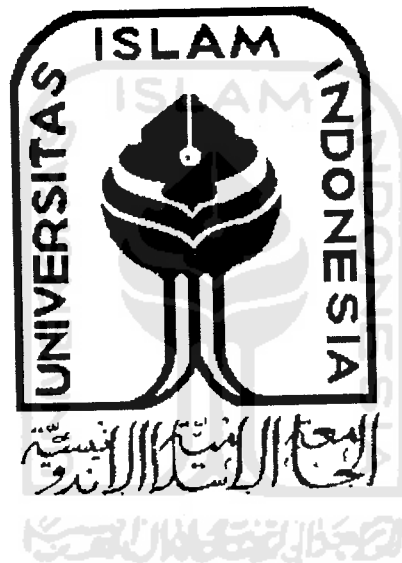


**SISTEM INFORMASI PENCARIAN LOKASI PERUMAHAN  
DI YOGYAKARTA BERBASIS WEB  
TUGAS AKHIR**

**Diajukan sebagai Salah Satu Syarat  
untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika**



oleh:

Nama : Hendrajati Nugroho

No. Mahasiswa : 00523132

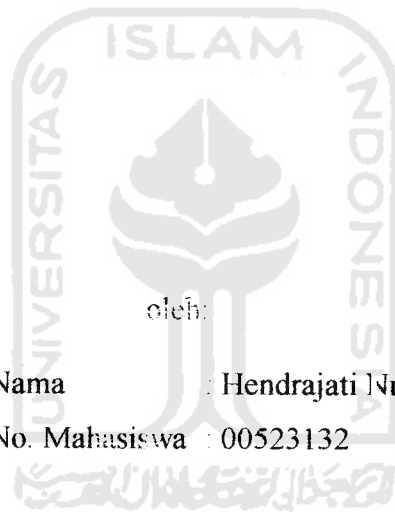
**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA  
YOGYAKARTA**

**2007**

**LEMBAR PENGESAHAN PEMBIMBING**

**SISTEM INFORMASI PENCARIAN LOKASI PERUMAHAN**  
**DI YOGYAKARTA BERBASIS WEB**

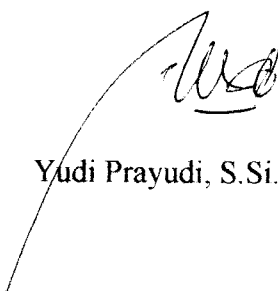
**TUGAS AKHIR**



oleh:  
Nama : Hendrajati Nugroho  
No. Mahasiswa : 00523132

Yogyakarta, 19 September 2007

Pembimbing,



Yudi Prayudi, S.Si.,M.Kom.

**LEMBAR PENGESAHAN PENGUJI**

**SISTEM INFORMASI PENCARIAN LOKASI PERUMAHAN  
DI YOGYAKARTA BERBASIS WEB**

**TUGAS AKHIR**

oleh:

Nama : Hendrajati Nugroho

No. Mahasiswa : 00523132

Telah Dipertahankan di Depan Sidang Penguji sebagai Salah Satu Syarat

untuk Memperoleh Gelar Sarjana Jurusan Teknik Informatika

Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 19 September 2007

**Tim Penguji**

Yudi Prayudi, S.Si., M.Kom

Ketua

Taufik Hidayat, ST., MCS

Anggota

Hendrik, ST

Anggota

Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Teknologi Industri

Universitas Islam Indonesia



Yudi Prayudi, S.Si., M.Kom.

## PERSEMBAHAN

Untuk Ibunda dan Ayahanda tercinta

Mbak Lia

Dek Irez

Dan Semua Teman - teman Seangkatan

Yang tak bisa saya sebutkan satu persatu





## MOTO

Sesungguhnya setelah kesulitan itu ada kemudahan

[Q.S:94:6]

Maka apabila kamu telah selesai (dari sesuatu urusan),

kerjakanlah dengan sungguh - sungguh (urusan) yang lain

[Q.S:94:7]

Jangan mau jadi pengecut, hidup sekali harus berarti. Ada yang berubah, ada yang bertahan, karena zaman tak bisa dilawan. Yang pasti, kepercayaan harus diperjuangkan

[Chairil Anwar]



## KATA PENGANTAR

Assalamu'alaikum Warohmatullahi Wabarokatuh

Alhamdulillah Rabbil'alamiin segala puji bagi Allah Tuhan semesta alam. Berkat rahmat dan hidayah-Nya sehingga Tugas Akhir yang berjudul "SISTEM INFORMASI Pencarian Perumahan di Yogyakarta Berbasis WEB" dapat terselesaikan. Shalawat dan salam semoga senantiasa tercurah kepada Muhammad SAW sebagai uswah dan inspirator perjuangan penyusun, beserta keluarga, sahabat dan umatnya hingga hari kiamat.

Tugas Akhir ini diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana Teknik Informatika Fakultas Teknologi Industri Univeraitas Islam Indonesia. Dalam menyelesaikan Tugas Akhir ini penyusun mendapat masukan dari berbagai pihak yang sangat bermanfaat bagi penyusun.

Penyusun dalam kesempatan ini menghaturkan terima kasih kepada :

1. Bapak Fathul Wahid, ST., M.Sc. selaku Dekan Fakultas Teknologi Industri.
2. Bapak Yudi Prayudi, S.Si.,M.Kom, selaku Ketua Jurusan Teknik Informatika.
3. Bapak Yudi Prayudi, S.Si.,M.Kom, selaku Dosen Pembimbing.
4. Ibunda dan Ayahanda tercinta, atas curahan bahasa kasih sayang yang tak terbalaskan
5. Mbak Lia dan dek Irez, dengan dorongan semangat yang tak pernah pudar.
6. Teman maen dalam suka dan duka Desri, Pintoko, Tyo, Maha, Alex, Desy, Shasha dan kawan – kawan yang tidak dapat disebutkan satu – persatu, atas bantuan berupa materi, non materi maupun do'anya.

Dalam penyelesaian Tugas Akhir ini penyusun menyadari bahwa masih banyak terdapat kesalahan dan kekurangannya, untuk itu penulis mengharapkan kritik dan saran yang membangun agar bisa berguna untuk masa mendatang.

Akhir kata, Semoga Tugas Akhir ini dapat bermanfaat bagi semua untuk diamalkan dan menjadi barokah dunia sampai akhirat.

Amin.

Wassalamu'alaikum Warohmatullahi Wabarokatuh

Yogyakarta, 19 September 2007



Penyusun

## DAFTAR ISI

	Halaman
JUDUL .....	i
LEMBAR PENGESAHAN PEMBIMBING .....	ii
LEMBAR PENGESAHAN PENGUJI .....	iii
PERSEMBAHAN .....	iv
MOTO .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	xii
DAFTAR TABEL .....	xv
SARI .....	xvi
ABSTRACT .....	xvii
TAKARIR .....	xviii
1 BAB I. PENDAHULUAN .....	1
1.1. Latar Belakang Permasalahan .....	1
1.2. Rumusan Permasalahan .....	2
1.3. Batasan Permasalahan .....	2
1.4. Tujuan Penelitian .....	3
1.5. Manfaat Penelitian .....	3
1.6. Sistematika Penulisan .....	4
2 BAB II. TINJAUAN PUSTAKA .....	5
2.1. Landasan Teori .....	5

2.1.1.	Pengertian Sistem Informasi dan Basis Data ( <i>Database</i> ) .....	5
2.1.2.	Sekilas Tentang Internet .....	16
2.1.3.	World Wide Web (WWW) .....	16
2.1.4.	Program Aplikasi .....	22
3	BAB III. ANALISA KEBUTUHAN .....	26
3.1.	Metode Analisis .....	26
3.2.	Analisis Masalah .....	26
3.3.	Hasil Analisis .....	27
3.3.1.	Gambaran Umum Sistem .....	27
3.3.2.	Kebutuhan Perangkat Lunak .....	27
3.3.3.	Analisis Kebutuhan Masukan .....	28
3.3.4.	Analisis Kebutuhan Keluaran .....	28
3.3.5.	Fungsionalitas yang Dikehendaki .....	29
3.3.6.	Kinerja yang Harus Dipenuhi .....	29
3.3.7.	Antarmuka Aplikasi untuk Operator .....	29
3.3.8.	Antarmuka Aplikasi untuk Calon Konsumen .....	29
4	BAB IV. PERANCANGAN .....	30
4.1.	Metode Perancangan .....	30
4.2.	Hasil Perancangan .....	30
4.3.	Data Flow Diagram (DFD) .....	31
4.3.1.	Context Diagram (Diagram Konteks) .....	31
4.3.2.	DFD Level-0 .....	32
4.3.3.	DFD Level-1 .....	33
4.3.4.	DFD Level-2 .....	34
4.4.	Struktur tabel dalam Basis Data .....	35

4.4.1.	Daftar Tabel .....	35
4.4.2.	Relasi antar Tabel .....	38
4.5.	Rincian Prosedur-Prosedur .....	39
4.5.1.	Log-In .....	40
4.5.2.	Input Data .....	40
4.5.3.	Penyajian Informasi .....	41
4.6.	Perancangan Antarmuka Berbasis Web .....	42
4.6.1.	Rancangan Layout Keseluruhan .....	42
4.6.2.	Log-In .....	43
4.6.3.	Navigasi .....	43
4.6.4.	Input Data Kecamatan .....	44
4.6.5.	Input Data Kelurahan .....	45
4.6.6.	Data Developer .....	45
4.6.7.	Data Perumahan .....	46
4.7.	Perancangan Antarmuka untuk Calon Pelanggan .....	47
4.7.1.	Pencarian Sederhana .....	48
4.7.2.	Pencarian Mode Ahli .....	49
5	<b>BAB V. IMPLEMENTASI</b> .....	50
5.1.	Batasan Implementasi .....	50
5.1.1.	Asumsi yang Dipakai .....	50
5.1.2.	Lingkungan Pengembangan .....	50
5.1.3.	Bahasa yang Dipakai .....	50
5.1.4.	Perangkat Lunak yang Digunakan .....	50
5.1.5.	Perangkat Keras yang Digunakan .....	52
5.2.	Implementasi .....	52

5.2.1.	Konfigurasi Komputer dan Instalasi Sistem .....	53
5.2.2.	Instalasi dan Konfigurasi Aplikasi Pendukung .....	53
5.2.3.	Pembangunan Aplikasi .....	54
6	BAB VI. ANALISIS KINERJA .....	63
6.1.	Dokumentasi Penggunaan Sistem .....	63
6.2.	Analisis Kinerja .....	63
6.2.1.	Aplikasi Operator .....	63
6.2.2.	Aplikasi untuk Calon Konsumen .....	64
7	BAB VII. KESIMPULAN DAN SARAN .....	73
7.1.	Kesimpulan .....	73
7.2.	Saran .....	73
	DAFTAR PUSTAKA.....	75
	LAMPIRAN	



## DAFTAR GAMBAR

Gambar 1	Macam-macam Contoh Entity .....	12
Gambar 2	Macam-macam atribut dari entitas Kamar.....	12
Gambar 3	Relasi antara pegawai dan Masalah.....	13
Gambar 4	Relationship One to One .....	13
Gambar 5	Relationship One to Many.....	13
Gambar 6	Relationship Many to Many.....	14
Gambar 7	Diagram Konteks Sistem Informasi Pencarian Perumahan Berbasis Web.....	32
Gambar 8	DFD Level-0 .....	32
Gambar 9	DFD Level-1 .....	33
Gambar 10	DFD Level-2 .....	34
Gambar 11	Relasi antar tabel.....	38
Gambar 12	Halaman Utama.....	43
Gambar 13	Rancangan Log-In .....	43
Gambar 14	Rancangan Navigasi Umum.....	44
Gambar 15	Rancangan Pengolahan Data kecamatan.....	44
Gambar 16	Rancangan Pengelolaan Data Kelurahan .....	45
Gambar 17	Rancangan Pengelolaan Data Developer .....	46
Gambar 18	Rancangan Pengelolaan Data Perumahan.....	47
Gambar 19	Pencarian Data Mode Sederhana .....	48
Gambar 20	Pencarian Data Mode Ahli.....	49
Gambar 21	Form Utama .....	55



Gambar 22	Login Box .....	56
Gambar 23	Navigasi Utama.....	56
Gambar 24	Data Kecamatan.....	57
Gambar 25	Entri Data Kecamatan.....	57
Gambar 26	Entri Data Keyword .....	58
Gambar 27	Form Data Kelurahan.....	58
Gambar 28	Entri Data Kelurahan .....	59
Gambar 29	Form Data Developer.....	59
Gambar 30	Entri Data Developer .....	60
Gambar 31	Form Data Perumahan .....	61
Gambar 32	Entri Data Perumahan.....	61
Gambar 33	Peringatan Untuk Kesalahan Login .....	64
Gambar 34	Peringatan Kesalahan Input Data.....	65
Gambar 35	Data Kelurahan Sebelum Diedit.....	65
Gambar 36	Kelurahan Akan Diubah .....	65
Gambar 37	Kelurahan setelah berubah.....	66
Gambar 38	Kelurahan Pendowoharjo yang salah akan dihapus.....	66
Gambar 39	Peringatan Bahwa data akan dihapus.....	66
Gambar 40	Kelurahan Pendowoharjo yang Telah Dihapus.....	67
Gambar 41	Halaman Pencarian .....	67
Gambar 42	Pencarian Mode Ahli .....	68
Gambar 43	Hasil Pencarian Sederhana.....	68
Gambar 44	Detail Informasi Developer.....	69

Gambar 45	Gagal Menemukan Data.....	70
Gambar 46	Pencarian Sederhana .....	71
Gambar 47	Pencarian Mode Ahli Dengan Menentukan Tipe Keyword..	71
Gambar 48	Pencarian Mode Ahli Dengan Menentukan Tipe Keyword Dan Pilihan Hasil Pencarian yang Sama Persis Dengan Keyword .....	72



## DAFTAR TABEL

Tabel 4.4.1.1	Tabel Kecamatan.....	36
Tabel 4.4.1.2	Tabel Kelurahan.....	36
Tabel 4.4.1.3	Tabel Developer.....	36
Tabel 4.4.1.4	Tabel Perumahan.....	37
Tabel 4.4.1.5	Tabel KeyWord.....	38



## SARI

*User friendly, accessible* dan *up-to-date* menjadi identitas bagi sebuah sistem aplikasi. Dengan beban yang terus bertambah dari waktu ke waktu tidak mereduksi kinerja sistem yang mengedepankan asas efektif dan efisien.

*Web* sebagai salah satu alternatif transfer data yang tidak tergantung pada lokasi dan waktu bukanlah pilihan yang salah. Dengan konektivitas yang bersifat terbuka, artinya dapat diakses oleh perangkat apa saja (*Personal Computer, Notebook, PDA*) membuat *web* menjadi prioritas utama.

Tersebarnya lokasi perumahan di Yogyakarta dan belum adanya semacam sistem informasi yang memudahkan bagi calon konsumen perumahan dalam memilih lokasi perumahan menjadi tantangan dalam penyusunan sistem. Sistem yang dikembangkan dengan bahasa pemrograman PHP ini hanya menggunakan sebuah antarmuka berbasis web, tetapi berbeda dalam pengaksesan halamannya. Operator harus melalui verifikasi nama dan *password* sebelum dapat mengakses seluruh fungsi yang ada dalam sistem. Sedangkan calon konsumen hanya dapat mengakses fungsi pencariannya dan tanpa harus melewati verifikasi nama dan *password*.

Studi kasus ini diimplementasikan secara *online* dan calon konsumen perumahan dapat mengakses aplikasi melalui *web browser*. Setelah dilakukan pengujian terhadap sistem terbukti bahwa calon konsumen dapat dengan mudah dan cepat mengakses informasi lokasi perumahan yang dikehendaki. Diharapkan sistem dapat diimplementasikan dalam penyediaan informasi mengenai lokasi perumahan.

Kata Kunci : *Web*, lokasi perumahan, PHP, MySQL.

## ABSTRACT

User friendly, accesable and up-to-date are the words describing for application systems. With resource that always have improved from time over time will not reduce system performace makes efectiveness and efeciency as their main objective.

Web is one of alternative ways to transfer data which is independent from time and location is not a false choice. With its open connectivity, means that it can be accessed from any devices (Personal Computer, Notebook, PDA) made it becomes the first priority.

Spreaded housing location in Yogyakarta and unavailability of such an information system that can help future house consumer for choosing prefered housing location is a major challenge for system development. This PHP based development system is using only one web based interface, but has a difference in page accessing. Operator must have to passed through name and password verification system before accessing all available functions. Whereas future consumer can only access searching function and without passing through name and password verification.

This study case is implemented as an online system and so future housing consumer can access this application via web browser. After doing several system testings, it is proven that future consumer can access to information related to prefered housing location, with easily and rapidly. Looking forward that this system can be implemented to supply information of housing location.

Keywords : Web, housing location, PHP, MySQL.

## TAKARIR



<i>anonymous</i>	user yang belum login
<i>up-to-date</i>	penambahan data baru
<i>developer</i>	pengembang perumahan
<i>disable</i>	menu tidak ditampilkan
<i>enable</i>	menu yang ditampilkan
<i>extended</i>	tambahan fasilitas
<i>database</i>	basis data perumahan
<i>general</i>	perumahan yang standar
<i>group</i>	kelompok perumahan
<i>interface</i>	antarmuka sistem informasi perumahan
<i>high priority</i>	user prioritas tinggi
<i>key</i>	kunci dalam tabel
<i>list</i>	daftar tabel perumahan
<i>request</i>	permintaan data perumahan
<i>service</i>	layanan untuk user
<i>uptime</i>	kelangsungan data perumahan
<i>value added service</i>	layanan berbasis internet

## BAB I

### PENDAHULUAN

#### 1.1. Latar Belakang Permasalahan

Pertumbuhan jumlah penduduk Indonesia yang cukup pesat telah menimbulkan permasalahan dan tantangan tersendiri, terutama dalam hal penyediaan tempat tinggal. Dan bagi sejumlah pihak, menganggap hal tersebut sebagai peluang usaha, terutama bagi pengembang (*developer*) perumahan. Para *developer* berlomba-lomba menyediakan hunian yang layak bagi konsumen, dengan berbagai pilihan lokasi, model, fasilitas dan juga harga. Tetapi dengan banyaknya pilihan-pilihan yang ditawarkan oleh para *developer* tersebut dapat menimbulkan permasalahan baru bagi calon konsumen.

Perkembangan teknologi *wireless*, khususnya teknologi seluler yang begitu pesat, harus dibarengi dengan perkembangan *platform* aplikasi pada teknologi seluler tersebut. Sehingga tidak hanya mengakomodasi kemajuan perangkat kerasnya saja, akan tetapi juga mampu menciptakan *value added service* bagi teknologi seluler itu sendiri, serta dapat membuka pasar baru dalam bisnis pertelekomunikasi dunia. Untuk alasan inilah dipilih aplikasi berbasis *web*, yang tidak bergantung pada sebuah *platform* tersendiri, untuk membantu mengetahui informasi lokasi perumahan di Yogyakarta.

## 1.2. Rumusan Permasalahan

Rumusan permasalahan merupakan pokok suatu permasalahan yang menjadi acuan untuk mencari solusi terbaik. Dalam penulisan skripsi ini yang menjadi rumusan permasalahan adalah bagaimana membangun perangkat lunak (*software*) untuk membantu calon konsumen perumahan, dalam memilih perumahan yang dikehendaki berdasarkan lokasi, melalui *web*, yang nantinya akan diproses di *web server* menggunakan bahasa pemrograman PHP dan hasil pencarian juga ditampilkan melalui *web*, layaknya sebuah *search engine* (mesin pencari).

## 1.3. Batasan Permasalahan

Mengingat begitu luasnya permasalahan yang terdapat dalam topik, maka permasalahan ini dibatasi pada pembuatan desain *software* sistem informasi untuk membantu calon konsumen untuk memilih lokasi perumahan yang dikehendaki, dan dalam permasalahan ini akan dibahas hal-hal berikut:

1. Keluaran sistem berupa *web* yang berisi informasi perumahan di Yogyakarta, berdasarkan lokasi calon konsumen (si pengirim SMS). Apabila masukan lokasi dari calon konsumen tidak ada perumahan, atau ada keterbatasan *database* yang ada, maka sistem akan menampilkan pesan bahwa lokasi perumahan tidak ditemukan.
2. Lokasi perumahan hanya dibatasi untuk wilayah Yogyakarta



3. Bahasa pemrograman yang dipakai adalah PHP dengan MySQL Server untuk mengolah *database*-nya.
4. Lingkungan pengembangan (*development environment*) yang digunakan adalah Macromedia Dreamweaver.

#### **1.4. Tujuan Penelitian**

Maksud dan tujuan yang ingin dicapai dalam skripsi ini adalah untuk membantu masyarakat, terutama calon konsumen perumahan dalam memilih lokasi perumahan yang dikehendaki.

#### **1.5. Manfaat Penelitian**

Manfaat yang dapat diambil dari pelaksanaan tugas mandiri selama satu bulan diantaranya adalah:

1. Mahasiswa
  - a. Mengembangkan ilmu yang telah diperoleh selama masa studi.
  - b. Melatih untuk berusaha menyelesaikan masalah yang nantinya akan dihadapi.
  - c. Untuk menambah wawasan cara berfikir dan bertindak dalam menghadapi lingkungan kerja yang sebenarnya.

## 2. Masyarakat

Membantu masyarakat umum untuk mempermudah dalam menentukan perumahan yang dikehendaki berdasarkan lokasi, melalui medium internet yang saat ini sudah sangat memasyarakat, yaitu *web browser*.

## 3. Perguruan Tinggi

- a. Sebagai pembanding atau alat ukur bagi pengembangan kualitas pendidikan di masa yang akan datang, antara teori dan praktek.
- b. Dapat memberikan sumbangan bagi perkembangan ilmu pengetahuan di bidang Informatika, terutama dalam pengembangan aplikasi berbasis *web*.

### 1.6. Sistematika Penulisan

Sistematika penulisan skripsi ini terbagi atas 7 bab, yaitu:

1. Pendahuluan
2. Tinjauan Pustaka
3. Analisis Kebutuhan
4. Perancangan
5. Implementasi
6. Analisis Kinerja
7. Kesimpulan dan Saran

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Landasan Teori**

##### **2.1.1. Pengertian Sistem Informasi dan Basis Data (*Database*)**

###### **2.1.1.1. Pengertian Sistem Informasi**

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau melaksanakan sasaran tertentu [JOG01].

Dari definisi di atas dapat disimpulkan bahwa sistem adalah kumpulan elemen-elemen yang saling berhubungan dan bertanggung jawab memproses masukan (*input*), sehingga menghasilkan keluaran (*output*).

###### **2.1.1.2. Pengertian Basis Data**

Dalam kehidupan sehari-hari tidak lepas dari basis data, baik yang disadari ataupun tidak. Dalam sebuah instansi, misalnya akan bergelut dengan data pegawai, dan lain-lainnya. Di dalam data pegawai biasanya terdapat data NIP, data nama, dan lain-lain yang disusun menurut huruf depan ataupun yang lainnya.

Agar data-data tersebut lebih bermanfaat maka data tersebut harus diorganisasikan dalam sebuah *file database*, yang bertujuan untuk memperoleh informasi tertentu yang dapat dimanfaatkan. Cara penyajian informasi pun bermacam-macam, antara lain divisualisasikan dengan grafik, diagram, gambar, dan sebagainya. Bagi sebuah organisasi atau perusahaan, informasi merupakan aset yang besar dan sangat berguna. Dengan informasi ini, organisasi akan mempengaruhi perkembangan usahanya.

Basis data (*database*) dapat diartikan sekumpulan data tentang suatu benda atau kejadian yang saling berhubungan [WAL01], pengertian basis data tersebut mungkin dianggap masih sangat umum. Dalam prakteknya mungkin dapat disimpulkan dalam penggunaan istilah basis data adalah sekumpulan dari data yang saling berhubungan [MAR02].

Untuk mendapatkan atau menghasilkan informasi yang baik, dalam arti tepat waktu, tepat guna dan tepat biaya, sangat bergantung dari faktor yang mempengaruhinya. Faktor yang mempengaruhi hal tersebut adalah:

1. Kualitas data yang menjadi bahan pembentukan informasi, dan
2. Proses pengolahan datanya.

Jika ada anggapan bahwa kualitas data yang menjadi bahan terbentuknya informasi, maka lepas dari itu semua, proses pengolahan datapun masih dipengaruhi oleh beberapa unsur lagi, yaitu:

1. *Hardware*, perangkat keras yang dibutuhkan untuk membantu pengolahan basis data. Dalam hal ini tentu saja perangkat komputer beserta perangkat-perangkat pendukungnya.
2. *Brainware*, yaitu perangkat manusianya. manusia sebagai pengelolanya.
3. *Software*, yaitu perangkat lunak yang digunakan untuk mengoperasikan komputer. Unsur *software* ini meliputi sistem operasi yang merupakan dasar pengoperasian komputer, program aplikasi yang telah dibuat pemrogram, serta yang tak kalah pentingnya adalah unsur DBMS (*Database Management System*).

#### 2.1.1.3. DBMS

DBMS merupakan *software* yang tugas utamanya adalah membantu dalam mewujudkan sistem basis data yang sempurna. Pekerjaan ini bukanlah suatu pekerjaan yang mudah. Basis data tersebut harus dapat diatur sehingga dapat tersimpan di dalam media simpan dengan baik tanpa menimbulkan kekacauan, dan dapat dipakai oleh banyak pemakai sesuai dengan kepentingan masing-masing pemakai. Jangan sampai ada yang ditonjolkan dan ada yang tidak dihiraukan dari masing-masing kepentingan pemakai tersebut.

Nama-nama *software* seperti dBase, Foxpro, dan Clipper merupakan nama yang sudah cukup dikenal dalam dunia *software*. Semua ini merupakan contoh dari PC DBMS, yaitu DBMS yang digunakan pada komputer jenis PC. Dalam skala yang lebih besar dikenal dengan DBMS yang bernama Sybase, DB2, Informix, Ingres, Progress, Interbase, Oracle, SQL Server dan masih banyak lagi.

#### 2.1.1.4. Bahasa Basis Data

Supaya pengguna dapat mengoperasikan DBMS, maka menggunakan bahasa yang dapat dimengerti. Terdapat 2 bahasa basis data, yaitu [FAT01]:

##### 1. *Data Definition Language* (DDL)

Bahasa ini berguna dalam pembuatan desain database, misalnya, pembuatan tabel, pembuatan *index*, pengubahan tabel, penyimpanan tabel dan sebagainya. Apabila *Data Definition Language* (DDL) dikompilasi akan berupa kumpulan tabel yang disimpan dalam *file* khusus yang disebut Kamus Data.

##### 2. *Data Manipulation Language* (DML)

Bahasa ini digunakan untuk melakukan manipulasi pada data. Operasi yang dapat dilakukan menggunakan DML meliputi:

- a. Penyimpanan data baru.
- b. Penghapusan data
- c. Pengubahan data.

#### 2.1.1.5. Desain Basis Data

Mendesain basis data (*database*) sebelum membuatnya di SQL merupakan suatu yang mendasar dan sangat penting. Dalam proses ini akan menggunakan ERD untuk mendesainnya. ERD (*Entity Relationship Diagram*) ialah sebuah pemodelan untuk mendesain basis data yang baik. Karena tanpa menggunakan ERD ini, proses pembuatan basis data akan berjalan lama dan tidak teratur dan dimungkinkan akan mengalami pendesainan berulang-ulang. Dalam mendesain basis data, yang harus diperhatikan ialah membuat relasi-relasi yang benar diantara tabel. Keharusan dalam pendesain basis data tersebut tujuannya untuk mempermudah di dalam pengembangan sistem yang sedang kita bangun selanjutnya.

Pada penanganan sistem informasi menggunakan komputer, sumberdaya yang dimaksud mencakup data, perangkat lunak DBMS, sistem komputer dan media penyimpanan, dan manusia yang menggunakan serta mengelola data. Dengan demikian dapat dikatakan bahwa sistem basis data merupakan bagian dari sistem informasi yang lebih besar.

Langkah-langkah desain basis data adalah:

1. Menetapkan isi

Langkah awal pembuatan basis data adalah menentukan isi. Hal ini akan menentukan target sebuah basis data. Penetapan tujuan ini dapat

dikerjakan bersama-sama dengan *end-user* atau pengguna program, sehingga basis data yang akan dibuat jelas sasarannya.

## 2. Pengelompokan data

Pengelompokan data akan menentukan jumlah tabel yang diperlukan. Data dikelompokkan berdasarkan informasinya. Sehingga tidak akan ditemui kerancuan penentuan tabel. Proses ini juga dinamakan Normalisasi.

## 3. Penyusunan tabel

Setelah diketahui pembagian data, akan diperoleh jumlah tabel yang diperlukan. Dalam penyusunan tabel ditentukan atribut apa saja yang diperlukan dalam suatu entitas. Selain itu, penentuan tipe data untuk isi tabel.

## 4. Hubungan antar tabel

Dalam aplikasinya penyusunan hubungan antartabel disebut dengan *mapping tabel*. Dari langkah ini diketahui kebutuhan *primary key*, *foreign key* sehingga akan diperoleh suatu tabel yang saling berhubungan dan menjaga validasi data.

Pokok dari perancangan basis data adalah bagaimana merancang struktur logikal dan fisikal dari satu atau lebih basis data untuk memenuhi kebutuhan informasi yang diperlukan oleh pemakai sesuai dengan aplikasi-aplikasi yang telah ditentukan. Ada beberapa tujuan perancangan basis data yang perlu diperhatikan [MAR01], yaitu:



- a. Memenuhi kebutuhan informasi sesuai dengan yang diperlukan oleh pemakai untuk aplikasi tertentu.
- b. Mempermudah pemahaman terhadap struktur informasi yang tersedia dalam basis data.
- c. Memberikan keterangan tentang persyaratan pemrosesan dan kemampuan sistem, seperti lama pengaksesan data, kapasitas memori yang harus ada, dan sebagainya.

#### **2.1.1.6. Pemodelan Data**

Model data adalah kumpulan perangkat konseptual untuk menggambarkan data, hubungan data, semantik (makna) data serta batasan data [FAT01].

Model data yang secara umum digunakan adalah Model Kerelasiaan Antar Entitas (*Entity Relationship Model*). *Entity Relationship Model* akan dibahas sebagai berikut.

##### **2.1.1.6.1. Pengertian *Entity***

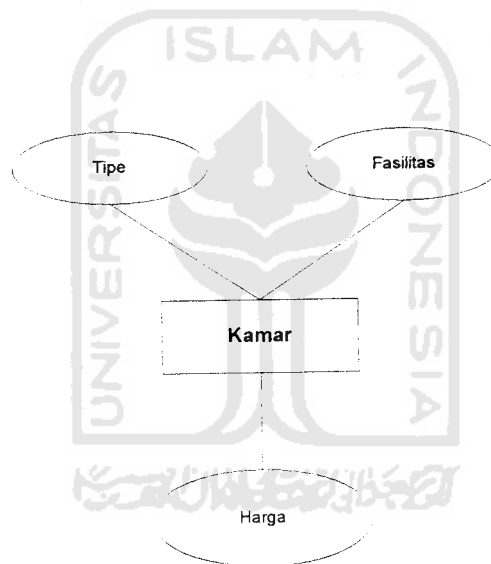
*Entity* merupakan objek riil yang dibedakan satu dengan yang lainnya dan tidak saling bergantung [PUJ01]. Objek dapat berupa benda konkrit ataupun abstrak. Misalnya: Mahasiswa, Mata kuliah, Pekerjaan dan lain-lain. *Entity* pada umumnya dilambangkan dengan kotak segi empat.



Gambar 1. Macam-macam contoh Entity

#### 2.1.1.6.2. Pengertian Atribut

Atribut menerangkan suatu *entity* dan mempunyai nilai [PUJ01], Sebagai contoh, entitas Kamar mempunyai atribut Tipe, Fasilitas, Harga dan sebagainya. Atribut dilambangkan dalam bentuk oval.



Gambar 2. Macam-macam atribut dari entitas Kamar

#### 2.1.1.6.3. Tipe-tipe Relasi ( *Relationship* )

Relasi dalam kaitannya dengan entitas merupakan hubungan antara entitas yang satu dengan entitas yang lain [FAT01]. Misalnya, entitas seorang pegawai dengan NIP = '99514080' dan Nama = 'Futuh Fauzan' (yang ada di himpunan entitas pegawai) mempunyai relasi dengan entitas sebuah masalah dengan kode<sub>msl</sub> = 'M-002' dan

nama\_msl = 'Bolos'. Relasi di antara kedua entitas tadi mengandung arti bahwa pegawai tersebut sedang mempunyai masalah tersebut di instansi tersebut.



Gambar 3. Relasi antara pegawai dan Masalah

Ada tiga tipe *relationship* antar entitas, yaitu [FAT01]:

### 1. *One to One Relationship (1:1)*

Pada bentuk relasi ini, satu entitas hanya berelasi dengan satu entitas lainnya.



Gambar 4. Relationship One to One

### 2. *One to Many Relationship (1:N)*

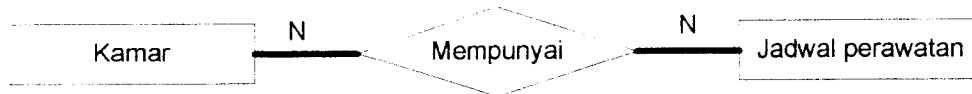
Pada relasi ini satu entitas bisa mempunyai relasi lebih dari satu pada entitas lainnya.



Gambar 5. Relationship One to Many

### 3. *Many to Many Relationship ( N: N )*

Pada relasi ini banyak entitas bisa realasi dengan banyak entitas lainnya.



Gambar 6. *Relasitionship Many to Many*

#### 2.1.1.7. Tipe-tipe Kunci (Key)

Key suatu properti yang menentukan apakah suatu kolom (*field*) pada suatu tabel sangat penting atau tidak. *Key* mempunyai 3 macam, di antaranya sebagai berikut:

##### 2.1.1.7.1. *Candidate Key*

*Candidate key* merupakan kumpulan atribut minimal dapat membedakan setiap baris data (*record*) dalam sebuah tabel secara unik dan fungsi dari *Candidate Key* adalah sebagai calon *primary Key*.

##### 2.1.1.7.2. *Primary Key*

*Candidate key* yang dipilih untuk mengidentifikasi tiap baris (*record*) secara unik disebut sebagai *Primary Key*. Dan ketentuan untuk *primary key* ialah *field* dari *record* tersebut harus benar-benar unik dan tidak boleh tidak boleh ada nilai *NULL*.

#### 2.1.1.7.3. *Foreign Key*

*Foreign key* merupakan *primary key* yang ada pada suatu tabel terhubung dengan tabel lain, jika suatu tabel mempunyai *primary key* yang dihubungkan dengan tabel yang lain, maka *key* yang ada pada tabel yang lain tersebut disebut *foreign key*, fungsinya sebagai penghubung antar tabel.

Dari ketiga macam kunci (*key*) tersebut mempunyai hubungan yang sangat erat dari ketiganya dan merupakan titik kunci untuk membangun suatu *relationship* beberapa tabel pada suatu sistem *database*.

#### 2.1.1.8. **Domain dan Tipe Data**

Tipe data menentukan kemampuan penentuan nilai yang akan dimasukkan pada tabel (Fathansyah, 1999). Pengaturan tipe data dikerjakan pada saat implementasi basis data. Contoh: atribut NIM, mempunyai tipe data integer, sehingga memungkinkan pengisian data antara -32.768 sampai 32.768.

Sedangkan *domain* merupakan kumpulan dari harga atomik yang dapat dimiliki oleh suatu kolom / atribut (Pujiyono, W., 2000). *Domain* dapat dikaitkan sebagai batasan-batasan nilai. Contoh: atribut IP hanya menerima nilai antara 1 sampai 4.

### 2.1.2. Sekilas Tentang Internet

Pada tahun 1969, DoD memberi tugas kepada ARPA untuk membangun sebuah mata rantai komunikasi antara DoD dengan militer yang tidak dapat disabotase oleh musuh mereka. Jaringan komunikasi yang diciptakan ini disebut ARPANET. Pada awalnya ARPANET hanya menghubungkan 4 situs saja. ARPANET yang ditemukan dan dikenal secara singkat sebagai internet, dibangun dari satu set protokol yang dikenal sebagai *Transmission Control Protocol Internet Protocol* atau TCP/IP. Pada tahun 1986 *U.S National Security Foundation* (NSF) membangun NFSNet yang menghubungkan lima pusat super komputer NSF di Amerika Serikat [AN03].

Walaupun masuknya NSFNet pada bidang ini merupakan faktor utama perkembangan internet, namun titik balik terbesarnya adalah tahun 1991. Ketika NSF mulai membuka komersialisasi lalu lintas pada jaringan, maka sejak tahun 1991 itu pula seluruh lalu lintas jaringan NSF dipergunakan oleh institusi pemerintah dan pendidikan. Jaringan-jaringan komersial segera muncul dan populasi internetpun berkembang pesat.

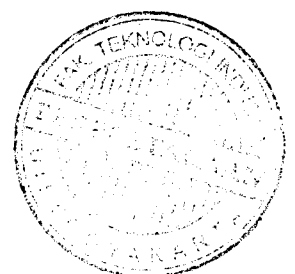
### 2.1.3. *World Wide Web* (WWW)

WWW pertama kali dibuat pada tahun 1991 di CERN, *European Particle Phisic Laboratory* di Jenewa Swiss [SID01]. Gagasan awalnya adalah agar fisikawan dan ilmuwan dapat berbagi informasi dalam bentuk multimedia. Dari gagasan tersebut kemudian diciptakanlah sebuah metode untuk transfer dan menampilkan informasi, yaitu metode transfer yang disebut HTTP (*Hypertext*

*Transfer Protocol*) yang dijalankan di TCP / IP. Protokol jaringan internet standar HTTP ini bekerja menggunakan internet standar, dimana *server* memberikan data, dan *client* menampilkan dan memprosesnya, informasi yang di transfer dibuat dengan menggunakan HTML. Setelah CERN membuat spesifikasi untuk WWW, orang mulai menulis *client* WWW dan *software server*, sehingga terbentuk WWW yang ada sekarang.

Setelah beberapa tahun, internet dikenal sebagai teknologi yang sulit untuk dipelajari dan digunakan. Hal ini sangat berbeda dengan pelayanan yang diberikan oleh *Bulletin Board System* (BBS), *Online Service*, aplikasi program yang biasa digunakan di komputer pribadi yang mudah pengoperasiannya. WWW menjadikan *interest* suatu sistem berbasis GUI (*Grafic User Interface*) yang tidak kalah dengan aplikasi PC lainnya dan sekarang telah menggantikan *Bulletin Board System* dan *Online Service*.

WWW sebenarnya merupakan suatu kumpulan informasi pada beberapa *server* komputer yang saling terhubung dalam jaringan internet. Informasi yang disajikan dalam *web* mempunyai *link-link* yang menghubungkan informasi tersebut ke informasi yang lain di dalam jaringan internet. *Link* ini mempunyai tanda khusus, yang dinyatakan teks biru dan garis bawah atau dalam bentuk gambar di dalam kotak. Suatu tanda yang pasti dapat digunakan untuk menentukan apakah suatu teks atau objek merupakan suatu *link* atau bukan, adalah dengan melihat perubahan pada petunjuk *mouse*. Jika petunjuk *mouse* berubah bentuk menjadi gambar tangan yang sedang menunjuk, berarti teks atau



gambar adalah sebuah *link* yang menunjuk ke informasi atau aplikasi lain dalam *web*.

Salah satu penyebab utama pertumbuhan *web* adalah kemudahan dalam penggunaannya. Pada aplikasi *web* cukup melakukan klik *mouse* ada suatu *link* untuk mendapatkan suatu informasi dan *link* tersebut secara otomatis akan membawa ke informasi lainya dalam *web*. Sistem yang menghubungkan informasi melalui *link* disebut dengan *hypertext*. Dengan semakin berkembangnya WWW, istilah *hypertext* ini menjadi *hypermedia*. *Link-link* penghubung antar informasi bukan lagi berupa teks tetapi juga berupa suatu *file* multimedia seperti gambar, suara atau *video*.

Dari uraian di atas, secara teknis, *web* tersusun atas berbagai komponen pendukung yang agar unjuk kerja seperti yang diuraikan secara umum diatas dapat berjalan secara optimal.

#### **2.1.3.1. HTTP (*Hypertext Transfer Protocol*)**

HTTP adalah suatu protokol yang menentukan aturan yang perlu diikuti oleh *web browser* dalam meminta atau mengambil suatu dokumen, dan *web server* dalam menyediakan dokumen yang diminta *web browser*. Protokol ini merupakan protokol standar yang dipergunakan untuk mengakses dokumen HTML.

#### **2.1.3.2. URL (*Uniform Resource Locator*)**

URL adalah suatu sarana yang digunakan untuk menentukan lokasi informasi pada suatu *web server*. URL dapat diibaratkan sebagai



suatu alamat (*path* atau subdirektori pada mode DOS) dimana alamat tersebut terdiri atas :

1. Protokol-transfer yang digunakan oleh suatu *browser* untuk mengambil informasi.
2. Nama-*host* komputer dimana informasi tersebut berada.
3. Jalur atau *path* serta nama file dari suatu lokasi.

Format umum dari suatu URL adalah : *protokol-transfer://host-name.name.domain.path\_nama\_file.html*. Sebagai contoh :

`http://localhost/hendra/skripsi/index.html`

1. `http` merupakan bentuk protokol, menyatakan cara pengambilan data yang digunakan. Bentuk lain misalnya: `ftp` (untuk *download* dan *upload*), `gopher`, `smtp`, `wais` dan lain-lain.
2. `localhost` merupakan nama *host* yang digunakan sebagai nama *server* komputer dan *web server*.
3. `/hendra/skripsi` adalah jalur atau *path* dari informasi yang dicari, *path* dapat dianalogikan sebagai *folder* atau direktori dalam *disk*.
4. `index.html` nama *file* HTML yang dipanggil.

### 2.1.3.3. HTML (Hypertext Markup Language)

HTML adalah bahasa yang sangat tepat dipakai untuk menampilkan informasi pada halaman *web*. Hal ini karena HTML menampilkan informasi dalam bentuk *hypertext*, HTML mendukung sekumpulan perintah yang dapat digunakan untuk mengatur bagaimana

informasi tersebut akan ditampilkan. Dengan kata lain HTML disebut suatu sistem standarisasi dalam melakukan penulisan dan *formatting hypertext* yang ditulis dalam dokumen teks ASCII (*American Standard Communication Information Interchange*), agar dapat menghasilkan tampilan visual yang terintegrasi [SID01]. HTML lebih menekankan pada penggambaran komponen-komponen struktur dan *formatting* di dalam halaman *web* dari pada menentukan cara penampilannya. Sedangkan untuk menampilkan cara penampihan data yang tersimpan dalam file teks HTML, ditentukan oleh *web browser*. *Web browser* digunakan untuk menterjemahkan perintah-perintah tersebut dalam bentuk susunan halaman dan *style* yang dikehendaki. Susunan halaman yang ditampilkan kekomputer lain melalui *web browser* disebut dengan *webpage* atau halaman *web*.

HTML pertama kali diilhami oleh Ted Nelson pada tahun 1965. *Hypertext* menurut Ted Nelson adalah suatu teks yang mempunyai *link* terhubung dengan teks lainnya. Untuk hubungan dengan berbagai media maka istilahnya menggunakan *hypermedia* dan bukan *hypertext*.

*World Wide Web* tidak terlepas dari HTML (*HyperText Markup Language*) yang merupakan dasar pembangunan sebuah *situs web*. Arti HTML dapat dilihat dari setiap elemennya, yaitu :

#### 1. HyperText

*World Wide Web* memungkinkan pengguna untuk berpindah dari satu lokasi *web* ke lokasi lain, ini dikarenakan adanya *hyperteks*

*link*, yaitu suatu teks yang berfungsi sebagai penghubung di dalam dan di antara *situs web*.

## 2. Markup

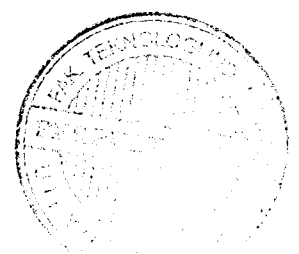
Untuk membuat suatu *link*, teks tebal, atau teks miring diperlukan suatu tanda yang disebut dengan *markup tag*, atau biasa disebut dengan *tag*.

## 3. Language

Sekalipun didefinisikan sebagai suatu *language* (bahasa), HTML bukanlah suatu bahasa pemrograman, karena *programmer* hanya menandari teks-teks sandi program yang akan ditampilkan dengan *tag*.

Struktur dasar dari setiap dokumen HTML adalah sebagai berikut:

```
<HTML>
<HEAD>
    <! Bagian kepala dari dokumen HTML>
    <TITLE> Judul Halaman Web </TITLE>
</HEAD>
<BODY>
    <! Isi dokumen HTML>
</BODY>
</HTML>
```



## 2.1.4. Program Aplikasi

### 2.1.4.1. PHP

PHP, yang merupakan akronim rekursif dari *PHP: Hypertext Preprocessor*, adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di *server* tetapi disertakan pada dokumen HTML [SIS01]. Pembuatan *web* ini merupakan kombinasi antara PHP sendiri sebagai bahasa pemrograman dan HTML sebagai pembangun halaman *web*. Ketika seorang pengguna internet akan membuka suatu situs terlebih dahulu *server* yang bersangkutan akan memproses semua perintah PHP di *server* lalu mengirimkan hasilnya dalam format HTML ke *web browser* pengguna internet. Dengan demikian seorang pengguna internet tidak dapat melihat kode program yang ditulis dalam PHP sehingga keamanan dari halaman *web* menjadi lebih terjamin.

PHP merupakan *software* yang bersifat *Open Source* dan mampu lintas *platform*. Selain dapat dibangun sebagai modul pada *web server* Apache dan sebagai *binary* yang berjalan sebagai CGI, PHP juga dapat bekerja dengan *web server* populer yang lain, seperti: Microsoft IIS, Omni HTTPD dan sebagainya. PHP menawarkan koneksitas yang baik dengan beberapa basis data, antara lain Oracle, MySQL, PostgreSQL, dll. Juga dapat berinteraksi dengan beberapa *library eksternal* seperti DF hingga *parse XML*. Juga

mendukung komunikasi dengan layanan lain melalui protokol IMAP, SNMP, NNTP, POP3 dan lain-lain.

#### 2.1.4.2. MySQL Server

MySQL, merupakan SQL DBMS berlisensi *open source* yang paling terkenal, yang dikembangkan, didistribusikan, dan didukung oleh MySQL AB ([www.mysql.com](http://www.mysql.com)). MySQL AB merupakan perusahaan komersial, yang didirikan oleh *programmer* MySQL. Perusahaan ini merupakan perusahaan *open source* generasi kedua yang menggabungkan nilai *open source* dan metodologi model bisnis yang sukses.

Sebuah *database* relasional menyimpan data dalam tabel-tabel terpisah daripada menyimpan semua data dalam sebuah ruang simpan yang besar. Hal ini akan menambah kecepatan dan fleksibilitas. Suku kata SQL dari “MySQL” merupakan singkatan dari *Structured Query Language*. SQL merupakan bahasa terstandar yang paling umum yang digunakan untuk mengakses *database*, dan SQL didefinisikan oleh ANSI/ISO SQL Standard. Standar SQL telah berkembang sejak 1986 dan sejumlah versinya telah muncul.

Kelebihan MySQL antara lain:

1. Keamanan lebih terjamin. Dengan adanya hak akses dan sistem *password* yang sangat fleksibel dan aman, dikarenakan lalu lintas *password* terenkripsi saat terhubung dengan *server*, dan mengijinkan verifikasi berbasis *host*.

2. Mampu menangani data yang sangat besar. Beberapa pengguna mempercayakan MySQL untuk mengelola 60.000 tabel yang berisi lebih dari 5 milyar *record*. Selain itu MySQL juga mengizinkan indeks dalam sebuah tabel sampai dengan 64 indeks. Dan setiap indeks dapat berisi 1 – 16 kolom, atau bagian kolom. Lebar maksimum indeks adalah 1000 *bytes* (767 *bytes* untuk tabel bertipe InnoDB).
3. Fleksibel dalam konektivitas. *Client* dapat berhubungan dengan MySQL Server melalui sejumlah protokol: soket TCP/IP untuk semua *platform*, *named-pipe* untuk Windows, *shared-memory* untuk Windows Server, atau *socket file* untuk sistem berbasis UNIX (Linux, Debian, FreeBSD, MacOS X)
4. Program *client* dapat ditulis dalam bahasa pemrograman apapun: C, C++, Delphi, PHP, Perl, Java dan sebagainya. Selain itu, sudah tersedia sejumlah *interface-library* yang memudahkan bagi *programmer* untuk mengakses MySQL Server, antara lain:
  - a. Connector/ODBC (MyODBC). Menyediakan dukungan untuk aplikasi *client* yang menggunakan koneksi ODBC (*Open Database Connectivity*). Sebagai contoh, Microsoft Access dapat membuka *database* dalam MySQL Server melalui Connector/ODBC ini. Program *client* juga dapat berjalan di Windows atau UNIX.

- b. Connector/J. Untuk mengakses MySQL Server melalui program berbasis Java
- c. MySQL Connector/NET. Untuk mengakses MySQL Server melalui aplikasi berbasis Microsoft .NET. Antarmuka yang ditulis 100% dalam bahasa C# ini mengimplementasikan secara penuh ADO.NET, dan *programmer* dapat memilih bahasa .NET favoritnya.



## BAB III

### ANALISIS KEBUTUHAN

#### 3.1 Metode Analisis

Metode analisis yang diperlukan sebagai bahan acuan adalah analisis dengan pendekatan terstruktur yang dilengkapi alat dan teknik yang membutuhkan pengembangan sistem, sehingga hasil dari analisis tersebut dapat dikembangkan ke dalam sebuah bentuk sistem yang dapat diimplementasikan secara detail dan terperinci.

Metode Analisis yang digunakan pada sistem ini adalah Metode Analisis berarah struktur data.

#### 3.2 Analisis Masalah

Masalah yang dimunculkan pada kasus ini adalah terdapatnya kendala pada pengaksesan informasi mengenai lokasi perumahan oleh calon konsumen yang dapat ditarik menjadi poin-poin sebagai berikut :

1. Pertumbuhan jumlah perumahan di Yogyakarta yang semakin pesat menimbulkan masalah baru bagi calon konsumen, terutama dari pemilihan perumahan yang cocok dengan kebutuhannya
2. Pengaksesan informasi lokasi perumahan dilakukan secara manual sehingga tidak efisien dan memakan banyak waktu dan tenaga.
3. Belum memanfaatkan kemajuan teknologi informasi terutama dalam penyediaan informasi lokasi perumahan yang dapat diakses kapan saja dan darimana saja



4. Tidak ada media untuk mengintegrasikan informasi.

### 3.3 Hasil Analisis

Hasil analisis dari sistem yang diharapkan.

#### 3.3.1 Gambaran Umum Sistem

Secara garis besar, sistem yang akan diwujudkan nantinya menangani pengaksesan informasi lokasi perumahan oleh calon konsumen. Pencarian lokasi perumahan berdasarkan daerah/ kecamatan yang diinginkan oleh calon konsumen.

Dikarenakan keinginan untuk mempermudah calon konsumen dalam memilih perumahan yang dikehendaki, kapan saja, dan di mana saja, maka diperlukan semacam sistem informasi perumahan berbasis *web*. Dengan demikian, calon konsumen cukup dengan mengirimkan lokasi kecamatan yang dikehendaki melalui antarmuka berbasis *web*, dan sistem informasi yang akan dibangun dapat menampilkan perumahan yang ada di kecamatan. Metode pencarian lokasi yang digunakan oleh sistem menggunakan metode NLP (*Natural Language Processing*).

#### 3.3.2 Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan untuk mendukung sistem yang akan dibangun diharapkan dapat memenuhi kebutuhan-kebutuhan antara lain :

1. Mampu memberikan informasi lokasi perumahan berdasarkan kecamatan yang ada di Yogyakarta yang dipilih oleh konsumen.
2. Mampu mengimplementasikan metode NLP dalam pencarian lokasi perumahan, sehingga sistem dapat bekerja lebih efisien.

3. Menampung informasi dari berbagai sumber dan *up to date*.

### 3.3.3 Analisis Kebutuhan Masukan

Sistem yang dibangun membutuhkan masukan-masukan sebagai berikut:

1. Data kecamatan

Daftar kecamatan-kecamatan yang berada di Yogyakarta, yang nantinya akan membantu dalam proses pencarian

2. Data kelurahan

Daftar kelurahan-kelurahan untuk memberikan lokasi perumahan secara lebih detail.

3. Daftar *developer*

Daftar *developer* yang membangun perumahan.

4. Data perumahan

Data perumahan-perumahan yang telah dibangun oleh *developer*.

### 3.3.4 Analisis Kebutuhan Keluaran

1. Sajian Informasi Perumahan

Sajian informasi mengenai lokasi perumahan berdasarkan kecamatan yang dikehendaki oleh calon konsumen. Informasi ini juga menyertakan nama *developer* yang membangun perumahan tersebut

2. Sajian Informasi *Developer*

Sajian berbagai informasi *developer*, yang berisi nama resmi *developer*, alamat *developer* beserta nomor telepon. Setelah mendapatkan informasi *developer*, maka calon konsumen dapat langsung menghubungi *developer*

yang bersangkutan untuk mendapatkan informasi mengenai perumahan yang diinginkan lebih lanjut.

### **3.3.5 Fungsionalitas yang dikehendaki**

Diharapkan sistem yang dibangun mampu menampung informasi yang dikirim oleh *user* tersebut di atas. Mampu menyajikan informasi perumahan berdasarkan lokasi/kecamatan yang diinginkan oleh calon konsumen, dan juga informasi tentang *developer* perumahan, sehingga calon konsumen dapat melakukan kontak bisnis secara langsung.

### **3.3.6 Kinerja yang harus dipenuhi**

Diharapkan sistem dapat bekerja dengan baik terutama pengaksesan informasi oleh calon konsumen melalui teknologi berbasis *web* yang akan ditampilkan di *web browser* favorit.

### **3.3.7 Antarmuka Aplikasi untuk Operator**

Antarmuka untuk operator yang dibangun dengan teknologi berbasis *web* bersifat ramah bagi pengguna (*user-friendly*), sehingga orang yang tidak terlalu menguasai komputer dapat menggunakannya dengan mudah.

### **3.3.8 Antarmuka untuk Calon Konsumen**

Antarmuka untuk calon konsumen juga berbasis *web* yang dapat diakses melalui aplikasi *web browser* favorit.

## BAB IV

### PERANCANGAN

#### 4.1 Metode Perancangan

Metode Perancangan yang digunakan pada perancangan Sistem Informasi Pencarian Lokasi Perumahan Berbasis *Web* ini adalah perancangan berorientasi obyek. Dengan menggunakan perancangan berorientasi obyek, struktur tabel, aliran data, logika program dan antarmuka program dapat tertata dengan baik dan mudah didokumentasikan. Ini dikarenakan dalam sebuah obyek, selain memiliki properti, atribut atau keterangan, sebuah obyek juga dapat memiliki obyek yang lain. Dengan demikian perancangan berorientasi obyek mempunyai struktur hierarki yang lebih jelas.

#### 4.2 Hasil Perancangan

Hasil perancangan meliputi perancangan Basis Data dan Perancangan Antarmuka, serta perancangan model perangkat lunak menggunakan *Data Flow Diagram* (DFD). Selain menggunakan DFD, juga dipergunakan table relationship yang menjelaskan hubungan antar tabel berdasarkan hubungan antar *primary key* dan *foreign key* yang dimiliki oleh tabel-tabel tersebut. Sedangkan perancangan antarmuka menggunakan bantuan perangkat lunak desain, Microsoft Visio. Perancangan antarmuka selain membantu dari segi dokumentasi, juga membantu pada saat pengembangan dan penulisan program (*coding*). Dengan demikian pada

saat terjadi kesalahan dalam proses pengembangan program, maka akan meminimalisasi perubahan antarmuka.

### **4.3 Data Flow Diagram**

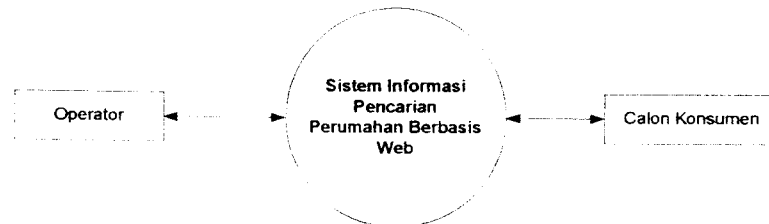
*Data Flow Diagram* yang sering diterjemahkan menjadi *Diagram Alir Data*. DAD [POH01] merupakan alat yang biasa dipakai untuk mendokumentasikan proses dalam sistem serta menekan pada fungsi-fungsi di dalam sistem, cara menggunakan informasi yang tersimpan dan pemindahan informasi antar fungsi di dalam sistem. Sedangkan DAD Konteks adalah DAD yang memperlihatkan sistem sebagai sebuah proses. Tujuannya adalah memberikan pandangan umum sistem dan memperlihatkan sebuah proses yang berinteraksi dengan lingkungannya.

#### **4.3.1. Context Diagram (Diagram Konteks)**

Diagram konteks [POH01] adalah kasus khusus DFD (bagian dari DFD yang berfungsi memetakan model lingkungan), yang dipresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem.

Untuk memetakan model lingkungan dari Sistem Informasi Pencarian Perumahan Berbasis Web, maka diperlukan sebuah diagram konteks yang digambarkan melalui gambar 8. Dalam gambar ini, operator sebagai pihak yang bertanggung jawab untuk mengelola aplikasi, memasukkan data-data yang diperlukan antara lain data kecamatan, kelurahan, developer dan perumahan ke sistem. Setelah sistem memperoleh data, calon konsumen perumahan dapat menggunakannya untuk mencari lokasi perumahan sesuai

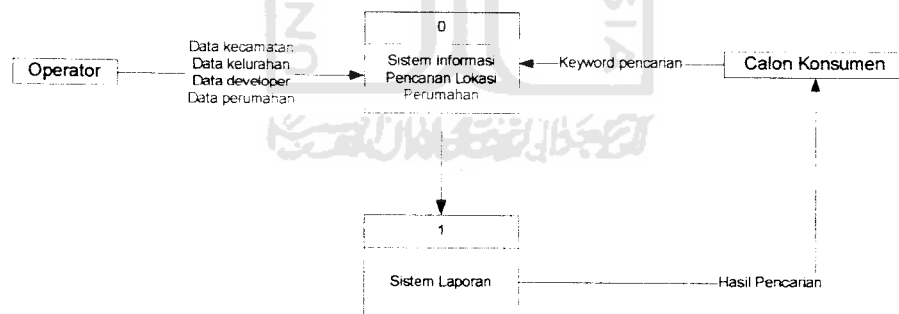
dengan keinginan calon konsumen tersebut, dan sistem akan menghasilkan keluaran sesuai dengan informasi yang diminta.



Gambar 7. Diagram Konteks Sistem Informasi Pencarian Perumahan Berbasis Web

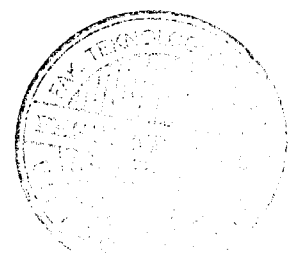
#### 4.3.2. DFD Level-0

Pada DFD Level-0 yang digambarkan dalam gambar 9, terlihat bahwa operator yang bertanggung jawab memasukkan ke dalam sistem data antara lain data kecamatan, kelurahan, *developer* dan perumahan.



Gambar 8. DFD Level-0

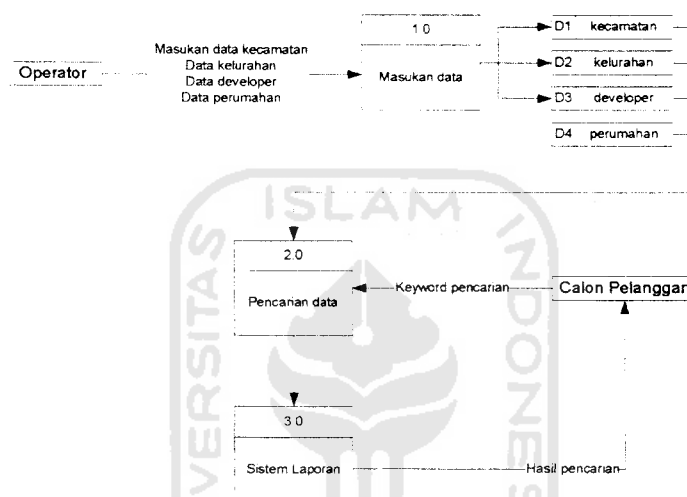
Calon pengguna berinteraksi dengan sistem melalui antarmuka berbasis *web* dengan cara memasukkan *keyword* pencarian, yang dapat berupa lokasi kecamatan ataupun kelurahan. *Keyword* diterima oleh sistem, dan sistem secara otomatis mencari data berdasarkan *keyword*, yang



selanjutnya diteruskan ke proses laporan, yang akan menampilkan hasil pencarian kepada calon konsumen juga melalui antarmuka berbasis *web*.

#### 4.3.3. DFD Level-1

DFD Level-1 merupakan penjabaran lebih lanjut dari DFD Level-0, yang ditunjukkan lewat gambar 10.



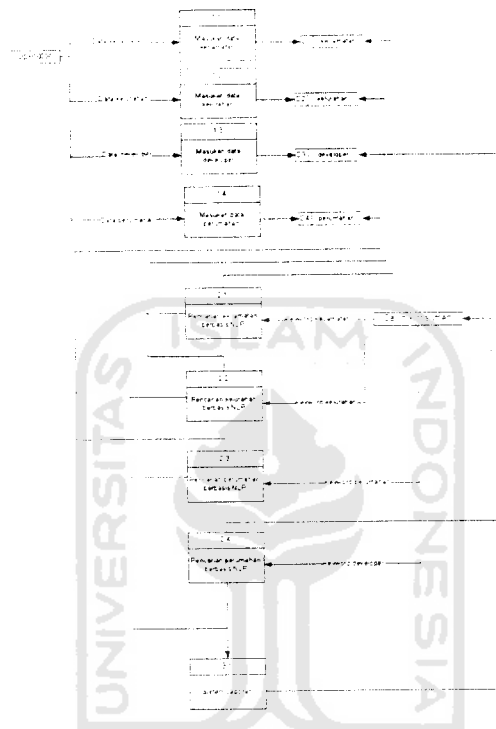
Gambar 9. DFD Level-1

Pada diagram tersebut terlihat bahwa pada proses masukan data, operator memasukkan data antara lain data kecamatan, data kelurahan, data *developer* dan data perumahan. Dan masing-masing data tersebut disimpan dalam tabel kecamatan, kelurahan, *developer* dan perumahan.

Setelah data-data tersebut telah tersedia, calon konsumen dapat masuk ke proses pencarian data dengan cara memasukkan *keyword* pencarian melalui antarmuka berbasis *web*, yang selanjutnya akan masuk ke sistem laporan yang akan menampilkan hasil pencarian kepada calon konsumen melalui antarmuka berbasis *web*.

#### 4.3.4. DFD Level-2

Untuk menjelaskan DFD Level-1 lebih lanjut, maka diperlukan DFD Level-2, yang digambarkan lewat gambar 11.



Gambar 10. DFD Level-2

Pada saat masukan data oleh operator ke sistem, akan terjadi 4 proses masukan data, yaitu proses masukan data kecamatan, proses masukan data kelurahan, proses masukan data *developer* dan proses masukan data perumahan. Dalam setiap proses masukan data, data akan tersimpan di masing-masing tabel, yaitu tabel kecamatan, kelurahan, *developer* dan perumahan.

Setelah semua proses masukan berjalan dengan lancar, calon pengguna dapat melakukan proses pencarian dengan cara memasukkan *keyword*. Jika *keyword* yang dimasukkan oleh calon konsumen adalah



*keyword* kecamatan, maka akan masuk ke dalam proses pencarian kecamatan, dan dilanjutkan ke proses laporan yang menampilkan hasil pencarian kepada calon konsumen. Begitu pula jika *keyword* yang dimasukkan adalah *keyword* kelurahan, maka secara otomatis akan masuk ke proses pencarian kelurahan. Dengan demikian, masing-masing *keyword* memiliki proses pencarian tersendiri.

#### 4.4 Struktur Tabel dalam Basis Data

Nama basis data yang digunakan adalah *sms\_nlp*. Untuk meminimalisasi penggunaan *super user/root* dalam mengakses basis data, maka dibuat *user* khusus *admin\_sms\_nlp* yang hanya memiliki ijin akses ke basis data *sms\_nlp*, dan tidak memiliki hak akses *GRANT* yang dapat digunakan untuk mengakses *user administration* (pengelolaan pengguna). Ini dikarenakan *super user/root* memiliki hak akses yang tidak terbatas untuk semua basis data dalam MySQL Server. Dan hal ini dapat menimbulkan celah keamanan yang cukup fatal, apabila ada pihak yang tidak dikehendaki mengetahui *password* milik *super user* tersebut, maka seluruh basis data yang tersimpan dalam *server* dapat dimodifikasi dengan mudah dan dapat memodifikasi hak akses *user* lain.

##### 4.4.1 Daftar tabel

###### 4.4.1.1. Tabel kecamatan

Nama tabel : kecamatan

*Primary key* : id

*Foreign key* : -

Fungsi : untuk menyimpan data kecamatan

No.	Nama field	Tipe	Ukuran	Keterangan
1.	Id	INTEGER AUTO_INCREMENT	11	Primary key
2.	Nama	VARCHAR	20	Nama kecamatan

#### 4.4.1.2. Tabel kelurahan

Nama tabel : kelurahan

Primary key : id

Foreign key : id\_kecamatan

Fungsi : untuk menyimpan data kelurahan

No.	Nama field	Tipe	Ukuran	Keterangan
1.	Id	INTEGER AUTO_INCREMENT	11	Primary key
2.	Nama	VARCHAR	20	Nama kelurahan
3.	Id_kecamatan	INTEGER	11	Foreign key dari tabel kecamatan, untuk mengidentifikasi kecamatan
4.	Kode_pos	CHAR	5	Kode pos dari kelurahan

#### 4.4.1.3. Tabel developer

Nama tabel : developer

Primary key : id

Foreign key : id\_kecamatan

Fungsi : untuk menyimpan data developer

No.	Nama field	Tipe	Ukuran	Keterangan
1.	Id	INTEGER AUTO_INCREMENT	11	Primary key

No.	Nama field	Tipe	Ukuran	Keterangan
2.	Nama	VARCHAR	25	Nama perumahan
3.	Alamat	VARCHAR	60	Alamat resmi developer
4.	Telp	VARCHAR	15	-
5.	Email	VARCHAR	60	-
6.	Id_kelurahan	INTEGER	11	Foreign key dari tabel kelurahan. Untuk mengidentifikasi kecamatan dimana developer tersebut berada

#### 4.4.1.4. Tabel perumahan

Nama tabel : perumahan

Primary key : id

Foreign key : id\_developer, id\_kecamatan

Fungsi : untuk menyimpan data perumahan

No.	Nama field	Tipe	Ukuran	Keterangan
1.	Id	INTEGER AUTO INCREMENT	11	Primary key
2.	Nama	VARCHAR	25	Nama perumahan
3.	Id_developer	INTEGER	11	Foreign key dari tabel developer. Untuk mengidentifikasi developer perumahan tersebut
4.	Id_kecamatan	INTEGER	11	Foreign key dari tabel kecamatan. Untuk mengidentifikasi lokasi kecamatan perumahan tersebut

## 4.4.1.5. Tabel keyword

Nama tabel : keyword

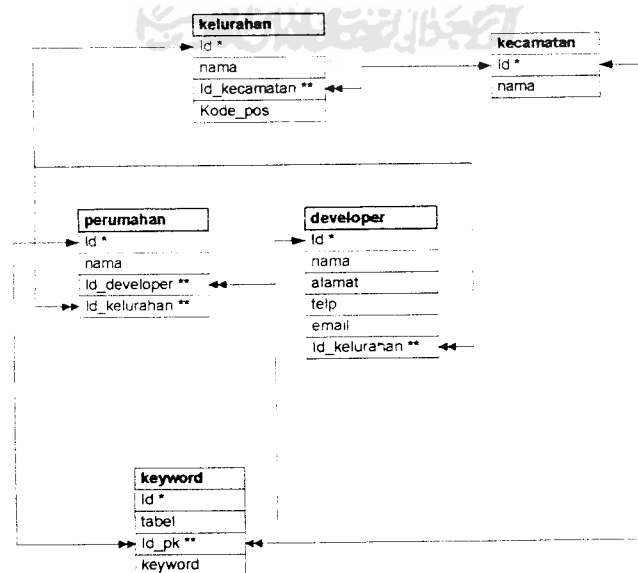
Primary key : id

Foreign key : id\_pk

Fungsi : menyimpan *keyword* yang nantinya digunakan untuk pencarian kecamatan, perumahan, dan *developer*

No.	Nama field	Tipe	Ukuran	Keterangan
1.	Id	INTEGER AUTO INCREMENT	11	Primary key
2.	Tabel	ENUM('kecamatan', 'perumahan', 'developer')	15	Tabel mana yang menjadi tabel induk
3.	Id_pk	INTEGER	11	Foreign key yang digunakan untuk menghubungkan dengan tabel induk
4.	Keyword	VARCHAR	60	Keyword yang digunakan sebagai basis pencarian

## 4.4.2 Relasi Antar Tabel



Gambar 11. Relasi antar tabel

Keterangan :

1. \* : *Primary Key*, \*\* : *Foreign Key*
2. → : *one*, →→ : *many*

Dari relasi antar tabel yang diperlihatkan dalam gambar 12 tabel *keyword* mengacu ke hampir semua tabel (tabel kecamatan, kelurahan, developer dan perumahan) melalui *field* yang bernama *id\_pk* yang dimilikinya. Juga terlihat bahwa tabel kecamatan dapat disebut sebagai *master table*, ini dikarenakan hampir semua tabel yang saling berrelasi akan berujung pada tabel ini. Sebagai gambaran, tabel *developer* mengacu ke tabel *kelurahan* melalui *field* *id\_kelurahan*, demikian juga tabel *perumahan* yang juga mengacu ke tabel *kelurahan* melalui *field* *id\_kelurahan* yang dimilikinya. Bila ditelusuri lebih jauh, tabel *kelurahan* mengacu ke tabel *kecamatan* melalui *field* *id\_kecamatan* yang dimilikinya.

#### 4.5 Rincian Prosedur – Prosedur

Berikut ini dijelaskan rincian – rincian prosedur yang digunakan dalam perancangan sistem yang akan dibangun, untuk memudahkan pemahaman tentang penggunaan aplikasi yang benar. Dengan terbangunnya pemahaman tentang penggunaan aplikasi yang benar, maka akan meminimalisasi kompleksitas perancangan *error management* (pengelolaan kesalahan) jika *user* misalkan salah memasukkan data, atau ketika secara tidak sengaja *user* mengklik tombol ‘Hapus’.

#### 4.5.1 Log-In

Dilakukan pencocokan nama *user* dan *password* dengan *password* yang terenkripsi dan tersimpan dalam *database*. Fungsi enkripsi *password* menggunakan fungsi *built-in* dalam MySQL Server, yaitu `PASSWORD()`. Jika sebuah *string* dikirim sebagai parameter ke fungsi tersebut, maka fungsi tersebut akan menghasilkan *string* yang telah terenkripsi sepanjang 41 karakter. Sebagai contoh, jika *password*-nya adalah 'andi', maka setelah pemanggilan fungsi

`PASSWORD('andi')` akan menghasilkan *string*:  
\*8D3280FA72CBF650764229A5248B89D0FD8A86A3.

#### 4.5.2 Input Data

Pemasukan data dilakukan oleh operator pada antarmuka berbasis *web* yang meliputi data :

1. Data kecamatan beserta *keyword*-nya
2. Data kelurahan beserta *keyword*-nya
3. Data perumahan beserta *keyword*-nya
4. Data *developer* beserta *keyword*-nya

Jika semua data telah dimasukkan dan sesuai dengan urutan, maka calon konsumen perumahan dapat melakukan pencarian lokasi perumahan berdasarkan lokasi yang dikehendakinya. Sebagai contoh, data perumahan membutuhkan data *developer* sebagai pihak yang membangun, data *developer* membutuhkan data kelurahan untuk mengetahui lokasi kantornya, dan data kelurahan membutuhkan data kecamatan sebagai daerah induk.

### 4.5.3 Penyajian Informasi

Sajian informasi yang berbeda untuk masing – masing *user* sesuai dengan jenis dan kepentingan *user*. Dalam perspektif aplikasi, *user* dibagi menjadi dua golongan, yaitu operator dan calon konsumen. Operator adalah *user* yang bertanggung jawab untuk mengelola data dengan benar agar dapat diakses dengan mudah oleh calon konsumen perumahan.

#### 1. Operator

Informasi mengenai proses masukan data, jika ada kegagalan dalam proses masukan, maka akan ditampilkan pesan kesalahan sehingga data yang tidak lengkap atau memiliki kesalahan tidak akan disimpan ke dalam basis data. Kesalahan masukan data dapat berakibat pada kesalahan penerimaan informasi yang digunakan oleh calon konsumen perumahan.

#### 2. Calon konsumen

Informasi yang dapat diterima oleh calon konsumen adalah hasil pencarian berdasarkan kata kunci (*keyword*) yang dimasukkan melalui antarmuka pencarian data yang berbasis *web*. Informasi tersebut berasal dari data-data yang dikelola oleh operator dan disampaikan melalui antarmuka yang juga berbasis *web*.

Informasi-informasi tersebut adalah:

- Kecamatan
- Kelurahan
- Perumahan
- Developer

## 4.6 Perancangan Antarmuka Berbasis Web

Sesuai dengan metode penyajian informasi yang sebelumnya telah diuraikan, perancangan antarmuka berbasis *web* terbagi menjadi dua, yaitu untuk operator dan untuk calon konsumen. Operator dapat mengakses seluruh fungsionalitas yang terdapat dalam aplikasi. Tetapi sebelum mengakses seluruh fungsionalitas yang ada, operator diwajibkan melakukan verifikasi nama dan *password* melalui halaman *Log-In* yang telah disediakan. Sedangkan calon konsumen tidak perlu melalui halaman *Log-In* untuk dapat menggunakan aplikasi, meskipun demikian, calon konsumen hanya dapat mengakses halaman pencarian, karena hanya fungsionalitas inilah yang sangat dibutuhkan oleh konsumen.

### 4.6.1 Rancangan *Layout* Keseluruhan

Memberikan gambaran umum tampilan keseluruhan dari antarmuka berbasis *web* yang didalamnya terbagi menjadi navigasi, *header*, *footer*, dan isi. Navigasi dalam tampilan keseluruhan ini memiliki 5 *link* utama, yaitu *link* ke Data Kecamatan, Data Kelurahan, Data Developer, Data Perumahan dan Pencarian data. Tepat berada di atas bagian navigasi, terdapat tombol yang digunakan untuk *Logout*, atau keluar dari sistem. Dengan alasan keamanan, setelah operator selesai menggunakan aplikasi diwajibkan untuk melakukan *logout*, dan setelah itu dapat menutup *web browser*. Ini dikarenakan pada saat *login*, informasi-informasi seperti *user ID* disimpan dalam *cache*, dan ketika *web browser* ditutup, *cache* tersebut tidak akan terhapus secara otomatis. Inilah gunanya proses *logout*, yang berfungsi untuk menghapus *cache* dan akan mengarahkan kembali (*redirect*) secara otomatis ke halaman *login*.



**Sistem Informasi Pencarian Lokasi Perumahan Berbasis Web**

Selamat datang, Administrator

Gambar 12. Halaman utama

#### 4.6.2 Rancangan Log-In

Digunakan untuk *login* ke sistem, dirancang dengan 2 buah kotak teks, sebuah tombol untuk *login* dan sebuah tombol untuk *me-reset* kotak teks, dengan rancangannya sebagai berikut :

**Login**

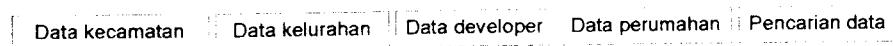
Nama pengguna

password

Gambar 13. Rancangan Log-In

#### 4.6.3 Navigasi

Navigasi berfungsi sebagai menu untuk mengakses fasilitas-fasilitas dalam sistem, antara lain: data kecamatan, data kelurahan, data *developer*, data perumahan, dan pencarian data.



Gambar 14. Rancangan Navigasi Umum

#### 4.6.4 Input Data Kecamatan

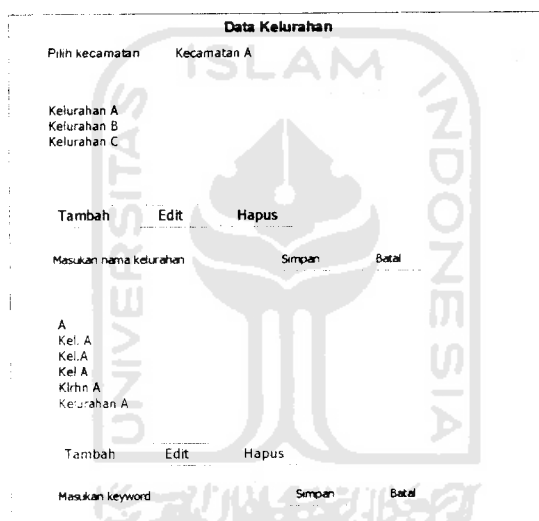
Rancangan untuk menampilkan data kecamatan, yang terdiri dari nama kecamatan, daftar *keyword*, dan kotak masukan data kecamatan. Tombol ‘Tambah’, ‘Edit’ dan ‘Hapus’ digunakan untuk mengelola data. Ketika tombol ‘Tambah’, ‘Edit’ dan ‘Hapus’ digunakan untuk mengelola data. Ketika tombol ‘Tambah’ dan ‘Edit’ diklik, fokus kontrol akan berpindah ke kotak masukan nama kelurahan yang berada di bawahnya. Tombol ‘Simpan’ digunakan untuk menyimpan data masukan dan tombol ‘Batal’ digunakan untuk membatalkan masukan.

The image shows two views of the 'Data Kecamatan' management interface. The top view is for entering village names, featuring a list of 'Kecamatan A', 'Kecamatan B', and 'Kecamatan C', followed by 'Tambah', 'Edit', and 'Hapus' buttons. Below this is a text input field labeled 'Masukan nama kelurahan' and 'Simpan' and 'Batal' buttons. The bottom view is for entering keywords, featuring a list of keyword examples: 'A', 'Kec. A', 'Kec.A', 'Kec A', 'Kcmtn A', and 'Kecamatan A', followed by 'Tambah', 'Edit', and 'Hapus' buttons. Below this is a text input field labeled 'Masukan keyword' and 'Simpan' and 'Batal' buttons.

Gambar 15. Rancangan Pengelolaan Data Kecamatan

#### 4.6.5 Input Data Kelurahan

Rancangan untuk menampilkan dan mengelola data kelurahan, beserta *keyword*. Layout rancangan ini sangat mirip dengan rancangan Data Kecamatan, hanya saja tepat berada di atas daftar kelurahan, terdapat *combo box* yang digunakan untuk menyaring data kelurahan berdasarkan kecamatan. Misalkan dalam *combo box* tersebut terpilih entri 'Kecamatan A', maka yang muncul dalam daftar kelurahan adalah kelurahan-kelurahan yang ada di kecamatan tersebut.



Gambar 16. Rancangan Pengelolaan Data Kelurahan

#### 4.6.6 Data Developer

Rancangan untuk mengelola data *developer*. Layout rancangan ini sedikit lebih kompleks dibandingkan dengan rancangan data kecamatan dan kelurahan. Bagian teratas terdapat dua buah *combo box* yang dapat digunakan untuk menyaring data *developer* berdasarkan kecamatan dan kelurahan. Dan pada bagian masukan data, juga terdapat dua buah *combo box* yang masing-masing menampilkan daftar kecamatan dan kelurahan, yang digunakan untuk menentukan

lokasi *developer* tersebut berada. Selain itu juga terdapat 4 kotak masukan yang masing-masing digunakan untuk memasukkan nama *developer*, alamat, telepon dan *e-mail*.

**Data Developer**

Pilih kecamatan      Kecamatan A  
Pilih kelurahan      Kelurahan B

Nama Developer	Alamat	Telp	E-mail

Tambah      Edit      Hapus

Pilih kecamatan      Kecamatan A      Pilih kelurahan      Kelurahan B

Masukan nama developer      Masukan e-mail

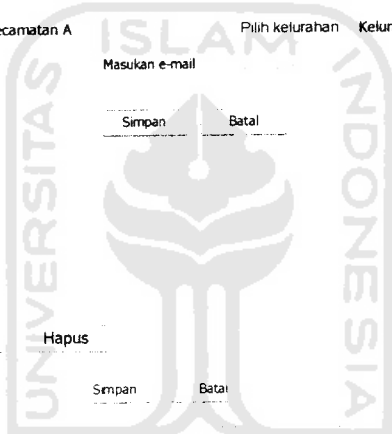
Masukan alamat

Masukan telp

P.T. AAAA  
P.T AAAAA  
PT. AAAAA  
PT AAAAA  
AAAA PT  
AAAA, PT  
AAAAAA

Tambah      Edit      Hapus

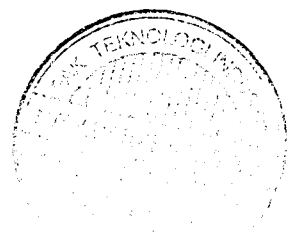
Masukan keyword      Simpan      Batal



Gambar 17. Rancangan Pengelolaan Data Developer

#### 4.6.7 Data Perumahan

Rancangan untuk mengelola data perumahan yang dibangun oleh *developer*. Hampir sama dengan rancangan data *developer*, rancangan perumahan ini juga memiliki penyaringan data berdasarkan kecamatan dan kelurahan. Selain itu juga terdapat penyaringan berdasarkan *developer*, yang diwakilkan melalui kontrol *combo box* yang tepat berada di sisi paling atas. Pada bagian masukan data, terdapat tiga buah *combo box* yang digunakan untuk memasukkan data *developer*,



data kecamatan dan data kelurahan. Serta sebuah *edit box* yang digunakan untuk memasukkan nama perumahan tersebut.

**Data Perumahan**

Developer    Developer A

Pilih kecamatan    Kecamatan A

Pilih kelurahan    Kelurahan B

Nama Perumahan	Kelurahan

Tambah    Edit    Hapus

---

Developer    Developer A

Masukan nama perumahan

Pilih kecamatan    Kecamatan A    Pilih kelurahan    Kelurahan B

Simpan    Batal

P.T. AAAA  
P.T AAAAA  
PT. AAAAA  
PT AAAAA  
AAAAA PT  
AAAAA, PT  
AAAAAAA

Tambah    Edit    Hapus

Masukan keyword    Simpan    Batal

Gambar 18. Rancangan Pengelolaan Data Perumahan

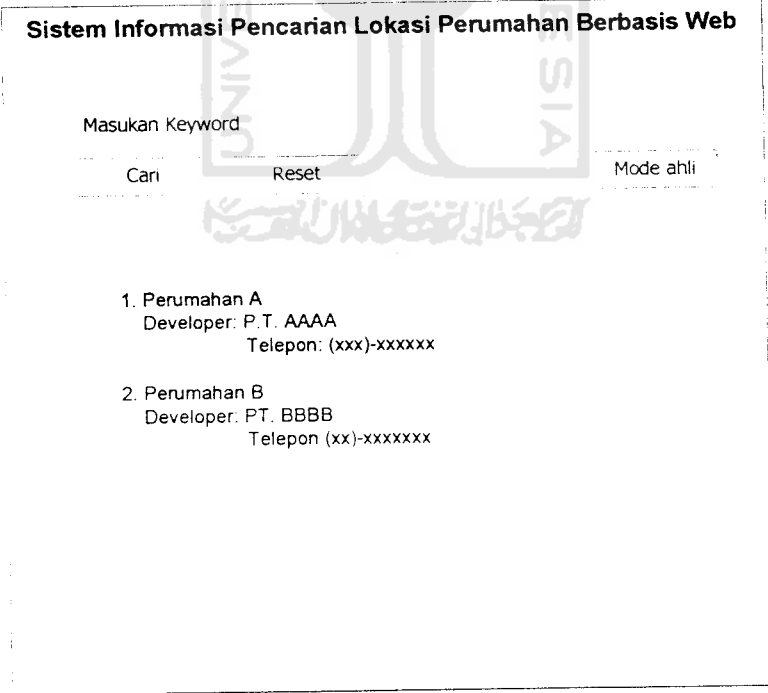
#### 4.7 Perancangan Antarmuka untuk Calon Pelanggan

Antarmuka yang digunakan untuk calon pelanggan hampir sama dengan antarmuka operator, yaitu sama-sama menggunakan antarmuka berbasis *web*. Seperti yang sudah diuraikan sebelumnya, antarmuka untuk calon pelanggan

hanyalah antarmuka yang digunakan untuk pencarian informasi. Meskipun demikian, operator juga dapat mengakses antarmuka pencarian.

#### 4.7.1. Pencarian Sederhana

5. Pencarian sederhana untuk calon pengguna mengambil *layout* dari *search engine* terkenal, yaitu Google. Keunggulan *layout* Google adalah kesederhanaan, dengan cara meminimalisir penggunaan citra (*image*) sehingga mempercepat proses *loading* halaman. Masukan yang disediakan hanyalah sebuah *edit box* yang digunakan untuk memasukkan kata kunci pencarian, serta tiga buah tombol: 'Cari' untuk mengaktifkan pencarian, 'Reset' untuk menghapus kata kunci dari *edit box*, dan 'Mode Ahli' untuk berpindah ke tampilan pencarian mode ahli.



**Sistem Informasi Pencarian Lokasi Perumahan Berbasis Web**

Masukan Keyword

Cari      Reset      Mode ahli

1. Perumahan A  
Developer: P.T. AAAA  
Telepon: (xxx)-xxxxxx
2. Perumahan B  
Developer: PT. BBBB  
Telepon (xx)-xxxxxx

Gambar 19. Pencarian data mode sederhana

### 5.7.1. Pencarian Mode Ahli

Pada pencarian data mode sederhana yang rancangannya ditampilkan dalam gambar 20, terdapat tombol 'Mode ahli' yang akan menampilkan halaman pencarian data mode ahli (*advanced*) yang rancangannya ditampilkan dalam gambar 21. Dalam halaman mode ahli, calon pelanggan dapat merinci *keyword* yang dimasukkannya, apakah kategori *keyword* kecamatan, kelurahan, *developer*, ataukah perumahan.

**Sistem Informasi Pencarian Lokasi Perumahan Berbasis Web  
(Mode Ahli)**

Masukan Keyword

Jenis keyword

- Kecamatan
- Kelurahan
- Developer
- Perumahan

Cari      Reset      Mode sederhana

1. Perumahan A  
Developer: P.T. AAAA  
Telepon: (xxx)-xxxxxx
2. Perumahan B  
Developer: PT. BBBB  
Telepon (xx)-xxxxxxx

Gambar 20. Pencarian data mode ahli

## **BAB V**

### **IMPLEMENTASI**

#### **5.1 Batasan Implementasi**

##### **5.1.1 Asumsi yang dipakai**

Dalam tahap implementasi digunakan asumsi – asumsi sebagai berikut :

1. Nama *database* yang digunakan adalah *sms\_nlp*, nama *user* basis data adalah 'admin\_sms\_nlp'.
2. Menggunakan sistem operasi Windows dan Apache Web Server minimal versi 2.0 untuk mengelola dokumen *web*.
3. *Web browser* yang digunakan untuk mengakses aplikasi setidaknya Internet Explorer 5.5, atau Netscape 6 atau Firefox 1.1, atau Opera 8.5.

##### **5.1.2 Lingkungan Pengembangan**

Lingkungan pengembangan untuk antarmuka dan desain menggunakan lingkungan Windows.

##### **5.1.3 Bahasa yang dipakai**

Menggunakan bahasa pemrograman PHP 5 untuk mengembangkan aplikasi.

##### **5.1.4 Perangkat Lunak yang digunakan**

Untuk mewujudkan sistem yang dibangun dibutuhkan beberapa perangkat lunak pendukung, diantaranya :



### 1. Sistem Operasi

Sistem Operasi yang digunakan untuk pengembangan aplikasi menggunakan sistem operasi Windows XP *Service Pack 2*.

### 2. *Web Server*

Web server yang digunakan untuk mengelola dokumen *web* adalah Apache Web Server 2.2.4.

### 3. *Interpreter*

Untuk menerjemahkan dokumen-dokumen *web* yang ditulis menggunakan bahasa pemrograman PHP 5, maka digunakan *interpreter* PHP 5.2.1 yang dikonfigurasi sebagai *module* dalam Apache Web Server.

### 4. Server Basisdata

Server Basis Data menggunakan MySQL Server 5.

### 5. Aplikasi untuk Pengembangan Perangkat Lunak

Beberapa aplikasi untuk pengembangan yang digunakan antara lain adalah:

#### a. Macromedia Dreamweaver 8

Digunakan untuk desain aplikasi, mulai dari desain antarmuka, sampai ke pengeditan *source code*.

#### b. Adobe Photoshop 7

Digunakan untuk mendukung desain ikon-ikon yang digunakan dalam aplikasi.

#### c. MySQL-Front

Digunakan untuk manajemen basisdata.

### 5.1.5 Perangkat Keras yang Digunakan

Selama proses pengembangan, aplikasi dirancang di atas perangkat keras dengan spesifikasi sebagai berikut:

1. Prosesor AMD Sempron 2500+ Socket 754
2. Mainboard Albatron K8M800
3. Memori 512MB
4. CD-RW Asus 52x32x52
5. Hardisk Seagate Barracuda 160GB SATA

Sedangkan aplikasi yang digunakan oleh operator dapat dijalankan pada perangkat keras setidaknya memiliki spesifikasi sebagai berikut:

1. Prosesor Intel Pentium III 450 MHz
2. Memori 128MB
3. Sisa ruang kosong pada hardisk setidaknya 80 MB untuk menyimpan *browsing cache*.

### 5.2 Implementasi

Secara garis besar implementasi sistem sebagai berikut :

1. Instalasi Apache Web Server 2.2.4
2. Instalasi interpreter PHP 5.2.1
3. Membangun Database Server.
4. Membangun bagian yang mendukung sistem aplikasi utama.

### 5.2.1 Konfigurasi Komputer dan Instalasi Sistem

Pada tahap awal yaitu mengkonfigurasi perangkat keras yang dibutuhkan. Dilanjutkan dengan instalasi Sistem Operasi windows XP dengan pilihan instalasi standar. Selanjutnya dilakukan instalasi *software* pendukung yang dibutuhkan untuk membangun sistem.

Komputer diberi konfigurasi sesuai dengan pada bagian asumsi. Setelah instalasi selesai dilakukan *update* sistem dan *driver* secara keseluruhan.

### 5.2.2 Instalasi dan Konfigurasi Aplikasi Pendukung

#### 5.2.2.1 Database Server

Setelah sistem operasi terinstal dan berjalan dengan baik, proses berikutnya dilanjutkan dengan instalasi aplikasi pendukung, Diantaranya adalah aplikasi Basis Data dan aplikasi *web server*.

Untuk instalasi *server* yang digunakan selama perancangan urutannya sebagai berikut:

1. Instalasi Apache Web Server 2.2.4
2. Instalasi *interpreter* PHP 5.2.1 yang dikonfigurasi sebagai *module* dalam Apache Web Server
3. Instalasi MySQL Database Server 5.
4. Instalasi MySQL-Front.
5. Instalasi Macromedia Dreamweaver 8.

### 5.2.3 Pembangunan Aplikasi

#### 5.2.3.1 Struktur Antarmuka

Untuk memberikan struktur antarmuka yang fleksibel dan dinamis, citra yang digunakan sebagai ikon-ikon tombol dan *background* (GIF) diletakkan di dalam *folder images\*. Sedangkan *source code file* yang digunakan sebagai SSI (*Server Side Include*) diletakkan di dalam *folder include\*, dan *source code file* utama diletakkan di dalam *folder* utama (*main folder*). File CSS (*Cascading Style Sheet*) yang membantu dalam pembentukan layout halaman diletakkan di dalam *folder css\*. Struktur halaman utama menggunakan *row frameset*, dengan *frame* atas yang digunakan sebagai *frame* navigasi, dan *frame* bawah yang digunakan sebagai *main frame* (*frame* utama).

#### 5.2.3.2 Mekanisme Login

Mekanisme *login* dapat digambarkan sebagai berikut

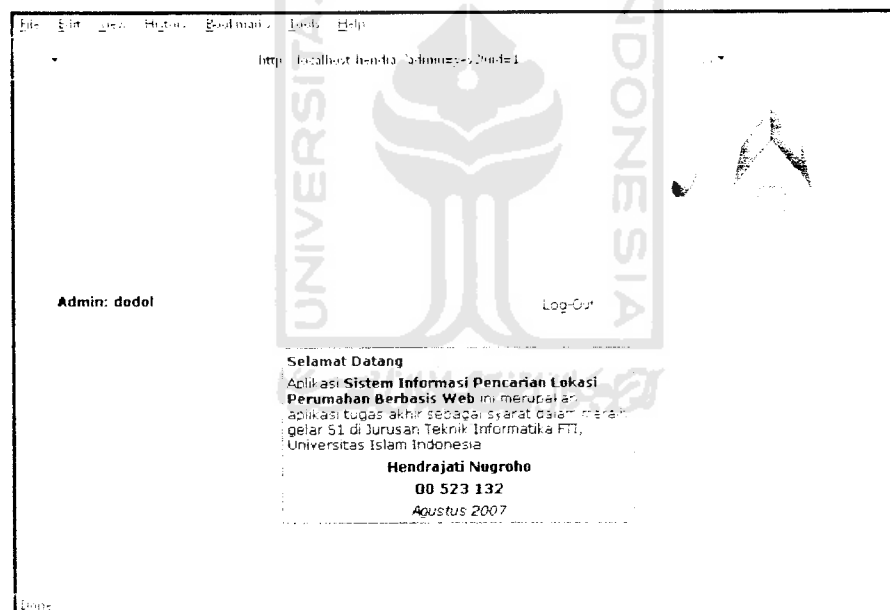
1. *User* menginputkan nama *user* dan *password* pada *login box*.
2. Form akan mengirimkan nama *user* dan *password* ke file `processlogin.php` yang akan melakukan verifikasi nama *user* dan *password*.
3. Dengan menggunakan fungsi dicocokkan dengan *user* dan *password* terenkripsi pada basis data.
4. Jika nama *user* dan *password* cocok, maka data *user* seperti nomor ID disimpan melalui mekanisme *session* yang dimiliki oleh PHP, dan *form* utama akan terbuka, jika tidak *login box* akan tetap muncul.

### 5.2.3.3 Add Edit dan Delete Data

Mekanisme *add edit* dan *delete* data dengan menggunakan masing-masing fungsi yang terdapat dalam *form*, untuk melakukan operasi data tersebut, hal ini untuk memudahkan pengembangan ulang dan koreksi kesalahan dan memudahkan pengembangan lanjut sistem.

### 5.2.3.4 Preview Keseluruhan

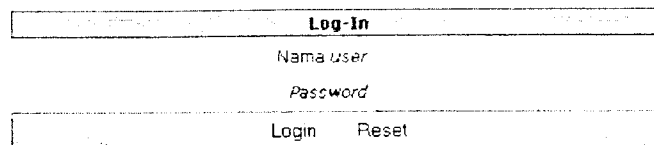
Tampilan keseluruhan antarmuka aplikasi berbasis Java dengan contoh sedang menampilkan *form* utama yang telah melewati proses *login*.



Gambar 21. Form utama

### 5.2.3.5 Kotak Login

Sebagai tempat untuk melakukan *login* ke dalam sistem, jika *user* belum *login* maka tidak dapat mengakses keseluruhan sistem kecuali navigasi dan *login box* itu sendiri.



The image shows a login interface. At the top, there is a header box labeled "Log-In". Below it are two input fields: "Nama user" and "Password". At the bottom, there are two buttons: "Login" and "Reset".

Gambar 22. Login box

### 5.2.3.6 Navigasi

Digunakan sebagai menu atau navigasi untuk sistem, termasuk di dalamnya yaitu Data Kecamatan, Data Kelurahan, Data Developer, Data Perumahan, Pencarian.

Admin: dodol

Log-Out

Gambar 23. Navigasi Utama

### 5.2.3.7 Form Data Kecamatan

Form ini dapat diakses jika tombol 'Data Kecamatan' pada navigasi utama diklik. Adapun bentuk dari form Data Kecamatan dapat dilihat di gambar 25.

Data Kecamatan			
No	Nama Kecamatan		
1.	JETIS	Edit	Hapus
2.	KALASAN	Edit	Hapus
3.	MERGANGSAH	Edit	Hapus

Tambah Data Kecamatan

Keyword
KECAMATAN JETIS KCMTN JETIS KCMTN JETIS KEC JETIS KECJETIS

Tambah Keyword    Edit Keyword    Hapus Keyword

Gambar 24. Data Kecamatan

*Group box* 'Keyword' yang terletak di bagian bawah digunakan untuk mengelola *keyword* yang dimiliki oleh kecamatan tersebut. *Keyword* ini nantinya digunakan dalam proses pencarian data. Untuk menambah data kecamatan, tombol 'Tambah Data Kecamatan' diklik, yang akan memunculkan kotak dialog seperti yang ditunjukkan dalam gambar 26.

Nama kecamatan pakualaman
Simpan    Batal

Gambar 25. Entri data kecamatan

Pada kotak dialog tersebut operator dapat mengisikan nama kecamatan. Hal yang sama juga terjadi kalau tombol 'Tambah Keyword' di *group box* 'Keyword'

diklik, akan memunculkan kotak input data seperti yang ditunjukkan dalam gambar 27.

Gambar 26. Entri data keyword

### 5.2.3.8 Form Data Kelurahan

Form yang berfungsi untuk mengelola data kelurahan ini dapat diakses melalui tombol 'Data Kelurahan' pada navigasi utama. Adapun bentuk dari form ini dapat dilihat di gambar 28.

No	Nama Kelurahan	Kode Pos		
1.	BUMIDO	55231	Edit	Hapus
2.	COKROKUSUMAN	55233	Edit	Hapus
3.	JETISHARJO	55235	Edit	Hapus

Keyword
KEL.BUMIDO
KEL.BUMIDO
KELURAHAN BUMIDO
KLRHN BUMIDO
KLRHN BUMIDO

Gambar 27. Form Data Kelurahan

Sama seperti form Data Kecamatan, form Data Kelurahan juga memiliki group box 'Keyword', untuk mengelola keyword yang dimiliki suatu kelurahan.



Untuk menambah data kelurahan, klik tombol ‘Tambah Kelurahan’ yang terletak di kotak *table* Daftar Kelurahan, yang akan menampilkan kotak Entri Data Kelurahan, seperti yang ditampilkan dalam gambar 29.

Nama Kelurahan: JETISHARJO
Kode Pos: 55233
Simpan    Batel

Gambar 28. Entri Data Kelurahan

### 5.2.3.9 Form Data Developer

*Form* yang berfungsi untuk mengelola data *developer* ini dapat diakses melalui tombol ‘Developer’ pada navigasi umum. Adapun bentuk *form* ini dapat dilihat pada gambar 30.

Data Developer					
Pilih Kecamatan --Pilih salah satu--		Pilih Kelurahan --Pilih salah satu--			
No	Kecamatan	Kelurahan	Developer	Alamat	Telp
Refresh					
1.	JETIS	COKROKUSUMAN	P.T. MERAPI ARSITAGRAHA	JL. A. M. SANGAJI 64	0274-543914
					Edt    Hapus
Tambah Data Developer					

Keyword
P.T. MERAPI ARSITAGRAHA
P.T. MERAPI ARSITAGRAHA
P.T. MERAPI ARSITAGRAHA
PT. MERAPI ARSITAGRAHA
PT. MERAPI ARSITAGRAHA
PT. MERAPI ARSITAGRAHA

Tambah Keyword    Edit Keyword    Hapus Keyword

Gambar 29. Form Data Developer

Sama seperti *form* Data Kecamatan, *form* Data Developer juga memiliki *group box* 'Keyword', untuk mengelola *keyword* yang dimiliki suatu *developer*. Untuk menambah data *developer*, klik tombol 'Tambah' yang terletak di kotak *table* 'Daftar Developer', yang akan menampilkan kotak Entri Data Developer, seperti yang ditampilkan dalam gambar 31.

Gambar 30. *Entri Data Developer*

Dalam *form* ini, kotak 'Nama Developer' digunakan untuk memasukkan nama *developer*, sedangkan *combo box* 'Badan Usaha' digunakan untuk menentukan jenis badan usaha dari *developer* tersebut. Kotak 'Alamat', 'Telp' dan 'E-mail' digunakan untuk memasukkan alamat, nomor telepon dan alamat *e-mail* milik *developer*. *Combo box* 'Kecamatan' dan 'Kelurahan' digunakan untuk menentukan lokasi *developer* tersebut berkantor.

#### 5.2.3.10 *Form* Data Perumahan

*Form* yang berfungsi untuk mengelola data perumahan ini dapat diakses melalui tombol 'Perumahan' pada navigasi umum. Adapun bentuk *form* ini dapat dilihat pada gambar 32.

**Data Perumahan**

Pilih Kecamatan --Pilih salah satu--      Pilih Kelurahan --Pilih salah satu--  
 Pilih Developer --Pilih salah satu--

No	Kecamatan	Kelurahan	Developer	Perumahan	
Refresh					
1.	JETIS	BUMIJO	P.T. MERAPI ARSITAGRAHA	INDAH PERMAI	Edit Hapus
2.	JETIS	COKROKUSUMAN	P.T. MERAPI ARSITAGRAHA	ANU LAGI	Edit Hapus

Tambah Data Perumahan

**Keyword**

KOMPL.INDAH PERMAI  
 KOMPL.INDAH PERMAI  
 KOMPL.INDAH PERMAI  
 KOMPLEKS INDAH PERMAI  
 PERUMAHAN INDAH PERMAI  
 PERUMAHAN INDAH PERMAI  
 PRMHN INDAH PERMAI  
 PRMHN INDAH PERMAI

Tambah Keyword    Edit Keyword    Hapus Keyword

Gambar 31. Form Data Perumahan

Sama seperti *form* Data Kecamatan, *form* Data Perumahan juga memiliki *group box* Keyword, untuk mengelola *keyword* yang dimiliki perumahan. Untuk menambah data perumahan, klik tombol ‘Tambah Perumahan’ yang terletak di bawah Daftar Perumahan, yang akan menampilkan kotak Entri Data Perumahan, seperti yang ditampilkan dalam gambar 33.

Developer P.T. MERAPI ARSITAGRAHA ▼  
 Nama INDAH PERMAI  
 Kecamatan --Pilih salah satu-- ▼  
 Kelurahan BUMIJO ▼  
 Simpan    Batal

Gambar 32. Entri Data Perumahan

Dalam *form* ini, kotak 'Nama Perumahan' digunakan untuk memasukkan nama perumahan. *Combo box* 'Kecamatan' dan 'Kelurahan' digunakan untuk menentukan lokasi perumahan tersebut.



## **BAB VI**

### **ANALISIS KINERJA**

#### **6.1 Dokumentasi Penggunaan Sistem**

Setelah selesai implementasi, tahap selanjutnya adalah dilakukan *testing* atau penggunaan secara nyata terhadap sistem yang telah berjalan, beberapa tes diantaranya adalah dengan menjalankan berdasarkan prosedur normal atau benar dan dijalankan dengan prosedur tidak benar atau salah. Prosedur tidak normal digunakan untuk melihat apakah sistem dapat menolak dan mengoreksi kesalahan jika seorang *user* menjalankannya dengan kesalahan.

#### **6.2 Analisis Kinerja**

##### **6.2.1 Aplikasi Operator**

Pengujian ke dalam sistem berbasis *web* untuk pengelolaan data yang dilakukan oleh operator.

##### **6.2.1.1 Log-In ke Sistem**

###### **6.2.1.1.1 Prosedur Normal**

Jika *user* menginputkan nama *user* dan *password* benar melalui *login box*, seperti yang ditampilkan pada gambar 23, maka sesi dimulai dan akan tampil navigasi untuk operator dengan dilengkapi tombol 'Logout', seperti yang ditampilkan pada gambar 24.

### 6.2.1.1.2 Prosedur Tidak Normal

Jika *user* menginputkan nama *user* dan *password* salah maka sesi tidak akan dijalankan dan akan tampil peringatan.

Maaf, sepertinya nama dan password-nya tidak cocok. Coba lagi

**Log-In**

Nama user |

password

Login    Reset

Gambar 33. Peringatan untuk Kesalahan Login

### 6.2.1.2 Input Data

Untuk dapat mengakses bagian pemasukan data, dibutuhkan *login* ke sistem terlebih dahulu dan harus sebagai operator. Setelah *user* administrator *login* ke sistem dan mengakses sistem maka yang tampil adalah halaman untuk operator yang diawali dengan navigasi untuk operator.

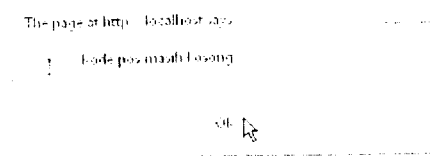
#### 6.2.1.2.1 Prosedur Normal

Jika *user* sudah *login* dan sebagai operator dan benar dalam pengisian data, seperti tidak adanya duplikasi data maka akan ditampilkan pesan bahwa pengisian data berhasil, pengujian dilakukan pada pemasukan data kelurahan. Proses pengisian data kelurahan sudah ditunjukkan pada gambar 29, dan data yang berhasil tersimpan sudah ditunjukkan pada gambar 28.

#### 6.2.1.2.2 Prosedur Tidak Normal

Jika *user* sudah *login* dan sebagai operator tetapi salah dalam melakukan pengisian data, seperti terjadinya *field* yang masih kosong maka akan ditampilkan

pesan bahwa pengisian data gagal, pengujian dilakukan pada pemasukan data kelurahan.



Gambar 34. Peringatan Kesalahan Input Data

### 6.2.1.3 Edit Data

Sama seperti halnya dalam penambahan data, bagian edit data juga merupakan hak dari operator.

#### 6.2.1.3.1 Prosedur Normal

Jika perubahan data benar maka ditampilkan pesan perubahan data berhasil tersimpan dalam daftar. Dalam hal ini contoh edit data dilakukan pada data kelurahan, yaitu kelurahan Tirtonirmolo yang terdapat dalam kecamatan Kasihan, Bantul. Seharusnya nama kelurahan tersebut adalah Tirtonirmolo, bukan Tirto Nirmolo, seperti yang diperlihatkan dalam gambar 36.

No	Nama Kelurahan	Kode Pos		
Refresh				
1.	BANGUNITWO	55184	Edit	Hapus
2.	RESEKIPARJO	55182	Edit	Hapus
3.	TAMANTIRTO	55183	Edit	Hapus
4.	TIRTO NIRMOLLO	55181	Edit	Hapus

Gambar 35. Data kelurahan sebelum diedit

TIRTO NIRMOLLO 55181 Edit Hapus

Tambah Data Kelurahan

Nama Kelurahan TIRTO NIRMOLLO  
Kode Pos 55181

Simpan Batal

Gambar 36. Kelurahan akan dirubah

3.	TAMANTIRTO	55183	Edit	Hapus
4.	TIRTONIRMOLO	55181	Edit	Hapus

*Gambar 37. Kelurahan setelah berubah*

#### 6.2.1.3.2 Prosedur Tidak Normal

Jika perubahan data salah maka ditampilkan pesan peringatan dan data tidak akan tersimpan dalam daftar. Dalam hal ini contoh edit data dilakukan pada data kelurahan, dan akan menampilkan pesan kesalahan seperti yang ditunjukkan pada gambar 35.

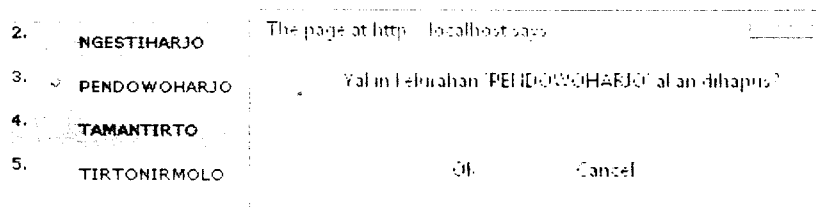
#### 6.2.1.4 Penghapusan Data

Bagian penghapusan data juga sama dengan bagian penambahan data yaitu hak dari operator. Pengujian menggunakan penghapusan data kelurahan.

Untuk mekanisme penghapusan data digunakan peringatan sehingga kesalahan dapat dihindari. Dalam contoh ini, ada kesalahan pemasukan data, yaitu kelurahan Pendowoharjo yang seharusnya masuk ke kecamatan Sewon, tetapi ternyata dimasukkan ke kecamatan Kasihan. Sehingga data kelurahan Pendowoharjo yang ada di kecamatan Kasihan akan dihapus.

3.	PENDOWOHARJO	55185	Edit	Hapus
----	--------------	-------	------	-------

*Gambar 38. Kelurahan Pendowoharjo yang salah akan dihapus*



*Gambar 39. Peringatan bahwa data akan dihapus*



No	Nama Kelurahan	Kode Pos		
Refresh				
1.	BANGUNJIWO	55184	Edit	Hapus
2.	NGESTIHARJO	55182	Edit	Hapus
3.	TAMANTIRTO	55183	Edit	Hapus
4.	TIRTO NIRMOLO	55181	Edit	Hapus

Gambar 40. Kelurahan Pendowoharjo yang telah dihapus

### 6.2.2 Aplikasi untuk Calon Konsumen

Agar calon konsumen dapat mengakses informasi perumahan, calon konsumen cukup mengakses aplikasi melalui *web browser* favoritnya. Agar aspek keamanan data dapat ditingkatkan, maka calon konsumen tidak boleh tahu URL (*Unified Resource Locator*) untuk mengakses mode operator. Ketika aplikasi diakses melalui URL biasa, maka yang tampak hanyalah halaman pencarian seperti yang ditunjukkan dalam gambar 42, tanpa halaman *log-in*.

Masukkan ke-4 koordinat

Uraian barang

100000

Gambar 41. Halaman pencarian

Halaman pencarian ini sebenarnya adalah halaman pencarian sederhana, yaitu pencarian data yang tidak memperdulikan lokasi datanya, apakah berada di tabel kecamatan, kelurahan ataupun *developer*. Pencarian sederhana ini juga

menemukan semua *string* yang mirip dengan *keyword* yang dimasukkan. Misalkan jika dimasukkan 'jetis', maka hasilnya dapat memunculkan 'Jetis' (nama kecamatan) dan 'Jetisharjo' (nama kelurahan). Dengan demikian, hasil pencarian sederhana ini cakupannya cukup luas.

Untuk mempersempit hasil pencarian, dapat digunakan mode ahli, dengan cara mengklik tombol 'Mode Ahli' (*advanced mode*), yang akan menampilkan halaman pencarian yang lebih detail.

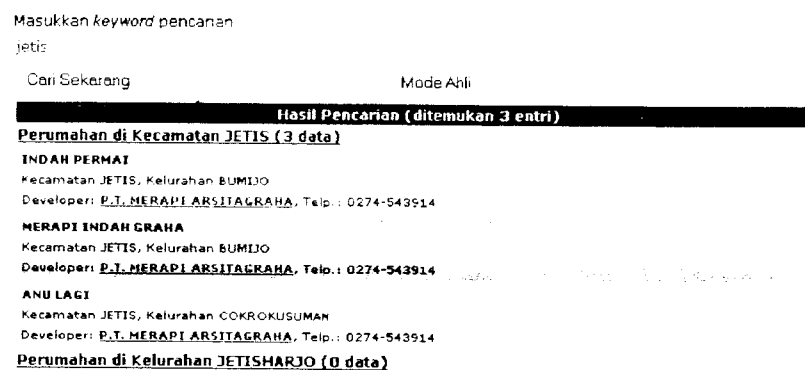


Gambar 42. Pencarian mode ahli

### 6.2.2.1. Pencarian Sederhana

#### 6.2.2.1.1. Prosedur Normal

Sebagai contoh kasus *keyword* yang akan digunakan adalah 'jetis', dan hasil pencarian dapat dilihat pada gambar 44.



Gambar 43. Hasil pencarian sederhana

Seperti yang terlihat dalam contoh hasil pencarian, sistem akan menemukan semua *keyword* yang memiliki kata 'jetis'. Jika yang ditemukan adalah kecamatan (Kecamatan Jetis), maka sistem akan mencari semua perumahan yang ada dalam kecamatan tersebut. Jika yang ditemukan adalah kelurahan (Kelurahan Jetisharjo), maka sistem juga akan mencari semua perumahan yang ada dalam kelurahan tersebut. Dalam contoh ini, kelurahan Jetisharjo tidak memiliki perumahan, karena itu dalam keterangannya ditampilkan '0 data'.

Jika ditemukan data perumahan, maka sistem akan menampilkannya dalam bentuk tiga baris, yaitu baris nama perumahan, baris lokasi (kecamatan dan kelurahan) dan baris *developer*. Nama *developer* ini juga berfungsi sebagai *hyperlink*, yang jika diklik akan menampilkan informasi yang lebih mendetail tentang *developer* tersebut.

Masukkan *keyword* pencarian  
merapi

Tipe *keyword* Developer Hasil harus sama persis dengan *keyword*

Cari Sekarang Mode Sederhana

---

**Detail Developer**

Detail informasi per developer  
**P.T. MERAPI ARSITAGRAHA**  
 Alamat: JL. A.M. SANGAJI 64  
 Kecamatan Jetis, Kelurahan Cokrokusuman  
 Telp.: 0274-543914

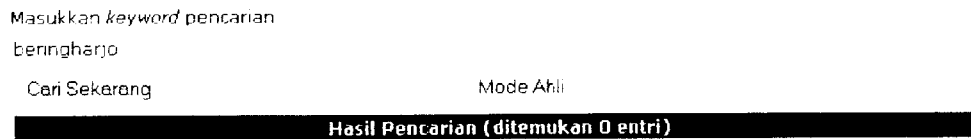
**Daftar Perumahan yang Dibangun**

Kecamatan	Kelurahan	Nama Perumahan
Jetis	Cokrokusuman	ANU LAGI
Jetis	Bumijo	INDAH PERMAI
Jetis	Bumijo	MERAPI INDAH GRAHA

Gambar 44. Detail informasi developer

#### 6.2.2.1.2. Prosedur Tidak Normal

Jika *keyword* yang digunakan dalam pencarian adalah 'beringharjo' yang merupakan nama pasar, bukan nama kecamatan maupun nama kelurahan, maka sistem akan menampilkan keterangan bahwa gagal menemukan data.



Gambar 45. Gagal menemukan data

## 6.2.2.2. Pencarian Mode Ahli

### 6.2.2.2.1. Prosedur Normal

Dalam pencarian mode ahli, ada sejumlah kontrol masukan (*input control*) yang perlu diperhatikan. Pertama, di bagian kiri bawah kotak *keyword* terdapat sebuah *combo box* 'Tipe keyword'. *Combo box* yang berisi item: 'Belum jelas', 'Kecamatan', 'Kelurahan', 'Developer' dan 'Perumahan' ini digunakan untuk menentukan kategori *keyword* tersebut, apakah termasuk kategori kecamatan, kelurahan, *developer* ataupun perumahan. Jika yang dipilih adalah 'Belum jelas' maka berarti *keyword* yang akan dicari belum diketahui dengan pasti kategorinya, atau dengan kata lain hampir sama dengan mode sederhana. Kontrol kedua yang perlu diperhatikan adalah *checkbox* yang terletak di sebelah kanan bawah kotak *keyword*. *Checkbox* ini digunakan untuk menentukan metode pencarian data, apakah data tersebut sama persis dengan *keyword* yang dimasukkan atau hanya mirip dengan *keyword* yang dimasukkan. Jika *checkbox* ini diberi tanda cek (diaktifkan), maka berarti data yang dicari harus sama persis dengan *keyword* yang dimasukkan.

Sebagai contoh, jika dimasukkan *keyword* 'merapi', maka akan ditemukan 'Merapi Indah Graha' (nama perumahan) dan 'P.T. Merapi Arsitagraha' (nama developer), seperti yang ditampilkan dalam gambar 47.

Masukkan *keyword* pencarian  
merapi

Tipe *keyword* Belum jelas ▾ Hasil harus sama persis dengan *keyword*

Cari Sekarang Mode Sederhana

**Hasil Pencarian (ditemukan 2 entri)**

**Perumahan (1 data)**  
**MERAPI INDAH GRAHA**  
 Kecamatan JETIS, Kelurahan BUMJO  
 Developer: **P.T. MERAPI ARSITAGRAHA**, Telp.: 0274-543914

**Developer (1 data)**  
**P.T. MERAPI ARSITAGRAHA**  
 Alamat: **JL. A.M. SANGAJI 64**  
 Kecamatan JETIS, Kelurahan COKROKUSUMAN  
 Telp.: **0274-543914**

Gambar 46. Pencarian sederhana

Untuk mempersempit pencarian, tipe *keyword* ditentukan sebagai ‘Developer’, yang akan menampilkan sebuah hasil pencarian, seperti yang ditampilkan dalam gambar 48.

Masukkan *keyword* pencarian  
merapi

Tipe *keyword* Developer ▾ Hasil harus sama persis dengan *keyword*

Cari Sekarang Mode Sederhana

**Hasil Pencarian (ditemukan 1 entri)**

**Developer (1 data)**  
**P.T. MERAPI ARSITAGRAHA**  
 Alamat: **JL. A.M. SANGAJI 64**  
 Kecamatan JETIS, Kelurahan COKROKUSUMAN  
 Telp.: **0274-543914**

Gambar 47. Pencarian mode ahli dengan menentukan tipe *keyword*

Jika *checkbox* yang berada di sebelah kanan ‘Tipe *keyword*’ diberi tanda cek (diaktifkan), maka berarti *keyword* yang dimasukkan harus persis sama dengan data yang akan dicari. Sebagai contoh dimasukkan *keyword* ‘merapi indah graha’ (nama perumahan), yang akan menampilkan hasil pencarian seperti yang ditampakkan dalam gambar 49. Jika tidak persis sama, maka data tidak akan ditemukan, seperti yang ditampilkan dalam gambar 46.

Masukkan *keyword* pencarian  
merapi indah graha

Tipe *keyword* Perumahan ▾ Hasil harus sama persis dengan *keyword*

Cari Sekarang Mode Sederhana

**Hasil Pencarian (ditemukan 1 entri)**

**Perumahan (1 data)**

**MERAPI INDAH GRAHA**  
Kecamatan JETIS, Kelurahan BUMIJO  
Developer: P.T. MERAPI ARSITAGRAHA, Telp.: 0274-543914

Gambar 48. Pencarian mode ahli dengan menentukan tipe *keyword* dan pilihan hasil pencarian yang persis sama dengan *keyword*

#### 6.2.2.2.2. Prosedur Tidak Normal

Sama seperti pencarian sederhana, dalam pencarian mode ahli akan menampilkan kegagalan pencarian jika *keyword* yang dipakai dalam pencarian ternyata belum dimasukkan ke dalam sistem oleh operator, yang akan menampilkan hasil pencarian seperti yang ditampilkan dalam gambar 46. Sistem juga akan mengalami kegagalan jika *checkbox* 'Hasil harus persis sama dengan *keyword*' diaktifkan. sedangkan *keyword* yang digunakan tidak sama persis dengan nama kecamatan, atau nama kelurahan, atau nama *developer* atau nama perumahan.

## BAB VII

### KESIMPULAN DAN SARAN

#### 7.1 Kesimpulan

Setelah dilakukan pengujian dapat ditarik kesimpulan sebagai berikut :

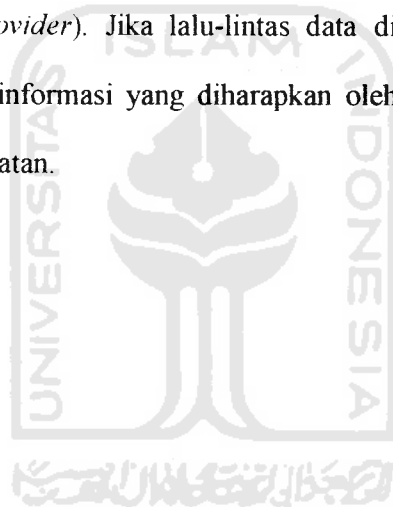
1. Aplikasi berbasis *web* ini cukup membantu calon konsumen perumahan untuk mencari informasi singkat mengenai perumahan, berdasarkan lokasi yang diinginkan
2. Aplikasi berinteraksi dengan calon konsumen melalui *web* ini terbukti efektif dan efisien karena dapat diakses kapan saja dan dimana saja, sepanjang tersedia sambungan ke internet.
3. Penggunaan aplikasi berbasis *web* sebagai pengelolaan *database* lebih menjamin keamanan data dengan hanya operator yang mempunyai hak akses penuh melalui verifikasi nama dan *password*, sedangkan calon konsumen cukup berinteraksi melalui halaman pencarian yang tidak membutuhkan verifikasi nama pengguna dan *password*.
4. Berita yang masuk tersimpan di aplikasi dan dapat disaring oleh operator. Hal ini untuk menghindari adanya ketidak konsistennya data.

#### 7.2 Saran

1. Dalam proses pengembangan, aplikasi-aplikasi yang digunakan cukup menguras sumber daya (*resources*) komputer, terutama bagian prosesor dan memori. Ini dikarenakan aplikasi ini mengakses *query* secara bertingkat secara simultan. Diharapkan ada penelitian sejenis yang dapat

mengembangkan aplikasi yang lebih hemat dalam pemakaian sumber daya komputer.

2. Hasil keluaran informasi yang diterima oleh calon konsumen sangat tergantung dari ada tidaknya sambungan internet di tempatnya berada.
3. Belum dapat menghasilkan informasi yang lebih mendetail tentang perumahan yang diinginkan, misalkan luas bangunan, luas tanah, fasilitas yang disediakan oleh perumahan tersebut dan sebagainya.
4. Sangat tergantung dengan kondisi jaringan internet yang dimiliki oleh ISP (*Internet Service Provider*). Jika lalu-lintas data di jaringan ISP tersebut sedang sibuk, maka informasi yang diharapkan oleh calon konsumen akan mengalami keterlambatan.

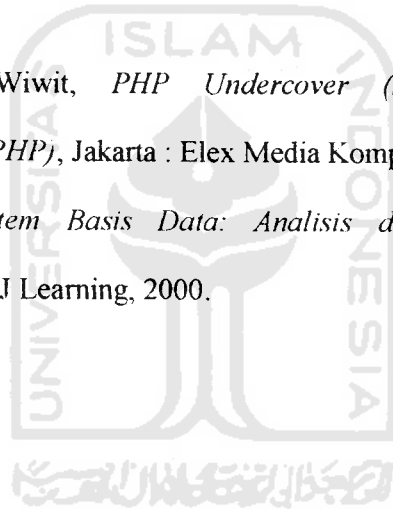




## DAFTAR PUSTAKA

- [ANO01] *Question Answering*, Wikipedia.org,  
[http://en.wikipedia.org/wiki/Question\\_answering](http://en.wikipedia.org/wiki/Question_answering), diedit 14 April 2007
- [ANO02] MySQL Server Online-Manual
- [ANO03] *Tentang Internet*, <http://www.klik-kanan.com/artikel/internet>, diakses Juni 2007.
- [FAT01] Fathansyah, *Basis Data*, Bandung: Informatika Bandung, 1999.
- [JOG01] Jogianto, H.M, *Analisis & Desain: Sistem Informasi Pendekatan terstruktur teori dan praktek aplikasi bisnis*, Yogyakarta: P.T. Andi Offset, 1999.
- [KAD01] Kadir, A, *Penuntun Praktis Belajar SQL*, Yogyakarta.: P.T. Andi Offset, 2002.
- [LID01] Liddy, E.D., *Enhanced text retrieval using Natural Language Processing*, USA: Bulletin of the American Society for Information Science. Vol. 24, No. 4., <http://www.asis.org/Bulletin/Apr-98/liddy.html>, 1998
- [MAR01] Martina, I., *Aplikasi Pemrograman Database Menggunakan Delphi*, Jakarta: PT Elex Media Komputindo, 2001.
- [MAR02] Martina, I., *Pemrograman Client Server Menggunakan Delphi 5.0*, Jakarta: PT Elex Media Komputindo, 2002.
- [POH01] Pohan, H.I. & Bahri, K.S., *Pengantar Perancangan Sistem*, Jakarta: Erlangga, 1997

- [PUJ01] Pujiyono, W., *Diktat Kuliah Basis Data*, Yogyakarta: Teknik Informatika Universitas Ahmad Dahlan, 2000.
- [SAG01] Sager, N, *Natural Language Information Processing: A Computer Grammar of English and Its Applications*, Massachusset - USA: Addison-Wesley, 1981.
- [SAN01] Sanaky Hujair dkk, *Pedoman Kuliah Kerja Nyata Universitas Islam Indonesia*, Yogyakarta : Pusat KKN, 2005.
- [SID01] Sidik Betha, *Pemrograman Web dengan PHP*, Bandung : Informatika Bandung, 2002.
- [SIS01] Siswoutomo Wiwit, *PHP Undercover (mengungkap rahasia pemrograman PHP)*, Jakarta : Elex Media Komputindo, 2005
- [WAL01] Waljianto, *Sistem Basis Data: Analisis dan Pemodelan data*, Yogyakarta: J&J Learning, 2000.



## LAMPIRAN

### KODE SUMBER

#### 9.1 Form Login

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1" />

<title>Login</title>
<link href="css/global.css" rel="stylesheet" type="text/css" />
</head>

<body>

<form name="login" id="login" action="processlogin.php"
method="post" target="_parent">

<table width="480" border="0" align="center" cellpadding="2"
cellspacing="2">

<?
if(isset($_GET['invalid'])) {
    if(strtolower($_GET['invalid'])=="yes") {
?>

    <tr>

        <td colspan="2" class="blackonred04">Maaf, sepertinya nama dan
<em>password</em>-nya tidak cocok. Coba lagi</td>

    </tr>

    <tr>

        <td colspan="2" class="normal">&nbsp;  </td>

    </tr>

<?
    }
}
```

```
?>
<tr>
  <td colspan="2" class="blueHeader1">Log-In</td>
</tr>
<tr>
  <td width="55%" align="right" class="normal">Nama <em>user</em>
</td>
  <td width="45%"><input name="nama" type="text" class="normal"
id="nama" size="20" maxlength="30" /></td>
</tr>
<tr>
  <td width="55%" align="right"
class="normal"><em>Password</em></td>
  <td width="45%"><input name="passw" type="password"
class="normal" id="passw" size="20" maxlength="40" /></td>
</tr>
<tr>
  <td colspan="2" align="right" class="blueHeader1"><input
type="submit" name="Submit" value="Login" />&nbsp;
  <input name="Reset" type="reset" id="Reset" value="Reset"
/></td>
</tr>
</table>
</form>
<script language="javascript">
document.login.nama.focus();
</script>
</body>
</html>
```

## 9.2 Untuk Memproses Pencarian

```
include_once("include/searchfunctions.php");

?>

<tr>

    <td colspan="3" class="normalgrayline">

        <table width="100%" cellpadding="2" cellspacing="1"
border="0">

<?

$hasil="";

$sql_perumahan="SELECT perumahan.id,perumahan.nama AS
perumahan,kecamatan.nama AS kecamatan,kelurahan.nama AS
kelurahan,kelurahan.kode_pos,developer.id AS
id_developer,developer.id_bu,developer.nama AS
developer,developer.telp FROM (perumahan INNER JOIN (kelurahan INNER
JOIN kecamatan ON kelurahan.id_kecamatan=kecamatan.id) ON
perumahan.id_kelurahan=kelurahan.id) INNER JOIN developer ON
perumahan.id_developer=developer.id";

if(empty($_GET['detail'])) {

    $strict=false;

    $exact=false;

    $TABEL=array("kecamatan","kelurahan","developer","perumahan");

    //http://localhost/hendra/search.php?keyword=merapi&mode=advan
ced&keytype=5

    if(!empty($_GET['mode']) && !empty($_GET['keytype'])) {

        $keytype=(int)$_GET['keytype'];

        if(strtolower($_GET['mode'])=="advanced") {

            $strict=($keytype>1);

            $exact=(!empty($_GET['exact']) &&
strtolower($_GET['exact'])=="true");

        }

    }

    $found=0;

    $judul="";

    $h_kecamatan=NULL;
```

```

$h_kelurahan=NULL;

$h_developer=NULL;

$h_perumahan=NULL;

//Cek apakah tipe pencariannya Mode Ahli atau bukan
if($strict) {

    //Ternyata benar Mode Ahli. Cek tipe keywordnya, apakah:

    switch($keytype) {

        case 2: //kecamatan;

            //echo "strict mode <br>\n";

            $sql="SELECT id,nama FROM kecamatan WHERE ";

            if($exact) {

                $sql.="nama='$_GET[keyword]' ORDER BY

nama";

            } else {

                $sql.="nama LIKE '%$_GET[keyword]%'

ORDER BY nama";

            }

            $rs_kecamatan=mysql_query($sql);

            if(mysql_num_rows($rs_kecamatan)>0) {

                while($kecamatan=mysql_fetch_row($rs_kecamatan)) {

                    $sidx=$kecamatan[1];

                    $sql="$sql_perumahan WHERE

kecamatan.id='".$kecamatan[0]."'";

                    if(count($h_kecamatan)<1 ||

!array_key_exists($sidx,$h_kecamatan))

                        $h_kecamatan[$sidx]=NULL;

                    $found+=hasil_perumahan($sql,$linkadmin,$linkmode,$ganjil,$h_k

ecamatan[$sidx]);

                }

            }

            break;

        case 3: //kelurahan

```

```

        $sql="SELECT id,nama FROM kelurahan WHERE ";
        if($exact) {
            $sql.="nama='$_GET[keyword]' ORDER BY
nama";
        } else {
            $sql.="nama LIKE '%$_GET[keyword]%'
ORDER BY nama";
        }
        $rs_kelurahan=mysql_query($sql);
        if(mysql_num_rows($rs_kelurahan)>0) {
            while($kelurahan=mysql_fetch_row($rs_kelurahan)) {
                $sidx=$kelurahan[1];
                $sql="$sql_perumahan WHERE
kelurahan.id='".$kelurahan[0]."'";
                if(count($h_kelurahan)<1 ||
!array_key_exists($sidx,$h_kelurahan))
                    $h_kelurahan[$sidx]=NULL;
                $found+=hasil_perumahan($sql,$linkadmin,$linkmode,$ganjil,$h_k
elurahan[$sidx]);
            }
        }
        break;
        case 4: //developer
            $sql="SELECT developer.*,kecamatan.nama AS
kecamatan,kelurahan.nama AS kelurahan FROM developer INNER JOIN
(kelurahan INNER JOIN kecamatan ON
kelurahan.id_kecamatan=kecamatan.id) ON
kelurahan.id=developer.id_kelurahan WHERE ";
            if($exact) {
                $sql.="developer.nama='$_GET[keyword]'
ORDER BY developer.nama";
            } else {
                $sql.="developer.nama LIKE
'%$_GET[keyword]%' ORDER BY developer.nama";
            }
        }
    }
}

```

```

$data=hasil_developer($sql,$linkadmin,$linkmode,$ganjil);

        if($data['jumlah']>0) {

                $jml=count($h_developer);

                $h_developer[$jml]=$data['hasil'];

                $found+=$data['jumlah'];

        }

        break;

        case 5: //perumahan

                if($exact) {

                        $sql="$sql_perumahan WHERE
perumahan.nama='$_GET[keyword]' ORDER BY perumahan";

                } else {

                        $sql="$sql_perumahan WHERE
perumahan.nama LIKE '%$_GET[keyword]%' ORDER BY perumahan";

                }

                $found+=hasil_perumahan($sql,$linkadmin,$linkmode,$ganjil,$h_p
erumahan);

                break;

        }

} else {

        if($exact) {

                $sql_keyword="SELECT id_pk,tabel,keyword FROM keyword
WHERE keyword='$_GET[keyword]' GROUP BY tabel ORDER BY BINARY
tabel,keyword";

        } else {

                //Kalo tipe pencarian Mode Sederhana

                $sql_keyword="SELECT id_pk,tabel,keyword FROM keyword
WHERE keyword LIKE '%$_GET[keyword]%' GROUP BY tabel ORDER BY BINARY
tabel,keyword";

        }

        $rs_keyword=mysql_query($sql_keyword);

        $ganjil=true;

        //Apakah keyword ditemukan?

```



```

if(mysql_num_rows($rs_keyword)>0) {
    $stabel="";
    //Jika keyword ditemukan, cek satu-persatu jenis
keyword, apakah:
    while($keyword=mysql_fetch_row($rs_keyword)) {
        $stabel=$keyword[1];
        $idx=array_search($stabel,$TABEL,true);
        if($idx!==FALSE) {
            switch($idx) {
                case 2: //developer
                    $ssql="SELECT
developer.*,kecamatan.nama AS kecamatan,kelurahan.nama AS kelurahan
FROM developer INNER JOIN (kelurahan INNER JOIN kecamatan ON
kelurahan.id_kecamatan=kecamatan.id) ON
kelurahan.id=developer.id_kelurahan WHERE
developer.id='".$keyword[0]."'";
                    $data=hasil_developer($ssql,$linkadmin,$linkmode,$ganjil);
                    if($data['jumlah']>0) {
                        $jml=count($h_developer);
                        $h_developer[$jml]=$data['hasil'];
                        $found+=$data['jumlah'];
                    }
                    break;
                case 3: //perumahan
                    $ssql="$ssql_perumahan WHERE
perumahan.id='".$keyword[0]."' ORDER BY perumahan";
                    $found+=hasil_perumahan($ssql,$linkadmin,$linkmode,$ganjil,$h_p
erumahan);
                    break;
                default: //Kecamatan & kelurahan
                    if($idx==0) { //Kalo kecamatan
                        $kecamatan=mysql_fetch_row(mysql_query("SELECT nama FROM
kecamatan WHERE id='".$keyword[0]."'"));

```

```

        $sidx=$kecamatan[0];

        $sql="$sql_perumahan WHERE
kecamatan.id='". $keyword[0]."' ORDER BY kelurahan,perumahan";

        if(count($h_kecamatan)<1
|| !array_key_exists($sidx,$h_kecamatan))

        $h_kecamatan[$sidx]=NULL;
        $found+=hasil_perumahan($sql,$linkadmin,$linkmode,$ganjil,$h_k
ecamatan[$sidx]);

        } elseif($sidx==1) { //Kalo
kelurahan

        $kelurahan=mysql_fetch_row(mysql_query("SELECT nama FROM
kelurahan WHERE id='". $keyword[0]."'"));

        $sidx=$kelurahan[0];

        $sql="$sql_perumahan WHERE
kelurahan.id='". $keyword[0]."' ORDER BY perumahan";

        if(count($h_kelurahan)<1
|| !array_key_exists($sidx,$h_kelurahan))

        $h_kelurahan[$sidx]=NULL;
        $found+=hasil_perumahan($sql,$linkadmin,$linkmode,$ganjil,$h_k
elurahan[$sidx]);

        }

        break;

    }

}

}

}

//Tampilkan

if(count($h_kecamatan)>0) {

    ksort($h_kecamatan);

    foreach($h_kecamatan as $key=>$avalue) {

        $hasil.="<t<tr><td
class=\"blackongray01\">Perumahan di Kecamatan ".ucfirst($key).
("".count($avalue). " data)</td></tr>\n";

```

```

        if(is_array($avalue)) { foreach($avalue as $value)
$hasil.=$value; }
    }
}

if(count($h_kelurahan)>0) {
    ksort($h_kelurahan);

    foreach($h_kelurahan as $key=>$avalue) {

        $hasil.="\t<tr><td
class=\"blackongray01\">Perumahan di Kelurahan ".ucfirst($key)."
(".count($avalue)." data)</td></tr>\n";

        if(is_array($avalue)) { foreach($avalue as $value)
$hasil.=$value; }
    }
}

if(count($h_perumahan)>0) {
    $hasil.="\t<tr><td class=\"blackongray01\">Perumahan
(".count($h_perumahan)." data)</td></tr>\n";

    foreach($h_perumahan as $value) $hasil.=$value;
}

if(count($h_developer)>0) {
    $hasil.="\t<tr><td
class=\"blackongray01\"><em>Developer</em> (" .count($h_developer)."
data)</td></tr>\n";

    foreach($h_developer as $value) $hasil.=$value;
}

echo "\t<tr>\n\t\t<td class=\"whiteonblack04\"
align=\"center\">Hasil Pencarian (ditemukan $found
entri)</td>\n\t</tr>\n$hasil";
} else {

    if(strtolower($_GET['detail'])=="developer" &&
!empty($_GET['id'])) {

        $back="search.php?keyword=$_GET[keyword]";

        if(strlen($linkadmin)>14) $back.="&$linkadmin";

        $back.="&$linkmode";

?>

```

```

        <tr>
            <td class="blueHeader1">Detail
<em>Developer</em></td>
        </tr>
        <tr>
            <td class="normal01"><a
href="<?=$back?>">&lt;&lt;&nbsp;&nbsp;&nbsp;<strong>Kembali ke
pencarian</strong></a></td>
        </tr>

```

<?

```

        $rs_developer=mysql_query("SELECT
developer.*,kecamatan.nama AS kecamatan,kelurahan.nama AS kelurahan
FROM developer INNER JOIN (kelurahan INNER JOIN kecamatan ON
kelurahan.id_kecamatan=kecamatan.id) ON
kelurahan.id=developer.id_kelurahan WHERE
developer.id='$_GET[id]'");

```

```

        if(mysql_num_rows($rs_developer)>0) {

```

```

            $developer=mysql_fetch_array($rs_developer,MYSQL_ASSOC);

```

```

            $nama=nama_developer($developer['id_bu'],$developer['nama']);

```

?>

```

        <tr>
            <td
class="normal"><strong><?=$nama?></strong></td>

```

```

        </tr>

```

```

        <tr>

```

```

            <td class="normal">Alamat:
<?=$developer['alamat']?></td>

```

```

        </tr>

```

```

        <tr>

```

```

            <td class="normal"><?="Kecamatan
".ucfirst(strtolower($developer['kecamatan'])).", Kelurahan
".ucfirst(strtolower($developer['kelurahan']))?></td>

```

```

        </tr>

```

```

        <tr>

```

```

            <td class="normal">Telp.:
<?=$developer['telp']?></td>

```

```

        </tr>

        <tr>
            <td class="blackongray01" align="center">Daftar
Perumahan yang Dibangun</td>
        </tr>

        <tr>
            <td><table width="100%" border="0" align="center"
cellpadding="2" cellspacing="2">
                <tr class="whiteonblack04">
                    <th scope="col" width="20%">Kecamatan</th>
                    <th scope="col" width="20%">Kelurahan</th>
                    <th scope="col" width="60%">Nama Perumahan</th>
                </tr>
            <?
                $sql="$sql_perumahan WHERE
developer.id='$_GET[id]' ORDER BY BINARY
perumahan,kecamatan,kelurahan";

                $rs_perumahan=mysql_query($sql_perumahan);

                $ganjil=true;

                //echo "$sql_perumahan<br>\n";
                while($perumahan=mysql_fetch_array($rs_perumahan,MYSQL_ASSOC))
                {

                    if($ganjil) { $class="normal"; }
                    else { $class="blackongray02"; }

                ?>

                <tr class="<?=$class?>">
                    <td width="20%" valign="top"
align="left"><?=ucfirst(strtolower($perumahan['kecamatan']))?></td>
                    <td width="20%" valign="top"
align="left"><?=ucfirst(strtolower($perumahan['kelurahan']))?></td>
                    <td width="60%" valign="top"
align="left"><strong><?=$perumahan['perumahan']?></strong></td>
                </tr>
            <?

```



```

        $ganjil=!$ganjil;
    }
}
}
?>
        </table></td>
    </tr>
</table>
</tr>
<?
//Fungsi-fungsi yang digunakan dalam pencarian
function nama_developer($bu,$nama) {
    $result="";
    switch($bu) {
        case 1: $result="P.T. $nama";
            break;
        case 2: $result="C.V. $nama";
            break;
        case 3: $result="PERUM. $nama";
            break;
    }
    return($result);
}
function hasil_perumahan($sql,$linkadmin,$linkmode,&$ganjil,&$array)
{
    //echo "<tr><td>$sql</td></tr>\n";
    $rs_perumahan=mysql_query($sql);
    $jumlah=mysql_num_rows($rs_perumahan);
    if($jumlah>0) {

```

```

        while($perumahan=mysql_fetch_array($rs_perumahan,MYSQL_ASSOC)
        {
            if($ganjil) { $class="normal01"; }

            else { $class="blackonblue01"; }
            $developer=nama_developer($perumahan['id_bu'],$perumahan['deve
            loper']);

            $hasil="\t<tr>\n\t\t\t<td><table width=\"100%\"
            cellpadding=\"2\" cellspacing=\"1\" class=\"$class\">\n";
            $hasil.="\t\t\t<tr><td><strong>$perumahan[perumahan]</strong></t
            d></tr>\n";

            $hasil.="\t\t\t<tr><td>Kecamatan $perumahan[kecamatan],
            Kelurahan $perumahan[kelurahan] ($perumahan[kode_pos])</td></tr>\n";

            $hasil.="\t\t\t<tr><td>Developer: <a href=\"search.php?\";

            if(strlen($linkadmin)>14) $hasil.="$linkadmin&";
            $hasil.="keyword=".urlencode($_GET['keyword'])."&$linkmode&det
            ail=developer&id=$perumahan[id_developer]\"><strong>$developer</stro
            ng></a>, &nbsp;Telp.: $perumahan[telp]</td></tr>\n";

            $hasil.="\t\t\t</table>\n\t\t\t</td>\n\t\t\t</tr>\n";

            $jml=count($array);

            $array[$jml]=$hasil;

            $ganjil=!$ganjil;

        }

    }

    return($jumlah);

}

```

```

function hasil_developer($sql,$linkadmin,$linkmode,&$ganjil) {
    //echo "$sql<br>\n";

    $rs_developer=mysql_query($sql);

    $jumlah=mysql_num_rows($rs_developer);

    $hasil="";

    if($jumlah>0) {

        $developer=mysql_fetch_array($rs_developer,MYSQL_ASSOC);

        if($ganjil) { $class="normal01"; }

        else { $class="blackonblue01"; }
    }
}

```

```

        $nama=nama_developer($developer['id_bu'],$developer['nama']);

        $hasil="\t<tr><td><table width=\"100%\" cellpadding=\"2\"
cellspacing=\"1\" border=\"0\" class=\"$class\">\n\t\t<tr><td><a
href=\"search.php?\";

        if(strlen($linkadmin)>14) $hasil.="$linkadmin&";

        $hasil.="keyword=".urlencode($_GET['keyword'])."&$linkmode&det
ail=developer&id=$developer[id]\"><strong>$nama</strong></a></td></t
r>\n\t\t\t<tr><td>Alamat:
<strong>$developer[alamat]</strong></td></tr>\n\t\t\t<tr><td>Kecamatan
$developer[kecamatan], Kelurahan
$developer[kelurahan]</td></tr>\n\t\t\t<tr><td>Telp.
:<strong>$developer[telp]</strong></td></tr>\n\t\t</table></td></tr>\n
";
        $ganjil=!$ganjil;
    }

    return(array("jumlah"=>$jumlah,"hasil"=>$hasil));
}
?>
<?

```

### 9.3 Untuk Menampilkan dan Mengedit Data Kecamatan

```

require_once("include/checklogin.php");
$stabel="kecamatan";

if($logged) {
    $sid=0;
    $act="";

    if(!empty($_GET['id'])) $sid=(int)$_GET['id'];
    if(!empty($_GET['act'])) $act=$_GET['act'];

    $rs_kecamatan=mysql_query("SELECT * FROM kecamatan ORDER BY
nama");

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

```



```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Data Kecamatan</title>

<link href="css/global.css" rel="stylesheet" type="text/css" />

<style type="text/css">

<!--

body {

    background-color: #FFFFFF;

    margin-left: 0px;

    margin-top: 0px;

    margin-right: 0px;

    margin-bottom: 0px;

}

-->

</style></head>

<script language="javascript">

function addnew() {

var form=document.kecamatan;

form.nama.value="";

form.id.value="0";

form.act.value="new";

if(document.getElementById("datainput")) {

    document.getElementById("datainput").style.visibility="visible";

    form.nama.focus();

}

}

function edit(id,nama) {

addnew();

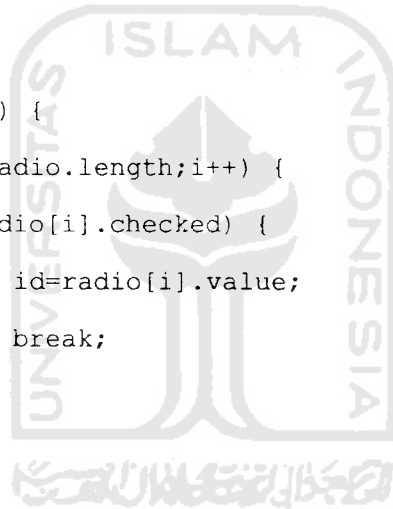
var form=document.kecamatan;

form.id.value=id;
```



```
form.nama.value=nama;
form.act.value="edit";
form.nama.select();
}
function cancelinput() {
var form=document.kecamatan;
if(document.getElementById("datainput") && form.nama) {
    form.nama.value="";
    form.id.value="0";
    form.act.value="";
    document.getElementById("datainput").style.visibility="hidden"
}
}
function removeitem(id,nama) {
if (window.confirm("Yakin kecamatan '"+nama+"' akan dihapus?")) {
    var form=document.kecamatan;
    form.id.value=id;
    form.act.value="remove";
    form.submit();
}
}
function checkinput() {
var form=document.kecamatan;
if(form.act.value=="new" || form.act.value=="edit") {
    if(form.nama.value=="") {
        window.alert("Nama kecamatan masih kosong");
        form.nama.focus();
        form.nama.select();
        return;
    }
}
```

```
} else if(form.act.value!="remove") return;
form.action="savekecamatan.php";
form.method="post";
form.target="_self";
form.submit();
}
function selectradio() {
form=document.kecamatan;
radio=form.id_kecamatan;
id=0;
try {
    if(radio.length>0) {
        for(i=0;i<radio.length;i++) {
            if(radio[i].checked) {
                id=radio[i].value;
                break;
            }
        }
    } else {
        if(radio.checked) id=radio.value;
    }
} catch(e) {
    if(radio.checked) id=radio.value;
}
if(id>0) {
    keyword="kecamatan.php?admin=yes&uid="+form.uid.value+"&id="+id;
    location.href=keyword;
}
}
</script>
```



```

<body>

<table width="720" border="0" align="center" cellpadding="2"
cellspacing="2">

  <tr>

    <td class="blueHeader1">Data Kecamatan </td>

  </tr>

  <tr>

    <td><table width="99%" border="0" align="center" cellpadding="2"
cellspacing="2">

      <tr>

        <th width="8%" class="whiteonblack04" scope="col">No</th>

        <th width="92%" class="whiteonblack04" scope="col">Nama
Kecamatan </th>

      </tr>

      <form name="kecamatan" id="kecamatan"
action="javascript:checkinput()">

        <input type="hidden" name="admin" id="admin" value="yes" />

        <input type="hidden" name="uid" id="uid" value="<?=$_GET['uid']?>"
/>

        <input type="hidden" name="act" id="act" value="<?=$act?>" />

        <?

          if(mysql_num_rows($rs_kecamatan)<1) {

            ?>

            <tr>

              <td align="left" valign="top">&nbsp;</td>

              <td align="left" valign="top">&nbsp;</td>

            </tr>

            <?

          } else {

            $i=1;

            $ganjil=true;

            ?>

```

```

<tr>
    <td colspan="2">
        <div id="listkecamatan" style="height:200px;
width:713px; overflow: auto; border: solid 1px #CCCCCC;">
            <table width="100%" border="0" cellpadding="2"
cellspacing="1">
<?
    while($kecamatan=mysql_fetch_array($rs_kecamatan,MYSQL_ASSOC))
    {
        if($id<1) $id=$kecamatan['id'];
        if($ganjil) {
            $class="normal01";
        } else {
            $class="blackonblue01";
        }
        $editbtn="edit$i";
        $removebtn="hapus$i";
        if((int)$kecamatan['id']==(int)$id) {
            $checked=" checked";
        } else {
            $checked="";
        }
    }
?>
    <tr class="<?=$class?>">
        <td width="8%" align="right"
valign="top"><?=$i?>.</td>
        <td width="56%" align="left" valign="top"><input
type="radio" name="id_kecamatan" id="id_kecamatan"
value="<?=$kecamatan['id']?>"<?=$checked?> onclick="selectradio()"/>
<?=$kecamatan['nama']?></td>
        <td width="18%" align="center" valign="top"><input
type="button" name="<?=$editbtn?>" id="<?=$editbtn?>" value="Edit"
onclick="edit('<?=$kecamatan['id']?>','<?=$kecamatan['nama']?>')" />
        <td width="18%" align="center" valign="top"><input
type="button" name="<?=$removebtn?>" id="<?=$removebtn?>"
value="Hapus"

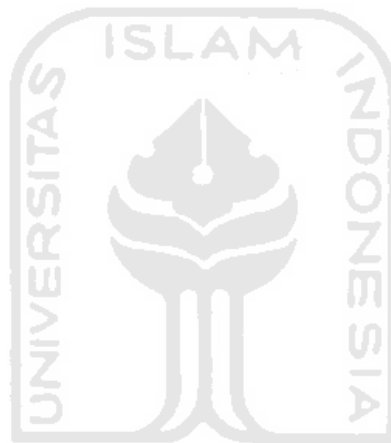
```

```

onclick="removeitem('<?=$kecamatan['id']?>','<?=$kecamatan['nama']?>
')" />

        </tr>
<?
        $i++;
        $ganjil=!$ganjil;
    }
?>
    </table>
</div>
</td>
</tr>
<?
    }
?>
</table></td>
</tr>
<tr>
    <td><input name="BtnTambah" type="button" id="BtnTambah"
value="Tambah Data Kecamatan" onclick="addnew()" /></td>
</tr>
<tr>
    <td><div id="datainput" style="visibility:hidden">
        <table width="90%" align="center" cellpadding="2"
cellspacing="1" class="blackonbrown02">
            <tr>
                <td width="45%" align="right">Nama kecamatan</td>
                <td width="55%"><input name="nama" id="nama"
class="normal" size="20" maxlength="30" /></td>
            </tr>
        </table>
    </div>
    </td>

```



```

        <td colspan="2" align="center"><input
type="submit" name="btncancel" id="btncancel" value="Batal"
/>&nbsp;<input type="button" name="btncancel" id="cancel"
value="Batal" onclick="cancelinput()" />

<input type="hidden" name="id" id="id" value="<?=$id?>" />

</form>

<script language="javascript">

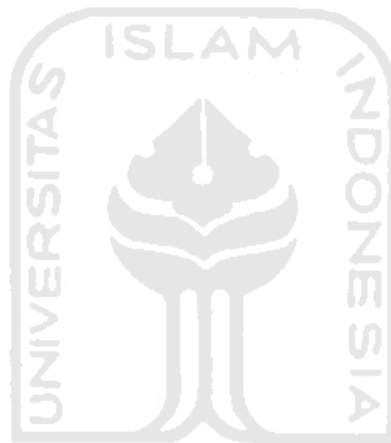
form=document.kecamatan;

if(form.act.value=="new") addnew();

</script>

        </td>
    </tr>
</table>
</div>
</td>
</tr>
<tr>
<td>
<?
require_once("include/keyword.php");
?>
</td>
</tr>
</table>
</body>
</html>
<?
}
?>
<?

```



#### 9.4 Untuk Menampilkan dan Mengedit Data Kelurahan

```
require_once("include/checklogin.php");
$tabel="kelurahan";
if($logged) {
    $id=0;
    $act="";
    $rs_kecamatan=mysql_query("SELECT id,nama FROM kecamatan ORDER
BY nama");
    if(!empty($_GET['id'])) $id=(int)$_GET['id'];
    if(!empty($_GET['act'])) $act=$_GET['act'];
    if(empty($_GET['kecamatan'])) {
        if(mysql_num_rows($rs_kecamatan)>0) {
            $kecamatan=mysql_fetch_row($rs_kecamatan);
            $kec=$kecamatan[0];
            mysql_data_seek($rs_kecamatan,0);
        }
    } else {
        $kec=(int)$_GET['kecamatan'];
    }
    $rs_kelurahan=mysql_query("SELECT * FROM kelurahan WHERE
id_kecamatan='$kec' ORDER BY nama");
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1" />
<title>Data Kecamatan</title>
<link href="css/global.css" rel="stylesheet" type="text/css" />
<style type="text/css">
<!--
```



```
body {
    background-color: #FFFFFF;
    margin-left: 0px;
    margin-top: 0px;
    margin-right: 0px;
    margin-bottom: 0px;
}
-->
</style></head>
<script language="javascript">
function addnew() {
var form=document.kelurahan;
if(form.kecamatan.length<1) {
    alert("Kecamatan masih kosong");
    return;
}
if(form.kecamatan.selectedIndex<0) {
    alert("Pilih salah satu kecamatan");
    form.kecamatan.focus();
    return;
}
form.nama.value="";
form.id.value="0";
form.act.value="new";
if(document.getElementById("datainput")) {
    document.getElementById("datainput").style.visibility="visible";
    form.nama.focus();
}
}
function edit(id,nama,kdpos) {
```

```
addnew();

var form=document.kelurahan;
form.id.value=id;
form.nama.value=nama;
form.kode_pos.value=kdpos;
form.act.value="edit";
form.nama.select();
}

function cancelinput() {
var form=document.kelurahan;
if(document.getElementById("datainput") && form.nama) {
    form.nama.value="";
    form.id.value="0";
    form.act.value="";
    document.getElementById("datainput").style.visibility="hidden"
}
}

function removeitem(id,nama) {
if (window.confirm("Yakin kelurahan '"+nama+"' akan dihapus?")) {
    var form=document.kelurahan;
    form.id.value=id;
    form.act.value="remove";
    form.submit();
}
}

function checkinput() {
var form=document.kelurahan;
if(form.kecamatan.length<1) {
    alert("Kecamatan masih kosong");
    return;
}
```

```
}  
  
if(form.kecamatan.selectedIndex<0) {  
    alert("Pilih salah satu kecamatan");  
    form.kecamatan.focus();  
    return;  
}  
  
if(form.act.value=="new" || form.act.value=="edit") {  
    if(form.nama.value=="") {  
        window.alert("Nama kecamatan masih kosong");  
        form.nama.focus();  
        return;  
    }  
    if(form.kode_pos.value=="") {  
        window.alert("Kode pos masih kosong");  
        form.kode_pos.focus();  
        return;  
    }  
    if(form.kode_pos.value.length!=5) {  
        window.alert("Jumlah karakter kode pos haruslah 5  
digit!");  
        form.kode_pos.focus();  
        form.kode_pos.select();  
        return;  
    }  
    try {  
        if(parseInt(form.kode_pos.value)<10000) {  
            window.alert("Format kode pos salah. Seharusnya  
terdiri dari angka sebanyak 5 digit");  
            form.kode_pos.focus();  
            form.kode_pos.select();  
            return;  
        }  
    }  
}
```

```

        }
    } catch(e) {
        window.alert("Format kode pos salah. Seharusnya terdiri
dari angka sebanyak 5 digit");
        form.kode_pos.focus();
        form.kode_pos.select();
        return;
    }
} else if(form.act.value!="remove") return;
if(form.kecamatan.selectedIndex>=0)
form.kecamatan.value=form.kecamatan.options[form.kecamatan.selectedI
ndex].value;
form.action="savekelurahan.php";
form.method="post";
form.target="_self";
form.submit();
}
function selectradio() {
form=document.kelurahan;
if(form.kecamatan.length<1) {
    alert("Kecamatan masih kosong");
    return;
}
if(form.kecamatan.selectedIndex<0) {
    alert("Pilih salah satu kecamatan");
    form.kecamatan.focus();
    return;
}
radio=form.id_kelurahan;
id=0;
try {
    if(radio.length>0) {

```



```

        for(i=0;i<radio.length;i++) {
            if(radio[i].checked) {
                id=radio[i].value;
                break;
            }
        }
    } else {
        if(radio.checked) id=radio.value;
    }
} catch(e) {
    if(radio.checked) id=radio.value;
}
if(id>0) {
    keyword="kelurahan.php?admin=yes&uid="+form.uid.value+"&kecamatan="+form.kecamatan.options[form.kecamatan.selectedIndex].value+"&id="+id;
    location.href=keyword;
}
}
function kecamatanchange() {
    form=document.kelurahan;
    if(form.kecamatan.selectedIndex>=0 && form.kecamatan.length>0) {
        location.href="kelurahan.php?admin=yes&uid="+form.uid.value+"&kecamatan="+form.kecamatan.options[form.kecamatan.selectedIndex].value;
    }
}
</script>
<body>
<form name="kelurahan" id="kelurahan"
action="javascript:checkinput()">
<input type="hidden" name="admin" id="admin" value="yes" />

```

```

<input type="hidden" name="uid" id="uid" value="<?=$_GET['uid']?>"
/>

<input type="hidden" name="act" id="act" value="<?=$_act?>" />

<table width="720" border="0" align="center" cellpadding="2"
cellspacing="2">

  <tr>

    <td class="blueHeader1">Data Kelurahan </td>

  </tr>

  <tr>

    <td>

      <table width="99%" border="0" align="center" cellpadding="2"
cellspacing="2" class="blackongray01">

        <tr valign="top">

          <td width="45%" align="right">Pilih Kecamatan </td>

          <td width="55%">&nbsp;<select name="kecamatan"
class="normal" id="kecamatan" onchange="kecamatanchange()">

<?

  while($kecamatan=mysql_fetch_row($rs_kecamatan)) {

    if($kecamatan[0]==$kec) { $selected="selected"; }

    else { $selected=""; }

?>

          <option value="<?=$kecamatan[0]?>"
<?=$selected?><?=$kecamatan[1]?></option>

<?

        }

?>

          </select>

        </td>

      </tr>

    </table>

  </td>

</tr>

```

```

<tr>
  <td><table width="99%" border="0" align="center" cellpadding="2"
cellspacing="2">
  <tr>
    <th width="7%" class="whiteonblack04" scope="col">No</th>
    <th width="53%" class="whiteonblack04" scope="col">Nama
Kelurahan </th>
    <th width="40%" class="whiteonblack04" scope="col">Kode Pos
</th>
  </tr>
  <tr class="normal">
    <td colspan="3"><input type="button"
name="btnrefresh" id="btnrefresh" value="Refresh"
onclick="location.reload()" /></td>
  </tr>
<?
  if(mysql_num_rows($rs_kelurahan)<1) {
?>
  <tr>
    <td align="left" valign="top">&nbsp;</td>
    <td colspan="2" align="left" valign="top">&nbsp;</td>
  </tr>
<?
  } else {
    $i=1;
    $ganjil=true;
?>
  <tr>
    <td colspan="3">
      <div id="listkelurahan" style="height:140px;
width:713px; overflow: auto; border: solid 1px #CCCCCC;">
        <table width="100%" border="0" cellpadding="2"
cellspacing="1">

```

```

<?
while($kelurahan=mysql_fetch_array($rs_kelurahan,MYSQL_ASSOC))
{
    if($id<1) $id=$kelurahan['id'];
    if($ganjil) {
        $class="normal01";
    } else {
        $class="blackonblue01";
    }
    $editbtn="edit$i";
    $removebtn="hapus$i";
    if((int)$kelurahan['id']==(int)$id) {
        $checked=" checked";
    } else {
        $checked="";
    }
?>
|
|  |

```



```

                $ganjil=!$ganjil;
            }
        ?>
        </table>
        </div>                </td>
    </tr>
<?
    }
?>
    </table></td>
</tr>
<tr>
    <td><input name="BtnTambah" type="button" id="BtnTambah"
value="Tambah Data Kelurahan" onclick="addnew()" /></td>
</tr>
<tr>
    <td><div id="datainput" style="visibility:hidden">
        <table width="90%" align="center" cellpadding="2"
cellspacing="1" class="blackenbrown02">
            <tr>
                <td width="45%" align="right">Nama
Kelurahan&nbsp;</td>
                <td width="55%"><input name="nama" id="nama"
class="normal" size="20" maxlength="30" /></td>
            </tr>
            <tr>
                <td width="45%" align="right">Kode
Pos&nbsp;</td>
                <td width="55%"><input name="kode_pos"
id="kode_pos" class="normal" size="5" maxlength="5" />
            </tr>
            <td colspan="2" align="center"><input
type="submit" name="btncsave" id="btncsave" value="Simpan"

```

```

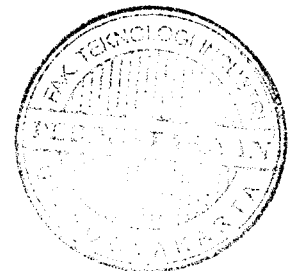
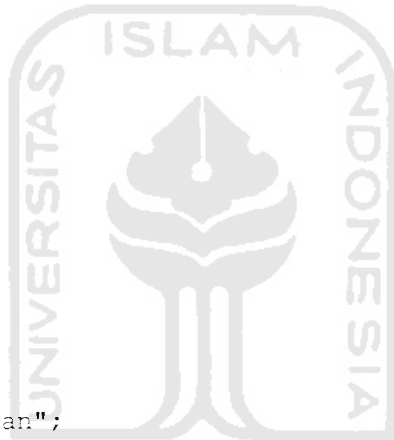
/>&nbsp;<input type="button" name="btncancel" id="cancel"
value="Batal" onclick="cancelinput()" />

<input type="hidden" name="id" id="id". value="<?=$id?>" />

</form>

<script language="javascript">
form=document.kelurahan;
if(form.act.value=="new") addnew();
</script>
</td>
</tr>
</table>
</div>
</td>
</tr>
<tr>
<td>
<?
$optional="kecamatan";
$optionalvalues="$kec";
require_once("include/keyword.php");
?>
</td>
</tr>
</table>
</body>
</html>
<?
}
?>
<?

```



## 9.5 Untuk Menampilkan dan Mengedit Data Perumahan

```
function buildname($bu,$nama) {
    switch((int)$bu) {
        case 1: return("P.T. $nama");
            break;
        case 2: return("C.V. $nama");
            break;
        case 3: return("PERUM $nama");
            break;
    }
}
require_once("include/checklogin.php");
$tabel="perumahan";
if($logged) {
    $id=0;
    $act="";
    $kec=0;
    $kel=0;

    $rs_kecamatan=mysql_query("SELECT id,nama FROM kecamatan ORDER
BY nama");

    if(!empty($_GET['id'])) $id=(int)$_GET['id'];
    if(!empty($_GET['act'])) $act=$_GET['act'];
    if(!empty($_GET['fkecamatan'])) $kec=(int)$_GET['fkecamatan'];

    $rs_kelurahan=mysql_query("SELECT id,id_kecamatan,nama FROM
kelurahan ORDER BY nama");

    if(!empty($_GET['fkelurahan']))
        $kel=(int)$_GET['fkelurahan'];

    $dev=0;

    $rs_developer=mysql_query("SELECT id,id_bu,nama FROM developer
ORDER BY nama");

    if(!empty($_GET['fdeveloper'])) $dev=(int)$_GET['fdeveloper'];
```



```
$optional="";

$optionalvalues="";

$ssql="SELECT perumahan.*,kecamatan.nama AS
kecamatan,kelurahan.nama AS kelurahan,".

"developer.id_bu,developer.nama AS developer FROM (perumahan
INNER JOIN developer ".

"ON perumahan.id_developer=developer.id) INNER JOIN (kelurahan
INNER JOIN kecamatan ".

"ON kelurahan.id_kecamatan=kecamatan.id) ON
perumahan.id_kelurahan=kelurahan.id ";

$where="WHERE ";

if($kel>0) {
    $optional="fkelurahan";
    $optionalvalues=$kel;
    $where.="kelurahan.id='$kel'";
} elseif($kec>0) {
    $where="kecamatan.id='$kec'";
    if(strlen($optional)>3) $optional.="";
    if(strlen($optionalvalues)>0) $optionalvalues.="";
    $optional.="fkecamatan";
    $optionalvalues.=$kec;
}

if($dev>0) {
    if(strlen($where)>6) $where.=" AND ";
    $where.="perumahan.id_developer='$dev'";
    if(strlen($optional)>3) $optional.="";
    if(strlen($optionalvalues)>0) $optionalvalues.="";
    $optional.="fdeveloper";
    $optionalvalues.=$dev;
}

if(strlen($where)>6) $sql.=$where;
```

```

        $rs_perumahan=mysql_query("$sql ORDER BY
kecamatan, kelurahan, developer, nama");

?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1" />

<title>Data Perumahan</title>

<link href="css/global.css" rel="stylesheet" type="text/css" />

<style type="text/css">

<!--
body {
    background-color: #FFFFFF;
    margin-left: 0px;
    margin-top: 0px;
    margin-right: 0px;
    margin-bottom: 0px;
}
-->

</style></head>

<script language="javascript">
function kecamatanchange () {
    form=document.perumahan;
    jml=0;
    while(form.kelurahan.length>0) form.kelurahan.remove(0);
    for(i=0;i<KELURAHAN.length;i++) {
        cari=form.kecamatan.options[form.kecamatan.selectedIndex].valu
e+"@";
        kel=KELURAHAN[i];
        if(form.kecamatan.selectedIndex==0 || kel.indexOf(cari)==0) {

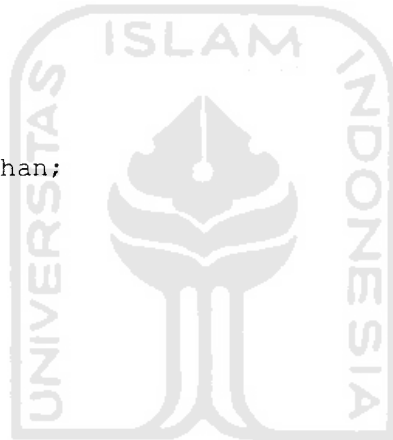
```



```

        kel=kel.substr(cari.length);
        j=kel.indexOf("|");
        option=document.createElement("OPTION");
        option.value=kel.substr(0,j);
        option.text=kel.substr(j+1);
        if(isIE) { form.kelurahan.add(option); }
        else if(isNN) { form.kelurahan.add(option,null); }
    }
    if (form.kelurahan.length>0) form.kelurahan.selectedIndex=0;
}
}
function addnew() {
var form=document.perumahan;
kecamatanchange();
form.nama.value="";
form.id.value="0";
form.act.value="new";
if(document.getElementById("datainput")) {
    document.getElementById("datainput").style.visibility="visible";
    form.nama.focus();
}
}
function edit(id,kel,dev,nama) {
    addnew();
    var form=document.perumahan;
    // alert(kel);
    for(i=form.kelurahan.length-1;i>=0;i--) {
        //alert(form.kelurahan.options[i].value);
        form.kelurahan.options[i].selected=(form.kelurahan.options[i].value==kel);
    }
}

```



```

        if(form.kelurahan.options[i].selected) break;
    }
    for(i=form.developer.length-1;i>=0;i--) {
        form.developer.options[i].selected=(form.developer.options[i].
value==dev);
        if(form.developer.options[i].selected) break;
    }
    form.id.value=id;
    form.nama.value=nama;
    form.act.value="edit";
    form.nama.select();
}
function cancelinput() {
var form=document.perumahan;
if(document.getElementById("datainput") && form.nama) {
    form.nama.value="";
    form.act.value="";
    document.getElementById("datainput").style.visibility="hidden"
}
}
function removeitem(id,nama) {
    if (window.confirm("Yakin perumahan '"+nama+"' akan dihapus?")) {
        var form=document.perumahan;
        form.id.value=id;
        form.act.value="remove";
        form.submit();
    }
}
function checkinput() {
    var form=document.perumahan;
    if(form.act.value=="new" || form.act.value=="edit") {

```

```
if(form.kelurahan.length<1) {
    alert("Kelurahan masih kosong");
    return;
}
if(form.kelurahan.selectedIndex<0) {
    alert("Pilih salah satu kelurahan");
    form.kelurahan.focus();
    return;
}
if(form.developer.selectedIndex<0) {
    alert("Developer masih kosong");
    form.developer.focus();
    return;
}
if(form.nama.value=="") {
    window.alert("Nama perumahan masih kosong");
    form.nama.focus();
    return;
}
} else if(form.act.value!="remove") return;

if(form.kelurahan.selectedIndex>=0)
form.kelurahan.value=form.kelurahan.options[form.kelurahan.selectedIndex].value;

if(form.developer.selectedIndex>=0)
form.developer.value=form.developer.options[form.developer.selectedIndex].value;

form.action="saveperumahan.php";
form.method="post";
form.target="_self";
form.submit();
}

function filterkecamatan() {
```



```

result="";

form=document.perumahan;

if(form.fkecamatan.selectedIndex>0)
result+"&fkecamatan="+form.fkecamatan.options[form.fkecamatan.selectedIndex].value;

return(result);
}

function filterkelurahan() {

result="";

form=document.perumahan;

if(form.fkelurahan.selectedIndex>0)
result+"&fkelurahan="+form.fkelurahan.options[form.fkelurahan.selectedIndex].value;

return(result);
}

function filterdeveloper() {

result="";

form=document.perumahan;

if(form.fdeveloper.selectedIndex>0)
result+"&fdeveloper="+form.fdeveloper.options[form.fdeveloper.selectedIndex].value;

return(result);
}

function selectradio() {

form=document.perumahan;

radio=form.id_perumahan;

id=0;

try {

if(radio.length>0) {

for(i=0;i<radio.length;i++) {

if(radio[i].checked) {

id=radio[i].value;

break;
}
}
}
}
}
}

```



```

        }
    }
} else {
    if (radio.checked) id=radio.value;
}
} catch(e) {
    if (radio.checked) id=radio.value;
}
if (id > 0) {
    url="perumahan.php?admin=yes&uid="+form.uid.value+"&id="+id+filterkecamatan()+filterkelurahan()+filterdeveloper;
    location.href=url;
}
}
function onfilter() {
    location.href="perumahan.php?admin=yes&uid="+document.perumahan.uid.value+filterkecamatan()+filterkelurahan()+filterdeveloper();
}
</script>
<body>
<form name="perumahan" id="perumahan"
action="javascript:checkinput()">
<input type="hidden" name="admin" id="admin" value="yes" />
<input type="hidden" name="uid" id="uid" value="<?=$_GET['uid']?>" />
<input type="hidden" name="act" id="act" value="<?=$_act?>" />
<table width="700" border="0" align="center" cellpadding="2" cellspacing="2">
<tr>
<td class="blueHeader1">Data Perumahan </td>
</tr>

```

```

<tr>
  <td>
    <table width="100%" border="0" align="center" cellpadding="2"
cellspacing="2" class="blackongray01">
      <tr valign="top">
        <td width="17%" align="right">Pilih Kecamatan </td>
        <td width="34%"><select name="fkecamatan"
class="normal01" id="fkecamatan" onchange="onfilter()">
          <option value="0">--Pilih salah satu--</option>
<?
  while($kecamatan=mysql_fetch_row($rs_kecamatan)) {
    if($kecamatan[0]==$kec) { $selected="selected"; }
    else { $selected=""; }
?>
    <option value="<?=$kecamatan[0]?>"
<?=$selected?><?=$kecamatan[1]?></option>
<?
  }
  mysql_data_seek($rs_kecamatan,0);
?>
    </select>      </td>
        <td width="16%" align="right">Pilih Kelurahan </td>
        <td width="33%"><select name="fkelurahan" id="fkelurahan"
class="normal01" onchange="onfilter()">
          <option value="0">--Pilih salah satu--</option>
<?
  while($kelurahan=mysql_fetch_row($rs_kelurahan)) {
    if($kec==0 || $kelurahan[1]==$kec) {
      if($kel==$kelurahan[0]) { $selected="selected"; }
      else { $selected=""; }
    }
?>

```

```

                <option value="<?=$kelurahan[0]?>"
<?=$selected?>><?=$kelurahan[2]?></option>
<?
        }
    }
    mysql_data_seek($rs_kelurahan,0);
?>
        </select>        </td>
</tr>
<tr>
        <td align="right" valign="top">Pilih
<em>Developer</em>&nbsp;   </td>
        <td colspan="3" valign="top"><select name="fdeveloper"
class="normal01" id="fdeveloper" onchange="onfilter()">
        <option value="0">--Pilih salah satu--</option>
<?
while($developer=mysql_fetch_row($rs_developer)) {
        $nama=buildname($developer[1],$developer[2]);
        if($dev==$developer[0]) { $selected="selected"; }
        else { $selected=""; }
?>
                <option value="<?=$developer[0]?>"
<?=$selected?>><?=$nama?></option>
<?
        }
    mysql_data_seek($rs_developer,0);
?>
        </select>
        </td>
        </tr>
</table>
</td>

```

```

</tr>

<tr>

    <td><table width="99%" border="0" align="center" cellpadding="2"
cellspacing="2">

        <tr>

            <th width="7%" class="whiteonblack04" scope="col">No</th>

            <th width="14%" class="whiteonblack04"
scope="col">Kecamatan</th>

            <th width="14%" class="whiteonblack04"
scope="col">Kelurahan</th>

            <th width="33%" class="whiteonblack04"
scope="col"><em>Developer</em></th>

            <th class="whiteonblack04" scope="col">Perumahan</th>

        </tr>

        <tr class="normal">

            <td colspan="5"><input type="button"
name="btnrefresh" id="btnrefresh" value="Refresh"
onclick="location.reload()" /></td>

        </tr>
<?

if(mysql_num_rows($rs_perumahan)<1) {
?>

<tr>

    <td align="left" valign="top">&nbsp;</td>

    <td colspan="4" align="left" valign="top">&nbsp;</td>

</tr>

<?

    } else {

        $i=1;

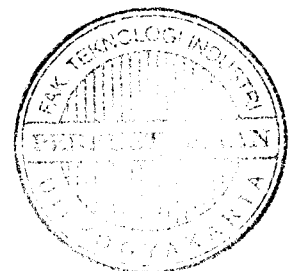
        $ganjil=true;

?>

    <tr>

        <td colspan="5" align="center">

```



```

        <div id="listkelurahan" style="height:160px;
width:720px; overflow: auto; border: solid 1px #CCCCCC;">
        <table width="100%" border="0" cellpadding="2"
cellspacing="1" align="center">
<?
    while($perumahan=mysql_fetch_array($rs_perumahan,MYSQL_ASSOC))
    {
        if($id<1) $id=$perumahan['id'];
        if($ganjil) {
            $class="normal01";
        } else {
            $class="blackonblue01";
        }
        $editbtn="edit$i";
        $removebtn="hapus$i";
        if((int)$perumahan['id']==(int)$id) {
            $checked=" checked";
        } else {
            $checked="";
        }
        $namadev=buildname($perumahan['id_bu'],$perumahan['developer']
    );
    ?>

        <tr class="<?=$class?>">
            <td width="7%" align="right"
valign="top"><?=$i?>.</td>
            <td width="15%" align="left"
valign="top"><?=$perumahan['kecamatan']?></td>
            <td width="14%" align="left"
valign="top"><?=$perumahan['kelurahan']?></td>
            <td width="33%" align="left"
valign="top"><?=$namadev?></td>
            <td width="31%" align="left" valign="top"><input
type="radio" name="id_perumahan" id="id_perumahan"
value="<?=$perumahan['id']?>" <?=$checked?>
onclick="selectradio()" />&nbsp;&nbsp;<?=$perumahan['nama']?></td>

```

```

        </tr>

        <tr>

                <td colspan="5" align="right"><input type="button"
name="<?=$editbtn?>" id="<?=$editbtn?>" value="Edit"
onclick="edit('<?=$perumahan['id']?>', '<?=$perumahan['kelurahan']?>'
, '<?=$perumahan['id_developer']?>', '<?=$perumahan['nama']?>')"
/>&nbsp;<input type="button" name="<?=$removebtn?>"
id="<?=$removebtn?>" value="Hapus"
onclick="removeitem('<?=$perumahan['id']?>', '<?=$perumahan['nama']?>
')" /></td>

        </tr>

<?

                $i++;
                $ganjil=!$ganjil;
        }
?>

        </table>

        </div>

        </tr>

<?

        }

?>

        </table></td>

</tr>

<tr>

        <td><input name="BtnTambah" type="button" id="BtnTambah"
value="Tambah Data Perumahan" onclick="addnew()" /></td>

</tr>

<tr>

        <td><div id="datainput" style="visibility:hidden">

                <table width="90%" align="center" cellpadding="2"
cellspacing="1" class="blackonbrown02">

                        <tr>

                                <td width="45%" align="right">Developer&nbsp;  </td>

```



```

        <td width="55%"><select name="developer"
id="developer" class="normal01" size="1">
<?
    while($developer=mysql_fetch_row($rs_developer)) {
        $nama=buildname($developer[1],$developer[2]);
    ?>
        <option
value="<?=$developer[0]?>"><?=$nama?></option>
<?
    }
?>
        </select>
    </td>
</tr>
<tr>
    <td width="45%" align="right">Nama&nbsp;</td>
    <td width="55%"><input type="text" name="nama"
id="nama" class="normal" size="25" maxlength="25" value="" />
</tr>
    <td width="45%" align="right">Kecamatan&nbsp;</td>
    <td width="55%"><select name="kecamatan"
id="kecamatan" class="normal01" size="1"
onchange="kecamatanchange()">
        <option value="0">--Pilih salah satu--
    </option>
<?
    while($kecamatan=mysql_fetch_row($rs_kecamatan)) {
    ?>
        <option
value="<?=$kecamatan[0]?>"><?=$kecamatan[1]?></option>
<?
    }
?>

```



```

        </select>
    </td>
</tr>
<tr>
    <td width="45%" align="right">Kelurahan&nbsp;</td>
    <td width="55%"><select name="kelurahan"
id="kelurahan" class="normal01" size="1">
        </select>
    </td>
</tr>
<tr>
    <td colspan="2" align="center"><input
type="submit" name="btncancel" id="btncancel" value="Batal"
/>&nbsp;<input type="button" name="btncancel" id="cancel"
value="Batal" onclick="cancelinput()" />
<input type="hidden" name="id" id="id" value="<?=$id?>" />
    </td>
</tr>
</form>
<script language="javascript">
isIE=(navigator.userAgent.indexOf("MSIE")>=0);
isNN=(navigator.userAgent.indexOf("Gecko"));
<?
    $strkelurahan="";
    while($kelurahan=mysql_fetch_row($rs_kelurahan)) {
        if(strlen($strkelurahan)>4) $strkelurahan.=",";
        $strkelurahan.="'".$kelurahan[1].@"'".$kelurahan[0]."|".$kelura
han[2]."'";
    }
    echo "KELURAHAN=new Array($strkelurahan);\n";
?>
form=document.perumahan;

```

```

if(form.act.value=="new") addnew();
</script>
        </table>
    </div>
</td>
</tr>
<tr>
    <td>
<?
        require_once("include/keyword.php");
?>
    </td>
</tr>
</table>
</body>
</html>
<?
}
?>
<?

```



## 9.6 Untuk Menampilkan dan Mengedit Data Developer

```

require_once("include/checklogin.php");
$tabel="developer";
if($logged) {
    $id=0;
    $act="";
    $kec=0;
    $kel=0;
    $rs_kecamatan=mysql_query("SELECT id,nama FROM kecamatan ORDER
BY nama");

```

```

if(!empty($_GET['id'])) $id=(int)$_GET['id'];

if(!empty($_GET['act'])) $act=$_GET['act'];

if(!empty($_GET['fkecamatan'])) $kec=(int)$_GET['fkecamatan'];

$rs_kelurahan=mysql_query("SELECT id,id_kecamatan,nama FROM
kelurahan ORDER BY nama");

if(!empty($_GET['fkelurahan']))

    $kel=(int)$_GET['fkelurahan'];

$optional="";

$optionalvalues="";

if($kel>0) {

    $sql="SELECT developer.*,kecamatan.nama AS
kecamatan,kelurahan.nama AS kelurahan FROM developer INNER JOIN
(kecamatan INNER JOIN kelurahan ON
kecamatan.id_kelurahan=kelurahan.id WHERE
developer.id_kelurahan='$kel' ORDER BY kecamatan,kelurahan,nama";

    $optional="fkelurahan";

    $optionalvalues=$kel;

} elseif($kec>0) {

    $sql="SELECT developer.*,kecamatan.nama AS
kecamatan,kelurahan.nama AS kelurahan FROM developer INNER JOIN
(kecamatan INNER JOIN kelurahan ON
kecamatan.id_kelurahan=kelurahan.id WHERE kecamatan.id='$kec' ORDER
BY kecamatan,kelurahan,nama";

    if(strlen($optional)>3) $optional.="";

    if(strlen($optionalvalues)>0) $optionalvalues.="";

    $optional="fkecamatan";

    $optionalvalues=$kec;

} else {

    $sql="SELECT developer.*,kecamatan.nama AS
kecamatan,kelurahan.nama AS kelurahan FROM developer INNER JOIN
(kecamatan INNER JOIN kelurahan ON
kecamatan.id_kelurahan=kelurahan.id ORDER BY
kecamatan,kelurahan,nama"

}

$rs_developer=mysql_query($sql);

```

```
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1" />

<title>Data Kecamatan</title>

<link href="css/global.css" rel="stylesheet" type="text/css" />

<style type="text/css">

<!--
body {
    background-color: #FFFFFF;
    margin-left: 0px;
    margin-top: 0px;
    margin-right: 0px;
    margin-bottom: 0px;
}
-->
</style></head>

<script language="javascript">

function kecamatanchange() {
    form=document.developer;
    jml=0;
    while(form.kelurahan.length>0) form.kelurahan.remove(0);
    for(i=0;i<KELURAHAN.length;i++) {
        cari=form.kecamatan.options[form.kecamatan.selectedIndex].valu
e+"@";
        kel=KELURAHAN[i];
        if(form.kecamatan.selectedIndex==0 || kel.indexOf(cari)==0) {
            kel=kel.substr(cari.length);
            j=kel.indexOf(".");
```



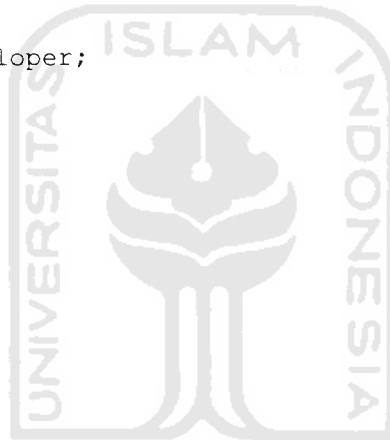
```

        option=document.createElement("OPTION");
        option.value=kel.substr(0,j);
        option.text=kel.substr(j+1);
        if(isIE) { form.kelurahan.add(option); }
        else if(isNN) { form.kelurahan.add(option,null); }
    }
    if (form.kelurahan.length>0) form.kelurahan.selectedIndex=0;
}
}

function addnew() {
var form=document.developer;
kecamatanchange();
form.nama.value="";
form.alamat.value="";
form.telp.value="";
form.id.value="0";
form.act.value="new";
if(document.getElementById "datainput") {
    document.getElementById "datainput".style.visibility="visible
";
    form.nama.focus();
}
}

function edit(id, kel, nama, bu, alamat, telp) {
    addnew();
    var form=document.developer;
    // alert(kel);
    for(i=form.kelurahan.length-1;i>=0;i--) {
        //alert(form.kelurahan.options[i].value);
        form.kelurahan.options[i].selected=(form.kelurahan.options[i].
value==kel);
    }
}

```



```
        if(form.kelurahan.options[i].selected) break;
    }
    form.id.value=id;
    form.nama.value=nama;
    form.id_bu.selectedIndex=bu-1;
    form.alamat.value=alamat;
    form.telp.value=telp;
    form.act.value="edit";
    form.nama.select();
}
function cancelinput() {
var form=document.developer;
if(document.getElementById("datainput") && form.nama) {
    form.nama.value="";
    form.alamat.value="";
    form.telp.value="";
    form.id.value="0";
    form.act.value="";
    document.getElementById("datainput").style.visibility="hidden"
}
}
function removeitem(id,nama) {
    if (window.confirm("Yakin developer '"+nama+"' akan dihapus?")) {
        var form=document.developer;
        form.id.value=id;
        form.act.value="remove";
        form.submit();
    }
}
function checkinput() {
```

```
var form=document.developer;

if(form.act.value=="new" || form.act.value=="edit") {
    if(form.kelurahan.length<1) {
        alert("Kelurahan masih kosong");
        return;
    }
    if(form.kelurahan.selectedIndex<0) {
        alert("Pilih salah satu kelurahan");
        form.kelurahan.focus();
        return;
    }
    if(form.nama.value=="") {
        window.alert("Nama developer masih kosong");
        form.nama.focus();
        return;
    }
    if(form.alamat.value=="") {
        window.alert("Alamat masih kosong");
        form.alamat.focus();
        return;
    }
    if(form.telp.value=="") {
        window.alert("No. telp masih kosong");
        form.telp.focus();
        return;
    }
} else if(form.act.value!="remove") return;

if(form.kelurahan.selectedIndex>=0)
form.kelurahan.value=form.kelurahan.options[form.kelurahan.selectedIndex].value;

form.action="savedeveloper.php";
```

```
form.method="post";
form.target="_self";
form.submit();
}
function selectradio() {
form=document.developer;
radio=form.id_developer;
id=0;
try {
    if(radio.length>0) {
        for(i=0;i<radio.length;i++) {
            if(radio[i].checked) {
                id=radio[i].value;
                break;
            }
        }
    } else {
        if(radio.checked) id=radio.value;
    }
} catch(e) {
    if(radio.checked) id=radio.value;
}
if(id>0) {
    url="developer.php?admin=yes&uid="+form.uid.value+"&id="+id;

    if(form.fkecamatan.selectedIndex>0)
url+="&fkecamatan="+form.fkecamatan.options[form.fkecamatan.selected
Index].value;

    if(form.fkelurahan.selectedIndex>0)
url+="&fkelurahan="+form.fkelurahan.options[form.fkelurahan.selected
Index].value;

    location.href=url;
```



```

    }
}

function fkecamatanchange() {
    form=document.developer;

    url="developer.php?admin=yes&uid="+form.uid.value;

    if(form.fkecamatan.selectedIndex>0)
url+="&fkecamatan="+form.fkecamatan.options[form.fkecamatan.selected
Index].value;

    location.href=url;
}

function fkelurahanchange() {
    form=document.developer;

    url="developer.php?admin=yes&uid="+form.uid.value;

    if(form.fkecamatan.selectedIndex>0)
url+="&fkecamatan="+form.fkecamatan.options[form.fkecamatan.selected
Index].value;

    if(form.fkelurahan.selectedIndex>0)
url+="&fkelurahan="+form.fkelurahan.options[form.fkelurahan.selected
Index].value;

    location.href=uri;
}
</script>

<body>

<form name="developer" id="developer"
action="javascript:checkinput()">

<input type="hidden" name="admin" id="admin" value="yes" />

<input type="hidden" name="uid" id="uid" value="<?=$_GET['uid']?>"
/>

<input type="hidden" name="act" id="act" value="<?=$act?>" />

<table width="720" border="0" align="center" cellpadding="2"
cellspacing="2">

    <tr>

        <td class="blueHeader1">Data Developer </td>

    </tr>

```

```

<tr>
  <td>
    <table width="100%" border="0" align="center" cellpadding="2"
cellspacing="2" class="blackongray01">
      <tr valign="top">
        <td width="17%" align="right">Pilih Kecamatan </td>
        <td width="34%"><select name="fkecamatan"
class="normal01" id="fkecamatan" onchange="fkecamatanchange()">
          <option value="0">--Pilih salah satu--</option>
<?
  while($kecamatan=mysql_fetch_row($rs_kecamatan)) {
    if($kecamatan[0]==$kec) { $selected="selected"; }
    else { $selected=""; }
?>
          <option value="<?=$kecamatan[0]?>"
<?=$selected?><?=$kecamatan[1]?></option>
<?
    }
    mysql_data_seek($rs_kecamatan,0);
?>
          </select>          </td>
        <td width="16%" align="right">Pilih Kelurahan </td>
        <td width="33%"><select name="fkelurahan" id="fkelurahan"
class="normal01" onchange="fkelurahanchange()">
          <option value="0">--Pilih salah satu--</option>
<?
  while($kelurahan=mysql_fetch_row($rs_kelurahan)) {
    if($kec==0 || $kelurahan[1]==$kec) {
      if($kel==$kelurahan[0]) { $selected="selected"; }
      else { $selected=""; }
    }
?>

```

```

        <option value="<?=$kelurahan[0]?>"
<?=$selected?>><?=$kelurahan[2]?></option>
<?
    }
}
mysql_data_seek($rs_kelurahan,0);
?>

</select>
</td>
</tr>
</table>
</td>
</tr>
<tr>
    <td><table width="99%" border="0" align="center" cellpadding="2"
cellspacing="2">
    <tr>
        <th width="7%" class="whiteonblack04" scope="col">No</th>
        <th width="14%" class="whiteonblack04"
scope="col">Kecamatan</th>
        <th width="14%" class="whiteonblack04"
scope="col">Kelurahan</th>
        <th width="33%" class="whiteonblack04"
scope="col">Developer</th>
        <th width="20%" class="whiteonblack04"
scope="col">Alamat</th>
        <th width="12%" class="whiteonblack04" scope="col">Telp</th>
    </tr>
    <tr class="normal">
        <td colspan="6"><input type="button"
name="btnrefresh" id="btnrefresh" value="Refresh"
onclick="location.reload()" /></td>
    </tr>
<?

```



```

if(mysql_num_rows($rs_developer)<1) {
?>
<tr>
    <td align="left" valign="top">&nbsp;</td>
    <td colspan="5" align="left" valign="top">&nbsp;</td>
</tr>
<?
    } else {
        $i=1;
        $ganjil=true;
?>
<tr>
    <td colspan="6" align="center">
        <div id="listkelurahan" style="height:160px;
width:720px; overflow: auto; border: solid 1px #CCCCCC;">
            <table width="100%" border="0" cellpadding="2"
cellspacing="1" align="center">
<?
                while($developer=mysql_fetch_array($rs_developer,MYSQL_ASSOC))
                {
                    if($id<1) $id=$developer['id'];
                    if($ganjil) {
                        $class="normal01";
                    } else {
                        $class="blackonblue01";
                    }
                    $editbtn="edit$i";
                    $removebtn="hapus$i";
                    if((int)$developer['id']==(int)$id) {
                        $checked=" checked";
                    } else {
                        $checked="";

```

```

    }

    $sid_bu=(int)$developer['id_bu'];

    switch($sid_bu) {

        case 1: $nama="P.T. ".$developer['nama'];

            break;

        case 2: $nama="C.V. ".$developer['nama'];

            break;

        case 3: $nama="PERUM. ".$developer['nama'];

            break;

    }

?>

    <tr class="<?=$class?>">

        <td width="7%" align="right"

valign="top"><?=$i?>.</td>

        <td width="15%" align="left"

valign="top"><?=$developer['kecamatan']?></td>

        <td width="14%" align="left"

valign="top"><?=$developer['kelurahan']?></td>

        <td width="34%" align="left" valign="top"><input

type="radio" name="id_developer" id="id_developer"

value="<?=$developer['id']?>" <?=$checked?>

onclick="selectradio()" />&nbsp;<?=$nama?></td>

        <td width="20%" align="left"

valign="top"><?=$developer['alamat']?></td>

        <td width="10%" align="left"

valign="top"><?=$developer['telp']?></td>

    </tr>

    <tr>

        <td colspan="6" align="right"><input type="button"

name="<?=$editbtn?>" id="<?=$editbtn?>" value="Edit"

onclick="edit('<?=$developer['id']?>','<?=$developer['id_kelurahan']

?>','<?=$developer['nama']?>','<?=$developer['id_bu']?>','<?=$develope

r['alamat']?>','<?=$developer['telp']?>')" />&nbsp;<input

type="button" name="<?=$removebtn?>" id="<?=$removebtn?>"

value="Hapus"

onclick="removeitem('<?=$developer['id']?>','<?=$developer['nama']?>

')" /></td>

    </tr>

```

```

<?
        $i++;
        $ganjil=!$ganjil;
    }
?>

</table>

</div>          </td>

</tr>

<?
    }
?>

</table></td>

</tr>

<tr>

    <td><input name="BtnTambah" type="button" id="BtnTambah"
value="Tambah Data Developer" onclick="addnew()" /></td>

</tr>

<tr>

    <td><div id="datainput" style="visibility:hidden">

        <table width="90%" align="center" cellpadding="2"
cellspacing="1" class="blackonbrown02">

            <tr>

                <td width="45%" align="right">Badan
Usaha&nbsp;  </td>

                <td width="55%"><select name="id_bu" id="id_bu"
class="normal01" size="1">

                    <option value="1" selected>P.T.</option>

                    <option value="2">C.V.</option>

                    <option value="3">PERUM</option>

                </select>

            </td>

        </tr>

```

```

        <tr>
            <td width="45%" align="right">Developer </td>
            <td width="55%"><input type="text" name="nama"
id="nama" class="normal" size="25" maxlength="25" value="" />
            </td>
        </tr>
        <tr>
            <td width="45%" align="right">Alamat </td>
            <td width="55%"><input type="text" name="alamat"
id="alamat" class="normal" size="30" maxlength="60" value="" />
        </tr>
        <tr>
            <td width="45%" align="right">Kecamatan </td>
            <td width="55%"><select name="kecamatan"
id="kecamatan" class="normal01" size="1"
onchange="kecamatanchange()" >
                <option value="0">--Pilih salah satu--
            </option>
            <?
                while($kecamatan=mysql_fetch_row($rs_kecamatan)) {
            ?>
                <option
value="<?=$kecamatan[0]?>"><?=$kecamatan[1]?></option>
            <?
                }
            ?>
        </select>
            </td>
        </tr>
        <tr>
            <td width="45%" align="right">Kelurahan </td>
            <td width="55%"><select name="kelurahan"
id="kelurahan" class="normal01" size="1">
                </select>
            </td>
        </tr>
    </table>
</pre>

```

```

        </td>
    </tr>
    <tr>
        <td width="45%" align="right">Telp&nbsp;  </td>
        <td width="55%"><input type="text" name="telp"
id="telp" class="normal" size="15" maxlength="15" />
    </td>
    </tr>
    <tr>
        <td colspan="2" align="center"><input
type="submit" name="btnsave" id="btnsave" value="Simpan"
/>&nbsp;  <input type="button" name="btncancel" id="cancel"
value="Batal" onclick="cancelinput()" />
<input type="hidden" name="id" id="id" value="<?=$id?" />
        </td>
    </tr>
</form>
<script language="javascript">
isIE=(navigator.userAgent.indexOf("MSIE")>=0);
isNN=(navigator.userAgent.indexOf("Gecko"));
<?
    $strkelurahan="";
    while($kelurahan=mysql_fetch_row($rs_kelurahan)) {
        if(strlen($strkelurahan)>4) $strkelurahan.=",";
        $strkelurahan.=" ".$kelurahan[1].@" ".$kelurahan[0].|" ".$kelura
han[2]." ";
    }
    echo "KELURAHAN=new Array($strkelurahan);\n";
?>
form=document.developer;
if(form.act.value=="new") addnew();
</script>

```



```

                </table>
            </div>
        </td>
    </tr>
    <tr>
        <td>
<?
        require_once("include/keyword.php");
?>
        </td>
    </tr>
</table>
</body>
</html>
<?
}
?>
<?

```



## 9.7 Untuk Membuat atau Membentuk Halaman Pencarian

```

$logged=false;

if(isset($_GET['admin']) && isset($_GET['uid']))
require_cnce("include/checklogin.php");

$keyword="";

$mode="simple";

if(!empty($_GET['keyword'])) $keyword=$_GET['keyword'];

if(!empty($_GET['mode'])) $mode=$_GET['mode'];

if($mode=="simple") {
    $txtMode="Mode Ahli";
} elseif($mode=="advanced") {
    $txtMode="Mode Sederhana";
}

```

```
}
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-
8859-1" />
<title>Pencarian Data</title>
<style type="text/css">
<!--
body {
    background-color: #FFFFFF;
}
-->
</style>
<link href="css/global.css" rel="stylesheet" type="text/css" />
<script language="javascript">
function setMode() {
    form=document.searchform;
    URL="search.php?";
    if(form.mode.value=="simple") {
        form.mode.value="advanced";
    } else if(form.mode.value=="advanced") {
        form.mode.value="simple";
    }
    if(form.keyword.value!="")
    URL+="keyword="+encodeURIComponent(form.keyword.value)+"&";
    //http://localhost/hendra/search.php?keyword=merapi&mode=simpl
e&
    if(form['admin']!=null && form['uid']!=null) {
```

```

        if(form.admin.value!="" && form.uid.value!="")
URL+="admin=yes&uid="+form.uid.value+"&";

    }

    URL+="mode="+form.mode.value;

    location.href=URL;

}

function searchNow() {

    form=document.searchform;

    URL="search.php?";

    if(form.keyword.value!="") {

        URL+="keyword="+encodeURIComponent(form.keyword.value)+"&";

    }

    if(eval("form.admin") && eval("form.uid")) {

        if(form.admin.value!="" && form.uid.value!="") {

            if(URL.lastIndexOf("&")<URL.length-1) URL+="&";

            URL+="admin=yes&uid="+form.uid.value+"&";

        }

    }

    URL+="mode="+form.mode.value+"&";

    if(form['keytype']!=null) {

        if(URL.lastIndexOf("&")<URL.length-1) URL+="&";

        URL+="keytype="+form.keytype.value;

    }

    if(form['exact']!=null) {

        if(form.exact.checked) {

            if(URL.lastIndexOf("&")<URL.length-1) URL+="&";

            URL+="exact="+form.exact.value;

        }

    }

    if(URL.lastIndexOf("&")==URL.length-1)
URL=URL.substr(0,URL.length-1);

```



```

        $i=(int)$key;

        $i++;

        $selected="";

        if(!empty($_GET['keytype'])) {
            if((int)$_GET['keytype']==$i)
                $selected="selected";
        }

        echo "\t\t<option
value=\"\$i\|\"$selected>$value</option>\n";

    }

    $checked="";

    if(!empty($_GET['exact'])) {
        $linkmode.="&exact=$_GET[exact]";
        if(strtolower($_GET['exact'])=="true") $checked="
checked";
    }

?>

</select></td>

    <td width="55%" class="normal"><input type="checkbox"
name="exact" id="exact" value="true" <?=$checked?>/>

    Hasil harus sama persis dengan <em>keyword</em></td>

</tr>

<?
}
?>

<tr>

    <td width="17%"><input name="cari" type="submit" id="cari"
value="Cari Sekarang" /></td>

    <td width="28%">&nbsp;</td>

    <td width="55%"><input name="Advanced" type="button"
id="Advanced" value="<?=$txtMode?>" onclick="setMode()" /></td>

</tr>

<?

```

```
$linkadmin="";
if($logged) {
?>
<input type="hidden" name="admin" id="admin" value="yes" />
<input type="hidden" name="uid" id="uid" value="<?=$_GET['uid']?>"
/>
<?
    $linkadmin="admin=yes&uid=$_GET[uid]";
} else require_once("include/myconfig.php");
if(!empty($_GET['keyword'])) require_once("include/result.php");
?>
</table>
</form>
<script language="javascript">
document.searchform.keyword.focus();
document.searchform.keyword.select();
</script>
</body>
</html>
```

