

• DAHLAN ABDULLAH • PUTU DODDY HEKA ARDANA • QURROTUL AINI
• MEINARINI CATUR UTAMI • ELSA YUNIARTI • IWAN ADHICANDRA



Sistem Informasi
REHABILITASI
DAN
REKONSTRUKSI
Pascabencana Berbasis WEB



**DAHLAN ABDULLAH
PUTU DODDY HEKA ARDANA
QURROTUL AINI
MEINARINI CATUR UTAMI
ELSA YUNIARTI
IWAN ADHICANDRA**

**SISTEM INFORMASI REHABILITASI DAN
REKONSTRUKSI PASCA BENCANA
BERBASIS WEB**

Diterbitkan Oleh:



**SEFA MEDIA
UTAMA**

CV. SEFA MEDIA UTAMA - ACEH

2023

SISTEM INFORMASI REHABILITASI DAN REKONSTRUKSI PASCA BENCANA BERBASIS WEB

Penulis : **DAHLAN ABDULLAH
PUTU DODDY HEKA ARDANA
QURROTUL AINI
MEINARINI CATUR UTAMI
ELSA YUNIARTI
IWAN ADHICANDRA**

Hak Cipta © 2023 pada Penulis

Hak Cipta dilindungi undang-undang. Dilarang memperbanyak atau memindahkan sebagian atau seluruh isi buku ini dalam bentuk apapun baik secara elektronik maupun mekanis, termasuk memfotokopi, merekan atau dengan sistem penyimpanan lainnya, tanpa izin tertulis dari penerbit dan penulis

Penerbit:

SEFA MEDIA UTAMA

Jl. Gerudong Pasee Aceh Utara

<http://sefamediautama.id/> Telp. 085260363550

Cetakan I : Maret 2023

ISBN: 978-623-09-2740-9

Halaman. 120

Ukuran 16,8 x 23 cm

KATA PENGANTAR

Assalamu'alaikum Wr. Wb.

Alhamdulillah wa syukurillah, puji syukur penulis panjatkan kepada Allah Subhanahu wata'ala yang telah memberikan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan penelitian dengan tepat pada waktunya, kemudian shalawat dan salam penulis sanjung sajikan kepada Nabi Besar Muhammad SAW yang telah membawa umatnya dari alam kebodohan ke alam yang penuh ilmu pengetahuan.

Buku ini berjudul "**Sistem Informasi Rehabilitasi dan Rekonstruksi Pasca Bencana Berbasis Web**". Banyak ilmu serta pengalaman baru dan berharga penulis peroleh dari kegiatan buku ini. Oleh karena itu, penulis ucapkan terimakasih banyak atas segala bantuan dan dukungan sehingga kegiatan penyusunan buku ini berjalan dengan lancar.

Penulis menyadari bahwa ilmu dan pengalaman yang penulis miliki belum sempurna. Oleh karena itu, kritik dan saran sangat penulis harapkan dari para pembaca buku ini. Semoga buku ini dapat memberikan ilmu dan informasi yang bermanfaat bagi penulis khususnya dan pembaca pada umumnya.

Wassalamu'alaikum Wr. Wb.

Lhokseumawe, 2023

Penulis
Dahlan Abdullah

DAFTAR ISI

	Halaman
KATA PENGANTAR	i
SINOPSIS	ii
DAFTAR ISI	iv
DAFTAR TABEL	vii
DAFTAR GAMBAR	viii
BAB I PENDAHULUAN	
1.1. Latar Belakang Masalah	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah	2
1.4. Tujuan Penulisan.....	2
1.5. Manfaat Penulisan	3
1.6. Metodologi Penulisan.....	3
1.7. Waktu dan Tempat Pelaksanaan	4
BAB II GAMBARAN UMUM INSTANSI	
2.1. Sejarah Singkat.....	5
2.2. Visi dan Misi.....	8
2.2.1. Visi.....	8
2.2.2. Misi.....	9
2.3. Struktur Organisasi.....	10
2.4. Pembagian Kerja.....	11
2.4.1. Kepala Pelaksana	11
2.4.2. Kepala Badan	12
2.4.3. Sekretariat.....	12
2.4.4. Sub Bagian Umum dan Kepegawaian.....	13
2.4.5. Sub Bagian Keuangan.....	13
2.4.6. Sub Bagian Program dan Pelaporan	14
2.4.7. Seksi Bidang Pencegahan dan Kesiapsiagaan	15
2.4.8. Seksi Bidang Kedaruratan dan Logistik	16
2.4.9. Seksi Rehabilitasi dan Rekonstruksi.....	17
2.4.10. Seksi Bidang Pemadam Kebakaran	19

BAB III LANDASAN TEORI

3.1. Rehabilitasi	20
3.2. Rekonstruksi	22
3.3. Bencana	25
3.3.1. Jenis-Jenis Bencana.....	26
3.3.2. Dampak Bencana	27
3.4. Pengertian Sistem.....	27
3.4.1. Karakteristik Sistem.....	28
3.4.2. Klasifikasi Sistem	29
3.4.3. Sistem yang Buruk	30
3.5. Pengertian Informasi	30
3.5.1. Jenis-Jenis Informasi	31
3.5.2. Kualitas Informasi	32
3.5.3. Siklus Hidup Informasi	32
3.6. Pengertian Sistem Informasi.....	33
3.6.1. Komponen Sistem Informasi.....	33
3.6.2. Teknik Memperoleh Informasi	35
3.7. Perancangan Sistem	37
3.7.1. Diagram Konteks.....	39
3.7.2. <i>Data Flow Diagram (DFD)</i>	39
3.7.3. <i>Entity Relationship Diagram (ERD)</i>	41
3.8. Tools Perangkat Lunak	44
3.8.1. <i>HTML, CSS, Javascript</i>	44
3.8.2. <i>Bootstrap</i>	53
3.8.3. <i>PHP</i>	55
3.8.4. <i>Website</i>	69
3.8.5. <i>Database MySQL</i>	69
3.8.6. <i>PHP MyAdmin</i>	75
3.8.7. <i>XAMPP</i>	76
3.8.8. <i>Text Editor</i>	82

BAB IV ANALISA DAN PEMBAHASAN

4.1. Analisis Sistem	86
4.1.1. Analisa Sistem Lama	86

4.1.2. Analisis Sistem Baru	87
4.2. Analisis Kebutuhan Non-Fungsional	87
4.2.1. Analisis Perangkat Keras (<i>Hardware</i>).....	87
4.2.2. Analisis Perangkat Lunak (<i>Software</i>)	88
4.3. Perancangan Sistem.....	88
4.3.1. <i>Diagram Konteks</i>	88
4.3.2. <i>Data Flow Diagram</i>	89
4.3.3. <i>Entity Relationship Diagram (ERD)</i>	99
4.4. Perancangan <i>Database</i>	100
4.4.1. Tabel Admin	100
4.4.2. Tabel Data Pascabencana.....	100
4.4.3. Tabel Data Permasalahan	100
4.4.4. Tabel Rencana Kerja	101
4.4.5. Tabel Laporan Kegiatan	101
4.5. Implementasi Sistem	101
BAB V PENUTUP	
5.1. Kesimpulan	109
5.2. Saran.....	109
DAFTAR PUSTAKA.....	110

DAFTAR TABEL

Tabel 3.1 Kode Warna	46
Tabel 3.2 Atribut Elemen <i>Body</i>	47
Tabel 3.3 Atribut Elemen <i>Font</i>	48
Tabel 3.4 Elemen Ragam Karakter	48
Tabel 3.5 Atribut Elemen <i>Image</i>	48
Tabel 3.6 Atribut Elemen Tabel	49
Tabel 3.7 Karakter Spesial Petik Ganda	59
Tabel 3.8 Operator Aritmatika	62
Tabel 3.9 Hasil Operator Aritmatika	62
Tabel 3.10 Operator Kombinasi Penugasan	63
Tabel 3.11 Operator Perbandingan.....	65
Tabel 3.12 Operator Logika	66
Tabel 3.13 Tipe Data <i>Character</i>	73
Tabel 4.1 Admin.....	100
Tabel 4.2 Data Pascabencana	100
Tabel 4.3 Data Isu	100
Tabel 4.4 Data Rencana Kerja	101
Tabel 4.5 Laporan Kegiatan	101

DAFTAR GAMBAR

Gambar 2.1 Struktur Organisasi	10
Gambar 3.1 Unsur-unsur Sistem.....	28
Gambar 3.2 <i>Entity</i>	40
Gambar 3.3 Aliran Data	40
Gambar 3.4 Proses.....	40
Gambar 3.5 Simpanan Data	41
Gambar 3.6 Entitas	42
Gambar 3.7 Atribut.....	42
Gambar 3.8 Hubungan	43
Gambar 3.9 <i>Link</i>	43
Gambar 3.10 Tahap Pertama <i>Instal Xampp</i>	79
Gambar 3.11 Tahap Kedua <i>Instal Xampp</i>	79
Gambar 3.12 Tahap Ketiga <i>Instal Xampp</i>	80
Gambar 3.13 Tahap Keempat <i>Instal Xampp</i>	80
Gambar 3.14 Tahap Kelima <i>Instal Xampp</i>	81
Gambar 3.15 Tahap Keenam <i>Instal Xampp</i>	81
Gambar 3.16 Tahap Ketujuh <i>Instal Xampp</i>	82
Gambar 3.17 Tahap Kedelapan <i>Instal Xampp</i>	82
Gambar 4.1 <i>Diagram Konteks</i>	89
Gambar 4.2 DFD Level 0.....	90
Gambar 4.3 DFD Level 1 Proses <i>Login</i>	91
Gambar 4.4 DFD Level 1 Proses Tambah Data Pasca Bencana	92
Gambar 4.5 DFD Level 1 Proses Tambah Data Isu	94
Gambar 4.6 DFD Level 1 Proses Tambah Data Rencana Kerja.	95
Gambar 4.7 DFD Level 1 Proses Tambah Data Laporan.....	97
Gambar 4.8 <i>Entity Relation Diagram</i>	99
Gambar 4.9 Halaman <i>Login</i>	102
Gambar 4.10 Halaman Kegiatan	102
Gambar 4.11 Halaman <i>Dashboard</i>	103
Gambar 4.12 Halaman Data Pascabencana	103
Gambar 4.13 Tambah Data Pascabencana.....	104
Gambar 4.14 Halaman Data Isu/Permasalahan	104

Gambar 4.15 Tambah Data Isu/Permasalahan	105
Gambar 4.16 Halaman Rencana Kerja	105
Gambar 4.17 Tambah Data Rencana Kerja.....	106
Gambar 4.18 Halaman Data Laporan	107
Gambar 4.19 Tambah Data Laporan.....	108
Gambar 4.20 Halaman Akun.....	108

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Di era *modern* ini, perkembangan teknologi tidak dapat dipungkiri semakin mempermudah untuk mendapatkan informasi yang cepat dan akurat. Yang banyak digunakan saat ini salah satunya sistem informasi berbasis *web* yang dimanfaatkan sebagai sarana untuk mempermudah, mempercepat dan mendapatkan keakuratan dalam memperoleh informasi. Pada era ini, banyak lembaga-lembaga yang memanfaatkan teknologi berbasis *web* sebagai penunjang kegiatan kerja yang menghasilkan informasi yang lebih fleksibel.

Untuk itu salah satu bidang yang perlu dikembangkan adalah bidang rehabilitasi dan rekonstruksi yang meliputi pengolahan data pascabencana. Sistem informasi rehabilitasi dan rekonstruksi ini dapat membantu dalam pendataan secara akurat dan tepat. Maka dari itu, penulis merancang suatu sistem untuk melakukan suatu kerja khususnya dalam penginputan data pascabencana di Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara.

Selama melaksanakan kegiatan Penelitian di Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara, penulis menemukan bahwa dalam penyampaian sistem informasi mengenai hal-hal yang terjadi pascabencana yaitu di bidang rehabilitasi dan rekonstruksi masih menggunakan cara konvensional dimana informasi mengenai laporan pascabencana masih diketik pada *file word* sehingga memakan banyak waktu dan juga kurang fleksibel. Untuk itulah dibutuhkan sistem yang berisi informasi tentang rehabilitasi dan rekonstruksi pascabencana yaitu dengan menggunakan pemrograman *php* berbasis *web*.

Berdasarkan pada permasalahan tersebut, penulis tertarik untuk meneliti dan membuat sistem informasi rehabilitasi dan rekonstruksi yang dapat membantu dan juga memudahkan Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara.

Berdasarkan permasalahan di atas, maka penulis mengangkat topik dengan judul “Sistem Informasi Rehabilitasi dan Rekonstruksi Pascabencana Berbasis *Web* pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara”.

1.2. Rumusan Masalah

Dari latar belakang di atas, maka rumusan masalah yang akan dibahas adalah bagaimana proses membangun dan mengimplementasikan sistem informasi rehabilitasi dan rekonstruksi pascabencana pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara.

1.3. Batasan Masalah

Adapun batasan masalah pada perancangan sistem ini, yaitu:

1. Perancangan sistem informasi rehabilitasi dan rekonstruksi pascabencana pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara.
2. Bahasa pemrograman menggunakan *PHP*.
3. Perancangan *database* menggunakan *MySQL*.
4. Perancangan sistem menggunakan *DFD*.

1.4. Tujuan Penulisan

Adapun tujuan dari penulisan ini, yaitu:

1. Untuk membangun sistem informasi rehabilitasi dan rekonstruksi pascabencana berbasis *web* pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara.

2. Menerapkan ilmu yang telah dipelajari selama bangku perkuliahan dan dapat menambah pengetahuan teknologi serta gambaran dunia kerja bagi mahasiswa.

1.5. Manfaat Penulisan

Manfaat dari penulisan penelitian tentang sistem informasi rehabilitasi dan rekonstruksi ini adalah:

1. Bagi Kantor Badan Penanggulangan Bencana Daerah, diharapkan sistem ini dapat mempermudah dalam pendataan di bagian rehabilitasi dan rekonstruksi dalam pengumpulan data pascabencana.
2. Mampu menganalisa dan merancang proses pembuatan sistem informasi rehabilitasi dan rekonstruksi pascabencana pada Kantor Penanggulangan Bencana Daerah Kabupaten Aceh Utara.
3. Menambah wawasan bagi pembaca dan penulis mengenai perancangan sistem informasi rehabilitasi dan rekonstruksi pascabencana.

1.6. Metodologi Penulisan

Adapun metode kepenulisan yang dilakukan oleh penulis yaitu sebagai berikut:

1. Studi Kepustakaan

Studi Kepustakaan yang penulis ambil meliputi buku-buku dan literatur-literatur yang berhubungan langsung dengan sistem tersebut. Media lain berupa internet yang berhubungan dengan sistem informasi.

2. Observasi

Dalam hal ini penulis melakukan observasi secara langsung untuk mempelajari, mengamati dan mengumpulkan data serta informasi dari objek yang sedang diteliti di Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara.

3. Wawancara

Wawancara yang dilakukan yaitu tanya jawab antara peneliti dan narasumber dengan tujuan untuk mendapatkan informasi mengenai penelitian yang dilakukan.

1.7. Waktu dan Tempat Pelaksanaan

Penelitian ini dilaksanakan pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara yang berlokasi di jalan Raya Banda Aceh-Medan Km.295 Landing-Lhoksukon, Kabupaten Aceh Utara. Waktu pelaksanaan kerja praktik ini selama 2 bulan pada tanggal 04 Juni 2021 sampai dengan 04 Agustus 2021. Dengan judul laporan yaitu Sistem Informasi Rehabilitasi dan Rekonstruksi Pascabencana Berbasis *Web* pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara.

BAB II

GAMBARAN UMUM INSTANSI

2.1. Sejarah Singkat

Pembentukan Badan Penanggulangan Bencana Daerah diawali dengan pembentukan BNPB (Badan Nasional Penanggulangan Bencana) sebagai induk perusahaan Badan Penanggulangan Bencana Daerah. Keduanya merupakan lembaga yang sama namun Badan Penanggulangan Bencana Daerah berada di skala daerah.

Sejarah terbentuknya Badan Nasional Penanggulangan Bencana tidak terlepas dari perkembangan penanggulangan bencana pada masa kemerdekaan dan perkembangan bencana alam seperti gempa bumi dahsyat di Samudra Hindia pada abad ke-20. Hal ini sangat dipengaruhi oleh situasi penanggulangan bencana, ruang lingkup dan latar belakang penanggulangan bencana.

Melihat kenyataan yang terjadi saat ini, berbagai bencana yang dipicu oleh kondisi geografis, geologis, hidrologis, dan demografis mendorong Indonesia menetapkan visi untuk membangun kemampuan negara dalam merespon bencana.

Wilayah Indonesia merupakan negara kepulauan terbesar di dunia. Wilayah ini juga terletak di antara benua Asia dan Australia, serta di antara Samudra Hindia dan Samudra Pasifik, dengan 17.508 pulau. Meski kepulauan ini kaya akan kekayaan alam dan keindahan yang luar biasa, perlukah masyarakat Indonesia mengetahui bahwa kepulauan ini memiliki 129 gunung berapi aktif yang disebut juga Cincin Api yang terletak di persimpangan tiga lempeng aktif dunia? Lempeng Indo-Australia, Eurasia dan Samudra Pasifik.

Cincin api dan perpotongan tiga lempeng tektonik menempatkan negara kepulauan pada risiko bencana alam. Di sisi lain, letak Indonesia yang berada pada kondisi tropis dan

hidrologis telah menimbulkan bencana alam lainnya seperti angin topan, curah hujan ekstrim, banjir, tanah longsor dan kekeringan. Tidak hanya pada kasus bencana alam sebagai ancaman, tetapi juga bencana non alam sering melanda tanah air seperti kebakaran hutan dan lahan, konflik sosial, maupun kegagalan teknologi.

Menghadapi ancaman bencana, pemerintah Indonesia berperan penting dalam membangun sistem penanggulangan bencana di tanah air. Pembentukan lembaga ini merupakan bagian dari sistem yang dilakukan dari waktu ke waktu. Organisasi ini telah ada sejak proklamasi kemerdekaan tahun 1945, dan perkembangan organisasi penanggulangan bencana dapat dibagi menurut periode waktu, sebagai berikut:

1. Tahun 1945-1966

Pemerintah Indonesia membentuk Badan Bantuan Keluarga Korban Perang (BPKKP). Badan tersebut didirikan pada tanggal 20 Agustus 1945, dengan fokus pada situasi perang pasca kemerdekaan di Indonesia. Misi dari badan ini adalah untuk membantu para korban perang dan keluarganya selama Perang Kemerdekaan.

2. Tahun 1966-1967

Pemerintah membentuk Badan Pertimbangan Penanggulangan Bencana Alam Pusat (BP2BAP) melalui Keputusan Presiden Nomor 256 Tahun 1966. Kepala lembaga ini adalah Menteri Sosial. Kegiatan BP2BAP berperan pada penanggulangan tanggap darurat dan bantuan korban bencana. Melalui keputusan ini, pengembangan paradigma penanggulangan bencana tidak hanya terfokus pada bencana buatan manusia, tetapi juga bencana alam.

3. Tahun 1967-1979

Frekuensi bencana alam terus meningkat. Perlunya penanganan bencana yang cermat dan terkoordinasi. Oleh karena itu, pada tahun 1967 Biro Kabinet mengeluarkan

Keputusan No. 14/U/KEP/I/1967 untuk membentuk Kelompok Koordinasi Nasional Penanggulangan Bencana Alam (TKP2BA).

4. Tahun 1979-1990

Pada periode ini, Kelompok Koordinasi Nasional Penanggulangan Bencana Alam (TKP2BA) ditingkatkan menjadi Badan Koordinasi Nasional Penanggulangan Bencana Alam (Bakornas PBA) yang diketuai oleh Menteri Koordinator Kesejahteraan Rakyat dan dibentuk sesuai dengan Keputusan Presiden Nomor 28 Tahun 1979. Kegiatan pengelolaan meliputi tahap pencegahan, penanggulangan kedaruratan dan rehabilitasi.

5. Tahun 1990-2000

Pada periode ini, bencana bukan hanya terjadi karena alam namun juga non alam serta sosial. Bencana non alam yang terjadi seperti kecelakaan transportasi, kegagalan teknologi, dan konflik sosial mewarnai pemikiran penanggulangan bencana. Karena hal tersebut maka penyempurnaan Badan Koordinasi Nasional Penanggulangan Bencana Alam menjadi Badan Koordinasi Nasional Penanggulangan Bencana (Bakornas PB). Melalui Keppres No. 43 Tahun 1990, ruang lingkup misi Bakornas PB diperluas, tidak hanya pada bencana alam, tetapi juga pada bencana non alam dan sosial. Keputusan Presiden Nomor 106 Tahun 1999 kembali menegaskan hal itu. Penanggulangan bencana memerlukan koordinasi antar departemen, antar pelaku dan antar disiplin yang terkoordinasi.

6. Tahun 2001-2005

Sebelumnya, Indonesia mengalami krisis multifaset. Bencana sosial yang terjadi di berbagai tempat telah menimbulkan masalah baru. Masalah-masalah tersebut memerlukan penanganan khusus karena berkaitan dengan perpindahan. Oleh karena itu, Bakornas PB kemudian berkembang menjadi Badan Koordinasi Nasional Penanggulangan Bencana dan Penanganan Pengungsi (Bakornas PBP). Kebijakan ini tertuang dalam Keputusan

Presiden Nomor 3 Tahun 2001 dan kemudian diperbarui dengan Keputusan Presiden Nomor 111 Tahun 2001.

7. Tahun 2005-2008

Gempa bumi dan tsunami yang terjadi di Aceh sekitar tahun 2004 menjadi pendorong Pemerintah Indonesia dan juga dunia meningkatkan pengelolaan penanggulangan. Melihat kondisi saat itu, pemerintah Indonesia mengeluarkan Peraturan Presiden Nomor 83 Tahun 2005 tentang Badan Koordinasi Nasional Penanganan Bencana (Bakornas PB). Badan ini berfungsi sebagai pendukung sistem unsur pelaksana penanggulangan bencana. Sejalan dengan itu, pengurangan risiko bencana menjadi perhatian utama.

8. Tahun 2008

Menanggapi sistem penanggulangan bencana saat itu, (UU 24 dan Perpres 8) pemerintah Indonesia sangat mementingkan pembentukan legalisasi, sistem dan anggaran. Setelah diterbitkannya Undang-Undang Nomor 24 tentang Penanggulangan Bencana pada tahun 2007, pemerintah selanjutnya menerbitkan Peraturan Presiden Nomor 8 Tahun 2008 tentang Badan Nasional Penanggulangan Bencana. Badan Nasional Penanggulangan Bencana terdiri dari penanggung jawab, unsur pembimbing penanggulangan bencana, dan unsur pelaksana penanggulangan bencana. Badan Nasional Penanggulangan Bencana mempunyai fungsi mengkoordinasikan pelaksanaan kegiatan penanggulangan bencana secara terencana dan menyeluruh. Perubahan formasi Badan Nasional Penanggulangan Bencana dari tahun 1945 sampai sekarang (BPBD, 2019).

2.2. Visi dan Misi

2.2.1. Visi

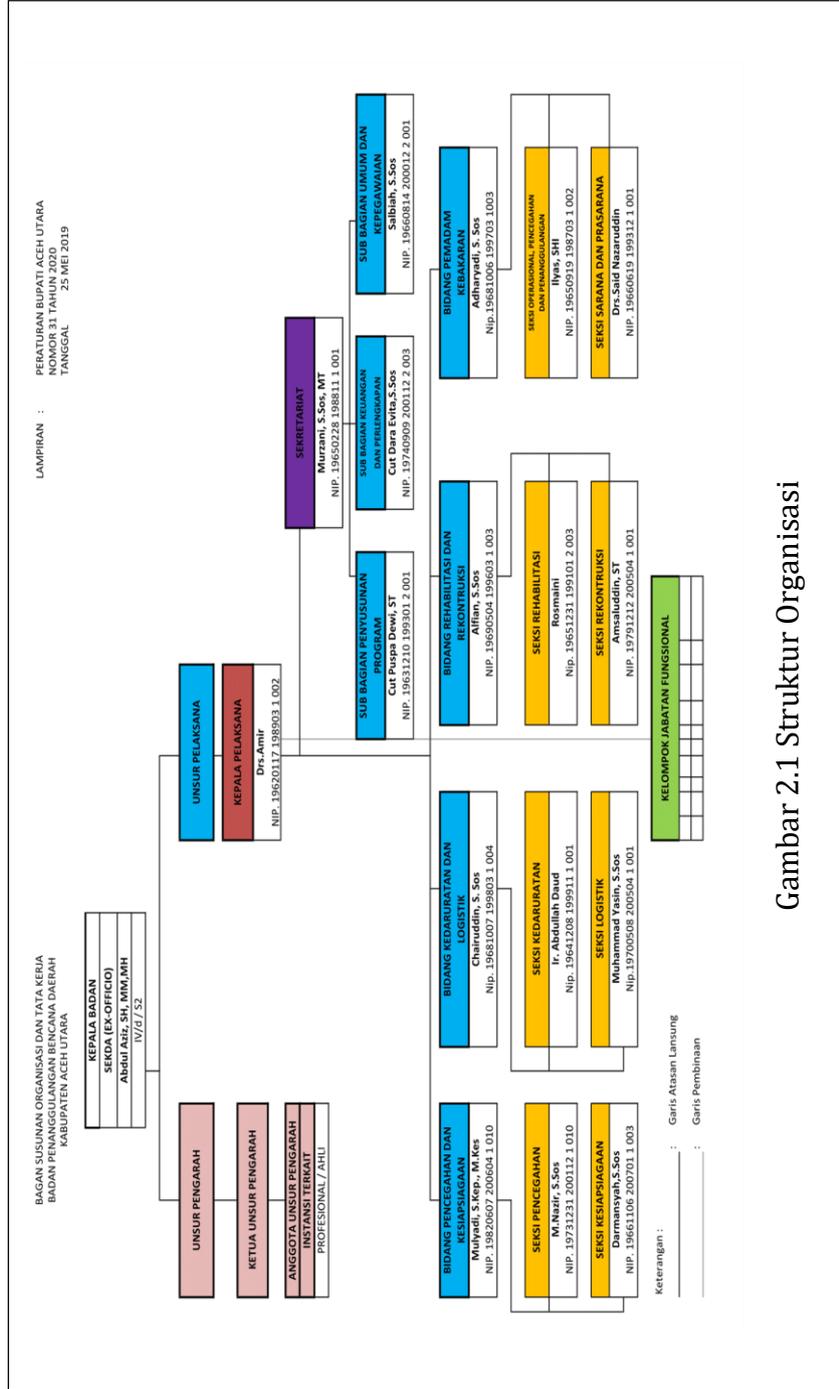
“Badan Penanggulangan Bencana Daerah yang siaga, tanggap dan tangkas dalam penanggulangan bencana”.

2.2.2. Misi

Misi dari Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara, yaitu:

1. Menjaga dan melindungi masyarakat dari ancaman bencana.
2. Melaksanakan penanggulangan bencana melalui pengurangan risiko.
3. Membangun sistem penanggulangan bencana yang handal.
4. Menyelenggarakan penanggulangan bencana secara terancang, terpadu, terkoordinasi dan menyeluruh.

2.3. Struktur Organisasi



Gambar 2.1 Struktur Organisasi

2.4. Pembagian Kerja

Pembagian kerja pada Badan Penanggulangan Bencana Daerah, yaitu:

2.4.1. Kepala Pelaksana

Kepala Pelaksana membantu kepala badan dalam penyelenggaraan tugas dan fungsi unsur pelaksana badan sehari-hari. Kepala Pelaksana mempunyai beberapa tugas pokok yaitu:

1. Memimpin dan membina badan dalam pelaksanaan tugas secara operasional yang ditetapkan berdasarkan ketentuan peraturan perundang-undangan dan kebijakan pemerintah kota.
2. Menyiapkan perumusan kebijakan umum pemerintah kota di bidang penanggulangan bencana.
3. Menetapkan kebijakan teknis di bidang penanggulangan bencana sesuai dengan kebijakan umum pemerintah kota.
4. Melakukan koordinasi dan kerjasama dengan instansi atau lembaga lainnya di bidang penanggulangan bencana.

Untuk menyelenggarakan tugasnya kepala pelaksana mempunyai fungsi yaitu:

1. Pelaksanaan pengendalian urusan ketatausahaan badan.
2. Pelaksanaan pengendalian penyusunan program kerja jangka pendek, jangka menengah, dan jangka panjang.
3. Pelaksanaan dan pengendalian kebijakan teknis di bidang pencegahan, kesiapsiagaan, kedaruratan, logistik, rehabilitasi, dan rekonstruksi bencana di kota.
4. Pelaksanaan pengendalian pengelolaan data dan informasi di bidang pencegahan, kesiapsiagaan, kedaruratan, logistik, rehabilitasi dan rekonstruksi bencana di kota.
5. Pelaksanaan pengendalian dan pengawasan terhadap kegiatan program kerja di bidang pencegahan, kesiapsiagaan, kedaruratan, logistik, rehabilitasi dan rekonstruksi bencana di kota.

6. Pelaksanaan pengendalian, pemantauan, monitoring, dan evaluasi pelaksanaan kegiatan di bidang pencegahan, kesiapsiagaan, kedaruratan, logistik, rehabilitasi dan rekonstruksi bencana di kota.
7. Pelaksanaan koordinasi dan kerjasama dengan instansi atau lembaga terkait lainnya di bidang pencegahan, kesiapsiagaan, kedaruratan, logistik, rehabilitasi dan rekonstruksi bencana di kota.
8. Pelaksanaan tugas kedinasan lainnya yang diberikan oleh walikota dan kepala Badan Penanggulangan Bencana Daerah sesuai dengan bidang tugasnya.

2.4.2. Kepala Badan

Membantu bupati dalam memimpin, mengkoordinasikan dan mengendalikan kebijakan Badan Penanggulangan Bencana Daerah.

2.4.3. Sekretariat

Sekretaris membantu kepala pelaksana dalam mengkoordinasikan penyusunan perencanaan strategis, program kerja tahunan, jangka menengah dan dan jangka panjang, pembinaan dan pelayanan administrasi, kepegawaian, penataan arsip, dokumentasi, ketatalaksanaan dan hubungan masyarakat, pengelolaan keuangan, aset, perlengkapan rumah tangga, pengendalian serta evaluasi dan pelaporan. Untuk menyelenggarakan tugasnya sekretaris Badan Penanggulangan Bencana Daerah mempunyai fungsi:

1. Pengkoordinasi, sinkronisasi dan integrasi program perencanaan dan perumusan kebijakan di lingkungan Badan Penanggulangan Bencana Daerah kota.
2. Pengkoordinasi penyusunan anggaran yang bersumber dari APBK, APBA, APBN, bantuan, pinjaman atau hibah luar negeri.

3. Pembinaan dan pelayanan administrasi ketatausahaan, hukum dan peraturan perundang-undangan, organisasi, tatalaksana, peningkatan kapasitas sumber daya manusia, keuangan, aset, perlengkapan, dan rumah tangga.
4. Pembinaan dan pelaksanaan hubungan masyarakat dan protokol.
5. Fasilitas pelaksanaan tugas dan fungsi unsur pengarah penanggulangan bencana.
6. Pengkoordinasian penyajian data dan informasi kebencanaan di wilayah kota.
7. Pengkoordinasian dalam penyusunan laporan penanggulangan bencana, laporan akuntabilitas kinerja, laporan keterangan pertanggungjawaban walikota dan laporan penyelenggaraan pemerintah kota.
8. Pengkoordinasian pelaksanaan penyusunan program kerja tahunan, jangka menengah dan jangka panjang.
9. Pelaksanaan koordinasi dan kerjasama dengan instansi atau lembaga terkait lainnya sesuai bidang tugas dan fungsinya.
10. Pelaksanaan tugas kedinasan lainnya yang diberikan oleh kepala pelaksana.

2.4.4. Sub Bagian Umum dan Kepegawaian

Sub Bagian Umum dan Kepegawaian membantu sekretaris dalam melaksanakan pengelolaan administrasi yang mempunyai tugas melaksanakan pengelolaan administrasi umum, kepegawaian, penataan arsip, dokumentasi, ketatalaksana, hubungan masyarakat, aset, perlengkapan dan rumah tangga.

2.4.5. Sub Bagian Keuangan

Sub Bagian Keuangan membantu sekretaris melaksanakan penyusunan pengelolaan administrasi keuangan badan dan pertanggungjawaban keuangan. Untuk

menyelenggarakan tugasnya sub bagian keuangan Badan Penanggulangan Bencana Daerah mempunyai fungsi yaitu:

1. Pengelolaan administrasi keuangan badan.
2. Pengelolaan administrasi penyusunan anggaran badan.
3. Pengelolaan pengendalian dan pertanggungjawaban administrasi keuangan badan.

2.4.6. Sub Bagian Program dan Pelaporan

Sub Bagian Program dan Pelaporan membantu sekretaris dalam melaksanakan pengelolaan penyusunan program dan pelaporan badan. Sub bagian program dan pelaporan mempunyai beberapa tugas pokok yaitu:

1. Melaksanakan penyusunan program kerja tahunan, jangka menengah dan jangka panjang.
2. Rencana strategis, penganggaran, melakukan monitoring, evaluasi dan analisis data dalam rangka pelaporan pelaksanaan kebijakan penanganan dan pengurangan risiko bencana di lingkungan Badan Penanggulangan Bencana Daerah.
3. Menyusun laporan akuntabilitas kinerja.
4. Laporan keterangan pertanggungjawaban walikota dan laporan penyelenggaraan pemerintah kota serta penyajian data dan informasi.

Untuk menyelenggarakan tugasnya sub bagian program dan pelaporan mempunyai fungsi yaitu:

1. Penyiapan bahan pengkoordinasian penyusunan program badan.
2. Pengumpulan, pengolahan, dan analisis data badan.
3. Pelaksanaan pembinaan hubungan masyarakat.
4. Pelaksanaan pengelolaan situs *web* badan.
5. Monitoring, evaluasi, dan penyusunan laporan kinerja badan.

2.4.7. Seksi Bidang Pencegahan dan Kesiapsiagaan

Bidang Pencegahan dan Kesiapsiagaan membantu kepala pelaksana dalam merumuskan, mengkoordinasikan dan melaksanakan kebijakan di bidang pencegahan, mitigasi, kesiapsiagaan pada prabencana dan pemberdayaan masyarakat. Untuk menyelenggarakan tugas tersebut, bidang pencegahan dan kesiapsiagaan mempunyai fungsi yaitu:

1. Perumusan kebijakan bidang pencegahan, mitigasi dan kesiapsiagaan pada prabencana serta pemberdayaan masyarakat.
2. Pengkoordinasian dan pelaksanaan kebijakan di bidang pencegahan, mitigasi dan kesiapsiagaan pada prabencana serta pemberdayaan masyarakat.

Dalam melaksanakan tugas dan fungsinya, kepala bidang pencegahan dan kesiapsiagaan dibantu oleh:

a. Kepala seksi pencegahan

Kepala seksi pencegahan membantu kepala bidang pencegahan dan kesiapsiagaan dalam melaksanakan pengelolaan penyelenggaraan pencegahan bencana. Untuk menyelenggarakan tugas tersebut, kepala seksi pencegahan mempunyai fungsi yaitu:

1. Pengumpulan dan pengolahan serta analisis data sumber bahaya atau ancaman bencana.
2. Penyusunan petunjuk teknis mitigasi bencana.
3. Pelaksanaan pemberdayaan masyarakat dalam pencegahan risiko bencana.

b. Kepala Seksi Kesiapsiagaan

Kepala seksi kesiapsiagaan membantu kepala bidang pencegahan dan kesiapsiagaan dalam melaksanakan pengelolaan kesiapsiagaan masyarakat menghadapi bencana. Untuk menyelenggarakan tugas tersebut, kepala seksi kesiapsiagaan mempunyai fungsi yaitu:

1. Pengumpulan dan pengolahan serta analisis data kesiapsiagaan bencana.
2. Penyusunan petunjuk teknis kesiapsiagaan bencana.
3. Pelaksanaan pelatihan kesiapsiagaan penanggulangan bencana.

2.4.8. Seksi Bidang Kedaruratan dan Logistik

Bidang Kedaruratan dan Logistik membantu kepala pelaksana dalam mengkoordinasikan dan melaksanakan kebijakan penanggulangan bencana saat tanggap darurat dan dukungan logistik. Untuk menyelenggarakan tugas tersebut bidang kedaruratan dan logistik mempunyai fungsi yaitu:

1. Perumusan kebijakan di bidang penanggulangan bencana pada saat tanggap darurat, penanganan pengungsi dan dukungan logistik.
2. Pengkoordinasian dan pelaksanaan kebijakan di bidang penanggulangan bencana pada saat tanggap darurat, penanganan pengungsi dan logistik.
3. Komando pelaksanaan penanggulangan bencana pada saat tanggap darurat.

Dalam melaksanakan tugas dan fungsinya, kepala bidang kedaruratan dan logistik dibantu oleh:

a. Kepala Seksi Kedaruratan

Kepala seksi kedaruratan membantu kepala bidang kedaruratan dan logistik dalam melaksanakan pengelolaan penanggulangan bencana pada saat tanggap darurat. Untuk menyelenggarakan tugas tersebut, Kepala seksi kedaruratan mempunyai fungsi yaitu:

1. Pengumpulan dan pengolahan serta analisis data tanggap darurat bencana.
2. Penyusunan petunjuk teknis tanggap darurat bencana.
3. Pelaksanaan penanganan korban bencana.
4. Pelaksanaan penanganan pengungsi.

b. Kepala Seksi Logistik

Kepala Seksi Logistik membantu kepala bidang kedaruratan dan logistik dalam melaksanakan pengelolaan distribusi bantuan bencana. Untuk menyelenggarakan tugas tersebut, Kepala seksi logistik mempunyai fungsi yaitu:

1. Pengumpulan dan pengolahan serta analisis data logistik bencana.
2. Penyusunan petunjuk teknis pengelolaan logistik bencana.
3. Pelaksanaan pengelolaan bantuan sosial korban bencana.

2.4.9. Seksi Rehabilitasi dan Rekonstruksi

Bidang Rehabilitasi dan Rekonstruksi membantu kepala pelaksana dalam mengkoordinasikan dan melaksanakan kebijakan penanggulangan bencana pada saat bencana. Untuk menyelenggarakan tugas tersebut kepala bidang rehabilitasi dan rekonstruksi mempunyai fungsi yaitu:

1. Perumusan kebijakan di bidang penanggulangan bencana pada pascabencana.
2. Pengkoordinasian dan pelaksanaan kebijakan di bidang penanggulangan bencana pada pascabencana.
3. Pelaksanaan penyusunan program/perencanaan teknis, pengelolaan data dan pengawasan di bidang rehabilitasi dan rekonstruksi pascabencana.
4. Pelaksanaan koordinasi dan kerjasama teknis dengan instansi atau lembaga terkait lainnya di bidang rehabilitasi dan rekonstruksi pascabencana.
5. Pengkoordinasian dan pelaksanaan rehabilitasi pemberdaya ekonomi dan peningkatan kondisi sosial, budaya, pelayanan publik, pelayanan utama dalam masyarakat pada pascabencana.
6. Pengkoordinasian kegiatan rehabilitasi pemberdayaan ekonomi dan peningkatan kondisi sosial, budaya, pelayanan publik, pelayanan utama dalam masyarakat pascabencana.

7. Pengkoordinasian dan pelaksanaan rehabilitasi dan rekonstruksi sarana dan prasarana umum pascabencana.
8. Pengkoordinasian dan pelaksanaan perencanaan teknis kegiatan relokasi.
9. Pelaksanaan tugas kedinasan lainnya yang diberikan oleh kepala pelaksanaan sesuai dengan bidang tugas dan fungsinya.

Dalam melaksanakan tugas dan fungsinya, kepala bidang rehabilitasi dan rekonstruksi dibantu oleh:

a. Kepala Seksi Rehabilitasi

Kepala Seksi rehabilitasi membantu kepala bidang rehabilitasi dan rekonstruksi dalam melaksanakan pengelolaan rehabilitasi pascabencana. Untuk menyelenggarakan tugas tersebut kepala seksi rehabilitasi mempunyai fungsi yaitu:

1. Pengumpulan dan pengolahan serta analisis data rehabilitasi pascabencana.
2. Penyusunan petunjuk teknis rehabilitasi pascabencana.
3. Pengkoordinasian dan pelaksanaan rehabilitasi lingkungan, prasarana dan sarana umum yang terkena bencana.
4. Pengkoordinasian dan pelaksanaan pemulihan kondisi sosial, budaya, ekonomi, keamanan dan ketertiban serta pemulihan fungsi pemerintahan dan pelayanan publik.

b. Kepala Seksi Rekonstruksi

Kepala Seksi Rekonstruksi membantu kepala bidang rehabilitasi dan rekonstruksi dalam melaksanakan pengelolaan rekonstruksi pascabencana. Untuk menyelenggarakan tugas tersebut Kepala Seksi Rekonstruksi mempunyai fungsi yaitu:

1. Pengumpulan dan pengolahan serta analisis data rekonstruksi pascabencana.
2. Penyusunan petunjuk teknis rekonstruksi pascabencana.
3. Pengkoordinasian pembangunan kembali prasarana dan sarana umum yang terkena bencana.

4. Pengkoordinasian pembangunan kembali rumah masyarakat yang terkena bencana.

2.4.10. Seksi Bidang Pemadam Kebakaran

Bidang Pemadam Kebakaran dalam menyelenggarakan tugas tersebut mempunyai fungsi yaitu:

1. Perumusan kebijakan, pelaksanaan kebijakan, pelaksanaan evaluasi dan pelaporan, serta pelaksanaan administrasi pencegahan dan kesiapsiagaan.
2. Perumusan kebijakan, pelaksanaan kebijakan, pelaksanaan evaluasi dan pelaporan, serta pelaksanaan administrasi sarana dan prasarana.
3. Perumusan kebijakan, pelaksanaan kebijakan, pelaksanaan evaluasi dan pelaporan, serta pelaksanaan administrasi pengendalian dan operasional pemadam kebakaran.
4. Pelaksanaan fungsi lain yang diberikan oleh kepala satuan sesuai dengan tugas dan fungsinya.

Dalam melaksanakan tugas dan fungsinya, kepala bidang Pemadam Kebakaran dibantu oleh:

1. Kepala Seksi Operasional, Pencegahan, dan Penanggulangan.
2. Kepala Seksi Sarana dan Prasarana.

BAB III

LANDASAN TEORI

3.1. Rehabilitasi

Menurut peraturan Kepala Badan Nasional Penanggulangan Bencana No.11 Tahun 2008 tentang pedoman rehabilitasi dan rekonstruksi pascabencana rehabilitasi adalah pemulihan dan perbaikan dari semua bidang pada pelayanan publik atau masyarakat di daerah pascabencana dengan usaha semaksimal mungkin dengan tujuan menormalisasikan semua aspek pemerintahan dan membenahi kehidupan masyarakat pada wilayah pascabencana, seperti perbaikan lingkungan daerah bencana, pelayanan kesehatan, pemulihan psikologis, keamanan dan ketertiban serta pemulihan layanan publik.

Kegiatan yang dilakukan pada bidang rehabilitasi, antara lain:

1. Perbaikan zona bencana merupakan aktivitas yang dilakukan pada daerah bencana dengan persyaratan teknis, sosial, ekonomi, dan budaya dan juga ekosistem sosial suatu area yang meliputi daerah pemukiman, daerah industri, daerah bangunan gedung dan area usaha.
2. Perbaikan prasarana dan sarana umum merupakan aktivitas perbaikan yang dilaksanakan untuk mencukupi kebutuhan transportasi, kemajuan di bidang ekonomi dan lingkungan sosial budaya masyarakat. Perbaikan tersebut meliputi perbaikan infrastruktur, dan fasilitas umum.
3. Pemberian bantuan renovasi tempat tinggal masyarakat merupakan kegiatan bantuan dari pemerintah sebagai bentuk bantuan kepada masyarakat yang rumahnya mengalami kerusakan pascabencana agar tempat tinggal tersebut dapat dihuni secara layak. Bantuan ini berupa bahan material dan biaya yang diperlukan berdasarkan kerusakan yang terjadi.

4. Pemulihan sosial psikologis merupakan aktivitas yang dilakukan untuk menopang pemulihan kondisi mental atau psikologis terhadap korban bencana. Upaya yang dilakukan berupa bantuan konseling dan konsultasi, pendampingan, pelatihan, dan aktivitas psikososial.
5. Pelayanan kesehatan merupakan kegiatan yang dilakukan untuk membantu pemulihan kondisi kesehatan bagi para masyarakat yang menjadi korban bencana. Pelayanan kesehatan masyarakat berupa membantu korban bencana yang sakit dan mengalami luka, menyediakan peralatan kesehatan, menyediakan obat-obatan, dan menyiapkan tenaga medis.
6. Rekonsiliasi merupakan kegiatan yang dilakukan untuk membantu masyarakat yang mengalami konflik sosial yang bertujuan untuk menurunkan ketegangan dan memulihkan kondisi sosial bermasyarakat. Upaya yang dilakukan yaitu dengan cara melibatkan tokoh-tokoh masyarakat dengan melihat kondisi, situasi, dan karakter masyarakat serta budaya masyarakat tersebut agar konflik yang terjadi dapat diselesaikan dengan jalan yang adil.
7. Pemulihan sosial ekonomi budaya merupakan kegiatan yang dilakukan untuk membantu masyarakat yang terkena dampak bencana yang bertujuan untuk memulihkan kondisi kehidupan sosial, budaya dan ekonomi seperti pada kondisi sebelum terjadinya bencana. Kegiatan yang dilakukan berupa menghidupkan kembali kegiatan sosial, budaya dan ekonomi, layanan kegiatan konseling, dan pelatihan.
8. Pemulihan keamanan dan ketertiban merupakan kegiatan yang dilakukan untuk memulihkan kondisi keamanan dan ketertiban yang terjadi pada daerah bencana agar kembali kepada kondisi sebelum terjadinya bencana. Kegiatan yang dilakukan berupa mengaktifkan kembali lembaga keamanan dan ketertiban di daerah bencana, meningkatkan peran masyarakat dalam kegiatan keamanan dan ketertiban, dan

mengkoordinasi lembaga yang berwenang di bidang keamanan dan ketertiban.

9. Pemulihan fungsi pemerintahan merupakan kegiatan yang ditujukan untuk memulihkan fungsi pemerintahan kembali seperti kondisi sebelum terjadi bencana. Upaya yang dilakukan yaitu mengaktifkan kembali pelaksanaan tugas-tugas pemerintahan secepatnya, penyelamat dan pengamanan dokumen-dokumen negara dan pemerintahan, pemulihan peralatan pendukung tugas-tugas pemerintahan, dan pengaturan kembali tugas-tugas lembaga pemerintahan yang terkait.
10. Pemulihan fungsi pelayanan publik merupakan kegiatan yang ditujukan untuk memulihkan kembali fungsi-fungsi pelayanan publik kepada masyarakat. Upaya yang dilakukan yaitu rehabilitasi dan pemulihan fungsi prasarana dan sarana pelayanan publik, menghidupkan kembali fungsi pelayanan publik pada lembaga yang terkait, dan pengaturan kembali fungsi pelayanan publik.

3.2. Rekonstruksi

Menurut Undang-Undang No. 24 Tahun 2007 pasal 1 ayat 12 tentang Penanggulangan Bencana Rekonstruksi adalah pembangunan kembali semua prasarana dan sarana, kelembagaan pada wilayah pascabencana, baik pada tingkat pemerintahan maupun masyarakat dengan sasaran utama tumbuh dan berkembangnya kegiatan perekonomian, sosial dan budaya, tegaknya hukum dan ketertiban, dan bangkitnya peran serta masyarakat dalam segala aspek kehidupan bermasyarakat pada wilayah pascabencana.

Kebijakan yang mendasari penyelenggaraan rekonstruksi ini adalah sebagai berikut:

1. Penanggulangan bencana merupakan tanggungjawab bersama antara Pemerintah, Pemerintah Daerah dan Masyarakat.

2. Pemerintah berkewajiban untuk menyiapkan program dan alokasi anggaran untuk rekonstruksi pascabencana.
3. Pemerintah memberikan fasilitasi dan pendampingan bantuan dana yang dimanfaatkan berdasarkan kearifan lokal.
4. Bantuan luar negeri, baik yang berasal dari pemerintah (bilateral-multilateral) maupun non-pemerintah diperkenankan, sepanjang bantuan tersebut tidak mengikat dan tidak bertentangan dengan peraturan perundang-undangan yang berlaku.
5. Membangun kembali dengan lebih baik dari sebelum kejadian bencana, dengan memahami bahwa suatu peristiwa bencana membawa hikmah untuk memberikan kesempatan dalam rangka meningkatkan kehidupan masyarakat melalui penataan prasarana, sarana dan sistem pelayanan masyarakat yang lebih baik dan lebih aman dari sebelum terjadinya bencana.
6. Upaya-upaya pengurangan risiko bencana, meliputi usaha pencegahan, mitigasi dan peningkatan kesiapsiagaan menghadapi keadaan darurat bencana harus diintegrasikan ke dalam keseluruhan proses rekonstruksi agar risiko bencana di masa yang akan datang dapat dikurangi semaksimal mungkin.
7. Pelaksanaan rekonstruksi harus dapat mendorong dikembangkannya atau direvisinya peraturan-perundangan dan standar-standar keselamatan yang lebih baik dalam berbagai aspek kehidupan, baik pada tingkat nasional maupun lokal, dan mengadaptasi pengetahuan terbaru mengenai bahaya dan kerentanan setelah kejadian bencana.
8. Menempatkan isu-isu ekosistem/lingkungan hidup dan sosial budaya secara proporsional dalam perencanaan rekonstruksi.
9. Melaksanakan rekonstruksi dengan proses yang akuntabel dan *auditable* serta memenuhi asas transparansi publik.

10. Penyelenggaraan rekonstruksi dilakukan di bawah koordinasi BNPB dan/atau Badan Penanggulangan Bencana Daerah (untuk tingkat daerah).

Strategi dalam penyelenggaraan rekonstruksi ini adalah:

1. Melibatkan partisipasi masyarakat sebesar mungkin, baik masyarakat yang terkena bencana maupun masyarakat secara umum, melalui proses memberdayakan masyarakat dalam berbagai kegiatan penyelenggaraan rekonstruksi dan dengan menciptakan situasi kondusif bagi peran serta masyarakat yang sebesar-besarnya dalam kegiatan rekonstruksi, melalui mekanisme pelibatan yang sederhana.
2. Memanfaatkan kearifan lokal berdasarkan pada kondisi aktual di lapangan, melalui program yang mengacu kepada kebijakan pemerintah dengan memperhatikan kondisi sosial dan budaya masyarakat.
3. Mendorong pengembangan kapasitas dalam pelaksanaan rekonstruksi, baik ketika perencanaan, pelaksanaan, monitoring maupun penegakkan aturan-aturan yang ada, untuk menjamin hasil rekonstruksi yang memiliki ketahanan yang lebih baik terhadap bencana di masa yang akan datang, baik di tingkatan pemerintahan, masyarakat, komunitas lokal maupun individu.
4. Mengutamakan solusi jangka panjang daripada penyelesaian masalah-masalah yang bersifat sementara.
5. Memberikan perhatian khusus kepada usaha-usaha berkelanjutan yang bersifat lokal.
6. Menggunakan proses perencanaan yang terintegrasi, dengan penetapan prioritas jangka pendek, menengah dan panjang.
7. Mengutamakan usaha-usaha untuk memulihkan kondisi ekonomi lokal dengan cepat sebagai bagian dari kegiatan prioritas jangka pendek, melalui pelibatan sebanyak-

banyaknya berbagai pelaku ekonomi lokal dalam proses rekonstruksi.

8. Mengintegrasikan teknologi maju dengan sumber daya lokal yang sesuai.
9. Menggunakan rencana implementasi yang sederhana.
10. Memastikan tersedianya akses informasi mengenai semua kegiatan rekonstruksi bagi semua pemangku kepentingan dalam rangka membangun komunikasi untuk menjamin akuntabilitas dan transparansi proses rekonstruksi.

3.3. Bencana

Bencana menurut Undang-undang Nomor 24 tahun 2007 pasal 1 ayat 1 tentang penanggulangan bencana yaitu peristiwa atau rangkaian peristiwa yang mengancam dan mengganggu kehidupan dan penghidupan masyarakat yang disebabkan, baik oleh faktor alam dan/atau faktor non-alam maupun faktor manusia sehingga mengakibatkan timbulnya korban jiwa manusia, kerusakan lingkungan, kerugian harta benda, dan dampak psikologis.

Untuk dapat dikatakan kejadian itu adalah bencana, maka bencana harus memenuhi kriteria berikut ini:

1. Ada peristiwa.
2. Terjadi karena faktor alam atau ulah manusia.
3. Terjadi secara tiba-tiba akan tetapi dapat juga terjadi perlahan-lahan/bertahap.
4. Menimbulkan hilangnya jiwa manusia, harta benda, kerugian sosial ekonomi, kerusakan lingkungan, dan lain-lain.
5. Berada di luar kemampuan masyarakat untuk menanggulangnya (Nurjanah dkk 2012, 11).

3.3.1. Jenis-Jenis Bencana

Sesuai dengan Undang-undang No 24 tahun 2007 tentang penanggulangan bencana, bencana dikelompokkan menjadi 3 jenis, yaitu sebagai berikut:

1. Bencana alam, yaitu bencana yang berasal dari fenomena alam seperti gempa bumi, letusan gunung berapi, banjir, topan, tsunami, dan lain-lain.
2. Bencana non alam, yaitu bencana yang diakibatkan oleh peristiwa non alam berupa gagal teknologi, gagal modernisasi, epidemik, dan wabah penyakit.
3. Bencana sosial, yaitu bencana yang diakibatkan oleh peristiwa atau rangkaian peristiwa yang diakibatkan oleh manusia yang meliputi konflik sosial antar kelompok, antar komunitas masyarakat dan teror.

Kemudian secara umum ada 6 jenis bencana, yaitu:

1. Bencana geologi, seperti longsor/gerakan tanah, letusan gunung api, dan gempa bumi/tsunami.
2. Bencana hydro-meteorologi, seperti kebakaran hutan, banjir, kekeringan, rob/air laut pasang dan badai/angin topan.
3. Bencana biologi, seperti penyakit tanaman/hewan dan epidemik.
4. Bencana kegagalan teknologi, seperti kelalaian manusia dalam mengoperasikan produk teknologi, kecelakaan transportasi, kesalahan desain teknologi, dan kecelakaan/kegagalan industri.
5. Bencana lingkungan, seperti kebakaran (*urban fire*), kebakaran hutan, abrasi pantai, dan pencemaran.
6. Bencana sosial, seperti konflik sosial, eksodus (pengungsian secara besar-besaran), dan terorisme/ledakan bom (Nurjanah dkk, 20).

3.3.2. Dampak Bencana

Dampak bencana adalah segala sesuatu yang merubah dan merugikan akibat kejadian dari bencana. Dampak yang dirasakan seperti kerusakan atau kehilangan harta benda, luka-luka, kematian, gangguan gaya hidup, kerusakan infrastruktur, kerugian ekonomi, kehilangan mata pencaharian, psikologi terganggu, kehilangan rumah, dan gangguan sistem pemerintahan (Kodoatie Sjarief 2006, 146).

Dampak bencana bervariasi tergantung pada kondisi, kerentanan lingkungan dan masyarakat (Hidayati 2005, 46). Karena berbagai organisasi baik pemerintahan maupun LSM yang telah memberikan bantuan seiring berjalannya waktu, dampak bencana secara fisik perlahan dapat teratasi (Nurrachman Nani 2007, 11).

Banyak persoalan yang dihadapi oleh para korban yang selamat saat terjadinya bencana seperti psikologis yang terganggu, penyesuaian diri terhadap konflik fisik, dan kehidupan sosial yang ada setelah terjadinya bencana. Tidak jarang dengan kondisi tersebut dapat menimbulkan konflik batin bagi para korban untuk bisa menerima kenyataan bahwa kondisi sekarang sudah tidak seperti dulu lagi (Arifin Saru 2008, 5). Bencana menjadi pengalaman traumatik karena bencana dapat merubah diri sendiri dan lingkungan dalam waktu sekejap (Nurrachman Nani 2007, 4).

3.4. Pengertian Sistem

Menurut Jogianto (2008) sistem merupakan kumpulan dari langkah-langkah yang saling berhubungan untuk mencapai suatu tujuan.

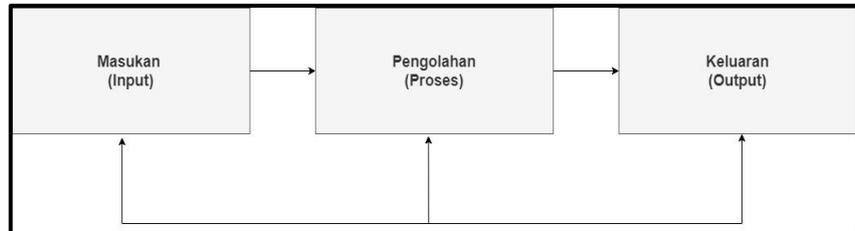
Menurut Abdul Kadir (2005) sistem adalah sekelompok elemen yang saling berkaitan untuk mencapai suatu tujuan.

Menurut Bodnar dan Hoowood (2004) sistem merupakan suatu hubungan kerja dari langkah-langkah yang

saling berhubungan satu sama lain lalu disatukan secara untuk mencapai tujuan tertentu.

Dari beberapa pengertian di atas, maka dapat disimpulkan bahwa sistem adalah sekumpulan komponen yang saling berkaitan untuk mencapai tujuan tertentu.

Secara umum unsur-unsur dari sebuah sistem, yaitu: masukan (*input*), pengolahan (*processing*) dan juga keluaran (*output*). Dapat digambarkan sebagai berikut:



Gambar 3.1 Unsur-unsur Sistem

3.4.1. Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat yang menandakan bahwa itu dikatakan sistem. Adapun karakteristiknya, yaitu (Jogianto, 2005).

1. Komponen Sistem (*Components*)

Yaitu suatu sistem yang dalam mencapai sasarnya melalui unsur-unsur yang saling berhubungan dan saling melengkapi serta bervariasi. Unsur-unsur sistem atau komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem memiliki karakter-karakter dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batasan sistem (*Boundary*)

Yaitu wilayah yang membatasi antara sistem yang satu dengan sistem lainnya terhadap lingkungan luarnya. Batas suatu sistem memperlihatkan ruang lingkup (*scope*) dari sistem tersebut.

3. Lingkungan luar sistem (*Environment*)

Yaitu segala sesuatu yang di luar ruang lingkup atau batasan sistem yang dapat mempengaruhi operasi sistem.

4. Sistem Penghubung (*Interface*)

Yaitu media penghubung subsistem yang satu dengan subsistem yang lainnya sehingga membentuk satu kesatuan. *Output* dari suatu subsistem akan menjadi *input* untuk subsistem yang lainnya dengan melalui penghubung agar subsistem tersebut dapat berintegrasi dengan subsistem yang lainnya sehingga membentuk satu kesatuan.

5. Sistem Masukan (*Input*)

Yaitu energi yang dimasukkan ke dalam sistem berupa masukan pemeliharaan dan masukan sinyal. Masukan pemeliharaan yaitu energi yang dimasukkan sehingga sistem dapat beroperasi. Sedangkan sistem signal yaitu energi yang diolah untuk mendapatkan keluaran.

6. Sistem Keluaran (*Output*)

Merupakan energi yang diolah dan dikelompokkan menjadi hasil yang bermanfaat untuk subsistem lain.

7. Sistem Sasaran

Yaitu suatu sistem dikatakan berhasil jika sudah tercapai tujuan dan sasaran yang telah direncanakan.

3.4.2. Klasifikasi Sistem

Klasifikasi sistem yaitu sebagai berikut:

1. Sistem abstrak (*abstract system*)

Sistem abstrak adalah sistem yang datangnya dari pemikiran atau ide yang secara fisik tidak kelihatan. Contohnya sistem yang berupa pemikiran-pemikiran hubungan antara manusia dan tuhan.

2. Sistem fisik (*physical system*)

Sistem fisik merupakan sistem yang dapat dilihat oleh mata dan memiliki bentuk fisik sesuai dengan kebutuhan.

Contohnya sistem komputer, sistem mesin dan sistem perangkat lunak.

3. Sistem tertentu (*deterministic system*)

Sistem tertentu merupakan sistem yang beroperasi secara otomatis yang dapat diprediksi dengan pasti sehingga menghasilkan *output* yang pasti. Contoh alarm, dan sistem komputer yang sudah dijadwal untuk *maintenance*.

3.4.3. Sistem yang Buruk

Ciri-ciri sistem yang buruk, antara lain:

1. Tidak memenuhi pengguna
2. Performance buruk
3. Rehabilitas rendah
4. Kegunaan rendah
5. Contoh-contoh kesulitan: Tidak terjadwal, tidak ada rencana anggaran, 100% over budget atau jadwal.

3.5. Pengertian Informasi

Menurut Krismiaji dan Zabaer (2005) informasi adalah suatu data yang telah diolah dan mempunyai fungsi dan juga manfaat.

Menurut Jogiyanto (2008) informasi merupakan data yang diproses menjadi sesuatu hal yang berguna bagi siapapun yang menggunakannya.

Menurut Edhy Sutanta (2011) informasi adalah hasil dari suatu pengolahan data yang berguna bagi user baik di masa sekarang dan juga di masa seterusnya.

Dari pengertian di atas dapat disimpulkan bahwa informasi adalah sekumpulan data yang diproses menjadi sesuatu hal yang memiliki manfaat untuk membuat suatu keputusan. Tanpa suatu informasi, suatu sistem tidak akan berjalan dengan lancar dan tidak akan dapat beroperasi.

Dengan kata lain sumber informasi adalah data. Data memperlihatkan suatu kejadian yang sedang terjadi, dimana

data tersebut akan diolah dan diterapkan dalam sistem menjadi *input* yang berguna dalam suatu sistem. Secara rinci, pengertian data sebagai berikut:

Data adalah gambaran suatu kejadian yang sedang dihadapi. Semakin banyak data dan kompleksnya aktivitas pengolahan data dalam suatu organisasi, baik itu organisasi besar maupun organisasi kecil, maka metode pengolahan data sangat dibutuhkan.

Dalam metode ini diharapkan dapat menyelesaikan kebutuhan pengolahan data yang ada, yaitu sebagai berikut:

1. Ruang penyimpanan data yang efisien.
2. Proses yang cepat untuk mendapatkan informasi yang dibutuhkan oleh unsur-unsur dalam organisasi, baik sebagian maupun keseluruhan.

3.5.1. Jenis-Jenis Informasi

Beberapa jenis-jenis informasi yaitu sebagai berikut:

1. *Absolute Information*

Yaitu induk dari informasi yang disampaikan dengan jaminan dan tidak diperlukan penjelasan selanjutnya.

2. *Substitutional Information*

Yaitu informasi yang mempunyai konsep yang digunakan pada beberapa informasi.

3. *Philosophic information*

Yaitu jenis informasi yang mengaitkan antara pengetahuan dan kebijakan.

4. *Subjective information*

Yaitu jenis informasi yang menghubungkan antara perasaan dan informasi manusia yang bergantung pada orang yang menyampaikan informasi.

5. *Objective information*

Yaitu jenis informasi yang tertuju pada informasi logis.

6. *Cultural information*

Yaitu jenis informasi yang ditekankan pada cultural.

3.5.2. Kualitas Informasi

Sutabri (2005) menjelaskan bahwa informasi dikatakan berkualitas jika memenuhi 3 hal berikut, yaitu:

1. Informasi harus akurat (*accurate*), yaitu informasi yang diberikan harus benar dan tidak menyesatkan. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut.
2. Tepat waktu, yaitu informasi yang disampaikan harus tepat waktu dan bukan informasi yang sudah lama karena sangat mempengaruhi dalam pengambilan keputusan. Di era ini, mahalnya nilai informasi disebabkan oleh cepatnya informasi tersebut didapatkan, sehingga diperlukan teknologi mutakhir untuk mendapatkan, memproses dan mengirimnya.
3. Relevan (*relevance*), yaitu informasi harus bermanfaat bagi penggunaannya.

3.5.3. Siklus Hidup Informasi

Siklus hidup informasi merupakan pengelolaan informasi melalui proses yang digunakan untuk mengetahui kapan harus diformulasikan, dirasionalisasi, dan pada akhirnya informasi tersebut dibuang/tidak digunakan lagi. Siklus hidup informasi meliputi:

1. Bagaimana informasi diciptakan
Systemic (berhubungan dengan suatu sistem),
environmental (berhubungan dengan lingkungan).
2. Bagaimana informasi disampaikan
Link yang dikirim dengan *email* ke *website* tertentu, disampaikan berbasis aplikasi melalui *expert system*, melalui *one to one presentation* seperti dari mulut ke mulut, komunikasi ad-hoc.

3. Bagaimana informasi setelah dikelola

Kompleksitas informasi, kompleksitas proses penciptaan, kompleksitas manajemen sistem, dampak finansial dari penciptaan informasi.

4. Jenis informasi yang diciptakan

Expert system (memindahkan informasi ke dalam struktur formal), *explicit* (diciptakan dan disimpan secara formal seperti *network share*, *network website/intranet*, *knowledge management system informal*, sistem manajemen dokumen, formal *knowledge management system*.

5. Nilai sumber

Usia informasi, kedekatan informasi dengan pelanggan, sumber informasi, dan interaksi sebelumnya dengan sumber tertentu.

3.6. Pengertian Sistem Informasi

Menurut Ida Nuraida (2008) sistem informasi merupakan langkah-langkah terstruktur yang dilakukan untuk menyediakan informasi yang berguna dalam penentuan suatu keputusan.

Menurut Yakub (2012) sistem informasi adalah proses pengelolaan informasi yang terbentuk dari komponen-komponen yang saling berhubungan.

Dari pengertian di atas, dapat disimpulkan bahwa sistem informasi adalah kumpulan dari proses pengolahan data yang menghasilkan informasi untuk menentukan suatu keputusan.

3.6.1. Komponen Sistem Informasi

Stair (1992) menjelaskan bahwa *Computer-Based Information System (CBIS)* dalam suatu organisasi terdiri dari:

1. Perangkat keras, yaitu elemen yang digunakan untuk melengkapi pemasukan data, proses data, dan keluaran data.

2. Perangkat lunak, yaitu perintah yang dialokasikan ke komputer.
3. *Database*, yaitu kumpulan data dan informasi yang dirancang dengan sebaik-baiknya sehingga mudah diakses oleh pengguna sistem informasi.
4. Telekomunikasi, yaitu komunikasi efektif yang dilakukan bersama-sama untuk menghubungkan antara *user* sistem dengan *user* komputer.
5. Manusia, yaitu *user* yang berkontribusi dalam sistem informasi, seperti manajer, programmer dan yang bertanggung jawab atas perawatan sistem.

Sementara Burch dan Grudnistki (1986) berpendapat bahwa sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*). Uraianya sebagai berikut:

1. Blok masukan (*input block*)

Input disini mewakili data yang masuk ke dalam suatu sistem informasi. Metode-metode dan media untuk menangkap data yang akan diinputkan yaitu berupa dokumen-dokumen dasar.

2. Blok model (*model block*)

Blok ini terdiri atas langkah-langkah, logika, dan model matematika yang akan memanipulasi data masukan dan data yang tersimpan di *database* dengan cara tertentu untuk menghasilkan *output* yang dihasilkan.

3. Blok keluaran (*out block*)

Blok ini merupakan produk keluaran yang memiliki informasi yang berkualitas dan dokumentasi yang berfungsi untuk semua tingkat manajemen serta semua pemakai sistem.

4. Blok teknologi (*technology block*)

Blok ini merupakan kotak alat (*tool-box*) dari sistem informasi yang berfungsi untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan

sekaligus mengirimkan *output* dan membantu pengendalian dari sistem secara keseluruhan.

5. Blok basis data (*database block*)

Database adalah sekumpulan data yang saling berkaitan satu sama lain yang tersimpan di perangkat keras komputer dan dimanipulasi pada perangkat lunak.

6. Blok kendali (*control block*)

Blok ini merupakan pengendalian untuk mencegah dan memperbaiki kesalahan-kesalahan yang terjadi pada rancangan sistem yang dapat merusak sistem tersebut.

3.6.2. Teknik Memperoleh Informasi

Pengumpulan data primer (data utama) memerlukan interaksi langsung dengan *user*. Sementara itu untuk data sekunder dapat dikumpulkan dari sumber-sumber yang ada. Pengumpulan data juga dapat ditemukan dari sumber sekunder seperti perpustakaan dan sumber dokumen lain dari lapangan serta laboratorium.

Teknik pengumpulan data sekunder pada umumnya menggunakan metode analisis substansi dokumen, surat keputusan, laporan, data statistik, maupun data non-statistik yang sudah terdokumentasi. Teknik pengumpulan data di lapangan dapat dilakukan dengan metode observasi atau survey. Survey dapat dilakukan dengan mendistribusikan kuesioner/daftar isian melalui pos atau wawancara. Wawancara juga dapat dilakukan langsung, melalui telepon, bahkan surat elektronik (*e-mail*). Teknik pengumpulan data di laboratorium biasanya dilakukan dengan cara uji coba/eksperimen.

Jadi secara umum, teknik pengumpulan data yaitu sebagai berikut:

1. Pengamatan

Teknik ini dilakukan dengan mengamati proses yang ada, yang meliputi pengamatan pada aliran-aliran informasi

yang kemudian dapat digambarkan kedalam bentuk grafik (seperti DAD, HIPO, dan lain-lain). Ada beberapa kelebihan pada teknik pengamatan ini, yaitu diantaranya:

- a. Data yang dikumpulkan mempunyai keandalan yang tinggi.
- b. Analisis sistem melalui observasi dapat melihat langsung proses-proses yang ada pada sistem.
- c. Dapat digambarkan lingkungan fisik dari kegiatan dan sebagainya.

2. Teknik wawancara secara langsung

Teknik wawancara melibatkan dua sisi antara *user* dan pengembang sistem informasi.

Teknik ini memiliki kelebihan, yaitu:

- a. Memberi kesempatan kepada pewawancara untuk memberikan motivasi agar yang diwawancarai bisa menjawab secara bebas dan terbuka.
- b. Memungkinkan pewawancara mengembangkan pertanyaan sesuai dengan situasi yang berkembang dan lain-lain.

Kelemahan dari teknik ini, yaitu:

- a. Prosesnya membutuhkan waktu yang lama.
- b. Keberhasilan suatu wawancara sangat tergantung pada kepandaian pewawancara dalam mengembangkan pertanyaan.
- c. Waktu yang dibutuhkan untuk wawancara seringkali mengganggu pekerjaan orang yang diwawancarai.

3. Teknik kuesioner/daftar pertanyaan

Teknik kuesioner memungkinkan untuk mendapatkan data dari sejumlah besar orang dengan biaya yang wajar. Isi dari kuesioner berisi pertanyaan terstruktur yang dapat dijawab tanpa harus bertemu muka.

Kelebihan dari teknik ini, yaitu:

- a. Responden tidak merasa terganggu.
- b. Karena biasanya mencantumkan identitas responden maka biasanya lebih obyektif.

Kelemahan dari teknik kuesioner, yaitu:

- a. Tidak diketahui 100% kebenaran dari jawaban para responden.
- b. Daftar pertanyaan cenderung tidak fleksibel karena tidak bisa dikembangkan sesuai dengan situasi.

3.7. Perancangan Sistem

Perancangan sistem adalah tahap setelah analisis dari siklus pengembangan sistem pendefinisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi menggambarkan bagaimana suatu sistem dibentuk dengan penggambaran dan perencanaannya.

Setelah tahap analisis sistem selesai dilakukan, maka analisis sistem telah mendapatkan gambaran yang jelas apa yang harus dikerjakan. Tiba waktunya sekarang bagi analisis sistem untuk memikirkan bagaimana membentuk sistem tersebut. Tahap ini disebut perancangan sistem.

Perancangan sistem dapat dibagi menjadi 2 bagian, yaitu:

1. Perancangan sistem secara umum/perancangan konseptual, perancangan logikal/perancangan secara makro.
2. Perancangan sistem terperinci/perancangan sistem secara fisik.

Di bawah ini merupakan tahapan-tahapan yang dilakukan dalam perancangan suatu sistem:

1. Mendefinisikan masalah, yaitu menetapkan hal-hal apa saja yang menjadi permasalahan yang sedang terjadi di sekitar

kita atau di sebuah perusahaan sehingga kita dapat menentukan cara menyelesaikan yang sesuai.

2. Analisis masalah, yaitu mengobservasi masalah yang sudah ditemukan sehingga bisa menemukan cara terbaik untuk memecahkan permasalahan terhadap perancangan sistem yang digambarkan.
3. Perancangan global, yaitu menjalankan perancangan terhadap sistem yang baru dengan menggunakan langkah-langkah perancangan secara garis besar.
4. Perancangan detail, yaitu menjalankan bentuk-bentuk perancangan dengan secara terperinci dan detail.
5. Program, yaitu membangun rancangan sistem yang baru dengan menggunakan bahasa pemrograman yang dibutuhkan pada program yang sedang digarap.
6. Testing, yaitu pengujian terhadap program yang sedang dirancang.
7. Implementasi, yaitu menentukan program yang telah dikerjakan kedalam sistem apakah sistem tersebut sudah sesuai dengan yang diinginkan.
8. Evaluasi, operasi dan perawatan, yaitu melakukan pemeriksaan terhadap program dan melakukan penawaran secara umum untuk memelihara program yang ada.
9. Dokumentasi, yaitu menduplikasi data melalui program yang telah dibuat untuk mengantisipasi hal-hal yang tidak diinginkan.

Sasaran-sasaran sistem yang akan dicapai dalam perancangan suatu sistem, yaitu:

1. Perancangan sistem harus bermanfaat, mudah dipahami, dan mudah digunakan.
2. Perancangan sistem harus dapat mendukung tujuan dari sebuah instansi/perusahaan.

3. Perancangan sistem harus efisien dan efektif untuk dapat mendukung pengolahan data rehabilitasi dan rekonstruksi pascabencana.
4. Perancangan sistem harus dapat mempersiapkan rancangan yang terperinci untuk masing-masing komponen dari sistem informasi.

Alat bantu untuk perancangan sistem, yaitu:

3.7.1. Diagram Konteks

Diagram konteks adalah level tertinggi yang menggambarkan keseluruhan sistem yang mewakili seluruh langkah-langkah yang terdapat pada sistem. Penggambaran diagram konteks terdiri dari:

1. Gambar sistem berupa 1 (satu) lingkaran dan beri nama system.
2. Gambar kotak entitas eksternal dan beri nama entitas.
3. Buatlah arus data dari setiap entitas eksternal.

3.7.2. *Data Flow Diagram (DFD)*

Menurut Andri Kristanto (2003: 55) *data flow diagram* adalah suatu langkah-langkah logika yang memaparkan darimana asal datanya, dimana datanya disimpan, dan tujuan dari data tersebut.

Menurut Tata Sutabri (2003: 163) *data flow diagram* adalah suatu jaringan sistem yang memaparkan sistem secara manual/komputerisasi atau gabungan dari keduanya yang saling berkaitan sesuai dengan aturan.

Data Flow Diagram (DFD) merupakan penggambaran suatu sistem yang telah ada atau sistem baru yang akan dikembangkan tanpa harus melihat keadaan lingkungan fisik dimana data tersebut mengalir.

Keunggulan dari *DFD* adalah:

1. *DFD* mudah dimengerti oleh orang teknik maupun non teknik.

2. Menampilkan gambaran sistem secara menyeluruh.
3. Memberikan tampilan komponen-komponen secara detail.

Simbol-simbol *DFD* antara lain:

a. *External Entity*

Simbol ini digunakan untuk menggambarkan tujuan data. *External entity* dapat berupa orang (*user*) atau sekelompok orang/bagian/pemakai sistem.



Gambar 3.2 *Entity*

b. *Data flow* (arus data)

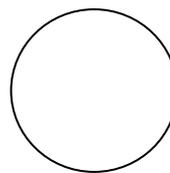
Simbol ini digunakan untuk menggambarkan aliran data yang berjalan. Data yang bergerak dari satu tempat di dalam sistem ke tempat lainnya. *Data flow* diberi simbol suatu panah.



Gambar 3.3 Aliran Data

c. *Process* (proses)

Simbol ini digunakan untuk proses pengolahan atau transformasi data. Kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.



Gambar 3.4 Proses

d. *Data store* (simpanan data)

Simbol ini digunakan untuk menggambarkan media penyimpanan data yang sudah disimpan. Datanya berupa:

1. Suatu *file* atau *database* yang tersimpan dalam komputer.
2. Suatu arsip atau catatan manual.
3. Suatu kontak tempat data di meja seseorang.
4. Suatu label acuan seseorang.
5. Suatu agenda atau buku.

Gambar 3.5 Simpanan Data

Adapun fungsi dari *Data Flow Diagram (DFD)*, yaitu:

1. Sebagai media pembuatan model untuk menggambarkan sistem yang akan dihubungkan satu sama lain dengan menggunakan data, baik secara manual maupun secara komputerisasi sehingga sistem dapat berfungsi dengan baik.
2. Sebagai alat pembuatan model yang menekankan pada fungsi sistem tersebut. Media perancangan sistem mudah dikomunikasikan oleh profesional kepada user maupun pembuat program.

3.7.3. *Entity Relationship Diagram (ERD)*

ERD adalah diagram yang memaparkan konsep relasi antara rancangan data tersimpan atau file yang digunakan untuk menggambarkan struktur data untuk menentukan hak pemakai (*user*).

Entity Relationship Diagram (ERD) dapat dikatakan sebagai *high level conceptual data model* yang dikembangkan oleh Chen (1976) konsep dasar yang memfasilitasi desain *database* termasuk entitas, hubungan dan atribut.

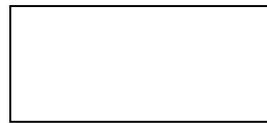
Fungsi-fungsi *ERD* adalah sebagai berikut:

1. Memperjelas hubungan antar entitas.
2. Menyatakan jumlah entitas dan pembatasan partisipasi antar entitas.
3. Penggunaan mudah dipahami.
4. Mudah disajikan oleh perancang basis data.

Simbol yang digunakan dalam *ERD*, antara lain:

1. Entitas

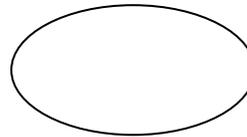
Entitas merupakan suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat.



Gambar 3.6 Entitas

2. Atribut

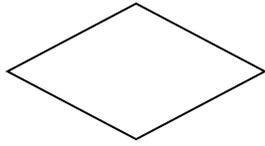
Atribut merupakan elemen yang berfungsi untuk menjelaskan karakter dari entitas. Nilai atribut adalah data aktual atau informasi yang disimpan pada atribut dalam suatu entitas atau hubungan.



Gambar 3.7 Atribut

3. Hubungan atau relasi

Relasi berfungsi untuk menghubungkan entitas yang satu dengan yang lainnya. Sebagaimana halnya entitas maka dalam hubungannya harus dibedakan antara hubungan atau bentuk hubungan antar entitas dengan isi dari hubungan itu sendiri.



Gambar 3.8 Hubungan

Jenis-jenis relasi antara lain:

1. Satu ke satu (*one to one*), yang berarti setiap entitas hanya dapat berhubungan dengan satu entitas lain, begitupun sebaliknya.
2. Satu ke banyak (*one to many*), yang berarti setiap entitas dapat berhubungan dengan lebih dari satu entitas lain, tetapi tidak sebaliknya.
3. Banyak ke banyak (*many to many*), yang berarti setiap entitas dapat berhubungan dengan entitas lain, begitupun sebaliknya.
4. Penghubung antara himpunan relasi/*link*. *Link* berfungsi untuk menghubungkan antara entitas dengan atribut atau entitas dengan hubungannya.

Gambar 3.9 *Link*

Ada beberapa pedoman yang perlu diperhatikan untuk membuat *ERD*, yaitu sebagai berikut:

1. Entitas harus memiliki banyak kejadian.
2. Hindari penggunaan atribut yang tidak perlu.
3. Berilah label yang jelas untuk semua komponen.
4. Pasangkan kardinalitas dan modalitas yang jelas dan benar.
5. Pecah atribut menjadi level serendah mungkin yang diperlukan.
6. Label harus merefleksikan istilah-istilah bisnis yang umum.
7. Asumsi harus disebutkan dengan jelas.

3.8. Tools Perangkat Lunak

Tools yang digunakan untuk membangun suatu sistem, yaitu sebagai berikut:

3.8.1. HTML, CSS, Javascript

1. HTML

1. Pengertian

HTML merupakan singkatan dari *Hypertext Markup Language* yang berisi sekumpulan tag-tag yang dituliskan dalam suatu *file* yang digunakan untuk membuat tampilan pada *website*. Dikatakan *markup language* karena *HTML* berfungsi untuk memformat file dokumen teks biasa untuk bisa ditampilkan pada *web browser* dengan bantuan tanda-tanda yang sudah ditentukan. Hal tersebut dapat dilakukan dengan menambahkan elemen atau yang sering disebut dengan tag. Tag yang digunakan oleh *HTML* ditandai dengan tag pembuka lebih kecil (<) dan tag penutup tanda lebih besar (>). Misalnya pasangan dari tag <contoh> adalah </contoh>. Dalam hal ini <contoh> disebut sebagai elemen dan biasanya dalam suatu elemen terdapat atribut-atribut untuk mengatur elemen itu. Jadi misalnya elemen <contoh> bila ditulis dengan atributnya, yaitu:

```
<contoh atribut1 = "atribut1" atribut2 = "atribut2"...>
```

Dalam penulisan tag *HTML* tidaklah case sensitive, artinya penggunaan huruf kecil ataupun besar tidaklah menjadi masalah.

HTML adalah bahasa pemrograman yang bebas dan tidak dimiliki oleh siapapun, pengembangannya dilakukan oleh banyak orang, dan bisa dikatakan sebagai sebuah bahasa yang dikembangkan bersama-sama secara global. Untuk itulah berkonsekuensi bahwa *platform web* adalah *platform* bebas.

2. Struktur dasar

Struktur dasar *HTML*, yaitu:

```
<html>
<head>
<title>berisi teks yang akan muncul pada titlebar
browser</title>
</head>
<body>
Berisi tentang teks, gambar, atau apapun yang ingin
ditampilkan pada halaman web dan pada bagian ini
</body>
</html>
```

Bagian yang diapit oleh tag `head` merupakan header dari halaman *HTML* dan tidak ditampilkan pada window browser. Bagian ini berisi tag-tag *header* seperti `<title>...</title>` yang berfungsi untuk mengeluarkan judul pada *title bar window web browser* dan tag lain, misalnya `<meta>`.

Bagian kedua, yang diapit oleh tag `body` merupakan bagian yang akan ditampilkan pada *window web browser* nantinya. Pada bagian ini akan dituliskan semua jenis informasi yang berupa teks dengan bermacam format maupun gambar yang nantinya akan disampaikan kepada pembaca.

Secara garis besar, terdapat 4 jenis elemen dari *HTML*, yaitu sebagai berikut:

1. Struktural

Tanda yang menentukan level atau tingkatan dari tulisan. Misalnya, `<h1>Saya</h1>` memerintahkan *browser* untuk menampilkan “Saya” sebagai tulisan tebal besar yang menunjukkan sebagai *heading 1*.

2. Presentasi

Tanda yang menentukan tampilan dari sebuah tulisan tidak peduli dengan level dari tulisan tersebut. Misalnya, `tebal` akan menampilkan tulisan yang tebal. Tanda presentasi ini sudah sering digunakan oleh CSS dan tidak

direkomendasikan untuk mengatur tampilan tulisan melalui HTML.

3. Hiperteks

Tanda yang berfungsi untuk membuat *link* di dokumen, baik *link* di dokumen itu sendiri atau ke dokumen lain. Misalnya ` Google ` akan membuat *link* ke situs google.

4. Elemen widget

Tanda yang membuat objek-objek lain. Misalnya tombol `<button>`, list ``, dan garis horizontal `<hr>`.

3. Kode warna

Pengaturan warna di halaman HTML menggunakan mode kombinasi RGB (red, green, blue) yang mana setiap warna ditampilkan dalam dua digit nilai hexadesimal (0, 1, 2,...,F). Setiap bagian dua digit kode menunjukkan banyaknya intensitas dari kombinasi warna merah, hijau dan biru. Sebagai contoh 00 pada dua digit pertama berarti tidak ada warna merah dalam kombinasi warna dan seterusnya. Berikut ini adalah contoh beberapa kode warna:

Tabel 3.1 Kode Warna

Warna	Heksadesimal
White	#FFFFFF
Black	#000000
Red	#FF0000
Green	#00FF00
Blue	#0000FF
Magenta	#FF00FF
Cyan	#00FFFF
Yellow	#FFFF00

4. Pengaturan properti halaman *web*

Properti dokumen diatur melalui atribut-atribut yang terdapat dalam elemen `<body>`. Sebagai contoh adalah

pengaturan warna latar belakang halaman, warna teks, warna link dan lain-lain.

Atribut elemen body, yaitu:

Tabel 3.2 Atribut Elemen *Body*

Atribut	Penjelasan
Background	Lokasi dan nama file (latar belakang image dokumen)
Bgcolor	Warna (warna latar belakang dokument, default putih)
Text	Warna (warna teks dokumen, default hitam)
Link	Warna (warna link dokumen, default biru)
Vlink	Warna (warna visited link dokumen, default ungu)
Alink	Warna (warna active link dokumen, default merah)

5. Heading, paragraph, break, dan horizontal ruler

Tag heading `<hx>` digunakan untuk memformat heading judul dan sub-judul dari suatu halaman *web*. Heading ini akan memperbesar ukuran huruf untuk setiap jenis heading. Ada enam buah heading yang dikenal di HTML, yaitu dari `<h1>` sampai `<h6>`.

Tag paragraph `<p>` berfungsi layaknya untuk pengaturan antar paragraph dalam halaman web. Dalam elemen paragraph terdapat atribut: `align = (left, center dan right)` yang berfungsi sebagai pengatur perataan paragraph, jadi cukup memilih salah satu dari ketiga pilihan tanpa harus memberi kurung buka dan tutup, dan defaultnya adalah `left`.

Tag break `
` berfungsi untuk memberikan baris baru suatu paragraph dalam halaman *web*. Tag break tidak memerlukan tag penutup break.

Tag horizontal ruler `<hr>` berfungsi untuk menampilkan garis horizontal tiga dimensi di dalam halaman web. Tag horizontal rules juga tidak membutuhkan elemen penutup.

6. Pemformatan karakter

Karakter pada halaman HTML dapat diformat sesuai desain yang ditentukan, baik ukuran, jenis, maupun warna dengan menggunakan tag ``.

Atribut elemen font, yaitu:

Tabel 3.3 Atribut Elemen *Font*

Atribut	Penjelasan
Size	Angka (ukuran huruf, default 3)
Color	Warna (warna huuf, default black)
Face	Jenis huruf (jenis huruf, default times new roman)

Elemen ragam karakter, yaitu:

Tabel 3.4 Elemen Ragam Karakter

Elemen	Penjelasan
<code>teks disini</code>	Menghasilkan huruf tebal
<code><i>teks disini</i></code>	Menghasilkan huruf miring
<code><u>teks disini</u></code>	Menghasilkan huruf bergaris bawah

7. Listing

Properti `` digunakan untuk menampilkan informasi dalam bentuk daftar (list). Ada dua jenis daftar yang dikenal di HTML, yaitu daftar dalam format bullet (unordered list ``) dan dalam bentuk nomor (ordered list ``).

8. Menempelkan gambar

Anda dapat menampilkan gambar dalam halaman HTML dengan format file JPG atau GIF. Untuk menampilkanya, gunakan tag IMG.

```
<IMG SRC=nama file gambar>
```

Atribut elemen *image*:

Tabel 3.5 Atribut Elemen *Image*

Atribut	Penjelasan
Src	Lokasi dan nama gambar
Alt	Teks (teks alternatif)
Width	Pixel (lebar gambar)
Height	Pixel (tinggi gambar)
Align	top, middle, bottom, left, right (perataan gambar)
Border	Pixel (tebal garis tepi gambar)

9. Membuat link

Elemen link berfungsi sebagai penghubung antara suatu halaman dengan halaman lain, atau ke URL lain, bahkan dalam satu halaman untuk berpindah ke sub judul yang lain. Format tag link adalah sebagai berikut:

```
<a href = URL_tujuan>hypertext</a>
```

URL_tujuan bernilai lokasi dan nama file yang akan dituju, sedangkan hypertext nilainya akan ditampilkan di browser sebagai *text link* atau tombol penghubung.

10. Membuat tabel

Tabel merupakan cara untuk menampilkan informasi dalam bentuk sel yang terdiri atas baris dan kolom. Untuk menampilkan data dalam bentuk tabel pada *HTML*, gunakan tag `<table> ... </table>`. Elemen tabel berisi properti `<tr>.. </tr>` untuk menentukan baris (*table row*) yang di dalamnya terdapat properti `<td>... </td>` untuk menampilkan data pada setiap sel tabel (*table data*).

Atribut elemen tabel:

Tabel 3.6 Atribut Elemen Tabel

Atribut	Penjelasan
Width	Panjang (lebar label, pixel atau persen)
Height	Panjang (tinggi label, pixel atau persen)
Border	Pixel (tebal garis tepi)
Cellspacing	Pixel (spasi antar sel)
Cellpadding	Pixel (spasi di dalam sel)
Align	Left, center, right (perataan tabel)
Bgcolor	Warna (warna latar belakang tabel)

11. Membuat form

Form *HTML* merupakan tag yang paling penting, dalam pemrograman internet khususnya dalam pembuatan aplikasi berbasis *web*. Form menyediakan properti masukan yang dapat berupa *textbox*, *check box*, *radio button*, *clan button*. Untuk mendeklarasikan sebuah form digunakan tag `<form>...</form>`.

Di dalam tag ini didefinisikan elemen-elemen form seperti yang telah disebutkan di atas. Selain tag, elemen form juga dapat menuliskan sembarang teks, tag, maupun Image.

Atribut elemen *form*:

1. *action* = lokasi dan nama file (file yang menangani form)
2. *method* = get, post (metoda HTTP untuk men-submit form)

Properti Masukan pada Elemen *Form*, yaitu:

1. Text Box `<input type="text">`
Digunakan untuk memasukkan input yang berupa teks.
size = ukuran dari textbox dalam karakter, default 20
maxlength = maksimal banyaknya karakter yang dapat diterima
name = nama dari variabel yang dikirim ke suatu aplikasi
value = akan menampilkan isinya sebagai nilai default
2. Password `<input type="password">`
Digunakan untuk memasukkan password.
size = ukuran dari textbox dalam karakter, default 20
maxlength = maksimal banyaknya karakter yang dapat diterima
name = nama dari variabel yang dikirim ke suatu aplikasi
3. Hidden `<input type="hidden">`
Digunakan untuk mengirim data ke suatu aplikasi yang tidak diinginkan untuk dilihat oleh browser.
name = nama dari variabel yang dikirim ke suatu aplikasi
value = nilai dari variabel
4. CheckBox `<input type="checkbox">`
Check box digunakan untuk dapat memilih lebih dari satu pilihan.
name = nama dari variabel yang dikirim ke suatu aplikasi
value = nilai dari variabelchecked (checkbox yang sudah ditandai)
5. Radio Button `<input type="radio">`

Radio button digunakan agar dapat memilih hanya salah satu pilihan.

name = nama dari variabel yang dikirim ke suatu aplikasi

value = nilai dari variabel checked (radio button yang sudah ditandai)

6. Push Button `<input type="button">`

Elemen ini biasanya digunakan dengan JavaScript atau VBScript untuk menghasilkan suatu aksi.

name = nama dari variabel yang dikirim ke suatu aplikasi

value = label teks di atas tombol

7. Submit `<input type="submit">`

Setiap elemen form membutuhkan tombol submit untuk mengirimkan nama dan nilainya ke suatu aplikasi yang ditentukan dalam atribut ACTION dalam elemen FORM.

name = nama dari variabel yang dikirim ke suatu aplikasi

value = label teks di atas tombol

8. Image Submit Button `<input type="image" src="url">`

Digunakan untuk menggantikan tombol standar submit dengan image.

name = nama dari variabel yang dikirim ke suatu aplikasi.

9. Reset `<input type="reset">`

Digunakan untuk mereset semua masukan dalam form.

value = ext label on the button

10. Text Area `<textarea> ... </textarea>`

Elemen untuk memasukkan teks secara leluasa seperti Notepad.

name = nama dari variabel yang dikirim ke suatu aplikasi

rows = panjang baris dalam karakter

Cols = tinggi dalam karakter

11. Select `<select> ... </select>`

Daftar isi dalam properti select menggunakan tag `<option>`

size = jumlah pilihan yang dapat terlihat

name = nama dari variabel yang dikirim ke suatu aplikasi

2. CSS

CSS merupakan singkatan dari *Cascading Style Sheet*. CSS dikembangkan pada 1996 oleh W3C (*World Wide Web Consortium*) yang digunakan untuk mengatur tampilan atau *style* dari elemen yang ada pada halaman yang dibuat dengan *HTML*. Tujuan dari pemakaian CSS ini sendiri untuk meminimalisir pengkodean yang ada pada *HTML* agar tidak menulis kode yang sama secara berulang-ulang untuk desain yang sama.

3. Javascript

Selain *HTML* dan *CSS* ada teknologi lain untuk sebuah *website* yaitu *Javascript*. *Javascript* pertama kali mulai diperkenalkan oleh *Netscape* dengan nama *Livescript*. Karena pada saat itu teknologi sedang tren, maka pihak dari *Netscape* memutuskan untuk mengganti namanya menjadi *Javascript* karena lebih *marketable* dibandingkan *Livescript*. Fungsi dari *Javascript* yaitu untuk menciptakan keinteraktifan antara halaman *web* dan *user*.

Pemakaian *JavaScript* dalam pembuatan *web* adalah dengan memasukkannya dalam *HTML*. *JavaScript* sebagai sebuah bahasa pemrograman untuk client dan server mempunyai elemen-elemen sebagai berikut:

- a. Kata kunci (keyword), statement, syntax dan grammar
- b. Aturan untuk skpresi, variabel dan literal
- c. Objek dan fungsi built-in.

Struktur javascript, yaitu:

```
<script type = "text/JavaScript">  
statement JavaScript  
</script>
```

Atau

```
<Script language = "JavaScript">  
statement JavaScript  
</script>
```

3.8.2. Bootstrap

Bootstrap adalah *framework front-end* yang intuitif dan powerful untuk pengembangan aplikasi *web* yang lebih cepat dan mudah. *Bootstrap* menggunakan *HTML*, *CSS*, dan *Javascript*. *Bootstrap* dikembangkan oleh Mark Otto dan Jacob Thornton dari twitter pada agustus 2011 di *Github*.

Sebagai sebuah *framework*, *bootstrap* menyediakan *template* yang mendefinisikan *style* dasar seluruh elemen *HTML* dalam sebuah *website*. Hal ini akan mempermudah dalam pembuatan *website* dengan keseragaman visual tanpa harus mendefinisikan *style* atribut untuk setiap elemen *HTML* secara berulang-ulang.

Selain elemen *HTML* dan *CSS*, *bootstrap* juga menyediakan beberapa komponen *JavaScript* dalam bentuk *plugin jQuery*. Adanya *plugin jQuery* ini memungkinkan *user bootstrap* untuk menggunakan beberapa fitur interaktif seperti *dialog box*, *tooltips*, *corousel*, dan sebagainya.

Bootstrap memiliki fitur-fitur *interface* yang bagus seperti *typography*, *form*, *button*, *tables*, *navigations*, *dropdowns*, *alerts*, *modal*, *tabs*, *accordion*, *carousel*, dan lain sebagainya.

Kelebihan-kelebihan *bootstrap*, yaitu:

1. Menghemat waktu

Dengan menggunakan *bootstrap*, kita dapat menghemat waktu dan tenaga karena *bootstrap* menyediakan berbagai desain *template* dan kelas yang dapat digunakan.

2. Fitur yang responsif

Bootstrap memiliki fitur desain yang responsif sehingga dapat memudahkan kita dalam mendesain sistem. Dengan fitur-fitur ini, halaman *web* akan tampil responsif di berbagai perangkat yang berbeda tanpa perlu mengubah kode markapnya.

3. Desain yang konsisten

Bootstrap memiliki desain dan *style* yang sama sehingga membuat tampilan *web* menjadi konsisten.

4. Mudah digunakan

Hanya dengan pengetahuan dasar *HTML* dan *CSS*, kita sudah bisa menggunakan *bootstrap* karena begitu mudah digunakan.

5. Didukung oleh semua *browser* populer

Bootstrap dapat digunakan pada semua *browser* modern seperti *mozilla firefox*, *google chrome*, *safari*, *internet explorer*, dan *opera*.

6. Gratis

Bootstrap merupakan *framework open source* yang dapat digunakan secara gratis.

Kekurangan-kekurangan *bootstrap*, yaitu:

1. Kemiripan dengan *website bootstrap* lain

Bootstrap adalah sebuah *framework* yang untuk menerapkan gaya visual pada sebuah *website*. Pengembangan *website* ini bertujuan untuk mempermudah pengembangan *website* dengan menggunakan *template* dalam kode yang digunakan. Oleh karena itu, sebagian besar *website* memiliki tampilan yang serupa. Jika ingin tampilan *website* yang berbeda, maka *developer* harus melakukan banyak modifikasi untuk kode pada *website* tersebut.

2. Perlu proses pembelajaran

Meskipun *bootstrap* dapat dipelajari dengan mudah, tapi membutuhkan waktu untuk mempelajari berbagai *CSS class* yang dapat digunakan. Selain itu, perlu juga beradaptasi dan bereksprimen dengan *style grid* yang digunakan oleh *bootstrap*.

3. Dapat memperlambat *website*

Bootstrap merupakan sebuah *framework* yang mempermudah pembuatan *website* responsif dan *mobile-friendly*. Namun jika tidak melalui proses optimasi yang baik, *website bootstrap* cenderung memiliki performa yang lebih buruk dalam segi *load time* untuk para *user*. Selain itu, *file* yang dibuat oleh *bootstrap* dapat mencapai ukuran relatif lebih besar

dibandingkan dengan *framework* lain, yang tentunya dapat memperlambat *website*.

3.8.3. PHP

PHP merupakan singkatan dari *Hypertext Preprocessor* yang dibuat pertama kali oleh Rasmus Lerdoft pada tahun 1994 (Solichin, 2016).

PHP adalah bahasa *scripting* yang berintegrasi dengan *HTML* dan dijalankan pada *server side* yang artinya semua sintaks yang dikelola akan dijalankan pada *server* dan yang dikirimkan ke *browser* hanya hasilnya saja berupa *HTML*. *PHP* merupakan bahasa yang paling sering digunakan oleh *programming web* karena merupakan pemrograman *open source*, sehingga para programming tidak perlu membeli lisensi untuk membuat suatu aplikasi *web*.

Kode program *PHP* dapat ditulis menyatu dengan *tag-tag HTML* dalam *file*. Kode *PHP* diawali dengan *tag* `<?php` dan ditutup dengan *tag* `?>`. *Tag* tersebut disisipkan diantara *tag* `<body>` dengan *tag* `</body>` dalam *tag-tag HTML*. Isi dari *tag PHP* merupakan *script* dari *PHP* yang akan diproses. *Script PHP* dapat dibuat dengan menggunakan *editor VI* pada *linux* atau *Advanced Editor* dan disimpan dengan ekstensi “.php”.

Web server akan menjalankan *PHP* untuk menerjemahkan bagian halaman tersebut. Halaman *Web* yang dilihat oleh *user* adalah gabungan antara perintah *HTML* standar dan *output* dari *interpreter PHP*.

Contoh penulisan kode *PHP* sebagai berikut:

```
<html>
<head>
  <title> Coba Menuliskan PHP </title>
</head>
<body>
<?php
```

```
$bilang = "Hello World";  
echo $bilang;  
?>  
</body>  
</html>
```

1. Komentar program

Komentar pada *PHP* juga mirip dengan sintaks komentar pada 'C'. 'C++' maupun *shell Unix*. Dapat digunakan tanda `//` untuk komentar satu baris atau `/*` diakhiri `*/` untuk komentar lebih dari satu baris atau bahkan menggunakan tanda `#` di awal komentar seperti sintaks komentar pada *Unix shell*.

2. Variabel

Variabel adalah tempat yang digunakan untuk menyimpan *value*, data atau informasi. Cara pendeklarasian variabel di *PHP* yaitu dengan memberikan tanda dolar (\$) dan kemudian karakter \$ disambung dengan nama variabel. Perlu diperhatikan bahwa nama variabel bersifat *case-sensitive*.

Untuk penamaan variabel, ada beberapa aturan yaitu:

- a. Nama variabel tidak boleh spasi, dan jika ingin dipisah maka harus diberi tanda *underscore* (`_`).
- b. Nama variabel tidak boleh diawali dengan angka.
- c. Pada penulisan nama variabel, tidak boleh ada operator aritmatika.
- d. Tidak boleh mengandung karakter khusus seperti: `%`, `@`, `#`, `!` dan lain-lain.

3. Tipe data

Tipe data adalah jenis data yang digunakan untuk pengolahan data di dalam komputer, dimana data ini sangat dibutuhkan di dalam pemrograman komputer. Jika salah memilih tipe data, maka program menjadi error atau akan terjadi pemborosan memori, karena tipe data ini erat kaitannya

dengan memori di dalam komputer. Cara penulisan tipe data cukup dengan mengisi tanda dolar (\$) di depan sebuah variabel.

\$nama_variabel

PHP mempunyai delapan jenis tipe data yang dibagi atas tiga bagian, yaitu:

Tipe data dasar, yaitu:

- a. *Boolean*
- b. *Integer*
- c. *Float*
- d. *String*

Tipe *compound*/tipe campuran:

- a. *Array*
- b. *Object*

Tipe *special*/tipe khusus:

- a. *Resource*
- b. *Null*

a. Tipe *boolean*

Boolean adalah tipe data yang mempunyai *output* nilai *true* dan *false*. Biasanya *boolean* digunakan untuk menyatakan kondisi. Artinya jika kondisi bernilai benar, program akan menjalankan suatu *action* tertentu, dan jika salah program akan menjalankan *action* yang lain.

b. Tipe *integer*

Integer merupakan tipe data yang digunakan untuk menyimpan bilangan bulat. Contohnya: 1,2,3,-1,-2,-3 dan lain-lain. Penulisan bilangan *integer* bisa dalam notasi desimal, hexadesimal, atau bisa juga dalam bentuk oktal, termasuk penanda - atau +.

Bilangan *integer* dalam PHP dapat dituliskan dalam bentuk sebagai berikut:

1. Bilangan desimal, contoh \$a = 8765;

2. Bilangan negatif desimal, contoh \$a = -354\$;
3. Bilangan oktal, contoh \$a = 017\$; sama dengan bilangan desimal 15;
4. Bilangan heksadesimal, contoh \$a = 0x13\$; = bilangan desimal 19.

c. Tipe *floating point*/tipe pecahan

Floating point adalah bilangan pecahan dengan presisi tertinggi. Bagian dari *floating point* yaitu *float*, *double*, dan *real*.

Bilangan *floating point* dapat dideklarasikan menggunakan sintaks berikut:

\$a=0.134\$;atau \$a=1.34e-1\$;

Nilai maksimum sebuah bilangan *floating point* adalah $\sim 1.8e308$ dengan ketelitiannya mencapai 14 digit desimal.

d. Tipe *string*

String adalah tipe data untuk menampung sederetan kata (kalimat) ataupun juga bisa menampung kata tunggal. *String* juga sering disebut dengan sekumpulan karakter, dan besarnya setiap karakter sama dengan *byte*.

String dapat dituliskan dengan tiga cara, yaitu:

1. Petik Tunggal

Cara penulisan *string* dengan petik tunggal harus diawali dengan petik tunggal dan diakhiri dengan petik tunggal juga ('.....'). Tetapi jika ingin menambahkan karakter khusus, maka harus dimasukkan (\) sebelum karakter khusus tersebut.

2. Petik ganda

Cara penulisan *string* dengan petik ganda harus diawali dengan petik ganda dan juga diakhiri dengan petik ganda ("....."). Dalam hal penulisan hampir sama dengan petik tunggal, tetapi penulisan dengan sintak-sintak petik ganda lebih fleksibel dan banyak karakter spesial yang bisa dimasukkan.

Tabel 3.7 Karakter Spesial Petik Ganda

Karakter	Pengertian
\n	Garis baru
\r	Enter/kembali ke paling kiri
\t	Tabulasi
\\	Backslash
\\$	Dolar
\"	Kutip ganda
\[0-7]{1,3}	Karakter yang ditampilkan dengan kode oktal
X[0-9a-Fa-f]{1,2}	Karakter yang ditampilkan dengan kode hexadesimal

Jika menuliskan karakter *backslash* (\) diikuti dengan karakter lain selain yang terdapat pada daftar tabel di atas, maka karakter *backslash* dan karakter di depannya akan ikut tercetak di tampilan.

3. Sintaks Heredoc

Untuk mendefinisikan suatu *string* selain cara di atas, bisa dilakukan dengan cara menggunakan *sintaks heredoc*. Untuk mendeklarasikan *sintaks heredoc* (<<<), harus dibuat nama pengenalan (*identifier*). Setelah <<< kemudian di baris baru, buat *string* yang diinginkan, kemudian di baris terakhir isi nama pengenalan (*identifier*) yang dibuat sebelumnya.

e. Tipe *array*

Array merupakan kumpulan data, di mana data yang disimpan tersebut dimasukkan dalam variabel, dan variabel tersebut memiliki nama yang sama.

Dalam membuat variabel *array* diperbolehkan menggunakan tipe data yang berbeda. Ini yang membedakan antara tipe *array* di PHP dengan bahasa pemrograman lain, di mana pada bahasa pemrograman lain, untuk mendeklarasikan suatu *array* harus tipe data yang sama. Hal ini dikarenakan pada *PHP* tidak perlu pendeklarasian variabel. Oleh sebab itu, setiap elemen *array* bisa dibuat berbagai macam tipe data dalam suatu nama variabel *array*.

Array dapat dibuat dengan pernyataan `array()`, yang di dalamnya ditambahkan dengan "*key => value*". Jika lebih dari satu, maka untuk membedakan "*key => value*" satu dengan yang lainnya dipisahkan dengan tanda (",").

Sintak dasar:

```
Array (key => value, key2 => value2,.....)
```

Key disini bisa bertipe *string*, *integer*, ataupun *float*. Jika bertipe *float*, maka pada saat pemanggilan *key* tersebut akan dipanggil menjadi *integer*. Dan untuk *value*, bisa bertipe apa saja dan bisa berbeda satu dengan yang lainnya.

f. Tipe data *object*

Untuk menginisialisasikan sebuah objek, digunakan perintah *new* seperti contoh sebagai berikut:

```
Class satu{  
Function hitung() {  
Echo "saya sedang belajar."  
}  
}  
$uji = new satu;  
$uji->hitung();
```

Program diatas akan menghasilkan tulisan "saya sedang belajar" yang dikerjakan pada bagian fungsi `hitung()`.

g. Tipe *null*

Nilai *null* terjadi ketika suatu variabel tidak diberi nilai sama sekali. Di dalam *PHP*, nilai *null* terjadi karena tiga hal, yaitu:

1. Karena variabel memang diberi nilai *null*.
2. Variabel tidak diberi nilai sama sekali atau belum di-set.
3. Jika diberi perintah `unset()`.

4. Konstanta

Konstanta memiliki fungsi yang sama dengan variabel, yaitu menyimpan data, bedanya adalah data yang disimpan konstanta nilainya tetap atau tidak bisa dirubah sama sekali. Konstanta bersifat *case-sensitive* yang berarti huruf besar atau kecil berpengaruh.

Berikut adalah perbedaan antara konstanta dan variabel:

- a. Sebelum nama konstanta, tidak terdapat tanda dolar (\$).
- b. Konstanta didefinisikan dengan fungsi `define()`, selain fungsi tersebut tidak bisa.
- c. Konstanta tidak mengikuti aturan *scope* seperti pada variabel, artinya bisa ditempatkan dimana saja.
- d. Nilai konstanta tidak bisa diubah atau dihapus.
- e. Konstanta diperbolehkan hanya nilai *scalar*.

5. Operator

Operator adalah sesuatu yang digunakan untuk memanipulasi dan mengubah suatu nilai variabel yang sudah kita masukkan menjadi sesuatu yang diinginkan. Nilai variabel yang dimodifikasi oleh operator disebut *operand*. Misalnya, `20 + 30`, `20` dan `30` tersebut merupakan *operand* sedangkan tanda `+` itu merupakan operator. Di dalam pemrograman *PHP* ada tiga operator utama, antara lain:

- a. Operator unary: Operator yang hanya menggunakan satu buah variabel untuk pengoperasiannya, yaitu operator negasi (`!`), *decrement* (pengurangan) dan operator *increment* (penambahan). Contoh pendeklarasian variabel: `$b++`, `--$b`.
- b. Operator binary: Terdiri atas beberapa operator yang saling bekerja satu sama lain, akan tetapi pengerjaan operator tergantung urutan penyelesaiannya. Operator inilah yang paling sering diterapkan dalam pemrograman *PHP*.
- c. Operator ternary: Operator yang sering digunakan untuk menggantikan fungsi `if else` karena operator ini

membandingkan dua buah pernyataan dan pernyataan ketiga menjadi hasilnya.

Kondisi ? jika benar: jika salah;

\$contoh = \$age > 16 ? 'adult': 'child';

Sintak di atas merupakan contoh operator ternary. Jika nilai *age* lebih besar dari 16 maka dimunculkan hasilnya *adult* dan jika tidak maka memunculkan hasil *child*.

1. Operator aritmatika

Semua bahasa pemrograman pasti menggunakan operasi aritmatika. Operator aritmatika adalah operator yang sangat umum, yaitu:

Tabel 3.8 Operator Aritmatika

Operator	Penjelasan
+	Tambah
-	Kurang
*	Kali
/	Bagi
%	Modulus (sisanya hasil bagi)

Contoh:

\$a = 30;

\$b = 10;

Maka bisa dibuat sebagai berikut:

Tabel 3.9 Hasil Operator Aritmatika

Operator	Proses	Hasil
+	\$a + \$b	40
-	\$a - \$b	20
*	\$a * \$b	300
/	\$a / \$b	3
%	\$a % \$b	0

Operator aritmatika memiliki aturan penulisan, yaitu harus dibuat variabel = ekspresi aritmatika. Artinya nilai ekspresi aritmatika akan diberikan kepada variabel. Sebelum sama dengan, harus ada satu variabel tunggal dan setelah sama

dengan berupa ekspresi atau rumus matematika, konstanta atau variabel.

2. Operator karakter

Operator karakter adalah operator yang digunakan untuk menangani karakter-karakter atau string. Fungsi operator karakter adalah untuk menggabungkan karakter-karakter atau menggabungkan string.

3. Operator penugasan

Operator penugasan adalah operator yang dalam pengisian nilai variabelnya menggunakan tanda sama dengan (=). Fungsi dari operator penugasan adalah untuk memberikan sebuah nilai dalam variabel.

4. Kombinasi operator penugasan

Pada operator penugasan, yang ingin melakukan penambahan nilai di mana nilai tersebut harus dijumlahkan dengan nilai yang lain untuk menaikkan nilainya sendiri. Misalnya, $f = f + k$. Dari penulisan kode tersebut, terdapat dua variabel yang sama. Dengan menerapkan kombinasi operator penugasan, kode tersebut bisa dipersingkat menjadi $f += k$.

Tabel operator kombinasi penugasan, yaitu sebagai berikut:

Tabel 3.10 Operator Kombinasi Penugasan

Penugasan	Sama seperti	Penjelasan
$f = k$	$f = k$	Variabel di kiri akan tergantung nilai variabel di kanan
$f += k$	$f = f + k$	Penambahan
$f -= k$	$f = f - k$	Pengurangan
$f *= k$	$f = f * k$	Perkalian
$f /= k$	$f = f / k$	Pembagian
$f \% = k$	$f = f \% k$	Modulus
$f . = k$	$f = f . k$	Menggabungkan string

5. Operator penugasan menurunkan dan menaikkan nilai

Operator penugasan ada juga yang untuk menaikkan atau menurunkan suatu nilai dan cara penulisan dan fungsinya pun hampir sama. Tetapi, dalam operator, penugasan yang digunakan sebagai penambah dan pengurang suatu nilai adalah angka satu.

Contoh operator penugasan menaikkan dan menurunkan nilai, yaitu:

a. ++Variabel

Digunakan untuk menaikkan angka bernilai satu pada variabel tersebut setiap pernyataan ini dikerjakan.

b. -Variabel

Digunakan untuk menurunkan angka bernilai satu pada variabel tersebut setiap pernyataan ini dikerjakan.

c. Variabel++

Digunakan untuk menaikkan angka bernilai satu pada variabel dengan syarat sebelumnya diberikan pernyataan yang sama.

d. Variabel—

Digunakan untuk menurunkan angka bernilai satu pada variabel dengan syarat sebelumnya diberikan pernyataan yang sama.

6. Operator perbandingan

Operator perbandingan berfungsi untuk membandingkan dua nilai dan hasilnya bernilai benar atau salah. Perlu diingat bahwa di dalam operator PHP ada “=” dan ada “= =”. Kedua operator tersebut sangat berbeda, di mana operator “= =” artinya suatu kondisi yang bernilai benar atau salah, sedangkan operator “=” merupakan pengisian nilai. Hal tersebut harus diperhatikan dengan saksama karena kalau salah bisa berakibat buruk bagi program yang dibuat dan akan sulit mencari kesalahannya, karena secara sintak, PHP akan menganggap itu benar padahal setelah dijalankan hasilnya

tidak sesuai dengan yang diinginkan atau sering disebut dengan kesalahan logika.

Tabel 3.11 Operator Perbandingan

Operator	Nama	Penjelasan
$a == b$	Sama dengan	Benar jika nilai a sama dengan b
$a === b$	Identik	Benar jika nilai a sama dengan nilai b dan keduanya memiliki kesamaan type
$a != b$	Tidak sama dengan	Benar jika nilai a tidak sama dengan nilai b
$a <> b$	Tidak sama dengan	Benar jika nilai a tidak sama dengan nilai b
$a !== b$	Tidak identik	Benar jika nilai a tidak sama dengan nilai b dan keduanya tidak memiliki kesamaan type
$a < b$	Lebih kecil dari	Benar jika nilai a lebih kecil dari nilai b
$a > b$	Lebih besar dari	Benar jika nilai a lebih besar dari nilai b
$a <= b$	Lebih kecil sama dengan	Benar jika nilai a lebih kecil atau sama dengan nilai b
$a >= b$	Lebih besar sama dengan	Benar jika nilai a lebih besar atau sama dengan nilai b

7. Operator logika

Operator logika hampir sama fungsinya dengan operator perbandingan, yaitu sama-sama membandingkan dua nilai dan hasilnya benar atau salah juga. Operator ini biasanya ditempatkan di struktur percabangan (*if* dan *else/elseif*). Walaupun hampir sama, tetapi operator perbandingan memiliki perbedaan dengan operator logika, yaitu pada operator perbandingan tidak dapat dilakukan pada nilai *boolean*. Lebih jelasnya, pada tabel berikut:

Tabel 3.12 Operator Logika

Contoh	Nama	Penjelasan
\$x and \$y	And	True jika keduanya true
\$x or \$y	Or	True jika salah satunya true atau keduanya true
\$x xor \$y	Xor	True jika salah satunya true tetapi tidak keduanya
! \$y	Not	True jika \$y bukan true
\$x % \$y	And	Nilainya true jika keduanya true
\$x \$y	Or	True jika salah satunya true atau keduanya true

8. Operator ternary

Operator ternary berfungsi sebagai pengganti fungsi `if` dan `else/elseif`, tetapi terkadang tidak direkomendasikan untuk pemula karena bisa sangat membingungkan.

Beberapa kelebihan menggunakan operator ternary, yaitu:

- Menghemat waktu baik dalam melakukan pemrograman maupun *maintanance*.
- Kode *PHP* menjadi lebih rapi dan enak dilihat bagi seorang programmer.
- Kode yang dibuat menjadi lebih singkat daripada menggunakan struktur kontrol.

6. Struktur Kontrol

Struktur kontrol adalah program yang akan dikerjakan setiap barisnya sampai baris terakhir. Secara umum, struktur program memiliki tiga buah struktur kontrol, yaitu urutan (*sequence*), pemilihan (*section*), dan perulangan (*looping*).

a. *If*

Pernyataan *if* merupakan struktur kontrol yang digunakan untuk kasus percabangan. Apakah program akan menjalani proses di bawahnya atau meloncati proses tersebut

dan mengerjakan proses lain. Proses yang dikerjakan tersebut bisa satu atau lebih intruksi dalam satu kelompok.

b. *If... Else*

Kondisi *if ... else* memiliki dua pilihan, yaitu jika kondisi terpenuhi dia akan mengerjakan *statement A*, jika tidak terpenuhi akan mengerjakan *statement B*. Kondisi *if ... else* digunakan jika suatu kondisi tersebut memungkinkan untuk dua pilihan, dan jika lebih dari dua pilihan *if ... else* tidak akan bisa digunakan.

c. *If... Elseif... Else*

Pernyataan *if ... elseif ... else* memiliki percabangan lebih dari dua, pernyataan ini merupakan perkembangan dari pernyataan *if ... else* yang memiliki dua kondisi. Dalam pernyataan ini, pemecahan dari masalah membutuhkan banyak percabangan karena alternatif yang diperoleh banyak.

d. Struktur Kontrol Alternatif

Untuk pemrograman *PHP*, ada sintaks alternatif untuk struktur kontrol *if, while, for, foreach, dan switch*. Fungsi utama kontrol alternatif ini adalah untuk mengelompokkan beberapa intruksi yang akan dieksekusi. Cara penulisan struktur kontrol alternatif adalah diawali dengan titik dua (:) dan diakhiri dengan *end if, endwhile, endfor, endforeach, dan endswitch*.

e. *While*

While merupakan salah satu struktur kontrol untuk perulangan. Perulangan *while* akan mengecek suatu kondisi di awal blok *sintaks*. Jika kondisi bernilai benar maka akan terjadi perulangan terus menerus sampai kondisi bernilai salah.

f. *Do... While*

Do ... while fungsinya hampir sama dengan *while* akan tetapi *statement* ini akan melakukan pengecekan kondisi di akhir blok *statement*. Berbeda dengan *while*, pengecekan dilakukan di awal blok *statement*. Sama dengan *while* pengulangan juga hanya akan dilakukan jika kondisi yang didefenisikan di dalamnya terpenuhi (bernilai benar). Namun,

bila kondisi tidak terpenuhi, dalam pengulangan ini proses minimal akan dilakukan satu kali.

g. *For*

Struktur kontrol *for* merupakan *statement* perulangan yang digunakan untuk melakukan perulangan yang telah diketahui jumlahnya. Pengulangan dengan menggunakan *statement* ini, kita harus memiliki sebuah variabel sebagai indeksinya. Namun, tipe data dari variabel yang akan digunakan sebagai indeks haruslah tipe data yang mempunyai urutan yang teratur, misalnya tipe data *int* (0, 1, 2 ...) atau *char* ('a', 'b', 'c' ...).

h. *Foreach*

Pada *PHP* versi 4 ke atas, penggunaan *foreach* mulai digunakan. Penerapan *foreach* kebanyakan di *array* karena sangat mudah dan praktis dalam pengaksesan *array*.

i. *Break*

Break berfungsi untuk memberhentikan dan keluar dari perulangan (*for*, *foreach*, *while*, *do ... while*, dan *switch*), dan akan melanjutkan ke baris program berikutnya. *Break* biasanya disimpan di dalam suatu struktur kontrol percabangan yang sudah diisi suatu angka. Apabila bernilai benar, maka *break* akan dikerjakan dan perulangan akan berhenti.

j. *Continue*

Continue hampir sama dengan *break*, tetapi fungsinya adalah melewati suatu tahap perulangan dan mengerjakan ke perulangan berikutnya, namun masih dalam *scope* perulangan itu sendiri.

k. *Switch*

Pernyataan *switch* hampir sama dengan pernyataan *if ... elseif ... else*. Kedua pernyataan tersebut sama-sama merupakan percabangan, dan sama-sama bisa memiliki banyak percabangan. Akan tetapi, terdapat perbedaan antara *switch* dengan *if ... elseif ... else* yaitu pada *switch* variabel yang dijadikan syarat nilainya harus pasti dan tidak boleh berupa ekspresi.

3.8.4. Website

World Wide Web (WWW) atau biasa disebut *web*, merupakan salah satu dari sumber daya internet yang berkembang pesat. Informasi *web* didistribusikan melalui pendekatan *hypertext*, yang memungkinkan suatu teks pendek menjadi acuan untuk membuka dokumen yang lain. Dengan pendekatan hiperteks ini seseorang dapat memperoleh informasi dengan melompat dari satu dokumen ke dokumen lain. Dokumen-dokumen yang diakses pun dapat tersebar di berbagai mesin, dan bahkan di berbagai negara.

Website adalah layanan yang menyediakan informasi kepada pengguna yang mengakses *website* dengan menggunakan *hyperlink* untuk memudahkan dalam pencarian melalui internet. *Website* berisi kumpulan halaman dan *file-file* pendukung yang diletakkan di dalam suatu situs yang dikenali melalui nama domain (*domain name*) dan alamat *IP (IP address)*. Sebuah *website* akan lebih sering dikunjungi jika informasi yang disediakan dikemas secara menarik dan juga selalu menampilkan informasi yang baru atau *up to date*.

Untuk mengirimkan dokumen halaman *web* ke *client*, digunakan teknologi protokol *HTTP*, yaitu *Hypertext Transfer Protocol*. Protokol ini ditingkatkan menggunakan enkripsi sehingga menjadi *HTTP* yang bisa dipakai untuk *website* yang memerlukan keamanan lebih baik, seperti *e-commerce* dan *online banking*.

Untuk memasuki halaman *website*, pengguna harus memiliki *software* yang disebut *browser*. *Website-website* yang ada di dunia ini membentuk suatu “dunia maya” yang dinamakan *World Wide Web* atau *WWW*.

3.8.5. Database MySQL

Basis data (*database*) adalah kumpulan informasi yang terintegrasi dengan baik dan disimpan secara sistematis dalam

komputer sehingga dapat diperiksa menggunakan program komputer untuk memperoleh informasi dari *database*.

Dalam proses *database* diperlukan suatu *software* yang disebut *DBMS (Database Management System)* yang merupakan suatu sistem perangkat lunak yang digunakan oleh pengguna untuk membuat, memelihara, mengontrol, memanipulasi dan mengakses *database* secara praktis dan efisien. Contohnya seperti: *DB2, Microsoft SQL Server, Oracle, MySQL, Sybase* dan lain-lain.

Kelebihan dari *MySQL* sebagai berikut:

1. *Portability*

MySQL dapat berjalan dengan stabil pada berbagai sistem operasi diantaranya seperti *Windows, Linux, FreeBSD, Mac OS X Server, Amiga, HP-UX* dan lainnya.

2. *Open source*

MySQL disalurkan secara *open source* (gratis), dibawah lisensi *GPL* sehingga kita dapat menggunakannya dengan gratis tanpa dipungut biaya.

3. *Multiuser*

MySQL dapat dipakai oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah. Hal ini memungkinkan sebuah *database server MySQL* dapat diakses *client* secara bersamaan.

4. *Performance tuning*

MySQL mempunyai kecepatan yang mengagumkan dalam menangani *query* sederhana.

5. *Column types*

MySQL mempunyai tipe kolom yang sangat kompleks, seperti *integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp*.

6. *Command dan function*

MySQL mempunyai operator dan fungsi yang secara penuh mendukung perintah *SELECT* dan *WHERE* dalam *query*.

7. Security

MySQL mempunyai beberapa lapisan sekuritas seperti *level subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail beserta *password* terenkripsi.

8. Scalability dan limits

MySQL mampu menangani *database* dalam skala yang besar, dengan jumlah *record* lebih dari 50 juta dan 60 ribu table dalam 5 miliar baris. Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada tiap labelnya.

9. Connectivity

MySQL dapat melakukan koneksi dengan *client* dengan menggunakan protokol TCP/IP, Unix Soket (Unix), Named Pipes (NT).

10. Localisation

MySQL dapat mendeteksi pesan kesalahan (*error code*) dengan menggunakan lebih dari 20 bahasa.

11. Interface

MySQL mempunyai *interface* terhadap berbagai aplikasi dan bahasa pemrograman.

12. Client dan tools

MySQL mempunyai berbagai *tool* yang dapat digunakan untuk administrasi *database* dan setiap *tool* yang ada disediakan petunjuk *online*.

13. Struktur tabel

MySQL mempunyai struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan *database* lainnya semacam *Oracle*.

Kekurangan dari *MySQL*, yaitu:

1. Kurang cocok untuk aplikasi *game* dan *mobile*.
2. *Technical support* yang kurang memadai.
3. Sulit mengelola data yang besar.

1. Tipe data

Pada *MySQL* mendukung banyak sekali tipe data. Dari semua tipe data tersebut dikelompokkan menjadi tiga kategori yaitu:

a. Tipe numerik

Di dalam *MySQL* mendukung banyak sekali tipe data. Dari semua jenis tipe data numerik atau angka, yang termasuk dalam tipe data numerik adalah tipe data untuk bilangan bulat (contohnya *INTEGER*, *SMALLINT*, *DECIMAL*, dan *NUMERIC*) dan tipe pecahan (contohnya *FLOAT*, *REAL* dan *DOUBLE PRECISION*). Selain tipe data di atas, untuk menyesuaikan dengan standar *SQL*, *MySQL* juga mendukung tipe data *integer* lainnya seperti *TINYINT*, *MEDIUMINT*, dan *BIGINT*.

b. Tipe *date and time*

Di dalam *MySQL* mengenal tipe data yang digunakan untuk melakukan penanganan tanggal dan waktu. Tipe data tersebut antara lain *DATETIME*, *DATE*, *TIMESTAMP*, *TIME* dan *YEAR*.

Jenis-jenis tipe data *date* dan *time*:

1. *Datetime*

Ukuran 8 byte dan merupakan suatu kombinasi dari jam dan tanggal. Jangkauan tipe data ini dari '1000-01-01 00:00:00' sampai dengan '9999-12-31 23:59:59'.

2. *Date*

Ukuran 3 byte, tanggal dari jangkauan '1000-01-01' sampai dengan '9999-12-31'.

3. *Timestamp*

Ukuran 4 byte, kombinasi tanggal dan jam dimulai dari '1970-01-01 00:00:00' sampai dengan tahun 2037.

4. *Time*

Memiliki ukuran 3 byte dan memiliki jangkauan dari '-838:59:59' sampai dengan '838:59:59'.

5. *Year*

Ukuran 1 byte dan memiliki rang antara 1901 sampai 2155.

c. Tipe *character*

Di dalam *MySQL* terdapat tipe *character*, tipe ini berhubungan dengan *string*.

Adapun jenis-jenis tipe data *character* adalah:

Tabel 3.13 Tipe Data *Character*

Type	Keterangan
BINARY, CHAR(M)	Ukuran M byte, $1 \leq M \leq 255$. Data string dengan data yang selalu tetap
VARBINARY, VARCHAR (M)	Ukuran dari tipe ini adalah bersifat fleksibel dan maksimal M byte.
TINYBLOB, TINYTEXT BLOB, TEXT	Tipe yang mampu mengendalikan tipe data yang ukurannya besar, dan
MEDIUMBLOB, MEDIUMTEXT LONGBLOB, LONGTEXT	dalam hal ini tergantung kapasitas penyimpanannya seberapa besar

2. *Query SQL*

Ada beberapa hal penting yang dimiliki oleh *MySQL* antara lain:

- Perintah *SQL* secara umum selalu diakhiri dengan titik koma (*semicolon*). Tetapi ada pengecualian untuk perintah-perintah tertentu, misalnya perintah *QUIT* dan *USE*.
- Server* akan menerima perintah kiriman dari *MySQL* untuk dieksekusi dan akan ditampilkan hasilnya. Setelah itu akan muncul *prompt mysql>* yang menandakan bahwa *MySQL* sudah siap menerima perintah berikutnya.
- Tampilan hasil *query* dari *MySQL* adalah dalam bentuk tabular (baris dan kolom). Baris pertama berupa label dan baris kedua adalah hasil *query*-nya.

3. *Database* dan tabel

a. Membuat *database*

Untuk membuat *database* gunakan perintah *create database*. Untuk penamaan *database*, gunakan kesepakatan yang sudah ada yaitu huruf kecil semua.

b. Menghapus *database*

Untuk melakukan penghapusan *database*, tidak terlalu susah yaitu hanya dengan mengetikkan perintah *drop database*.

c. Membuat tabel

Setelah *database* dibuat dan dipilih dengan perintah *use*, selanjutnya yaitu membuat tabel.

d. Menghapus tabel

Untuk melakukan penghapusan tabel, digunakan perintah *drop table*.

e. Mengubah struktur tabel

Tabel yang sudah pernah dibuat kemungkinan akan diubah. Perubahan tersebut bisa berupa penambahan, penghapusan, atau penggantian struktur tabel. Untuk melakukan perubahan struktur tabel, gunakan perintah *alter*.

Fasilitas-fasilitas yang diberikan *alter* antara lain:

1. *ADD*

- a. menambahkan kolom baru;
- b. menambahkan *index* atau *key* kolom;
- c. menambahkan *constraint*;
- d. menambahkan *primary key* dan lain-lain.

2. *ALTER*

Semua perubahan struktur tabel harus diawali dengan perintah *alter*.

3. *CHANGE*

- a. digunakan untuk mengubah urutan posisi kolom pada tabel;
- b. mengubah nama kolom.

4. *DROP*

- a. untuk melakukan penghapusan kolom yang sudah ada;

- b. menghapus *primary key* kolom;
- c. menghapus *index* kolom.

4. *Input* data tabel

Setelah membuat database dan tabel, selanjutnya yaitu mengisi data di dalam tabel tersebut. Input data di dalam tabel bisa dilakukan dengan dua cara, yaitu:

- a. *Load* data dari file teks.
- b. Menggunakan perintah *query insert*.

5. *Update* data tabel

Ada dua cara yang dilakukan untuk mengubah isi data di dalam tabel, yaitu:

- a. Melakukan *load* data seperti yang dilakukan dalam pengisian data, tetapi sebelum di-*load* data di file harus diubah terlebih dahulu dan data yang ada di tabel harus dihapus.
- b. Dengan mengetikkan pernyataan *update*.

6. Pencarian data tabel

Untuk menampilkan data pada tabel yang sudah dibuat adalah dengan menggunakan perintah *select*.

Sintak umum: `select nama_field from nama_tabel where kondisi_diinginkan`

Untuk `nama_field` bisa terdiri atas semua kolom atau hanya sebagian dari kolom di dalam tabel, `nama_tabel` merupakan nama tabel yang ingin dimunculkan isinya, sedangkan pernyataan setelah `where` yaitu `kondisi_diinginkan` merupakan suatu kondisi, yang jika diisi dapat digunakan sebagai syarat pencarian.

3.8.6. *PHP MyAdmin*

Tool yang digunakan untuk mempermudah *administrasi server MySQL* pada penelitian kali ini adalah *PHPMyAdmin* yang

merupakan *front-end MySQL* berbasis *web* yang dibuat menggunakan *PHP*. *PHP MyAdmin* berfungsi untuk membuat, mengedit, menghapus database, menghapus tabel, serta membuat dan menghapus relasi antar tabel, mensortir data, dan lain-lain sesuai kebutuhan.

PHPMyAdmin ini memiliki beberapa fitur seperti manipulasi *database*, tabel, index, dan mengekspor data ke dalam berbagai format data dan juga tersedia dalam 50 bahasa lebih termasuk bahasa Indonesia. Dengan *PHPMyAdmin*, pengolahan atau manipulasi *database* jadi lebih mudah, karena tidak perlu menghafal sintaknya.

Kelebihan *PHP MyAdmin*, yaitu sebagai berikut:

1. Mempunyai *interface* (antarmuka) yang bersifat *user friendly*.
2. Mempunyai berbagai fitur yang dapat digunakan untuk menambah, menghapus, menyalin, dan juga memperbaharui *database*, *view*, tabel, *index*, *fields*. Kemudian melakukan pemeliharaan *database* melalui menu konfigurasi, melakukan eksekusi terhadap *query MySQL*, serta mengelola akun pengguna *MySQL*.
3. Melakukan *import* data yang berformat *MySQL* atau *SVG*.
4. Melakukan *ekspor* data ke berbagai format lain seperti *XML*, *CSV*, *PDF*, dan lain sebagainya.
5. Mengelola banyak *server*.
6. Dapat membuat perintah *query* dengan fitur *QBE (Query by Example)*.
7. Mampu melakukan pencarian global pada *database*.
8. Dapat mengubah data dalam format apapun.

3.8.7. XAMPP

Xampp adalah pengembangan dari *LAMP (Linux Apache, MySQL, PHP and PERL)* yang dikembangkan oleh *Apache Friend* yang didirikan Kai 'Oswalad' Seidler dan Kay Vogelgesang pada

tahun 2002, proyek mereka bertujuan untuk mempromosikan penggunaan *Apache Web server*.

Xampp merupakan perangkat lunak berbasis *web* yang bersifat *open source* (bebas) dan juga mendukung di berbagai sistem operasi seperti *OX Linux*, *OS Windows*, *Mac OS*, dan juga *Soloris*.

Xampp adalah singkatan dari X (*cross platform*), A (*Apache*), M (*MySQL/Maria DB*), P (*PHP*), dan P (*Perl*).

1. X = *cross platform*

Kode penanda dari *software cross platform* yang berarti dapat dijalankan di berbagai sistem operasi, seperti *Windows*, *Linux*, *Mac OS*, dan *Soloris*.

2. A = *Apache*

Aplikasi *web server* gratis dan bisa dikembangkan oleh banyak orang (*open source*) untuk membuat halaman *website* yang benar berdasarkan kode program *PHP* yang ditulis oleh pengembang *web developer*.

3. M = *MySQL/MariaDB*

Aplikasi *database server* yang menerapkan bahasa pemrograman *SQL (Structured Query Language)* yang digunakan untuk mengelola dan membuat sistem *database* yang terstruktur dan sistematis, seperti mengolah, mengedit, dan menghapus data melalui *database*.

4. P = *PHP*

Bahasa pemrograman khusus berbasis *web* untuk keperluan pada sisi *server (back end)*, sehingga bisa digunakan untuk membuat halaman *website* menjadi lebih dinamis dengan menerapkan *server-side scripting*.

5. P = *Perl*

Bahasa pemrograman yang memenuhi berbagai kebutuhan (*cross platform*) yang bisa berjalan di banyak sistem operasi sehingga sangat fleksibel. Biasanya digunakan untuk membuat *website* dinamis seperti *CMS (Content Management System) WordPress*.

Komponen penting dalam *Xampp*, yaitu:

a. *Control panel*

Layanan yang digunakan untuk memberikan akses yang lebih leluasa untuk mengatur *database*, mengunggah *file*, mengelola fitur dan juga melakukan konfigurasi terkait *website* yang sedang dikerjakan.

b. *Htdocs*

Htdocs adalah nama sebuah folder bagian dari *Xampp* yang berfungsi untuk menyimpan berbagai *file* dan dokumen yang akan ditampilkan pada *website*. Kapasitas penyimpanan di dalam *htdocs* bergantung pada *storage hardisk* yang ada di komputer.

c. *PHPMyAdmin*

Program untuk mengelola *database website* yang sedang dikembangkan.

d. *Config*

Menu yang digunakan untuk melakukan konfigurasi dasar *Xampp*.

e. *Netstat*

Menu yang digunakan untuk memastikan apakah ada aplikasi lain yang menggunakan *port xampp* pada perangkat yang digunakan.

Xampp merupakan sebuah paket instalasi untuk *PHP,APACHE* dan *MySQL*. Dengan begitu tidak perlu lagi untuk menginstal tiga *software* tersebut.

1. *Instal XAMPP*

a. *Download xampp* melalui *website apache*



Gambar 3.10 Tahap Pertama *Instal Xampp*

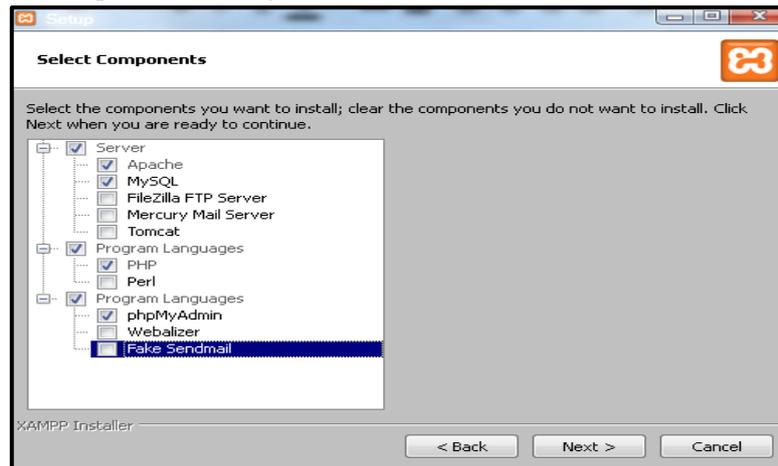
- b. Setelah proses unduh berhasil maka langkah selanjutnya *instal software*. Jika ada pesan apakah anda yakin akan menginstalnya, maka pilih *YES*.
- c. Klik tombol *next*



Gambar 3.11 Tahap Kedua *Instal Xampp*

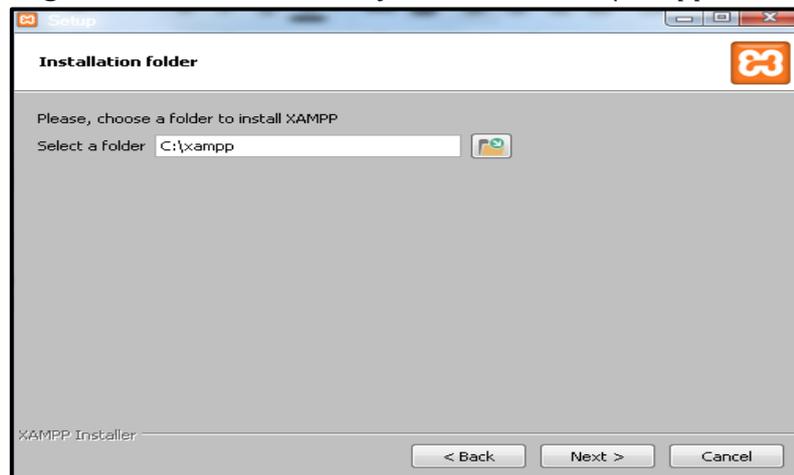
- d. Pada tampilan berikutnya akan muncul pilihan mengenai komponen mana dari *XAMPP* yang ingin dan tidak ingin Anda *instal*. Beberapa pilihan seperti *Apache* dan *PHP* adalah bagian penting untuk menjalankan *website* dan akan

otomatis diinstal. Silahkan centang *MySQL* dan *phpMyAdmin*, untuk pilihan lainnya biarkan saja.



Gambar 3.12 Tahap Ketiga *Instal Xampp*

- e. Selanjutnya silahkan pilih folder tujuan dimana *XAMPP* ingin Anda instal. Contohnya di direktori *C:\xampp*.



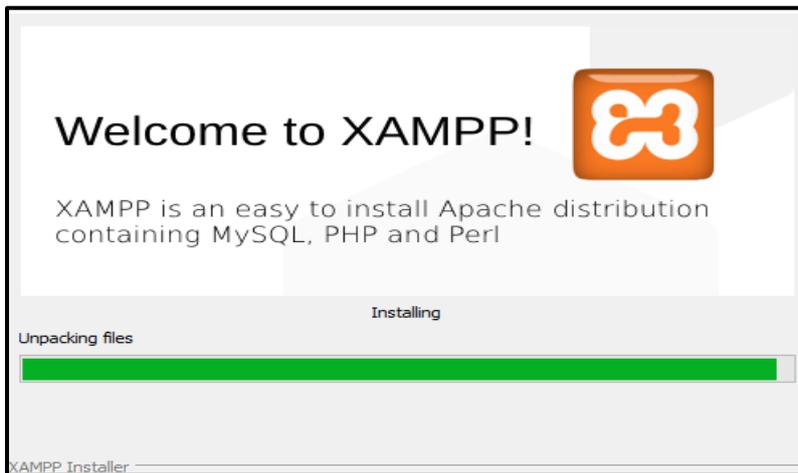
Gambar 3.13 Tahap Keempat *Instal Xampp*

- f. Pada halaman selanjutnya, akan ada pilihan apakah Anda ingin menginstal Bitnami untuk *XAMPP*, dimana nantinya dapat Anda gunakan untuk *install WordPress, Drupal, dan Joomla* secara otomatis.



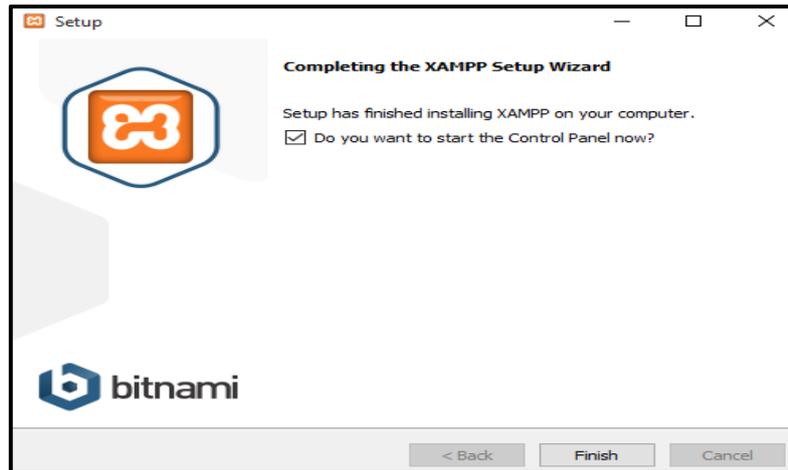
Gambar 3.14 Tahap Kelima *Instal Xampp*

- g. Pada langkah ini proses instalasi *XAMPP* akan dimulai. Silahkan klik tombol *Next*.



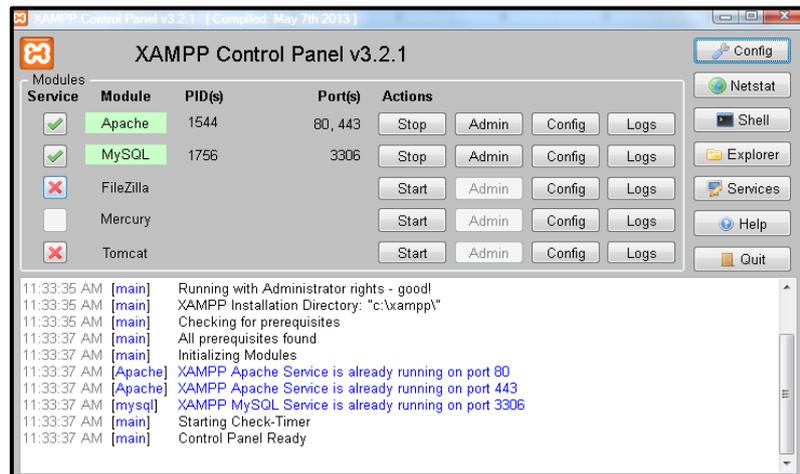
Gambar 3.15 Tahap Keenam *Instal Xampp*

- h. Setelah berhasil diinstal, akan muncul notifikasi untuk langsung menjalankan *control panel*. Silakan klik *Finish*.



Gambar 3.16 Tahap Ketujuh *Instal Xampp*

- i. Silakan buka aplikasi *XAMPP* kemudian klik tombol *Start* pada *Apache* dan *MySQL*. Jika berhasil dijalankan, *Apache* dan *MySQL* akan berwarna hijau seperti gambar di bawah ini.



Gambar 3.17 Tahap Kedelapan *Instal Xampp*

3.8.8. Text Editor

Text editor merupakan sebuah program yang digunakan untuk mengedit serta mengkreasi beberapa text file, lebih tepatnya file yang berisi bahasa *programming* atau tempat

dimana menulis kode yang digunakan untuk *coding*. Dengan demikian, text editor bisa menangani "*hand coding*" dengan berbagai macam bahasa, seperti; *HTML, CSS, JavaScript, PHP, Ruby, Python*, dan yang lain-lainnya.

Pada perancangan sistem informasi rehabilitasi dan rekonstruksi pascabencana berbasis *web* menggunakan teks editor yaitu *sublime text*. *Sublime Text* merupakan salah satu *text editor* yang paling terkenal karena memiliki fitur *syntax highlighting, colour scheme* yang menarik, kemampuan untuk menambatkan beberapa *package*, bisa bekerja di *Mac, Windows*, dan juga mesin *Linux*, dan masih banyak yang lainnya.

Sublime Text adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai *platform operating system* dengan menggunakan teknologi *PhytonAPI*.

Sublime Text mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur *syntax highlight* hampir di semua bahasa pemrograman yang didukung ataupun dikembangkan oleh komunitas seperti; *C, C++, C#, CSS, D, Dylan, Erlang, HTML, Groovy, Haskell, Java, JavaScript, LaTeX, Lisp, Lua, Markdown, MATLAB, OCaml, Perl, PHP, Python, R, Ruby, SQL, TCL, Textile* dan *XML*.

Biasanya bagi bahasa pemrograman yang didukung atau pun belum didukung secara *default* dapat lebih dimaksimalkan atau didukung dengan menggunakan *add-ons* yang bisa didownload sesuai kebutuhan *user*.

Salah satu kelebihan *Sublime Text* adalah *text editor* ini memiliki *low resource usage* dimana dia sangat *lightweight* karena tidak mengambil banyak data, namun tetap bisa menawarkan fitur-fitur yang maju. Sehingga, menjadi salah satu *text editor* terbaik. Selain itu, keuntungan paling utama *Sublime text* adalah ia menawarkan *shortcut* dan *search tool* agar bisa langsung mencari fungsi-fungsi tertentu dan melakukan perubahan pun lebih gampang. Kemudian, *Sublime Text*

memiliki *API Python*, alhasil beberapa plugin bisa diintegrasikan dengan solusi *text editing*.

Fitur yang diunggulkan dari aplikasi *Sublime Text*, yaitu sebagai berikut:

1. *Goto Anything*

Fitur yang sangat membantu dalam membuka *file* ataupun menjelajahi isi dari *file* hanya dengan beberapa *keystrokes*.

2. *Multiple Selections*

Fitur ini memungkinkan *user* untuk mengubah secara interaktif banyak baris sekaligus, mengubah nama variabel dengan mudah, dan memanipulasi *file* lebih cepat dari sebelumnya.

3. *Command Pallete*

Dengan hanya beberapa *keystrokes*, *user* dapat dengan cepat mencari fungsi yang diinginkan, tanpa harus menavigasi melalui menu.

4. *Distraction Free Mode*

Bila *user* memerlukan fokus penuh pada aplikasi ini, fitur ini dapat membantu *user* dengan memberikan tampilan layar penuh.

5. *Split Editing*

Dapatkan hasil yang maksimal dari monitor layar lebar dengan dukungan editing perpecahan. Mengedit sisi *file* dengan sisi, atau mengedit dua lokasi di satu *file*. Anda dapat mengedit dengan banyak baris dan kolom yang *user* inginkan.

6. *Instant Project Switch*

Menangkap semua *file* yang dimasukkan ke dalam *project* pada aplikasi ini. Terintegrasi dengan fitur *Goto Anything* untuk menjelajahi semua *file* yang ada ataupun untuk beralih ke *file* dalam *project* lainnya dengan cepat.

7. *Plugin API*

Dilengkapi dengan *plugin API* berbasis *Python* sehingga membuat aplikasi ini sangat tangguh.

8. *Customize Anything*

Aplikasi ini memberikan *user* fleksibilitas dalam hal pengaturan fungsional dalam aplikasi ini.

9. *Cross Platform*

Aplikasi ini dapat berjalan hampir disemua *operating system modern* seperti Windows, OS X, dan *Linux based operating system*.

BAB IV

ANALISA DAN PEMBAHASAN

4.1. Analisis Sistem

Analisis sistem adalah teknik penyelesaian masalah dengan menguraikan bagian-bagian komponen dengan maksud mendefinisikan masalah dan mengevaluasi permasalahan-permasalahan, hambatan-hambatan yang terjadi dan juga kebutuhan-kebutuhan yang diharapkan agar dapat diusulkan perbaikan-perbaikan yang bertujuan untuk membuat sistem tersebut bisa saling berinteraksi satu sama lain. Analisis sistem adalah langkah awal yang menjadi fondasi dalam pengembangan sistem untuk sistem informasi yang dihasilkan nantinya.

Setelah melaksanakan penelitian terhadap sistem yang sudah ada, maka kita dapat menganalisa dan mengevaluasi sehingga dapat dijadikan perbandingan terhadap sistem yang akan dirancang. Dengan adanya sistem baru diharapkan dapat menyelesaikan permasalahan yang terjadi dan dapat meningkatkan fungsi yang lebih baik lagi dari sistem yang telah ada.

4.1.1. Analisa Sistem Lama

Dari hasil penelitian Penelitian yang dilaksanakan selama dua bulan di Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara penulis menganalisa sistem yang dipakai untuk pendataan di bagian rehabilitasi dan rekonstruksi pascabencana masih menggunakan pencatatan manual yang memiliki risiko yang lebih besar seperti kehilangan informasi pascabencana dan juga tidak efisien karena menggunakan waktu yang lebih lama.

4.1.2. Analisis Sistem Baru

Perancangan sistem yang ingin diterapkan di sana nantinya akan menggunakan sistem yang dibangun dari bahasa pemrograman *PHP* dan juga menggunakan *database MySQL*. Dari sistem tersebut diharapkan dapat mempermudah para karyawan di bagian rehabilitasi dan rekonstruksi pascabencana, sehingga data yang diolah lebih akurat, dapat digunakan dimanapun dan lebih efisien.

4.2. Analisis Kebutuhan Non-Fungsional

Kebutuhan non fungsional dapat membantu mempermudah proses pengolahan data rehabilitasi dan rekonstruksi pascabencana pada sistem. Dari deskripsi kebutuhan non fungsional dapat diketahui hal yang diperlukan untuk membangun sebuah sistem. Adapun kebutuhan dijabarkan sebagai berikut:

4.2.1. Analisis Perangkat Keras (*Hardware*)

Perangkat keras adalah seluruh komponen atau unsur peralatan yang digunakan untuk menunjang pembuatan sebuah sistem informasi. Adapun minimum perangkat keras yang untuk menjalankan aplikasi ini adalah sebagai berikut:

1. *RAM* 2 GB
2. *Hardisk* 128 GB untuk menyimpan data
3. *Monitor* 14 inch
4. *Keyboard*
5. *Mouse*
6. *Prosesor* menggunakan *Pentium 2*

Perangkat keras yang berada pada Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara telah memenuhi standar bahkan lebih untuk menjalankan aplikasi yang di bangun.

4.2.2. Analisis Perangkat Lunak (*Software*)

Perangkat lunak atau *software* adalah komponen penting dalam setiap program yang akan dijalankan. Untuk itu diperlukan beberapa spesifikasi perangkat lunak yang dibutuhkan untuk menjalankan sistem informasi ini. Adapun spesifikasi perangkat lunak yaitu sebagai berikut:

1. Sistem Operasi Windows 7 Ultimate
2. *Database Mysql*
3. *Web Browser (Google Chrome Rekomendasi)*
4. *Xampp*

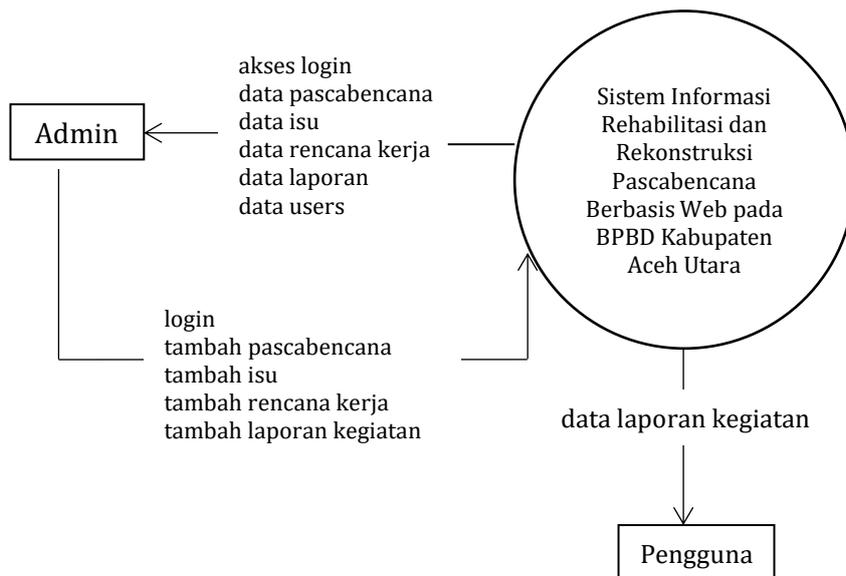
Perangkat lunak yang ada pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara sudah memenuhi standar untuk menjalankan aplikasi yang di bangun.

4.3. Perancangan Sistem

Tahap ini akan dilakukan proses perancangan suatu sistem informasi tentang data rehabilitasi dan rekonstruksi pascabencana sehingga menghasilkan informasi yang cepat, tepat, dan akurat. Perancangan dengan model sistem dengan menentukan rancangan input dan informasi apa saja yang dihasilkan nantinya.

4.3.1. *Diagram Konteks*

Diagram konteks ini menggambarkan secara garis besar seluruh proses yang terjadi pada sistem yang telah dirancang. Disini akan diperlihatkan gambaran secara umum proses dari sistem rehabilitasi dan rekonstruksi pascabencana. Adapun *diagram konteks* yang akan dirancang adalah sebagai berikut:



Gambar 4.1 *Diagram Konteks*

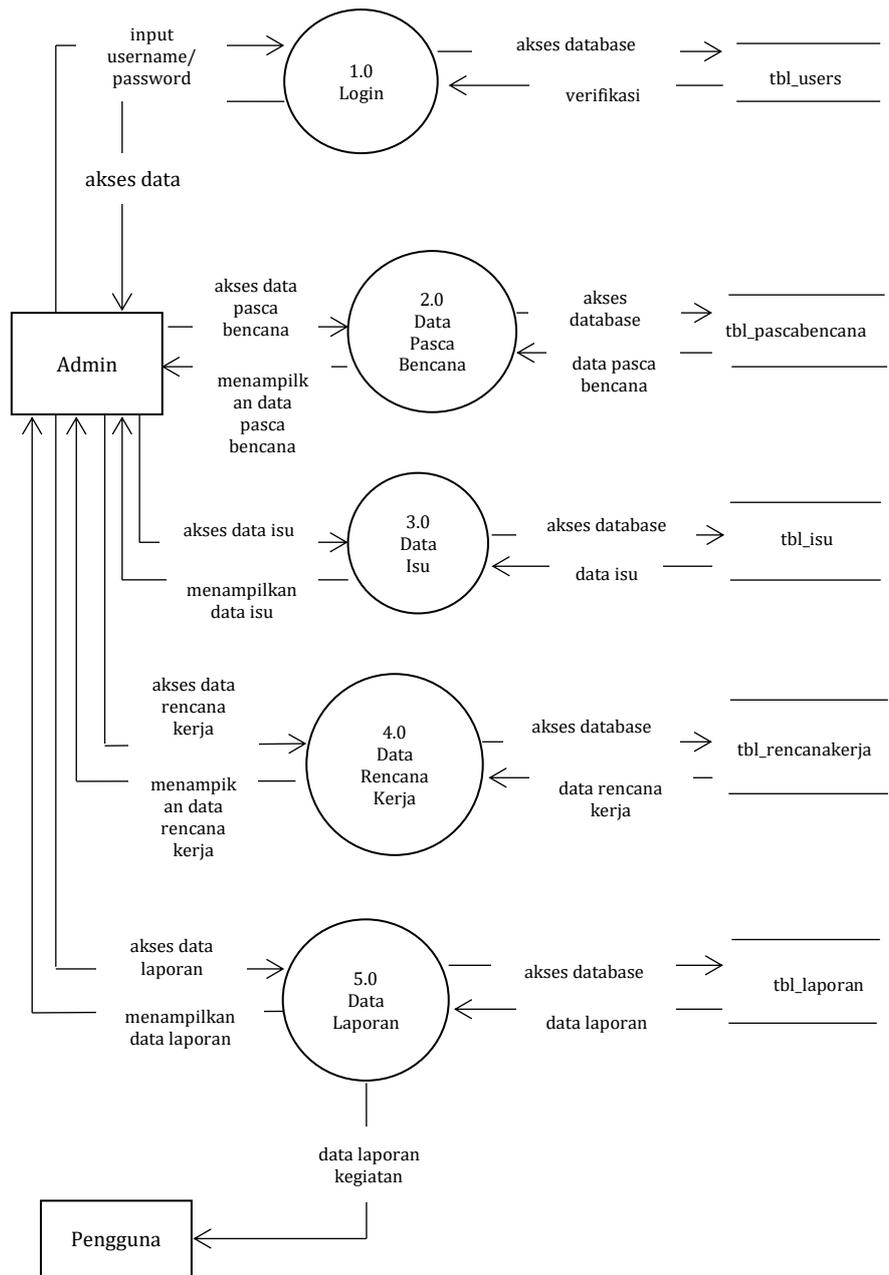
Keterangan:

Pada *diagram konteks* ini admin akan *login* dengan memasukkan *username* dan juga *password* kemudian akan masuk ke dalam sistem. Admin dapat melakukan penambahan data pascabencana, data isu/permasalahan, data rencana kerja, data laporan, dan data *users*. Untuk pengguna bisa melihat tampilan dari kegiatan untuk bidang rehabilitasi dan rekonstruksi.

4.3.2. *Data Flow Diagram*

1. *DFD Level 0*

Diagram level 1 merupakan diagram yang menggambarkan komponen-komponen yang mewakili lingkaran-lingkaran kecil yang terdapat pada diagram konteks. Adapun prosesnya yaitu:



Gambar 4.2 DFD Level 0

Keterangan:

1. Proses 1.0 adalah proses *login*

Admin akan melakukan *login* dengan memasukkan *username* dan *password* kemudian sistem akan melakukan verifikasi dengan memeriksa kecocokan *database* pada tabel *users* sehingga akan menampilkan sistem.

2. Proses 2.0 adalah proses data pascabencana

Admin akan melakukan akses data pascabencana menuju ke database lalu akan menghasilkan output untuk data pascabencana.

3. Proses 3.0 adalah proses data isu

Admin akan melakukan akses data isu/permasalahan menuju ke database dan akan menghasilkan output data isu/permasalahan yang terjadi.

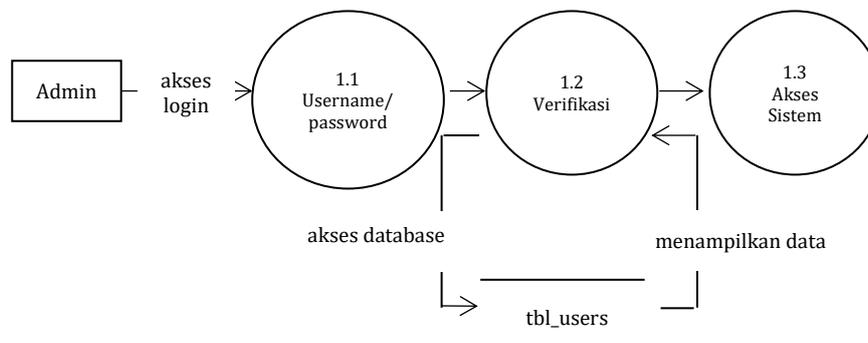
4. Proses 4.0 adalah proses data rencana kerja

Admin akan melakukan akses data rencana kerja menuju ke database dan akan menghasilkan output data rencana kerja yang dirancang yang berasal dari isu yang ada pada data isu/permasalahan yang terjadi.

5. Proses 4.0 adalah proses data laporan

Admin akan melakukan akses data laporan menuju ke database dan akan menghasilkan output data laporan kegiatan pascabencana yang dilakukan.

2. DFD Level 1 Proses Login

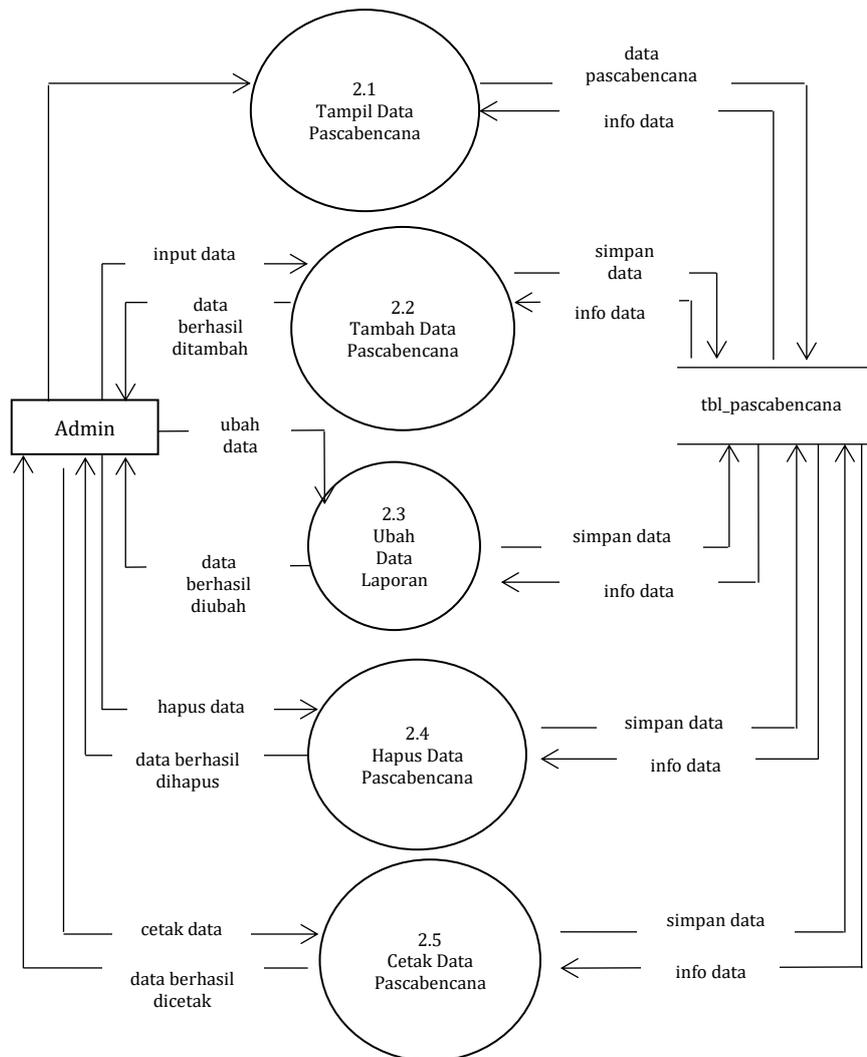


Gambar 4.3 DFD Level 1 Proses *Login*

Keterangan:

Pada diagram di atas akan menjelaskan proses *login*. Pada proses ini admin akan menginputkan *username* dan *password* pada *login*. Kemudian *login* akan diperiksa pada *database users*. Jika datanya valid maka proses *login* sukses, tapi jika datanya tidak valid maka proses *login* akan gagal.

3. DFD Level 1 Proses Data Pasca Bencana



Gambar 4.4 DFD Level 1 Proses Tambah Data Pasca Bencana

Keterangan:

1. Tampil data

Pada proses ini, seluruh data yang ada pada tabel data pascabencana akan ditampilkan, *database* tersebut berisi data-data pascabencana yang terjadi di Kabupaten Aceh Utara.

2. Tambah data

Proses ini merupakan proses dimana admin akan menambahkan data-data yang terjadi pascabencana dan data tersebut akan disimpan pada *database*.

3. Ubah data

Proses ini merupakan proses dimana jika pada data pascabencana ada hal-hal yang perlu diubah maka admin akan mengubahnya disini dan jika berhasil maka data yang sebelumnya akan berubah sesuai dengan apa yang diubah.

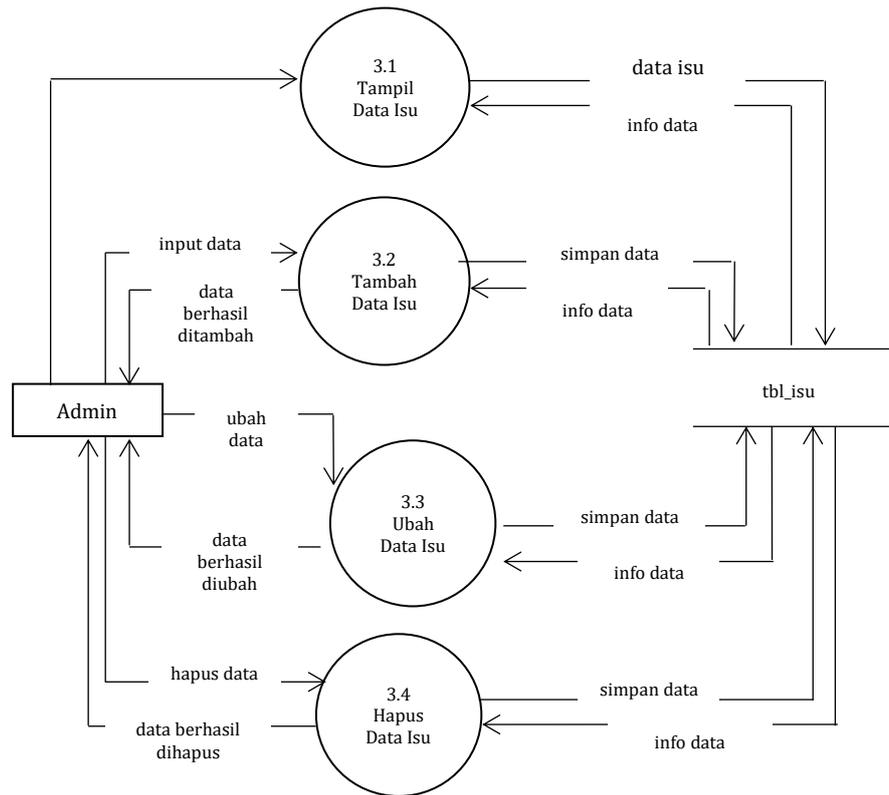
4. Hapus data

Proses ini merupakan proses penghapusan data yang terdapat pada tabel pascabencana.

5. Cetak data

Proses ini merupakan proses untuk mencetak data pascabencana.

4. DFD Level 1 Proses Data Isu



Gambar 4.5 DFD Level 1 Proses Tambah Data Isu

Keterangan:

1. Tampil data

Pada proses ini, seluruh data yang ada pada tabel data isu akan ditampilkan, *database* tersebut berisi data-data isu/permasalahan yang terjadi di Kabupaten Aceh Utara.

2. Tambah data

Proses ini merupakan proses dimana admin akan menambahkan data-data mengenai isu/permasalahan di bidang rehabilitasi dan rekonstruksi, kemudian data tersebut akan disimpan pada *database*.

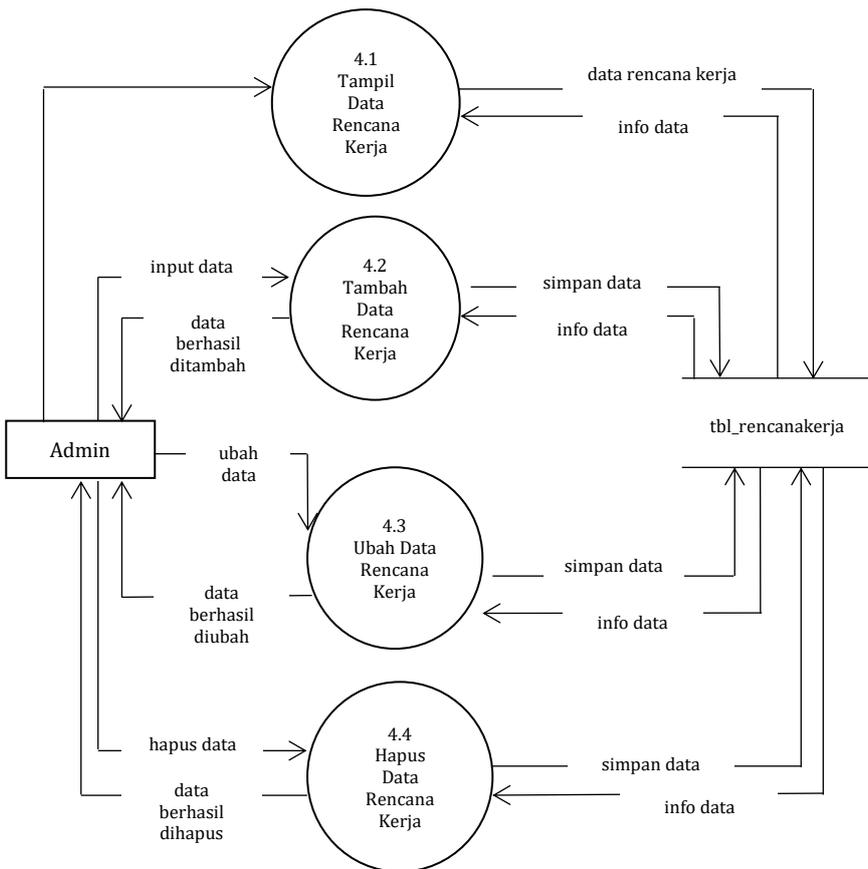
3. Ubah data

Proses ini merupakan proses dimana jika pada data isu ada hal-hal yang perlu diubah maka admin akan mengubahnya disini dan jika berhasil maka data yang sebelumnya akan berubah sesuai dengan apa yang diubah.

4. Hapus data

Proses ini merupakan proses penghapusan data yang terdapat pada tabel isu.

5. DFD Level 1 Proses Data Rencana Kerja



Gambar 4.6 DFD Level 1 Proses Tambah Data Rencana Kerja

Keterangan:

1. Tampil data

Pada proses ini, seluruh data yang ada pada tabel data rencana kerja akan ditampilkan, *database* tersebut berisi perencanaan yang akan dilakukan di bidang rehabilitasi dan rekonstruksi yang terjadi di Kabupaten Aceh Utara.

2. Tambah data

Proses ini merupakan proses dimana admin akan menambahkan data-data rencana kerja dan data tersebut akan disimpan pada *database*.

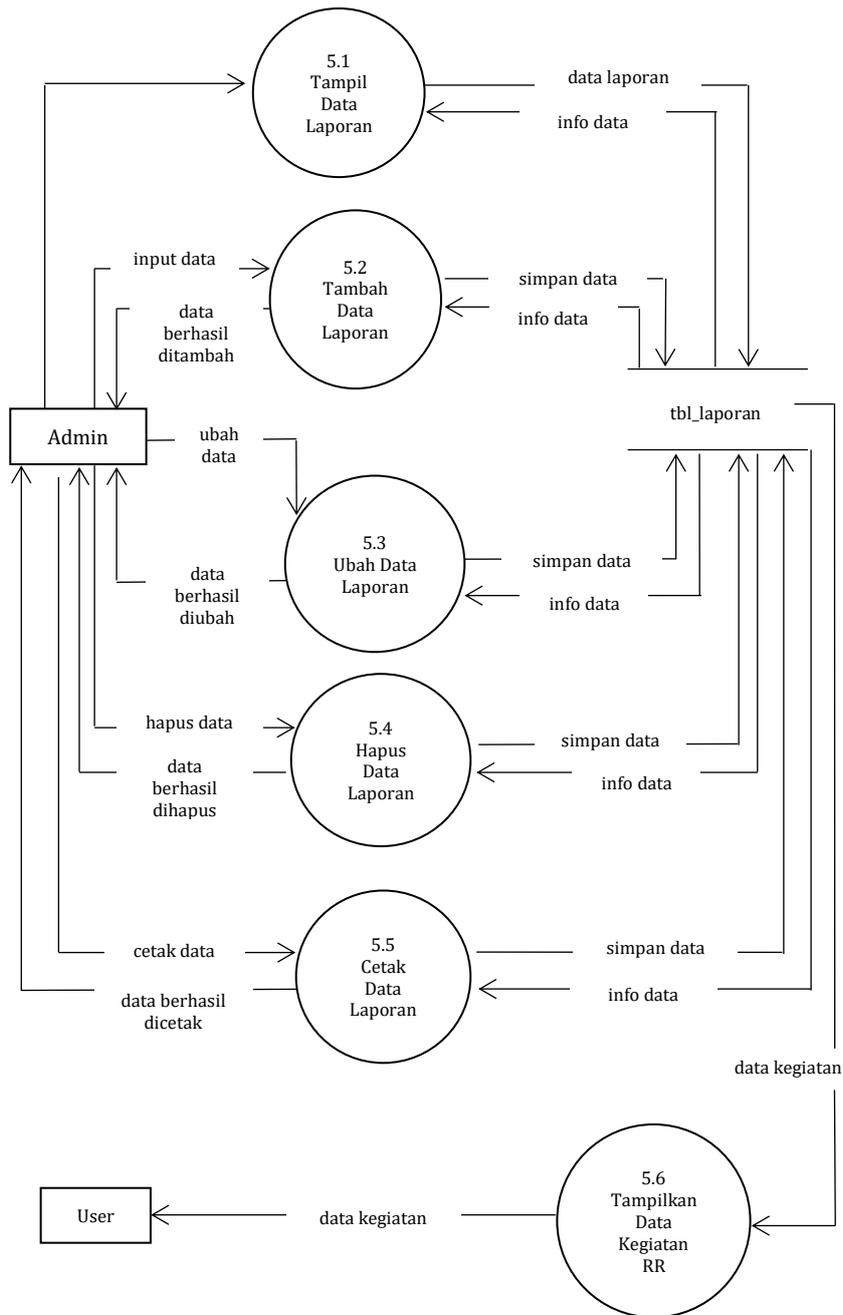
3. Ubah data

Proses ini merupakan proses dimana jika pada data rencana kerja ada hal-hal yang perlu diubah maka admin akan mengubahnya disini.

4. Hapus data

Proses ini merupakan proses penghapusan data yang terdapat pada tabel rencana kerja.

6. DFD Level 1 Proses Data Laporan



Gambar 4.7 DFD Level 1 Proses Tambah Data Laporan

Keterangan:

1. Tampil data

Pada proses ini, seluruh data yang ada pada tabel data laporan akan ditampilkan, *database* tersebut berisi data-data laporan yang terjadi di Kabupaten Aceh Utara.

2. Tambah data

Proses ini merupakan proses dimana admin akan menambahkan data laporan dan data tersebut akan disimpan pada *database*.

3. Ubah data

Proses ini merupakan proses dimana jika pada data laporan ada hal-hal yang perlu diubah maka admin akan mengubahnya disini dan jika berhasil maka data yang sebelumnya akan berubah sesuai dengan apa yang diubah.

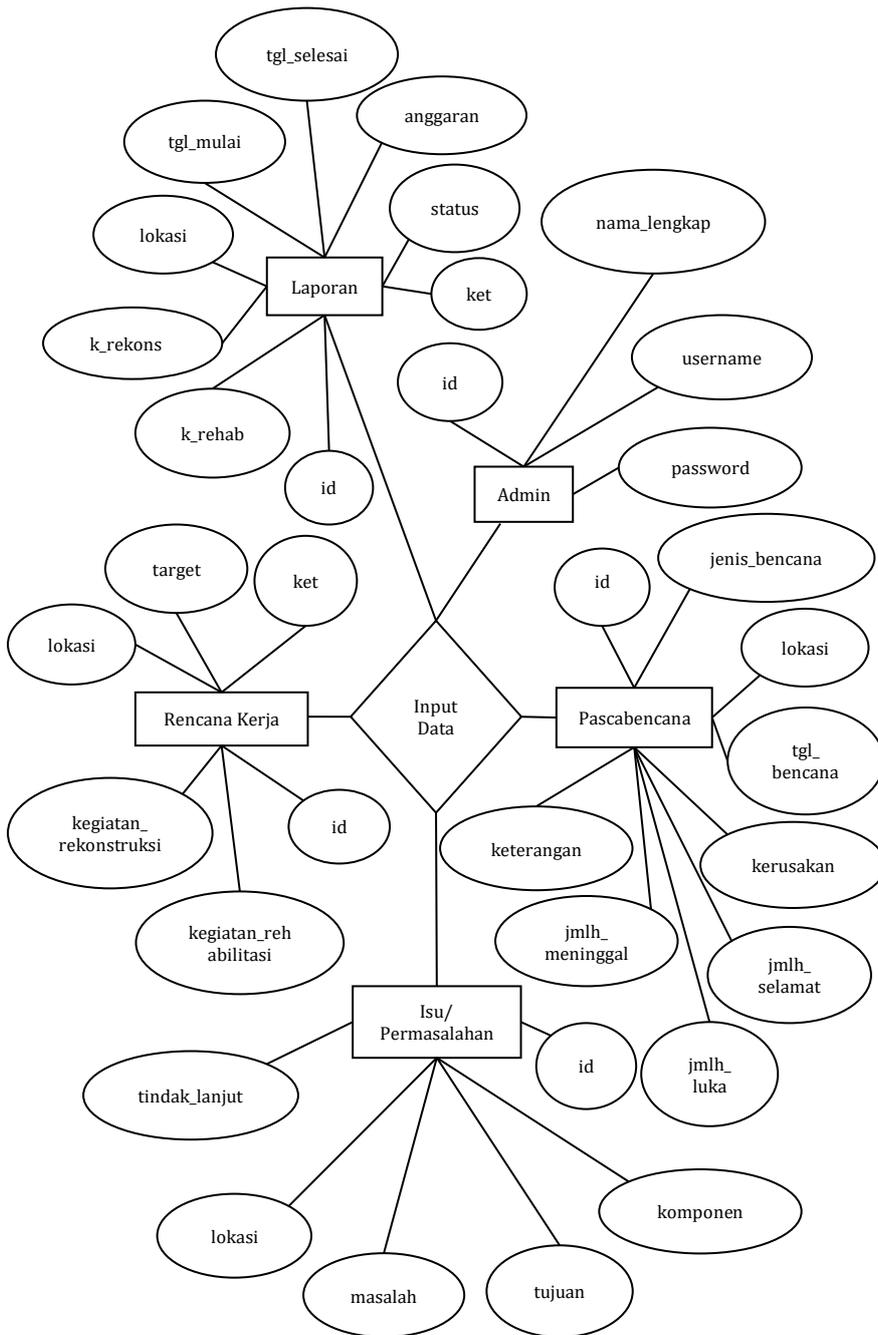
4. Hapus data

Proses ini merupakan proses penghapusan data yang terdapat pada tabel laporan.

5. Cetak data

Proses ini merupakan proses untuk mencetak data laporan.

4.3.3. Entity Relationship Diagram (ERD)



Gambar 4.8 Entity Relation Diagram

4.4. Perancangan *Database*

4.4.1. Tabel Admin

Tabel 4.1 Admin

No	Field Name	Data Type	Size	Keterangan
1	id	Int	11	Primary key/nomor
2	nama_lengkap	Varchar	100	Nama lengkap admin
3	username	Varchar	30	Username admin
4	password	Varchar	50	Password admin

4.4.2. Tabel Data Pascabencana

Tabel 4.2 Data Pascabencana

No	Field Name	Data Type	Size	Keterangan
1	id	Int	11	Primary key/nomor
2	jenis_bencana	Varchar	200	Jenis bencana yang terjadi
3	lokasi	Varchar	200	Lokasi bencana
4	tgl_bencana	Date		Tanggal terjadinya bencana
5	kerusakan	Varchar	300	Kerusakan pada saat bencana
6	jmlh_selamat	Int	11	Jumlah korban yang selamat
7	jmlh_luka	Int	11	Jumlah korban yang luka
8	jmlh_meninggal	Int	11	Jumlah korban meninggal
9	keterangan	Text		Keterangan

4.4.3. Tabel Data Permasalahan

Tabel 4.3 Data Isu

No	Field Name	Data Type	Size	Keterangan
1	id	Int	11	Primary Key/nomor
2	komponen	Varchar	255	Komponen
3	tujuan	Varchar	300	Tujuan
4	masalah	Text		Masalah
5	lokasi	Varchar	200	Lokasi bencana
6	tindak_lanjut	Varchar	300	Tindak lanjut

4.4.4. Tabel Rencana Kerja

Tabel 4.4 Data Rencana Kerja

No	Field Name	Data Type	Size	Keterangan
1	Id	Int	11	Primery key/nomor
2	kegiatan_rehabilitasi	Varchar	100	Kegiatan rehabilitasi
3	kegiatan_rekonstruksi	Varchar	100	Kegiatan rekonstruksi
4	lokasi	Varchar	100	Lokasi kegiatan
5	target	Varchar	100	Target selesai
6	ket	Text		Keterangan

4.4.5. Tabel Laporan Kegiatan

Tabel 4.5 Laporan Kegiatan

No	Field Name	Data Type	Size	Keterangan
1	Id	Int	11	Primery key/nomor
2	k_rehab	Varchar	255	Kegiatan rehabilitasi
3	k_rekons	Varchar	255	Kegiatan rekonstruksi
4	Lokasi	Varchar	100	Lokasi kegiatan
5	tgl_mulai	Date		Tanggal mulai kegiatan
6	tgl_selesai	Date		Tanggal selesai kegiatan
7	Anggaran	Int	11	Anggaran untuk kegiatan
8	Status	Varchar	200	Status untuk kegiatan yang dilaksanakan
9	Ket	Text		Keterangan kegiatan

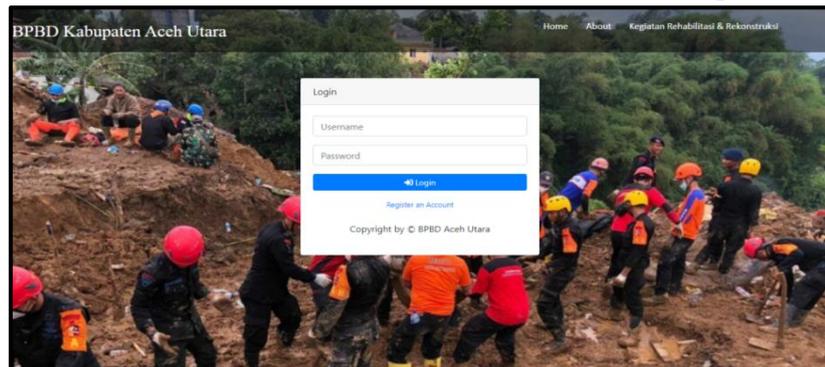
4.5. Implementasi Sistem

Implementasi sistem adalah tahapan yang menggambarkan hasil dari perancangan yang sudah dikerjakan sebelumnya, dimana implementasi sistem akan menggunakan *HTML*, *PHP*, dan *database MySQL* untuk mengembangkan aplikasi berbasis *web*.

1. Halaman *Login*

Halaman ini adalah *form* untuk *login* admin. Disini admin akan mengisi *username* dan *password* yang sesuai dengan apa yang ada pada *database* agar bisa masuk ke dalam sistem dan

menggunakan sistem. Ada menu about yaitu berisi tentang BPBD dan juga menu kegiatan yaitu kegiatan bagian rehabilitasi dan rekonstruksi. Kedua menu ini bisa dilihat oleh siapa saja.



Gambar 4.9 Halaman *Login*

a. Halaman Kegiatan Rehabilitasi dan Rekonstruksi

Halaman ini berisi tentang kegiatan yang dilaksanakan oleh bidang rehabilitasi dan rekonstruksi. Halaman ini bisa diakses tanpa login tetapi di halaman ini pengguna tidak memiliki akses untuk memanipulasi data.

Data Kegiatan Rehabilitasi & Rekonstruksi

Show 10 entries Search:

No	Kegiatan Rehabilitasi	Kegiatan Rekonstruksi	Lokasi	Tanggal Mulai	Tanggal Selesai	Anggaran	Status	Keterangan
1	-	Rekonstruksi jembatan	Gampong Beuracat Kec. Pirak Tim	2020-03-19	2020-08-28	800000000	Selesai	-
2	Rehabilitasi tanggul	-	Gampong Beuracat Kec. Pirak Tim	2020-07-23	2020-08-15	900000000	Selesai	-

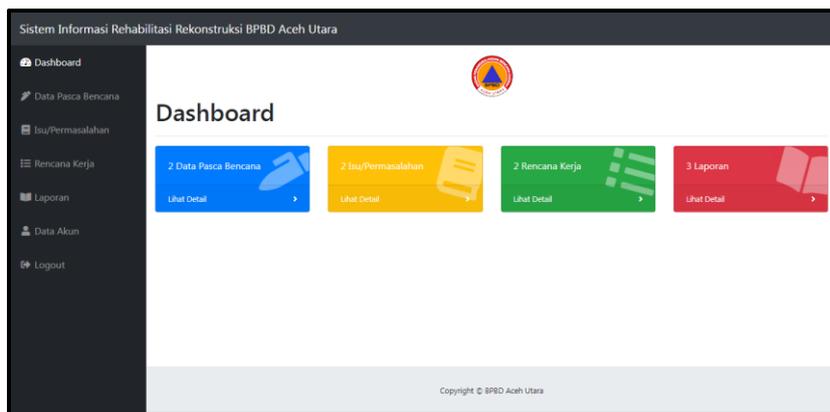
Showing 1 to 2 of 2 entries Previous 1 Next

Copyright © BPBD Aceh Utara

Gambar 4.10 Halaman Kegiatan

2. Halaman *Dashboard*

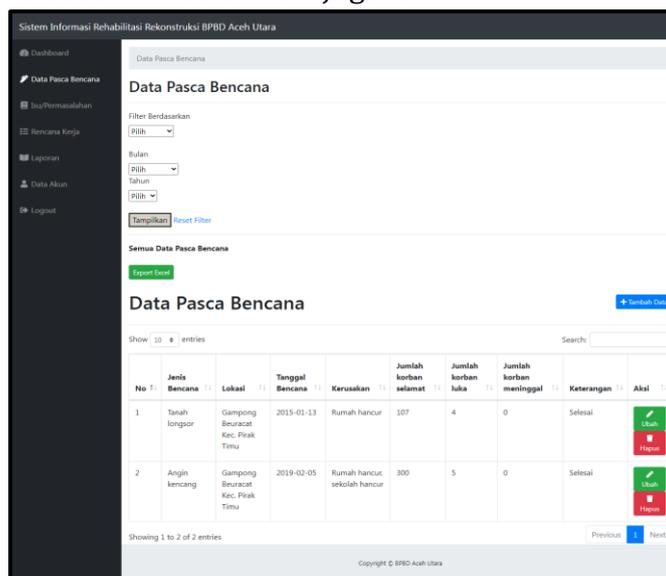
Halaman ini merupakan tampilan setelah admin berhasil masuk melalui halaman *login*. Ini merupakan halaman *dashboard* dimana disini terdapat menu data pascabencana, isu/permasalahan, rencana kerja, laporan, data akun dan juga menu untuk *logout*.



Gambar 4.11 Halaman *Dashboard*

3. Halaman Data Pascabencana

Halaman data pascabencana berisi tentang data-data yang terjadi pascabencana, seperti jenis bencana, lokasi, tanggal bencana, kerusakan, jumlah korban selamat, jumlah korban luka, jumlah korban meninggal, dan juga keterangan. Pada menu ini admin dapat menambah, ubah, dan juga hapus data. Pada halaman ini juga terdapat cetak data ke dalam excel bisa dipilih berdasarkan bulan dan juga tahun.



Gambar 4.12 Halaman Data Pascabencana

a. Tambah Data Pasca Bencana

Halaman ini merupakan halaman yang digunakan admin untuk menambah data pascabencana. Data yang bisa ditambah berupa jenis bencana, lokasi, tanggal bencana, kerusakan, jumlah selamat, jumlah luka, jumlah meninggal, dan keterangan.

The screenshot shows a web interface for adding disaster data. The form contains the following fields:

- Jenis Bencana:
- Masukan jenis Bencana:
- Lokasi:
- Masukan Lokasi Bencana:
- Tanggal Bencana:
- Kerusakan:
- Masukan Kerusakan Bencana:
- Jumlah selamat:
- Masukan Jumlah Selamat:
- Jumlah Luka:
- Masukan Jumlah Luka:
- Jumlah Meninggal:
- Masukan Jumlah Meninggal:
- Keterangan:
- Masukan keterangan:

Buttons:

Gambar 4.13 Tambah Data Pascabencana

4. Halaman Data Isu/Permasalahan

Halaman ini adalah halaman data isu/permasalahan yang berisi komponen, tujuan, masalah, lokasi, dan juga tindak untuk permasalahan yang terjadi. Pada halaman ini admin dapat menambah, mengedit dan menghapus data.

The screenshot shows a table of issues/problems. The table has the following columns: No, Komponen, Tujuan, Masalah, Lokasi, Tindak Lanjut, and Aksi. There are two entries listed.

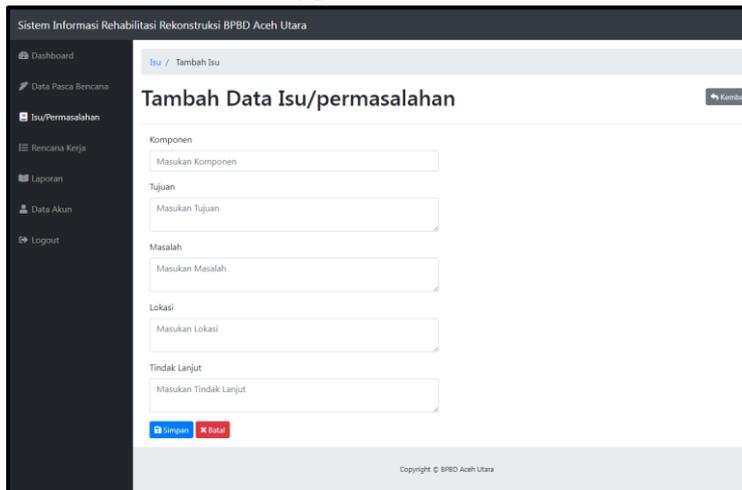
No	Komponen	Tujuan	Masalah	Lokasi	Tindak Lanjut	Aksi
1	Kerusakan tanggul	Ditujukan untuk mempercepat proses pembangunan tanggul	- Keterlambatan karena ketersediaan material - Kurangnya tenaga kerja	Gampong Beuracat Kec. Pirak Tim	- Berkoordinasi antar kontraktor untuk menambah tenaga kerja - Segera menambah tenaga kerja dari luar kota.	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>
2	Kerusakan rumah warga	Ditujukan agar rumah warga layak untuk ditempati	Keterlambatan pembayaran uang muka kepada kontraktor sehingga menghambat pembelian material	Gampong Alue Drien Kec. Lhoksukon	Berkoordinasi kepada pihak yang bertanggung jawab masalah keuangan agar prosesnya bisa cepat terselesaikan	<input type="button" value="Ubah"/> <input type="button" value="Hapus"/>

Showing 1 to 2 of 2 entries

Gambar 4.14 Halaman Data Isu/Permasalahan

a. Tambah Data Isu/Permasalahan

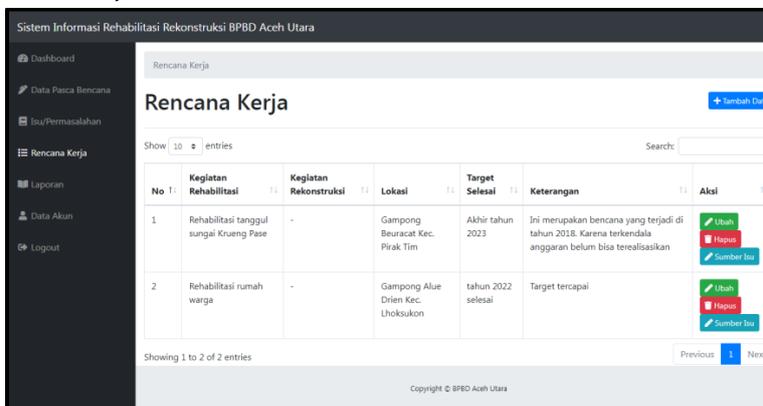
Halaman ini merupakan halaman yang digunakan admin untuk menambah data isu/permasalahan.



Gambar 4.15 Tambah Data Isu/Permasalahan

5. Halaman Rencana Kerja

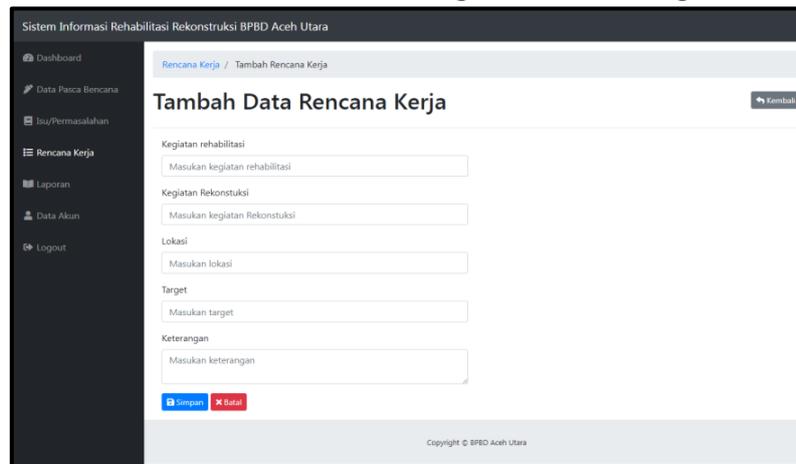
Halaman ini adalah halaman rencana kerja yang berisi kegiatan rehabilitasi, kegiatan rekonstruksi, lokasi bencana, target selesainya perencanaan tersebut, dan juga keterangan. Pada halaman ini admin dapat menambah, mengedit, menghapus data dan juga terdapat link untuk sumber data dari rencana kerja tersebut.



Gambar 4.16 Halaman Rencana Kerja

a. Tambah Data Rencana Kerja

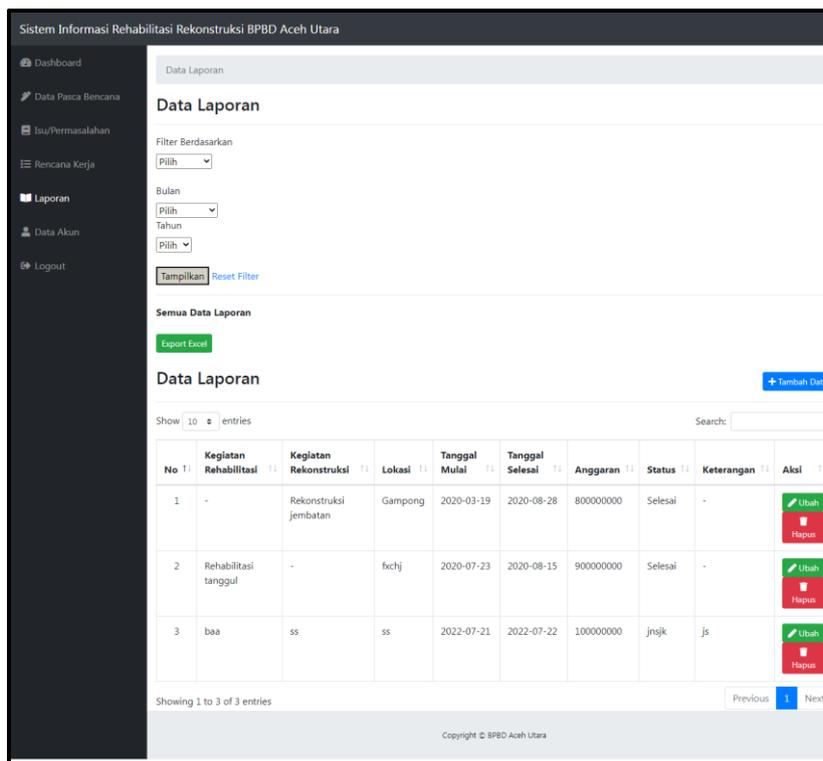
Halaman ini merupakan halaman yang digunakan admin untuk menambah data rencana kerja yang berisi kegiatan rehabilitasi, rekonstruksi, lokasi, target, dan keterangan.



Gambar 4.17 Tambah Data Rencana Kerja

6. Halaman Laporan Kegiatan

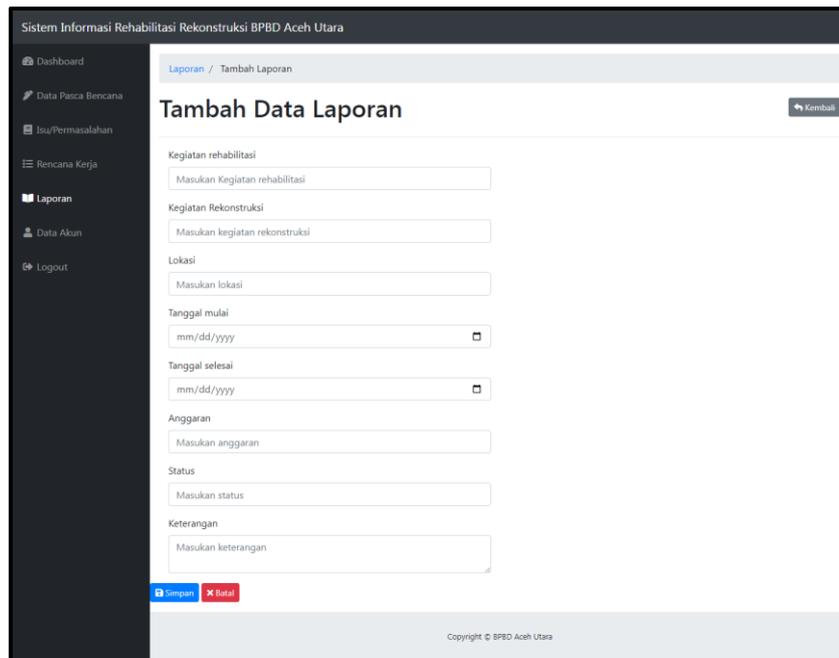
Halaman laporan kegiatan berisi tentang data-data kegiatan yang dilaksanakan oleh bagian rehabilitasi dan rekonstruksi, seperti kegiatan rehabilitasi, kegiatan rekonstruksi, lokasi, tanggal mulai, tanggal selesai, anggaran, status, dan keterangan. Pada halaman ini terdapat fitur edit untuk mengubah data, fitur hapus untuk menghapus data. Pada halaman laporan kegiatan juga terdapat cetak data ke dalam excel bisa dipilih berdasarkan bulan dan juga tahun.



Gambar 4.18 Halaman Data Laporan

a. Tambah Data Laporan

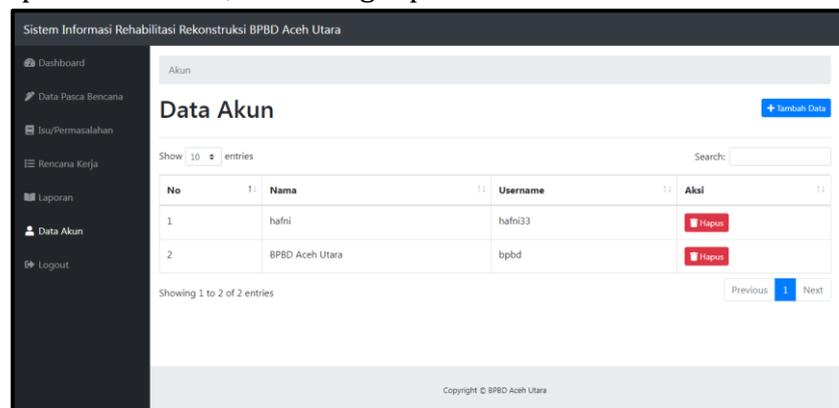
Halaman tambah data laporan adalah halaman yang digunakan untuk menambah data laporan kegiatan yang dilaksanakan oleh BPBD Kabupaten Aceh Utara.



Gambar 4.19 Tambah Data Laporan

7. Halaman Akun

Halaman ini adalah halaman data akun yang berisi nama, dan *username* yang sudah terdaftar. Pada halaman ini admin dapat menambah, dan menghapus data.



Gambar 4.20 Halaman Akun

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil pengamatan dalam penelitian yang telah dilakukan pada Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara dapat disimpulkan bahwa:

1. Dalam merancang dan membuat sistem informasi rehabilitasi dan rekonstruksi pascabencana ada beberapa tahap yang dilakukan, yaitu dengan memahami sistem yang sedang berjalan, merumuskan masalah yang ada, selanjutnya merancang serta membuat suatu sistem informasi yang mampu mengatasi masalah yang sedang berlangsung.
2. Sistem ini dikembangkan dengan melakukan perancangan sistem menggunakan *diagram konteks*, *DFD*, dan *ERD* yang membentuk suatu sistem yang dapat memberikan *platform* pendataan informasi rehabilitasi dan rekonstruksi pascabencana, yaitu data pascabencana, data isu/permasalahan, data rencana kerja dan juga laporan kegiatan rehabilitasi dan rekonstruksi.

5.2. Saran

Hasil penelitian ini masih membutuhkan penyempurnaan dan perbaikan serta beberapa saran penulis sampaikan:

1. Diharapkan akan dikembangkan menjadi lebih baik lagi untuk pendataan pascabencana seperti penambahan fitur sesuai dengan kebutuhan instansi.
2. Diharapkan akan memberikan manfaat sebagai referensi tambahan sejenis di masa mendatang.

DAFTAR PUSTAKA

- Andri Kristanto, 2003. *Perancangan Sistem Informasi dan Aplikasinya*. Yogyakarta: Penerbit Gava Media.
- Arifin, S. 2008. *Metode Kebijakan Mitigasi Bencana Alam bagi Difabel (studi kasus di Kabupaten Bantul, Yogyakarta)*. Jurnal Fenomena, 6 (1). ISSN: 1693-4296.
- Bodnar, George H. dan William S. Hopwood. 2004. *Sistem Informasi Akuntansi*. Edisi 9. Yogyakarta: Andi.
- BPBD, 2019. "Sejarah BPBD", <https://bpbd.malangkab.go.id/pd/slug?title=sejarah-bpbd>, diakses 10 Mei 2022.
- Burch, John dan Grudnitski, Gary. 1986. *Information Systems Theory and Practice*. New York: John Wiley & Sons.
- Hidayati, D. 2005. *Panduan Merintis Siaga Bencana Berbasis Masyarakat*. Jakarta: LIPI – Bidang Pendidikan Kelautan, CO REMAP.
- Indonesia. *Undang-Undang Nomor 24 Tahun 2007 tentang Penanggulangan Bencana*. Lembaran Negara RI Tahun 2007 Nomor 12. Sekretariat Negara. Jakarta.
- Jogiyanto. 2008. *Metodologi Penelitian Sistem Informasi*. Yogyakarta: CV Andi Offset.
- Kadir, Abdul. 2005. *Pengenalan Sistem Informasi*, Yogyakarta: Andi.
- Kadir, Abdul. 2008. *Belajar Database Menggunakan MySQL*, Yogyakarta: Andi.
- Kodoatie, Robert J., dan Sjarief. 2006. *Pengelolaan Sumber Terpadu-Banjir, Longsor, Kekeringan dan Tsunami*. Jakarta: Yarsif Watampone (Anggota IKAPAI).
- Krismiaji. 2005. *Sistem Informasi Akuntansi*, Yogyakarta: Unit Penerbit dan Percetakan Akademi Manajemen Perusahaan YKPN.
- Nani, Nurrachman. 2007. *Pemulihan Trama: Panduan Praktis Pemulihan Trauma Akibat Bencana Alam*. LPSP3. UI Jakarta.

- Nuraida, Ida. 2008. *Manajemen Administrasi Perkantoran*. Yogyakarta: Kanisius.
- Nurjanah. 2012. *Manajemen Bencana*. Bandung: Alfabeta.
- Solichin, A. 2016. *Pemograman Web dengan PHP dan MySQL*. Jakarta: Penerbit Budi Luhur.
- Stair. 1992. *Principle of Information System a Managerial Approach*. Boston, USA: Boyd & Fraser Publishing Company.
- Sutabri, Tata. 2003. *Analisis Sistem Informasi*. Yogyakarta: Andi
- Sutabri, Tata. 2005. *Sistem Informasi Manajemen*. Yogyakarta: Andi Offset.
- Sutabri, Tata. 2012. *Konsep Sistem Informasi*. Yogyakarta: Penerbit Andi.
- Sutanta, Edhy. 2011. *Sistem Informasi Manajemen*. Yogyakarta: Andi.
- Yakub. 2012. *Pengantar Sistem Informasi*. Edisi Pertama. Yogyakarta: Graha Ilmu.

Sistem Informasi
**REHABILITASI
DAN
REKONSTRUKSI**
Pascabencana Berbasis WEB

Di era modern ini, perkembangan teknologi informasi menuntut untuk terus mengikuti perkembangan teknologi, begitu juga dengan instansi yang akan melakukan pendataan pascabencana seperti sistem informasi rehabilitasi dan rekonstruksi. Pada Kantor Badan Penanggulangan Bencana Daerah terdapat beberapa pengelolaan data yang masih dilakukan secara konvensional salah satunya yaitu pendataan rehabilitasi dan rekonstruksi pascabencana. Proses seperti ini dapat mempersulit petugas dalam pendataan dan pembuatan laporan, sehingga hal ini dapat memperlambat dan menghambat kerja, serta memungkinkan data dan informasi yang diperoleh kurang akurat, kurang efektif dan efisien. Sehingga penulis mencoba membuat sebuah Sistem Informasi Rehabilitasi dan Rekonstruksi Pascabencana pada Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara yang lebih optimal dan terorganisir dengan baik. Untuk menunjang hal tersebut, dibuatkan sebuah sistem informasi rehabilitasi dan rekonstruksi pascabencana yang terkomputerisasi menggunakan bahasa pemrograman *PHP* dan *database MySQL* sehingga mampu mendukung kinerja pegawai dalam meningkatkan efisiensi pengelolaan data rehabilitasi dan rekonstruksi pada Badan Penanggulangan Bencana Daerah. Dari sistem yang penulis rancang yaitu sistem rehabilitasi dan rekonstruksi didapat kemudahan dalam memasukkan data pascabencana karena pada sistem yang penulis rancang tinggal menginput data sehingga lebih memudahkan petugas untuk bekerja dibandingkan sebelumnya, serta terdapat kolom pencarian agar lebih mudah mencari data yang sudah lama tersimpan. Hasil dari penelitian ini adalah menghasilkan sistem informasi rehabilitasi dan rekonstruksi pascabencana pada Kantor Badan Penanggulangan Bencana Daerah Kabupaten Aceh Utara yang dapat menyimpan data pascabencana, isu/permasalahan, rencana kerja dan juga laporan kegiatan di bidang rehabilitasi dan rekonstruksi. Dari sistem yang dirancang ini diharapkan dapat membantu dan meringankan pekerjaan pegawai serta dapat dikembangkan lebih baik lagi kedepannya.



ISBN 978-623-09-2740-9



9 786230 927409