

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Identifikasi Permasalahan

Langkah awal dalam pembuatan sistem adalah mengidentifikasi permasalahan yang ada pada saat proses monitoring kegiatan tanggap darurat bencana untuk membuat sebuah solusi yang disajikan dalam bentuk aplikasi. Langkah identifikasi dilakukan dengan cara observasi dan wawancara langsung ke Palang Merah Indonesia Kota Surabaya, sehingga bisa dilakukan tindakan solusi yang tepat untuk menyelesaikan masalah tersebut.

Apabila terjadi bencana, Palang Merah Indonesia memberangkatkan tim SATGANA. Tim yang diberangkatkan harus membuat laporan atas bencana yang telah terjadi. Permasalahan yang terjadi di Palang Merah Indonesia saat ini yaitu jika tim diberangkatkan, tim SATGANA tidak termonitor dengan baik, untuk surat tugas seringkali terlambat untuk diberikan, dan assessment yang seharusnya segera dilaporkan setidaknya 30 menit setelah sampai di lokasi, sering mengalami keterlambatan atau bahkan tidak diberikan. Permasalah kedua yaitu dari setiap laporan yang ditulis, tidak tersimpan dengan baik, sehingga banyak beberapa laporan yang hilang ataupun rusak. Laporan ini juga tidak diolah dan menghasilkan informasi yang diharapkan oleh pihak staff bidang penanggulangan bencana.

3.2 Analisis Permasalahan

Berdasarkan identifikasi permasalahan, proses berikutnya adalah melakukan analisis permasalahan, proses ini dibutuhkan untuk mencari penyebab utama munculnya permasalahan pada saat penugasan dalam penanganan bencana di Kota Surabaya. Beberapa hal yang dapat mempengaruhi proses monitoring adalah sebagai berikut:

- A. Kurangnya informasi mengenai bencana yang dilaporkan oleh tim SIBAT, dan pada saat bencana diberitakan, sehingga pihak PMI terkadang harus mengkonfirmasi kembali ke pihak lurah setempat.
- B. Seringnya tim SATGANA tidak menerima surat tugas saat penugasan tanggap darurat bencana.
- C. Kurangnya informasi dan *assessment* yang dikirimkan ke PMI Kota Surabaya oleh tim SIBAT dan SATGANA.
- D. Kurangnya informasi yang dapat dihasilkan dari laporan-laporan terjadinya bencana yang telah dilaporkan oleh tim SATGANA

3.3 Perancangan Sistem

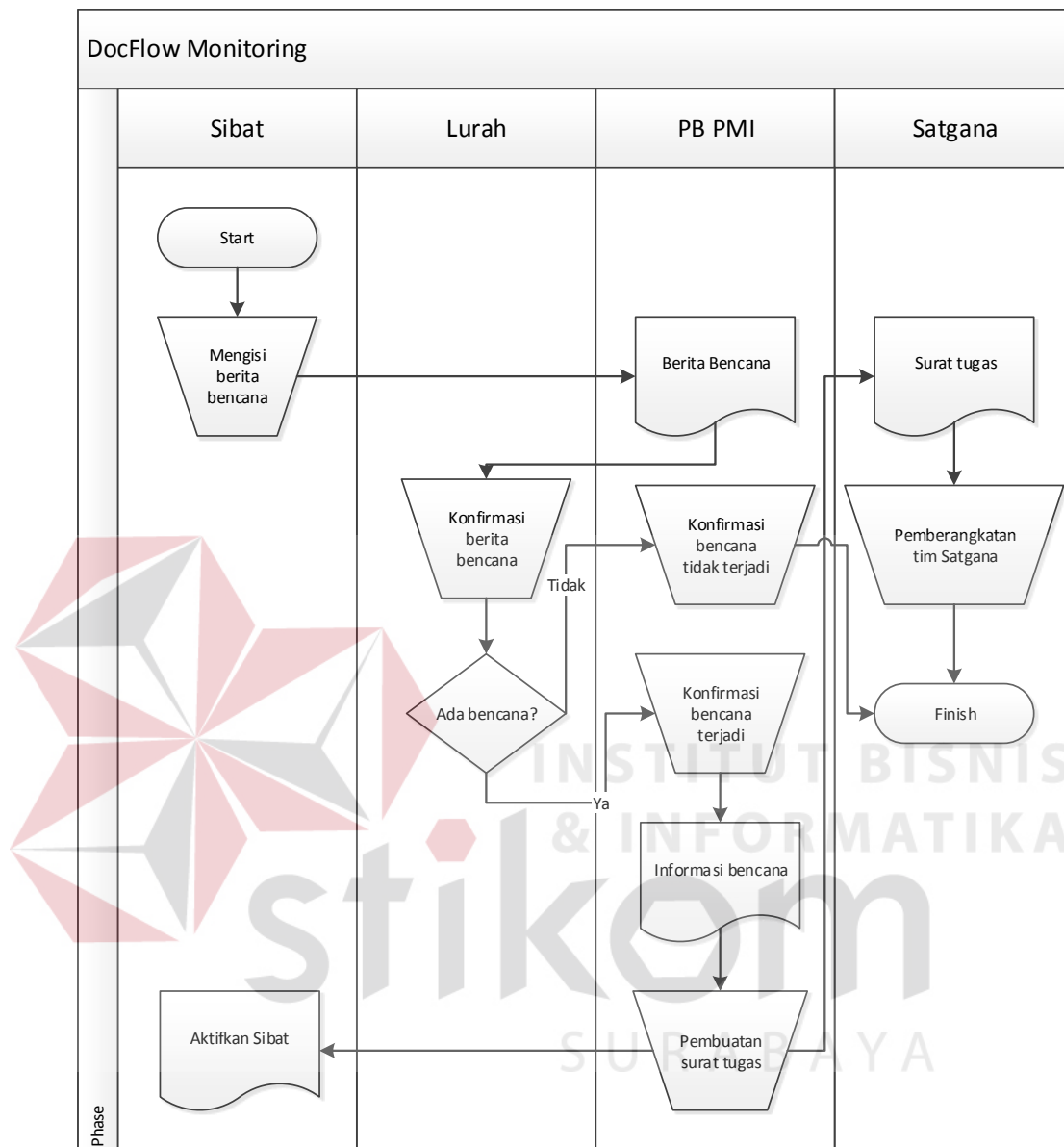
Setelah dilakukan analisis terhadap sistem, maka langkah selanjutnya adalah perancangan sistem. Perancangan sistem ini bertujuan untuk mendefinisikan kebutuhan-kebutuhan fungsional, menggambarkan aliran data dan alur sistem, dan sebagai tahap persiapan sebelum implementasi sistem. Perancangan sistem 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 sistem ini adalah sebagai berikut:

- a. *Document Flow*.
- b. Arsitektur Sistem.
- c. Blok Diagram.
- d. *System Flow*.
- e. Diagram *HIPO* (*Hirarchy Input Process Output*).
- f. *Data Flow Diagram (DFD)*, yang didalamnya terdapat: *context diagram*, *DFD Level 0*, dan *DFD Level 1*.
- g. *Entity Relationship Diagram (ERD)*, yang didalamnya meliputi: *Conceptual Data Model (CDM)*, dan *Physical Data Model (PDM)*.
- h. *Data Dictionary*.
- i. *Desain Input Output*.

3.3.1 *Document Flow* Penugasan Tim SATGANA

Document Flow adalah bagan yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. *Document Flow* juga digunakan untuk mengkomunikasikan aliran data dan prosedur proses informasi yang diperlukan dalam aplikasi. Gambar 3.1 adalah *document flow* penugasan tim SATGANA.



Gambar 3.1 *Document Flow* Penugasan Tim SATGANA

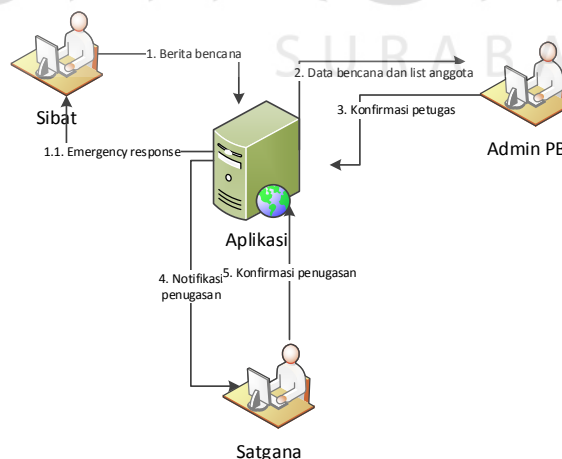
Proses monitoring dimulai saat suatu bencana terjadi di wilayah Surabaya, tim SIBAT akan menghubungi pihak PMI Surabaya, dan menginformasikan bahwa telah terjadi bencana pada wilayahnya. Tim SIBAT juga akan menginformasikan jenis bencana yang terjadi dan besar skala dari bencana tersebut. Setelah pihak PMI

menerima berita tersebut, pihak PMI akan mencatat dan menghubungi lurah dari wilayah yang telah terkena bencana untuk menanyakan dan mengkonfirmasi apakah tempat tersebut memang mengalami bencana atau tidak. Apabila memang telah terjadi bencana, maka pihak PMI akan memerintahkan lurah untuk mengumpulkan tim SIBAT dan mengaktifkan sistem *emergency* sementara.

Disaat yang sama, pihak PMI akan mengolah berita bencana, dan mencari petugas SATGANA untuk diberangkatkan. Apabila tim SATGANA sudah terbentuk serta, maka pihak PMI akan membuatkan surat tugas untuk petugas yang berangkat.

3.3.2 Arsitektur Sistem

Arsitektur sistem menggambarkan konsep perencanaan dan pengoperasian dasar dari suatu sistem Komputer dan sebagai rancangan arsitektur kebutuhan aplikasi yang akan dibangun. Alur sistem didalamnya menjelaskan kebutuhan *input* dan *output* sistem. Arsitektur pada aplikasi yang dibuat dijelaskan pada Gambar 3.2.



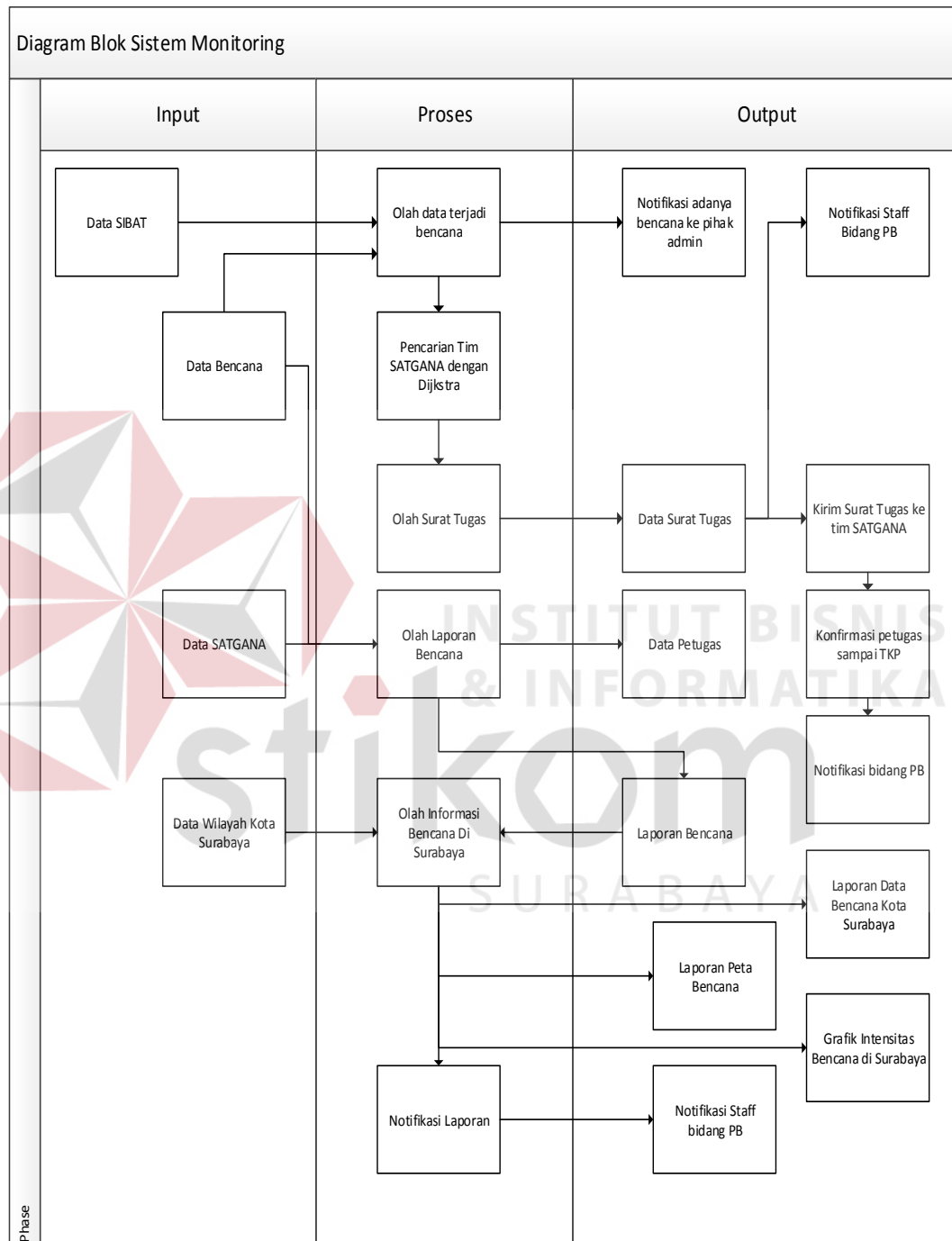
Gambar 3.2 Arsitektur Penugasan tim SATGANA

Proses penugasan dimulai saat suatu bencana terjadi di wilayah Surabaya, tim SIBAT akan mengirimkan berita bencana secara *online* dari aplikasi yang ada, tim SIBAT akan mengisi form yang telah disediakan serta melakukan *upload* gambar terjadinya bencana di wilayah tersebut. Pada saat yang sama setelah data dikirimkan, aplikasi akan mengirimkan notifikasi *emergency* pada semua tim SIBAT wilayah tersebut. Kemudian dikirimkan juga notifikasi bahwa telah terjadi bencana kepada *admin* PB PMI, notifikasi ini berisi data-data bencana yang terjadi, dan *list* petugas tim SATGANA yang telah diproses oleh aplikasi dengan menggunakan metode *dijkstra*.

Setelah *admin* PB mengkonfirmasi petugas yang akan diberangkatkan, aplikasi akan mengirimkan notifikasi pada tim SATGANA yang ditugaskan, disini tim SATGANA juga akan mendapat proses *generate* surat tugas yang akan dikirimkan melalui email tim tersebut. Setelah tim SATGANA mendapatkan berita penugasan dan surat tugas terkait, maka tim akan berangkat ke lokasi bencana, dan apabila sudah sampai di lokasi bencana, tim SATGANA akan mengkonfirmasi dan mengirimkan data *assessment* bencana.

Setelah bencana telah usai, maka tim SATGANA harus melaporkan dari bencana yang telah ditangani. Laporan-laporan ini nantinya akan menghasilkan beberapa informasi untuk pihak PB. Informasi tersebut antara lain, seluruh kegiatan tanggap darurat yang telah dilakukan oleh PMI Kota Surabaya, peta bencana, dan penugasan-penugasan tim SATGANA yang bersangkutan. Informasi-informasi ini akan menjadi pendukung keputusan untuk perlu tidaknya memberikan sosialisasi ke wilayah yang terkena bencana.

3.3.3 Blok Diagram



Gambar 3.3 Blok Diagram

Gambar 3.3 berguna untuk menggambarkan garis besar yang menjadi *input*, proses, dan *output* dalam Monitoring Penanggulangan Bencana di Palang Merah Indonesia Kota Surabaya.

a. Input

Pada sistem ini terdapat masukan (*input*) yang berasal dari data master, diantaranya:

1. Data SIBAT

Data merupakan data dari masing-masing relawan Palang Merah Indonesia Kota Surabaya yang telah menjadi anggota tim SIBAT.

2. Data Tim SATGANA

Data merupakan data dari masing-masing relawan Palang Merah Indonesia Kota Surabaya yang telah menjadi anggota tim SATGANA.

3. Data Bencana

Data ini merupakan laporan data-data bencana yang telah terjadi di kota Surabaya.

4. Data Wilayah

Dengan bantuan *Google Maps API*, data wilayah ini akan berfungsi saat anggota SATGANA melaporkan adanya bencana. Data wilayah ini akan secara otomatis menunjukkan wilayah yang terkena bencana tersebut.

b. Proses

Data yang berasal dari *input* di atas, kemudian diolah dan diproses

yang kemudian akan menghasilkan suatu *output*. Proses-proses pada aplikasi penugasan ini sebagai berikut:

1. Proses data terjadi bencana

Pada proses ini aplikasi akan menyimpan dan membaca data-data yang telah dikirimkan tim SIBAT untuk selanjutnya diproses dalam memberikan notifikasi dari bencana yang terjadi kepada admin.

2. Proses olah surat tugas

Pada proses ini, aplikasi akan membuat surat tugas yang ditujukan kepada tim yang diberangkatkan

3. Proses olah laporan bencana

Pada proses ini aplikasi akan menyimpan dan membaca data-data yang telah dikirimkan tim SATGANA untuk selanjutnya diproses dalam mengolah laporan bencana yang terjadi.

4. Proses olah informasi bencana di Surabaya

Proses ini merupakan proses lanjutan dari proses laporan bencana, proses ini untuk mengolah data-data dari semua laporan bencana yang terjadi di kota Surabaya. Proses ini akan menghasilkan beberapa informasi, yaitu informasi laporan bencana di Surabaya, dan peta bencana.

5. Proses notifikasi

Proses ini untuk mengirimkan notifikasi kepada *staff* bidang penanggulangan bencana bahwa laporan dari bencana yang terjadi

sudah diinputkan, dan informasi serta peta bencana di Surabaya telah di *update*.

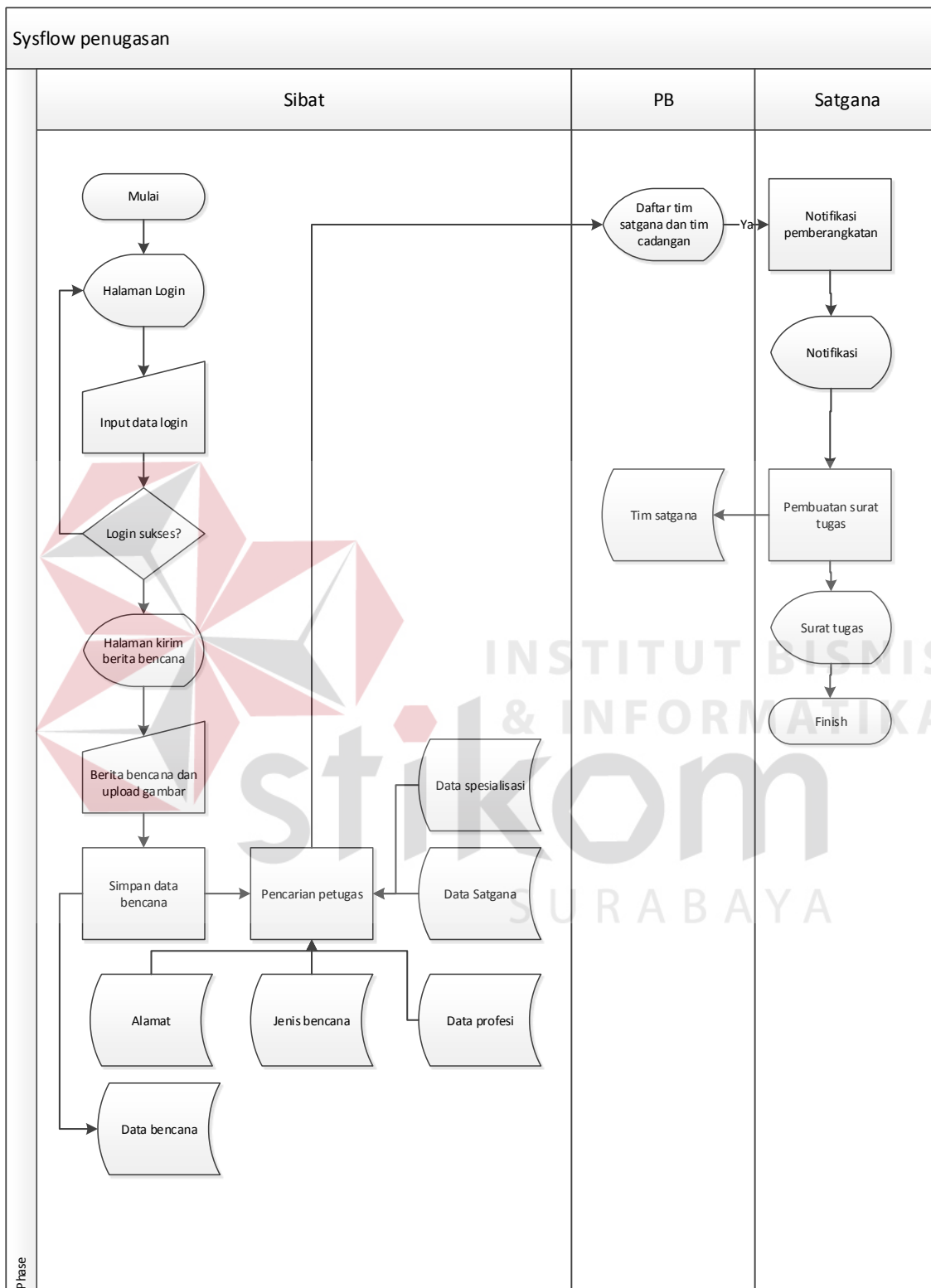
c. *Output*

Output yang dihasilkan oleh aplikasi dari proses-proses di atas yaitu berupa daftar dan informasi sebagai berikut:

1. Notifikasi adanya bencana yang telah terjadi
2. Data surat tugas untuk petugas yang diberangkatkan
3. Informasi tim SATGANA yang bertugas.
4. Informasi laporan bencana yang telah terjadi
5. Informasi seluruh bencana yang telah terjadi di Kota Surabaya
6. Informasi peta bencana Surabaya
7. Heatmap bencana di kota Surabaya
8. Pengiriman notifikasi kepada *staff* bidang PB

3.3.4 *System Flow* Penugasan Tim SATGANA

System Flow adalah penggambaran aliran dokumen dalam sistem dan merupakan proses kerja dalam sistem. *System Flow* ini juga representasi aliran data lanjutan dari *document flow*. Jika *document flow* menggambarkan aliran data secara manual atau yang selama ini terjadi diorganisasi, maka *system flow* ini menggambarkan aliran data pada sistem yang nantinya dibangun pada aplikasi.



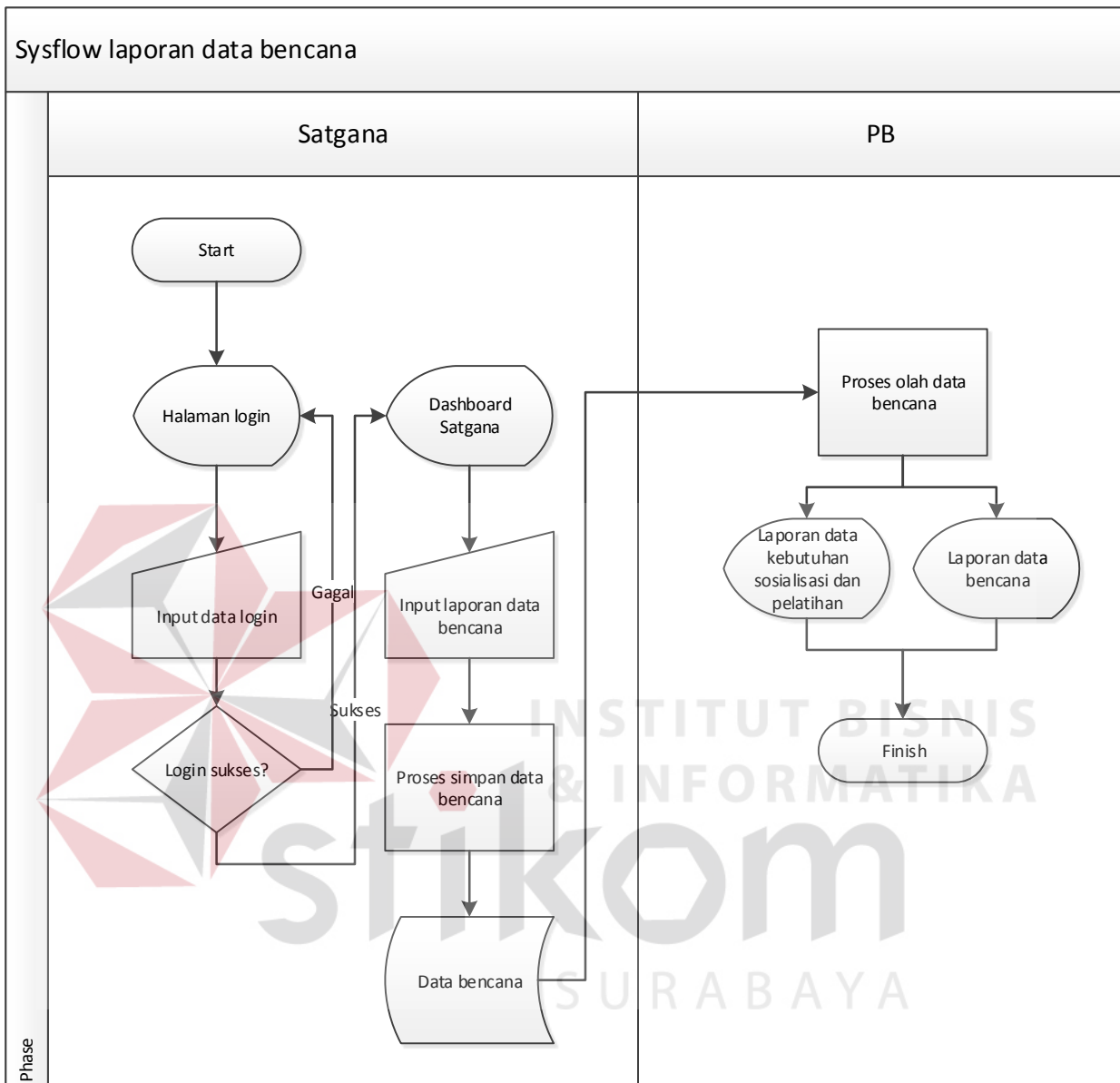
Gambar 3.4 *System Flow* Penugasan tim SATGANA

Gambar 3.4, proses penugasan dimulai saat suatu bencana terjadi di wilayah Surabaya, tim SIBAT akan mengirimkan berita bencana secara *online* dari aplikasi yang ada, tim SIBAT akan mengisi form yang telah disediakan serta melakukan *upload* gambar terjadinya bencana di wilayah tersebut. Pada saat yang sama setelah data dikirimkan, aplikasi akan mengirimkan notifikasi *emergency* pada semua tim SIBAT wilayah tersebut. Kemudian dikirimkan juga notifikasi bahwa telah terjadi bencana kepada *admin* PB PMI, notifikasi ini berisi data-data bencana yang terjadi, dan *list* petugas tim SATGANA yang telah diproses oleh aplikasi dengan menggunakan metode *dijkstra*.

Setelah *admin* PB mengkonfirmasi petugas yang akan diberangkatkan, aplikasi akan mengirimkan notifikasi pada tim SATGANA yang ditugaskan. Disaat yang sama saat petugas diberangkatkan, maka akan ada notifikasi yang dikirimkan kepada *admin* PB dan proses *generate* surat tugas yang akan dikirimkan melalui email petugas tersebut.

3.3.5 System Flow Laporan Data Bencana

System Flow selanjutnya yang akan dibuat adalah *System flow* laporan data bencana. *System flow* ini menggambarkan proses-proses yang terdapat pada pelaporan bencana. Gambar 3.5 adalah *System Flow* Laporan Data Bencana.



Gambar 3.5 *System Flow* Pelaporan tim SATGANA

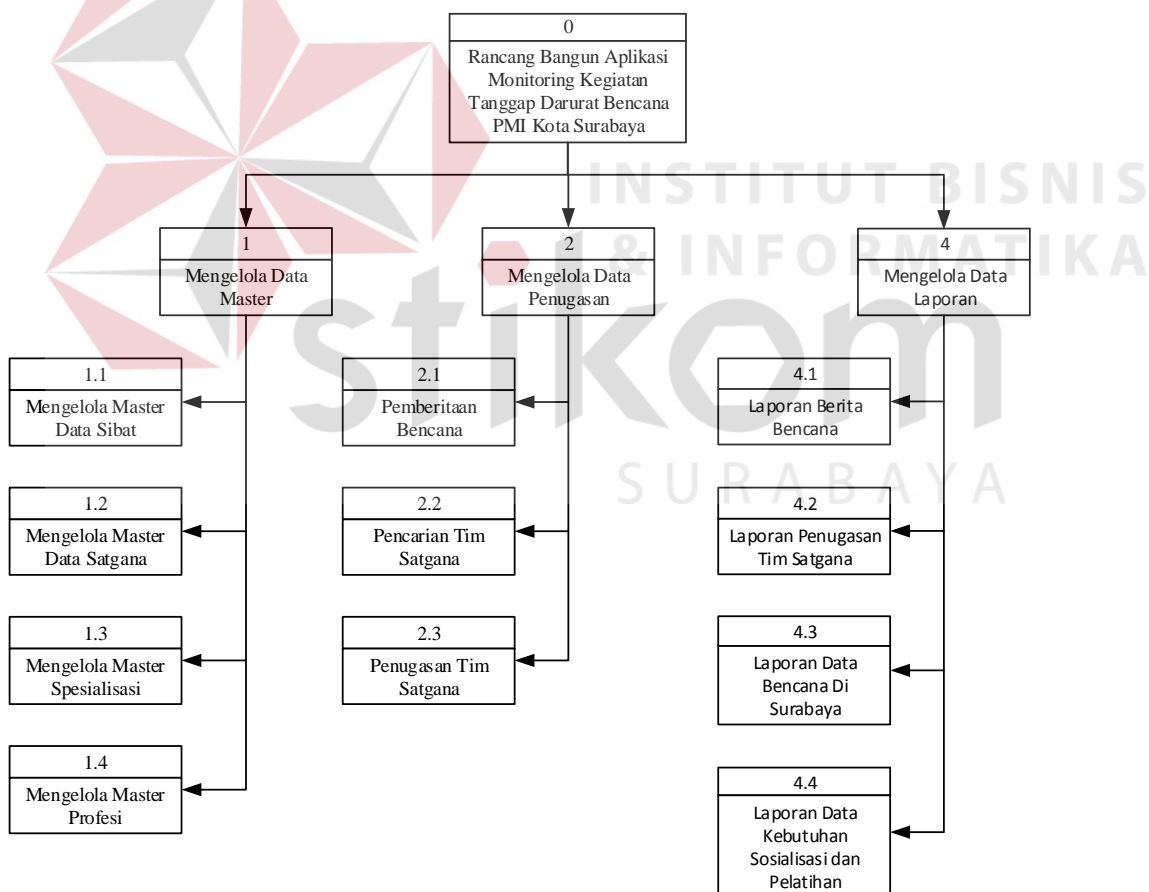
Pada Gambar 3.5, proses pelaporan dimulai saat tim SATGANA yang telah dikirimkan untuk menanggulangi bencana telah kembali. Petugas yang bertindak sebagai komandan, akan melakukan *login* kedalam aplikasi dan memasukkan data-

data bencana yang telah terjadi. Petugas juga bisa mengisi formulir dan membuat laporan dari bencana tersebut.

Dari laporan-laporan ini, aplikasi dapat mengolah data bencana dan pihak admin PB bisa mendapatkan informasi tentang bencana-bencana yang telah terjadi di kota Surabaya.

3.3.6 Diagram *HIPO*

Diagram *HIPO* menggambarkan hirarki proses-proses yang ada di dalam aplikasi penugasan SATGANA. Diagram ini digambarkan pada Gambar 3.6:



Gambar 3.6 Diagram *HIPO*

Diagram *HIPO* diatas menunjukkan tiga proses utama dalam sistem yaitu : pertama, mengelola data master, berguna untuk *input* atau *update* data master. Kedua, mengelola data penugasan yang berfungsi dalam memberangkatkan tim SATGANA. Proses ini terdiri dari Pemberitaan bencana yang telah diinformasikan oleh tim SIBAT, Pencarian tim SATGANA yang dilakukan secara otomatis dengan metode *dijkstra*, dan penugasan tim SATGANA. Ketiga, proses mengelola laporan yang didalamnya terdapat proses membuat laporan dari data bencana yang telah terjadi, mengelola data petugas yang telah berangkat, mengelola semua data bencana berskala besar yang telah terjadi di Surabaya berdasarkan periode waktu tertentu, dan mengelola laporan untuk wilayah-wilayah yang sekiranya membutuhkan sosialisasi atau pelatihan tentang siaga bencana.

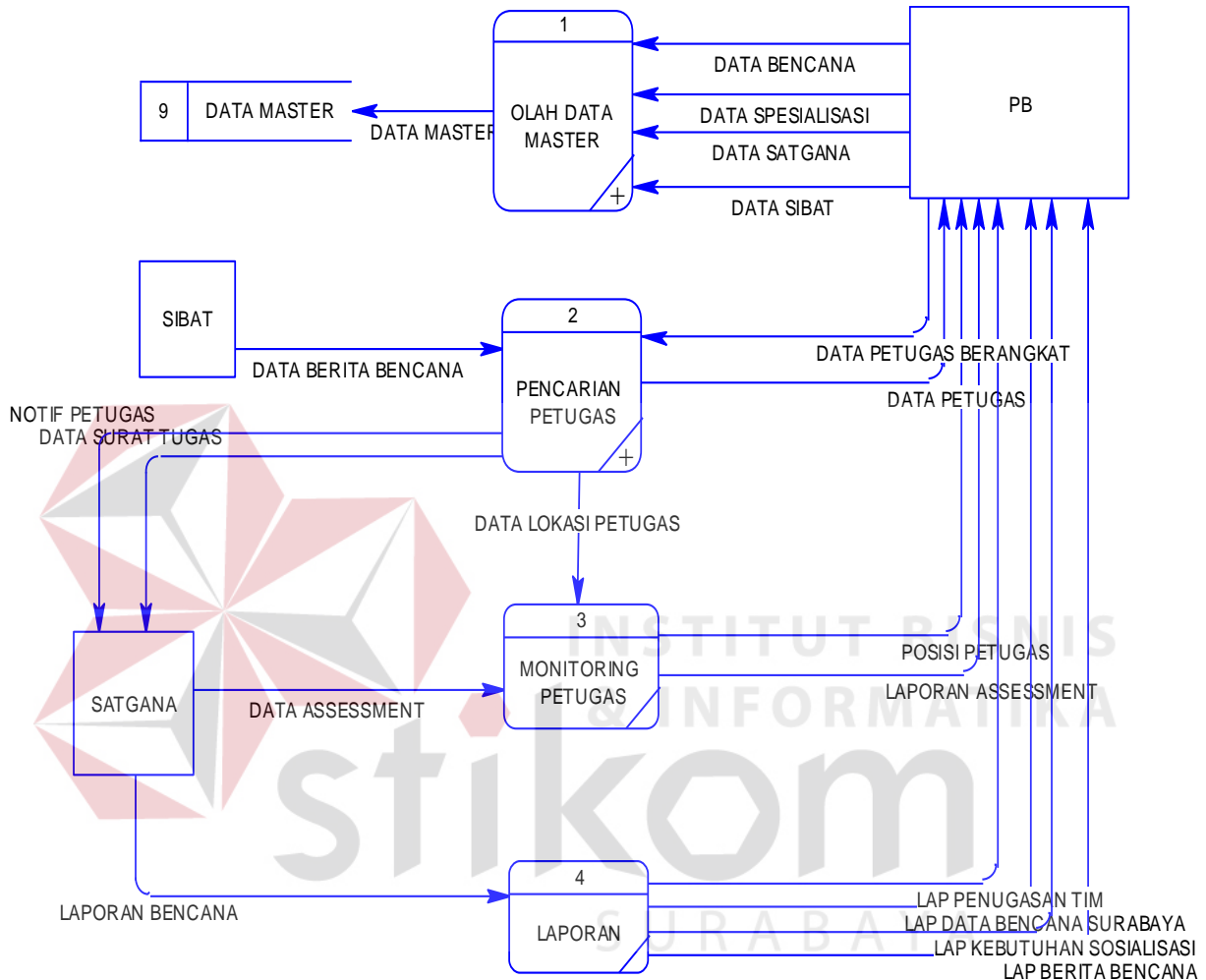
3.3.7 Data Flow Diagram

Data Flow Diagram (DFD) merupakan gambaran aliran data yang terdapat dalam sistem. Diagram ini menjelaskan secara lebih detail tentang proses yang terdapat pada diagram. *DFD* ini berfungsi untuk menggambarkan proses aliran data yang terjadi di dalam sistem mulai dari yang paling tinggi sampai yang paling rendah, sehingga nantinya akan dimungkinkan proses dekomposisi, partisi, atau pembagian sistem ke dalam bagian-bagian yang lebih kecil dan lebih sederhana. Adapun penjelasan dari *DFD* tersebut dapat dilihat sebagai berikut :

A. Context Diagram

Context Diagram merupakan diagram pertama dalam rangkaian *DFD* yang menunjukkan entitas-entitas yang berhubungan dengan sistem. Diagram ini juga akan

mengelola laporan. Entitas yang saling berhubungan pada *DFD Level 0* adalah SIBAT, admin PB dan SATGANA. Gambar 3.8 adalah gambaran *DFD Level 0* :

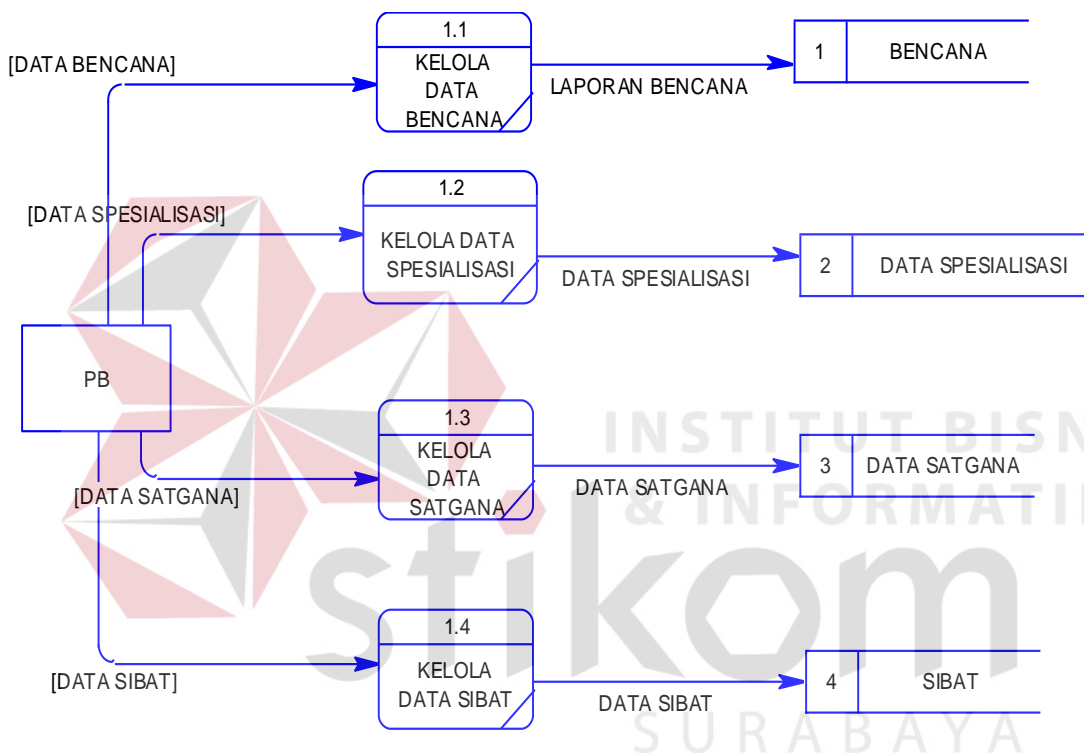


Gambar 3.8 *DFD Level 0*

Pada *DFD Level 0*, terdapat empat proses utama yang digunakan di dalam Aplikasi. Setiap proses memiliki inputan data yang berasal dari entitas yang bersangkutan. Selanjutnya, empat proses tersebut juga dijelaskan lebih detail kedalam *DFD Level 1*.

C. DFD Level 1 Mengelola Data Master

Pada *DFD Level 1* mengelola data master ini adalah *decompose* dari proses mengelola data master yang terdapat pada *DFD Level 0*. Proses ini menjelaskan *input* data baru yang berasal dari entitas dan disimpan ke dalam *data store*. Gambar 3.9 adalah gambaran proses *DFD Level 1* :

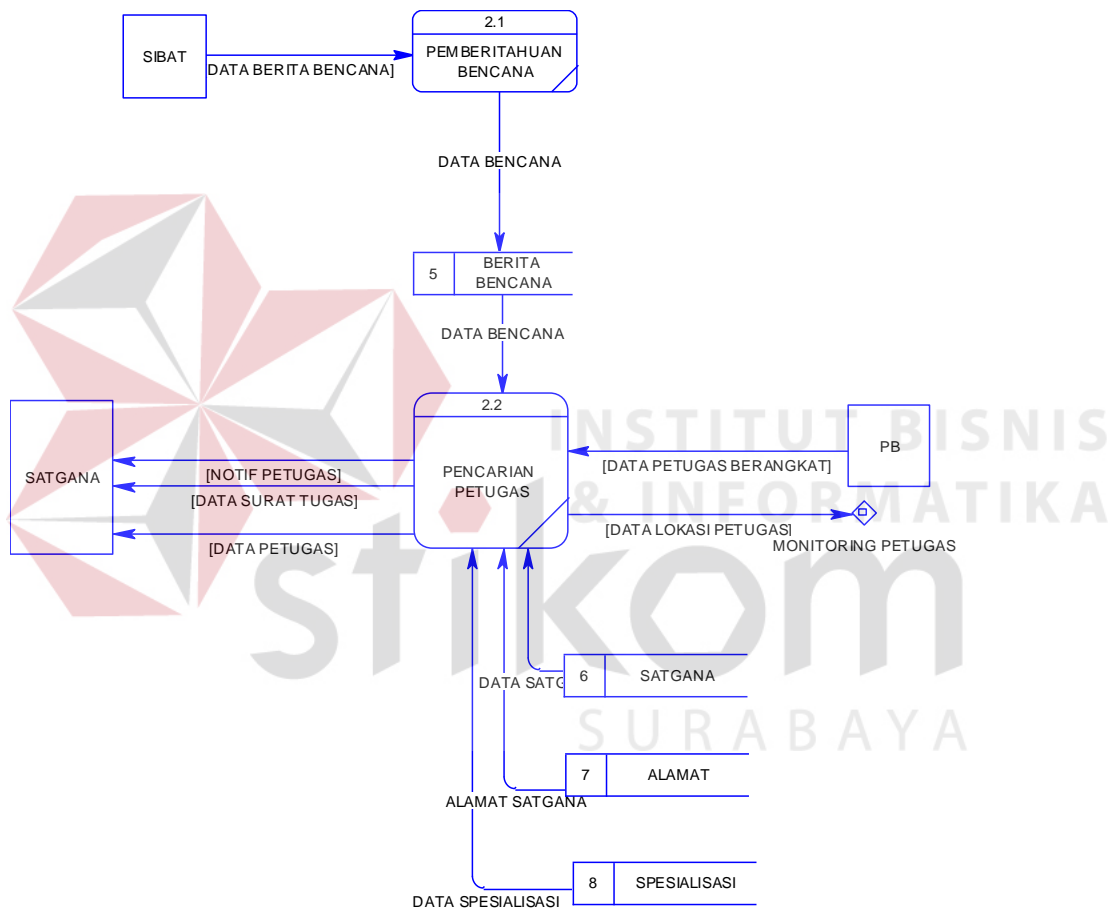


Gambar 3.9 DFD Level 1 Mengelola Data Master

Dari gambar diatas terdapat ada empat proses yang berfungsi mengelola data master, yaitu : mengelola data SIBAT, mengelola data SATGANA, megelola data spesialisasi, dan mengelola data profesi. Selanjutnya masing-masing proses tersebut menyimpan data ke dalam *data store*.

D. DFD Level 1 Proses Penugasan Tim

Pada *DFD Level 1* mengelola data master ini adalah *decompose* dari proses mengelola data master yang terdapat pada *DFD Level 0*. Proses ini menjelaskan *input* data baru yang berasal dari entitas dan disimpan ke dalam *data store*. Gambar 3.10 adalah gambaran proses *DFD Level 1* :



Gambar 3.10 DFD Level 1 Proses penugasan tim

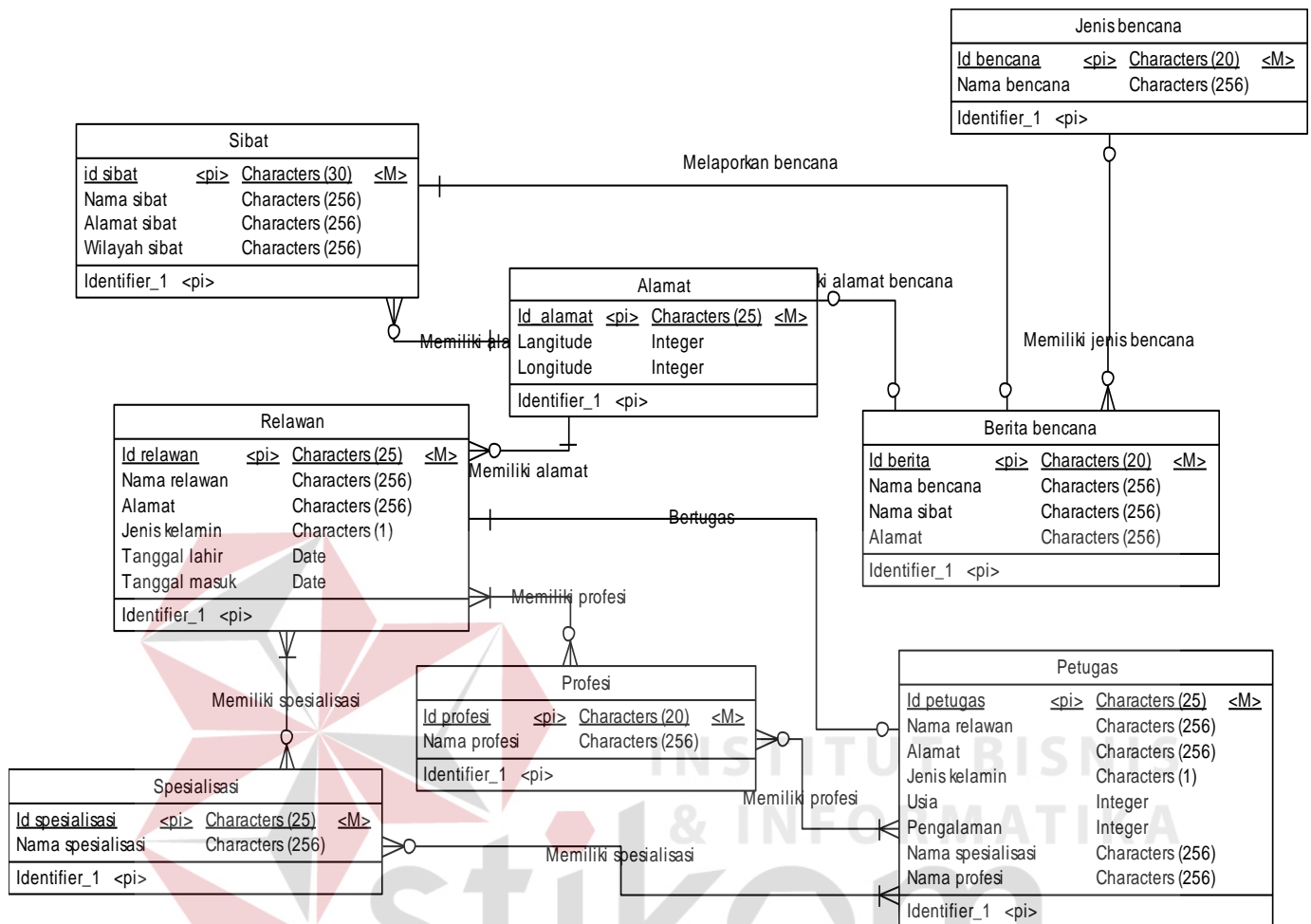
Dari Gambar 3.9 terdapat ada dua proses yang berfungsi mengelola data master, yaitu : pemberitahuan bencana dan pencarian petugas. Pada masing-masing proses tersebut terdapat proses simpan dan *read* data dari *data store*.

3.3.8 Entity Relationship Diagram

Entity Relationship Diagram atau biasa disingkat *ERD* adalah suatu desain sistem yang digunakan untuk menggambarkan kebutuhan tabel dalam sistem. Tabel ini akan digambarkan dalam bentuk *entity* dan memiliki atribut serta saling berhubungan atau relasi satu sama lain. Penggambaran *ERD* lebih jelasnya adalah sebagai berikut :

A. Conceptual Data Model

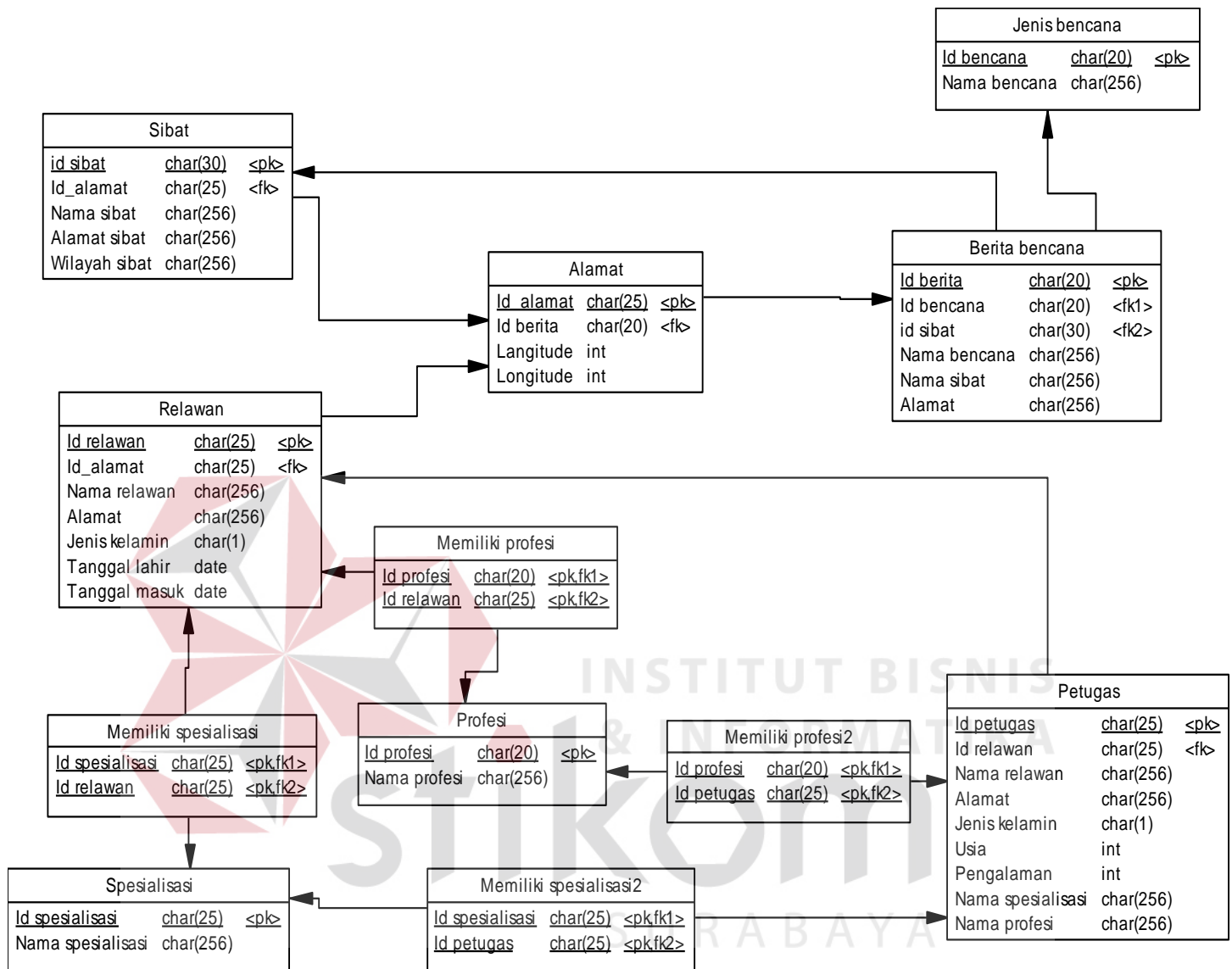
Conceptual Data Model (CDM) menggambarkan secara keseluruhan konsep struktur basis data yang dirancang pada suatu sistem. Pada *CDM* ini sudah terdapat beberapa atribut di setiap tabel yang digunakan untuk menampung data yang terkait didalamnya. Seperti yang terlihat pada gambar, bahwa ada 8 tabel yang saling berelasi satu sama lain, yaitu tabel SIBAT, tabel relawan, tabel alamat, tabel spesialisasi, tabel profesi, tabel petugas, tabel berita bencana, dan tabel jenis bencana. Desain CDM dapat dilihat pada Gambar 3.11.



Gambar 3. 11 Conceptual Data Model

B. Physical Data Model

Physical Data Model (PDM) menggambarkan secara detail tentang konsep struktur basis data yang dirancang untuk suatu sistem. *PDM* merupakan hasil *generate* dari *CDM*. Pada *PDM* ini juga sudah tergambar jelas relasi antar tabelnya, dengan ditunjukkan *primary key* dan *foreign key* pada masing-masing tabel. Nantinya *PDM* ini digenerate untuk menghasilkan *database* dalam *Database Management System (DBMS)*. Desain *PDM* dapat dilihat pada Gambar 3.12.



Gambar 3. 12 Physical Data Model

3.3.9 Struktur Database

Struktur *database* merupakan uraian struktur fisik dari tabel-tabel yang terdapat pada *database*. Fungsinya adalah menyimpan data-data yang saling berhubungan. Adapun struktur *database* tersebut dapat dijelaskan lebih rinci sebagai berikut:

A. Tabel SIBAT

Nama tabel : SIBAT

Primary key : id_SIBAT

Foreign key : -

Fungsi : Menyimpan data SIBAT

Tabel 3.1 SIBAT

No	Field Name	Data Type	Length	Constraint
1	Id_SIBAT	Varchar	30	PK
2	Nama_SIBAT	Varchar	256	
3	Alamat_SIBAT	Varchar	256	
4	Wilayah_SIBAT	Varchar	256	

B. Tabel Alamat

Nama tabel : alamat

Primary key : id_alamat

Fungsi : Menyimpan data alamat

Tabel 3.2 Alamat

No	Field Name	Data Type	Length	Constraint
1	Id_alamat	Varchar	25	PK
2	Langitude	Integer		
3	Longitude	Integer		

C. Tabel Jenis Bencana

Nama tabel : jenis_bencana

Primary key : Id_bencana

Fungsi : Menyimpan data jenis bencana

Tabel 3.3 Jenis bencana

No	Field Name	Data Type	Length	Constraint
1	Id_bencana	Varchar	20	PK
2	Nama_bencana	Varchar	256	

D. Tabel Relawan

Nama tabel : relawan

Primary key : Id_relawan

Foreign key : -

Fungsi : Menyimpan data relawan

Tabel 3.4 Relawan

No	Field Name	Data Type	Length	Constraint
1	Id_relawan	Varchar	25	PK
2	Nama_relawan	Varchar	256	
3	Alamat	Varchar	256	
4	Jenis_kelamin	Varchar	1	
5	Tanggal_lahir	Date		
6	Tanggal_masuk	Date		

E. Tabel Spesialisasi

Nama tabel : spesialisasi

Primary key : Id_spesialisasi

Foreign key : -

Fungsi : Menyimpan data spesialisasi

Tabel 3.5 Spesialisasi

No	Field Name	Data Type	Length	Constraint
1	Id_spesialisasi	Varchar	25	PK
2	Nama_spesialisasi	Varchar	256	

F. Tabel Profesi

Nama tabel : Profesi

Primary key : Id_profesi

Foreign key : -

Fungsi : Menyimpan data profesi

Tabel 3.6 Profesi

No	Field Name	Data Type	Length	Constraint
1	Id_profesi	Varchar	20	PK
2	Nama_profesi	Varchar	256	

G. Tabel Berita Bencana

Nama tabel : Berita_bencana

Primary key : Id_berita

Foreign key : -

Fungsi : Menyimpan data berita bencana

Tabel 3.7 Berita bencana

No	Field Name	Data Type	Length	Constraint
1	Id_berita	Varchar	20	PK
2	Nama_bencana	Varchar	256	FK
3	Nama_SIBAT	Varchar	256	FK
4	Alamat	Varchar	256	FK

H. Tabel Petugas

Nama tabel : petugas

Primary key : Id_petugas

Foreign key : -

Fungsi : Menyimpan data petugas

Tabel 3.8 Data Petugas

No	Field Name	Data Type	Length	Constraint
1	Id_petugas	Varchar	25	PK
2	Nama_relawan	Varchar	256	
3	Alamat	Varchar	256	
4	Jenis_kelamin	Varchar	1	
5	Usia	Integer		
6	Pengalaman	Integer		
7	Nama_spesialisasi	Varchar	256	
8	Nama_profesi	Varchar	256	

3.3.10 Desain *Input Output*

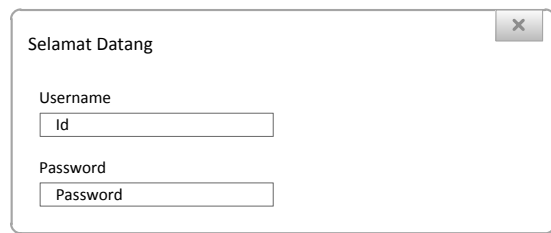
Desain *input output* adalah rancangan form-form yang diimplementasikan kedalam sistem dan berfungsi sebagai antar muka pengguna dengan sistem. Rancangan ini menerima *input* / masukan data dari pengguna dan memberikan hasilnya berupa *output* laporan. Adapun desain *input output* tersebut adalah sebagai berikut :

A. Desain *Input*

A.1 Desain Form *Login*

Form *Login* ini digunakan untuk keamanan sistem. Tujuannya adalah supaya sistem digunakan oleh orang yang berhak memakai dan berjalan sesuai hak aksesnya

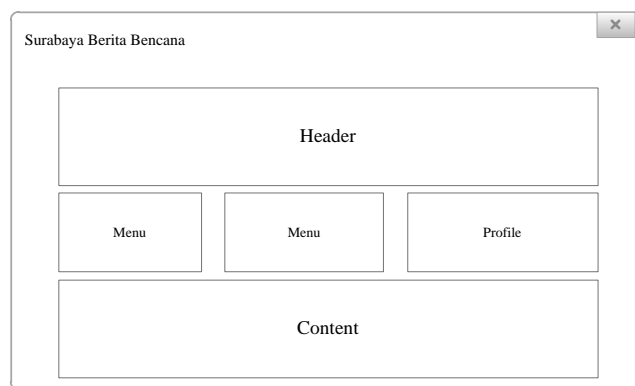
masing-masing. Adapun *field* yang harus diisi dalam form ini adalah *id* dan *password*. Pada form ini ada tombol login, dimana tombol ini berfungsi untuk secara otomatis memvalidasi pengguna tersebut dan mencari hak aksesnya, apakah sebagai SIBAT, admin PB atau tim SATGANA.



Gambar 3.13 Desain Form *Login*

A.2 Desain *Layout* Aplikasi

Berikut adalah desain *layout* aplikasi. Di dalam layout dibagi menjadi beberapa bagian, yaitu: *header*, menu, dan konten. Pada bagian *header* terdapat foto PMI serta berita terbaru mengenai bencana di Surabaya. Pada bagian menu terdapat beberapa menu yang digunakan untuk mengakses fitur-fitur yang ada di aplikasi, dan *profile* merupakan informasi dari user yang bersangkutan. Sedangkan pada bagian konten digunakan untuk menampilkan visi-misi PMI kota Surabaya.



Gambar 3.14 Desain *Layout* Aplikasi

A.3 Desain Form Pemberitaan Bencana

Form pemberitaan bencana digunakan oleh SIBAT untuk melakukan pemberitahuan jika telah terjadi bencana di wilayahnya. Di dalam form ini terdapat beberapa isian untuk mengirimkan berita bencana yang telah terjadi.

Form Berita Bencana

Cek Posisi

Tanggal/Jam :

Detail Form Bencana

Berita Bencana

Alamat Lokasi

Jenis Bencana

Keterangan

Upload Gambar

Click to upload..

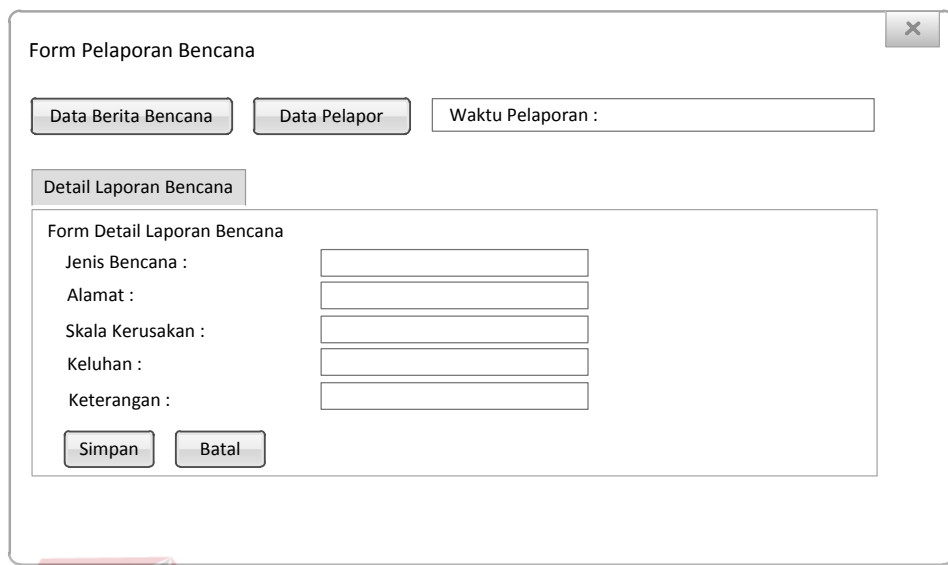
Kirim

Batal

Gambar 3.15 Desain Form Pemberitaan Bencana

A.4 Desain Pelaporan Bencana

Form pelaporan bencana digunakan oleh SATGANA untuk melaporkan bencana yang telah terjadi. Di dalam form ini terdapat beberapa isian yang merupakan syarat-syarat untuk melakukan pelaporan bencana. Untuk menyimpan data digunakan tombol simpan setelah semua *field* terisi.



Form Pelaporan Bencana

Data Berita Bencana Data Pelapor Waktu Pelaporan :

Detail Laporan Bencana

Form Detail Laporan Bencana

Jenis Bencana :

Alamat :

Skala Kerusakan :

Keluhan :

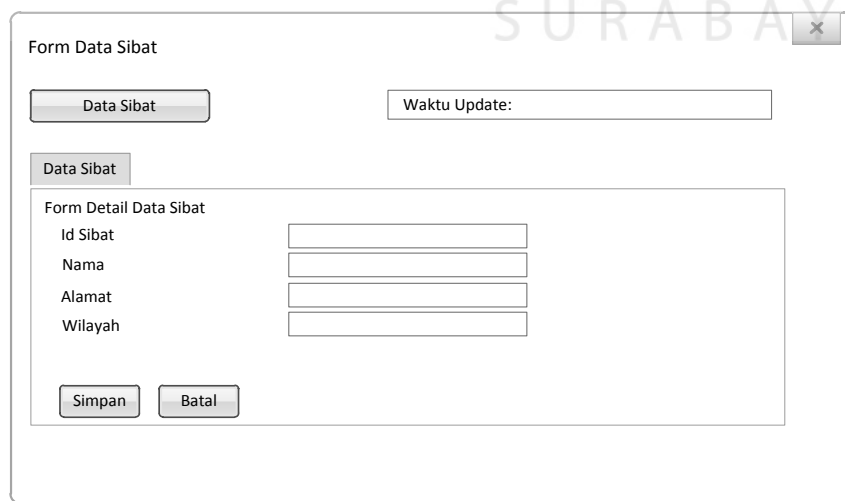
Keterangan :

Simpan Batal

Gambar 3.16 Desain Form Pelaporan Bencana

A.5 Desain Form *Input* Data SIBAT

Form ini digunakan oleh admin PB untuk melakukan *update* maupun menambahkan data pada database SIBAT. Di dalam form ini terdapat beberapa isian yang merupakan masukan untuk *update* data SIBAT. Untuk menyimpan data digunakan tombol simpan setelah semua *field* terisi.



Form Data Sibat

Data Sibat Waktu Update:

Data Sibat

Form Detail Data Sibat

Id Sibat

Nama

Alamat

Wilayah

Simpan Batal

Gambar 3.17 Desain Form Input Data SIBAT

A.6 Desain Form *Input* Data SATGANA

Form ini digunakan oleh admin PB untuk melakukan *update* maupun menambahkan data pada database SATGANA. Di dalam form ini terdapat beberapa isian yang merupakan masukan untuk *update* data SATGANA. Untuk menyimpan data digunakan tombol simpan setelah semua *field* terisi.

Gambar 3.18 Desain Form *Input* Data SATGANA

A.7 Desain Form *Input* Data Spesialisasi

Form ini digunakan oleh admin PB untuk melakukan *update* maupun menambahkan data pada database Spesialisasi. Di dalam form ini terdapat beberapa isian yang merupakan masukan untuk *update* data Spesialisasi. Untuk menyimpan data digunakan tombol simpan setelah semua *field* terisi.

Gambar 3.19 Desain Form Spesialisasi

A.8 Desain Form *Input* Data Profesi

Form ini digunakan oleh admin PB untuk melakukan *update* maupun menambahkan data pada database Profesi. Di dalam form ini terdapat beberapa isian yang merupakan masukan untuk *update* data Profesi. Untuk menyimpan data digunakan tombol simpan setelah semua *field* terisi.

Gambar 3.20 Desain Form *Input* Data Profesi

B. Desain *Output*

B.1 Desain Notifikasi Bencana Pada Admin PB

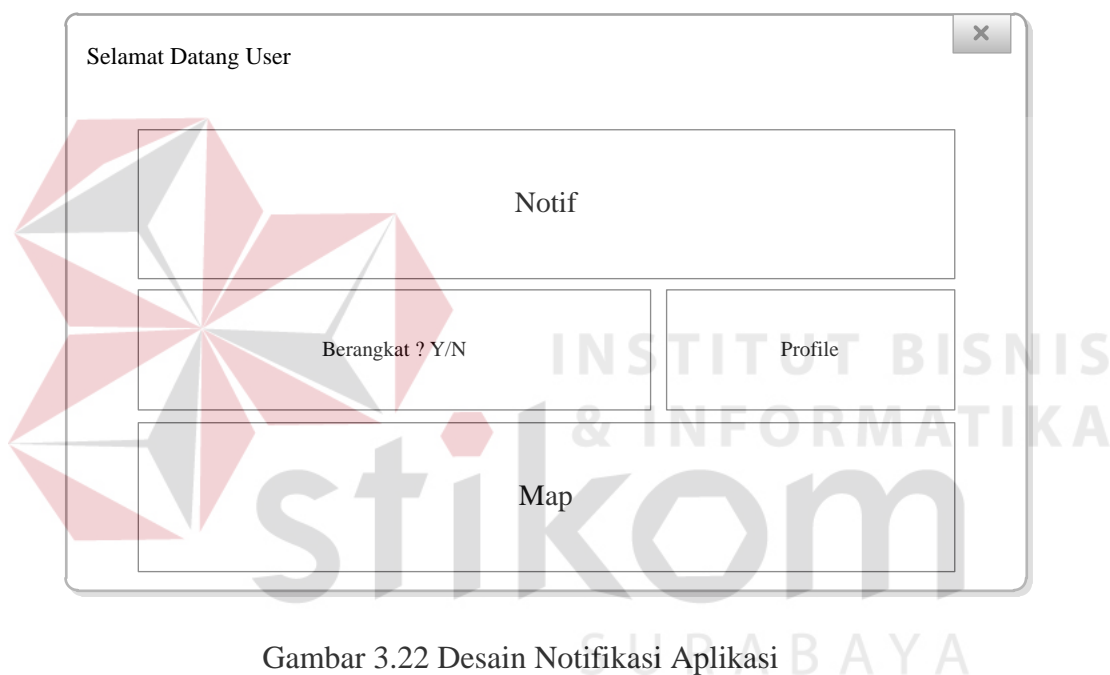
Desain ini merupakan hasil tampilan dari notifikasi dan data petugas yang diterima oleh admin PB saat bencana diinformasikan oleh tim SIBAT. Data petugas yang ada, telah diproses oleh aplikasi dengan metode *dijkstra*. Di dalam tampilan ini dibagi menjadi beberapa bagian, yaitu: *notif*, petugas, petugas cadangan, dan *map*. Pada bagian *notif* terdapat berita bencana yang telah dilaporkan. Pada bagian petugas, merupakan tim SATGANA yang telah dipilih dengan metode, dan petugas cadangan merupakan tim SATGANA yang disiapkan apabila tim utama tidak dapat bertugas, dan *profile* merupakan informasi dari user yang bersangkutan. Sedangkan pada bagian *map* digunakan untuk menampilkan peta tempat terjadi bencana dan lokasi petugas yang diberangkatkan.



Gambar 3.21 Desain Notifikasi Aplikasi

B.2 Desain Notifikasi Penugasan

Desain ini merupakan hasil tampilan dari notifikasi yang dikirimkan admin PB kepada tim SATGANA. Di dalam tampilan ini dibagi menjadi beberapa bagian, yaitu: *notif*, *option*, dan *map*. Pada bagian *notif* terdapat berita bencana yang telah dilaporkan. Sedangkan pada bagian *map* digunakan untuk menampilkan peta tempat terjadi bencana dan lokasi petugas yang akan diberangkatkan.



Gambar 3.22 Desain Notifikasi Aplikasi

3.3.11 Desain Uji Coba

Pengujian sistem dilakukan dengan cara melakukan berbagai percobaan terhadap beberapa fungsi yang tersedia untuk membuktikan bahwa aplikasi telah berjalan sesuai dengan tujuan. Pengujian sistem ini menggunakan metode *Black Box Testing*. Berikut ini adalah perancangan uji coba pada aplikasi penugasan :

Tabel 3.9 Desain Uji Coba Aplikasi

No	Nama Tes	Proses	Input	Output yang diharapkan
1	Uji coba Form <i>Login</i>	Login aplikasi	Id, password	Pengguna bisa mengakses aplikasi
2	Uji coba pemberitaan bencana	Simpan data berita bencana	Berita bencana	Data berita bencana dapat tersimpan
		Notifikasi bencana	Berita bencana	Admin PB mendapatkan notifikasi dari SIBAT
3	Uji coba mengelola data master	Simpan data SIBAT	Data SIBAT	Data pada <i>database</i> terupdate
		Simpan data SATGANA	Data SATGANA	Data pada <i>database</i> terupdate
		Simpan data spesialisasi	Data spesialisasi	Data pada <i>database</i> terupdate
		Simpan data bencana	Data bencana	Data pada <i>database</i> terupdate
4	Uji coba pencarian petugas	Data petugas		Setelah berita bencana dikirimkan SIBAT, aplikasi secara otomatis mencari dan menampilkan data petugas yang bisa diberangkatkan oleh admin PB
		Notifikasi tim SATGANA		Petugas mendapat notifikasi untuk berita bencana
		Pembuatan surat tugas		Apabila petugas bisa untuk diberangkatkan, maka petugas tersebut menerima surat tugas via email
5	Uji coba laporan	Laporan berita bencana	Laporan bencana	Menampilkan hasil laporan dari SATGANA tentang bencana yang terjadi
		Laporan penugasan tim	Laporan bencana	Memperbarui data penugasan tim SATGANA
		Laporan data bencana Surabaya	Laporan bencana	Menampilkan data bencana yang terjadi di Surabaya

No	Nama Tes	Proses	Input	Output yang diharapkan
		Laporan kebutuhan sosialisasi dan pelatihan	Data bencana Surabaya	Menampilkan kebutuhan sosialisasi dan pelatihan mengenai siaga bencana pada wilayah tertentu

