.

<b>Rima Apps</b>	15 May 2016 22:10:16
Dasborad	<b>Edit Data Masalah</b>
Data Master	Nama Masalah <input type="text"/>
<ul style="list-style-type: none"> <li>Master Pengguna</li> <li>Master Zona</li> <li>Master Pelanggan</li> <li>Master Sales</li> </ul>	Keterangan <input type="text"/>
Kelola Jadwal	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
Monitoring	
2016 @PT. Royal Inti Mandiri Abadi	

Gambar 3.46 Desain *User Interface* Halaman *Update Masalah*

#### r. **Desain *User Interface* Halaman *Master Harapan***

Halaman ini berfungsi untuk *maintenance* data harapan, yang bertugas untuk melakukan *maintenance* data harapan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *insert* dan *update* data harapan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.47.



The screenshot displays the 'Master Harapan' interface. On the left is a sidebar with the following menu items: 'Rima Apps', 'Dasborad', 'Data Master' (with a dropdown arrow), 'Kelola Jadwal' (with a right arrow), and 'Monitoring' (with a right arrow). Under 'Data Master', there are sub-items: 'Master Pengguna', 'Master Zona', 'Master Pelanggan', and 'Master Sales'. The main content area is titled 'Master Harapan' and features a blue '+ Tambah Harapan' button. Below this is a 'Daftar Harapan' section with a search input field and a 'Records Per Page' dropdown set to '10'. A table with the following structure is shown:

No	Masalah	Harapan	Aksi
			<a href="#">Edit</a>

The footer of the page reads '2016 @PT. Royal Inti Mandiri Abadi'.

Gambar 3.47 Desain *User Interface* Halaman *admin Master Harapan*

s. **Desain *User Interface* Halaman Tambah Master Harapan**

Halaman ini berfungsi untuk menambah data harapan, yang bertugas untuk melakukan *maintenance* data harapan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat menambahkan data harapan baru. Desain *user interface* halaman ini dapat dilihat pada gambar 3.48

<b>Rima Apps</b>	15 May 2016 22:10:16
<b>Dasborad</b>	<b>Tambah Data Harapan</b>
<b>Data Master</b>	ID Harapan <input type="text"/>
Master Pengguna	Masalah <input type="text"/>
Master Zona	Harapan <input type="text"/>
Master Pelanggan	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
Master Sales	
<b>Kelola Jadwal</b>	
<b>Monitoring</b>	
	2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.48 Desain *User Interface* Halaman Tambah Harapan

t. **Desain *User Interface* Halaman *Update Master* Harapan**

Halaman ini berfungsi untuk *update* data harapan, yang bertugas untuk melakukan *maintenance* data harapan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat merubah data harapan sesuai kebutuhan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.49.

<b>Rima Apps</b>	15 May 2016 22:10:16
Dasborad	<b>Edit Data Harapan</b>
Data Master	ID Harapan <input type="text"/>
Master Pengguna	Masalah <input type="text"/>
Master Zona	Harapan <input type="text"/>
Master Pelanggan	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
Master Sales	
Kelola Jadwal	
Monitoring	
	2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.49 Desain *User Interface* Halaman *Update* Harapan

**u. Desain *User Interface* Halaman Manajemen Jadwal Kunjungan**

Halaman ini berfungsi untuk mengelola jadwal kunjungan, yang bertugas untuk mengelola jadwal kunjungan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *insert*, *update*, dan *delete* jadwal kunjungan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.50.

**Rima Apps** 15 May 2016 22:10:16

**Manajemen Jadwal Kunjungan**

[+ Buat Jadwal](#)

**Daftar Jadwal Kunjungan**

10 Records Per Page Search:

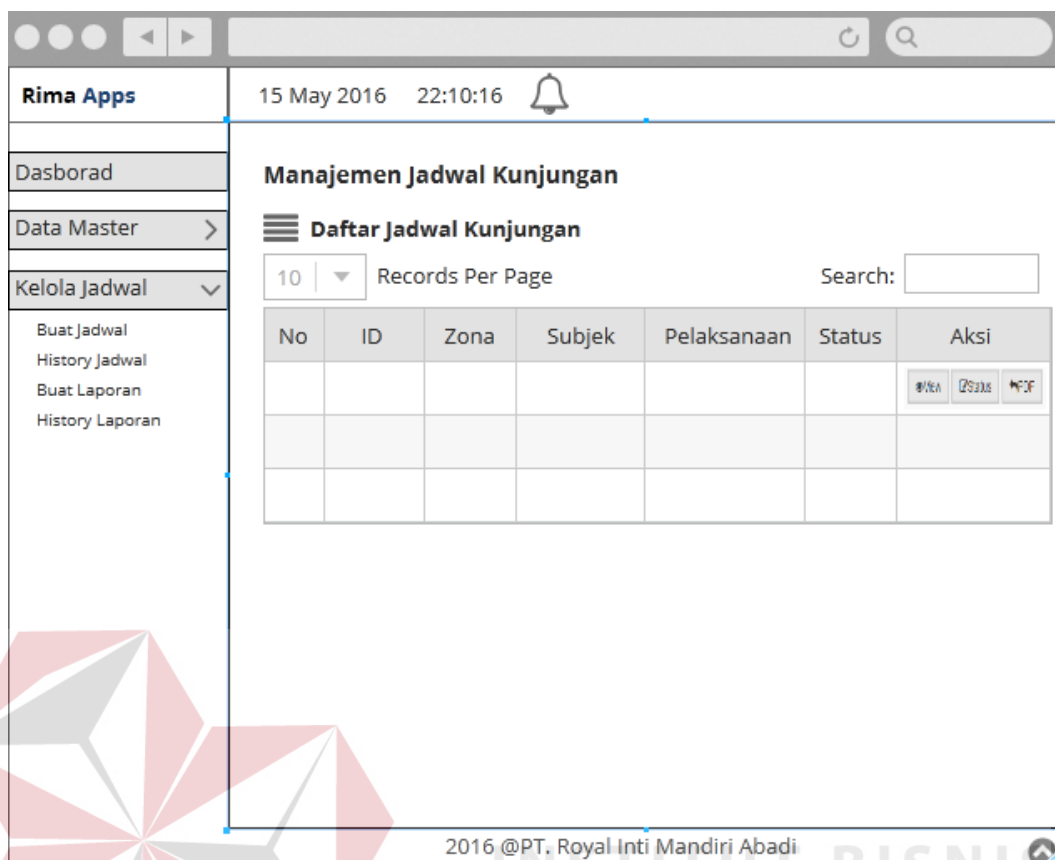
No	ID	Zona	Subjek	Pelaksanaan	Status	Aksi
						<a href="#">View</a> <a href="#">Edit</a>

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.50 Desain *User Interface* Manajemen Jadwal Kunjungan Admin

Halaman dibawah ini berfungsi untuk mengelola jadwal kunjungan, yang bertugas untuk mengelola jadwal kunjungan ini adalah Kepala Perwalian. Kepala Perwalian dapat melakukan edit status kunjungan dan mencetak jadwal kunjungan.

Desain *user interface* halaman ini dapat dilihat pada gambar 3.51.



Gambar 3.51 Desain *User Interface* Manajemen Jadwal Kunjungan KP

v. **Desain *User Interface* Halaman Tambah Jadwal**

Halaman ini berfungsi untuk menambah data jadwal kunjungan, yang bertugas untuk menambah data jadwal kunjungan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat menambahkan data jadwal kunjungan baru pada masing-masing zona. Desain *user interface* halaman ini dapat dilihat pada gambar 3.52.

Rima Apps	15 May 2016 22:10:16	
Dasborad	<b>Tambah Jadwal</b>	
Data Master >	ID Jadwal	<input type="text"/>
Kelola Jadwal v	Nama Zona	<input type="text" value="v"/>
Buat Jadwal	Nama Sales	<input type="text"/>
History Jadwal	Subjek Kunjungan	<input type="text"/>
Hasil Kunjungan	Lokasi Kunjungan	<input type="text" value="v"/>
History Kunjungan	Deskripsi Kunjungan	<input type="text"/>
Monitoring >	Tanggal Kunjungan	<input type="text"/>
		<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
	2016 @PT. Royal Inti Mandiri Abadi	

Gambar 3.52 Desain *User Interface* Tambah Jadwal Kunjungan

w. **Desain *User Interface* Halaman *Update* Jadwal Kunjungan**

Halaman ini berfungsi untuk *update* jadwal kunjungan, yang bertugas untuk melakukan *update* jadwal kunjungan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat merubah data pelanggan sesuai kebutuhan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.53. Dalam halaman zona tidak dapat diubah.

Rima Apps	15 May 2016 22:10:16	
Dasborad	<b>Edit Jadwal</b>	
Data Master >	Nama Zona	<input type="text"/>
Kelola Jadwal ▾	Nama Sales	<input type="text"/>
Buat Jadwal	Subjek Kunjungan	<input type="text"/>
History Jadwal	Lokasi Kunjungan	<input type="text"/>
Hasil Kunjungan	Deskripsi Kunjungan	<input type="text"/>
History Kujungan	Tanggal Kunjungan	<input type="text"/>
Monitoring >	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>	
2016 @PT. Royal Inti Mandiri Abadi		

Gambar 3.53 Desain *User Interface Update Jadwal Kunjungan*

#### x. **Desain *User Interface* Halaman *History Jadwal***

Halaman ini berfungsi untuk menampilkan semua history jadwal kunjungan yang telah dilaksanakan oleh masing-masing zona, yang bertugas untuk mengelola history jadwal kunjungan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan melihat *detail history* jadwal kunjungan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.54.

The screenshot displays the 'Rima Apps' interface. At the top, it shows the date '15 May 2016' and time '22:10:16' next to a notification bell icon. The main content area is titled 'Daftar History Jadwal Kunjungan'. Below the title, there is a sub-header 'Data History Jadwal Kunjungan' with a hamburger menu icon. A 'Records Per Page' dropdown is set to '10', and a search bar is present. The table below has the following structure:

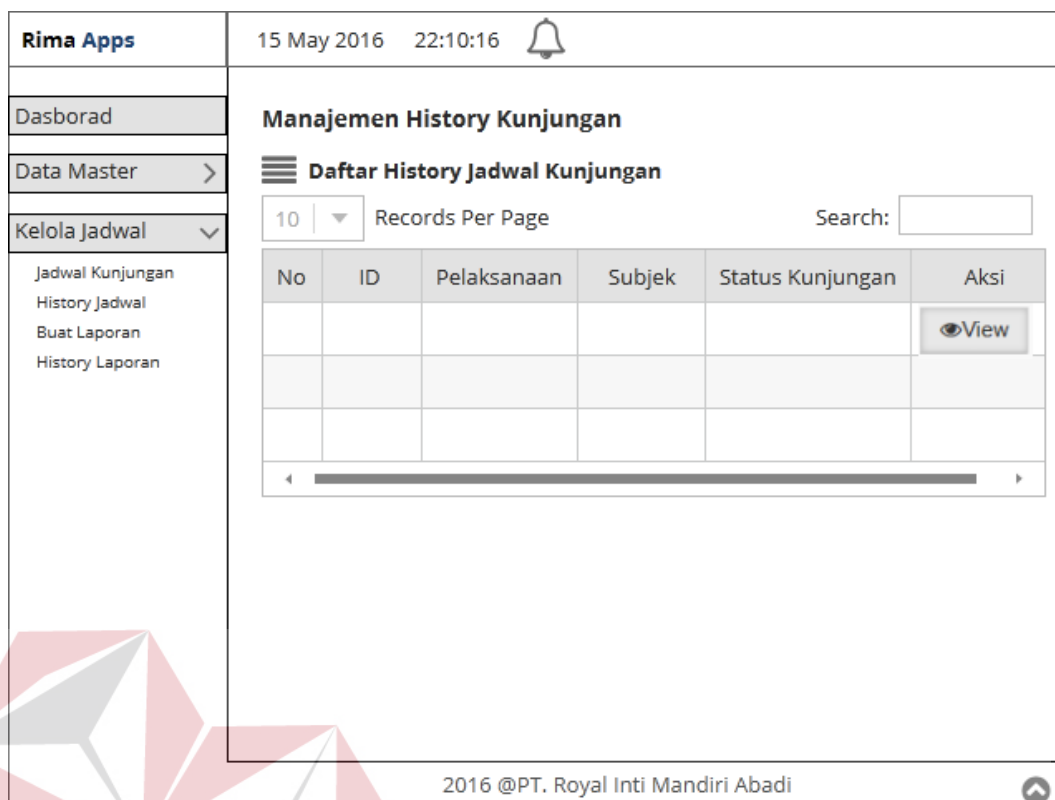
No	ID Kunjungan	Zona	Subjek	Pelaksanaan	Status	Aksi
						<a href="#">View</a>

The footer of the application shows '2016 @PT. Royal Inti Mandiri Abadi' and a small circular icon.

Gambar 3.54 Desain *User Interface History Jadwal Kunjungan Admin*

Halaman dibawah ini berfungsi untuk menampilkan semua history jadwal kunjungan yang telah dilaksanakan oleh masing-masing zona, yang bertugas untuk mengelola history jadwal kunjungan ini adalah Kepala Perwalian. Kepala Perwalian dapat melihat *detail history* jadwal kunjungan. Desain *user interface* halaman ini dapat dilihat pada gambar 3.55.

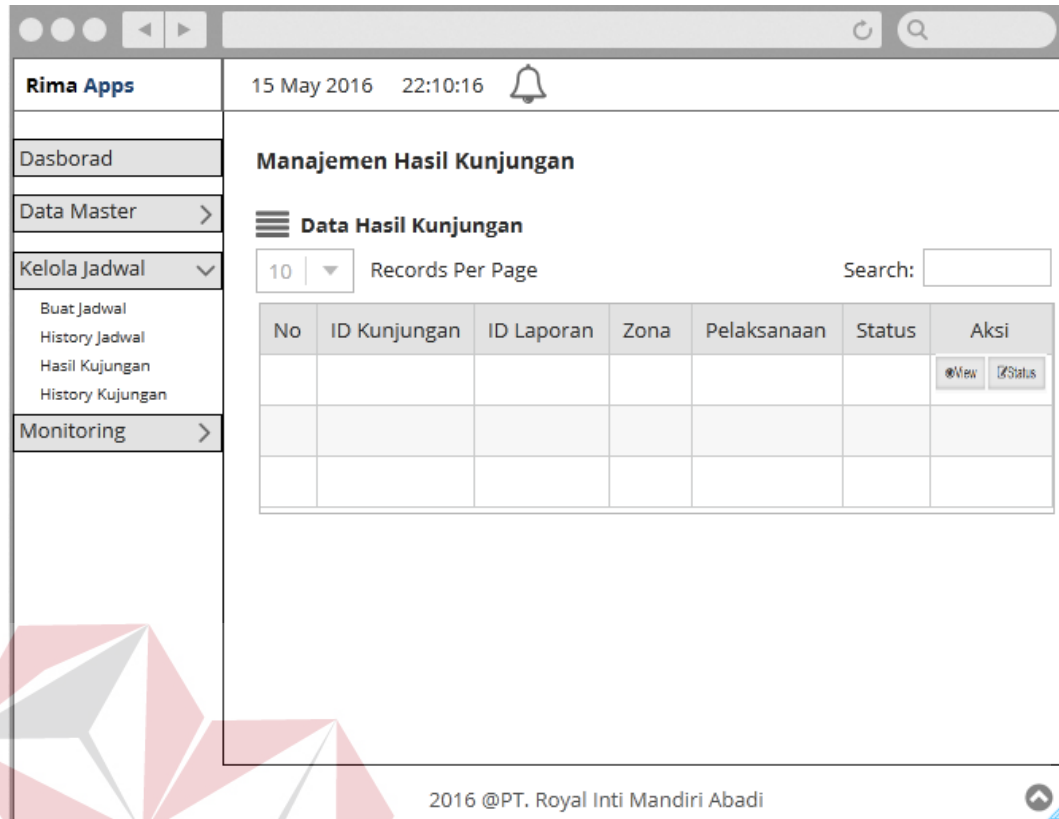




Gambar 3.55 Desain *User Interface* History Jadwal Kunjungan KP

#### y. **Desain *User Interface* Halaman Hasil Kunjungan**

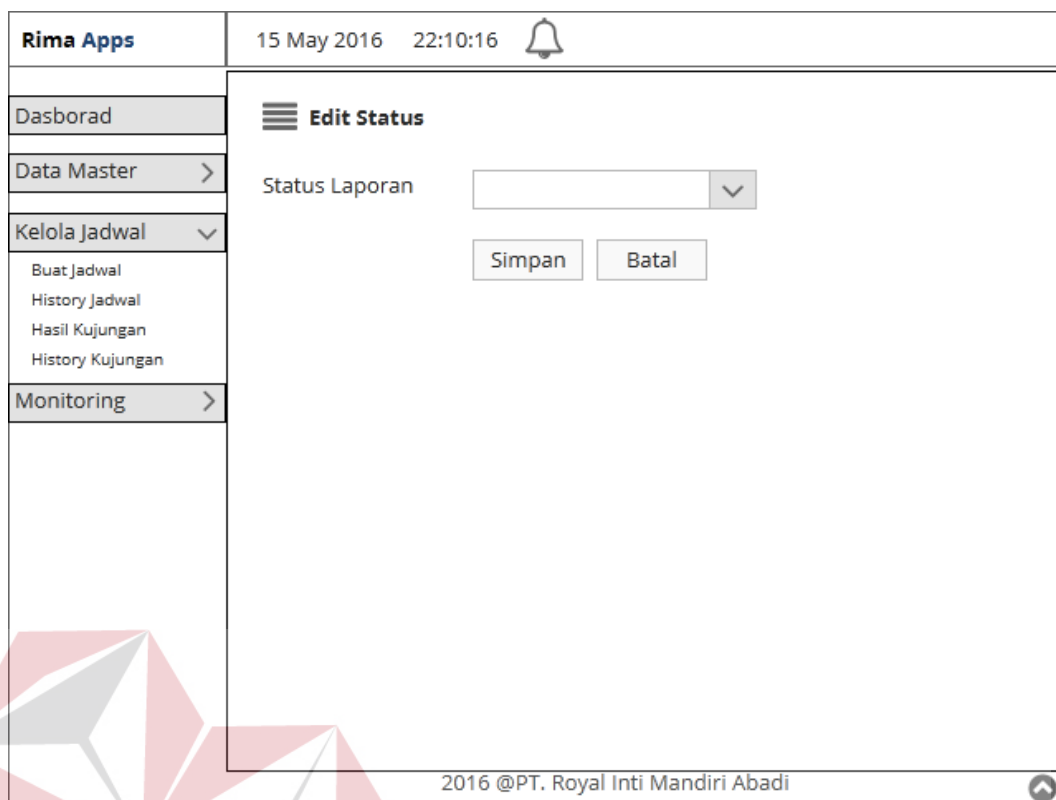
Halaman ini berfungsi untuk menampilkan semua laporan hasil kunjungan yang telah diinputkan oleh masing-masing zona, yang bertugas untuk mengelola history jadwal kunjungan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *approval* atau menolak laporan hasil kunjungan yang telah diinputkan oleh masing-masing zona sebelumnya. Desain *user interface* halaman ini dapat dilihat pada gambar 3.56.



Gambar 3.56 Desain *User Interface* Hasil Kunjungan

#### z. Desain *User Interface* Halaman Edit Status Kunjungan Pelanggan

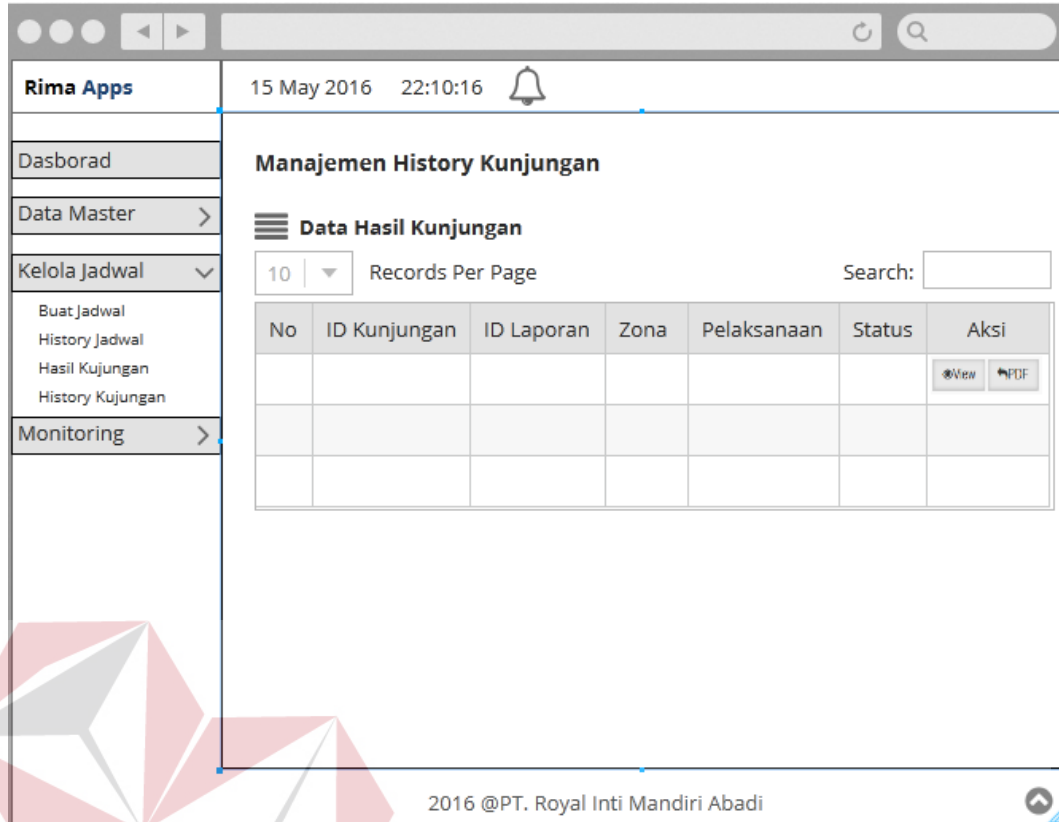
Halaman ini berfungsi untuk mengubah status laporan hasil kunjungan yang telah diinputkan oleh masing-masing zona, yang bertugas untuk mengelola halaman edit status kunjungan ini adalah *Administrator* / Kepala Divisi *Customer Service*. *Administrator* dapat melakukan *approval* atau menolak laporan hasil kunjungan yang telah diinputkan oleh masing-masing zona sebelumnya. Desain *user interface* halaman ini dapat dilihat pada gambar 3.57.



Gambar 3.57 Desain *User Interface* Edit Status Kunjungan

**aa. Desain *User Interface* Halaman *History* Kunjungan**

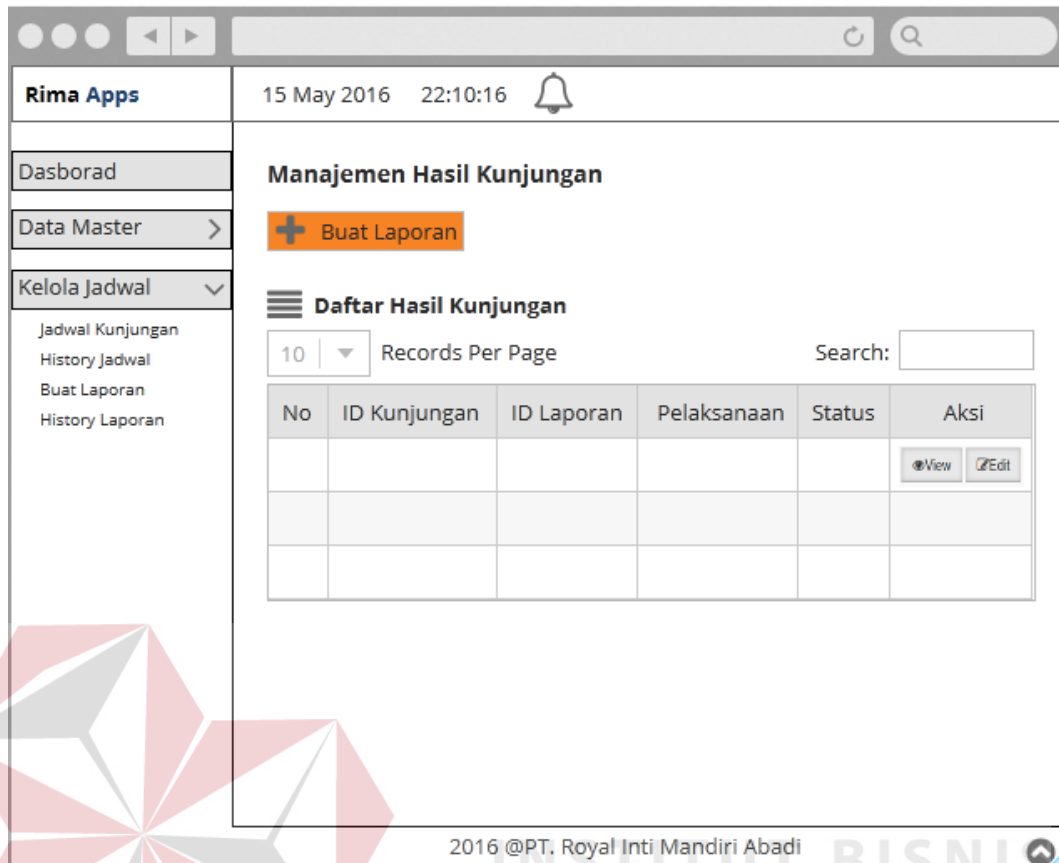
Halaman ini berfungsi untuk menampilkan semua history hasil kunjungan yang telah dilaksanakan oleh masing-masing zona dan telah disetujui oleh Kepala Divisi *Customer Service*, yang bertugas untuk mengelola history jadwal kunjungan ini adalah *Administrator* / Kepala Divisi *Customer Service*. Desain *user interface* halaman ini dapat dilihat pada gambar 3.58.



Gambar 3.58 Desain *User Interface* History Kunjungan

bb. **Desain *User Interface* Halaman Hasil Kunjungan**

Halaman ini berfungsi untuk menampilkan laporan hasil kunjungan yang dilakukan *sales*, yang bertugas untuk menampilkan laporan kunjungan ini adalah Kepala Perwalian. Desain *user interface* halaman ini dapat dilihat pada gambar 3.59.



Gambar 3.59 Desain *User Interface* Hasil Kunjungan

#### cc. Desain *User Interface* Halaman Buat Laporan

Halaman ini berfungsi untuk membua laporan hasil kunjungan yang dilakukan *sales*, yang bertugas untuk membuat laporan kunjungan ini adalah Kepala Perwalian. Desain *user interface* halaman ini dapat dilihat pada gambar 3.60.

**Rima Apps** 15 May 2016 22:10:16

**Buat Laporan**

ID Laporan

Nama Zona

ID Kunjungan

Masalah

Harapan

Wallet Share

Harga Kompetitor

Program Kompetitor

File Lampiran

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.60 Desain *User Interface* Buat Laporan

**dd. Desain *User Interface* Halaman Edit Laporan**

Halaman ini berfungsi untuk mengubah laporan hasil kunjungan yang dilakukan *sales*, yang bertugas untuk mengubah laporan kunjungan ini adalah Kepala Perwalian. Desain *user interface* halaman ini dapat dilihat pada gambar 3.61.

**Rima Apps** 15 May 2016 22:10:16

**Dasboard** **Edit Laporan**

**Data Master** > ID Laporan

**Kelola Jadwal** ▾ Nama Zona

Jadwal Kunjungan ID Kunjungan

History Jadwal

Buat Laporan Masalah  ▾

History Laporan Harapan  ▾

Wallet Share

Harga Kompetitor

Program Kompetitor

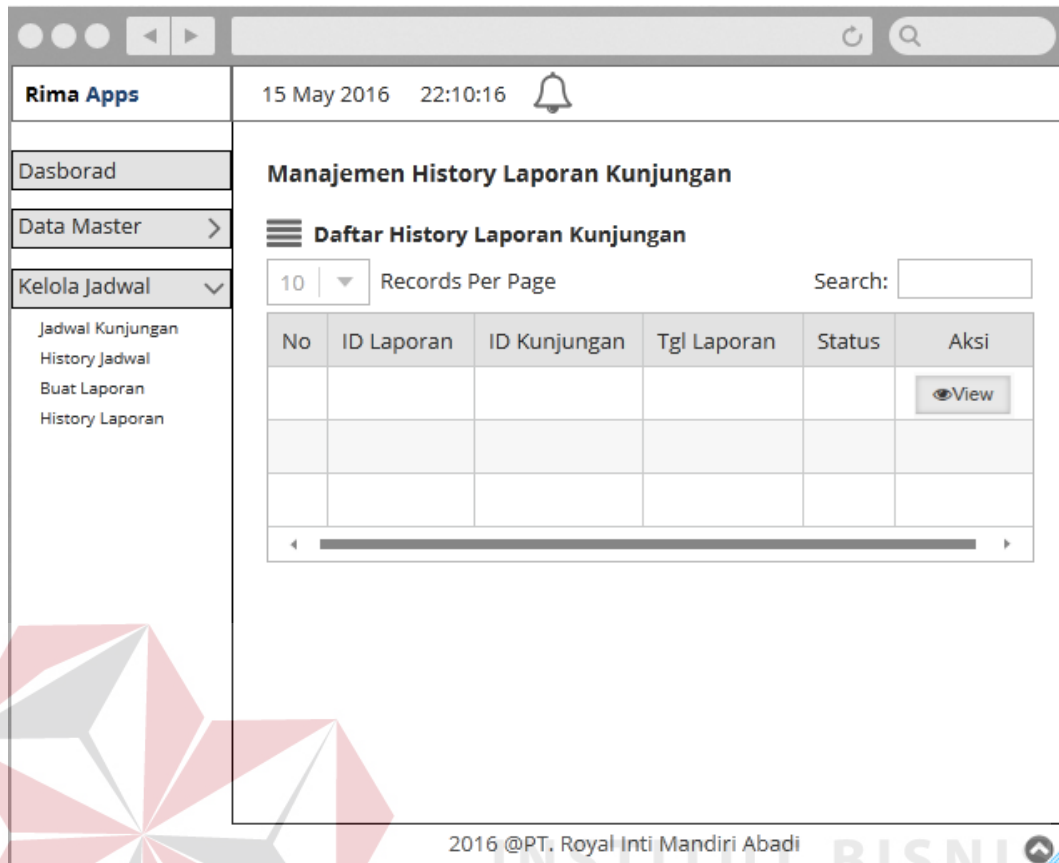
File Lampiran

2016 @PT. Royal Inti Mandiri Abadi

Gambar 3.61 Desain *User Interface Update Laporan*

**ee. Desain *User Interface Halaman History Laporan***

Halaman ini berfungsi untuk menampilkan *history* laporan yang sudah di setujui oleh *Customer Service*, yang bertugas untuk menampilkan *history* laporan ini adalah Kepala Perwalian. Desain *user interface* halaman ini dapat dilihat pada gambar 3.62.

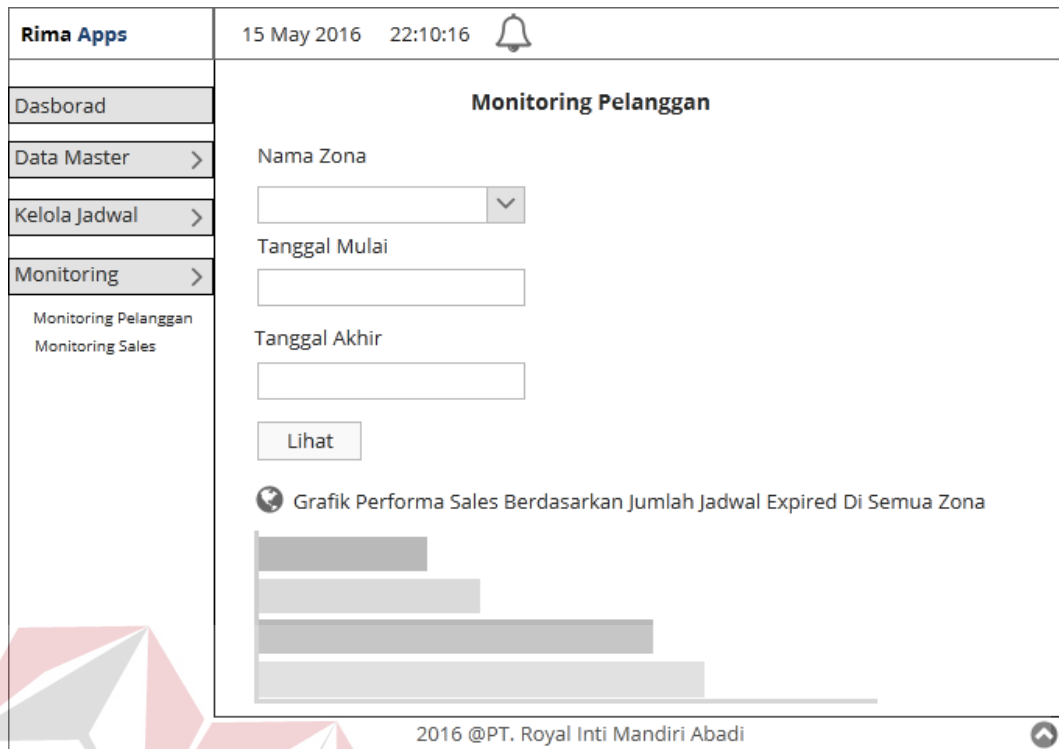


Gambar 3.62 Desain *User Interface* History Laporan

**ff. Desain *User Interface* Halaman *Monitoring* Pelanggan**

Halaman ini berfungsi untuk menampilkan *monitoring* pelanggan, yang bertugas untuk menampilkan halaman *monitoring* pelanggan ini adalah *Customer Service*. Desain *user interface* halaman ini dapat dilihat pada gambar 3.63.

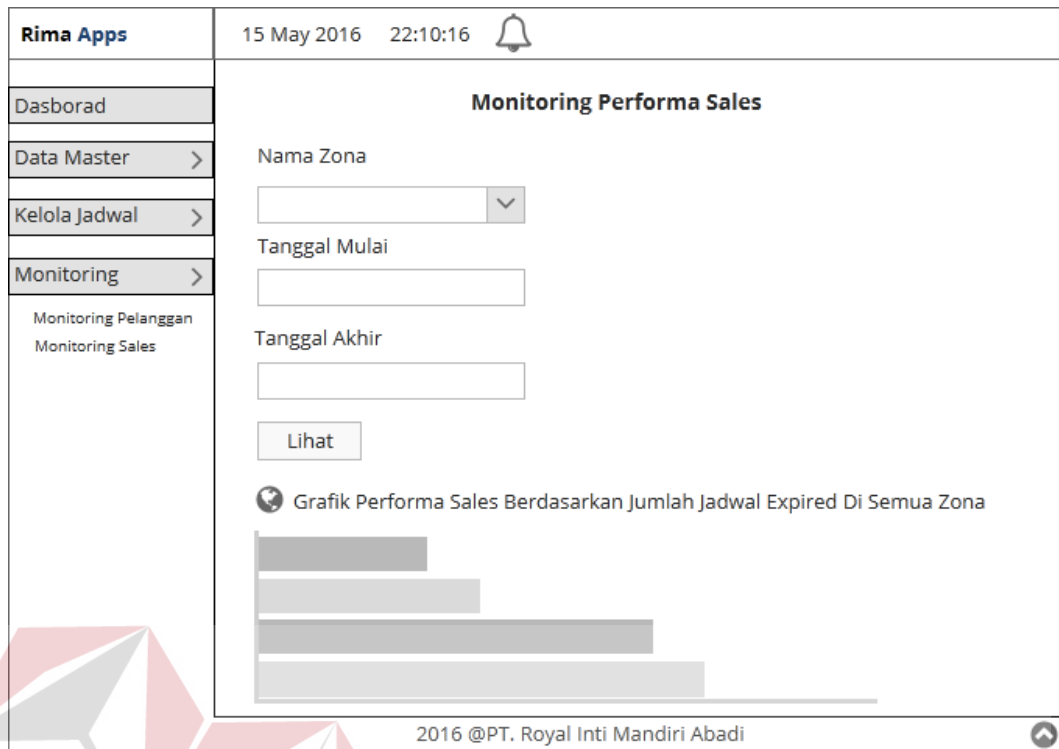




Gambar 3.63 Desain *User Interface Monitoring Pelanggan*

**gg. Desain *User Interface Halaman Monitoring Sales***

Halaman ini berfungsi untuk menampilkan *monitoring sales*, yang bertugas untuk menampilkan halaman *monitoring sales* ini adalah *Customer Service*. Desain *user interface* halaman ini dapat dilihat pada gambar 3.64.



Gambar 3.64 Desain *User Interface Monitoring Sales*

### 3.3.15 Desain *Input/Output*

Desain *input/output* digunakan untuk menggambarkan *input* dan *output* dari lingkungan eksternal sistem yang akan dimasukkan kedalam sistem. Desain *input* dalam sistem ini tidak digambarkan karena semua input dimasukkan langsung kedalam sistem. Desain *output* yang akan dihasilkan sistem adalah jadwal kunjungan dan laporan kunjungan.

#### a. Desain *Output Jadwal Kunjungan*

Desain jadwal kunjungan ini digunakan sales untuk mencatat hasil kunjungan ke pelanggan. Jadwal kunjungan ini dicetak oleh masing-masing kepala perwalian zona yang sebelumnya jadwal sudah dibuat oleh Customer Service. Desain *output* jadwal kunjungan dapat dilihat pada gambar 3.65.



**PT. Royal inti Mandiri Abadi**  
**Distributor & Suppler**  
 Jl. Ambengan No.1, Blok N-O Surabaya, 60272  
 Telp. (031) 5452992, 5327888

---

**ID Kunjungan** :  
**Tanggal Pelaksanaan** :  
**ID Laporan** :  
**Tanggal Laporan** :  
**Subjek Kunjungan** :  
**Nama Sales** :  
**Nama Toko** :  
**Lokasi Kunjungan** :  
**Deskripsi Kunjungan** :  
**Delivery Habbit** :  
**wallet share** :  
**Harga Kompetitor** :  
**Program kompetitor** :  
**Masalah** :  
**Harapan** :  
**Souvenir** :




INSTITUT BISNIS  
& INFORMATIKA

stikom

Gambar 3.65 Desain *Output* Jadwal Kunjungan

**b. Desain *Output* Laporan Kunjungan**

Desain jadwal laporan kunjungan ini digunakan *Customer Service* untuk melihat hasil kunjungan *sales* ke pelanggan. Jadwal kunjungan ini dicetak oleh *Customer Service*. Desain *output* jadwal kunjungan dapat dilihat pada gambar 3.66.

 <b>PT. Royal inti Mandiri Abadi</b> Distributor & Supplier Jl. Ambengan No.1, Blok N-O Surabaya, 60272 Telp. (031) 5452992, 5327888	
<b>ID Kunjungan</b>	: 0032
<b>Tanggal Pelaksanaan</b>	: 14 Juli 2016
<b>ID Laporan</b>	: 0006
<b>Tanggal Laporan</b>	: 16 juli 2016
<b>Subjek Kunjungan</b>	: Mencari Info Sebanyak-banyaknya dari pelanggan
<b>Nama Sales</b>	: Sumanto
<b>Nama Toko</b>	: Anggrek Raya Sejati
<b>Lokasi Kunjungan</b>	: Jalan Ketegan Satria, Sidoarjo
<b>Deskripsi Kunjungan</b>	: info kompetitor dan lain sebagainya
<b>Delivery Habbit</b>	: Pengiriman sering lambat
<b>wallet share</b>	: 5 sak perbulan
<b>Harga Kompetitor</b>	: Semen Gresik Rp.20.000,
<b>Program kompetitor</b>	: Hadiah piring
<b>Masalah</b>	: Sering telat dalam mengirim
<b>Harapan</b>	: Pelayanan harus ditingkatkan
<b>Souvenir</b>	: Piring dan Gelas Setiap Pembelian diatas 2ton

Gambar 3.66 Desain Laporan Kunjungan

### 3.3.15 Rancangan Pengujian dan Evaluasi Sistem

Rancang bangun aplikasi penjadwalan dan *monitoring* yang telah dirancang dan dibangun harus diuji untuk mengetahui apakah sistem dapat berjalan sesuai dengan yang diharapkan. Uji coba yang dilakukan adalah proses *login*, pengelolaan data *master*, penjadwalan kunjungan, status kunjungan, status laporan kunjungan dan *monitoring*. Pengujian yang dilakukan akan menggunakan metode *Black Box Testing*. Pengujian ini akan berfokus pada apakah unit program telah memenuhi kebutuhan (*requirement*) yang disebutkan dalam spesifikasi. Cara pengujian akan dilakukan dengan menjalankan unit atau modul, kemudian diamati apakah hasil dari

unit atau modul sesuai dengan proses yang diinginkan. Hal-hal yang akan diujikan akan ditampilkan secara detail pada tabel 3.13.

Tabel 3.13 Rencana Pengujian Rancang Bangun Aplikasi Penjadwalan Kunjungan dan *Monitoring*

<b>Requirement yang diuji</b>	<b>Fungsi yang diuji</b>
<b>Login</b>	Melakukan <i>login</i>
<b>Data Pengguna</b>	Melakukan operasi CRU ( <i>Create, Read, dan Update</i> ) pada data pengguna
<b>Data Zona</b>	Melakukan operasi CRU ( <i>Create, Read, dan Update</i> ) pada data zona
<b>Data Sales</b>	Melakukan operasi CRU ( <i>Create, Read, dan Update</i> ) pada data <i>sales</i>
<b>Data Pelanggan</b>	Melakukan operasi CRU ( <i>Create, Read, dan Update</i> ) pada data pelanggan
<b>Data Masalah</b>	Melakukan operasi CRU ( <i>Create, Read, dan Update</i> ) pada data Masalah
<b>Data Harapan</b>	Melakukan operasi CRU ( <i>Create, Read, dan Update</i> ) pada data Harapan
<b>Data Penjadwalan Kunjungan</b>	Melakukan penjadwalan kunjungan
	Menampilkan <i>detail</i> jadwal kunjungan
	Melakukan operasi CRUD ( <i>Create, Read, Update dan Delete</i> ) pada data penjadwalan kunjungan
<b>Data Hasil Kunjungan</b>	Membuat laporan hasil kunjungan
	Melakukan operasi CRU ( <i>Create, Read, Update</i> ) pada data hasil kunjungan
<b>Data Laporan Kunjungan</b>	Menampilkan data dan mencetak laporan kunjungan
<b>Monitoring</b>	Menampilkan grafik monitoring <i>sales</i> dan pelanggan

**a. Desain Uji Coba Login**

Desain uji coba *login* bertujuan untuk menguji apakah fungsi *login* dapat berjalan sesuai yang diharapkan. Desain uji coba *login* dapat dilihat pada tabel 3.14.

Tabel 3.14 Desain Uji Coba *Login*

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Melakukan <i>login</i>	<i>Username</i> dan <i>Password</i>	<i>Login</i> berhasil dan masuk ke halaman <i>dashboard</i>	
2	Melakukan <i>login</i> dengan <i>format</i> yang salah	<i>Username</i> dan <i>Password</i>	Muncul pesan “ <i>Username</i> atau <i>Password</i> Anda Salah”	

**b. Desain Uji Coba Data Pengguna**

Desain uji coba data pengguna bertujuan untuk menguji apakah fungsi CRU (*Create, Read, dan Update*) pada data pengguna dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data pengguna dapat dilihat pada tabel 3.15.

Tabel 3.15 Desain Uji Coba Data *User*

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman <i>Master Pengguna</i>	Memilih menu “ <i>Master Pengguna</i> ”	Masuk ke halaman <i>Master Pengguna</i>	
2	Memasukan data pengguna	Nama pengguna dan <i>password</i>	Data pengguna berhasil disimpan dan ditampilkan pada tabel pengguna	

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
3	Mengubah data pengguna	Nama pengguna dan <i>password</i>	Data pengguna berhasil diperbarui dan ditampilkan pada tabel pengguna	

### c. Desain Uji Coba Data Zona

Desain uji coba data zona bertujuan untuk menguji apakah fungsi CRU (*Create, Read, Update*) pada data zona dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data zona dapat dilihat pada tabel 3.16.

Tabel 3.16 Desain Uji Coba Data Zona

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman <i>Master Zona</i>	Memilih menu " <i>Master Zona</i> "	Masuk ke halaman <i>Master Zona</i>	
2	Memasukan data <i>Zona</i>	Nama zona dan alamat zona	Data zona berhasil disimpan dan ditampilkan pada tabel data zona	
3	Mengubah data zona	Nama zona dan alamat zona	Data zona berhasil diperbarui dan ditampilkan pada tabel data zona	

### d. Desain Uji Coba Data Sales

Desain uji coba data *sales* bertujuan untuk menguji apakah fungsi CRU (*Create, Read, Update, dan Delete*) pada data *sales* dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data *sales* dapat dilihat pada tabel 3.17.

Tabel 3.17 Desain Uji Coba Data *Sales*

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman <i>Master Sales</i>	Memilih menu “ <i>Master Sales</i> ”	Masuk ke halaman <i>Master Sales</i>	
2	Memasukan data <i>sales</i>	Nama zona, nama <i>sales</i> , <i>email</i> , alamat dan telepon	Data <i>sales</i> berhasil disimpan dan ditampilkan pada tabel data <i>sales</i>	
3	Mengubah data <i>sales</i>	Nama zona, nama <i>sales</i> , <i>email</i> , alamat, telepon dan status	Data <i>sales</i> berhasil diperbarui dan ditampilkan pada tabel data <i>sales</i>	

e. **Desain Uji Coba Data Pelanggan**

Desain uji coba data pelanggan bertujuan untuk menguji apakah fungsi CRU (*Create, Read, Update*) pada data pelanggan dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data pelanggan dapat dilihat pada tabel 3.18.

Tabel 3.18 Desain Uji Coba Data Pelanggan

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman <i>Master Pelanggan</i>	Memilih menu “ <i>Master Pelanggan</i> ”	Masuk ke halaman <i>Master Pelanggan</i>	
2	Memasukan data Pelanggan	Nama zona, nama toko, nama pemilik,	Data pelanggan berhasil disimpan dan ditampilkan pada tabel data pelanggan	



<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
		alamat toko dan telepon toko		
3	Mengubah data pelanggan	Nama toko, nama pemilik, alamat toko dan telepon toko	Data pelanggan berhasil diperbarui dan ditampilkan pada tabel data pelanggan	

#### f. Desain Uji Coba Data Masalah

Desain uji coba data zona bertujuan untuk menguji apakah fungsi CRU (*Create, Read, Update*) pada data masalah dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data masalah dapat dilihat pada tabel 3.19.

Tabel 3.19 Desain Uji Coba Data Masalah

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman <i>Master Masalah</i>	Memilih menu “ <i>Master Masalah</i> ”	Masuk ke halaman <i>Master Masalah</i>	
2	Memasukan data Zona	Nama masalah dan keterangan	Data masalah berhasil disimpan dan ditampilkan pada tabel data masalah	
3	Mengubah data masalah	Nama masalah dan keterangan	Data masalah berhasil diperbarui dan ditampilkan pada tabel data masalah	

**g. Desain Uji Coba Data Harapan**

Desain uji coba data harapan bertujuan untuk menguji apakah fungsi CRU (*Create, Read dan Update*) pada data harapan dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data harapan dapat dilihat pada tabel 3.20.

Tabel 3.20 Desain Uji Coba Data Harapan

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman <i>Master Harapan</i>	Memilih menu “ <i>Master Harapan</i> ”	Masuk ke halaman <i>Master Harapan</i>	
2	Memasukan data harapan	Nama masalah, nama harapan	Data harapan berhasil disimpan dan ditampilkan pada tabel data harapan	
3	Mengubah data harapan	Nama masalah, nama harapan	Data harapan berhasil diperbarui dan ditampilkan pada tabel data harapan	

**h. Desain Uji Coba Data Penjadwalan Kunjungan**

Desain uji coba data penjadwalan kunjungan bertujuan untuk menguji apakah fungsi membuat, mengubah, dan menghapus pada data penjadwalan kunjungan dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data penjadwalan kunjungan dapat dilihat pada tabel 3.21.

Tabel 3.21 Desain Uji Coba Data Penjadwalan Kunjungan

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman Manajemen Jadwal Kunjungan	Memilih menu “Buat Jadwal”	Masuk ke halaman Manajemen Jadwal Kunjungan	
2	Membuat Jadwal Kunjungan	Nomor zona, nama sales, subjek kunjungan, lokasi kunjungan, deskripsi kunjungan dan tanggal kunjungan	Data Jadwal Kunjungan berhasil disimpan dan ditampilkan pada tabel daftar jadwal kunjungan	
3	Menampilkan detail jadwal kunjungan	Menekan <i>button</i> “view”	Tampil <i>detail</i> jadwal kunjungan	
4	Mengubah data jadwal kunjungan	subjek kunjungan, deskripsi kunjungan dan tanggal kunjungan	Data jadwal kunjungan berhasil diperbarui dan ditampilkan pada tabel daftar jadwal kunjungan	

### i. Desain Uji Coba Data Hasil Kunjungan

Desain uji coba data hasil kunjungan bertujuan untuk menguji apakah fungsi membuat laporan kunjungan dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data hasil kunjungan dapat dilihat pada tabel 3.22.

Tabel 3.22 Desain Uji Coba Data Hasil Kunjungan

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman hasil kunjungan	Memilih menu “Buat Laporan”	Masuk ke halaman hasil kunjungan	
2	Membuat data laporan hasil kunjungan	Id laporan, nama zona, id kunjungan, <i>delivery habit, wallet share, harga competitor, program competitor, masalah, harapan, souvenir dan pilih file</i>	Data laporan hasil kunjungan berhasil disimpan dan ditampilkan pada tabel data hasil kunjungan	
3	Mengubah data laporan hasil kunjungan	<i>Delivery habit, wallet share, harga competitor, program competitor, masalah,</i>	Data laporan hasil kunjungan berhasil diperbarui dan ditampilkan pada tabel data hasil kunjungan	

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
		harapan, <i>souvenir</i> dan pilih file		

**j. Desain Uji Coba Data Laporan Kunjungan**

Desain uji coba data laporan kunjungan bertujuan untuk menguji apakah fungsi menampilkan laporan kunjungan dan mencetak laporan kunjungan dapat berjalan sesuai dengan yang diharapkan. Desain uji coba data laporan kunjungan dapat dilihat pada tabel 3.23.

Tabel 3.23 Desain Uji Coba Data Laporan Kunjungan

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan halaman <i>History</i> Kunjungan	Memilih menu “ <i>History</i> Kunjungan”	Masuk ke halaman <i>History</i> Kunjungan	
2	Mencetak Laporan Kunjungan	Menekan tombol <i>icon</i> pdf	Tampil Laporan Kunjungan dan dapat dicetak	

**k. Desain Uji Coba *Monitoring***

Desain uji coba *monitoring* bertujuan untuk menguji apakah fungsi menampilkan grafik kinerja sales dan pelanggan dapat berjalan sesuai dengan yang diharapkan. Desain uji coba *monitoring* dapat dilihat pada tabel 3.24.

Tabel 3.24 Desain Uji Coba *Monitoring*

<i>Test Case ID</i>	<i>Tujuan</i>	<i>Input</i>	<i>Output yang diharapkan</i>	<i>Status</i>
1	Menampilkan grafik kinerja <i>sales</i>	Memilih menu <i>Monitoring sales</i>	Tampil grafik kinerja <i>sales</i>	
2	Menampilkan grafik kinerja pelanggan	Memilih menu <i>Monitoring pelanggan</i>	Tampil grafik kinerja pelanggan	

