

# **SKRIPSI**

## **PENGEMBANGAN SISTEM INFORMASI KESEHATAN DAERAH DI DINAS KESEHATAN**



**Disusun Oleh :**

**HERNA DIAN PRIHATINY NAPAN**

**NIM 04.12.642**

**JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
SEPTEMBER 2008**

10/11/19

PATRIARCA ISAMOTU NATIONI NARRADUNENAE  
NATANDON DUNO DI NARRAE

10/11/19

NARRAN NARRADUNENAE NARRAE NARRAE  
NARRAE NARRAE NARRAE

1-2 OCTOBERI NARRAE NARRAE

NARRADUNENAE NARRAE NARRADUNENAE NARRAE NARRADUNENAE

NARRADUNENAE NARRADUNENAE NARRADUNENAE

NARRADUNENAE NARRADUNENAE NARRADUNENAE

10/11/19

LEMBAR PERSETUJUAN

PENGEMBANGAN SISTEM INFORMASI KESEHATAN DAERAH  
DI DINAS KESEHATAN

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh  
Gelar Sarjana Teknik Komputer dan Informatika Strata Satu (S-1)*

Disusun Oleh :  
**HERNA DIAN PRIHATINY NAPAN**


NIM : 04.12.642

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II


  
**Ir. F. Yudi Limpraptono, MT**  
NIP.Y 103 950 0274

  
**Yan Watequlis, ST**  
NIP 132 310 105

Mengetahui,

Ketua Jurusan Teknik Elektro S-1



  
**Ir. F. Yudi Limpraptono, MT**  
NIP.Y 103 950 0274

JURUSAN TEKNIK ELEKTRO S-1  
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI NASIONAL MALANG  
2008





**INSTITUT TEKNOLOGI NASIONAL MALANG**  
**FAKULTAS TEKNOLOGI INDUSTRI**  
**JURUSAN TEKNIK ELEKTRO S-1**  
Jl. Karanglo KM.2 Malang

**BERITA ACARA UJIAN SKRIPSI**  
**FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Herna Dian Prihatiny Napan  
Nim : 04.12.642  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer dan Informatika  
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI KESEHATAN**  
**DAERAH DI DINAS KESEHATAN**  
Dipertahankan di hadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :  
Hari : Jumat  
Tanggal : 26 September 2008  
Dengan Nilai : 84,75 (A) *84*



**Ketua Majelis Penguji**

**Ir. Mochtar Asroni, MSME**  
NIP.Y. 1018100036

**Sekretaris Majelis Penguji**

**Ir. F. Yudi Limpraptono, MT**  
NIP Y. 1039500274

**Penguji I**

**DR. Cahyo Crysdian, MSc**  
NIP P. 1030400412

**Penguji II**

**Sotvohadi, ST, MSc**  
NIP Y. 1030400412



## **Abstrak**

*Sistem kesehatan memiliki banyak unit pelayanan kesehatan yang saling terkait, mulai dari pelayanan kesehatan dasar di desa dan kecamatan sampai ke tingkat nasional. Jaringan sistem pelayanan tersebut memerlukan sistem informasi yang saling mendukung dan terkait agar setiap kegiatan dan program kesehatan dapat diketahui, dipahami, diantisipasi dan dikelola dengan sebaik-baiknya. Akan tetapi, pada kenyataannya masih banyak daerah yang belum memiliki basis data kesehatan yang berbasis pada aplikasi komputer. Skripsi ini akan membahas pengembangan aplikasi sistem informasi kesehatan daerah pada dinas kesehatan di kabupaten Flores Timur. Aplikasi dikembangkan dengan model client-server berbasis web yang menghubungkan dinas kesehatan dengan puskesmas dan unit kerja lainnya melalui jaringan sehingga pengiriman data akan lebih cepat dan efisien.*

*Pengembangan aplikasi ini diharapkan dapat menunjang kinerja dinas kesehatan dalam melakukan pendataan dan pelaporan tentang informasi kesehatan yang selalu ter-update dari sumber kesehatan masyarakat.*

**Kata Kunci :** *sistem informasi kesehatan, dinas kesehatan, Flores Timur.*

## **Abstract**

*Health system has various units that are linked to each other, from the small health service in remote village and county areas until to the large scale in the national level. This complex and large scale network system for the health service needs an information system that can support and can be relying on each other in order to know, understand, anticipate, and manage well the activities from the health program.*

*Unfortunately, in the reality many regions there are not equipped with health database which is based on the computerized application. This thesis will discuss and explain the development of an application of health information system program at health department in the East Flores.*

*This application is developed as a web based client-server model that linked between East Flores health department with other public medical center in remote places and other operational units through the network thus the data transfer will be faster and more efficient, with an expectation to support the operational in health department in data entry and report of health information that can be update from the public health sources.*

**Keywords :** *Health Information System Program, Health Department, East Flores.*

## **KATA PENGANTAR**

Dengan mengucapkan syukur kehadiran Tuhan YME yang dengan segala Kasih dan Anugerah-Nya, telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul : “**PENGEMBANGAN SISTEM INFORMASI KESEHATAN DAERAH DI DINAS KESEHATAN**”

Pembuatan skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata I di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis banyak mendapat bantuan baik moril maupun materiil, saran dan dorongan semangat dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku rektor ITN Malang.
2. Bapak Ir. Mochtar Asroni, MSEE selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
4. Bapak Ir. F. Yudi Limpraptono, MT dan Bapak Yan Watequlis S, ST selaku Dosen Pembimbing. Terima kasih telah membimbing Penulis dengan penuh kesabaran dan atas semua yang bapak berikan kepada Penulis.
5. Bapak DR. Cahyo Crysdiyan, MSc dan Bapak Sotyohadi, ST, MSc selaku Dosen Penguji yang telah memberikan pertanyaan dan saran yang sangat membantu dalam penyempurnaan skripsi ini.

6. Kedua orang tua dan saudara-saudara tercinta atas segala dukungan, nasehat dan doanya.
7. Denny Lale atas segala doa, cinta kasih, semangat, perhatian serta semua bantuannya.
8. Teman-teman dan semua pihak yang telah membantu Penulis dalam menyelesaikan skripsi ini yang tidak bisa Penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan.

Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, September 2008

Penulis



## DAFTAR ISI

<b>LEMBAR PERSETUJUAN</b> .....	i
<b>BERITA ACARA UJIAN SKRIPSI</b> .....	ii
<b>ABSTRAK</b> .....	iii
<b>KATA PENGANTAR</b> .....	iv
<b>DAFTAR ISI</b> .....	vi
<b>DAFTAR TABEL</b> .....	x
<b>DAFTAR GAMBAR</b> .....	xii
<b>BAB I PENDAHULUAN</b> .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Tujuan .....	3
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian .....	4
1.5.1. Metode Pengumpulan Data .....	4
1.5.2. Metode Pengembangan Sistem .....	5
1.6. Sistematika Penulisan.....	5
<b>BAB II DASAR TEORI</b> .....	7
2.1. Pengertian Dasar Sistem .....	7
2.1.1. Karakteristik Sistem .....	7
2.1.2. Klasifikasi Sistem .....	10
2.2. Pengertian Dasar Informasi.....	11
2.3. Sistem Informasi .....	13

2.3.1. Sistem Informasi Kesehatan Daerah .....	14
2.4. Pengertian <i>Database</i> dan RDBMS .....	16
2.4.1. <i>Database</i> .....	16
2.4.2. <i>Relational Database dan Management Sistem</i> .....	17
2.5. Microsoft SQL .....	18
2.5.1. SQL .....	18
2.5.2. Microsoft SQL Server 2005 .....	19
2.6. ASP.NET.....	20
2.7. Perangkat Analisis dan Perancangan .....	21
2.7.1. Diagram Alir Data.....	21
2.7.2. Pemodelan Data .....	22
2.8. Metode Pengembangan Sistem Waterfall .....	23
<b>BAB III PERANCANGAN SISTEM</b> .....	<b>25</b>
3.1. Sitem Saat Ini .....	25
3.1.1. Spesifikasi Sistem Saat Ini .....	28
3.1.2. Kelebihan Sistem Saat Ini .....	28
3.1.3. Kekurangan Sistem Saat Ini .....	29
3.2. Sistem Informasi Kesehatan Daerah .....	29
3.2.1. Pengguna Sistem .....	30
3.2.2. Spesifikasi Sistem .....	31
3.3. Perancangan Sistem .....	32
3.3.1. Diagram Alir Data (DAD) .....	32
3.3.1.1. DAD Level 0 .....	32
3.3.1.2. DAD Level 1 .....	33

3.3.1.3. DAD Level 2 .....	34
3.3.2. Desain Basisdata .....	36
3.3.2.1. Basisdata Admin .....	36
3.3.2.1.1. Relasi Antar Tabel.....	36
3.3.2.1.2. Struktur Tabel-Tabel Yang Digunakan.....	38
3.3.2.2. Basisdata SIMKESDA .....	43
3.3.2.2.1. Diagram Entitas Relasi.....	43
3.3.2.2.2. Struktur Tabel-Tabel Yang Digunakan.....	57
3.3.3. Desain Antarmuka Aplikasi .....	87
3.3.3.1. Desain Halaman Login.....	88
3.3.3.2. Desain Menu Aplikasi.....	88
3.3.3.3. Desain Halaman Utama .....	89
3.3.3.4. Desain Halaman Entri Data.....	90
3.3.3.5. Desain Halaman Laporan.....	91
<b>BAB IV IMPLEMENTASI DAN PENGUJIAN .....</b>	<b>93</b>
4.1. Implementasi Sistem .....	93
4.1.1. Koneksi Basis Data .....	93
4.1.2. Pengoperasian Basis Data .....	95
4.2. Pengujian Hasil .....	99
4.2.1. Pengujian Entri Data .....	99
4.2.2. Pengujian Laporan .....	101
4.2.3. Pengujian Hak Akses Pengguna .....	104
4.2.4. Pengujian pada Beberapa Web Browser .....	107



<b>BAB V PENUTUP</b> .....	109
5.1. Kesimpulan .....	109
5.2. Saran.....	109
<b>DAFTAR PUSTAKA</b> .....	110
<b>LAMPIRAN</b> .....	111

## DAFTAR TABEL

### BAB II DASAR TEORI

Tabel 2.1 Daftar sejumlah pernyataan SQL.....	19
---	----

### BAB III ANALISIS DAN PERANCANGAN SISTEM

Tabel 3.1 Struktur Tabel T_USER.....	38
Tabel 3.2 Struktur Tabel T_ROLE.....	39
Tabel 3.3 Struktur Tabel T_SCOPE.....	39
Tabel 3.4 Struktur Tabel T_MENU_APP.....	40
Tabel 3.5 Struktur Tabel T_SUBMENU_APP.....	40
Tabel 3.6 Struktur Tabel T_HLM_APP.....	41
Tabel 3.7 Struktur Tabel T_ROLE_HLM.....	42
Tabel 3.8 Struktur Tabel T_HLM_ADMIN.....	42
Tabel 3.9 Struktur Tabel D_BULAN.....	58
Tabel 3.10 Struktur Tabel D_KABUPATEN.....	58
Tabel 3.11 Struktur Tabel D_KECAMATAN.....	58
Tabel 3.12 Struktur Tabel D_DESA.....	59
Tabel 3.13 Struktur Tabel D_TRIWULAN.....	59
Tabel 3.14 Struktur Tabel D_SATUAN.....	60
Tabel 3.15 Struktur Tabel D_OBAT.....	60
Tabel 3.16 Struktur Tabel D_JNS_PENYAKIT.....	60
Tabel 3.17 Struktur Tabel D_PENYAKIT.....	61
Tabel 3.18 Struktur Tabel D_PELAYANAN.....	62
Tabel 3.19 Struktur Tabel D_PANGKAT.....	62

Tabel 3.20 Struktur Tabel D_DATA_DINAS .....	62
Tabel 3.21 Struktur Tabel D_GUDANG .....	63
Tabel 3.22 Struktur Tabel T_PUSKESMAS .....	64
Tabel 3.23 Struktur Tabel D_JNS_KEGIATAN .....	65
Tabel 3.24 Struktur Tabel D_KEGIATAN.....	66
Tabel 3.25 Struktur Tabel D_SUB_KEGIATAN .....	66
Tabel 3.26 Struktur Tabel D_ITEM_KEGIATAN .....	67
Tabel 3.27 Struktur Tabel D_ASAL_KELUHAN.....	67
Tabel 3.28 Struktur Tabel T_PEMAKAIAN_OBAT .....	68
Tabel 3.29 Struktur Tabel T_PEMAKAIAN_OBAT_DETIL.....	69
Tabel 3.30 Struktur Tabel D_KELOMPOK_UMUR .....	71
Tabel 3.31 Struktur Tabel T_REKAP_LB1 .....	71
Tabel 3.32 Struktur Tabel T_BULANAN_LB3 .....	72
Tabel 3.33 Struktur Tabel T_BULANAN_LB4 .....	73
Tabel 3.34 Struktur Tabel T_KEG_JAMKESMAS_PKM.....	73
Tabel 3.35 Struktur Tabel T_JAMKESMAS_PUSKESMAS .....	76
Tabel 3.36 Struktur Tabel T_PENANGANAN_KELUHAN_PKM .....	78
Tabel 3.37 Struktur Tabel T_PENYAKIT_TERBANYAK_PKM.....	79
Tabel 3.38 Struktur Tabel T_POLINDES_TKT_PERKEMBANGAN.....	80
Tabel 3.39 Struktur Tabel T_JML_JNS_PENGOBATAN_TRADISIONAL.....	81
Tabel 3.40 Struktur Tabel T_TOKOH_AGM_ADAT_MASY .....	82
Tabel 3.41 Struktur tabel T_JAMKESMAS_RS .....	83
Tabel 3.42 Struktur Tabel T_PENANGANAN_KELUHAN_RS .....	86
Tabel 3.43 Struktur Tabel T_PENYAKIT_TERBANYAK_RS .....	87



## **DAFTAR GAMBAR**

### **BAB II DASAR TEORI**

Gambar 2.1 Karakteristik Sistem .....	10
Gambar 2.2 Simbol Utama DFD.....	22
Gambar 2.3 Model Air Terjun (Waterfall).....	24

### **BAB III PERANCANGAN SISTEM**

Gambar 3.1 Aplikasi yang digunakan saat ini .....	26
Gambar 3.2 Alur Data dari Sumber Kesehatan Masyarakat ke Dinas Kesehatan .....	27
Gambar 3.3 Desain Sistem.....	31
Gambar 3.4 DAD Level 0 .....	32
Gambar 3.5 DAD Level 1 .....	33
Gambar 3.6 DAD Level 2 .....	35
Gambar 3.7 CDM Basisdata ADMIN_SIMKES .....	37
Gambar 3.8 PDM Basisdata ADMIN_SIMKES.....	37
Gambar 3.9 CDM Data Utama .....	44
Gambar 3.10 PDM Data Utama.....	45
Gambar 3.11 CDM Kegiatan LB Puskesmas.....	46
Gambar 3.12 PDM Kegiatan LB Puskesmas .....	47
Gambar 3.13 CDM Pemakaian Obat dan Angka Kesakitan.....	48
Gambar 3.14 PDM Pemakaian Obat dan Angka Kesakitan .....	49
Gambar 3.15 CDM Jamkesmas Puskesmas.....	50
Gambar 3.16 PDM Jamkesmas Puskesmas .....	51
Gambar 3.17 CDM Inspeksi Kesehatan Puskesmas. ....	52

Gambar 3.18 PDM Inspeksi Kesehatan Puskesmas.....	53
Gambar 3.19 CDM Tempat-Tempat Memenuhi Syarat Kesehatan.....	54
Gambar 3.20 PDM Tempat-Tempat Memenuhi Syarat Kesehatan .....	55
Gambar 3.21 CDM Bidang Pendidikan Kesehatan.....	56
Gambar 3.22 PDM Bidang Pendidikan Kesehatan .....	57
Gambar 3.23 Desain Halaman Login.....	88
Gambar 3.24 Desain Menu Halaman Admin.....	88
Gambar 3.25 Desain Menu Halaman Sistem Informasi.....	89
Gambar 3.26 Desain Menu Halaman Utama Sistem Informasi.....	89
Gambar 3.27 Desain Menu Halaman Entri .....	90
Gambar 3.28 Desain Menu Halaman Laporan.....	91
Gambar 3.29 Desain Laporan pada Crystal Reports.....	91
 <b>BAB IV IMPLEMENTASI DAN PENGUJIAN HASIL</b>	
Gambar 4.1 Kesalahan Entri Data.....	99
Gambar 4.2 Pesan Kesalahan Entri Data .....	100
Gambar 4.3 Entri Data .....	100
Gambar 4.4 Entri Berhasil. ....	100
Gambar 4.5 Tampilan Laporan Hasil Entri Data .....	101
Gambar 4.6 Dialog Ekspor Laporan .....	102
Gambar 4.7 Dialog Simpan File.....	102
Gambar 4.8 Hasil Ekspor Laporan Dalam Format Word Document.....	103
Gambar 4.9 Hasil Ekspor Laporan Dalam Format Excell Spreadsheet.....	103
Gambar 4.10 Mencetak Laporan Dalam Format Portable Document.....	104
Gambar 4.11 Entri Pengguna Baru .....	105

Gambar 4.12 Menentukan Hak Akses Pengguna.....	105
Gambar 4.13 Login Operator Dinas Kesehatan .....	106
Gambar 4.14 Menu Aplikasi Berdasarkan Hak Akses Pengguna.....	106
Gambar 4.15 Tampilan aplikasi pada Internet Explorer .....	107
Gambar 4.16 Tampilan aplikasi pada Mozilla Firefox .....	107
Gambar 4.17 Tampilan aplikasi pada Opera.....	108



# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Dalam era seperti saat ini, begitu banyak sektor kehidupan yang tidak terlepas dari peran serta dan penggunaan teknologi komputer dan internet, khususnya pada bidang-bidang dan lingkup pekerjaan. Semakin hari, kemajuan teknologi komputer, baik dibidang piranti lunak maupun perangkat keras berkembang dengan sangat pesat, disisi lain juga berkembang kearah yang sangat mudah dari segi pengaplikasian dan murah dalam biaya. Solusi untuk bidang kerja apapun akan ada cara untuk dapat dilakukan melalui media komputer dan internet, termasuk dalam bidang kesehatan.

Departemen Kesehatan telah menetapkan visi Indonesia Sehat 2010 yang ditandai dengan penduduknya yang hidup sehat dalam lingkungan yang sehat, berperilaku sehat, dan mampu menjangkau pelayanan kesehatan yang bermutu yang disediakan oleh pemerintah dan/atau masyarakat sendiri, serta ditandainya adanya peran serta masyarakat dan berbagai sektor pemerintah dalam upaya upaya kesehatan.

Dalam upaya mencapai visi dan misi yang telah ditetapkan tersebut, infrastruktur pelayanan kesehatan telah dibangun sedemikian rupa mulai dari tingkat nasional, propinsi, kabupaten dan seterusnya sampai ke pelosok. Setiap unit infrastruktur pelayanan kesehatan tersebut menjalankan program dan pelayanan kesehatan menuju pencapaian visi dan misi Depkes tersebut. Setiap jenjang memiliki sistem kesehatan

yang yang saling terkait mulai dari pelayanan kesehatan dasar di desa dan kecamatan sampai ke tingkat nasional.

Jaringan sistem pelayanan kesehatan tersebut memerlukan sistem informasi yang saling mendukung dan terkait, sehingga setiap kegiatan dan program kesehatan yang dilaksanakan dan dirasakan oleh masyarakat dapat diketahui, difahami, diantisipasi dan di kelola dengan sebaik-baiknya.

Saat ini masih banyak daerah (propinsi, kabupaten maupun kota) yang belum memiliki database kesehatan yang berbasis pada aplikasi komputer. Padahal yang diharapkan oleh Departemen Kesehatan adalah adanya database kesehatan dari daerah (Propinsi/Kabupaten/Kota) bisa langsung terhubung / terpantau lewat website atau sarana data elektronik lainnya.

Dalam kaitannya dengan Sistem Kesehatan ini, maka daerah ( dari tingkat kabupaten atau kota sampai kecamatan-kecamatan ) perlu menetapkan sistem kesehatannya sebagai sub sistem dari sistem pemerintahan daerah, yang penyelenggaraannya disesuaikan dengan aspirasi, potensi, serta kebutuhan setempat dengan memperhatikan prioritas pembangunan kesehatan masing-masing.

Dengan demikian perlu disusun sebuah Sistem Informasi Kesehatan Daerah yang akan diterapkan di tingkat kabupaten atau kota sampai kecamatan-kecamatan.

## **1.2 Rumusan Masalah**

Berdasarkan hal di atas maka timbul suatu permasalahan bagaimana membangun dan mengembangkan suatu aplikasi Sistem Informasi Kesehatan Daerah untuk mengelola data tentang pelayanan dan informasi kesehatan yang dapat menunjang kinerja dinas kesehatan pemerintah daerah.

### **1.3 Tujuan**

Mengembangkan Sistem Informasi Kesehatan Daerah untuk mengelola data tentang pelayanan dan informasi kesehatan di dinas kesehatan Pemerintah Daerah.

### **1.4 Batasan Masalah**

Agar permasalahan mengarah sesuai dengan tujuan maka pembahasan dibatasi pada hal-hal sebagai berikut :

1. Penerapan sistem informasi ini dibatasi hanya pada tingkat pemerintah daerah kabupaten, yaitu kabupaten Flores Timur.
2. Secara garis besar sistem informasi ini memiliki beberapa modul aplikasi, dalam hal ini dibatasi hanya pada modul Sistem Informasi pada Dinas Kesehatan.
3. Sistem informasi dengan model client server berbasis web, dimana aplikasi client dapat mengakses basisdata. Aplikasi client harus dapat dijalankan pada semua sistem operasi GUI dengan menggunakan web browser.
4. Sistem basisdata yang digunakan merupakan sistem basisdata terpusat.
5. Sistem jaringan yang digunakan tidak termasuk dalam pembahasan.
6. Sistem dikembangkan dengan menggunakan aplikasi ASP.NET dan sistem basisdata server menggunakan MS SQL Server 2005.

## **1.5 Metodologi Penelitian**

### **1.5.1 Metode Pengumpulan data**

Data merupakan sumber atau bahan mentah yang sangat berharga bagi proses menghasilkan informasi. Oleh sebab itu dalam pengambilan data perlu dilakukan penanganan secara cermat dan hati-hati, sehingga data yang diperoleh dapat bermanfaat dan berkualitas.

Dalam pengumpulan data penyusun menggunakan metode sebagai berikut :

#### **1. Studi Lapangan**

Dengan metode ini data-data diperoleh langsung dari sumber yang bersangkutan, dimana peneliti berhadapan langsung dengan obyek yang diteliti, yang dilakukan dengan cara :

##### **a. Survey**

Teknik pengumpulan data dengan cara terjun secara langsung dan mencatat secara sistematis terhadap obyek masalah.

##### **b. Wawancara / Interview**

Teknik pengumpulan data dengan jalan mengadakan komunikasi atau Tanya jawab secara langsung dengan pimpinan atau pegawai dinas kesehatan tentang sistem yang diterapkan.

#### **2. Studi Pustaka / Literatur**

Pengumpulan data ini dilakukan dengan cara mencari bahan-bahan kepustakaan sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan obyek penelitian.

### **1.5.2 Metode Pengembangan Sistem**

Metode pengembangan sistem yang digunakan adalah sebagai berikut :

#### **1. Analisis dan perancangan sistem**

Menganalisa kebutuhan sistem dan perancangan sistem perangkat lunak yang melibatkan identifikasi dan deskripsi abstraksi sistem perangkat lunak.

#### **2. Implementasi dan pengujian sistem**

Pada tahap ini, dilakukan implementasi hasil rancangan kedalam baris-baris kode program yang dapat dimengerti oleh mesin agar dapat direalisasikan sebagai serangkaian program atau unit program. Pengujian sistem melibatkan verifikasi bahwa setiap unit program telah memenuhi spesifikasinya.

#### **3. Integrasi dan pengujian sistem**

Unit program atau program individual diintegrasikan dan diuji sebagai sistem yang lengkap untuk menjamin bahwa persyaratan sistem telah dipenuhi. Setelah pengujian sistem, perangkat lunak dikirim kepada pelanggan.

### **1.6 Sistematika Penulisan**

Sistematika penulisan yang digunakan dalam penyusunan skripsi ini adalah sebagai berikut :

#### **BAB I : PENDAHULUAN**

Meliputi latar belakang, rumusan masalah, tujuan, batasan masalah dan sistematika penulisan.

#### **BAB II : DASAR TEORI**

Berupa penjelasan tentang Sistem Informasi Kesehatan, teori

ASP.Net, teori SQL Server 2005.

**BAB III : PERANCANGAN SISTEM**

Berisi analisa sistem yang dibutuhkan dan perencanaan objek uji atau desain sistem yang akan dikembangkan

**BAB IV : IMPLEMENTASI DAN PENGUJIAN**

Menyajikan pembuatan objek uji, hasil pengujian serta pembahasan dari hasil pengujian yang dilakukan.

**BAB V : PENUTUP**

Berisi kesimpulan dan saran dari hasil pembahasan pada skripsi ini.



## **BAB II**

### **DASAR TEORI**

#### **2.1 Pengertian Dasar Sistem**

Secara sederhana suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu. Teori sistem secara umum pertama kali diuraikan oleh Kenneth Boulding, terutama menekankan pentingnya perhatian terhadap setiap bagian yang membentuk sebuah sistem.

Teori sistem mengatakan bahwa setiap unsur pembentuk organisasi adalah penting dan harus mendapat perhatian yang utuh. Unsur atau komponen pembentuk organisasi di sini bukan hanya bagian-bagian yang tampak secara fisik, tetapi juga hal-hal yang mungkin bersifat abstrak atau konseptual, seperti misi, pekerjaan, kegiatan, kelompok informal dan lain sebagainya.

Unsur-unsur yang mewakili suatu sistem secara umum adalah masukan (input), pengolahan (processing), dan keluaran (output). Di samping itu suatu sistem dapat pula dikembangkan hingga menyertakan media penyimpanan. Sistem dapat terbuka dan tertutup. Sistem informasi biasanya adalah sistem terbuka, yang berarti menerima berbagai masukan dari lingkungan sekitarnya.

##### **2.1.1 Karakteristik Sistem**

Sebuah sistem terdiri atas bagian-bagian yang saling berkaitan dan bervariasi bersama-sama untuk mencapai beberapa sasaran dan maksud. Sebuah sistem

bukanlah seperangkat unsur yang tersusun secara teratur, tetapi terdiri atas unsur yang dapat dikenal yang saling melengkapi karena suatu maksud, tujuan dan sasaran.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yaitu :

a. **Komponen Sistem (*Components*)**

Suatu sistem terdiri atas sejumlah komponen yang saling berinteraksi, yang bekerjasama membentuk suatu kesatuan.

b. **Batas Sistem (*Boundary*)**

Batas sistem merupakan daerah yang membatasi antara sistem yang satu dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem memungkinkan suatu sistem dipandang sebagai suatu kesatuan yang tidak dapat dipisah-pisahkan.

c. **Lingkungan Luar Sistem (*Environment*)**

Lingkungan luar dari sistem adalah apapun di luar ruang lingkup sistem yang mempengaruhi operasi sistem. Lingkungan luar dapat bersifat menguntungkan dan dapat pula bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi bagi sistem tersebut, yang dengan demikian lingkungan luar tersebut harus selalu dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus dikendalikan, jika tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

d. **Penghubung Sistem (*Interface*)**

Interface merupakan media penghubung antara suatu subsistem dengan subsistem lainnya. Penghubung ini memungkinkan sumber daya mengalir dari suatu subsistem ke subsistem lainnya. Keluaran (*Output*) dari suatu

subsistem akan menjadi masukan (*Input*) untuk subsistem lainnya dengan melalui penghubung. Dengan demikian terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. Masukan Sistem (*Input*)

Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa pemeliharaan(*maintenance input*) dan sinyal(*signal input*). Sebagai contoh, di dalam suatu unit sistem komputer, “program” adalah maintenance input yang digunakan untuk mengoperasikan komputer sementara “data” adalah signal input yang akan diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

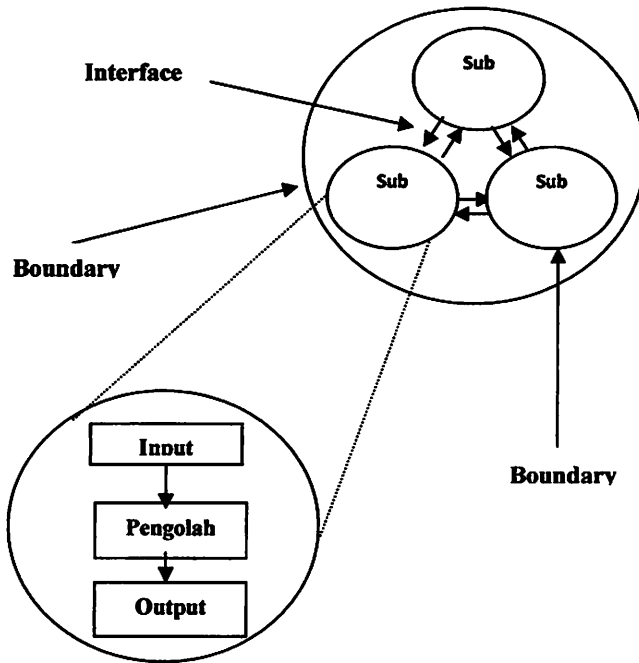
Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan untuk subsistem lain.

g. Pengolah Sistem (*Proses*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran Sistem (*Objective*)

Suatu sistem mempunyai tujuan atau sasaran. Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.



Gambar 2.1 Karakteristik Sistem<sup>[8]</sup>

### 2.1.2 Klasifikasi Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang, seperti :

a. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu sistem yang berupa pemikiran tentang hubungan antara manusia dengan Tuhan. Sedangkan sistem fisik merupakan sistem yang ada secara fisik, seperti sistem komputer, sistem penjualan, dan lain sebagainya.

b. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang dan malam. Sedangkan sistem buatan manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*. Sistem informasi berbasis komputer merupakan contohnya, karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

c. Sistem deterministik dan sistem probabilistik

Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem deterministik. Sistem komputer adalah contoh dari sistem yang tingkah lakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan. Sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

d. Sistem terbuka dan sistem tertutup

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak dipengaruhi oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

## 2.2 Pengertian Dasar Informasi

Informasi sangat penting artinya bagi suatu sistem yang akan dibuat dalam organisasi. Informasi adalah data yang telah diklasifikasikan atau diolah untuk

digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang.

Kualitas informasi terkadang juga dipakai untuk menyatakan informasi yang baik. Kualitas dari suatu informasi tergantung dari 3 (tiga) hal, yaitu :

a. Akurat

Informasi harus bebas dari kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi mungkin banyak mengalami gangguan (noise) yang dapat mengubah atau merusak informasi tersebut.

b. Tepat waktu

Informasi yang sampai pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam pengambilan keputusan. Bila pengambilan keputusan terlambat maka dapat berakibat fatal bagi organisasi. Dewasa ini informasi mahal karena harus cepat dikirim dan didapatkan sehingga memerlukan teknologi mutakhir untuk mendapatkan, mengolah dan mengirimnya.

c. Relevan

Informasi tersebut mempunyai manfaat untuk pemakaiannya. Relevansi informasi untuk setiap orang, satu dengan yang lain adalah berbeda.

## 2.3 Sistem Informasi

Dalam artian umum, sistem informasi merupakan interrelasi antara beberapa komponen yang menyimpan, mengambil, menerima, memproses, dan mendistribusikan informasi untuk mendukung kegiatan pengambil keputusan, dan mengontrol sebuah organisasi. Sedangkan dalam arti khusus, sistem informasi adalah perangkat lunak yang digunakan untuk meningkatkan kinerja sebuah sistem dalam menangani informasi atau data.<sup>[5]</sup>

Ada dua macam komponen dalam sistem informasi, yaitu :

### 1. Komponen dasar

- a. Input : meliputi elemen yang *dicapture*, dirakit yang masuk ke sistem untuk diproses.
- b. Proses : meliputi proses transformasi yang mengubah input menjadi output.
- c. Output : meliputi transfer elemen yang dihasilkan oleh proses ke tujuan.

### 2. Komponen tambahan

- a. Feedback : data yang menyatakan performansi sistem.
- b. Control : meliputi *monitoring* dan evaluasi *feedback* untuk menentukan apakah sistem mencapai tujuan. Fungsinya adalah membuat penyesuaian untuk input sistem dan pemrosesan komponen untuk memastikan sistem menghasilkan output yang tepat.

Dalam sistem informasi, ada tiga tahap yang harus dipenuhi, yaitu input, proses dan output, dimana output memungkinkan suatu umpan balik yang dapat

merubah atau memodifikasi suatu input. Di sinilah suatu sistem informasi berperan sebagai pengambil keputusan dalam menangani suatu permasalahan.

Di dalam pengembangan sistem informasi, dikenal istilah *SDLC* (*System Development Life Cycle*) atau “Siklus Hidup Pengembangan Sistem”<sup>[7]</sup>, yang terdiri dari enam tahap :

1. Perencanaan sistem
2. Analisis sistem
3. Rancangan sistem general atau konseptual
4. Evaluasi dan pemilihan sistem
5. Rancangan sistem terinci atau fungsional
6. Implementasi sistem

Empat tahap pertama disebut tahap *FRONT-END*, dua yang terakhir disebut tahap *BACK-END*. Setelah sistem baru dikembangkan dan dikonversikan ke operasi, selanjutnya menuju ke tahap pemeliharaan sistem yang berlangsung beberapa tahun, 10 sampai 20 tahun atau lebih lama. Jika sistem ini tidak lagi efisien dan efektif untuk tahap dipelihara, maka tidak dilanjutkan dan sistem baru dikembangkan untuk menggantikannya, *SDLC* mulai dari awal lagi.

### **2.3.1 Sistem Informasi Kesehatan Daerah<sup>[10]</sup>**

Sistem Informasi Kesehatan Daerah adalah suatu aplikasi sistem informasi untuk mengelola data tentang pelayanan dan informasi kesehatan pemerintah daerah. Perangkat lunak pemerintahan daerah ini yang dirancang untuk menunjang kinerja Dinas kesehatan di Pusat maupun di Provinsi, mengetahui secara dini segala bentuk



informasi kesehatan yang selalu *ter-update* dari sumber kesehatan masyarakat yaitu Puskesmas.

Secara garis besar sistem informasi kesehatan memiliki beberapa fasilitas atau modul sebagai berikut:

1. **Modul basisdata**, sebagai media penyimpanan, pengolahan, dan penyampaian data kependudukan dan data pendukung lainnya.
2. **Modul Administrator**, digunakan untuk proses administrasi SIMKESDA termasuk disini adalah untuk manajemen user, manajemen hak akses user, manajemen role user sesuai dengan kewenangannya.
3. **Modul Setup Data Utama**, digunakan untuk memasukkan data utama seperti data desa, data kecamatan, data puskesmas, data penduduk, data posyandu, dll.
4. **Modul Profil Kesehatan**, digunakan untuk mendata kebutuhan berdasarkan rencana kerja dan anggaran satuan kerja perangkat daerah setelah memperhatikan ketersediaan barang milik daerah yang ada.
5. **Modul Standar Pelayanan Minimal (SPM)**, digunakan untuk mengolah data SPM sesuai dengan standar pemerintah.
6. **Modul Indikator Indonesia Sehat (IIS)**, digunakan untuk mendata dan menganalisa tentang indikator indonesia sehat.
7. **Modul Perijinan**, digunakan untuk penetapan ijin penyelenggaraan pelayanan kesehatan oleh masyarakat.
8. **Modul Penyuluhan Kesehatan**, digunakan untuk mendata hasil penyuluhan kesehatan masyarakat.

9. **Modul Pencegahan dan Penanggulangan Penyakit**, digunakan untuk melakukan pendataan dan pelaporan tentang pencegahan dan penanggulangan penyakit.
10. **Modul Kesehatan Keluarga**, digunakan untuk melakukan pendataan dan pelaporan tentang pencegahan dan penanggulangan penyakit.
11. **Modul Tata Usaha**, digunakan untuk mengelola administrasi umum di Dinas Kesehatan.
12. **Modul Kesehatan Lainnya**, digunakan untuk melakukan pendataan dan pelaporan tentang data kesehatan yang lain.
13. **Modul Sistem Informasi Puskesmas**, digunakan untuk melakukan pengolahan data dan informasi yang ada di puskesmas kecamatan.
14. **Modul Informasi Eksekutif dan Pelaporan**, sebagai sistem informasi untuk memberikan sistem pelaporan dari data yang dihasilkan, seperti daftar inventaris barang dan aset pada setiap SKPD, daftar barang yang hilang dan rusak, daftar nilai barang dan aset, daftar mutasi barang dan aset, dll.

## **2.4 Pengertian Database dan RDBMS**

### **2.4.1 Database**

Basis data (*database*) adalah suatu pengorganisasian sekumpulan data yang saling terkait sehingga memudahkan aktivitas untuk memperoleh informasi. Basis data dimaksudkan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas. Sebuah konsep database memiliki beberapa hal sebagai berikut :

- Entitas : merupakan tempat informasi direkam, dapat berupa orang, tempat, kejadian dan lain-lain. Sebagai contoh dalam kasus administrasi siswa maka terdapat entity siswa, mata kuliah, guru, pembayaran.
- Atribut : disebut juga data elemen, data field, atau data item yang digunakan untuk menerangkan suatu entitas dan mempunyai harga tertentu, misalnya atribut dari entitas siswa diterangkan oleh, nama, tanggal lahir, alamat.
- Data Value : informasi atau data aktual yang disimpan pada tiap data, elemen, atau atribut.
- File/Tabel : kumpulan record sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda nilai datanya.
- Record/Tuple : kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu entitas secara lengkap. Satu record mewakili satu data atau informasi.

Untuk mengelola basis data diperlukan perangkat lunak yang disebut DBMS.

DBMS adalah perangkat lunak sistem yang memungkinkan para pemakai membuat, memelihara, mengontrol, dan mengakses basis data dengan cara yang praktis dan efisien.

#### **2.4.2 Relational Database dan Management Sistem**

Relational Database adalah kumpulan data yang saling berelasi yang dipakai/ada dalam suatu lingkup tertentu, misalkan instansi, perusahaan dan lain-lain atau kasus tertentu. RDBMS (Relational Database Management System) merupakan koleksi atau kumpulan data yang di dalamnya memiliki suatu sistem yang mengatur relasi di dalamnya bersama dengan satu set program yang berfungsi untuk melakukan manajemen sistem terhadap data tersebut.

Selanjutnya dalam RDBMS semua data disimpan dalam tabel-tabel, di mana sebuah tabel menyimpan informasi mengenai sebuah subjek tertentu. Dengan RDBMS, sebuah database akan dengan mudah dikelola walaupun jumlah datanya banyak dan kompleks, seperti pendefinisian data, mana data yang akan dimuat ke dalam sebuah database, bagaimana mengelolanya, serta bagaimana membagi data. Ide RDBMS ini yaitu menggunakan konsep matematika aljabar relasional untuk membagi data dalam beberapa himpunan (set) yang saling berhubungan dalam subset. Dalam model relasional, data dipisahkan dalam beberapa set yang paralel dengan struktur tabel. Struktur tabel ini mengandung elemen data individual yang disebut kolom atau *field*. Satu set kumpulan kolom disebut *record*.

## 2.5 Microsoft SQL

### 2.5.1 SQL

SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengakses basis data yang tergolong relasional.

Sesungguhnya SQL tidak terbatas hanya untuk mengambil data (*query*), tetapi juga dapat dipakai untuk menciptakan tabel, menghapus tabel, menambahkan data ke tabel, menghapus data pada tabel, mengganti data pada tabel, dan berbagai operasi yang lain.

Pernyataan	Keterangan
SELECT	Untuk mengambil data
INSERT	Untuk menambahkan data
UPDATE	Untuk mengganti data
DELETE	Untuk menghapus data

CREATE TABLE	Untuk menciptakan tabel
--------------	-------------------------

Tabel 2.1 Daftar sejumlah pernyataan SQL<sup>[9]</sup>

### 2.5.2 Microsoft SQL Server 2005

MS SQL Server adalah salah satu produk Relational Database Management System (RDBMS) populer saat ini. Fungsi utamanya adalah sebagai database server yang mengatur semua proses penyimpanan data dan transaksi suatu aplikasi. Popularitas SQL Server akhir-akhir ini mulai menanjak dan setara dengan pesaing terdekatnya yaitu Oracle.

Saat ini SQL Server 2005 dan 2008 adalah versi terbaru dari SQL Server dengan penambahan fitur-fitur baik database SQL Server itu sendiri maupun sistem-sistem penunjangnya seperti Integration Service, Analytical Services, Reporting Services, Notification Services, dan Service Broker. Berbeda dengan versi-versi pendahulunya, SQL Server 2005 adalah sebuah terobosan yang revolusioner di mana .NET Framework menyatu menjadi bagian dari database ini yang memberikan keleluasaan bagi programmer untuk mengembangkan aplikasi-aplikasi dan mengintegrasikannya dengan fasilitas-fasilitas yang ada di dalam SQL Server 2005.

Terdapat beberapa keunggulan pada SQL Server 2005 dibandingkan versi-versi pendahulunya, antara lain :

- a. CLR Integration, implikasinya dapat membuat kode program menggunakan Visual Basic.Net atau C# dan menjalankannya sebagai stored procedure, trigger, dan user-defined function.
- b. Enkripsi dan Dekripsi Data. SQL Server 2005 memiliki infrastruktur pengelolaan key, guna meng-enkrip data berlapis-lapis dalam bentuk hierarki.

Setiap lapisan meng-enkrip lapisan bawahnya menggunakan kombinasi Certificate, Asymmetric Key, dan Symmetric Key.

- c. Tersedia berbagai web control yang dapat digunakan untuk aplikasi secara cepat. Kita dapat dengan mudah mengkaitkan data ke web control sebagaimana layaknya memprogram windows application. Hal ini sangat mempercepat pembuatan aplikasi dibandingkan harus menyusun kode-kode HTML secara manual.
- d. Dukungan terhadap tipe data XML, XML Query atau XQuery.

## **2.6 ASP.NET**

ASP.NET adalah teknologi baru dalam pemrograman web yang merupakan kelanjutan dari teknologi ASP 3.0. Perbedaan utama dibanding ASP klasik adalah penggunaan .NET Framework sebagai fondasi pemrograman. Selain itu, ASP.Net memiliki beberapa kelebihan dibanding ASP klasik versi 3, antara lain:

- a. Kemudahan mengakses berbagai library .Net Framework secara konsisten dan powerful yang mempercepat pengembangan aplikasi.
- b. Dalam ASP.Net beberapa bahasa pemrograman seperti VB.Net, C#.Net, J#.Net, dan C++.Net dapat digunakan secara penuh sebagaimana layaknya bekerja di windows application.
- c. Tersedia berbagai web control yang dapat digunakan untuk aplikasi secara cepat. Kita dapat dengan mudah mengkaitkan data ke web control sebagaimana layaknya memprogram windows application. Hal ini sangat mempercepat pembuatan aplikasi dibandingkan harus menyusun kode-kode HTML secara manual.


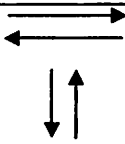
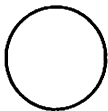
- d. Code Behind, artinya kode-kode pemrograman yang menjadi logic aplikasi ditempatkan terpisah dengan kode user interface yang berbentuk HTML. Ini sangat memudahkan dalam debugging, karena kode untuk presentation layer tidak tercampur dengan kode application logic.


## 2.7 Perangkat Analisis dan Perancangan

### 2.7.1 Diagram Aliran Data

Diagram Aliran Data atau yang biasa disebut dengan *DFD (Data Flow Diagram)* merupakan alat perancangan sistem yang berorientasi pada alur data dengan konsep dekomposisi dapat digunakan untuk penggambaran analisa maupun rancangan sistem yg mudah dikomunikasikan oleh profesional sistem kepada pemakai maupun pembuat program<sup>[7]</sup>.

Ada beberapa simbol DFD yang banyak dipakai, yaitu :

No.	Simbol	Penjelasan
01		Kesatuan luar adalah lingkungan luar sistem yang dapat berupa orang, departemen atau sistem lain yang memberikan input ataupun menerima output dari sistem.
02		Arus data adalah aliran data yang mengalir diantara proses, simpanan data dan kesatuan luar.
03		Proses adalah kerja atau kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan

		keluar dari proses.
04		Simpanan Data adalah merupakan simpanan data yang berupa file.

Gambar 2.2 Simbol Utama DFD<sup>[6]</sup>

### 2.7.2 Pemodelan Data

Model data adalah sekumpulan cara / peralatan / *tool* untuk mendeskripsikan data-data, hubungannya satu sama lain, semantiknya, serta batasan konsistensi.

Ada dua model data, yaitu : *Entity Relationship Diagram* (ERD) dan model relasional. Keduanya menyediakan cara untuk mendeskripsikan perancangan basis data pada peringkat logika.

Model ERD atau *Conceptual Data Model* (CDM) adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (*entity*) serta hubungan (*relationship*) antara entitas-entitas itu.

Model Relasional atau *Physical Data Model* (PDM) adalah model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unik.

Di dalam ERD atau CDM maupun PDM, relasi (hubungan) setiap entitas mempunyai derajat hubungan (kardinalitas) yang menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas yang lain. Relasi kardinalitas yang terjadi di antara dua himpunan entitas dapat berupa :



- a. 1 ke 1 (*one to one*), setiap entitas pada suatu himpunan entitas berhubungan paling banyak satu entitas pada himpunan entitas yang lain, begitu juga sebaliknya.
- b. 1 ke N (*one to many*), setiap entitas berhubungan dengan banyak entitas pada himpunan entitas yang lain, tetapi tidak sebaliknya.
- c. N ke 1 (*many to one*), setiap entitas berhubungan dengan paling banyak satu entitas pada himpunan entitas yang lain, tetapi tidak sebaliknya.
- d. N ke N (*many to many*), setiap entitas pada suatu himpunan dapat berhubungan dengan entitas pada himpunan entitas yang lain, demikian sebaliknya.

## **2.8 Metode Pengembangan Sistem *Waterfall***

Model air terjun (*waterfall*), merupakan salah satu metode pengembangan sistem yang memiliki beberapa langkah, yaitu :

### **1. Analisis dan definisi persyaratan**

Pelayanan, batasan, dan tujuan sistem ditentukan melalui konsultasi dengan pengguna sistem. Persyaratan ini kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem.

### **2. Perancangan sistem dan perangkat lunak**

Proses perancangan sistem membagi persyaratan dalam sistem perangkat keras atau perangkat lunak. Kegiatan ini menentukan arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan deskripsi abstraksi sistem perangkat lunak yang mendasar dan hubungan-hubungannya.

### 3. Implementasi dan pengujian unit

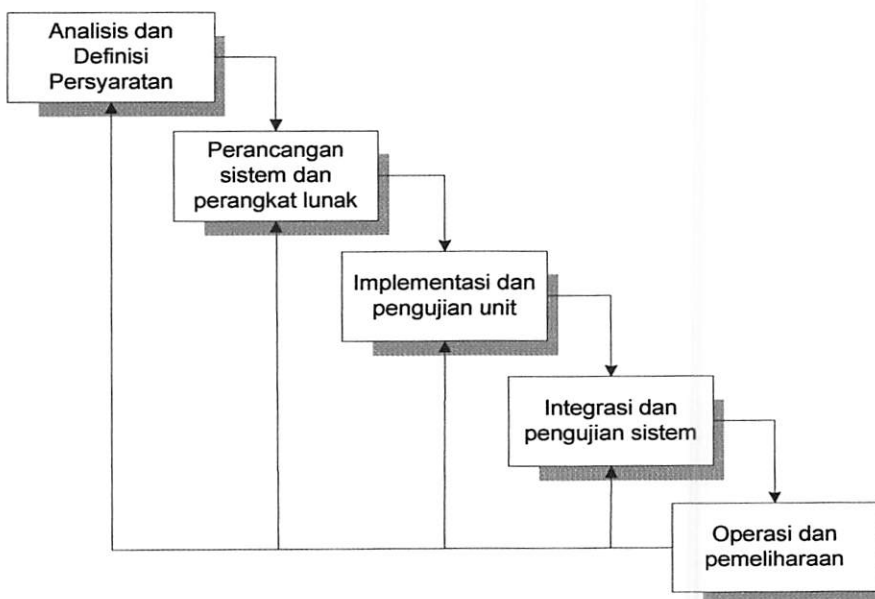
Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program atau unit program. Pengujian unit melibatkan verifikasi bahwa setiap unit telah memenuhi spesifikasinya.

### 4. Integrasi dan pengujian sistem

Unit program atau program individual diintegrasikan dan diuji sebagai sistem yang lengkap untuk menjamin bahwa persyaratan sistem telah dipenuhi. Setelah pengujian sistem, perangkat lunak dikirim kepada pelanggan.

### 5. Operasi dan pemeliharaan

Biasanya (walaupun tidak seharusnya), ini merupakan fase siklus hidup yang paling lama. Sistem diinstal dan dipakai. Pemeliharaan mencakup koreksi dari berbagai kesalahan yang tidak ditemukan pada tahap-tahap terdahulu, perbaikan atas implementasi unit sistem dan pengembangan sistem, sementara persyaratan-persyaratan baru ditambahkan.



Gambar 2.3 Model Air Terjun (*Waterfall*)<sup>[4]</sup>

## **BAB III**

### **PERANCANGAN SISTEM**

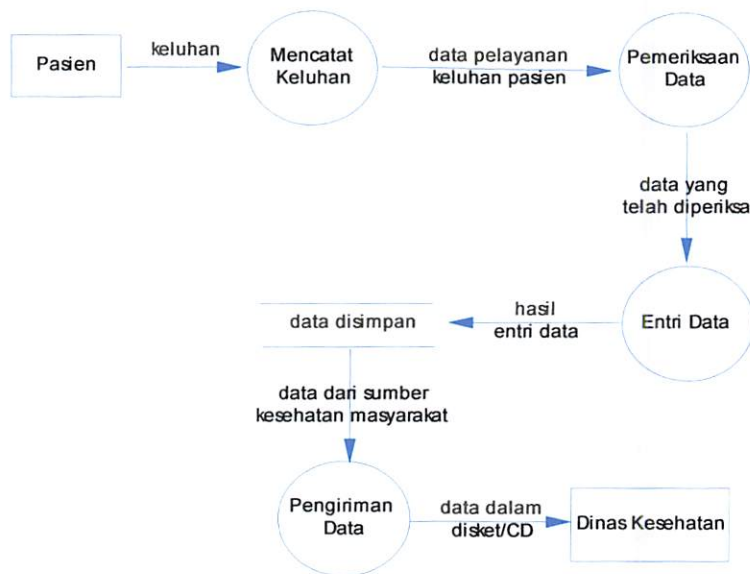
Pada bab ini dijelaskan mengenai analisis dan perancangan sistem aplikasi. Analisis ditujukan untuk memberikan gambaran secara umum terhadap aplikasi. Hal ini berguna untuk menunjang perancangan aplikasi yang akan dikembangkan sehingga kebutuhan akan aplikasi tersebut dapat diketahui sebelumnya. Kemudian hasil analisis akan menjadi dasar untuk melakukan perancangan atau desain aplikasi sesuai kebutuhan sistem.

Dalam merancang aplikasi pada proyek akhir ini terlebih dahulu dilakukan pembuatan desain proses, desain data, serta desain antar muka aplikasi. Desain proses berguna untuk mengintegrasikan semua proses yang terjadi dalam aplikasi yang akan dibuat. Desain data berguna untuk mengetahui data apa saja yang dibutuhkan dalam proses yang akan dikerjakan. Sedangkan perancangan antarmuka berfungsi sebagai antar muka interaksi antara pengguna dengan sistem aplikasi yang dibuat, sehingga pengguna dapat mengoperasikan aplikasi yang dibuat.

#### **3.1 Sistem Saat Ini**

Sistem yang ada saat ini sudah berbasis komputer, tetapi masih belum menerapkan sistem basis data yang terstruktur. Aplikasi yang digunakan terbatas hanya pada aplikasi *Microsoft Office Excel* yang disediakan oleh dinas kesehatan kabupaten.





Gambar 3.2 Alur Data dari Sumber Kesehatan Masyarakat Ke Dinas Kesehatan

Dari gambar di atas, dapat dijelaskan bahwa pendataan dari keluhan pasien masih dilakukan secara manual oleh petugas pendaftaran pasien pada setiap sumber kesehatan masyarakat. Berikutnya data tersebut akan diteruskan ke petugas periksa yang akan memeriksa keabsahan data untuk berikutnya akan di entri ke dalam program aplikasi yang digunakan saat ini, yaitu *Microsoft Office Excel*. Proses entri data dilakukan oleh petugas dari tiap bidang pada sumber kesehatan masyarakat.

Data yang dihasilkan akan dicetak dalam bentuk laporan atau disimpan ke dalam media penyimpanan seperti *disket*, *CD*, dan *flashdisk*, kemudian laporan tersebut akan dikirimkan ke dinas kesehatan di tingkat kabupaten / kota oleh petugas di setiap sumber kesehatan masyarakat. hal ini tentu membutuhkan waktu, tenaga dan biaya yang tidak sedikit karena jarak beberapa sumber kesehatan masyarakat ke dinas kesehatan cukup jauh.

### 3.1.1 Spesifikasi Sistem Saat Ini

Spesifikasi sistem yang berjalan saat ini antara lain :

1. Sistem yang digunakan adalah aplikasi *Microsoft Office Excel* untuk mengelola data dan informasi kesehatan di puskesmas.
2. Sistem ini hanya menggunakan satu buah komputer untuk entri data yang menggunakan sistem operasi *Microsoft Windows XP SP1* atau *SP2* yang tidak terhubung dengan jaringan, baik jaringan antar puskesmas maupun jaringan di dinas kesehatan.
3. Karena hanya menggunakan satu buah komputer dan tidak terhubung dengan jaringan, maka sistem ini tidak memiliki komputer server.
4. Informasi yang dihasilkan dicetak dalam bentuk laporan atau disimpan ke media penyimpanan sebelum dikirim oleh petugas di puskesmas ke dinas kesehatan kabupaten/kota.
5. Pengoperasian sistem tidak terlalu rumit sehingga cukup mudah dimengerti dan digunakan oleh petugas atau operator.

### 3.1.2 Kelebihan Sistem Saat Ini

Kelebihan yang dimiliki sistem saat ini antara lain :

1. Aplikasi tidak terpengaruh pada jaringan komputer karena bukan aplikasi *client-server*, dengan begitu proses entri data dan penyimpanan akan lebih cepat.
2. Data kesehatan per periode waktu tertentu dapat diketahui melalui data yang disimpan pada komputer.

3. Data-data kesehatan dapat di outputkan menjadi bentuk laporan-laporan sebagai data arsip dinas kesehatan.
4. Jika ada perubahan format laporan, maka tidak terlalu sulit untuk merubah format laporan yang terdapat pada aplikasi *excel*.

### **3.1.3 Kekurangan Sistem Saat Ini**

Kekurangan yang dimiliki sistem saat ini antara lain :

1. Aplikasi ini harus didistribusikan ke setiap unit kesehatan masyarakat yang ada di kabupaten karena sistem yang digunakan bukan aplikasi *client-server*.
2. Setiap pergantian periode pelaporan, aplikasi tersebut harus disimpan dengan nama yang berbeda sesuai periode pelaporan. Dengan cara seperti ini, ada banyak tumpukan *file* aplikasi di setiap komputer sehingga cukup sulit dalam pencarian data karena terdapat tumpukan *file* di setiap komputer.
3. Pengiriman data dari unit kesehatan ke dinas kesehatan kabupaten/kota yang masih dalam bentuk manual, tentunya membutuhkan waktu, tenaga dan biaya yang tidak sedikit. Hal ini mengakibatkan sistem kesehatan yang kurang efisiensi terutama untuk pengembangan pelayanan kesehatan di daerah-daerah terpencil.

## **3.2 Sistem Informasi Kesehatan Daerah**

Sistem Informasi Kesehatan di Dinas Kesehatan Kabupaten Flores Timur memiliki fungsi untuk mengelola data tentang pelayanan dan informasi kesehatan di

dinas kesehatan Pemerintah Daerah. Data yang diolah pada aplikasi ini mengambil masukan (*input*) data dari beberapa sumber, yaitu Puskesmas, Rumah Sakit, GFK (Gudang Farmasi Kabupaten). Pengiriman data dapat dilakukan secara fisik dengan media penyimpanan (flashdisk, disket, atau CD) atau koneksi lewat jaringan yang sudah dibangun terlebih dahulu. Jaringan harus menghubungkan Dinas Kesehatan dengan semua sumber data.

Dinas Kesehatan berfungsi sebagai unit utama dalam sistem informasi kesehatan daerah. Semua data kesehatan bisa diakses oleh unit Dinas Kesehatan. Di unit ini juga terdapat sistem informasi eksekutif yang digunakan untuk menyajikan data secara global dan mudah dibaca oleh pimpinan, sehingga akan mudah untuk melihat data secara global dan sebagai pendukung pengambilan keputusan.

### **3.2.1 Pengguna Sistem**

Pengguna sistem informasi kesehatan daerah antara lain :

- a. Administrator, dalam hal ini adalah administrator Dinas Kesehatan, yaitu pengguna yang memiliki hak akses penuh terhadap aplikasi Sistem Informasi Kesehatan Daerah.
- b. Pimpinan, yaitu pejabat-pejabat yang memiliki hak akses untuk menggunakan aplikasi.
- c. Operator, yaitu pengguna yang akan memasukkan data-data ke aplikasi.

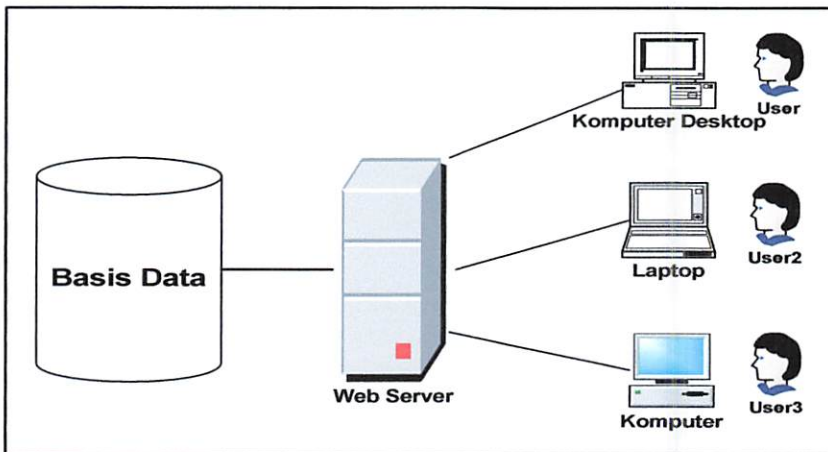
Setiap pengguna tersebut di atas dibatasi oleh hak akses aplikasi kecuali administrator yang memiliki hak akses penuh termasuk menentukan hak akses pengguna maupun menambah pengguna sistem.



### 3.2.2 Spesifikasi Sistem

Sesuai dengan tujuan dari pengembangan sistem informasi kesehatan berbasis web, maka spesifikasi dari sistem informasi kesehatan daerah adalah sebagai berikut:

1. Sistem informasi dengan model *client - server* berbasis web, dimana aplikasi *client* dapat mengakses basis data. Aplikasi *client* harus dapat dijalankan pada semua sistem operasi GUI dengan menggunakan web browser.
2. Sistem basisdata server menggunakan MS SQL Server 2005 dengan menggunakan satu buah server dengan sistem operasi Windows 2003 Server.
3. Sebuah modul *backoffice* pada setiap bagian untuk entri data.
4. Sebuah modul pelaporan dan pendukung keputusan yang dapat memberikan data-data dan rekapan yang akurat yang sangat berguna untuk pengambilan keputusan bagi *top management* di Dinas Kesehatan.
5. Sistem informasi ini, terutama bagian aplikasi client diharapkan dapat dengan mudah dipakai oleh banyak orang dan operator yang mengoperasikan sistem ini.



Gambar 3.3 Desain Sistem

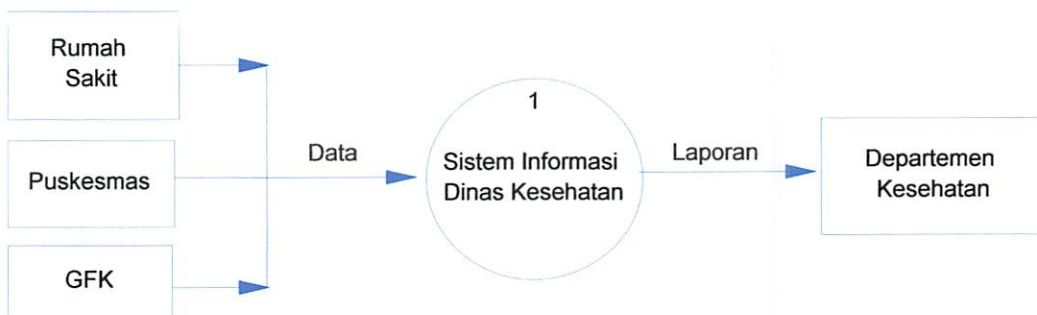
### 3.3 Perancangan Sistem

#### 3.3.1 Diagram Alir Data (DAD)

Dalam merancang suatu aplikasi, diperlukan suatu desain sistem yang dibuat dengan pemodelan proses. Model proses dari sistem informasi puskesmas ini dibuat dalam bentuk diagram alir data (DAD). DAD menunjukkan bagaimana alur kerja sistem informasi secara nyata.

##### 3.3.1.1 DAD Level 0

Proses pengelolaan data yang dilakukan oleh sumber kesehatan masyarakat seperti Puskesmas, Rumah Sakit dan Gudang Farmasi Kabupaten(GFK), berasal dari keluhan pasien untuk rumah sakit dan puskesmas serta data persediaan sarana dan prasarana kesehatan untuk GFK akan menghasilkan suatu rekapitulasi atau laporan-laporan secara bertahap yang dapat dikirim atau diakses secara online oleh dinas kesehatan kabupaten/kota Flores Timur.

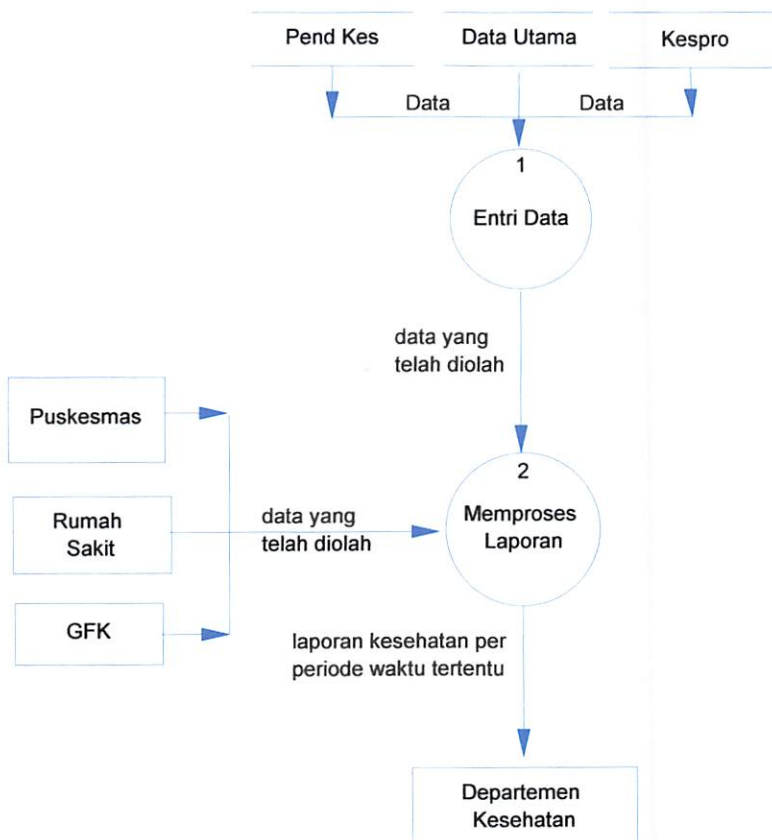


Gambar 3.4 DAD Level 0

### 3.3.1.2 DAD Level 1

Untuk memperjelas proses pada DAD level 0, maka perlu dilakukan pengembangan ke level berikutnya. Proses yang lebih rinci dapat dilihat seperti pada gambar 3.2. Data-data yang telah diolah di puskesmas dikirim ke dinas kesehatan untuk memproses laporannya.

Pada dinas kesehatan juga terdapat proses entri data utama dan entri data dari beberapa modul pada bidang Kespro dan Pendidikan Kesehatan. Proses entri data dilakukan sesuai dengan hak akses aplikasi masing-masing pengguna.



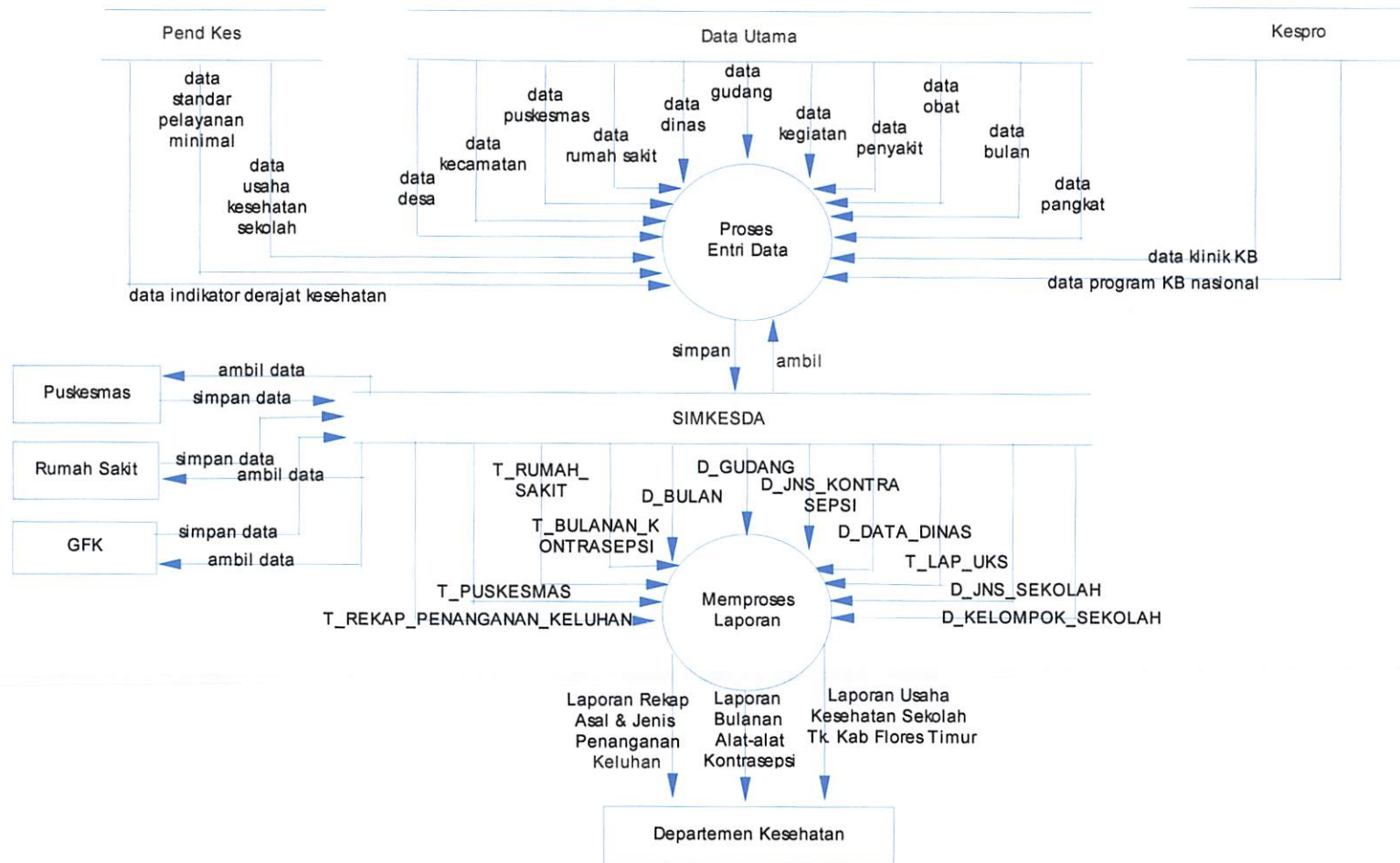
Gambar 3.5 DAD Level 1

### **3.3.1.3 DAD Level 2**

Proses yang lebih rinci dapat dilihat seperti pada DFD Level 2. Proses entri data dari beberapa bidang pada dinas kesehatan mengambil inputan berupa data-data kesehatan dari setiap bidang yang bersangkutan kemudian data-data tersebut diproses pada entri data menjadi tabel-tabel yang disimpan pada basisdata yang diberi nama SIMKESDA.

Tabel-tabel yang sudah dibuat sesuai dengan proses entri data pada aplikasi ini kemudian digunakan untuk memproses laporan tentang informasi dan pelayanan kesehatan per periode waktu tertentu. Laporan yang diproses berupa laporan rekapitulasi tahunan dari sumber-sumber kesehatan masyarakat (puskesmas, rumah sakit dan gudang farmasi kabupaten) dan laporan dari data bidang-bidang kesehatan yang telah dientrikan sebelumnya.

Laporan ini digunakan sebagai bahan pertimbangan pengambilan keputusan pada departemen kesehatan, pemerintah daerah, dinas kesehatan itu sendiri ataupun instansi-instansi lain yang membutuhkan laporan tentang informasi dan pelayanan kesehatan ini.



Gambar 3.6 DAD Level 2

### **3.3.2 Desain Basisdata**

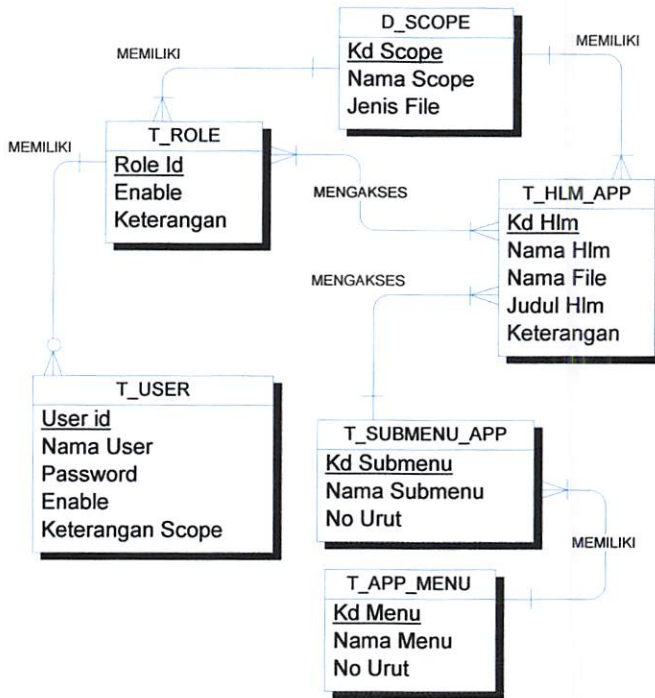
Terdapat dua macam basisdata yang digunakan pada aplikasi sistem informasi puskesmas ini, yaitu basisdata untuk admin sistem dan basisdata untuk sistem informasi puskesmas yang menjadi satu bagian dari basisdata sistem informasi kesehatan daerah itu sendiri.

Basisdata untuk admin berfungsi untuk menyimpan data-data seperti pengguna sistem, menu halaman aplikasi, dan pengaturan hak akses pengguna terhadap aplikasi. Sedangkan basisdata utama yaitu basisdata sistem informasi itu sendiri berfungsi untuk menyimpan semua informasi kesehatan yang akan dikelola oleh sistem.

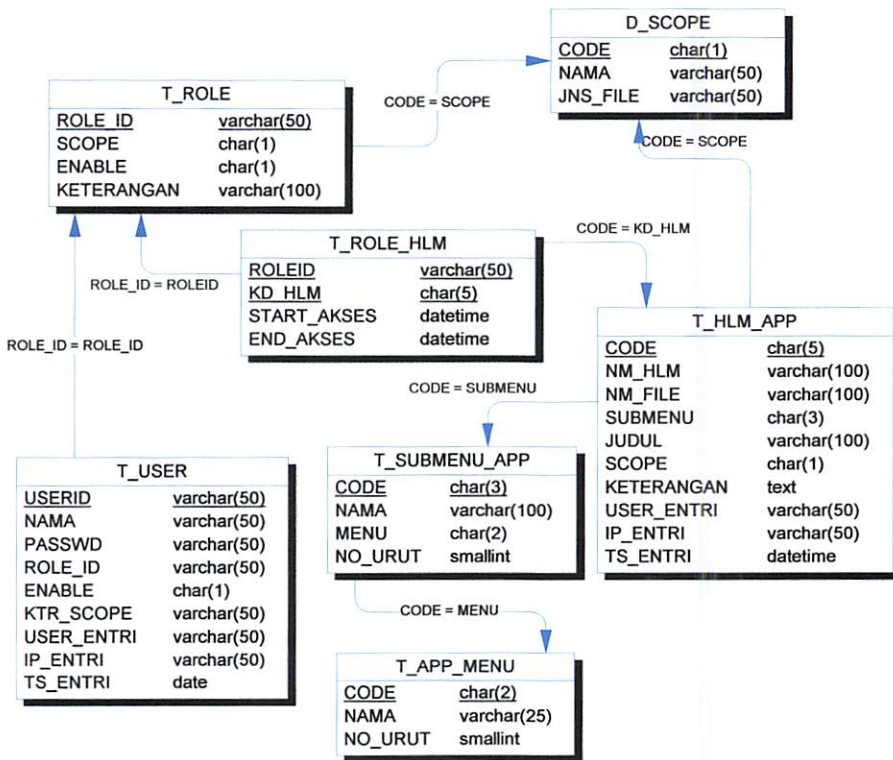
#### **3.3.2.1 Basisdata Admin**

##### **3.3.2.1.1 Relasi Antar Tabel**

Relasi antar tabel pada basisdata *ADMIN\_SIMKES* digambarkan dalam bentuk konsep atau *Conceptual Data Model (CDM)* dan dalam bentuk fisik atau *Physical Data Model (PDM)*, seperti yang ditunjukkan pada gambar berikut :



Gambar 3.7 CDM Basisdata ADMIN\_SIMKES



Gambar 3.8 PDM Basisdata ADMIN\_SIMKES

### 3.3.2.1.2 Struktur Tabel – Tabel Yang Digunakan

Berdasarkan CDM di atas, maka tabel – tabel yang digunakan antara lain :

#### 1. Tabel T\_USER

Tabel ini digunakan untuk menyimpan data pengguna sistem, berisi informasi yang diperlukan tentang pengguna sistem

Tabel 3.1 Struktur Tabel T\_USER

No.	Nama Kolom	Tipe Data	Keterangan
1.	USERID	Varchar(50)	<i>Primary Key (PK)</i> , identitas pengguna aplikasi
2.	NAMA	Varchar(100)	Nama pengguna
3.	PASSWD	Varchar(50)	Password
4.	ROLEID	Varchar(50)	<i>Foreign Key (FK)</i> , kode jenis pengguna aplikasi
5	ENABLE	Char(1)	Pengguna aplikasi aktif atau tidak
6.	KTR_SCOPE	Varchar(50)	Kode instansi asal pengguna aplikasi
7.	USER_ENTRI	Varchar(50)	Pengguna aktif yang entri data pada tabel ini
8.	IP_ENTRI	Varchar(50)	Alamat komputer yang digunakan untuk entri data pengguna sistem
9.	TS_ENTRI	DateTime	Waktu entri data



## 2. Tabel T\_ROLE

Tabel ini berfungsi untuk menyimpan tipe / jenis pengguna sistem berdasarkan hak akses terhadap aplikasi

Tabel 3.2 Struktur Tabel T\_ROLE

No.	Nama Kolom	Tipe Data	Keterangan
1.	ROLEID	Varchar(50)	<i>Primary Key (PK)</i> , Nama <i>Role</i> pengguna aplikasi
2.	SCOPE	Char(1)	<i>Foreign Key (FK)</i> , Kode <i>Scope</i> /batasan instansi <i>role</i> aplikasi
3.	ENABLE	Char(1)	<i>Role</i> aktif atau tidak
4.	KETERANGAN	Varchar(50)	Keterangan tentang <i>Role</i>

## 3. Tabel D\_SCOPE

Berfungsi untuk menyimpan data batasan atau ruang lingkup instansi pengguna aplikasi, seperti untuk dinas kesehatan, puskesmas, pengelola obat, maupun rumah sakit.

Tabel 3.3 Struktur Tabel T\_SCOPE

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(1)	<i>Primary Key (PK)</i> , Kode <i>SCOPE</i> instansi pengguna aplikasi
2.	NAMA	Varchar(50)	Nama instansi

3.	JNS_FILE	Varchar(50)	Jenis file yang yang diakses
----	----------	-------------	------------------------------

#### 4. Tabel T\_MENU\_APP

Tabel ini berfungsi untuk menyimpan daftar menu aplikasi sistem informasi kesehatan daerah.

Tabel 3.4 Struktur Tabel T\_MENU\_APP

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(2)	<i>Primary Key (PK)</i> , Kode Menu
2.	NAMA	Varchar(25)	Nama Menu
3.	NO_URUT	Smallint	Nomer urut menu

#### 5. Tabel T\_SUBMENU\_APP

Berfungsi untuk menyimpan daftar submenu dari menu aplikasi

Tabel 3.5 Struktur Tabel T\_SUBMENU\_APP

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(3)	<i>Primary Key (PK)</i> , Kode Submenu
2.	NAMA	Varchar(100)	Nama Submenu
3.	MENU	Char(2)	<i>Foreign Key (FK)</i> , Kode Menu
4.	NO_URUT	Smallint	Nomer urut Submenu

## 6. Tabel T\_HLM\_APP

Tabel ini fungsinya untuk menyimpan nama halaman aplikasi yang akan di akses melalui submenu aplikasi berdasarkan batasan pengguna aplikasi

Tabel 3.6 Struktur Tabel T\_HLM\_APP

No.	Nama Kolom	Type Data	Keterangan
1.	CODE	Char(5)	<i>Primary Key (PK)</i> , Kode Halaman Aplikasi
2.	NAMA	Varchar(100)	Nama Halaman
3.	NM_FILE	Varchar(100)	Nama file halaman
4.	SUBMENU	Char(3)	<i>Foreign Key (FK)</i> , Kode Submenu
5.	JUDUL	Varchar(100)	Judul Halaman
6.	SCOPE	Char(1)	<i>Foreign Key (FK)</i> , Kode <i>Scope</i> halaman
7.	KETERANGAN	Varchar(100)	Keterangan halaman aplikasi
8.	USER_ENTRI	Varchar(100)	Pengguna yang entri data halaman
9.	IP_ENTRI	Varchar(100)	Alamat komputer entri
10.	TS_ENTRI	Datetime	Waktu entri data

## 7. Tabel T\_ROLE\_HLM

Fungsinya untuk mengelola data hak akses setiap pengguna terhadap halaman aplikasi.

Tabel 3.7 Struktur Tabel T\_ROLE\_HLM

No.	Nama Kolom	Tipe Data	Keterangan
1.	ROLEID	Varchar(50)	<i>Primary Key (PK)</i> , Kode role
2.	KD_HLM	Char(5)	<i>Primary Key (PK)</i> , Kode halaman
3.	START_AKSES	Smalldatetime	Waktu mulai akses aplikasi
4.	END_AKSES	Smalldatetime	Waktu berhenti akses aplikasi

## 8. Tabel T\_HLM\_ADMIN

Tabel ini berfungsi untuk menyimpan daftar menu aplikasi halaman admin.

Tabel 3.8 Struktur Tabel T\_HLM\_ADMIN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(5)	<i>Primary Key (PK)</i> , Kode Halaman Aplikasi
2.	NAMA	Varchar(100)	Nama Halaman
3.	NM_FILE	Varchar(50)	Nama file halaman

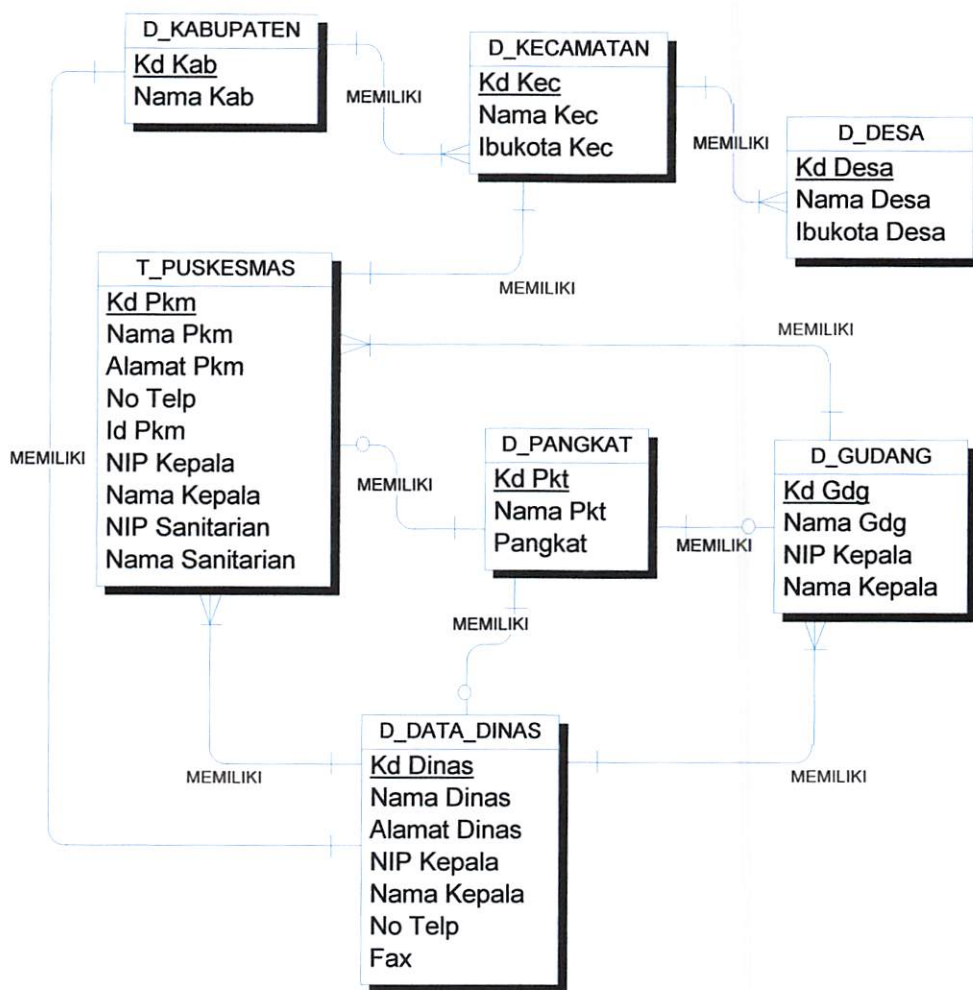
4.	KETERANGAN	Varchar(100)	Keterangan aplikasi	halaman
----	------------	--------------	------------------------	---------

### 3.3.2.2 Basisdata SIMKESDA

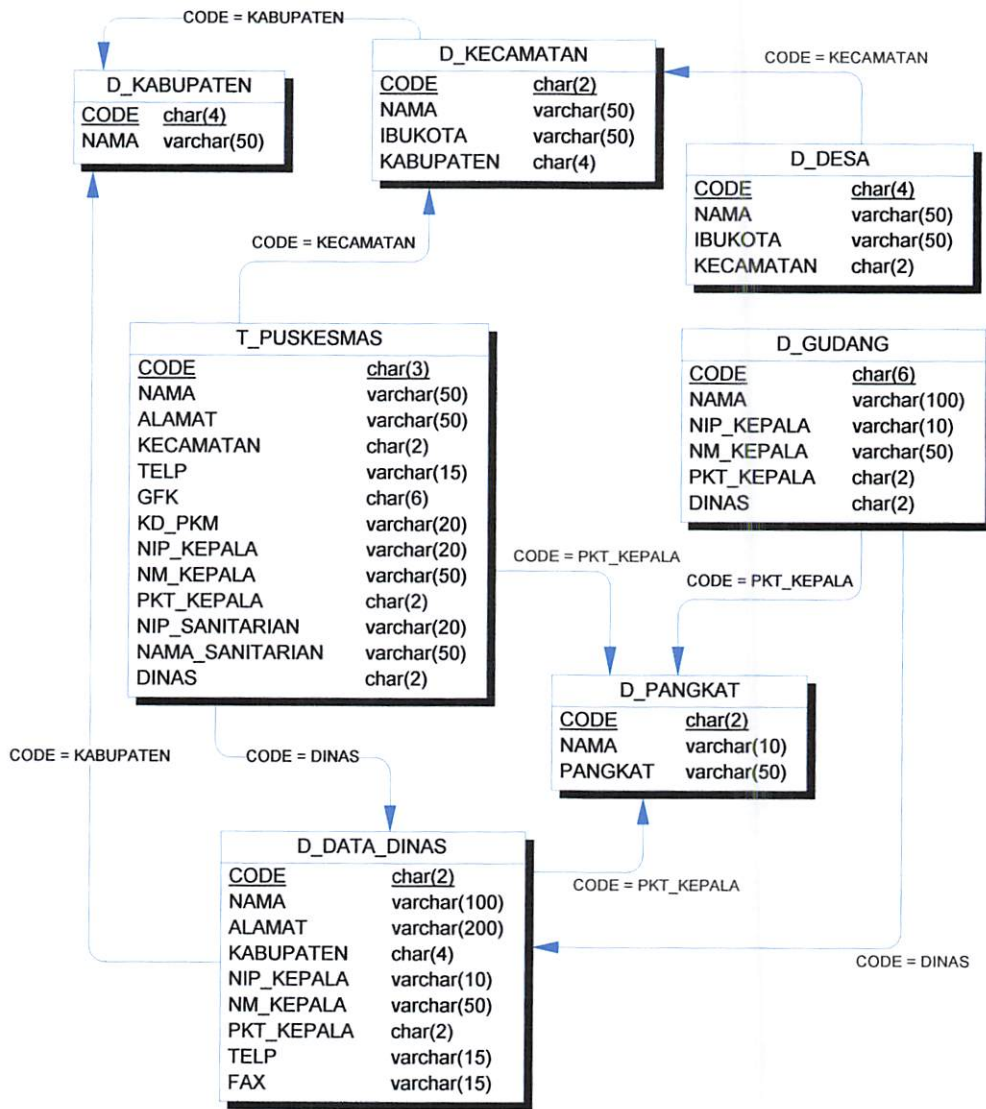
Basisdata sistem informasi dinas kesehatan menjadi satu dengan basisdata sistem informasi kesehatan daerah yang berada di server, yaitu “SIMKESDA”.

#### 3.3.2.2.1 Diagram Entitas Relasi

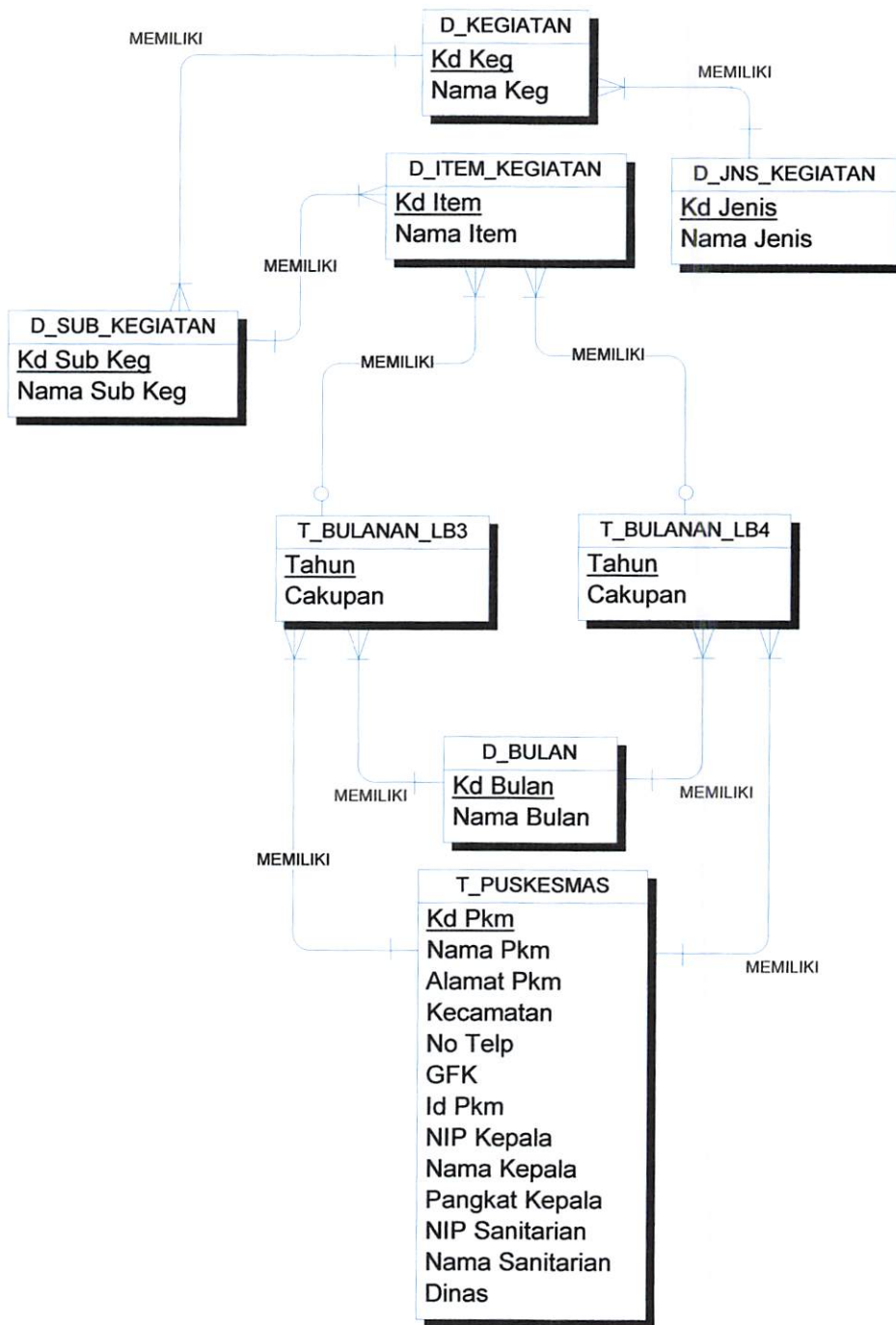
Relasi antar tabel pada basisdata *SIMKESDA* digambarkan dalam bentuk konsep atau *Conceptual Data Model (CDM)* dan dalam bentuk fisik atau *Physical Data Model (PDM)*, seperti yang ditunjukkan pada gambar berikut :



Gambar 3.9 CDM Data Utama

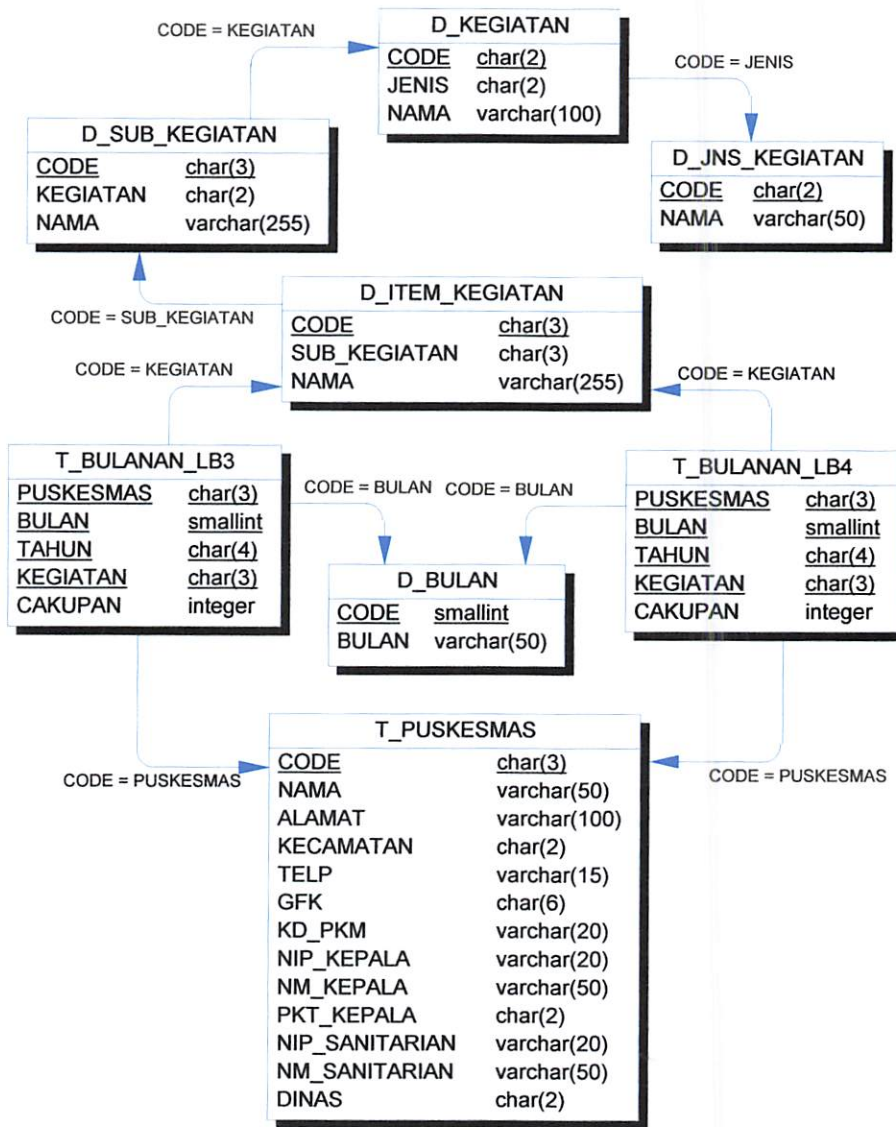


Gambar 3.10 PDM Data Utama

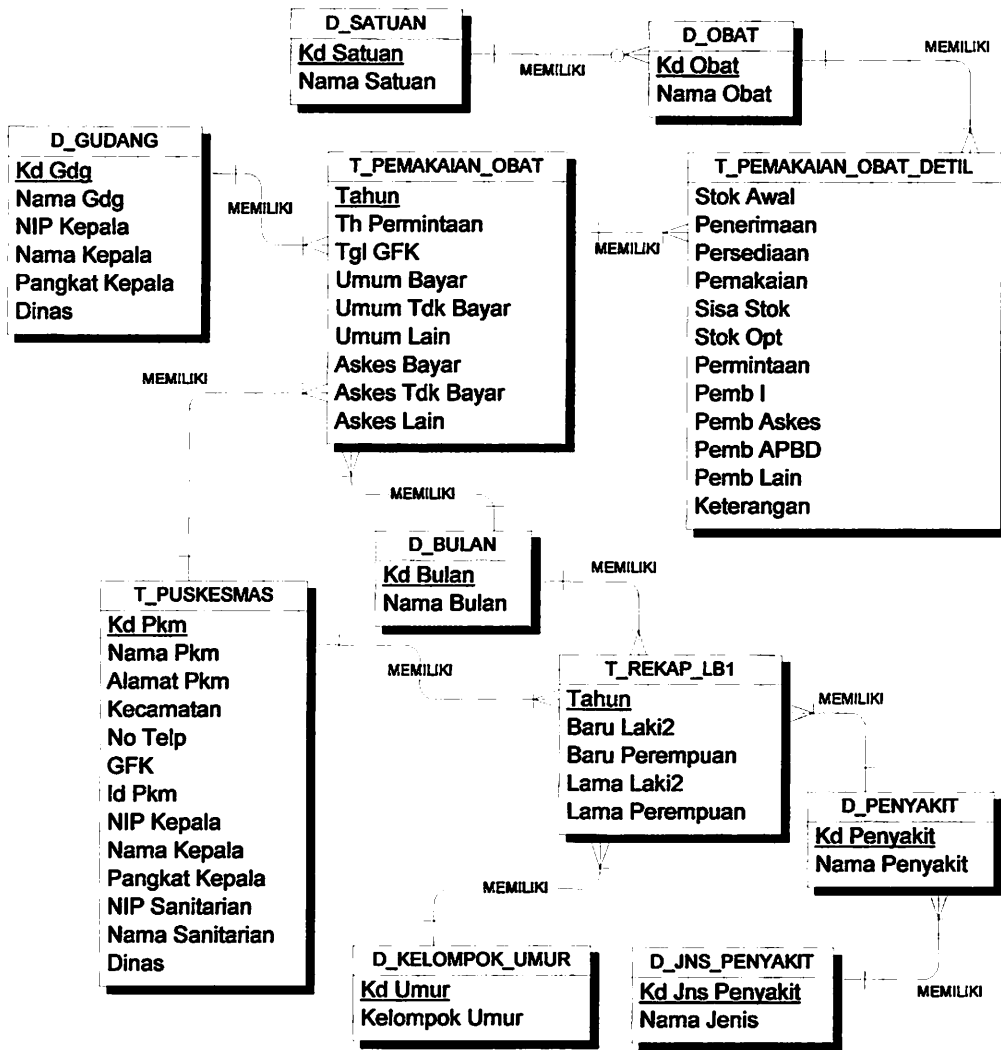


Gambar 3.11 CDM Kegiatan LB Puskesmas



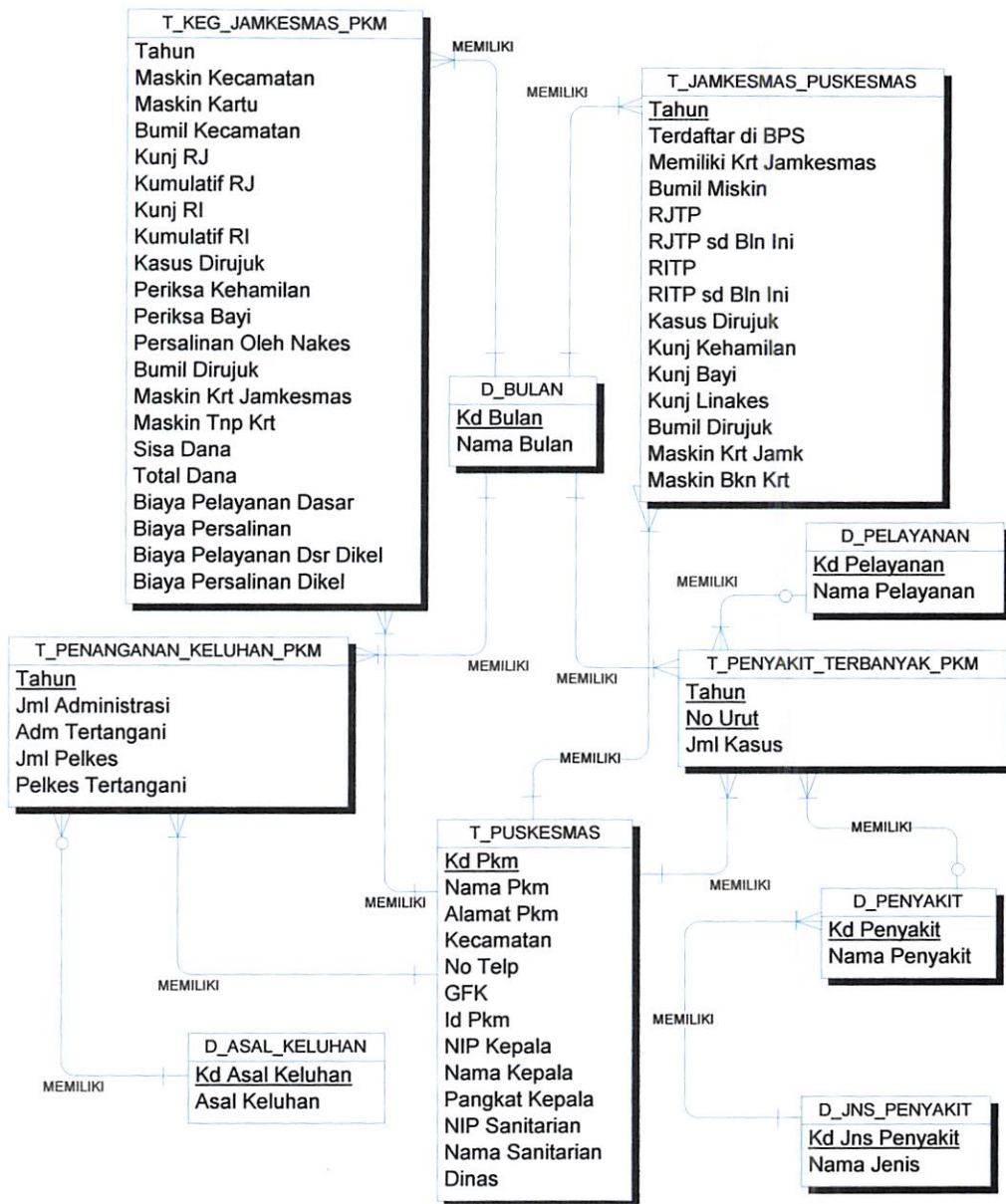


Gambar 3.12 PDM Kegiatan LB Puskesmas

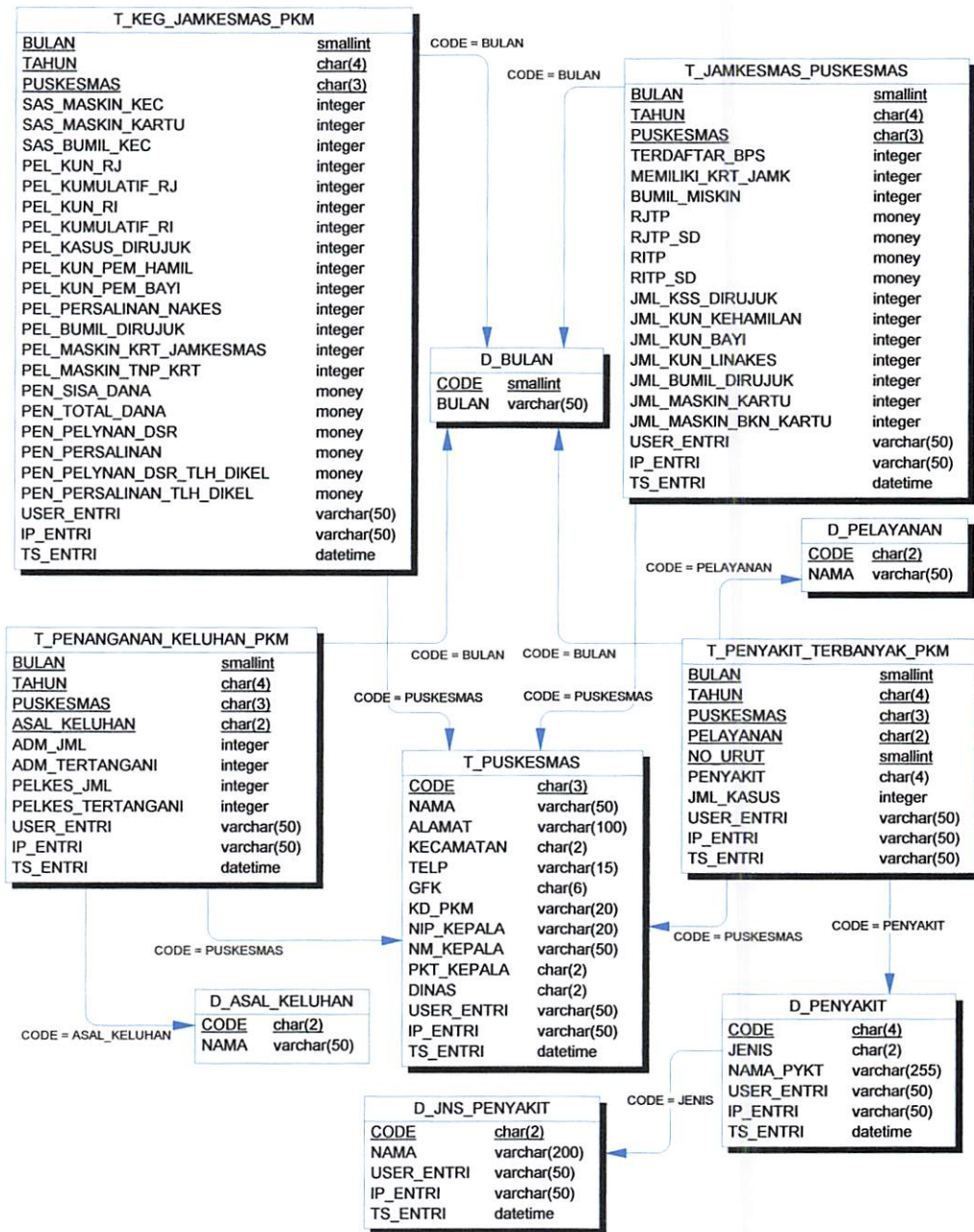


Gambar 3.13 CDM Pemakaian Obat dan Angka Kesakitan



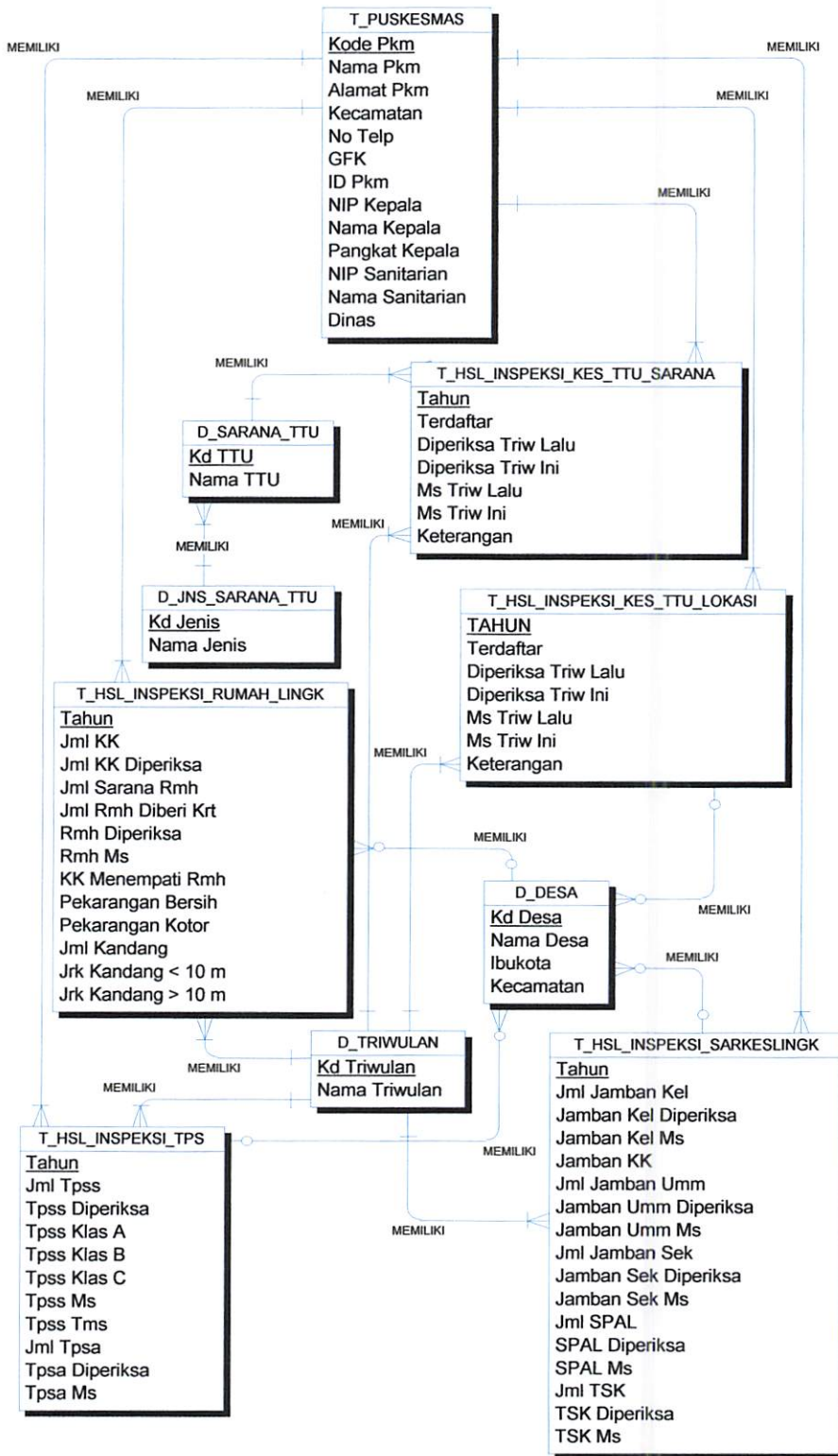


Gambar 3.15 CDM Jamkesmas Puskesmas

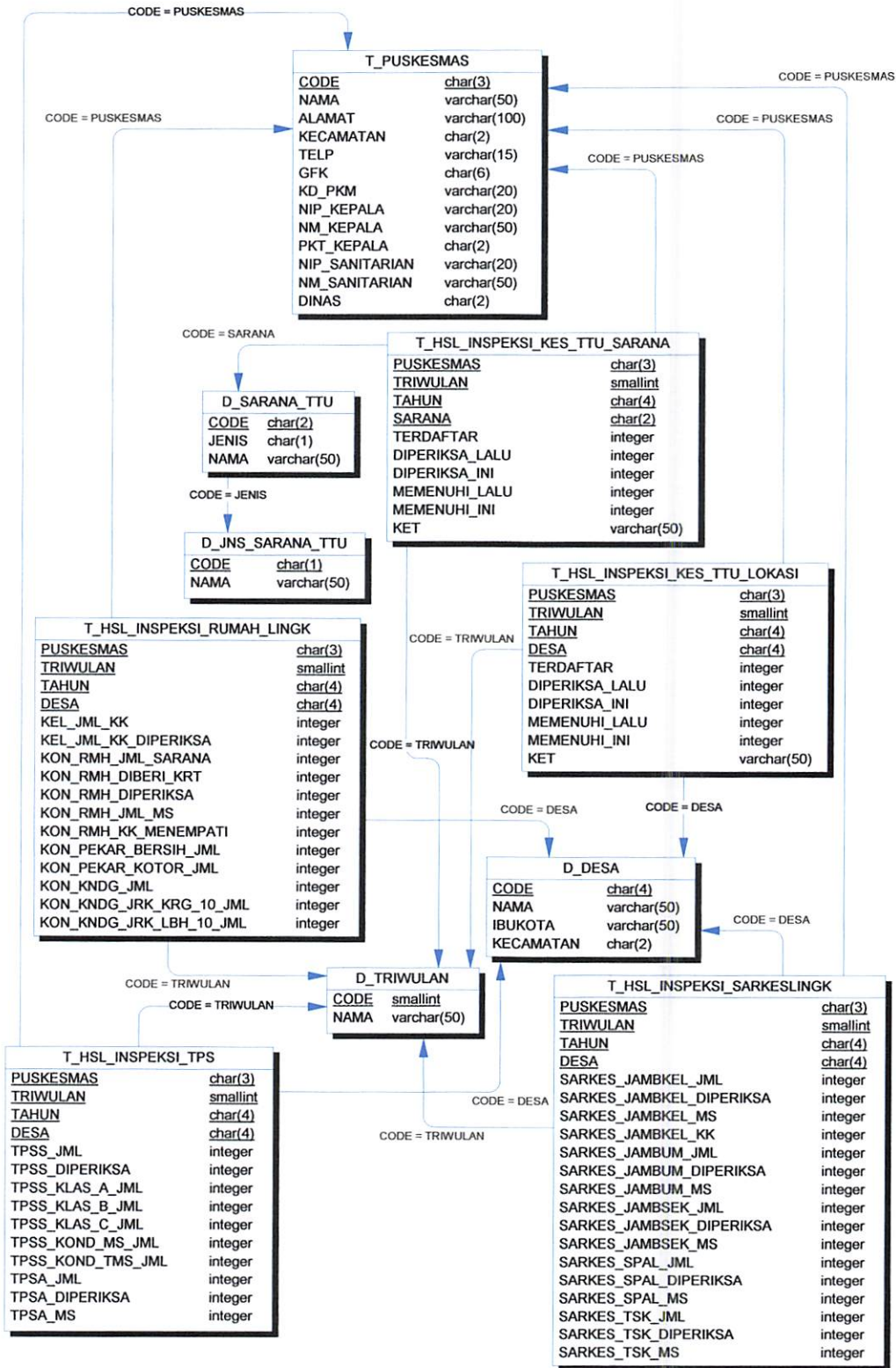


Gambar 3.16 PDM Jamkesmas Puskesmas

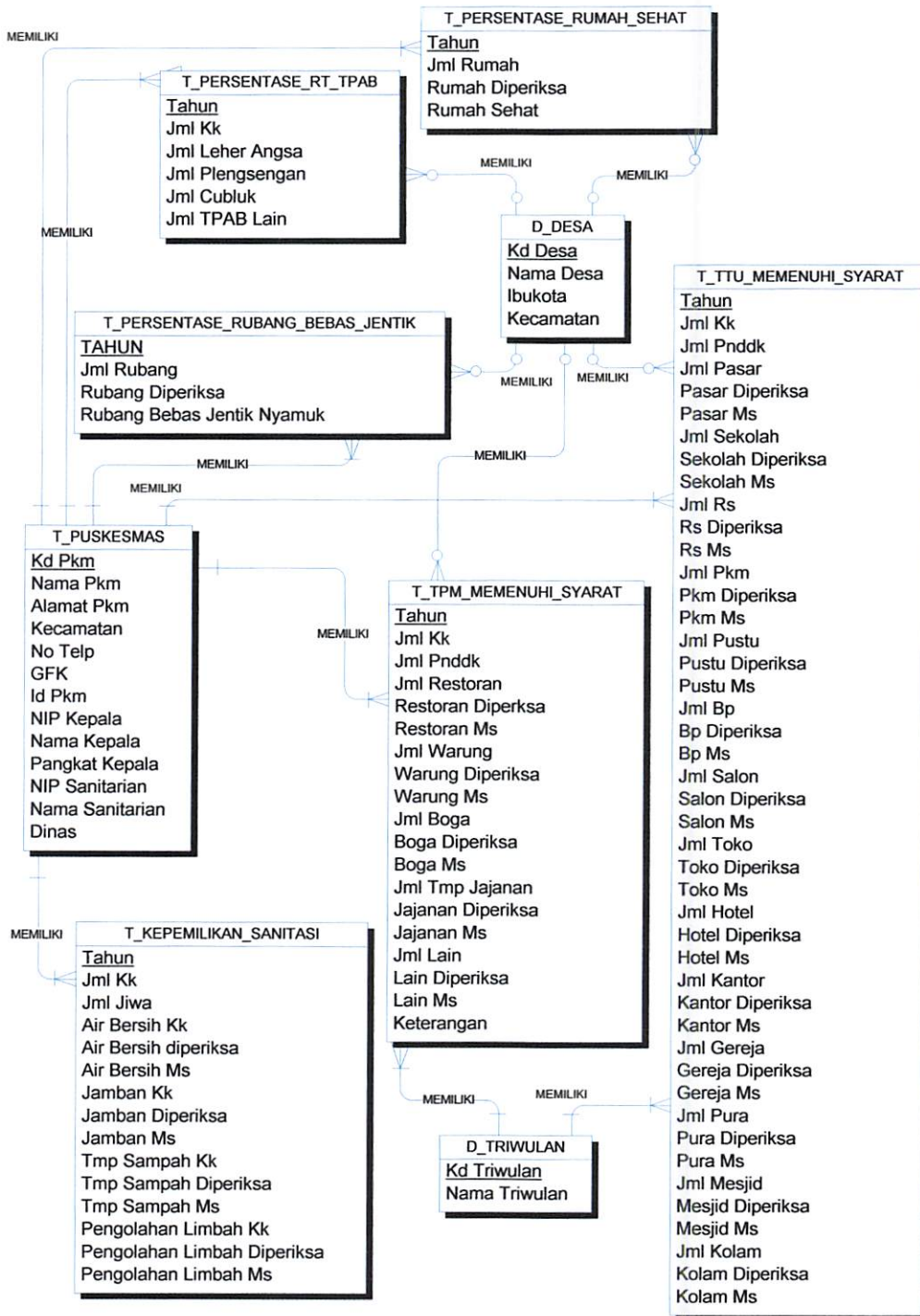




Gambar 3.17 CDM Inspeksi Kesehatan Puskesmas

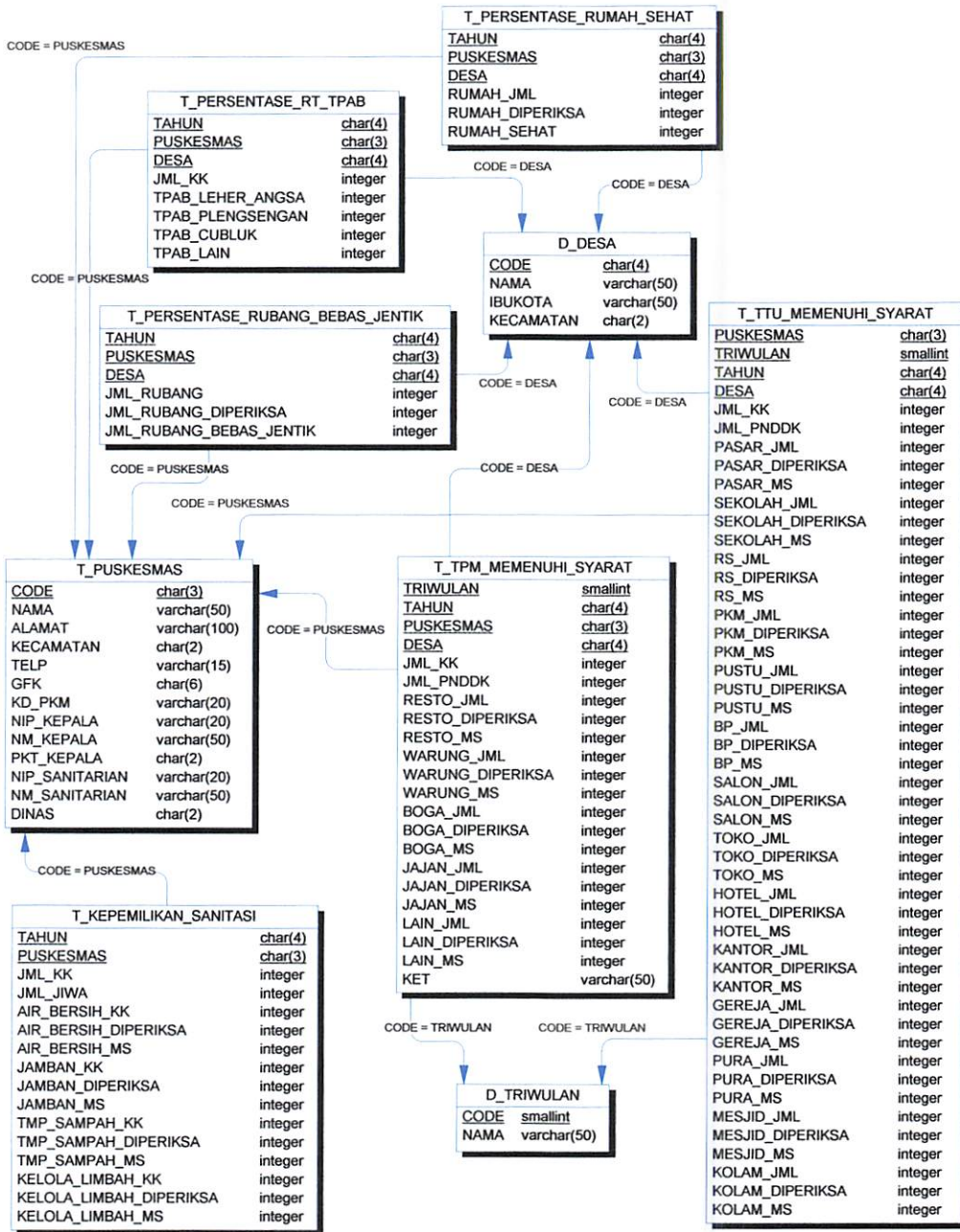


Gambar 3.18 PDM Inspeksi Kesehatan Puskesmas

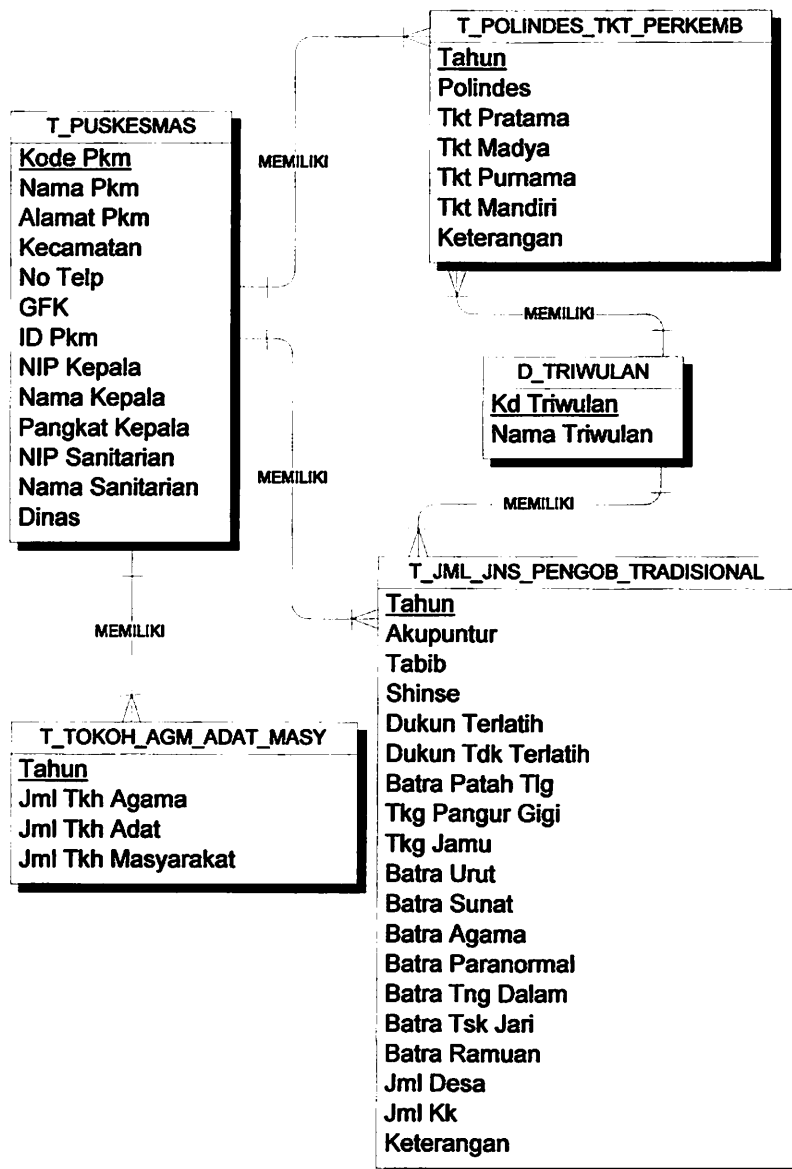


Gambar 3.19 CDM Tempat-Tempat Memenuhi Syarat Kesehatan

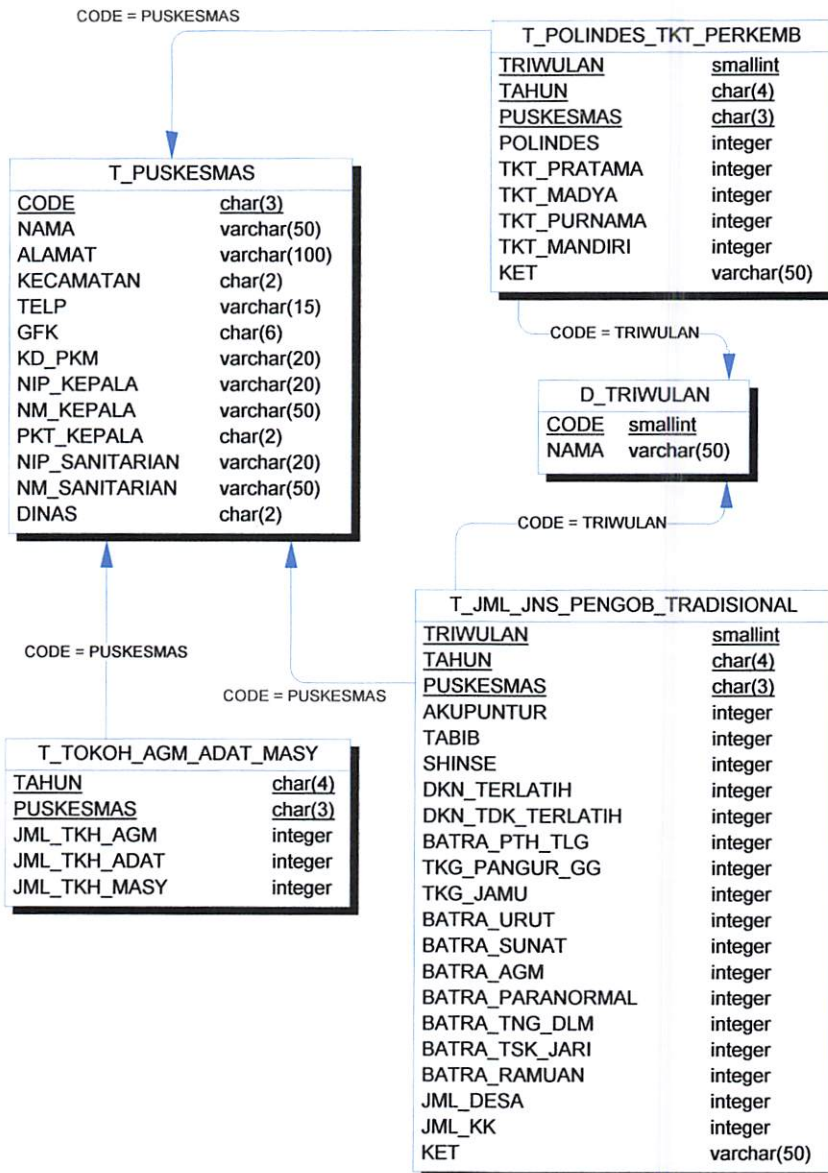




Gambar 3.20 PDM Tempat-Tempat Memenuhi Syarat Kesehatan



Gambar 3.21 CDM Bidang Pendidikan Kesehatan



Gambar 3.22 PDM Bidang Pendidikan Kesehatan

### 3.3.2.2.2 Struktur Tabel – Tabel Yang Digunakan

Berdasarkan CDM dan PDM di atas, maka tabel – tabel yang digunakan antara lain :

1. Tabel D\_BULAN

Dengan banyaknya data yang direkap perbulan sehingga dibutuhkan tabel yang mendeskripsikan data nama bulan.

Tabel 3.9 Struktur Tabel D\_BULAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Smallint	<i>PK</i> , Kode Angka Bulan
2.	BULAN	Varchar(50)	Nama Bulan

## 2. D\_KABUPATEN

Tabel ini berfungsi untuk menyimpan data kabupaten yang ada di Indonesia.

Tabel 3.10 Struktur Tabel D\_KABUPATEN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(4)	<i>PK</i> , Kode Kabupaten
2.	NAMA	Varchar(50)	Nama Kabupaten

## 3. Tabel D\_KECAMATAN

Tabel ini berfungsi untuk menyimpan data kecamatan berdasarkan kabupaten di Indonesia.

Tabel 3.11 Struktur Tabel D\_KECAMATAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(2)	<i>PK</i> , Kode Kecamatan
2.	NAMA	Varchar(50)	Nama Kecamatan
3.	IBUKOTA	Varchar(50)	Pusat Pemerintahan Kecamatan
4.	KABUPATEN	Char(4)	<i>FK</i> , Kode Kabupaten

#### 4. Tabel D\_DESA

Tabel yang berfungsi untuk menyimpan data desa berdasarkan kecamatan.

Tabel 3.12 Struktur Tabel D\_DESA

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(4)	<i>PK</i> , Kode Desa
2.	NAMA	Varchar(50)	Nama Desa
3.	IBUKOTA	Varchar(50)	Pusat Pemerintahan Desa
4.	KECAMATAN	Char(2)	<i>FK</i> , Kode Kecamatan
5.	USER_ENTRI	Varchar(50)	Pengguna yang entri data
6.	IP_ENTRI	Varchar(50)	Alamat komputer yang entri data
7.	TS_ENTRI	Varchar(50)	Waktu entri data

#### 5. Tabel D\_TRIWULAN

Tabel ini berfungsi untuk mendeskripsikan data triwulan dalam setahun.

Tabel 3.13 Struktur Tabel D\_TRIWULAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Smallint	<i>PK</i> , Kode Angka Triwulan
2.	NAMA	Varchar(50)	Nama Triwulan

#### 6. Tabel D\_SATUAN

Tabel ini fungsinya untuk mendeskripsikan data satuan seperti satuan obat dan kontrasepsi.

Tabel 3.14 Struktur Tabel D\_SATUAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(2)	<i>PK</i> , Kode Satuan
2.	NAMA	Varchar(10)	Nama Satuan

7. Tabel D\_OBAT

Tabel ini untuk menyimpan data obat.

Tabel 3.15 Struktur Tabel D\_OBAT

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	char(3)	<i>PK</i> , Kode Obat
2.	NAMA	varchar(100)	Nama Obat
3.	SATUAN	char(2)	<i>FK</i> , kode satuan obat
4.	USER_ENTRI	varchar(50)	Pengguna yang entri data
5.	IP_ENTRI	varchar(50)	Alamat komputer
6.	TS_ENTRI	Datetime	Waktu entri data

8. Tabel D\_JNS\_PENYAKIT

Tabel ini digunakan untuk mendeskripsikan jenis-jenis penyakit yang ada.

Tabel 3.16 Struktur Tabel D\_JNS\_PENYAKIT

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	char(2)	<i>PK</i> , Kode Jenis
2.	NAMA	varchar(200)	Nama Jenis

3.	USER_ENTRI	varchar(50)	
S4.	IP_ENTRI	varchar(50)	
5.	TS_ENTRI	datetime	

### 9. Tabel D\_PENYAKIT

Digunakan untuk menyimpan data-data penyakit berdasarkan jenis penyakitnya.

Tabel 3.17 Struktur Tabel D\_PENYAKIT

No.	Nama Kolom	Tipe data	Keterangan
1.	CODE	Char(2)	<i>PK</i> , Kode Penyakit
2.	JENIS	Char(2)	<i>FK</i> , Kode Jenis Penyakit
3.	NAMA	Varchar(255)	Nama Penyakit
4.	USER_ENTRI	Varchar(50)	
5.	IP_ENTRI	Varchar(50)	
6.	TS_ENTRI	Datetime	

### 10. D\_PELAYANAN

Tabel ini digunakan untuk mendeskripsikan jenis pelayanan di sumber kesehatan masyarakat, yaitu pelayanan Rawat Jalan dan Rawat Inap.

Tabel 3.18 Struktur Tabel D\_PELAYANAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(1)	<i>PK</i> , Kode Pelayanan
2.	NAMA	Varchar(20)	Nama Pelayanan

### 11. D\_PANGKAT

Digunakan untuk mendeskripsikan jenis – jenis pangkat pegawai kesehatan.

Tabel 3.19 Struktur Tabel D\_PANGKAT

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(2)	<i>PK</i> , Kode Pangkat
2.	NAMA	Varchar(10)	Nama golongan pangkat
3	PANGKAT	Varchar(50)	Nama Pangkat

### 12. Tabel D\_DATA\_DINAS

Digunakan untuk menyimpan data dinas kesehatan yang ada.

Tabel 3.20 Struktur Tabel D\_DATA\_DINAS

No.	Nama Kolom	Tipe data	Keterangan
1.	CODE	char(2)	<i>PK</i> , Kode Dinas
2.	NAMA	varchar(100)	Nama Dinas
3.	ALAMAT	varchar(200)	Alamat Dinas
4.	KABUPATEN	char(4)	<i>FK</i> , Kode Kabupaten
5.	NIP_KEPALA	varchar(10)	Nip Kepala Dinas



6.	NM_KEPALA	varchar(50)	Nama Kepala Dinas
7.	PKT_KEPALA	varchar(2)	<i>FK</i> , Kode pangkat kepala dinas
8.	TELP	varchar(15)	No telepon dinas
9.	FAX	varchar(15)	No fax dinas
10.	USER_ENTRI	Varchar(50)	
11.	IP_ENTRI	Varchar(50)	
12	TS_ENTRI	Datetime	

### 13. D\_GUDANG

Tabel ini berfungsi untuk menyimpan data gudang farmasi kabupaten atau instansi pengelola obat.

Tabel 3.21 Struktur Tabel D\_GUDANG

No.	Nama Kolom	Tipe data	Keterangan
1.	CODE	char(6)	<i>PK</i> , Kode Gudang
2.	NAMA	varchar(100)	Nama Gudang
3.	NIP_KEPALA	varchar(10)	Nip kepala gudang
4.	NM_KEPALA	varchar(50)	Nama kepala gudang
5.	PKT_KEPALA	char(2)	<i>FK</i> , Kode Pangkat kepala gudang

6.	DINAS	char(2)	<i>FK</i> , Kode Dinas gudang
7.	USER_ENTRI	varchar(50)	
8.	IP_ENTRI	varchar(50)	
9.	TS_ENTRI	datetime	

#### 14. Tabel T\_PUSKESMAS

Tabel ini berfungsi untuk menyimpan data puskesmas yang ada di kecamatan-kecamatan.

Tabel 3.22 Struktur Tabel T\_PUSKESMAS

No.	Nama Kolom	Tipe data	Keterangan
1.	CODE	char(3)	<i>PK</i> , Kode Puskesmas
2.	NAMA	varchar(50)	Nama Puskesmas
3.	ALAMAT	varchar(100)	Alamat Puskesmas
4.	KECAMATAN	char(2)	<i>FK</i> , Kode Kecamatan
5.	TELP	varchar(15)	No Telepon Puskesmas
6.	GFK	char(6)	<i>FK</i> , Kode Gudang Farmasi
7.	KD_PKM	varchar(20)	Kode Khusus Puskesmas

8.	NIP_KEPALA	varchar(20)	Nip kepala
9.	NM_KEPALA	varchar(50)	Nama kepala
10.	PKT_KEPALA	char(2)	FK, Kode Pangkat kepala
11.	NIP_SANITARIAN	Varchar(20)	Nip Pejabat Sanitarian
12.	NAMA_SANITARIAN	Varchar(50)	Nama Sanitarian
13.	DINAS	char(2)	FK, Kode Dinas
14.	USER_ENTRI	Varchar(50)	
15.	IP_ENTRI	Varchar(50)	
16.	TS_ENTRI	Datetime	

15. Tabel D\_JNS\_KEGIATAN

Berfungsi untuk mendeskripsikan 2 jenis kegiatan puskesmas, yaitu LB3 dan LB4.

Tabel 3.23 Struktur Tabel D\_JNS\_KEGIATAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	Char(2)	PK, Kode Jenis
2.	NAMA	Varchar(50)	Nama Jenis Kegiatan

16. Tabel D\_KEGIATAN

Tabel ini digunakan untuk menyimpan macam-macam kegiatan puskesmas secara umum.

Tabel 3.24 Struktur Tabel D\_KEGIATAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	char(2)	<i>PK</i> , Kode Kegiatan
2.	JENIS	char(2)	<i>FK</i> , Kode jenis kegiatan
3.	NAMA	varchar(100)	Nama kegiatan
4.	USER_ENTRI	varchar(50)	
5.	IP_ENTRI	varchar(50)	
6.	TS_ENTRI	datetime	

17. Tabel D\_SUB\_KEGIATAN

Tabel ini digunakan untuk menyimpan macam-macam sub kegiatan berdasarkan kelompok kegiatan dari tabel D\_KEGIATAN.

Tabel 3.25 Struktur Tabel D\_SUB\_KEGIATAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	char(3)	<i>PK</i> , Kode Sub Kegiatan
2.	KEGIATAN	char(2)	<i>FK</i> , Kode kegiatan
3.	NAMA	varchar(255)	Nama sub kegiatan
4.	USER_ENTRI	varchar(50)	
5.	IP_ENTRI	varchar(50)	

6.	TS_ENTRI	datetime	
----	----------	----------	--

### 18. Tabel D\_ITEM\_KEGIATAN

Tabel ini berfungsi untuk menyimpan bentuk-bentuk kegiatan berdasarkan sub kegiatan.

Tabel 3.26 Struktur Tabel D\_ITEM\_KEGIATAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	char(3)	<i>PK</i> , Kode Item Kegiatan
2.	SUB_KEGIATAN	char(4)	<i>FK</i> , Kode sub kegiatan
3.	NAMA	varchar(100)	Nama item kegiatan
4.	USER_ENTRI	varchar(50)	
5.	IP_ENTRI	varchar(50)	
6.	TS_ENTRI	datetime	

### 19. Tabel D\_ASAL\_KELUHAN

Tabel ini digunakan untuk mendeskripsikan asal keluhan yang ada di sumber kesehatan masyarakat.

Tabel 3.27 Struktur Tabel D\_ASAL\_KELUHAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	char(2)	<i>PK</i> , Kode Asal Keluhan

2.	NAMA	varchar(100)	Nama Asal Keluhan
----	------	--------------	-------------------

## 20. Tabel T\_PEMAKAIAN\_OBAT

Tabel ini berfungsi untuk menyimpan data rekap per bulan tentang pemakaian dan permintaan obat di puskesmas.

Tabel 3.28 Struktur Tabel T\_PEMAKAIAN\_OBAT

No.	Nama Kolom	Tipe data	Keterangan
1.	BULAN	Smallint	PK, Bulan Pelaporan
2.	TAHUN	Char(4)	PK, Tahun Pelaporan
3.	PUSKESMAS	Char(3)	PK, Kode Puskesmas
4.	PERMINTAAN_BLN	Smallint	Bulan Permintaan
5.	PERMINTAAN_THN	Char(4)	Tahun Permintaan
6.	GFK	Char(6)	FK, Kode Gudang Farmasi
7.	TGL_GFK	Smalldatetime	Tanggal GFK
8.	UMUM_BYR	Int	Jumlah kunjungan yang bayar
9.	UMUM_TDK_BYR	Int	Jumlah kunjungan yang tidak bayar

10.	UMUM_LAIN	Int	Jumlah kunjungan lain-lain
11.	ASKES_BYR	Int	Kunjungan Askes yang bayar
12.	ASKES_TDK_BYR	Int	Kunjungan Askes yang tidak bayar
13.	ASKES_LAIN	Int	Kunjungan Askes lain-lain
14.	USER_ENTRI	Varchar(50)	
15.	IP_ENTRI	varchar(50)	
16.	TS_ENTRI	Varchar(50)	

## 21. Tabel T\_PEMAKAIAN\_OBAT\_DETIL

Tabel ini menyimpan data detil dari tabel T\_PEMAKAIAN\_OBAT.

Tabel 3.29 Struktur Tabel T\_PEMAKAIAN\_OBAT\_DETIL

No.	Nama Kolom	Tipe data	Keterangan
1.	BULAN	smallint	<i>PK</i> , Bulan Pelaporan
2.	TAHUN	char(4)	<i>PK</i> , Tahun Pelaporan
3.	PUSKESMAS	char(3)	<i>PK</i> , Kode Puskesmas
4.	OBAT	char(3)	<i>PK</i> , Kode Obat
5.	STOK_AWAL	int	Stok awal obat

6.	PENERIMAAN	int	Penerimaan
7.	PERSEDIAAN	int	Persediaan yang ada
8.	PEMAKAIAN	int	Pemakaian
9.	SISA_STOK	int	Sisa stok
10.	STOK_OPT	int	Stok Opt
11.	PERMINTAAN	int	Permintaan obat
12.	I	int	Pemberian I
13.	ASKES	int	Pemberian Askes
14.	APBD	int	Pemberian APBD
15.	LAIN2	int	Pemberian Lain-lain
16.	KET	varchar(50)	Keterangan
17.	USER_ENTRI	varchar(50)	
18.	IP_ENTRI	varchar(50)	
19.	TS_ENTRI	varchar(50)	

## 22. Tabel D\_KELOMPOK\_UMUR

Tabel ini berfungsi untuk mendeskripsikan kelompok umur pasien



Tabel 3.30 Struktur Tabel D\_KELOMPOK\_UMUR

No.	Nama Kolom	Tipe Data	Keterangan
1.	CODE	char(2)	<i>PK</i> , Kode
2.	NAMA	varchar(10)	Kelompok Umur

23. Tabel T\_REKAP\_LB1

Berfungsi untuk menyimpan data rekapitulasi angka kesakitan di puskesmas.

Tabel 3.31 Struktur Tabel T\_REKAP\_LB1

No.	Nama Kolom	Tipe data	Keterangan
1.	BULAN	smallint	<i>PK</i> , Bulan Pelaporan
2.	TAHUN	char(4)	<i>PK</i> , Tahun Pelaporan
3.	PUSKESMAS	char(3)	<i>PK</i> , Kode Puskesmas
4.	PENYAKIT	char(4)	<i>PK</i> , Kode Penyakit
5.	UMUR	char(2)	<i>PK</i> , Kode Kelompok Umur
6.	BARU_L	int	Jumlah Laki-laki (baru)
7.	BARU_P	int	Jumlah Perempuan (baru)
8.	LAMA_L	int	Jumlah Laki-laki (lama)
9.	LAMA_P	int	Jumlah Perempuan (lama)
10.	USER_ENTRI	varchar(50)	

11.	IP_ENTRI	varchar(50)	
12.	TS_ENTRI	Datetime	

#### 24. Tabel T\_BULANAN\_LB3

Tabel ini berfungsi untuk menyimpan cakupan kegiatan-kegiatan LB3 puskesmas tiap bulan.

Tabel 3.32 Struktur Tabel T\_BULANAN\_LB3

No.	Nama Kolom	Tipe Data	Keterangan
1.	BULAN	smallint	<i>PK</i> , Bulan Pelaporan
2.	TAHUN	char(4)	<i>PK</i> , Tahun Pelaporan
3.	PUSKESMAS	char(3)	<i>PK</i> , Kode Puskesmas
4.	KEGIATAN	char(3)	<i>PK</i> , Kode Kegiatan LB3
5.	CAKUPAN	int	Jumlah/cakupan kegiatan LB3
6.	USER_ENTRI	varchar(50)	
7.	IP_ENTRI	varchar(50)	
8.	TS_ENTRI	datetime	

#### 25. Tabel T\_BULANAN\_LB4

Tabel ini berfungsi untuk menyimpan cakupan kegiatan-kegiatan LB4 puskesmas tiap bulan.

Tabel 3.33 Struktur Tabel T\_BULANAN\_LB4

No.	Nama Kolom	Tipe Data	Keterangan
1.	BULAN	smallint	<i>PK</i> , Bulan Pelaporan
2.	TAHUN	char(4)	<i>PK</i> , Tahun Pelaporan
3.	PUSKESMAS	char(3)	<i>PK</i> , Kode Puskesmas
4.	KEGIATAN	char(3)	<i>PK</i> , Kode Kegiatan LB4
5.	CAKUPAN	int	Jumlah/cakupan kegiatan LB4
6.	USER_ENTRI	varchar(50)	
7.	IP_ENTRI	varchar(50)	
8.	TS_ENTRI	datetime	

26. Tabel T\_KEG\_JAMKESMAS\_PKM

Digunakan untuk menyimpan data-data kegiatan Jamkesmas di Puskesmas per bulan

Tabel 3.34 Struktur Tabel T\_KEG\_JAMKESMAS\_PKM

No.	Nama Kolom	Tipe data	Keterangan
1.	BULAN	smallint	<i>PK</i> , Bulan pelaporan
2.	TAHUN	char(4)	<i>PK</i> , Tahun Pelaporan
3.	PUSKESMAS	char(3)	<i>PK</i> , Kode Puskesmas

4.	SAS_MASKIN_KEC	int	Sasaran Maskin se kecamatan
5.	SAS_MASKIN_KARTU	int	Sasaran maskin memiliki kartu jamkesmas
6.	SAS BUMIL_KEC	int	Sasaran bumil se kecamatan
7.	PEL_KUN_RJ	int	Pelayanan kunjungan RJ
8.	PEL_KUMULATIF_RJ	int	Pelayanan kumulatif RJ
9.	PEL_KUN_RI	int	Pelayanan kunjungan RI
10.	PEL_KUMULATIF_RI	int	Pelayanan kumulatif RI
11.	PEL_KASUS_DIRUJUK	int	Pelayanan kasus dirujuk
12.	PEL_KUN_PEM_HAMIL	int	Pemeriksaan kehamilan
13.	PEL_KUN_PEM_BAYI	int	Pemeriksaan bayi
14.	PEL_PERSALINAN_NAK ES	int	Persalinan

15.	PEL_BUMIL_DIRUJUK	int	Bumil dirujuk
16.	PEL_MASKIN_KRT_JAM KESMAS	int	Maskin memiliki kartu jamkesmas
17.	PEL_MASKIN_TNP_KRT	int	Maskin tanpa kartu
18.	PEN_SISA_DANA	money	Sisa dana tahun lalu
19	PEN_TOTAL_DANA	money	Total dana yankes maskin
20.	PEN_PELYNAN_DSR	money	Biaya Pelayanan dasar
21.	PEN_PERSALINAN	money	Biaya persalinan
22.	PEN_PELYNAN_DSR_TL H_DIKEL	money	Biaya pelayanan dasar dikeluarkan
23.	PEN_PERSALINAN_TLH _DIKEL	money	Biaya persalinan dikeluarkan
24.	USER_ENTRI	varchar(50)	
25.	IP_ENTRI	varchar(50)	
26.	TS_ENTRI	datetime	

## 27. Tabel T\_JAMKESMAS\_PUSKESMAS

Tabel ini digunakan untuk menyimpan data rekapitulasi pelayanan jamkesmas di puskesmas per bulan.

Tabel 3.35 Struktur Tabel T\_JAMKESMAS\_PUSKESMAS

No.	Nama Kolom	Tipe data	Keterangan
1.	BULAN	smallint	PK, Bulan pelaporan
2.	TAHUN	char(4)	PK, Tahun Pelaporan
3.	PUSKESMAS	char(3)	PK, Kode Puskesmas
4.	TERDAFTAR_BPS	int	Peserta yang terdaftar
5.	MEMILIKI_KRT_JAMK	int	Peserta memiliki kartu jamkesmas
6.	BUMIL_MISKIN	int	Peserta bumil miskin
7.	RJTP	money	RJTP bulan ini
8.	RJTP_SD	money	RJTP s/d bulan ini
9.	RITP	money	RITP bulan ini
10.	RITP_SD	money	RITP s/d bulan ini

11.	JML_KSS_DIRUJUK	int	Jumlah kasus dirujuk
12.	JML_KUN_KEHAMILAN	int	Jumlah kunjungan kehamilan
13.	JML_KUN_BAYI	int	Pemeriksaan bayi
14.	JML_KUN_LINAKES	int	Persalinan
15.	JML_BUMIL_DIRUJUK	int	Bumil dirujuk
16.	JML_MASKIN_KARTU	int	Maskin memiliki kartu jamkesmas
17.	JML_MASKIN_BKN_KARTU	int	Maskin tanpa kartu
18.	USER_ENTRI	varchar(50)	
19	IP_ENTRI	varchar(50)	
20.	TS_ENTRI	varchar(50)	

## 28. Tabel T\_PENANGANAN\_KELUHAN\_PKM

Tabel ini digunakan untuk mendata asal dan jenis penanganan keluhan di puskesmas.

Tabel 3.36 Struktur Tabel T\_PENANGANAN\_KELUHAN\_PKM

No.	Nama Kolom	Tipe Data	Keterangan
1.	BULAN	smallint	<i>PK</i> , Bulan Pelaporan
2.	TAHUN	char(4)	<i>PK</i> , Tahun Pelaporan
3.	PUSKESMAS	char(3)	<i>PK</i> , Kode Puskesmas
4.	ASAL_KELUHAN	char(2)	<i>FK</i> , Kode asal keluhan
5.	ADM_JML	int	Jumlah keluhan jenis administrasi
6.	ADM_TERTANGANI	int	jumlah yang tertangani
7.	PELKES_JML	int	Jumlah keluhan jenis pelayanan kesehatan
8.	PELKES_TERTANGANI	int	Jumlah yang tertangani
9.	USER_ENTRI	varchar(50)	
10.	IP_ENTRI	varchar(50)	
11.	TS_ENTRI	datetime	



### 29. Tabel T\_PENYAKIT\_TERBANYAK\_PKM

Tabel ini berfungsi untuk menyimpan data 10 penyakit terbanyak dalam satu bulan di puskesmas yang bersangkutan.

Tabel 3.37 Struktur Tabel T\_PENYAKIT\_TERBANYAK\_PKM

No.	Nama Kolom	Tipe Data	Keterangan
1.	BULAN	smallint	PK, Bulan Pelaporan
2.	TAHUN	char(4)	PK, Tahun Pelaporan
3.	PUSKESMAS	char(3)	PK, Kode Puskesmas
4.	PELAYANAN	char(2)	FK, jenis pelayanan
5.	NO_URUT	smallint	No urut penyakit
6.	PENYAKIT	char(4)	FK, Kode penyakit
7.	JML_KASUS	int	Jumlah kasus
8.	USER_ENTRI	varchar(50)	
9.	IP_ENTRI	varchar(50)	
10.	TS_ENTRI	datetime	

### 30. Tabel T\_POLINDES\_TKT\_PERKEMBANGAN

Tabel ini digunakan untuk menyimpan data triwulanan tentang poliklinik desa dan tingkat perkembangannya.

Tabel 3.38 Struktur Tabel T\_POLINDES\_TKT\_PERKEMBANGAN

No.	Nama Kolom	Tipe Data	Keterangan
1.	TRIWULAN	smallint	PK, Triwulan Pelaporan
2.	TAHUN	char(4)	PK, Tahun Pelaporan
3.	PUSKESMAS	char(3)	PK, Kode Puskesmas
4.	POLINDES	int	tingkat Polindes
5.	TKT_PRATAMA	int	Tingkat Perkembangan
6.	TKT_MADYA	int	Tingkat Perkembangan
7.	TKT_PURNAMA	int	Tingkat Perkembangan
8.	TKT_MANDIRI	int	Tingkat Perkembangan
9.	KET	varchar(50)	Keterangan
10.	USER_ENTRI	varchar(50)	
11.	IP_ENTRI	varchar(50)	
12.	TS_ENTRI	datetime	

31. Tabel T\_JML\_JNS\_PENGOBATAN\_TRADISIONAL

Tabel ini berfungsi untuk menyimpan data tentang jumlah dan jenis pengobatan tradisional yang ada di ruang lingkup puskesmas.

Tabel 3.39 Struktur Tabel T\_JML\_JNS\_PENGOBATAN\_TRADISIONAL

No.	Nama Kolom	Tipe Data	Keterangan
1.	TRIWULAN	smallint	PK, Triwulan Pelaporan
2.	TAHUN	char(4)	PK, Tahun Pelaporan
3.	PUSKESMAS	char(3)	PK, Kode Puskesmas
4.	AKUPUNTUR	int	Jumlah ahli akupuntur
5.	TABIB	int	Jumlah tabib
6.	SHINSE	int	Jumlah shinse
7.	DKN_TERLATIH	int	Jumlah dukun terlatih
8.	DKN_TDK_TERLATIH	int	Jumlah dukun tidak terlatih
9.	BATRA_PTH_TLG	int	Ahli patah tulang
10.	TKG_PANGUR_GG	int	Ahli gigi
11.	TKG_JAMU	int	Tukang jamu
12.	BATRA_URUT	int	Tukang urut
13.	BATRA_SUNAT	int	Mantri sunat
14.	BATRA_AGM	int	Ahli Pengobatan melalui agama

15.	BATRA_PARANOR MAL	int	paranormal
16.	BATRA_TNG_DLM	int	Ahli tenaga dalam
17.	BATRA_TSK_JARI	int	Ahli tusuk jari
18.	BATRA_RAMUAN	int	Ahli ramuan
19.	JML_DESA	int	Jumlah desa
20.	JML_KK	int	Jumlah kepala keluarga
21.	KET	varchar(50)	Keterangan
22.	USER_ENTRI	varchar(50)	
23.	IP_ENTRI	varchar(50)	
24.	TS_ENTRI	datetime	

### 32. Tabel T\_TOKOH\_AGM\_ADAT\_MASY

Digunakan untuk menyimpan data jumlah tokoh agama, tokoh adat, dan tokoh masyarakat dalam ruang lingkup puskesmas.

Tabel 3.40 Struktur Tabel T\_TOKOH\_AGM\_ADAT\_MASY

No.	Nama Kolom	Tipe Data	Keterangan
1.	TAHUN	char(4)	PK, Tahun Pelaporan

2.	PUSKESMAS	char(3)	PK, Kode Puskesmas
3.	JML_TKH_AGM	int	Jumlah tokoh agama
4.	JML_TKH_ADAT	int	Jumlah tokoh adat
5.	JML_TKH_MASY	int	Jumlah tokoh masyarakat
6.	USER_ENTRI	varchar(50)	
7.	IP_ENTRI	varchar(50)	
8.	TS_ENTRI	datetime	

33. Tabel T\_JAMKESMAS\_RS

Tabel ini digunakan untuk menyimpan data rekapitulasi pelayanan jamkesmas di rumah sakit per bulan.

Tabel 3.41 Struktur tabel T\_JAMKESMAS\_RS

No.	Nama Kolom	Tipe data	Keterangan
1.	BULAN	smallint	PK, Bulan pelaporan
2.	TAHUN	char(4)	PK, Tahun Pelaporan
3.	RS	char(2)	PK, Kode Rumah Sakit
4.	TOT_TMP_TIDUR	int	Jumlah total tempat tidur
5.	TMP_TIDUR_KLS3	int	Jumlah tempat

			tidur kelas III
6.	PASIEN_RJ	int	Jumlah pasien rawat jalan
7.	PASIEN_RJ_KLS3	int	Jumlah pasien rawat jalan kelas III
8.	PASIEN_RI	int	Jumlah pasien rawat inap
9.	PASIEN_RI_KLS3	int	Jumlah pasien rawat inap kelas III
10.	HR_RWT_PAS_KLS3	int	Jumlah hari perawatan pasien kelas III
11.	KUN_MISKIN_RJ	int	Jumlah kunjungan rawat jalan maskin
12.	KUN_MISKIN_RI	int	Jumlah kunjungan rawat inap maskin
13.	KUN_MISKIN_HR_RWT	int	Jumlah hari perawatan maskin

14.	KUN_KEMATIAN_KRG_48	int	Jumlah kematian kurang dari 48 jam
15.	KUN_KEMATIAN_LBH_48	int	Jumlah kematian lebih dari 48 jam
16.	KUN_PERSALINAN	int	Jumlah persalinan
17.	MASKIN_KARTU	int	Maskin memiliki kartu jamkesmas
18.	MASKIN_BKN_KARTU	int	Maskin tanpa kartu
19.	USER_ENTRI	varchar(50)	
20.	IP_ENTRI	varchar(50)	
21.	TS_ENTRI	datetime	

#### 34. Tabel T\_PENANGANAN\_KELUHAN\_RS

Tabel ini digunakan untuk mendata asal dan jenis penanganan keluhan di rumah sakit.

Tabel 3.42 Struktur Tabel T\_PENANGANAN\_KELUHAN\_RS

No.	Nama Kolom	Tipe Data	Keterangan
1.	BULAN	smallint	PK, Bulan Pelaporan
2.	TAHUN	char(4)	PK, Tahun Pelaporan
3.	RS	char(2)	PK, Kode Rumah Sakit
4.	ASAL_KELUHAN	char(2)	FK, Kode asal keluhan
5.	ADM_JML	int	Jumlah keluhan jenis administrasi
6.	ADM_TERTANGANI	int	jumlah yang tertangani
7.	PELKES_JML	int	Jumlah keluhan jenis pelayanan kesehatan
8.	PELKES_TERTANGANI	int	Jumlah yang tertangani
9.	USER_ENTRI	varchar(50)	
10.	IP_ENTRI	varchar(50)	
11.	TS_ENTRI	datetime	

### 35. Tabel T\_PENYAKIT\_TERBANYAK\_RS

Tabel ini berfungsi untuk menyimpan data 10 penyakit terbanyak dalam satu bulan di rumah sakit yang bersangkutan.



Tabel 3.43 Struktur Tabel T\_PENYAKIT\_TERBANYAK\_RS

No.	Nama Kolom	Tipe Data	Keterangan
1.	BULAN	smallint	<i>PK</i> , Bulan Pelaporan
2.	TAHUN	char(4)	<i>PK</i> , Tahun Pelaporan
3.	RS	char(2)	<i>PK</i> , Kode Rumah Sakit
4.	PELAYANAN	char(2)	<i>FK</i> , jenis pelayanan
5.	NO_URUT	smallint	No urut penyakit
6.	PENYAKIT	char(4)	<i>FK</i> , Kode penyakit
7.	JML_KASUS	int	Jumlah kasus
8.	USER_ENTRI	varchar(50)	
9.	IP_ENTRI	varchar(50)	
10.	TS_ENTRI	datetime	

### 3.3.3 Desain Antarmuka Aplikasi

Sesuai dengan spesifikasi sistem di atas, sistem informasi ini diharapkan dapat dengan mudah dipakai oleh banyak orang dan operator yang mengoperasikan sistem ini. Untuk itu harus dibuat desain antarmuka yang mudah dipahami dan tidak terlalu rumit.

Ada empat macam desain antarmuka pada aplikasi ini baik halaman untuk admin maupun halaman sistem informasi itu sendiri, yaitu desain halaman login,

halaman utama, entri data, dan laporan. Sedangkan desain menu ditampilkan berdasarkan hak akses setiap pengguna sistem.

### 3.3.3.1 Desain Halaman Login

The image shows a login page design with a header section. Below the header, there are two input fields: 'Username' and 'Password'. Below these fields are two buttons: 'Login' and 'Reset'. To the right of the input fields, there is a welcome message: 'Selamat Datang di Sistem Informasi Kesehatan Daerah Kabupaten Flores Timur'.

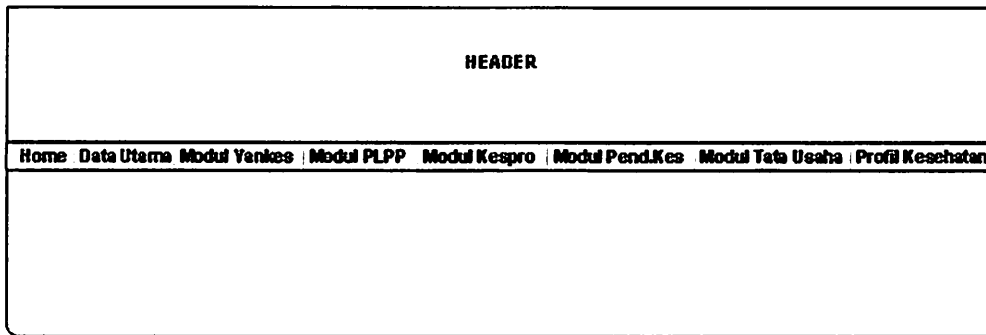
Gambar 3.23 Desain Halaman Login

Desain halaman login untuk admin dan sistem informasi tidak ada perbedaan, hanya terdapat dua entri untuk nama pengguna dan password. Kemudian dua tombol untuk login dan reset halaman.

### 3.3.3.2 Desain Menu Aplikasi

HEADER	
Home	Administrator
Halaman Utama Peraturan Logout	Entri Hak Akses Aplikasi Entri Halaman Entri Menu Aplikasi Entri Role Entri Submenu Aplikasi Entri User

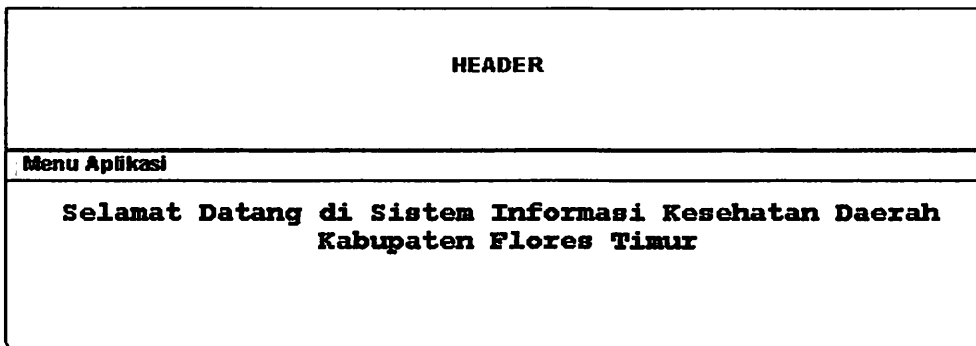
Gambar 3.24 Desain Menu Halaman Admin



Gambar 3.25 Desain Menu Halaman Sistem Informasi

Ada perbedaan antara menu halaman admin dan halaman sistem informasi. Untuk halaman admin desain menu adalah statis tidak berubah - ubah. Sedangkan menu sistem informasi didesain secara dinamis, dimana tampilan akan berubah-ubah sesuai dengan hak akses pengguna aplikasi dan ruang lingkup instansi.

### 3.3.3.3 Desain Halaman Utama



Gambar 3.26 Desain Menu Halaman Utama Sistem Informasi

Tidak ada perbedaan pada desain halaman utama baik untuk halaman admin maupun halaman sistem informasi.

### 3.3.3.4 Desain Halaman Entri Data

HEADER

Menu Aplikasi

Label Data :

<input type="checkbox"/> [Edit]			
<input type="checkbox"/> [Edit]			
<input type="checkbox"/> [Edit]			

Gambar 3.27 Desain Menu Halaman Entri

Desain halaman entri antara halaman admin dan sistem informasi tidak ada perbedaan. Label data diisi dengan judul data yang akan dientri, sedangkan data entri bisa berupa kotak *TextBox* ataupun *ComboBox* yang datanya akan disimpan ke dalam tabel. Tampilan tabel digunakan untuk menampilkan data yang telah tersimpan, sedangkan tanda centang berfungsi untuk memberi tanda pada bahwa data yang dicentang adalah data yang akan dihapus, begitu juga dengan teks *[Edit]* berfungsi untuk member tanda pada program bahwa data pada baris tersebut akan dihapus.

### 3.3.3.5 Desain Halaman Laporan

HEADER

Menu Aplikasi

Judul Laporan

Parameter : Pilih... ▼

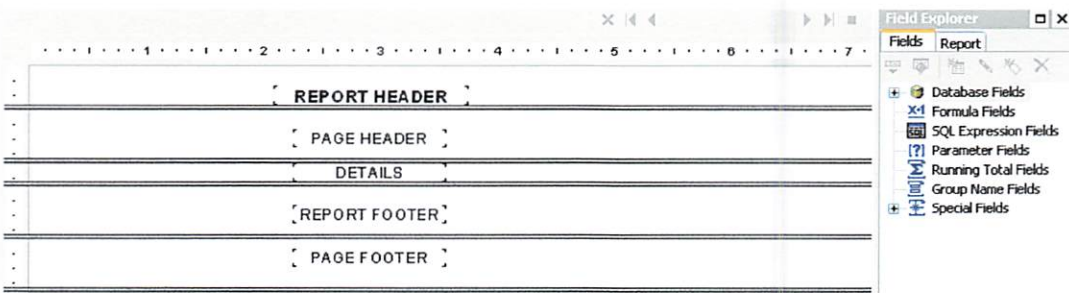
Refresh

Export Report : MS Word Document (DOC) ▼    Export    Print

Tampilan Laporan

Gambar 3.28 Desain Menu Halaman Laporan

Desain halaman laporan terdiri dari beberapa kotak pilihan atau *ComboBox* yang fungsinya sebagai *parameter* atau masukan untuk laporan. *Parameter* ini bisa berupa bulan, tahun, triwulan, dan nama sumber kesehatan masyarakat. Sedangkan kotak pilihan paling bawah berfungsi untuk memilih jenis dokumen jika laporan akan disimpan ke dalam format dokumen lain, misalnya *Word Document*, *PDF*, atau *Excel*. Untuk tampilan laporan pada halaman laporan, dibuat dengan menggunakan *Crystal Reports*. Berikut ini desain tampilan laporan pada *Crystal Reports*.



Gambar 3.29 Desain Laporan pada *Crystal Reports*

Sistem pelaporan yang digunakan untuk sistem informasi di dinas kesehatan terdiri dari beberapa laporan, antara lain :

1. Laporan Rekap Tahunan Kegiatan Semua Puskesmas (LB3&LB4)
2. Laporan Rekap Tahunan Asal dan Jenis Penanganan Keluhan
3. Laporan Rekap 10 Penyakit Terbanyak Rawat Inap Puskesmas dan RS
4. Laporan Rekap 10 Penyakit Terbanyak Rawat Jalan Puskesmas dan RS
5. Laporan Rekap Pendanaan Pelayanan Jamkesmas Puskesmas
6. Laporan List Penderita AFP
7. Laporan List Tersangka DBD
8. Laporan Perkembangan HIV / AIDS
9. Laporan Persentase Tempat Umum dan Pengelolaan Makanan (TUPM) Sehat
10. Laporan Persentase Rumah Sehat
11. Laporan Persentase Keluarga Memiliki Akses Air Bersih
12. Laporan Persentase Rumah Tangga yang Memiliki Tempat Pembuangan Air Besar
13. Rekapitulasi Laporan Bulanan Klinik KB Tingkat Kabupaten/Kota
14. Laporan Data Persediaan Alokon
15. Laporan Bulanan Pengendalian Lapangan Program Keluarga Berencana Nasional Tingkat Kabupaten/Kota
16. Laporan Indikator Derajat Kesehatan 2010

## **BAB IV**

### **IMPLEMENTASI DAN PENGUJIAN**

#### **4.1 Implementasi Sistem**

Tahap implementasi pengembangan perangkat lunak merupakan proses perubahan spesifikasi sistem menjadi sistem yang dapat dijalankan. Tahap ini merupakan lanjutan dari proses perancangan, yaitu proses pemrograman perangkat lunak sesuai dengan spesifikasi dan desain sistem.

Aplikasi sistem informasi kesehatan ini menggunakan basis data SQL Server 2005 yang berfungsi sebagai media penyimpanan data atau informasi yang terkumpul, yang terdiri dari beberapa tabel yang saling berhubungan. Sedangkan untuk menjembatani antara informasi yang akan dibuat dengan basis data yang ada, digunakan ASP.Net 2.0. Implementasi dari desain aplikasi berupa implementasi struktur data dari masing-masing proses. Program bantu untuk mengimplementasikan struktur data pada masing-masing proses menggunakan Visual Studio.Net 2005.

Ada beberapa program utama yang sangat penting agar aplikasi ini dapat berjalan. Diantaranya adalah Koneksi basis data, pembacaan data, penambahan data, perubahan data, dan penghapusan data.

##### **4.1.1 Koneksi Basis Data**

Teknik pemrograman dengan menggunakan aplikasi yang berbasis *.Net Framework* adalah teknik pemrograman yang berorientasi pada objek. Untuk itu, dibuat suatu *Class* untuk membuat program utama koneksi database, dalam hal ini adalah *Class DbConnection*.

Sebagai landasan untuk melakukan koneksi basis data SQL Server, digunakan *Class* dari *SqlConnection*. Ada tiga *parameter* atau variabel yang harus diketahui untuk melakukan koneksi, yaitu nama komputer server, nama pengguna basis data, dan nama basis data.

```
Imports System
Imports System.Data
Imports System.Data.SqlClient

Public Class DBConnection

Public Function getSQLConn() As SqlConnection
    Dim sqlConn As SqlConnection = New SqlConnection(getConnString())
    Return sqlConn
End Function

Public Function getConnInfo() As CrystalDecisions.Shared.ConnectionInfo
    Dim connInfo As New CrystalDecisions.Shared.ConnectionInfo()
    With connInfo
        .ServerName = "10.0.0.2"
        .DatabaseName = "SIMKESDA"
        .UserID = " simkes"
        .Password = " simkes"
    End With
    Return connInfo
End Function

Public Function getConnString() As String
    Dim connInfo As CrystalDecisions.Shared.ConnectionInfo = getConnInfo()
    Dim conn As String = "workstation id=" & connInfo.ServerName
    conn &= ";packet size=4096;user id=" & connInfo.UserID
    conn &= ";password=" & connInfo.Password
    conn &= ";data source=" & connInfo.ServerName
    conn &= ";persist security info=False;initial catalog=" &
    connInfo.DatabaseName
    Return conn
End Function

Public Function getConfConn() As SqlConnection
    Dim sqlConn As SqlConnection = New SqlConnection(getConfString())
    Return sqlConn
End Function
```



```

Public Function getConfInfo() As
CrystalDecisions.Shared.ConnectionInfo
  Dim connInfo As New CrystalDecisions.Shared.ConnectionInfo()
  With connInfo
    .ServerName = "10.0.0.2"
    .DatabaseName = "ADMIN_SIMKES"
    .UserID = "simkes"
    .Password = "simkes"
  End With
  Return connInfo
End Function

Public Function getConfString() As String
  Dim connInfo As CrystalDecisions.Shared.ConnectionInfo =
getConfInfo()
  Dim conn As String = "workstation id=" & connInfo.ServerName
  conn &= ";packet size=4096;user id=" & connInfo.UserID
  conn &= ";password=" & connInfo.Password
  conn &= ";data source=" & connInfo.ServerName
  conn &= ";persist security info=False;initial catalog=" &
connInfo.DatabaseName
  Return conn
End Function

End Class

```

Di dalam kode program di atas, terdapat dua koneksi basis data, yaitu basis data utama *SIMKESDA* dan basis data untuk admin *ADMIN\_SIMKES*. Selanjutnya untuk membuat koneksi ke basis data, setiap unit program harus mengakses program di atas dengan membuat objek baru dari kelas *DbConnection* dan memanggil fungsi *getSqlConn()*.

#### 4.1.2 Pengoperasian Basis Data

Seperti proses koneksi di atas, pengoperasian basis data juga menggunakan suatu kelas, dalam hal ini dibuat *class MainLibrary*. Proses utama dari pengoperasian basis data meliputi pembacaan data, penyimpanan, pengubahan, dan penghapusan data.

```

Imports Microsoft.VisualBasic
Imports System
Imports System.Data
Imports System.Data.SqlClient

Public Class MainLibrary
    Dim dbLib As New DBConnection

    Dim dbConn As SqlConnection = dbLib.GetSQLConn
    Dim dbCmd As SqlCommand = dbConn.CreateCommand
    Dim drVar As SqlDataReader

    Private Const strVal As String = "CODE"
    Private Const strText As String = "VALUE"
    Public IPClient As String =
        HttpContext.Current.Request.UserHostAddress
    Dim CPH As String = "ContentPlaceholder1"

    Public Function inputData(ByVal tabel As String, ByVal field As String,
        ByVal value As String) As Boolean
        field &= ", USER_ENTRI, IP_ENTRI, TS_ENTRI"
        value &= ", '" & IPClient & "', '" & Format(Now(), "M/d/yyyy
            hh:mm:ss") & "'"

        Dim q As String = "INSERT INTO " & tabel & "(" & field & ") "
        q &= "VALUES(" & value & ")"

        If dbConn.State = ConnectionState.Closed Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        Try
            dbCmd.ExecuteNonQuery()
            Return True
        Catch ex As Exception
            Return False
        End Try
        dbConn.Close()
    End Function

    Public Function UpdateData(ByVal tabel As String, ByVal fieldUpdate As
        String, ByVal kondisi As String) As Boolean
        Dim q As String = "UPDATE " & tabel & " SET " & fieldUpdate & " WHERE
            " & kondisi

        If dbConn.State = ConnectionState.Closed Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
    End Function

```

```

    Try
        dbCmd.ExecuteNonQuery()
        Return True
    Catch ex As Exception
        Return False
    End Try
    dbConn.Close()

End Function

Public Function DeleteData(ByVal tabel As String, ByVal condition As
String) As Boolean
    Dim q As String = "DELETE FROM " & tabel & " WHERE (" & condition &
    ")"

    If dbConn.State = ConnectionState.Closed Then
        dbConn.Open()
    End If

    dbCmd.CommandText = q
    Try
        dbCmd.ExecuteNonQuery()
        Return True
    Catch ex As SqlException
        Return False
    End Try
    dbConn.Close()
End Function

Public Sub comboBind(ByVal combo As
System.Web.UI.WebControls.DropDownList, ByVal data As DataView)
    combo.DataSource() = data
    combo.DataTextField = strText
    combo.DataValueField = strVal
    combo.DataBind()
End Sub

Public Sub showDataGrid(ByVal grid As
System.Web.UI.WebControls.GridView, ByVal data As DataView)
    grid.DataSource = data
    grid.DataBind()
End Sub

Public Sub showDataGrid(ByVal grid As
System.Web.UI.WebControls.GridView, ByVal data As DataView, ByVal
index As Integer)
    grid.PageIndex = index
    grid.DataSource = data
    grid.DataBind()
End Sub

Public Sub CheckGrid(ByVal grid As System.Web.UI.WebControls.GridView,
ByVal checkState As Boolean)
    For Each row As GridViewRow In grid.Rows
        Dim cb As CheckBox = row.FindControl("cbselect")
        If cb IsNot Nothing Then
            cb.Checked = checkState
        End If
    End For
End Sub

```

```

Public Function getData(ByVal tabel As String) As DataView
    Dim dt As New DataTable
    Dim dr As DataRow
    dt.Columns.Add(New DataColumn("CODE", GetType(String)))
    dt.Columns.Add(New DataColumn("VALUE", GetType(String)))

    Dim q As String = "SELECT CODE, NAMA FROM " & tabel & " ORDER BY
    CODE"

    If (dbConn.State = ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar = dbCmd.ExecuteReader()

    While drVar.Read()
        dr = dt.NewRow()
        dr(0) = drVar.GetString(0)
        dr(1) = drVar.GetString(1)
        dt.Rows.Add(dr)
    End While
    drVar.Close()
    dbConn.Close()

    Dim dv As New DataView(dt)
    Return dv

End Function

End Class

```

Pada potongan kode program di atas, untuk pengoperasian basis data digunakan beberapa fungsi, yaitu *inputData (tabel ,field, value)*, *UpdateData (tabel ,fieldupdate, kondisi)*, *DeleteData (tabel ,condition)*, dan *getData (tabel)*. Tiga fungsi pertama mengembalikan nilai *Boolean* digunakan untuk penambahan data, pengubahan data, dan penghapusan data. Sedangkan fungsi yang terakhir mengembalikan *DataView* digunakan untuk pembacaan data atau menampilkan data yang diinginkan.



## 4.2 Pengujian Hasil

Sistem informasi kesehatan daerah pada dinas kesehatan merupakan aplikasi untuk pelaporan dan pendataan tentang informasi kesehatan. Pengoperasian aplikasi berupa entri data dan pelaporan, oleh karena itu pengujian hasil difokuskan pada proses entri data, pelaporan, hak akses pengguna terhadap aplikasi dan pengujian hasil pada berbagai *web browser*.

### 4.2.1 Pengujian Halaman Entri Data

Proses entri data di dinas kesehatan meliputi entri data utama dan entri data dari beberapa bidang. Sebagai sampel untuk proses entri data dilakukan pengujian pada entri data indikator derajat kesehatan.

Pengujian dilakukan pada penanganan kesalahan entri dan penyimpanan data.

Lebih jelasnya akan ditunjukkan pada gambar berikut :

No.	Kode	Indikator	Target 2010	Pencapaian 2008
<input type="checkbox"/> [Edit]	1 01	Angka kematian Bayi per 1.000 Kelahiran Hidup	175	75
<input type="checkbox"/> [Edit]	2 03	Angka kematian Ibu Melahirkan per 100.000 Kelahiran Hidup	50	20
<input type="checkbox"/> [Edit]	3 05	Angka Kesakitan malaria per 1.000 penduduk	100	10
<input type="checkbox"/> [Edit]	4 12	Persentase Rumah Sehat	100	50

Gambar 4.1 Kesalahan Entri Data

Pada gambar di atas ditunjukkan entri data yang salah, dimana kolom entri yang seharusnya diisi angka diisi dengan huruf atau teks. Setelah dilakukan penekanan tombol simpan, maka program akan menampilkan pesan bahwa entri gagal seperti yang ditunjukkan pada gambar berikut :



Gambar 4.2 Pesan Kesalahan Entri Data

Proses yang sama akan terus dilakukan selama ada kesalahan entri, begitu juga jika ada kolom entri yang masih kosong, minimal berisi angka nol. Sedangkan jika entri data sesuai dengan tipe data tiap kolom entri, maka data akan langsung disimpan ke dalam tabel. program akan menampilkan pesan bahwa entri berhasil dan menampilkan sebagian data yang telah disimpan ke dalam *gridview*.



Gambar 4.3 Entri Data



Gambar 4.4 Entri Berhasil

## 4.2.2 Pengujian Halaman Laporan

Pengujian laporan atau *Report* dilakukan pada laporan hasil entri program.

Berikut tampilan laporan hasil entri data indikator derajat kesehatan :

INDIKATOR DERAJAT KESEHATAN 2010

Target : 2010  
Pencapaian : 2008

Export Report : MS Word Document (DOC)

INDIKATOR DERAJAT KESEHATAN 2010

1. DERAJAT KESEHATAN  
Indikator derajat kesehatan dan target yang hendak dicapai di tahun 2010

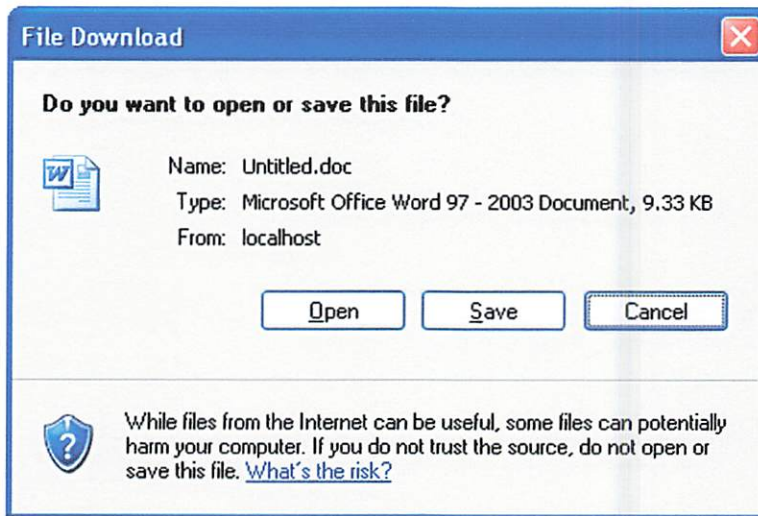
INDIKATOR	TARGET 2010	PENCAPAIAN 2008
MORTALITAS :		
1. Angka kematian Bayi per 1.000 Kelahiran Hidup	175	75
2. Angka kematian Ibu Melahirkan per 100.000 Kelahiran Hidup	5	4
STATUS GIZI :		
3. Persentase Balita Dengan Gizi Buruk	25	15

2. PROSES DAN MASUKAN  
Indikator proses dan masukan dan target yang hendak dicapai di tahun 2010

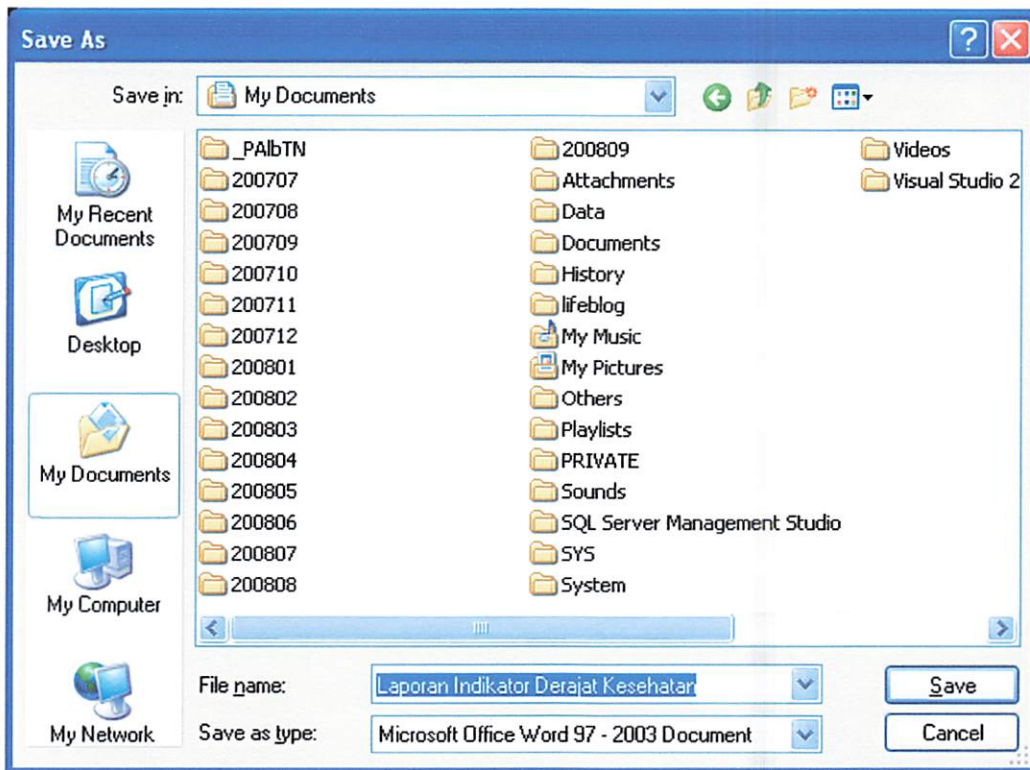
INDIKATOR	TARGET 2010	PENCAPAIAN 2008
PELAYANAN KESEHATAN :		
4. Persentase bayi yang mendapat ASI Eksklusif	500	200

Gambar 4.5 Tampilan Laporan Hasil Entri Data

Pada aplikasi laporan ini, terdapat *parameter* atau inputan untuk menampilkan data yang berbeda, yaitu tahun target, tahun pencapaian, dan jenis indikator. Terdapat dua fasilitas untuk mencetak laporan dan menyimpan laporan dalam format *Word*, *Rich Text Format*, *Excel*, dan *Portable Document / pdf*. Gambar berikut menjelaskan proses ekspor laporan dan mencetak laporan.

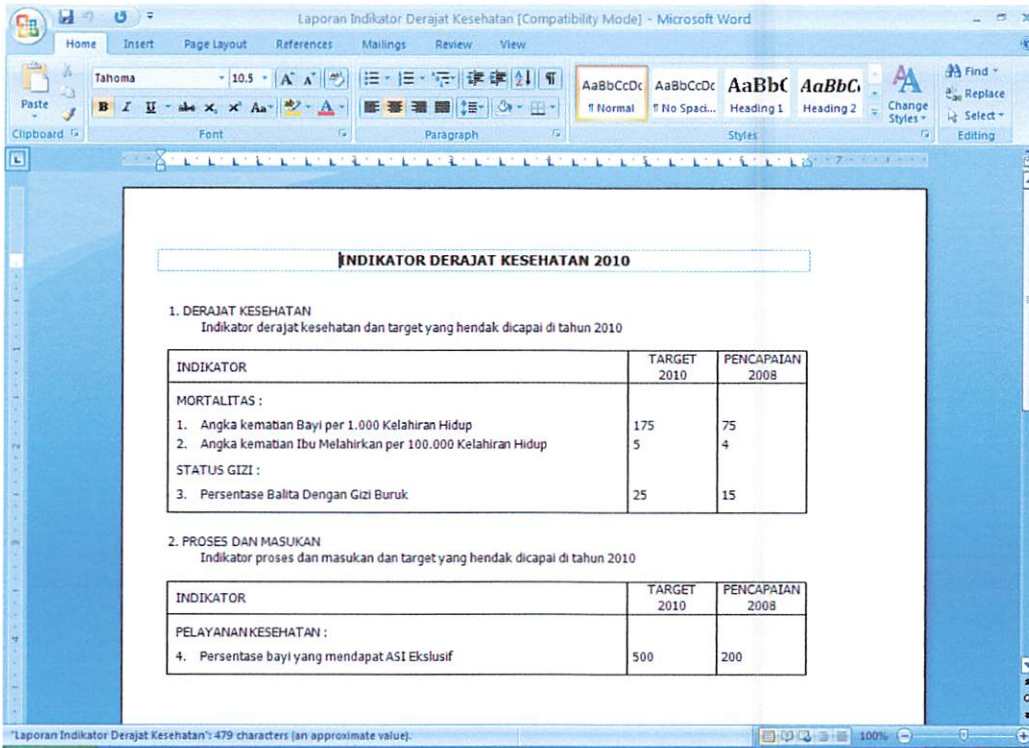


Gambar 4.6 Dialog Ekspor Laporan

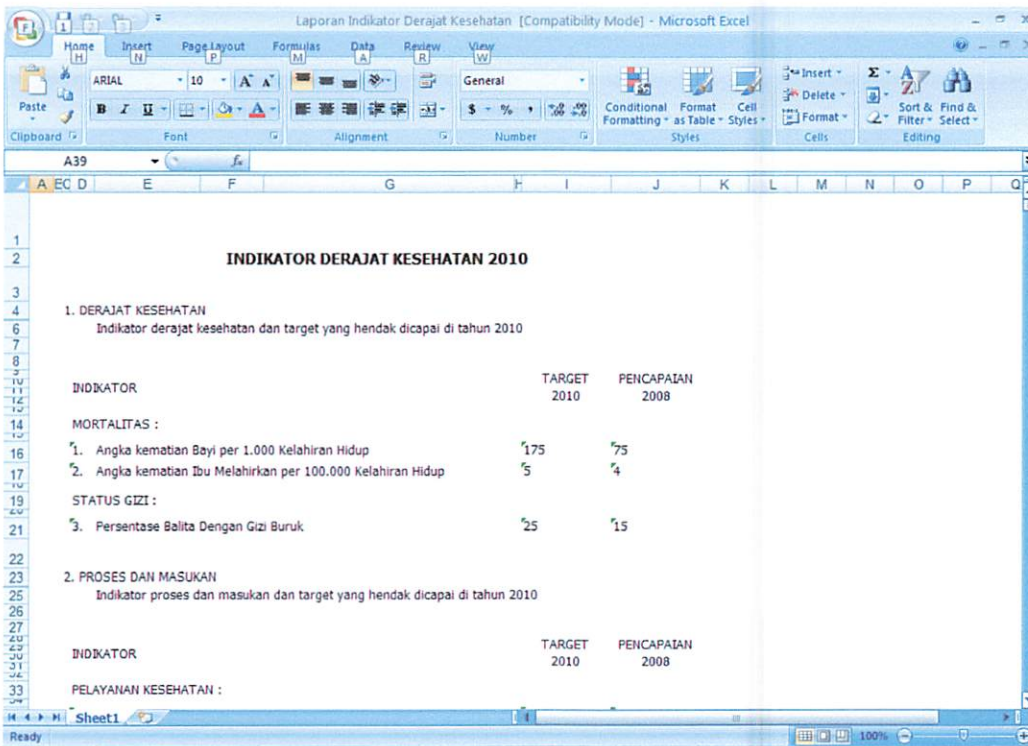


Gambar 4.7 Dialog Simpan File



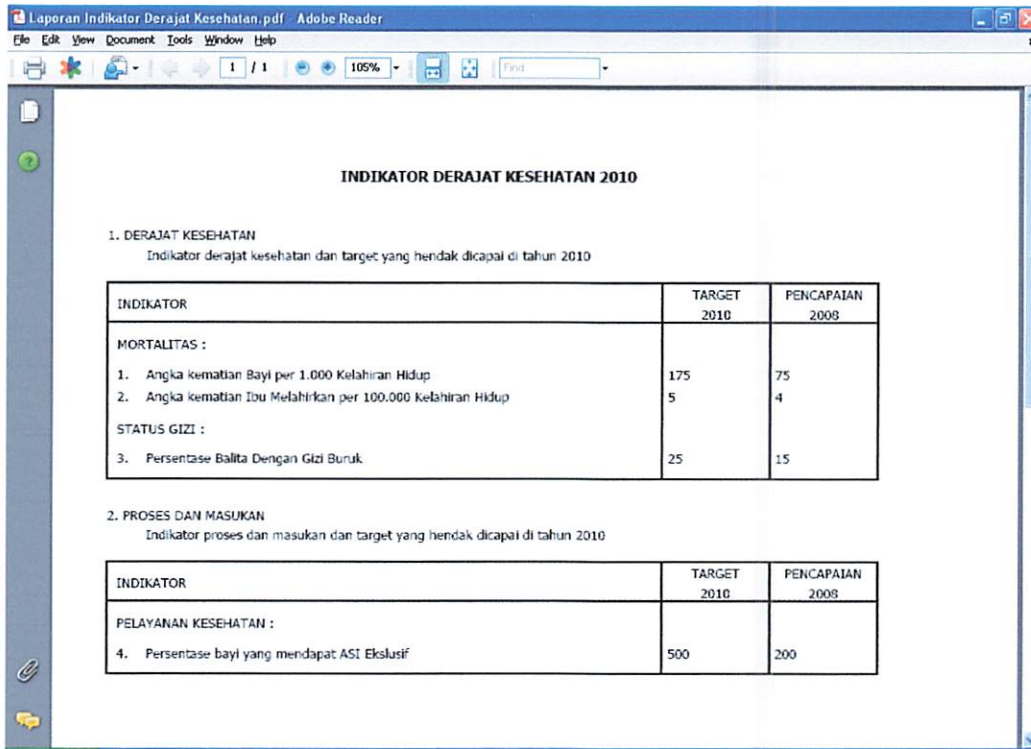


Gambar 4.8 Hasil Ekspor Laporan Dalam Format *Word Document*



Gambar 4.9 Hasil Ekspor Laporan Dalam Format *Excell Spreadsheet*

Untuk mencetak laporan, dibutuhkan perangkat lunak pendukung agar bisa berjalan dengan baik, yaitu Adobe Reader, karena laporan akan dicetak dalam format *portable document (pdf)*. Dengan menekan tombol *Print*, program akan langsung menampilkan laporan dalam format *pdf* dan siap untuk dicetak ke atas kertas.



Gambar 4.10 Mencetak Laporan Dalam Format *Portable Document*

### 4.2.3 Pengujian Hak Akses Pengguna

Setiap pengguna sistem informasi kesehatan daerah ini dibatasi oleh hak akses aplikasi kecuali administrator yang memiliki hak akses penuh termasuk menentukan hak akses pengguna maupun menambah pengguna sistem.

Batasan ini diatur oleh *roleid* pengguna yang ditentukan oleh administrator. Pengujian pertama dilakukan dengan menambah pengguna baru dengan nama “dian” sebagai operator dinas kesehatan dengan nama *roleid* “opr\_dinas” dan unit kerjanya adalah bidang-bidang pada dinas kesehatan.

### Entri User Baru

Username :

Nama :

Password :

Verify Password :

Role ID :  Unit Kerja

Enable :

	NO	USER NAME	NAMA	ROLE ID	ENABLE	KET. SCOPE
<input type="checkbox"/> [EDIT]	1	eeng	eeng	adminpkm	1	001
<input type="checkbox"/> [EDIT]	2	fs	mz fais	admin	1	01
<input type="checkbox"/> [EDIT]	3	quilis	Yan Watequilis S.	admin	1	01

Gambar 4.11 Entri Pengguna Baru

Selanjutnya administrator menentukan hak akses pengguna terhadap aplikasi sistem informasi kesehatan berdasarkan *roleid* pengguna tersebut. Misalkan operator dinas dibatasi hanya bisa mengakses data yang terdapat di dinas kesehatan tetapi tidak memiliki hak akses atas data-data yang terdapat di sumber kesehatan masyarakat demikian pula sebaliknya .

### Entri Role Halaman

Role ID :

Menu :

Halaman :

	NO	ROLE ID	HALAMAN
<input type="checkbox"/> [DELETE]	1	opr_dinas	Entri Data Desa (92210)
<input type="checkbox"/> [DELETE]	2	opr_dinas	Entri Data Kecamatan (34018)
<input type="checkbox"/> [DELETE]	3	opr_dinas	Entri Data Puskesmas (27471)

Gambar 4.12 Menentukan Hak Akses Pengguna

Selanjutnya pengujian dilakukan dengan menggunakan pengguna “dian” sebagai *username* untuk *login* pada aplikasi sistem informasi kesehatan.



Gambar 4.13 Login Operator Dinas Kesehatan



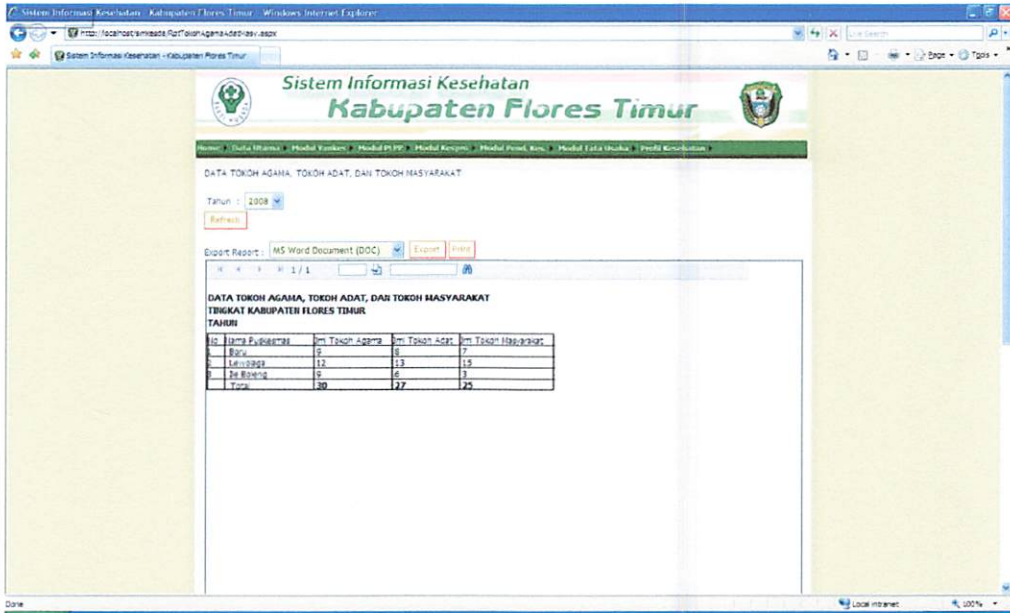
Gambar 4.14 Menu Aplikasi Berdasarkan Hak Akses Pengguna

Dari gambar di atas, setelah pengguna login dan masuk ke halaman utama, menu yang ditampilkan adalah menu sesuai hak akses pengguna yang telah ditentukan pada pengujian sebelumnya.



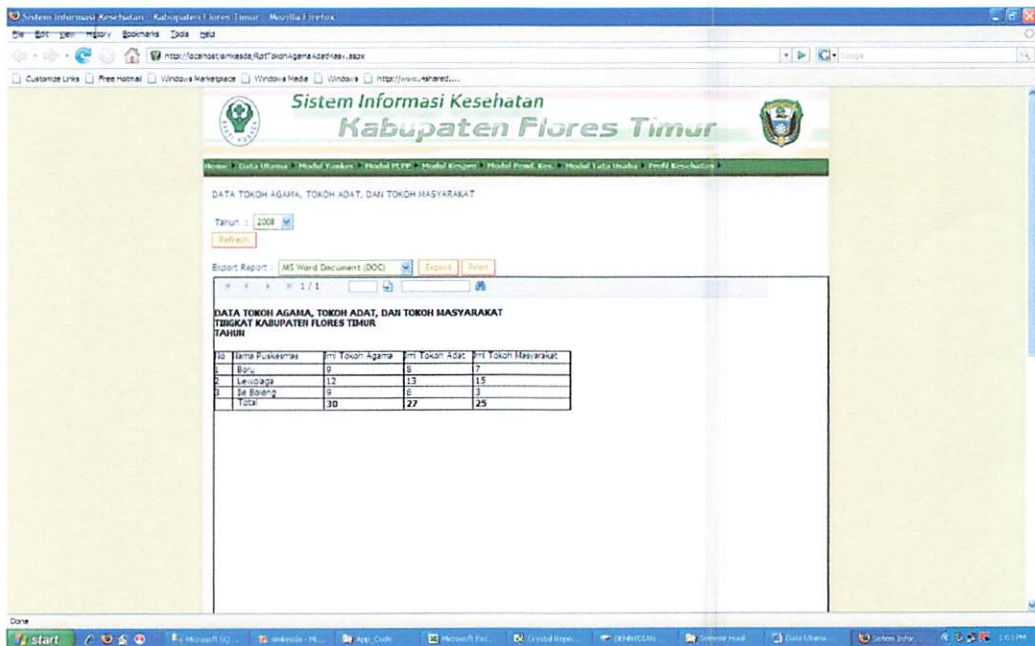
#### 4.2.4 Pengujian pada Beberapa *Web Browser*

- Internet Explorer :



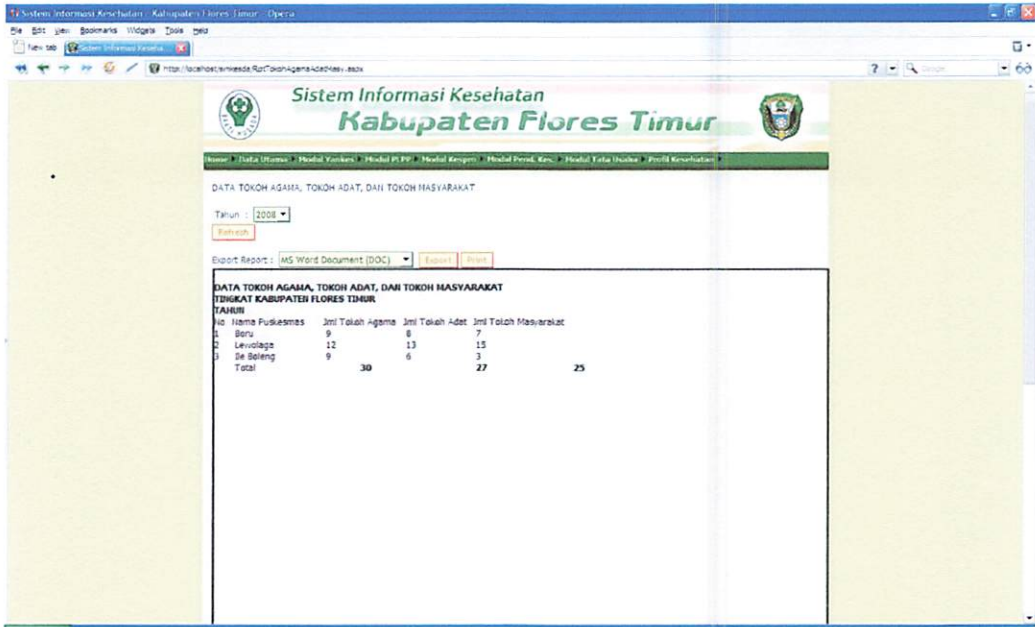
Gambar 4.15 Tampilan aplikasi pada Internet Explorer

- Mozilla Firefox :



Gambar 4.16 Tampilan aplikasi pada Mozilla Firefox

• Opera :



Gambar 4.17 Tampilan aplikasi pada Opera

## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Berdasarkan hasil pengujian sistem yang dilakukan, dapat diambil kesimpulan beberapa hal antara lain :

1. Dengan fasilitas sistem pelaporan pada aplikasi ini, proses pembuatan laporan lebih cepat sehingga dapat mendukung kegiatan pengambilan keputusan.
2. Hasil laporan pada aplikasi bisa disimpan ke dalam format *file* dokumen seperti *Word Document*, *Excel*, *Rich Text Format*, dan *Portable Document*. Sedangkan untuk mencetak laporan dibutuhkan *software* pendukung, yaitu *Acrobat Reader*, karena laporan akan dicetak dalam format *file pdf*.
3. Setelah dilakukan pengujian, aplikasi ini dapat dijalankan pada beberapa browser, seperti Internet Explorer, Mozilla Firefox dan Opera.

#### 5.2 Saran

Sistem informasi kesehatan ini masih dapat dikembangkan lebih jauh lagi karena dalam pembuatannya masih banyak menggunakan batasan karena pertimbangan luasnya sistem dan sumber daya manusia yang akan menggunakan sistem ini. Untuk pengembangan lebih lanjut dari aplikasi ini dapat ditambahkan fasilitas untuk *backup* data, sehingga jika terjadi kerusakan data atau sistem pada server, data dapat dikembalikan seperti sebelumnya.

## DAFTAR PUSTAKA

- [1] Sommerville, Ian. 2003. *Software Engineering : Rekayasa Perangkat Lunak Jilid I*. Jakarta: Erlangga.
- [2] Martin, Joe, & Bret Tomson. 2004. *Belajar Sendiri ASP.Net dalam 24 Jam*. Yogyakarta: Penerbit Andi.
- [3] Djuandi, Feri. 2006. *Jurus Baru Pemrograman SQL Server 2005*. Jakarta: Elex Media Komputindo.
- [4] Kroenke, David M. 2004. *Database Processing : Dasar-Dasar, Desain, dan Implementasi*. Jakarta: Erlangga.
- [5] Watequlis, Yan. 2006. *Diktat Kuliah Sistem Informasi*. Malang.
- [6] Pakereng, Ineke, M.A, & Teguh Wahyono. 2004. *Sistem Basis Data*. Yogyakarta : Graha Ilmu
- [7] H.S, Suryadi D., & Bunawan. 1995. *Pengantar Metodologi Pengembangan Sistem Informasi*. Jakarta: Gunadarma.
- [8] Sutabri, Tata, *Analisa Sistem Informasi*, Andi, Yogyakarta, 2004
- [9] Kadir, Abdul, *Pengenalan Sistem Informasi*, Andi, Yogyakarta, 2003
- [10] Sanjoyo, Raden. 2006. Sistem Informasi Kesehatan. 15 Mei 2008. <http://www.yoyoke.web.ugm.ac.id/sik.pdf>
- [11] Yuhefizar. 2004. *Memahami Konsep Database*. 22 Agustus 2008. <http://media.diknas.go.id/media/document/170.pdf>.
- [12] Dyah. 2007. *Entity Relationship Diagram*. 22 Agustus 2008. <http://blog.its.ac.id/dyah03tc/2007/10/05/modul-1-entity-relationship-diagram-erd/index.htm>
- [13] [www.ilmukomputer.com/umum/choirul-dasar-asp.net.php](http://www.ilmukomputer.com/umum/choirul-dasar-asp.net.php)



# LAMPIRAN



## FORMULIR BIMBINGAN SKRIPSI

Nama : **HERNA DIAN PRIHATINY NAPAN**  
Nim : **04.12.642**  
Masa Bimbingan : **21 JULI 2008 s/d 21 JANUARI 2009**  
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI KESEHATAN DAERAH  
PADA DINAS KESEHATAN DI KABUPATEN FLORES  
TIMUR**

NO.	TANGGAL	URAIAN	PARAF PEMBIMBING
1.	20/08	Bab I & Bab II Bab II baru ditambah	
2.	2/9	Bab I & II	
3.	17/9	Bab I - III	
4.		Bab IV	
5.		Seminar	
6.	24/08	Bab V	
7.	22/08	Ke Pelin	
8.			
9.			
10.			

Malang, 22-9-2008  
Dosen Pembimbing

**Ir. F. Yudi Limpraptono, MT**  
NIP. Y. 103 9500 274



## FORMULIR BIMBINGAN SKRIPSI

Nama : **HERNA DIAN PRIHATINY NAPAN**  
Nim : **04.12.642**  
Masa Bimbingan : **21 JULI 2008 s/d 21 JANUARI 2009**  
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI KESEHATAN DAERAH  
PADA DINAS KESEHATAN DI KABUPATEN FLORES  
TIMUR**

NO.	TANGGAL	URAIAN	PARAF PEMBIMBING
1.	15/8-08	Bab I	
2.	20/8-08	Bab II	
3.	2/9-08	Bab III	
4.	10/9-08	Bab IV	
5.	17/9-08	Seminar	
6.	23/9-08	Bab V	
7.	23/9-08	Acc Akhir	
8.			
9.			
10.			

Malang, 23-9-2008

Dosen Pembimbing

**Yan Wategulis, ST**  
NIP. 132310105

Form S-4B



**INSTITUT TEKNOLOGI NASIONAL MALANG**  
**FAKULTAS TEKNOLOGI INDUSTRI**  
**JURUSAN TEKNIK ELEKTRO S-1**  
Jl. Karanglo KM.2 Malang

### FORMULIR PERBAIKAN SKRIPSI

Nama : Herna Dian Prihatiny Napan  
Nim : 04.12.642  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Komputer dan Informatika  
Masa Bimbingan : 21 Juli 2008 s/d 21 Januari 2009  
Judul Skripsi : **PENGEMBANGAN SISTEM INFORMASI  
KESEHATAN DAERAH DI DINAS KESEHATAN**

No	Tanggal	Uraian	Paraf
1.	26 September 2008	1. Ubah judul dengan mencoret kabupaten Flores Timur dari judul. Tambahkan kabupaten Flores Timur dalam batasan masalah. 2. Jelaskan secara detil tentang sistem saat ini 3. Buat DAD Level 2	

#### Disetujui

Penguji I

**DR. Cahyo Crysdian, MSc**  
NIP P. 1030400412

Penguji II

**Sotyonadi, ST, MSc**  
NIP Y. 1039700309

#### Mengetahui

Dosen Pembimbing I

**Ir. F. Yudi Limpraptono, MT**  
NIP .Y. 1039500274

Dosen Pembimbing II

**Yan Wategulis S, ST**  
NIP . 132310105

## MainLibrary.vb

```
Imports Microsoft.VisualBasic

Imports System
Imports System.Data
Imports System.Data.SqlClient

Public Class MainLibrary
    Dim dbLib As New
    DBConnection

    Dim dbConn As SqlConnection
    = dbLib.GetSQLConn
    Dim dbCmd As SqlCommand =
    dbConn.CreateCommand
    Dim drVar As SqlDataReader

    Private Const strVal As String
    = "CODE"
    Private Const strText As
    String = "VALUE"
    Public IPClient As String =
    HttpContext.Current.Request.Use
    rHostAddress
    Dim CPH As String =
    "ContentPlaceholder1"

    Public Function inputData(ByVal
    tabel As String, ByVal field As
    String, ByVal value As String)
    As Boolean
        field &= ", USER_ENTRI,
    IP_ENTRI, TS_ENTRI"
        value &= ", ', '" &
    IPClient & ", '" &
    Format(Now(), "M/d/yyyy
    hh:mm:ss") & ""

        Dim q As String =
    "INSERT INTO " & tabel & "(" &
    field & ")"
        q &= "VALUES(" & value
    & ")"

        If dbConn.State =
    ConnectionState.Closed Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        Try

        dbCmd.ExecuteNonQuery()
        Return True
        Catch ex As Exception
        End Try
    End Function

```

```
        Return False
    End Try
    dbConn.Close()

    End Function

    Public Function
    UpdateData(ByVal tabel As
    String, ByVal fieldUpdate As
    String, ByVal kondisi As
    String) As Boolean
        Dim q As String =
    "UPDATE " & tabel & " SET " &
    fieldUpdate & " WHERE " &
    kondisi

        If dbConn.State =
    ConnectionState.Closed Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        Try

        dbCmd.ExecuteNonQuery()
        Return True
        Catch ex As Exception
        Return False
        End Try
        dbConn.Close()

    End Function

    Public Function
    DeleteData(ByVal tabel As
    String, ByVal condition As
    String) As Boolean
        Dim q As String =
    "DELETE FROM " & tabel &
    " WHERE (" & condition & ")"

        If dbConn.State =
    ConnectionState.Closed Then
            dbConn.Open()
        End If

        dbCmd.CommandText = q
        Try

        dbCmd.ExecuteNonQuery()
        Return True
        Catch ex As
    SqlException
        Return False
        End Try
        dbConn.Close()

    End Function

    Public Sub comboBind(ByVal
    combo As
```

```

System.Web.UI.WebControls.DropDownList, ByVal data As
DataView)
    combo.DataSource() =
data
    combo.DataTextField =
strText
    combo.DataValueField =
strVal
    combo.DataBind()
End Sub

Public Sub
showDataGrid(ByVal grid As
System.Web.UI.WebControls.GridView, ByVal data As DataView)
    grid.DataSource = data
    grid.DataBind()
End Sub

Public Sub
showDataGrid(ByVal grid As
System.Web.UI.WebControls.GridView, ByVal data As DataView,
ByVal index As Integer)
    grid.PageIndex = index
    grid.DataSource = data
    grid.DataBind()
End Sub

Public Sub CheckGrid(ByVal grid As
System.Web.UI.WebControls.GridView, ByVal checkState As
Boolean)
    For Each row As
GridViewRow In grid.Rows
        Dim cb As CheckBox
= row.FindControl("cbselect")
        If cb IsNot Nothing
Then
            cb.Checked =
checkState
        End If
    Next
End Sub

Public Function
getDecimal(ByVal x As String)
As String
    x =
x.Replace(",", ".")
    Return x
End Function

Public Function
cekData(ByVal tabel As String,

```

```

ByVal field As String, ByVal
seek As String) As Boolean
    Dim q As String =
"SELECT " & field & " FROM " &
tabel & " WHERE " & seek
    Dim hasil As Boolean =
False

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()

    If drVar.Read Then
        hasil = True
    End If
    drVar.Close()
    dbConn.Close()
    Return hasil

End Function

Public Function
getCodeData(ByVal tabel As
String, ByVal field As String,
ByVal kondisi As String) As
String
    Dim q As String =
"SELECT " & field & " FROM " &
tabel & " WHERE CODE = '" &
kondisi & "'"
    Dim hasil As String =
""

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()

    If drVar.Read Then
        hasil =
drVar.GetString(0)
    End If
    drVar.Close()
    dbConn.Close()
    Return hasil
End Function

Public Function getDSTgl() As
DataView
    Dim dt As DataTable =
New DataTable

```

```

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

        Dim i As Integer

        For i = 1 To 31
            Dim dr As DataRow =
dt.NewRow()
            dr(0) =
i.ToString()
            dr(1) =
i.ToString()
            dt.Rows.Add(dr)
        Next
        Dim dv As DataView =
New DataView(dt)
        Return dv
    End Function

    Public Function
getDSBulan() As DataView
        Dim dt As New DataTable
        Dim dr As DataRow
        dt.Columns.Add(New
DataColumn("CODE",
GetType(String)))
        dt.Columns.Add(New
DataColumn("VALUE",
GetType(String)))

        Dim q As String =
"SELECT * FROM D_BULAN ORDER BY
CODE"

        If (dbConn.State =
ConnectionState.Closed) Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

        While drVar.Read()
            dr = dt.NewRow()
            dr(0) =
drVar.GetValue(0).ToString
            dr(1) =
drVar.GetString(1).Trim
            dt.Rows.Add(dr)
        End While
        drVar.Close()
        dbConn.Close()
    End Function

```

```

        Dim dv As New
DataView(dt)
        Return dv
    End Function

    Public Function
getDSTriwulan() As DataView
        Dim dt As New DataTable
        Dim dr As DataRow
        dt.Columns.Add(New
DataColumn("CODE",
GetType(String)))
        dt.Columns.Add(New
DataColumn("VALUE",
GetType(String)))

        Dim q As String =
"SELECT * FROM D_TRIWULAN ORDER
BY CODE"

        If (dbConn.State =
ConnectionState.Closed) Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

        While drVar.Read()
            dr = dt.NewRow()
            dr(0) =
drVar.GetValue(0).ToString
            dr(1) =
drVar.GetString(1).Trim
            dt.Rows.Add(dr)
        End While
        drVar.Close()
        dbConn.Close()

        Dim dv As New
DataView(dt)
        Return dv
    End Function

    Public Function
getDSMinggu() As DataView
        Dim dt As DataTable =
New DataTable

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

        Dim i As Integer

```

```

        For i = 1 To 52
            Dim dr As DataRow =
dt.NewRow()
            dr(0) =
i.ToString()
            dr(1) =
i.ToString()
            dt.Rows.Add(dr)
        Next
        Dim dv As DataView =
New DataView(dt)
        Return dv
    End Function

```

```

    Public Function
getDSTahun(ByVal awal As
Integer, ByVal akhir As
Integer) As ICollection
        Dim dt As DataTable =
New DataTable

```

```

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

```

```

        Dim i As Integer

```

```

        For i = awal To akhir
            Dim dr As DataRow =
dt.NewRow()
            dr(0) =
i.ToString()
            dr(1) =
i.ToString()
            dt.Rows.Add(dr)
        Next
        Dim dv As DataView =
New DataView(dt)
        Return dv
    End Function

```

```

    Public Function
getData(ByVal tabel As String)
As DataView
        Dim dt As New DataTable
        Dim dr As DataRow
        dt.Columns.Add(New
DataColumn("CODE",
GetType(String)))
        dt.Columns.Add(New
DataColumn("VALUE",
GetType(String)))

```

```

        Dim q As String =
"SELECT CODE, NAMA FROM " &
tabel & " ORDER BY CODE"

```

```

        If (dbConn.State =
ConnectionState.Closed) Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

```

```

        While drVar.Read()
            dr = dt.NewRow()
            dr(0) =
drVar.GetString(0)
            dr(1) =
drVar.GetString(1)
            dt.Rows.Add(dr)
        End While
        drVar.Close()
        dbConn.Close()

```

```

        Dim dv As New
DataView(dt)
        Return dv

```

```

    End Function

```

```

    Public Function
getData(ByVal tabel As String,
ByVal order As String) As
DataView
        Dim dt As New DataTable
        Dim dr As DataRow
        dt.Columns.Add(New
DataColumn("CODE",
GetType(String)))
        dt.Columns.Add(New
DataColumn("VALUE",
GetType(String)))

```

```

        Dim q As String =
"SELECT CODE, NAMA FROM " &
tabel & " ORDER BY " & order

```

```

        If (dbConn.State =
ConnectionState.Closed) Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

```

```

        While drVar.Read()
            dr = dt.NewRow()
            dr(0) =
drVar.GetString(0)

```



```

        dr(1) =
drVar.GetString(1)
        dt.Rows.Add(dr)
        End While
        drVar.Close()
        dbConn.Close()

        Dim dv As New
DataView(dt)
        Return dv

    End Function

    Public Function
getData(ByVal tabel As String,
ByVal filter As String, ByVal
kondisi As String) As DataView
        Dim dt As New DataTable
        Dim dr As DataRow
        dt.Columns.Add(New
DataColumn("CODE",
GetType(String)))
        dt.Columns.Add(New
DataColumn("VALUE",
GetType(String)))

        Dim q As String =
"SELECT CODE, NAMA FROM " &
tabel & " WHERE " & filter & "
= '" & kondisi & "' ORDER BY
CODE"

        If (dbConn.State =
ConnectionState.Closed) Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()
        While drVar.Read()
            dr = dt.NewRow()
            dr(0) =
drVar.GetString(0)
            dr(1) =
drVar.GetString(1)
            dt.Rows.Add(dr)
        End While
        drVar.Close()
        dbConn.Close()

        Dim dv As New
DataView(dt)
        Return dv

    End Function

    Public Function
getDSBlnDistinct(ByVal tabel As
String) As DataView

```

```

        Dim dt As DataTable =
New DataTable

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

        Dim q As String =
"SELECT DISTINCT T.BULAN,
B.BULAN FROM " & tabel & " AS T
INNER JOIN "
        q &= "D_BULAN AS B ON
T.BULAN = B.CODE ORDER BY
T.BULAN"

        dbConn.Open()
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()
        If drVar.HasRows Then
            While drVar.Read()
                Dim dr As
DataRow = dt.NewRow()
                dr(0) =
drVar.GetInt16(0).ToString
                dr(1) =
drVar.GetString(1)
                dt.Rows.Add(dr)
            End While
        Else
            Dim dr As DataRow =
dt.NewRow()
            dr(0) = "00"
            dr(1) = "kosong"
            dt.Rows.Add(dr)
        End If
        drVar.Close()
        dbConn.Close()

        Dim dv As DataView =
New DataView(dt)
        Return dv

    End Function

    Public Function
getDSThnDistinct(ByVal tabel As
String) As DataView

        Dim dt As DataTable =
New DataTable

```

```

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

        Dim q As String =
"SELECT DISTINCT TAHUN FROM " &
tabel & " ORDER BY TAHUN"

        dbConn.Open()
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

        If drVar.HasRows Then
            While drVar.Read()
                Dim dr As
DataRow = dt.NewRow()
                dr(0) =
drVar.GetString(0)
                dr(1) =
drVar.GetString(0)
                dt.Rows.Add(dr)
            End While
        Else
            Dim dr As DataRow =
dt.NewRow()
            dr(0) = "00"
            dr(1) = "kosong"
            dt.Rows.Add(dr)
        End If
        drVar.Close()
        dbConn.Close()
        Dim dv As DataView =
New DataView(dt)
        Return dv

    End Function

    Public Function
getDSPuskDistinct(ByVal tabel
As String, ByVal kondisi As
String) As DataView
        Dim dt As DataTable =
New DataTable

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

        Dim q As String =
"SELECT DISTINCT T.PUSKESMAS,
P.NAMA FROM " & tabel & " AS T
INNER JOIN "

```

```

P.NAMA FROM " & tabel & " AS T
INNER JOIN "
        q &= "T_PUSKESMAS AS P
ON T.PUSKESMAS = P.CODE ORDER
BY T.PUSKESMAS"

        dbConn.Open()
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

        If drVar.HasRows Then
            While drVar.Read()
                Dim dr As
DataRow = dt.NewRow()
                dr(0) =
drVar.GetString(0)
                dr(1) =
drVar.GetString(1)
                dt.Rows.Add(dr)
            End While
        Else
            Dim dr As DataRow =
dt.NewRow()
            dr(0) = "00"
            dr(1) = "kosong"
            dt.Rows.Add(dr)
        End If

        drVar.Close()
        dbConn.Close()

        Dim dv As DataView =
New DataView(dt)
        Return dv

    End Function

    Public Function
getDSPuskDistinct(ByVal tabel
As String, ByVal kondisi As
String) As DataView
        Dim dt As DataTable =
New DataTable

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

        Dim q As String =
"SELECT DISTINCT T.PUSKESMAS,
P.NAMA FROM " & tabel & " AS T
INNER JOIN "

```

```

        q &= "T_PUSKESMAS AS P
ON T.PUSKESMAS = P.CODE WHERE "
& kondisi & " ORDER BY
T.PUSKESMAS"

```

```

        dbConn.Open()
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

        If drVar.HasRows Then
            While drVar.Read()
                Dim dr As
DataRow = dt.NewRow()
                dr(0) =
drVar.GetString(0)
                dr(1) =
drVar.GetString(1)
                dt.Rows.Add(dr)
            End While
        Else
            Dim dr As DataRow =
dt.NewRow()
                dr(0) = "00"
                dr(1) = "kosong"
                dt.Rows.Add(dr)
        End If
        drVar.Close()
        dbConn.Close()

```

```

        Dim dv As DataView =
New DataView(dt)
        Return dv

```

```

    End Function

```

```

    Public Function
getDSRSDistinct(ByVal tabel As
String) As DataView

```

```

        Dim dt As DataTable =
New DataTable

```

```

        dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
        dt.Columns.Add(New
DataColumn(strText,
GetType(String))

```

```

        Dim q As String =
"SELECT DISTINCT T.RS, R.NAMA
FROM " & tabel & " AS T INNER
JOIN "

```

```

        q &= "T_RUMAH_SAKIT AS
R ON T.RS = R.CODE ORDER BY
T.RS"

```

```

        dbConn.Open()
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()

```

```

        If drVar.HasRows Then
            While drVar.Read()
                Dim dr As
DataRow = dt.NewRow()
                dr(0) =
drVar.GetString(0)
                dr(1) =
drVar.GetString(1)
                dt.Rows.Add(dr)
            End While
        Else
            Dim dr As DataRow =
dt.NewRow()
                dr(0) = "00"
                dr(1) = "kosong"
                dt.Rows.Add(dr)
        End If

```

```

        drVar.Close()
        dbConn.Close()

```

```

        Dim dv As DataView =
New DataView(dt)
        Return dv

```

```

    End Function

```

```

    Public Function
getDSCakupan(ByVal tabel As
String, ByVal kondisi As
String) As Integer
        Dim hasil As Integer =
0
        Dim q As String =
"SELECT SUM(CAKUPAN) AS CAKUPAN
FROM " & tabel & " WHERE " &
kondisi

```

```

        If (dbConn.State =
ConnectionState.Closed) Then
            dbConn.Open()
        End If
        dbCmd.CommandText = q
        drVar =
dbCmd.ExecuteReader()
        While drVar.Read()
            hasil =
drVar.GetInt32(0)
        End While
        drVar.Close()
        dbConn.Close()

```

```

        Return hasil

```

```

End Function

Public Function
getDSKegLB(ByVal tabel As
String, ByVal kondisi As
String) As DataView
    Dim dt As DataTable =
New DataTable
    Dim dr As DataRow
    dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
    dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

    Dim q As String =
"SELECT DISTINCT T.KEGIATAN,
I.NAMA FROM " & tabel & " AS T
INNER JOIN D_ITEM_KEGIATAN "
q &= "AS I ON
T.KEGIATAN = I.CODE WHERE " &
kondisi & " ORDER BY
T.KEGIATAN"

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()
    While drVar.Read()
        dr = dt.NewRow
        dr(0) =
drVar.GetString(0)
        dr(1) =
drVar.GetString(1)
        dt.Rows.Add(dr)
    End While
    drVar.Close()
    dbConn.Close()

    Dim dv As New
DataView(dt)
    Return dv

End Function

Public Function
getDSPangkat() As DataView
    Dim dt As DataTable =
New DataTable
    Dim dr As DataRow
    dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))

```

```

dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

    Dim q As String =
"SELECT CODE, PANGKAT FROM
D_PANGKAT ORDER BY CODE"

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()
    While drVar.Read()
        dr = dt.NewRow
        dr(0) =
drVar.GetString(0)
        dr(1) =
drVar.GetString(1)
        dt.Rows.Add(dr)
    End While
    drVar.Close()
    dbConn.Close()

    Dim dv As New
DataView(dt)
    Return dv

End Function

Public Function
getKecPuskesmas(ByVal puskes As
String) As String
    Dim q As String =
"SELECT KECAMATAN FROM
T_PUSKESMAS WHERE CODE = '" &
puskes & "'"

    If dbConn.State =
ConnectionState.Closed Then
        dbConn.Open()
    End If
    Dim hasil = ""
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()
    If drVar.Read Then
        hasil =
drVar.GetString(0).Trim
    End If
    drVar.Close()
    dbConn.Close()

    Return hasil

End Function

```

```

Function getNoUrut(ByVal
tabel As String, ByVal kondisi
As String) As Integer
    Dim nmr As Integer

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText =
"SELECT NO_URUT FROM " & tabel
& " WHERE " & kondisi & " ORDER
BY NO_URUT DESC "
    drVar =
dbCmd.ExecuteReader()

    If drVar.Read() Then
        nmr =
drVar.GetValue(0)
    End If
    drVar.Close()
    dbConn.Close()
    nmr += 1
    Return nmr

End Function

```

```

Public Function
getJumlah(ByVal field As
String, ByVal thn As String,
ByVal puskes As String) As
Integer
    Dim hasil As Integer =
0

    Dim q As String =
"SELECT SUM(" & field & ")AS
JML FROM T_PENANGANAN_KELUHAN "
    q &= "WHERE TAHUN = '"
& thn & "' AND PUSKESMAS = '" &
puskes & "' GROUP BY TAHUN,
PUSKESMAS"

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()
    While drVar.Read()
        hasil =
drVar.GetInt32(0)
    End While
    drVar.Close()
    dbConn.Close()

    Return hasil

End Function

```

```

Public Function
getJumlah(ByVal field As
String, ByVal thn As String,
ByVal puskes As String, ByVal
keluhan As String) As Integer
    Dim hasil As Integer =
0

    Dim q As String =
"SELECT SUM(" & field & ")AS
JML FROM T_PENANGANAN_KELUHAN "
    q &= "WHERE TAHUN = '"
& thn & "' AND PUSKESMAS = '" &
puskes & "' AND ASAL_KELUHAN =
'"

    q &= keluhan & "' GROUP
BY TAHUN,PUSKESMAS"

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()
    While drVar.Read()
        hasil =
drVar.GetInt32(0)
    End While
    drVar.Close()
    dbConn.Close()

    Return hasil

End Function

```

```

Public Function
getDSPengelolaPestisida() As
DataView
    Dim dt As DataTable =
New DataTable
    Dim dr As DataRow
    dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
    dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

    Dim q As String =
"SELECT NO_REG, NAMA FROM
D_PERUSAHAAN_PENGELOLA_PEST
ORDER BY NO_REG"

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q

```

```

        drVar =
dbCmd.ExecuteReader()
    While drVar.Read()
        dr = dt.NewRow
        dr(0) =
drVar.GetString(0)
        dr(1) =
drVar.GetString(1)
        dt.Rows.Add(dr)
    End While
    drVar.Close()
    dbConn.Close()

    Dim dv As New
DataView(dt)
    Return dv

End Function

Public Function
getDSUnitKerjaNotIn(ByVal x As
String) As DataView
    Dim dt As DataTable =
New DataTable
    Dim dr As DataRow
    dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
    dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

    Dim q As String =
"SELECT CODE, NAMA FROM
D_UNIT_KERJA WHERE CODE NOT IN
(" & x & ") ORDER BY CODE"

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()
    While drVar.Read()
        dr = dt.NewRow
        dr(0) =
drVar.GetString(0)
        dr(1) =
drVar.GetString(1)
        dt.Rows.Add(dr)
    End While
    drVar.Close()
    dbConn.Close()

    Dim dv As New
DataView(dt)
    Return dv

```

```

End Function

Public Function
getDSIndikatorKeg() As DataView
    Dim dbconn2 As
SqlConnection =
dbLib.getSQLConn
    Dim dbconn3 As
SqlConnection =
dbLib.getSQLConn
    Dim dbconn4 As
SqlConnection =
dbLib.getSQLConn
    Dim dbcmd2 As
SqlCommand =
dbconn2.CreateCommand
    Dim dbcmd3 As
SqlCommand =
dbconn3.CreateCommand
    Dim dbcmd4 As
SqlCommand =
dbconn4.CreateCommand
    Dim drvar2 As
SqlDataReader
    Dim drvar3 As
SqlDataReader
    Dim drvar4 As
SqlDataReader

    Dim dt As DataTable =
New DataTable
    Dim dr As DataRow
    dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
    dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

    Dim q As String =
"SELECT CODE, NAMA FROM
D_INDIKATOR_KEGIATAN WHERE
PARENT = '000'"

    If (dbConn.State =
ConnectionState.Closed) Then
        dbConn.Open()
    End If
    dbCmd.CommandText = q
    drVar =
dbCmd.ExecuteReader()
    While drVar.Read()
        dr = dt.NewRow
        dr(0) =
drVar.GetString(0)
        dr(1) =
drVar.GetString(1)
        dt.Rows.Add(dr)

```

```

        Dim q2 As String =
"SELECT CODE, NAMA FROM
D_INDIKATOR_KEGIATAN WHERE
PARENT = '" &
drVar.GetString(0) & "'"
        If (dbconn2.State =
ConnectionState.Closed) Then
            dbconn2.Open()
        End If
        dbcmd2.CommandText
= q2
        drvar2 =
dbcmd2.ExecuteReader
        While drvar2.Read
            dr = dt.NewRow
            dr(0) =
drvar2.GetString(0)
            dr(1) = "--" &
drvar2.GetString(1)
            dt.Rows.Add(dr)
            Dim q3 As
String = "SELECT CODE, NAMA
FROM D_INDIKATOR_KEGIATAN WHERE
PARENT = '" &
drvar2.GetString(0) & "'"
            If
(dbconn3.State =
ConnectionState.Closed) Then
                dbconn3.Open()
            End If

            dbcmd3.CommandText = q3
            drvar3 =
dbcmd3.ExecuteReader
            While
drvar3.Read
                dr =
dt.NewRow
                dr(0) =
drvar3.GetString(0)
                dr(1) = "--
--" & drvar3.GetString(1)
                dt.Rows.Add(dr)
                Dim q4 As
String = "SELECT CODE, NAMA
FROM D_INDIKATOR_KEGIATAN WHERE
PARENT = '" &
drvar3.GetString(0) & "'"
                If
(dbconn4.State =
ConnectionState.Closed) Then
                    dbconn4.Open()
                End If

                dbcmd4.CommandText = q4

```

```

drvar4 =
dbcmd4.ExecuteReader
        While
drvar4.Read
            dr =
dt.NewRow
            dr(0) =
drvar4.GetString(0)
            dr(1) =
"-----" & drvar4.GetString(1)
            dt.Rows.Add(dr)
        End While
        drvar4.Close()
        dbconn4.Close()
        End While
        drvar3.Close()
        dbconn3.Close()
        End While
        drvar2.Close()
        dbconn2.Close()
        End While
        drVar.Close()
        dbConn.Close()

        Dim dv As New
DataView(dt)
        Return dv
    End Function

    Public Function
getDSData(ByVal tabel As
String, ByVal order As String)
As DataView
        Return getData(tabel,
order)
    End Function

    Public Function
getDSData(ByVal tabel As
String, ByVal filter As String,
ByVal kondisi As String) As
DataView
        Return getData(tabel,
filter, kondisi)
    End Function

    Public Function getDSDesa()
As ICollection
        Return
getDSData("D_DESA")
    End Function

    Public Function
getDSDesa(ByVal kondisi As
String) As ICollection

```

```
Return
getDSData("D_DESA",
"KECAMATAN", kondisi)
End Function
```

```
Public Function
getDSKecamatan() As ICollection
Return
getDSData("D_KECAMATAN")
End Function
```

```
Public Function
getDSDinas() As ICollection
Return
getDSData("D_DATA_DINAS")
End Function
```

```
Public Function
getDSJnsKontrasepsi() As
ICollection
Return
getDSData("D_JNS_KONTRASEPSI")
End Function
```

```
Public Function
getDSKeluhan() As ICollection
Return
getDSData("D_ASAL_KELUHAN")
End Function
```

```
Public Function
getDSKontrasepsi(ByVal kondisi
As String) As ICollection
Return
getDSData("D_KONTRASEPSI",
"JENIS", kondisi)
End Function
```

```
Public Function
getDSKegiatan(ByVal kondisi As
String) As ICollection
Return
getDSData("D_KEGIATAN",
"JENIS", kondisi)
End Function
```

```
Public Function
getDSSubKegiatan(ByVal kondisi
As String) As ICollection
Return
getDSData("D_SUB_KEGIATAN",
"KEGIATAN", kondisi)
End Function
```

```
Public Function
getDSItemKegiatan(ByVal kondisi
As String) As ICollection
```

```
Return
getDSData("D_ITEM_KEGIATAN",
"SUB_KEGIATAN", kondisi)
End Function
```

```
Public Function
getDSSatuan() As ICollection
Return
getDSData("D_SATUAN")
End Function
```

```
Public Function
getDSJnsKegiatan() As
ICollection
Return
getDSData("D_JNS_KEGIATAN")
End Function
```

```
Public Function
getDSPengelolaObat() As
ICollection
Return
getDSData("D_PENGELOLA_OBAT")
End Function
```

```
Public Function getDSObat()
As ICollection
Return
getDSData("D_OBAT")
End Function
```

```
Public Function
getDSObat(ByVal order As
String) As ICollection
Return
getDSData("D_OBAT", order)
End Function
```

```
Public Function
getDSKemasanObat() As
ICollection
Return
getDSData("D_KEMASAN_OBAT")
End Function
```

```
Public Function
getDSJnsPenyakit() As
ICollection
Return
getDSData("D_JNS_PENYAKIT")
End Function
```

```
Public Function
getDSPelayanan() As ICollection
Return
getDSData("D_PELAYANAN")
End Function
```



```
Public Function
getDSPenyakit() As ICollection
Return
getDSData("D_PENYAKIT",
"JENIS")
End Function
```

```
Public Function
getDSPenyakit(ByVal kondisi As
String) As ICollection
Return
getDSData("D_PENYAKIT",
"JENIS", kondisi)
End Function
```

```
Public Function getDSRS()
As ICollection
Return
getDSData("T_RUMAH_SAKIT")
End Function
```

```
Public Function getDSTTU()
As ICollection
Return
getDSData("D_TEMPAT_UMUM")
End Function
```

```
Public Function getDSTPM()
As ICollection
Return
getDSData("D_TPM")
End Function
```

```
Public Function
getDSVaksin() As ICollection
Return
getDSData("D_VAKSIN", "NAMA")
End Function
```

```
Public Function
getDSJnsKelamin() As
ICollection
Return
getDSData("D_JNS_KELAMIN")
End Function
```

```
Public Function
getDSJnsSaranaTU() As
ICollection
Return
getDSData("D_JNS_SARANA_TTU")
End Function
```

```
Public Function
getDSSaranaTU() As ICollection
Return
getDSData("D_SARANA_TTU")
End Function
```

```
Public Function
getDSSaranaTU(ByVal kondisi As
String) As ICollection
Return
getDSData("D_SARANA_TTU",
"JENIS", kondisi)
End Function
```

```
Public Function
getDSHslPeriksa() As
ICollection
Return
getDSData("D_HSL_PERIKSA",
"CODE ASC")
End Function
```

```
Public Function
getDSUnitPelkes() As
ICollection
Return
getDSData("D_UNIT_PELKES")
End Function
```

```
Public Function
getDSJnsPerusahaan() As
ICollection
Return
getDSData("D_JNS_PERUSAHAAN")
End Function
```

```
Public Function
getDSPestisida() As ICollection
Return
getDSData("D_PESTISIDA")
End Function
```

```
Public Function
getDSJnsIndikator() As
ICollection
Return
getDSData("D_JNS_INDIKATOR")
End Function
```

```
Public Function
getDSIndikator(ByVal kondisi As
String) As ICollection
Return
getDSData("D_INDIKATOR",
"JENIS", KONDISI)
End Function
```

```
Public Function
getDSSubIndikator(ByVal kondisi
As String) As ICollection
Return
getDSData("D_SUB_INDIKATOR",
"INDIKATOR", kondisi)
End Function
```

```

Public Function
getDSKlpSekolah() As
ICollection
Return
getDSData("D_KELOMPOK_SEKOLAH")
End Function

```

```

Public Function
getDSJnsSekolah(ByVal kondisi
As String) As ICollection
Return
getDSData("D_JNS_SEKOLAH",
"KELOMPOK", kondisi)
End Function

```

```

Public Function
getDSYesNo() As ICollection
Return
getData("D_JAWAB")
End Function

```

```

Public Function
getDSUmumKhusus() As
ICollection
Return
getData("D_JNS_PELAYANAN")
End Function

```

```

Public Function
getDSSarKes() As ICollection
Return
getData("D_SARANA_KESEHATAN")
End Function

```

```

Public Function
getDSUnitKerja() As ICollection
Return
getData("D_UNIT_KERJA")
End Function

```

```

Public Function
getDSSubUnitKerja(ByVal kondisi
As String) As ICollection
Return
getData("D_SUB_UNIT_KERJA",
"UNIT_KERJA", kondisi)
End Function

```

```

Public Function
getDSFasKes() As ICollection
Return
getData("D_FASILITAS_KESEHATAN"
)
End Function

```

```

Public Function
getDSSarkes(ByVal kondisi As
String) As ICollection

```

```

Return
getDSData("D_SARANA_KESEHATAN",
"CODE", kondisi)
End Function

```

```

Public Function
getDSSubSarkes(ByVal kondisi As
String) As ICollection
Return
getDSData("D_SUB_SARANA_KESEHAT
AN", "SARKES", kondisi)
End Function

```

```

Public Function
getDSProgKeg() As ICollection
Return
getData("D_PROGRAM_KEGIATAN")
End Function

```

```

Public Function
getDSSubProgKeg(ByVal kondisi
As String) As ICollection
Return
getDSData("D_SUB_PROGRAM_KEGIAT
AN", "PROGRAM", kondisi)
End Function

```

```

Public Function
getDSKejadianLuarBiasa() As
ICollection
Return
getData("D_KEJADIAN_LUAR_BIASA"
)
End Function

```

```

Public Function getDSUTD()
As ICollection
Return
getData("D_UNIT_TRANSFUSI_DARAH
")
End Function

```

```

Public Function
getDSFungsiProgram() As
ICollection
Return
getData("D_FUNGSI_PROGRAM")
End Function

```

```

Public Function
getDSSubFungsiProgram(ByVal
kondisi As String) As
ICollection
Return
getData("D_SUB_FUNGSI_PROGRAM",
"FUNGSI_PROG", kondisi)
End Function

```

```

Public Function
getDSSubItemFungsiProgram(ByVal
kondisi As String) As
ICollection
Return
getData("D_SUB_ITEM_FUNGSI_PROG
RAM", "SUB_FUNGSI", kondisi)
End Function

```

```

Public Function
getDSPersenChe() As ICollection
Return
getData("D_PERSEN_CHE")
End Function

```

```

Public Function
getDSKabupaten() As ICollection
Return
getData("D_KABUPATEN")
End Function
Public Function
getDSPosyandu() As ICollection
Return
getData("T_POSYANDU")
End Function

```

```

Public Function getDSUpk()
As ICollection
Return getData("D_UPK")
End Function

```

```

Public Function
getDSUpkPer(ByVal triw As
Int16, ByVal thn As String) As
ICollection
Dim dt As DataTable =
New DataTable
Dim dr As DataRow
dt.Columns.Add(New
DataColumn(strVal,
GetType(String)))
dt.Columns.Add(New
DataColumn(strText,
GetType(String)))

```

```

Dim q As String =
"SELECT T.UPK, U.NAMA FROM
T_INFO_KUNCI_TBC_DETIL AS T "
q &= "INNER JOIN D_UPK
AS U ON T.UPK = U.CODE WHERE
T.TRIWULAN = '"
q &= triw & "' AND
T.TAHUN = '" & thn & "' ORDER
BY U.CODE"

```

```

If (dbConn.State =
ConnectionState.Closed) Then
dbConn.Open()
End If

```

```

dbCmd.CommandText = q
drVar =
dbCmd.ExecuteReader()
While drVar.Read()
dr = dt.NewRow
dr(0) =
drVar.GetString(0)
dr(1) =
drVar.GetString(1)
dt.Rows.Add(dr)
End While
drVar.Close()
dbConn.Close()

```

```

Dim dv As New
DataView(dt)
Return dv
End Function
Public Function
getJmlPddkKab(ByVal thn As
String) As Integer
Dim q As String =
"SELECT TOTAL PNDDK FROM
V_JML_PENDUDUK_KAB WHERE TAHUN
= '" & thn & "'"
If dbConn.State =
ConnectionState.Closed Then
dbConn.Open()
End If
dbCmd.CommandText = q
drVar =
dbCmd.ExecuteReader
If drVar.Read Then
Return
drVar.GetInt32(0)
End If
drVar.Close()
dbConn.Close()
Return 0
End Function

```

```

Public Function
getDSPuskesmas() As ICollection
Return
getDSData("T_PUSKESMAS")
End Function

```

```

Public Function
getDSGudang() As ICollection
Return
getDSData("D_GUDANG")
End Function

```

```

Public Function
getDSRumahSakit() As
ICollection
Return
getDSData("T_RUMAH_SAKIT")
End Function

```