

**PERANCANGAN
SISTEM INFORMASI BEM
BERBASIS WEB**

PERANCANGAN SISTEM INFORMASI BEM BERBASIS WEB

**Cahyo Prianto
Anisa Rosalina
Yola Vegita**

*“Salah satu kunci kebahagiaan adalah menggunakan uang
Anda untuk pengalaman bukan untuk keinginan.”
Bj Habibie.*

Kata Pengantar

Assalamualaikum warahmatullahi wabarakatuh. Segala puji bagi Allah SWT yang telah memberikan kemudahan sehingga dapat menyelesaikan pembuatan buku aplikasi ini, tanpa pertolongan-Nya mungkin penulis tidak akan sanggup menyelesaikannya dengan baik. Shalawat dan salam semoga terlimpah curahkan kepada Nabi Muhammad SAW beserta sahabat dan keluarga Beliau. Proses pembuatan buku ini juga tidak terlepas dari bantuan pihak Pembimbing. Oleh karena itu, pada kata pengantar ini penulis menyampaikan terimakasih kepada Tuhan YME dan seluruh pihak yang terkait dalam pembuatan Buku yang berjudul "Perancangan Sistem Informasi Badan Eksekutif Mahasiswa Berbasis Website". Penulis telah membuat laporan ini dengan sebaik-baiknya, diharapkan memberikan kritik dan saran dari semua pihak yang bersifat membangun, terimakasih.

Bandung, Jawa Barat
Februari, 2020

List of Figures

2.1	Daftar Atribut <i>HTML</i>	21
2.2	Hirarki Operator	32
2.3	Operasi Aritmatika	33
2.4	Hirarki Operator Aritmatika	34
2.5	Operator Bitwise	36
2.6	Operator Perbandingan	37
2.7	Perbandingan dengan tipe yang berbeda	37
2.8	Operator Eksekusi	39
2.9	Operator Penambahan dan Pengurangan	39
2.10	Operator Logika	40
2.11	Operator Array	41
2.12	Struktur IF	45
2.13	if else	47

2.14	if elseif else	48
2.15	While	50
2.16	Do-While	52
2.17	For	53
2.18	Data Definition Language (DDL)	66
2.19	Data Manipulation Language (DML)	66
2.20	Data Control Language (DCL)	67
2.21	Opsi di MySQL Server	68
2.22	Pengertian Database	74
2.23	Fungsi Database	75
2.24	Jenis-Jenis DB	77
2.25	DB MySQL	80
2.26	DB Oracle	81
2.27	DB Microsoft SQL Server	82
2.28	DB MariaDB Enterprise	84
2.29	Pengertian Perangkat Keras	85
2.30	Fungsi Perangkat Keras	86
2.31	Macam-macam Perangkat Keras	87
2.32	Contoh Perangkat Keras	88

3.1	Download Sublime	93
3.2	Klik Save	93
3.3	Klik File	94
3.4	pilih folder penyimpanan file sublime	94
3.5	Centang kotak	95
3.6	Klik Install	95
3.7	Install Selesai	96
3.8	Download XAMPP	97
3.9	Installer Language	97
3.10	Error 1	98
3.11	Error 2	98
3.12	Klik next	99
3.13	Centang App	99
3.14	Penyimpanan folder	100
3.15	Klik finish	101
3.16	Klik yes	101
3.17	Mengaktifkan XAMPP	102
3.18	XAMPP aktif	102
3.19	Tampilan Locallhost	103

3.20 Membuat database baru dengan MySQL 104

3.21 Database Baru MySQL 105

3.22 Database berhasil dibuat 105

3.23 Add User to Database 105

3.24 Manage Privilage 106

3.25 PhpMyAdmin 107

3.26 Export Database dari PhpMyAdmin 108

3.27 Importt Database dari PhpMyAdmin 109

3.28 Empty Database di phpMyAdmin 111

3.29 Drop Database di phpMyAdmin 112

3.30 Rename Database 112

3.31 Memilih database mydatabase 113

3.32 membuat table siswa 114

3.33 membuat Struktur Table 114

3.34 Konfirmasi Creat Table 115

3.35 Memilih Database dan Table 116

3.36 Memilih Insert 116

3.37 Mengisi Data Pada Tabel 117

3.38 input data 117

3.39 Memilih Database dan Table	118
3.40 Konfirmasi Perintah Select	118
3.41 Menuju Ke Halaman Home	119
3.42 Menu Ke Halaman Ekspor	120
3.43 Caption	120
3.44 Properti Tambah Sebelum Ekspor	121
3.45 Pilihan Apakah Diekspor Ke File	121
3.46 Jika hasil ekspor sekedar ditampilkan saja.	122
3.47 Konfirmasi Jika Hasil Ekspor Disimpan Ke File	122
3.48 Kembali Ke Menu Utama	123
3.49 Menu Ke Halaman Impor	123
3.50 Membuka file ekspor	124
3.51 Konfirmasi hasil impor.	125
3.52 Menghapus Database dengan menu Drop	126
3.53 Konfirmasi Penghapusan Database	126
3.54 Konfirmasi bahwa penghapusan database berhasil	127
3.55 Menu Privillage	128
3.56 Menu Add new User Untuk Menambah User Baru	128
3.57 Menentukan User name, Host dan Password MySQL	129

3.58	Isin Hak Akses User	129
3.59	Konfirmasi Pembuatan User Account Baru	130
4.1	Use Case	132
4.2	Actifity Diagram Login	133
4.3	Actifity Diagram Kelola Info	134
4.4	Actifity Diagram Kelola Event	134
4.5	Actifity Diagram Kelola Needme	135
4.6	Actifity Diagram Kelola Tentang	135
4.7	Actifity Diagram Kelola Profil	136
4.8	Actifity Diagram View Informasi	136
4.9	Actifity Diagram View Event	137
4.10	Actifity Diagram View Needme	137
4.11	Activiity Diagram View Tentang	138
4.12	Actifity Diagram View Profil	138
4.13	Sequence Diagram Login	139
4.14	Sequence Diagram Kelola Info	140
4.15	Sequence Diagram Kelola Event	141
4.16	Sequence Diagram Kelola Needme	142
4.17	Sequence Diagram Kelola Tentang	143

4.18	Sequence Diagram Kelola Profil	144
4.19	Sequence Diagram View Info	145
4.20	Sequence Diagram View Event	146
4.21	Sequence Diagram View Needme	146
4.22	Sequence Diagram View Tentang	147
4.23	Sequence Diagram View Profil	148
4.24	Class Diagram	149
5.1	Halaman Login Admin	183
5.2	Halaman Awal Admin	184
5.3	Halaman Kelola Info Hiburan	184
5.4	Halaman Kelola Info Lomba Akademik	185
5.5	Halaman Kelola Info Lomba Olahraga	185
5.6	Halaman Kelola Info Lomba Seni	186
5.7	Halaman Kelola Info Kosan	186
5.8	Halaman Kelola Info Kantin	187
5.9	Halaman Kelola Info Laundry	187
5.10	Halaman Kelola Info Tailor	188
5.11	Halaman Kelola Info Catering	188
5.12	Halaman Awal User	189

5.13 Halaman Awal User2	189
5.14 Halaman Info Minat Dan Bakat	190
5.15 Halaman Info Komunitas	190
5.16 Halaman Info Hiburan	191
5.17 Halaman Info Lomba Akademik	191
5.18 Halaman Info Lomba Olahraga	192
5.19 Halaman Info Lomba Seni	192
5.20 Halaman Info Kosan	193
5.21 Halaman Info Kantin	193
5.22 Halaman Info Laundry	194
5.23 Halaman Info Tailor	194
5.24 Halaman Info Catering	195
5.25 Halaman Info Logo BEM	195
5.26 Halaman Info Visi dan Misi	196
5.27 Halaman Info Struktur Organisasi	196
5.28 Halaman Info Pengurus Inti	197
5.29 Halaman Info Pengurus Internal Kampus	197
5.30 Halaman Info Pengurus Kesejahteraan Mahasiswa	198
5.31 Halaman Info Pengurus Eksternal Kampus	198

Daftar Isi

Kata Pengantar	ii
1 PENDAHULUAN	1
1.1 LATAR BELAKANG	1
1.2 PENGENALAN TENTANG SISTEM INFORMASI	3
1.2.1 Konsep Dasar Sistem	3
1.2.2 Konsep Dasar Informasi	7
1.2.3 Konsep Dasar	10
2 PENGGUNAAN TOOLS	13
2.1 PENGENALAN BAHASA PEMROGRAMAN <i>HTML</i>	13
2.1.1 Pengertian <i>HTML</i>	13
2.1.2 Definisi <i>HTML</i> Menurut Para Ahli	14
2.1.3 Sejarah <i>HTML</i>	15
2.1.4 Fungsi <i>HTML</i>	19
2.1.5 Struktur <i>HTML</i>	20
2.1.6 Cara Kerja <i>HTML</i>	22
2.1.7 Gambaran Umum	23
2.1.8 Tag <i>block-level</i>	23
2.1.9 Tag <i>Inline</i>	24
2.1.10 Kelebihan dan Kekurangan <i>HTML</i>	25
2.2 PENGENALAN BAHASA PEMROGRAMAN <i>PHP</i>	25
2.2.1 Pengenalan <i>PHP</i>	25
2.2.2 Pengertian <i>PHP</i> Menurut Beberapa Ahli	28
2.2.3 Sejarah <i>PHP</i>	29
2.2.4 Fungsi <i>PHP</i>	30
2.2.5 Operator	31
2.2.6 Struktur Kontrol	44

2.2.7	Prevdefine Variable	58
2.2.8	Penangann Session	60
2.3	PENGENALAN PHPMYADMIN	63
2.3.1	Pengertian phpMyAdmin	63
2.3.2	Sejarah phpMyAdmin	63
2.3.3	Fitur-Fitur phpMyAdmin	64
2.4	PENGENALAN MySQL	64
2.4.1	Pengertian MySQL	64
2.4.2	Sejarah MySQL	65
2.4.3	Mengenal SQL	65
2.4.4	Pengertian MySQL	67
2.4.5	Fungsi-Fungsi di MySQL Server	67
2.4.6	Login dan Logout MySQL Server	67
2.4.7	Opsi di MySQL Server	68
2.4.8	Perintah Administrasi di MySQL Server	68
2.5	PENGENALAN CSS	69
2.5.1	Pengertian CSS	69
2.5.2	Cara Kerja CSS	69
2.5.3	Peran CSS	70
2.5.4	Macam-Macam CSS	70
2.6	HUBUNGAN CSS, HTML, PHP DAN MySQL	71
2.7	PENGENALAN XAMPP	72
2.7.1	Pengertian XAMPP	72
2.7.2	Fungsi XAMPP	73
2.7.3	Bagian-Bagian Penting di Dalam XAMPP	73
2.8	PENGENALAN DATABASE	74
2.8.1	Pengertian Database	74
2.8.2	Fungsi Database	75
2.8.3	Jenis-Jenis Database	77
2.8.4	Pembagian Basis Data	78
2.8.5	Contoh Beberapa Basis Data	79
2.9	PERANGKAT KERAS	85
2.9.1	Pengertian Perangkat Keras	85
2.9.2	Fungsi Perangkat Keras	86
2.9.3	Macam-Macam Perangkat Keras	87
2.9.4	Contoh Perangkat Keras	88
3	INSTALASI DAN CARA PENGGUNAAN TOOLS	91
3.1	CARA MENGINSTALL MYSQL DI WINDOWS DAN LINUX	91

3.2	CARA INSTALLASI SUBLIME	93
3.3	INSTALLASI XAMPP	97
3.3.1	Proses Instalasi XAMPP di Windows	97
3.3.2	Menjalankan XAMPP	101
3.3.3	Membuat Database dengan XAMPP	103
3.4	MEMBUAT DATABASE MENGGUNAKAN MySQL DATABASE MELALUI cPANEL	104
3.4.1	Export database menggunakan phpMyAdmin	108
3.4.2	Import database menggunakan phpMyAdmin	108
3.4.3	Cara Mengelola Database di phpMyAdmin	109
3.4.4	Menghapus Database di phpMyAdmin	110
3.4.5	Mengganti Nama Database di phpMyAdmin	111
3.4.6	Membuat Database Baru di phpMyAdmin	112
3.4.7	Upload Database ke phpMyAdmin	113
3.4.8	Membuat Table	113
3.4.9	Insert Table	115
3.4.10	Menampilkan Table	118
3.4.11	Ekspor Struktur Database	119
3.4.12	Impor Struktur Database	122
3.4.13	Menghapus Table dan Database	125
3.4.14	Membuat User dan Hak Akses	127
4	PERANCANGAN APLIKASI	131
4.1	PERANCANGAN APLIKASI	131
4.1.1	Use Case	131
4.1.2	Activity Diagram	132
4.1.3	Sequence Diagram	139
4.1.4	Class Diagram	149
4.1.5	Model "Air Terjun" (<i>Waterfall</i>)	150
4.1.6	Tahap-tahap metode <i>Waterfall</i>	150
4.1.7	Keunggulan dan Kelemahan Metode <i>Waterfall</i>	151
4.2	CARA PEMBUATAN APLIKASI	152
4.2.1	<i>Cascading Style Sheet (CSS)</i>	152
4.2.2	Tampilan Admin	161
4.2.3	Tampilan User	171
5	CARA MENJALANKAN APLIKASI	183
5.1	TAMPILAN ADMIN	183

Bibliography

199

PENDAHULUAN

1.1 LATAR BELAKANG

Perkembangan didalam dunia teknologi informasi saat ini sudah sedemikian pesat dan merambah pada berbagai sisi kehidupan manusia. Perkembangan yang didukung oleh adanya perangkat keras ataupun perangkat lunak yang semakin hebat kemampuannya, yang memudahkan manusia untuk melakukan kegiatan mereka sehari-hari. Teknologi informasi tidak dapat kita pisahkan dengan perkembangan dunia informasi internet pada saat ini, informasi yang disajikan di internet sudah sangat global dan selalu diusahakan untuk on time sehingga waktu update suatu informasi sangat cepat. Banyak sekali situs-situs yang menyediakan informasi, yang sebagian besar diantaranya menggunakan basis data yang lebih dikenal dengan nama *data base* sebagai tempat penyimpanan data dalam menyediakan informasi yang diperlukan oleh perancang. Perancangan aplikasi berbasis *WEB*, telah terbukti sebagai pencari informasi yang sangat mudah dalam memperoleh data dan informasi yang dibutuhkan. Kemudahan serta kepraktisan distribusi informasi dan pertukaran antara pengguna mengatur bagaimana aplikasi dapat berinteraksi dengan beberapa user. Perancangan aplikasi berbasis *WEB* dapat dioperasikan dan dijalankan di *WEB* server ataupun pada client side dalam bentuk browser. [4]

Dengan demikian informasi yang dibutuhkan, baik untuk berinteraksi dengan user, verifikasi data yang dimasukkan oleh user, mengubah bentuk tombol dan sebagainya dapat dilakukan dengan mudah pada perancangan aplikasi berbasis *WEB*. Hal ini juga berkaitan dalam menyimpan data. Pengolahan data dan penyimpanannya merupakan materi yang cukup kompleks dalam membangun sebuah aplikasi. Karena kecepatan, keutuhan, dan keamanan data merupakan faktor yang sangat penting dalam pembangunan sebuah aplikasi. Banyak manfaat yang dapat diperoleh oleh organisasi dalam penggunaan perancangan

aplikasi berbasis *WEB*. Hal tersebut dapat mendukung kinerja organisasi, diantaranya adalah kemudahan berinteraksi dengan *user (admin)*, keamanan data organisasi lebih terjaga, dan lebih ekonomis. Hal-hal tersebut merupakan beberapa tinjauan alasan bahwa program aplikasi *WEB* lebih unggul untuk saat ini. *WEB* merupakan media *alternative* yang diharapkan dapat membantu kinerja organisasi dalam pengolahan data, khususnya bagi organisasi yang bergerak dalam dunia teknologi informasi.

Pada era globalisasi dunia untuk saat ini, diperlukan adanya kecepatan, keakuratan dalam suatu pencarian sebuah informasi. Bayangkan saja dengan begitu banyak data yang perancang punya dan perancang memerlukan salah satu informasi, yang ada perancang harus mencarinya satu persatu dengan waktu yang tidak sedikit atau juga membuang waktu, apalagi informasi yang perancang cari berbentuk berkas yang tidak sedikit jumlahnya. Walaupun dengan penyimpanan data dikomputer dalam bentuk file tidak mudah bagi para penggunanya untuk mencari suatu karakter yang ada pada file tersebut, karena karakter mungkin saja berada pada sisi file bukan pada judul file. Dengan begitu hambatan dalam menyelesaikan suatu pekerjaan akan memakan banyak waktu. Salah satu data yang dapat diambil contoh adalah data organisasi. Membangun yang tidak sedikit jumlahnya, dimana setiap tahun akan terus bertambah seiring dengan banyaknya jumlah anggota mahasiswa yang bekerja tiap tahun.

Dengan adanya beberapa karakter yang ada pada data organisasi misalnya saja dilihat dari nama, jabatan, npm, jurusan, alamat. Dapat dibayangkan kesulitan yang dapat timbul bila ada pencarian informasi dari salah satu karakter tersebut di atas, apalagi jika data-datanya belum tersusun secara rapi. Tentu akan banyak masalah timbul bila hal tersebut tidak segera diatasi dengan data yang lebih *efisien*, sehingga memudahkan dalam pencarian suatu karakter pada data. Data dengan menggunakan pencarian nama file yang ada, tanpa tahu apa yang ada didalamnya tidaklah efektif, harus ada yang lebih rinci dalam pencarian suatu informasi yang dibutuhkan. Oleh karena itu, diperlukan suatu basis data yang merupakan kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk suatu bangunan data untuk menginformasikan suatu organisasi dalam suatu batasan tertentu.

Dengan menggunakan basis data perancang membuat pembudayaan dari sistem manual ke sistem komputerisasi yang terpusat yang dapat mengendalikan data

informasi, konsep perancang aplikasi berbasis *WEB* muncul sebagai salah satu sebuah solusi seiring dengan kecepatan olah dan ruang simpan yang bertambah. Perancangan situs *WEB* membutuhkan banyak pertimbangan, perancang tidak hanya memutuskan apa yang harus disertakan dalam situs, tetapi juga merancang bagaimana menyajikan informasi pada situs. Perancangan awal merupakan langkah penting dalam membuat situs *WEB*. Tanpa rancangan yang baik. Situs *WEB* anda mungkin tidak sukses, artinya dalam skenario terburuk, tidak ada orang yang ingin melihatnya. Menanyakan beberapa pertanyaan penting kepada perancang dan mengembangkan rencana berdasarkan jawaban-jawaban tersebut akan membantu perancang membuat situs *WEB* yang sukses dan menarik. Dari permasalahan tersebut di atas untuk memudahkan data dan pencariannya, mencoba membuat suatu programan PHP dan MySQL berbasis web, yang dapat memudahkan dalam penyusunan dan pencarian data. Maka organisasi membangun ini mengalami perkembangan yang luar biasa, banyaknya data yang tersimpan secara baik.

1.2 PENGENALAN TENTANG SISTEM INFORMASI

1.2.1 Konsep Dasar Sistem

1.2.1.1 Definisi Sistem

Pengertian System Menurut Para Ahli :

1. Fat

Sistem adalah suatu himpunan “benda” nyata atau abstrak yang terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan untuk mencapai tujuan tertentu secara efisien dan efektif.

2. Indrajit (2001 : 2)

mengemukakan bahwa system mengandung arti kumpulan-kumpulan dari komponen-komponen yang dimiliki unsur keterkaitan antara satu dengan lainnya.

3. Jogianto (2005 : 2)

mengemukakan bahwa system adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.

4. Murdick, R.G, (1991 : 27)

suatu system adalah seperangkat elemen yang membentuk kumpulan atau procedure-procedure/bagan-bagan pengolah yang mencari suatu tujuan tertentu.

5. Jerry FutGerald, (1982 : 5)

system adalah suatu jaringan kerja dari procedure-prosedure yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.

6. Davis, G.B, (1991 : 5)

system secara fisik adalah kumpulan dari elemen-elemen yang beroperasi bersama-sama untuk menyelesaikan suatu sasaran.

7. Dr. Ir. Harijono Djojodihardjo (1984 : 78)

suatu system adalah sekumpulan objek yang mencakup hubungan fungsional antara tiap-tiap objek dan hubungan antara ciri tiap objek, dan yang secara keseluruhan merupakan suatu kesatuan secara fungsional.

8. Lani Sidharta (1995 : 9)

system adalah himpunan dari bagian-bagian yang saling berhubungan yang secara bersama mencapai tujuan-tujuan yang sama.

Dengan demikian, system adalah suatu jaringan kerja dari procedure-prosedure yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu. Pendekatan system yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam system. Menurut *Richard F. Neuschel* suatu prosedur adalah suatu urutan operasi *klerikal* (tuliskan-menulis), yang melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi.

1.2.1.2 Karakteristik Sistem

Supaya system itu dikatakan system yang baik, maka memiliki karakteristik yaitu :

1. Komponen

Suatu system terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen system terdiri dari komponen yang berupa subsistem atau bagian-bagian dari system.

2. Batasan Sistem (*Boundary*)

Batasan system merupakan daerah yang membatasi antara suatu system dengan system yang lain atau dengan lingkungan luarnya. Batasan system ini memungkinkan suatu system dipandang sebagai satu kesatuan. Batasan system menunjukkan ruang lingkup (*scope*) dari system tersebut.

3. Lingkungan Luar Sistem (*environment*)

Lingkungan luar system (*environment*) adalah diluar batas dari system yang mempengaruhi operasi system. Lingkungan dapat bersifat menguntungkan yang harus tetap dijaga dan yang merugikan yang harus dijaga dan dikendalikan, kalau tidak akan mengganggu kelangsungan hidup system.

4. Penghubung Sistem (*Interface*)

Penghubung system merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain. Keluaran (*output*) dari subsistem lain melalui penghubung.

5. Masukkan Sistem (*Input*)

Masukkan adalah energy yang dimasukkan kedalam system, yang dapat berupa perawatan (*maintenance input*), dan masukkan sinyal (*signal input*). *Maintenance input* adalah energy yang dimasukkan agar system dapat beroperasi. *Signal input* adalah energy yang diproses untuk didapatkan keluaran. Contoh dalam system *computer* program adalah *maintenance input* sedangkan data adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran Sistem (*Output*)

Keluaran system adalah hasil dari energy yang dioalah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh *computer* menghasilkan panas yang merupakan sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

7. Pengolahan System

Suatu system menjadi bagian pengolah yang akan merubah masukan menjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi, system akuntansi akan mengolah data menjadi laporan-laporan keuangan.

8. Sasaran System

Suatu system pasti mempunyai tujuan atau sasaran (*objective*). Sasaran dari system sangat menentukan input yang dibutuhkan system dan keluaran yang akan dihasilkan system.

1.2.1.3 Klasifikasi Sistem

Sistem dapat diklasifikasikan dalam beberapa sudut pandang :

1. Klasifikasi system sebagai :

(a) Sistem Abstrak (*Abstracct System*)

System abstrak adalah system yang berupa pemikiran-pemikiran atau ide-ide yang tidak tampak seperti fisik.

(b) System Fisik (*Physical System*)

System fisik adalah system yang ada secara fisik

2. Sistem diklasifikasikan sebagai :

(a) System Alamiyah (*Natural System*)

System alamiyah adalah system yang terjadi melalui proses alam, tidak dibuat oleh manusia.

(b) System Buatan Manusia (*Human Made System*)

System buatan manusia adalah system yang dibuat oleh manusia yang melibatkan interaksi antara manusia dengan mesin (*human mechine system*).

3. System diklasifikasikan sebagai :

(a) System tertentu (*Deterministicl System*)

System tertentu adalah system yang beroperasi dengan tingkah laku yang sudah dapat diprediksi, sebagai keluaran system yang dapat diramalkan.

(b) System tak tentu (*Probalistic System*)

System tak tentu adalah system yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur *probabilistic*.

4. System diklasifikasikan sebagai :

(a) System tertutup (*Close Sistem*)

System tertutup adalah system yang tidak terpengaruh dan tidak berhubungan dengan lingkungan luar, system bekerja otomatis tanpa ada turut campur lingkungan luar. Secara teoritis system tertutup ini ada, kenyataannya tidak ada system yang benar-benar tertutup, yang ada hanya *relatively closed system*.

(b) System terbuka (*Open Sistem*)

System terbuka adalah system yang berhubungan dan terpengaruh dengan lingkungan luarnya. System ini menerima input dan output dari lingkungan luar atau subsistem lainnya, karena system terbuka terpengaruh lingkungan luar maka harus mempunyai pengendali yang baik.

1.2.2 Konsep Dasar Informasi

1.2.2.1 Definisi Data

1. Pengertian Data

Menurut *Gordon B. Davis*, data adalah bahan mentah bagi informasi, dirumuskan sebagai kelompok lambang-lambang tidak acak menunjukkan jumlah-jumlah, tindakan-tindakan, hal-hal dan sebagainya.

2. Metode Pengumpulan Data :

- (a) Melalui pengamatan sendiri secara langsung
- (b) Melalui wawancara
- (c) Melalui perkiraan korespondensi
- (d) Melalui daftar pertanyaan

3. Pengertian Pengelolaan Data :

George R. Terry Ph.D menyatakan, Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan.

8 unsur pokok pengolahan data, yaitu :

- (a) Membaca
- (b) Menulis, mengetik
- (c) Mencatat atau mencetak
- (d) Menyortir
- (e) Menyampaikan atau memindahkan
- (f) Menghitung
- (g) Membandingkan
- (h) Menyimpan

1.2.2.2 Definisi Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Data kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah kejadian yang terjadi pada saat tertentu.

Menurut *Gordon B. Davis*, informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi penerima dan mempunyai nilai nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang ada keputusan-keputusan yang akan datang [8]

1. Fungsi Informasi

Fungsi utamanya yaitu menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi, karena informasi berguna memberikan gambaran tentang suatu permasalahan sehingga pengambil keputusan dapat menentukan keputusan lebih cepat, informasi juga memberikan *standard*, aturan maupun *indicator* bagi pengambil keputusan.

2. Kegunaan informasi tergantung pada :

(a) Tujuan penerima

Bila tujuannya untuk memberi bantuan, maka informasi itu harus membantu penerima dalam apa yang ia usahakan untuk memperolehnya.

(b) Ketelitian penyampain dan pengolahan data

Dalam menyampaikan dan mengolah data, inti dan pentingnya informasi harus dipertahankan.

(c) Waktu

yaitu tentang keupdatean informasi

(d) Ruang dan tempat

Ketersediaan informasi dalam ruangan atau tempat yang tepat.

(e) Bentuk

Dapatkah informasi itu digunakan secara efektif. Apakah informasi itu menunjukkan hubungan-hubungan yang diperlukan, bidang-bidang yang memerlukan perhatian manajemen? dan apakah informasi itu menentukan situasi-situasi yang ada hubungannya.

(f) Semantic

Apakah hubungan antara kata-kata dan arti yang diinginkan cukup jelas? apakah ada kemungkinan salah tafsir?

1.2.2.3 Siklus Informasi

Data yang diolah untuk menghasilkan informasi menggunakan model proses yang tertentu. Misalkan suhu dalam *Fahrenheit* diubah ke *celcius*. Dalam hal ini digunakan model matematika berupa rumus konversi dari derajat *fahrenheit* menjadi satuan derajat *celcius*. Data yang diolah melalui suatu

model menjadi informasi tersebut, yang berarti menghasilkan keputusan dan melakukan tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya yang disebut dengan siklus informasi. Siklus ini juga disebut dengan siklus pengolahan data.

1.2.2.4 Nilai Informasi (*Value Of Information*)

Biaya informasi terdiri dari :

1. Biaya perangkat keras
Merupakan biaya tetap atau biaya tertanam dan akan meningkat untuk tingkat-tingkat mekanisasi yang lebih tinggi.
2. Biaya untuk analisis
Merupakan biaya tertanam, dan biasanya akan meningkat sesuai dengan tingkat mekanisasi yang lebih tinggi
3. Biaya untuk tempat dan faktor control lingkungan
Biaya ini setengah berubah/semi variable. Biasanya biaya ini meningkat sesuai dengan tingkat mekanisasi yang tinggi.
4. Biaya perubahan
Biaya ini merupakan biaya tertanam dan meliputi setiap jenis perubahan dari suatu metode ke metode yang lain.
5. Biaya operasi
Biaya ini pada dasarnya merupakan biaya variable dan meliputi biaya macam-macam pegawai, pemeliharaan fasilitas dan system.

1.2.3 Konsep Dasar

1.2.3.1 Konsep Dasar Sistem Informasi

System informasi adalah suatu system didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat

manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan.

1.2.3.2 Konsep Sistem Informasi

System informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (*building block*) yaitu :

1. Blok masukan (*input block*)

Input mewakili data yang masuk ke dalam system informasi. Input disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dasar.

2. Blok model (*model block*)

Blok ini terdiri dari kombinasi prosedur, logika dan metode matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang sudah diinginkan.

3. Blok keluaran (*output block*)

Produk dari system informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai system.

4. Blok teknologi (*technology block*)

Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian diri secara keseluruhan.

Teknologi terdiri dari unsur utama :

- (a) Teknisi (*human ware atau brain ware*)
- (b) Perangkat lunak (*software*)
- (c) Perangkat keras (*hardware*)

5. Blok basis data (*data base block*)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras *computer* dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok kendali (*control block*)

Banyak faktor yang dapat merusak system informasi, misalnya bencana alam, api, temperature tinggi, air, debu, kecurangan-kecurangan, kejang-galan system itu sendiri, kesalahan-kesalahan ketidakefisienan, sabotase dan sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak system dapat dicegah atau bila terlanjur terjadi kesalahan dapat langsung diatasi.

PENGUNAAN TOOLS

2.1 PENGENALAN BAHASA PEMROGRAMAN *HTML*

2.1.1 Pengertian *HTML*

HTML adalah singkatan dari *Hypertext Mark Up Language* yang merupakan bahasa standar pemrograman untuk membuat suatu *website* yang bisa diakses dengan *internet*. Dengan kata lain halaman *website* yang kita lihat dan kita baca disusun dengan menggunakan bahasa ini dan kemudian diterjemahkan oleh komputer agar dapat dipahami oleh penggunanya. *HTML* merupakan standar pembuatan *website* secara luas agar laman *website* dapat ditampilkan pada layar komputer.

HTML adalah bahasa *mark up web* yang utama dan dijalankan secara alami di setiap browser serta di-maintain oleh *World Wide Web Consortium*. *HTML* disusun dengan kode dan simbol tertentu yang dimasukkan ke dalam sebuah file atau dokumen. Jadi setiap Anda membuka *website* apapun dengan menggunakan browser maka web tersebut dibuat dengan menggunakan *HTML*. *HTML* adalah singkatan dari *Hyper Text Markup Language*. Tapi mengetahui singkatannya saja tidak akan cukup. Karena itu saya akan menyajikan arti dari kata kata tersebut.[6]

1. *HyperText* adalah metode dimana kita “berpindah” disekeliling web, dengan mengeklik sebuah teks yang bernama hyperlink. *HyperText* adalah sebuah teks khusus di *internet*, dimana saat teks tersebut diklik, akan membawa kita ke halaman web selanjutnya/halaman web lain yang telah ditentukan.

2. *Markup* adalah hal yang dilakukan oleh tag HTML kepada teks yang ada didalamnya. *HTML* menandai teks yang berada didalamnya sebagai tipe teks tertentu. Misalnya saja jika kita menandai sebuah teks dengan tag html `<i>`, maka teks tersebut akan berubah menjadi *italic* (huruf yang miring). Sedangkan jika kita menandainya dengan ``, maka teks tersebut akan berubah menjadi bold (huruf tebal).
3. *Language* yang berarti bahasa. *HTML* adalah sebuah bahasa, yang memiliki kata kata berupa kode dan syntax seperti bahasa yang lain. Berdasarkan cara kerja *HTML*, Anda bisa menggunakannya untuk membuat struktur konten pada *website* dan aplikasi web. *HTML* merupakan level terbawah dari teknologi frontend dan berfungsi sebagai dasar styling yang bisa Anda tambahkan dengan *CSS* dan fungsionalitas yang dapat dijalankan menggunakan *JavaScript*. [2]

2.1.2 Definisi *HTML* Menurut Para Ahli

Berikut ini ada beberapa pengertian *HTML* menurut para ahli:

1. Lia Kuswayatno

HTML merupakan halaman yang berada pada suatu situs internet atau web. *HTML* merupakan metode yang menautkan (*link*) satu dokumen ke dokumen lain melalui teks

2. Deris Setiaan

HTML merupakan *framework internet*, hampir semua situs web yang ada menggunakan *HTML* untuk menampilkan teks, grafik, suara, dan animasinya

3. Diar Puji Oktavian

HTML adalah suatu bahasa yang dikenali oleh web browser untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan teks biasa (*plain text*)

4. Menurut Arief (2011c:23)

"*HTML (HyperText Markup Language)* merupakan salah satu format yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan dihalaman web".

5. Menurut Sibero (2011c:19)

“*Hyper Text Markup Language* atau *HTML* adalah bahasa yang digunakan pada dokumen web sebagai bahasa untuk pertukaran dokumen *web*”. Dokumen *HTML* terdiri dari komponen yaitu tag, elemen dan atribut. Tag adalah tanda awal < dan tanda akhir > yang digunakan sebagai pengapit suatu elemen. Elemen adalah nama penanda yang diapit oleh tag yang memiliki fungsi dan tujuan tertentu pada dokumen *HTML*. Elemen dapat memiliki elemen anak dan juga nilai. Elemen anak adalah suatu elemen yang berada didalam elemen pembuka dan elemen penutup induknya. Nilai yang dimaksud adalah suatu teks atau karakter yang berada diantara elemen pembuka dan elemen penutup. Atribut adalah properti elemen yang digunakan untuk mengkhususkan suatu elemen. Elemen dapat memiliki atribut yang berbeda pada tiap masing-masingnya.

6. Menurut Astamal (2006:8)

HTML (Hyper Text Markup Language) adalah simbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang dimaksudkan untuk menampilkan halaman pada web browser. Tag-tag *HTML* selalu diawali dengan dan diakhiri dengan dimana x tag *HTML* seperti b, i, u dan sebagainya. Namun ada juga tag yang tidak diakhiri dengan tanda. [6]

2.1.3 Sejarah *HTML*

HTML dibuat oleh Tim Berners-Lee, seorang ahli fisika di lembaga penelitian *CERN* yang berlokasi di Swiss. Dia memiliki ide tentang sistem *hypertext* yang berbasis internet. *Hypertext* merujuk pada teks yang memuat referensi (*link*) ke teks lain yang bisa diakses langsung oleh *viewer*. Tim merilis versi pertama *HTML* pada tahun 1991, dan di dalamnya terdiri atas 18 *HTML tag*. Sejak saat itu, setiap kali bahasa *HTML* merilis versi terbarunya, selalu ada *tag* dan *attribute* (*tag modifier*) terbaru.

Berdasarkan *HTML Element Reference* milik *Mozilla Developer Network*, untuk saat ini, ada 140 *HTML tag* meskipun sebagiannya sudah usang (tidak lagi didukung oleh versi terbaru *browser*). Berkat popularitasnya yang terus meningkat, *HTML* kini dianggap sebagai web standard yang resmi. Spesifikasi *HTML* di-maintain dan dikembangkan oleh *World Wide Web Consortium* (*W3C*). Cek versi terbaru dari bahasa ini di *website W3C*.

Upgrade HTML besar-besaran terjadi pada tahun 1994, dan hasilnya adalah pengenalan HTML5. Pada upgrade tersebut, terdapat *semantic* baru yang memberitahukan arti dari kontennya sendiri, seperti `<article>`, `<header>`, dan `<footer>`. HTML pertama kali diciptakan oleh IBM pada tahun 1980. Saat itu tercetus ide untuk menempatkan elemen elemen yang berguna untuk menandai bagian suatu dokumen seperti judul, alamat dan isi dokumen. Pada akhirnya elemen element tersebut dibentuk menjadi suatu program untuk melakukan pemformatan dokumen secara otomatis. Bahasa pemrograman untuk melakukan tugas ini disebut sebagai bahasa *markup*. IBM menamai program ini sebagai *Generalized Markup Language*. Pada tahun 1986, konsep ini disetujui oleh ISO (*International Standard Organization*) sebagai standar untuk pembuatan dokumen. Bahasa ini dinamai oleh ISO sebagai *Standard Generalized Markup Language (SGML)* HTML sendiri, adalah bagian dari SGML. Tim Berners-Lee dari CERN mengemukakan sebuah ide tentang pembuatan skrip bahasa pemrograman dan dokumen yang dapat diakses oleh semua komputer tanpa melihat jenis platformnya. Sejak saat itu HTML menjadi lebih populer dibandingkan SGML.

Didalam HTML telah mengalami berbagai perkembangan dari tiap-tiap versinya. Berikut adalah versi-versi dari perkembangan HTML tersebut:

1. HTML Versi 1.0

HTML Versi 1.0 merupakan pionir yang di dalamnya masih terdapat banyak sekali kelemahan hingga wajar jika tampilan yang dihasilkan sangat sederhana. Kemampuan yang dimiliki versi 1.0 ini antara lain heading, paragraf, *hypertext*, list, serta cetak tebal dan miring pada teks. Versi ini juga mendukung peletakan image pada dokumennya tanpa memperbolehkan teks di sekelilingnya (*wrapping*)

2. HTML Versi 2.0

Pada HTML Versi ini, penambahan kualitas HTML terletak pada kemampuannya untuk menampilkan suatu form pada dokumen. Dengan adanya form ini, kita dapat memasukkan nama, alamat, serta saran dan kritik. HTML versi 2.0 ini merupakan pionir dari adanya web interaktif.

3. HTML Versi 3.0

Versi HTML3.0 menambahkan beberapa fasilitas baru seperti *FIGURE* yang merupakan perkembangan dari *IMAGE* untuk meletakkan gambar

dan tabel. Selain itu, *HTML* ini juga mendukung adanya rumus-rumus matematika dalam dokumennya. Versi ini yang disebut *HTML+* tidak bertahan lama dan segera digantikan dengan versi 3.2.

4. *HTML* Versi 3.2

HTML versi ini merupakan *HTML* yang sering digunakan. Di dalamnya terdapat suatu teknologi untuk meletakkan teks di sekeliling gambar, gambar sebagai latar belakang, tabel, frame, style sheet dan lain-lain. Selain itu pada *HTML* versi ini Kita bisa menggunakan script di luar *HTML* untuk mendukung kinerja *HTML* kita tersebut, seperti *Javascript*, *VBScript* dan lain-lain.

5. *HTML* Versi 4

HTML ini memuat banyak sekali perubahan dan revisi dari pendahulunya yaitu *HTML* 3.2. Perubahan ini hampir terjadi di segalaperintah *HTML* seperti tabel, *image*, *link*, *text*, *meta*, *imagemaps*, *form* dan lain-lain.

6. *HTML* Versi 4.01

HTML versi 4.01 merupakan revisi dari *HTML* 4.0. Versi terbaru ini memperbaiki kesalahan-kesalahan kecil (*minor errors*) pada versi terdahulunya. *HTML* 4.01 ini juga menjadi standarisasi untuk elemen dan atribut dari script *XHTML* 1.0.

7. *HTML* Versi 5.0

Teknologi ini mulai diluncurkan pada tahun 2009, tetapi pada tanggal 4 Maret 2010 terdapat sebuah informasi bahwa *W3C* (*World Wide Web Consortium*) dan *IETF* (*Internet Engineering Task Force*) yaitu sebuah organisasi yang menangani *HTML* sejak versi 2.0 telah mengembangkan versi *HTML* terbaru, yaitu versi 5.0.

HTML 5 adalah sebuah prosedur pembuatan tampilan web baru yang merupakan penggabungan antara *CSS*, *HTML* itu sendiri dengan *Javascript*. [6]

Beberapa kelebihan yang dimiliki oleh *HTML5* adalah:

1. *Cleaner code*

Karena sebagian besar kode telah termasuk di dalam sintaks *HTML5*, maka kode nampak terlihat lebih sederhana daripada penggabungan antara *HTML*, *CSS* dan *Java Script*.

2. *Greater consistency*

HTML5 telah melakukan banyak sekali penambahan sintaks yang dibuat dalam struktur lebih baik dan lebih sederhana daripada sintaks- sintaks sebelumnya. Hal ini membuat developer terbantu dalam meningkatkan konsistensi dalam membangun sebuah *web*.

3. *Improve Semantics*

Berbagai elemen kode di dalam *HTML5* yang telah distandarisasi, maka nilai semantik dari sebuah web dapat lebih ditingkatkan. Itu berarti bahwa bagian- bagian dari web seperti *header*, *nav*, *footer* dan beberapa bagian lainnya terdefinisi dengan jelas maksud serta tujuannya. Selain itu juga terbentuk dalam sebuah “*machine readable format*”.

4. *Improved Accessibility*

Teknologi *HTML5* yang memudahkan struktur pembangunan sebuah *web*, maka *developer* dapat membangun pemahaman yang lebih detail mengenai halaman web.

5. *Client-side Database*

HTML5 menyediakan model database *SQL* yang baru dengan *API* yang dapat dibangun dalam konsep lokal, dalam hal ini di sisi client.

6. *Geolocation*

HTML5 mempunyai *API* yang terintegrasi terhadap geolocation, fasilitas tersebut dapat diakses melalui GPS atau fasilitas lain seperti Google Latitude pada *iphone*.

7. *Offline Application Cache*

Pengguna dapat terus melakukan interaksi dengan aplikasi meskipun mereka terputus dari jaringan *internet*.

8. *Smarter Forms*

Terdapat semacam *regular expression (regex)* yang membuat form mampu mengenali secara lebih baik tentang input, validasi data dan interaksi dengan elemen lain (misal : format email, password dll)

9. Sharper focus on Web Application Requirments

HTML5 membuat sebuah mekanisme yang lebih mudah dalam hal pembuatan *front end*, aplikasi chat, *tools drag and drop*, *video player*, pengolah grafis dan masih banyak lagi.

2.1.4 Fungsi *HTML*

Html yang dikenal sebagai bahasa pemrograman dalam ilmu komputer memiliki beberapa fungsi, diantaranya adalah sebagai berikut:

1. Fungsi utama *html* yang diketahui adalah untuk membuat suatu halaman website yang bisa dibaca dan dipahami oleh pengguna dengan lebih mudah. Seluruh laman *website* yang ada dalam internet dibuat dengan *html* dan tidak ada pengecualian.
2. Menandai teks pada suatu laman, *html* ditulis pada suatu halaman dokumen dengan tag atau simbol tertentu dimana simbol dan tag tersebut akan menandai teks menjadi tebal, miring, bergaris tebal dan lain sebagainya. Misal jika kita membuat suatu teks menjadi teks miring atau *italic*, dalam laman *html* dituliskan kode `<i>`, `` untuk teks tebal dan `<u>` untuk teks bergaris bawah.
3. Sebagai dasar *website*, *website* yang dibuat tentunya memiliki beberapa fitur yang dibuat dengan menggunakan *java script* (untuk mengatur perilaku web), implemetasi bahasa pemrograman server *PHP*, dan mendesain web menggunakan *CSS*. Semua bahasa tersebut dapat diaplikasikan jika *web* memiliki bahasa *html* sebagai dasarnya.
4. Menampilkan tabel, gambar, video, dan lainnya. Biasanya dalam website atau blog kita tidak bisa langsung meletakkan tabel, gambar maupun video oleh sebab itu komponen tersebut diletakkan pada *web* dengan menggunakan bahasa *html*.

5. Menandai elemen dan membuat *online form*, html juga berfungsi untuk menandai bagian- bagian dalam website diantaranya *header*, *main*, *footer*, *navigation* dan lain sebagainya. Selain itu html juga biasanya digunakan sebagai bahasa dalam membuat suatu online form atau formulir digital.

2.1.5 Struktur HTML

Berikut ini ada beberapa struktur HTML:

1. Elemen

Elemen terdiri atas tiga bagian, yaitu tag pembuka, isi, dan tag penutup. Contohnya untuk menampilkan judul dokumen *HTML* pada web browser digunakan element title, dimana: `<title>` ini adalah tag pembuka judul dokumen *HTML*.

2. Tag

Tag merupakan teks khusus (*markup*) berupa dua karakter "<" dan ">", sebagai contoh `<body>` adalah tag dengan nama body. Tag ditulis secara berpasangan, yang terdiri atas tag pembuka dan tag penutup (ditambahkan karakter "/" setelah karakter "<"), sebagai contoh `<body>` ini adalah tag pembuka isi dokumen *HTML*, dan `</body>` ini adalah tag penutup isi dokumen *HTML*. Yang merupakan tag-tag dasar dalam *HTML* adalah:

- `<html> </html>`

Tag `<html>` dan `</html>` digunakan untuk menandai awal dan akhir dari suatu file HTML.

- `<title> </title>`

Tulisan yang berada diantara tag `<title>` dan `</title>` akan ditampilkan oleh browser pada bagian title dan merupakan title dari jendela *browser*.

- `<head> </head>`

Berisi keterangan informasi, seperti title dan jenis dokumen, ditulis diantara `<head> </head>`.

- `<body> </body>`

Bagian tag `<body>` `</body>` menandai awal dan akhir dari badan dokumen *HTML*. Tag ini memiliki sejumlah attribut yang dapat ditentukan.

Contoh :

```
1 <html>
2
3 <head>
4
5 <title>Contoh Document HTML</title>
6 </head>
7
8 <body>SELAMAT MEMBACA</body>
9
10 </html>
```

3. Atribut

Atribut terdapat di dalam script sebuah elemen dan memberikan informasi tambahan tentang elemen. Atribut selalu ditentukan dalam tag awal Atribut ditulis dalam pasangan nama/nilai.

Nilai dari atribut harus selalu tertutup dalam tanda kutip. Tanda kutip ganda adalah yang paling umum, tapi gaya tanda kutip tunggal juga diperbolehkan. Dalam beberapa situasi, ketika nilai atribut itu sendiri berisi tanda kutip, maka perlu menggunakan tanda kutip tunggal Misalnya: `name = 'John "leo" Nelson'`.

Di bawah ini adalah daftar dari beberapa atribut yang dapat digunakan pada setiap elemen *HTML* : [6]

Atribut	Deskripsi
Class	Menentukan satu atau lebih classnames untuk sebuah elemen (mengacu pada kelas dalam style sheet)
Id	Menentukan id unik untuk sebuah elemen
Style	Menentukan inline CSS style untuk elemen
Title	Menentukan informasi tambahan mengenai elemen (ditampilkan sebagai tool tip)

Figure 2.1: Daftar Atribut *HTML*

2.1.6 Cara Kerja *HTML*

Bahasa pemrograman *HTML* adalah sebuah file yang diakhiri dengan ekstensi `.html` atau `.htm`. Ekstensi file ini bisa dilihat dengan menggunakan web browser apa pun (seperti *Google Chrome*, *Safari*, atau *Mozilla Firefox*). Untuk membuka ekstensi ini biasanya dilakukan penulisan tempat folder dan nama filenya pada *browser*. Browser tersebut membaca file *HTML* dan merender kontennya sehingga *user* internet bisa melihat dan membacanya. Biasanya, kebanyakan situs web menyertakan sejumlah halaman *HTML* yang berbeda-beda. Contohnya, beranda utama, halaman ‘tentang kami’, halaman kontak yang semuanya memiliki dokumen *HTML* terpisah. Pada halaman *HTML* masing-masing terdiri atas seperangkat tags, yang mengacu pada building block halaman *website*. Tag tersebut membuat hirarki yang menyusun konten hingga menjadi bagian, paragraf, heading, dan block konten lainnya. Sebagian besar element pada bahasa pemrograman *HTML* memiliki tag pembuka dan penutup yang menggunakan syntax `<tag></tag>`.

Berikut contoh kode dari susunan atau struktur *HTML* ;

```
1 <div>
2   <h1>Heading ke-satu</h1>
3   <h2>Heading ke-dua</h2>
4   <p>Paragraph</p>
5   
6   <p>Paragraph kedua<a href="https://example.com">
7     hyperlink</a>
8 </p>
9 </div>
```

1. Perintah teratas dan terbawah adalah division sederhana (`<div></div>`) yang bisa Anda gunakan untuk mark up bagian konten yang lebih besar.
2. Susunan *HTML* di atas terdiri atas heading (`<h1></h1>`), subheading (`<h2></h2>`), dua paragraf (`<p></p>`), dan satu gambar (``).
3. Paragraf kedua meliputi sebuah link (`<a>`) dengan attribute `href` yang terdiri atas URL tujuan.
4. Tag gambar memiliki dua attribute, `src` untuk path gambar dan `alt` untuk deskripsi gambar

2.1.7 Gambaran Umum

Tag *HTML* memiliki dua tipe utama: *block-level* dan *inline tags*.

1. Elemen *block-level* memakai semua space yang tersedia dan selalu membuat line baru di dalam dokumen. Contoh dari tag *block* adalah heading dan paragraf.
2. Elemen *inline* hanya memakai space sesuai dengan kebutuhannya dan tidak membuat line baru di halaman. Biasanya elemen ini akan memformat isi konten dari elemen *block-level*. Contoh dari tag *inline* adalah link dan emphasized strings.

2.1.8 Tag *block-level*

Tiga tag *block-level* yang harus dimiliki oleh setiap dokumen *HTML* adalah `<html>`, `<head>`, dan `<body>`.

1. Tag `<html></html>` adalah elemen level tertinggi yang menyertakan setiap halaman *HTML*.
2. Tag `<head></head>` menyimpan informasi meta, seperti judul dan charset halaman.
3. Tag `<body></body>` melampirkan semua konten yang muncul pada suatu halaman.

```
1 <html>
2   <head>
3     <!-- META INFORMATION -->
4   </head>
5   <body>
6     <!-- PAGE CONTECT -->
7   </body>
8 </html>
```

- Heading memiliki 6 level di *HTML*. Level tersebut bervariasi, mulai dari `<h1></h1>` sampai ke `<h6></h6>`, di mana `h1` merupakan level heading tertinggi dan `h6` adalah level terendah. Paragraf dibuka dan ditutup dengan tag `<p></p>`, sedangkan *blockquote* menggunakan tag

`<blockquote></blockquote>.`

- Division merupakan bagian konten yang lebih besar dan biasanya terdiri atas beberapa paragraf, gambar, kadang-kadang `blockquote`, dan elemen lebih kecil lainnya. Kita bisa membuat mark up dengan menggunakan tag `<div></div>`. Di dalam elemen `div` juga terdapat tag `div` lainnya.
- Anda juga bisa menggunakan tag `` untuk list yang berurutan dan `` untuk list yang tidak berurutan. Masing-masing list item harus dibuka dan ditutup dengan tag ``. Sebagai contoh, di bawah ini adalah tampilan dasar dari list tidak berurutan dalam HTML:

```
1 <ul>
2   <li>List item 1</li>
3   <li>List item 2</li>
4   <li>List item 3</li>
5 \</ul>
```

2.1.9 Tag Inline

Sebagian besar tag inline digunakan untuk memformat teks. Sebagai contoh, tag `` akan render elemen ke format bold, sedangkan tag `` akan ditampilkan dalam format italic.

Hyperlink adalah elemen inline yang mewajibkan adanya tag `<a>` dan attribute `href` untuk mengindikasikan tujuan link:

```
1 <a href="https://example.com/">Click me!</a>
```

Gambar (image) juga merupakan elemen inline. Anda dapat menambahkan satu gambar dengan menggunakan `` tanpa harus membubuhkan tag penutup. Hanya saja, Anda disarankan menggunakan attribute `src` untuk menentukan path gambar, misalnya: ``

2.1.10 Kelebihan dan Kekurangan *HTML*

2.1.10.1 Kelebihan *HTML*

1. Bahasa yang digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar.
2. Dijalankan secara alami di setiap web browser.
3. Memiliki *learning curve* yang mudah.
4. Open-source dan sepenuhnya gratis.
5. Bahasa markup yang rapi dan konsisten.
6. Standard web yang resmi di-maintain oleh *World Wide Web Consortium (W3C)*.
7. Mudah diintegrasikan dengan bahasa backend, seperti PHP dan Node.js.

2.1.10.2 Kekurangan *HTML*

1. Paling sering digunakan untuk halaman web statis. Untuk fitur dinamis, Anda bisa menggunakan *JavaScript* atau bahasa backend, seperti *PHP*.
2. *HTML* tidak memungkinkan *user* untuk menjalankan logic. Alhasil, semua halaman web harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti *header* dan *footer*.
3. Fitur-fitur baru tidak bisa digunakan secara cepat di sebagian *browser*.
4. Terkadang perilaku *browser* susah untuk diprediksi (misalnya, *browser* lama tidak selalu bisa render tag yang lebih baru).[7]

2.2 PENGENALAN BAHASA PEMROGRAMAN *PHP*

2.2.1 Pengenalan *PHP*

PHP adalah singkatan dari “*PHP: Hypertext Preprocessor*”, yaitu bahasa pemrograman disisi server yang digunakan secara luas untuk penanganan

pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan *HTML*. Ketika Anda mengakses sebuah *URL*, maka web browser akan melakukan request ke sebuah web server [7].

PHP adalah bahasa pemrograman yang sering disisipkan ke dalam *HTML*. Sejarah *PHP* pada awalnya merupakan kependekan dari Personal Home Page (Situs personal). *PHP* pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu *PHP* masih bernama *Form Interpreted (FI)*, yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Bahasa pemrograman *PHP* biasanya tidak digunakan pada keseluruhan pengembangan *website*, melainkan dikombinasikan dengan beberapa bahasa pemrograman lain. Misalnya saja untuk mengatur tampilan, *layout*, dan berbagai macam menu menggunakan *CSS*. Selain itu, terdapat juga beberapa *framework PHP*; *Laravel*, *Phalcon*, *Codigniter*, *Symfoni* yang saat ini banyak tersedia di internet untuk memudahkan proses pengembangan *website* menggunakan bahasa pemrograman tersebut. Bahasa pemrograman ini menggunakan sistem *server-side*. *Server-side programming* adalah jenis bahasa pemrograman yang nantinya *script/program* tersebut akan dijalankan/diproses oleh *server*. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari.

Di bawah ini adalah contoh penggunaan *PHP* untuk menghubungkan dan membuat database.[7]

```
1 <?php
2 $nameserver = "localhost";
3 $username = "username";
4 $password = "password";
5 //Create connection
6 $conn = new mysqli($nameserver, $username, $password);
7 //Check connection
8 if ($kon->connect_error){
9     die("Koneksi gagal:".$kon->connect_error);
10 }
11 //Create database
12 $sql = "CREATE DATABASE myDB";
13 if ($kon->query($sql) === TRUE){
14     echo "Database telah berhasil dibuat";
15 } else {
16     echo "Error, tidak dapat membuat database:".$kon->error;
```

```
17 }  
18 $conn->close();  
19 ?>
```

Pada setiap bahasa pemrograman, terdapat yang namanya *variabel*. *variabel* adalah suatu lokasi penyimpanan (di dalam memori komputer) yang berisikan nilai atau informasi yang nilainya tidak diketahui maupun telah diketahui. Nilai – nilai yang terdapat pada *variabel* ini memiliki berbagai jenisnya. Pada Bahasa pemrograman pun nilai – nilai ini biasanya disebut sebagai data – data. Pada *PHP* ini, ada beberapa tipe data yang dapat digunakan antara lain tipe data berbentuk *Integer*, *Float*, *String*, *Boolean*, *Array*, *Object*, *NULL*.

Pada bahasa pemrograman lainnya, tipe data – tipe data tersebut haruslah dituliskan berdasarkan format bahasa pemrograman itu sendiri. Pada *PHP*, penulisan tipe data – tipe data tersebut sangat mudah sekali, hanya perlu membuat sebuah *variabel* dan menambahkan tanda dolar (\$) di depannya. Berikut pengertian dan contoh penggunaan tipe data tersebut pada bahasa pemrograman *PHP* :

1. Berikut contoh *variabel* tipe data *Integer*
: `$x = 20;`
2. Berikut contoh *variabel* tipe data *Float*
: `$x = 20.1547521;`
3. Berikut contoh *variabel* tipe data *String*
: `$x = "Saya";`
4. Berikut contoh *variabel* tipe data *Boolean*
: `$x = true; | $x = false;`
5. Berikut contoh *variabel* tipe data *Array*
: `$x = array("saya", "suka", "kamu");`
6. Berikut contoh *variabel* tipe data *Object*
: `$x = new namaClass();`
7. Berikut contoh *variabel* tipe data *NULL*
: `$x = null;`

Selain itu, pada *PHP* terdapat juga yang namanya Constans. Constans ini merupakan sebuah tanda pengenalan untuk sebuah nilai yang simple. Nilai ini tidak dapat diubah. Untuk membuat Constans ini dapat menggunakan syntax “define(nama variabel, nilai variabel, case-sensitive)”. Case-sensitive disini memberitahukan apakah Constans itu harus sama persis atau tidak. Kemudian untuk memunculkan variabel yang telah di berikan nilai, dapat menggunakan syntax “echo” dan “print” untuk memunculkannya pada halaman website. Contoh penggunaannya adalah “ echo \$x; ” dan “ print \$x; ”. Tetapi banyak para *programmer* lebih memilih menggunakan echo dalam memunculkan atau menampilkan sesuatu pada bahasa pemrograman *PHP* ini. [7]

Pada *PHP* ini sendiri memiliki beberapa Operators. Operators ini berguna untuk membantu dalam pengoperasian sebuah variabel atau nilai. Berikut beberapa contoh operator pada *PHP* :

2.2.2 Pengertian *PHP* Menurut Beberapa Ahli

1. Menurut Arief (2011c:43)

PHP adalah Bahasa server-side –scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena *PHP* merupakan *server-side-scripting* maka sintaks dan perintah-perintah *PHP* akan diesksekusi diserver kemudian hasilnya akan dikirimkan ke browser dengan format *HTML*. Dengan demikian kode program yang ditulis dalam *PHP* tidak akan terlihat oleh user sehingga keamanan halaman web lebih terjamin. *PHP* dirancang untuk membuat halaman web yang dinamis, yaitu halaman web yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman web.

2. Sedangkan menurut Nugroho (2006 b:61)

“*PHP* atau singkatan dari Personal Home Page merupakan bahasa skrip yang tertanam dalam *HTML* untuk dieksekusi bersifat server side”. *PHP* termasuk dalam open source product, sehingga source code *PHP* dapat diubah dan didistribusikan secara bebas. Versi terbaru *PHP* dapat diunduh secara gratis melalui situs resmi *PHP* : <http://www.php.net>.

PHP juga dapat berjalan pada berbagai web server seperti *IIS (Internet Information Server)*, *PWS (Personal Web Server)*, *Apache*, *Xitami*. *PHP* juga mampu berjalan di banyak sistem operasi yang beredar saat ini, diantaranya : Sistem Operasi Microsoft Windows (semua versi), Linux,

Mac Os, Solaris. *PHP* dapat dibangun sebagai modul web server Apache dan sebagai binary yang dapat berjalan sebagai CGI (*Common Gateway Interface*). *PHP* dapat mengirim HTTP header, dapat mengatur cookies, mengatur authentication dan redirect user.

Salah satu keunggulan yang dimiliki *PHP* adalah kemampuannya untuk melakukan koneksi ke berbagai macam *software* sistem manajemen basis data atau Database Management Sistem (*DBMS*), sehingga dapat menciptakan suatu halaman web dinamis.

PHP mempunyai koneksitas yang baik dengan beberapa *DBMS* seperti Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgreSQL, Adabas, FilePro, Velocis, dBase, Unix dbm, dan tidak terkecuali semua database ber- interface ODBC.

Hampir seluruh aplikasi berbasis web dapat dibuat dengan *PHP*. Namun kekuatan utama adalah konektivitas basis data dengan web. Dengan kemampuan ini kita akan mempunyai suatu sistem basis data yang dapat diakses.[7]

2.2.3 Sejarah *PHP*

Pada awalnya *PHP* merupakan kependekan dari Personal Home Page (Situs personal). *PHP* pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu *PHP* masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya *PHP/FI*. Dengan perilsan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan *PHP*.

Pada November 1997, dirilis *PHP/FI* 2.0. Pada rilis ini, interpreter *PHP* sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan *PHP/FI* secara signifikan. Pada tahun 1997, sebuah perusahaan bernama *Zend* menulis ulang interpreter *PHP* menjadi lebih bersih, lebih baik, dan lebih cepat.

Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk *PHP* dan meresmikan rilis tersebut sebagai *PHP* 3.0 dan singkatan *PHP* diubah menjadi akronim berulang *PHP*: Hypertext Preprocessing. Pada pertengahan

tahun 1999, Zend merilis interpreter *PHP* baru dan rilis tersebut dikenal dengan *PHP* 4.0. *PHP* 4.0 adalah versi *PHP* yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis *PHP* 5.0. Dalam versi ini, inti dari interpreter *PHP* mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam *PHP* untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode *PHP* tanpa menginstall software server. Versi terbaru dan stabil dari bahasa pemrograman *PHP* saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.[7]

2.2.4 Fungsi *PHP*

Bahasa pemrograman *PHP* merupakan bahasa pemrograman yang kebanyakan digunakan untuk mengembangkan website. Anda dapat membuka artikel kami mengenai bahasa pemrograman untuk membuat website. Jadi sebenarnya untuk membuat sistem berbasis web tidak harus menggunakan *PHP*. Namun karena fiturnya yang menarik dan dirasa memudahkan, maka kebanyakan pengembangan menggunakan bahasa pemrograman ini.

Sebelum ada *PHP*, pengembang kebanyakan menggunakan bahasa pemrograman HTML. Bahasa pemrograman ini sudah dapat membuat tampilan website, akan tetapi tampilan yang dihasilkan masih biasa saja. Biasa saja karena website masih bersifat statis atau tetap, tidak bisa menyesuaikan tampilan dengan kondisi pengguna. Semenjak adanya *PHP* maka pembuatan website dapat dikembangkan dan diatur agar menjadi website yang dinamis. Dimana website dapat menyesuaikan tampilan sesuai dengan perangkat yang digunakan oleh pengguna. Tidak hanya itu, website juga dapat melakukan input output dengan database. *PHP* sendiri biasanya disisipkan ke dalam kode *HTML* dan dikombinasikan dengan bahasa pemrograman lain, seperti *CSS*.[7]

2.2.5 Operator

Ada tiga jenis operator utama yang dikenal dalam PHP, antara lain:

1. Operator unary, dimana operator mengoperasikan hanya satu nilai variable, contohnya operator lawan/negasi (“!”), operator penambahan (increment, “++”), dan operator pengurangan (decrement, “-”).
2. Operator binary, terdiri dari beberapa operator yang mempertimbangkan urutan penyelesaian. Kelompok ini paling sering digunakan dalam PHP.
3. Operator ternary, operator yang membandingkan dua pernyataan, dimana kebenarannya ditentukan oleh ekspresi ketiga.

Operator memiliki beberapa urutan (hirarki) eksekusi, mana operator yang harus didahulukan dan mana operator yang dieksekusi selanjutnya. Urutan operator perlu diperhatikan jika ada ekspresi yang menggunakan lebih dari satu operator.[7] Evaluasi urutan operator bisa dimulai dari kiri atau dari kanan, istilah ini disebut juga dengan Associativity.

Associativity	Operator	Keterangan
Non - associative	new	new
Kiri	[Array ()
Non - associative	++ --	Penambahan/pengurangan
Non - associative	~ - (int) (float) (string) (array) (object) @	Type
Non - associative	Instanceof	Type
Kanan	!	Logika
Kiri	* / %	Aritmatika
Kiri	+ - .	Aritmatika dan string
Kiri	<< >>	Bitwise
Non - associative	< <= > >=	Perbandingan
Non - associative	== != === !==	Perbandingan
Kiri	&	Bitwise dan referensi
Kiri	^	Bitwise
Kiri		Bitwise
Kiri	&&	Logika
Kiri		Logika
Kiri	? :	Ternary
Kanan	+ += - - * = /= . =	Pemberian nilai
Kiri	and	logika
Kiri	xor	Logika
Kiri	or	Logika

Figure 2.2: Hirarki Operator

Asosiatif kiri berarti ekspresi dievaluasi dari kiri ke kanan, dan sebaliknya asosiatif kanan berarti ekspresi dievaluasi dari kanan ke kiri.

```

1 <?php
2 $a = 3 * 3 % 5; // (3 * 3) % 5 = 4
3 $a = true ? 0 : true ? 1 : 2;
4 // (true ? 0 : true) ? 1 : 2 = 2
5 $a = 1;
6 $b = 2;
7 $a = $b += 3; // $a=($b+=3) -> $a=5, $b=5
8 ?>

```

2.2.5.1 Operator Aritmatika

Aturan penulisan aritmatika adalah variable = ekspresi aritmatika, ini berarti bahwa variable akan bernilai sesuai hasil yang diberikan dari ekspresi aritmatika

yang dilakukan. Disisi sebelah kiri hanya boleh ada satu variable tunggal saja dan tidak boleh ada yang lain, sedangkan sebelah kanan bisa berupa ekspresi/rumus aritmatika, konstanta atau variable. [7] Operasi aritmatika yang umum digunakan dalam pemrograman, antara lain:

Operator	Keterangan
+	Tambah
-	Kurang
/	Bagi
*	Kali
%	Sisa bagi

Figure 2.3: Operasi Aritmatika

```
1 <?php
2 // Contoh penulisan operasi aritmatika yang benar
3 $var_1 = 8;
4 $var_2 = $var_1;
5 echo $var_2;
6 $panjang = 10;
7 $lebar = 5;
8 $luas = $panjang * $lebar;
9 echo $luas;
10 // Contoh penulisan operasi aritmatika yang salah
11 8 = $var_1;
12 4 + 7 = $var_2;
13 $panjang * $lebar = $luas;
14 ?>
```

2.2.5.2 Hirarki Operator Artimatika

Seringkali ekspresi aritmatika memerlukan beberapa operator yang berbeda, sehingga kita harus tahu urutan penyelesaian dari masing-masing operator, agar nilai yang dihasilkan benar sesuai dengan yang diharapkan.[7]

Hirarki	Operator	Keterangan
1	* / % ()	Tergantung pada posisinya, urutan dari kiri didahulukan kemudian sebelah kanan
2	+ atau -	Tergantung pada posisinya, urutan dari kiri didahulukan kemudian sebelah kanan.

Figure 2.4: Hirarki Operator Aritmatika

```
1 <?php
2 $nilai1 = 12 + 4 * 3 / 6;
3 echo $nilai1;
4 // 12 + 12 / 6
5 // 12 + 2
6 // hasilnya 14
7 $nilai2 = 210 / ( 6 * ( 4 + 5      2 ) )
8 echo $nilai2;
9 // 210 / ( 6 * 7 )
10 // 210 / 42
11 // hasilnya 5
12 ?>
```

2.2.5.3 Operator Pemberi Nilai

Operator pemberi nilai diwakili oleh tanda sama dengan (“=”), yang berarti bahwa operan sebelah kiri akan diberi nilai dengan ekspresi disebelah kanan.

```
1 <?php
2 $a = ($b = 4) + 5;
3 // $a bernilai 9 dan $b bernilai 4.
4 ?>
```

Operator pemberi nilai bisa juga dikembangkan lagi menjadi operator kombinasi, yang bertujuan untuk menyederhanakan sintaks. Tambahan operator tersebut berguna saat kita mencari nilai Total suatu nilai, dan metode yang digunakan adalah dengan cara perulangan (looping).[7]

```
1 <?php
2 $total = 0;
3 while ( $total < 10 )
4 {
5 $total = $total + 1;
```

```
6 }
7 echo $total;
8 // akan sama artinya dengan berikut ini
9 $total = 0;
10 while ( $total < 10 )
11 {
12 $total += 1;
13 }
14 echo $total;
15 ?>
```

Operator kombinasi juga bisa dikenakan pada penggabungan tipe string

```
1 <?php
2 $a = 3;
3 $a += 5;
4 // $a bernilai 8, atau bisa dituliskan: $a = $a + 5;
5 $b = "Hello ";
6 $b .= "Tiffany!";
7 // $b bernilai "Hello Tiffany!", atau bisa dituliskan: $b = $b .
8 "Tiffany!";
9 ?>
```

2.2.5.4 Operator Bitwise

Operator bitwise adalah operator yang bertujuan untuk mengoperasikan bilangan biner (angka 0 dan 1). Jika bilangan operan berupa bilangan desimal maka harus dikonversikan terlebih dahulu menjadi bilangan biner, baru kemudian dioperasikan dengan operator bitwise. Jika operan kedua-duanya berupa string maka nilai operan harus dikonversikan terlebih dahulu menjadi nilai ASCII (sesuai table ASCII), baru kemudian dioperasikan dengan operator bitwise.[7]

Contoh	Nama	Hasil
<code>\$a & \$b</code>	And	Bit di set 1 jika kedua-duanya yaitu \$a dan \$b bernilai 1
<code>\$a \$b</code>	Or	Bit di set satu jika nilai salah satu dari \$a atau \$b bernilai 1
<code>\$a ^ \$b</code>	Xor	Bit di set satu jika nilai \$a dan \$b memiliki perbedaan
<code>~ \$a</code>	Not	Bit akan di set 1 jika \$a bernilai 0, dan di set 1 jika \$a dinilai !
<code>\$a << \$b</code>	\$b digeser ke kiri	Menggeser \$a sebanyak \$b langkah ke kiri (disetiap langkah “dikalikan dengan 2”)
<code>\$a >> \$b</code>	\$b digeser ke kanan	Menggeser bit \$a sebanyak \$b langkah ke kanan (disetiap langkah berarti “dibagi dengan dua”)

Figure 2.5: Operator Bitwise

```

1 <?php
2 $a = 8; // nilai dalam bit : 1000
3 $b = 9; // nilai dalam bit : 1001
4 $ab_and = $a & $b;
5 echo $ab_and; // output 8, dalam biner: 1000
6 $ab_or = $a | $b;
7 echo $ab_or; // output 9, dalam biner: 1001
8 $ab_xor = $a ^ $b;
9 echo $ab_xor; // output 1, dalam biner 0001 atau 1
10 $ab_not = ~$a;
11 echo $ab_not; // output 7, dalam biner 0111 atau 111
12 ?>
13 <?php
14 echo "12" ^ "9";
15 // Output adalah Character Backspace (ascii 8)
16 // ('1' (ascii 49)) ^ ('9' (ascii 57)) = #8
17 echo "hallo" ^ "hello";
18 // Output adalah ascii #0 #4 #0 #0 #0
19 // 'a' ^ 'e' = #4
20 ?>

```

2.2.5.5 Operator Perbandingan

Operator perbandingan adalah untuk membandingkan dua nilai. Perlu diperhatikan bahwa operator “=” tidaklah sama dengan operator “==”, jika operator “==” merupakan operator perbandingan dan akan menghasilkan nilai benar atau salah, sedangkan operator “=” merupakan operator pemberi nilai. Jika tidak berhati-hati dalam penggunaannya, maka akan menyebabkan kesalahan program yang cukup sulit untuk dideteksi, karena secara sintaks *PHP*

menganggap hal tersebut sudah benar, namun secara logika akan menghasilkan nilai yang salah.[7]

Contoh	Nama	Hasil
\$a == \$b	Sama dengan	Benar jika \$a sama dengan \$b
\$ === \$	Identik	Benar jika \$a sama dengan \$b, dan keduanya memiliki kesamaan type
\$!= \$b	Tidak sama	Benar jika \$a tidak sama dengan \$b
\$ <> \$b	Tidak sama	Benar jika \$a tidak sama dengan \$b
\$a !==	Tidak identik	Benar jika \$a tidak sama dengan \$b, dan keduanya tidak memiliki kesamaan type
\$a < \$b	Kurang dari	Benar jika \$a kurang dari \$b
\$a > \$b	Lebih dari	Benar jika \$a lebih besar dari \$b
\$a <= \$b	Kurang dari atau sama dengan	Benar jika \$a kurang dari atau sama dengan \$b
\$a >= \$b	Lebih dari atau sama dengan	Benar jika \$a lebih dari atau sama dengan \$b

Figure 2.6: Operator Perbandingan

Jika membandingkan antara integer dengan string, maka string akan dikonversi ke angka terlebih dahulu. Jika membandingkan dua angka string, maka keduanya-duanya akan dikonversikan menjadi integer

Tipe Operan 1	Tipe Operan 2	Hasil
Null atau string	string	Konversi dari null ke "", perbandingan numerik atau lexical
Bool atau null	anything	Konversi ke bool, FALSE < TRUE
String, resource atau number	String, resource atau number	Menterjemahkan string dan resource ke angka
Array	Array	Array dengan anggota yang terkecil, jika key dari operan 1 tidak ditemukan dalam operan 2 maka array tidak bisa dibandingkan
Array	Anything	Array selalu lebih tinggi
Object	Anything	Object selalu lebih tinggi

Figure 2.7: Perbandingan dengan tipe yang berbeda

```
1 <?php
2 $x="1"; //string dengan nilai angka 1
3 $y=1; //integer dengan nilai angka 1
4 echo "test == :".($x == $y);
5 // TRUE
6 echo "<br>test === :".($x === $y);
7 // FALSE, tidak identik
8 ?>
```

2.2.5.6 Operator Kontrol Kesalahan (*Error*)

PHP mendukung satu operator kontrol kesalahan yaitu tanda (“@”). Ketika operator kontrol kesalahan dikenakan pada ekspresi, bisa jadi jika ada kesalahan maka kesalahan tersebut akan diabaikan. Jika fitur `track_errors` diaktifkan (ada file `php.ini`), maka pesan kesalahan akan bisa ditampilkan, ekspresi kesalahan disimpan dalam *variable* `$php_errormsg`. *Variable* ini akan selalu ditimpa setiap kali ada kesalahan.[7]

```
1 <?php
2 /* Kesalahan pada file */
3 $my_file = @file ( 'non_existent_file' ) or
4   die ( "File gagal dibuka: error was '$php_errormsg' " );
5 // tidak hanya fungsi tapi juga bekerja ekspresi
6 $value = @ $cache [ $key ];
7 ?>
```

2.2.5.7 Operator Eksekusi

PHP mendukung satu operator eksekusi yaitu dengan menggunakan backticks (“`”). PHP akan mengeksekusi baris kode yang berada diantara backticks seperti mengeksekusi perintah shell (*shell command*). Penggunaan operator backtick identik dengan fungsi `shell_exec()`. [7]

```
1 <?php
2 // Tampilkan direktori dan file di UNIX/LINUX
3 $output = `ls -al`;
4 echo "<pre>$output</pre>";
5 // Tampilkan direktori dan file di Windows
6 $output = `dir`;
7 echo "<pre>$output</pre>";
8 ?>
```

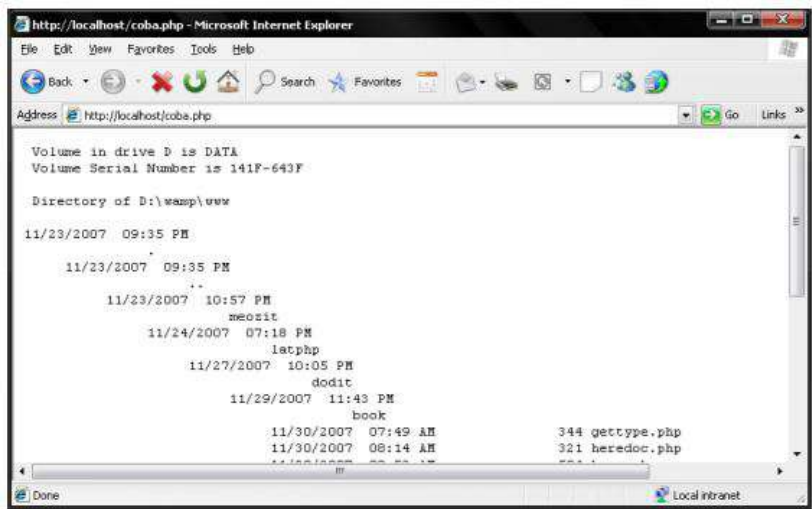



Figure 2.8: Operator Eksekusi

2.2.5.8 Operator Penambahan dan Pengurangan

Operator aritmatika khusus lainnya adalah penambahan (increment, ++) dan pengurangan (decrement, -). Operator increment adalah operator untuk menambah nilai variable dengan satu, sebaliknya decrement adalah pengurangan nilai variabel dengan satu. Peletakkan operator ++ atau - bisa sebelum atau sesudah variable, tergantung kebutuhan.

Operator	Keterangan	Contoh	Persamaan
++\$variable	Penjumlahan dengan 1	\$total += 5	\$total = \$total + 5
\$variable++	Penjumlahan dengan 1	\$total -= 5	\$total = \$total - 5
-- \$variable	Pengurangan dengan 1	\$total *= 5	\$total = \$total * 5
\$variable --	Pengurangan dengan 1	\$total /= 5	\$total = \$total / 5

Figure 2.9: Operator Penambahan dan Pengurangan

```
1 <?php
2 $nilai_1 = 100;
3 echo $nilai_1 . "<br>"; // 100
4 echo ++$nilai_1 . "<br>"; // 101
5 echo $nilai_1 . "<br>"; // 101
6 $nilai_2 = 100;
7 echo $nilai_2 . "<br>"; // 100
8 echo $nilai_2++ . "<br>"; //100
9 echo $nilai_2 . "<br>"; // 101
10 $nilai_3 = 100;
11 echo $nilai_3 . "<br>"; // 100
12 echo --$nilai_3 . "<br>"; // 99
13 echo $nilai_3 . "<br>"; // 99
14 $nilai_4 = 100;
15 echo $nilai_4 . "<br>"; //100
16 echo $nilai_4-- . "<br>"; //100
17 echo $nilai_4 . "<br>"; //99
18 ?>
```

2.2.5.9 Operator Logika

Beberapa nilai ekspresi hasil dari operator perbandingan bisa dihubungkan dengan beberapa ekspresi yang lain agar diperoleh perbandingan dan nilai logika baru. Untuk menghubungkannya maka diperlukan operator logika.

Contoh	Nama	Hasil
\$a and \$b	And	TRUE jika keduanya : \$a dan adalah \$b TRUE
\$a or \$b	Or	TRUE jika \$a dan adalah \$b TRUE
\$a xor \$b	Xor	TRUE jika \$a dan adalah \$b TRUEtapi tidak keduanya
! \$a	Not	TRUE jika \$a bukan TRUE
\$a && \$b	And	TRUE jika keduanya : \$a dan \$b adalah TRUE
\$a \$b	Or	TRUE, jika \$a atau \$b adalah TRUE

Figure 2.10: Operator Logika

2.2.5.10 Operator String

Ada dua jenis operator string. Antara lain :

- 1. concatenation (“.”), dimana string sebelah kanan digabungkan dengan string sebelah kiri yang akan menghasilkan string baru hasil dari penggabungan.
- 2. concatenation dengan operator pemberi nilai (“.= ”), dimana string di sebelah kanan digabungkan dengan string sebelah kiri yang hanya membutuhkan satu variable penampung string saja.

```
1 <?php
2 $a = "Hello ";
3 $b = $a . "World!"; // sekarang $b berisi "Hello World!"
4 $a = "Hello ";
5 $a .= "World!"; // sekarang $a berisi "Hello World!"
6 ?>
```

2.2.5.11 Operator Array

Contoh	Nama	Hasil
\$a + \$b	Penggabungan	Gabungan dri \$a dan \$b
\$a == \$b	Sama Dengan	TRUE jika \$a dan \$b meiliki kesamaan key/value
\$a === \$b	Identik	TRUE jika \$a dan \$b meiliki kesamaan key/value termasuk kesamaa jenis tipenya identik
\$a != \$b	Tidak sama dengan	TRUE jika \$a tidak sama dengan \$b
\$a <> \$b	Tidak sama dengan	TRUE jika \$a tidak sama dengan \$b
\$a !==	Tidak identik	TRUE jika \$a tidak identik dengan \$b

Figure 2.11: Operator Array

Operator “+” adalah menggabungkan array sebelah kanan ke array sebelah kiri, jika ada key yang sama maka key tidak akan ditimpa.

```
1 <?php
2 $a = array( "a" => "apple", "b" => "banana" );
3 $b = array( "a" => "pear", "b" => "strawberry", "c" => "cherry" );
```

```

4 $c = $a + $b; // Menggabungkan $a dan $b
5 echo "Hasil penggabungan dari \$a and \$b: \n";
6 var_dump($c);
7 $c = $b + $a; // Menggabungkan $b dan $a
8 echo "Hasil penggabungan dari \$b dan \$a: \n";
9 var_dump($c);
10 /* Hasil Penggabungan adalah :
11 Penggabungan dari $a dan $b:
12 array(3) {
13     ["a"]=>
14     string(5) "apple"
15     ["b"]=>
16     string(6) "banana"
17     ["c"]=>
18     string(6) "cherry"
19 }
20 Penggabungan dari $b dan $a:
21 array(3) {
22     ["a"]=>
23     string(4) "pear"
24     ["b"]=>
25     string(10) "strawberry"
26     ["c"]=>
27     string(6) "cherry"
28 }
29 */
30 ?>

```

Contoh perbandingan array :

```

1 <?php
2 $a = array("apple", "banana");
3 $b = array(1 => "banana", "0" => "apple");
4 var_dump($a == $b); // bool(true)
5 var_dump($a === $b); // bool(false) karena tidak identik, tidak
6 setipe
7 ?>

```

1. Tipe Array

Array adalah sekumpulan data yang disimpan dalam suatu variable dengan nama yang sama, dan untuk membedakan antara data satu dengan data yang lain digunakan index atau keys. Sedangkan masing masing data dalam *array* disebut dengan element.

Array dalam *PHP* sedikit berbeda dengan *array* pada bahasa pemrograman yang lain. Pada umumnya setiap element *array* harus memiliki kesamaan tipe, tetapi hal tersebut tidak berlaku bagi *PHP*. Di dalam *PHP* diperbolehkan adanya perbedaan jenis tipe data di setiap element dalam *array*, karena perlu diingat bahwa *PHP* tidak memerlukan pendeklarasian variable, termasuk variable *array*. Dengan kata lain, masing-masing element *array* bisa memiliki berbagai macam jenis tipe data yang tidak sejenis dalam satu nama variable *array*.

(a) Menciptakan Array Dengan "Array()"

Array bisa diciptakan dengan pernyataan `array()` yang disertai dengan "key=>value" dan diletakkan diantara tanda kurung `array()`. Jika jumlah "key=>value" lebih dari satu, maka disetiap "key=>value" harus pisahkan dengan tanda koma (",").

Adapun strukturnya adalah sebagai berikut :

```
1 array( key => value , ... )
```

"key" bisa bertipe integer atau string, namun jika key bertipe float maka key akan dipenggal menjadi integer. Sedangkan "value" bisa bermacam-macam tipe walaupun satu sama lain tidak se-tipe.

```
1 <?php $arr = array( "foo" => "bar", 12 => true );
2 echo $arr[ "foo" ]; // bar echo $arr[12]; // 1 ?>
```

Array multidimensi juga dimungkinkan penggunaannya dalam PHP. Untuk lebih jelasnya perhatikan contoh berikut ini:

```
1 <?php $arr = array ( "somearray1" => array(6 => 5,
2 13 => 9, "a" => 42), "somearray2" => array(7 => 6,
3 14 => 10, "b" => 43), "somearray3" => array(8 => 7,
4 15 => 11, "c" => 44), ); echo $arr[ "somearray1" ][6];
5 // 5 echo $arr[ "somearray1" ][13]; // 9 echo
6 $arr[ "somearray1" ][ "a" ]; // 42 ?>
```

Jika "key" tidak ditentukan maka key integer terbesar akan diambil, selanjutnya key baru diperoleh dari key integer terbesar + 1. Jika key yang kita tentukan sudah ada dalam key array tersebut, maka key yang telah ada akan ditimpa.

```
1 <?php // Array ini sama artinya... array(5 => 43, 32,
2 56, "b" => 12);
3
4 // ...dengan array ini array(5 => 43, 6 => 32, 7 => 56, "
  b"
```

```

5 => 12);
6 ?>

```

(b) Memodifikasi Array Dengan Tanda Siku ("[]")

Kita bisa menambah atau membuang suatu element dalam array.

Untuk menambahkan suatu nilai/value ke dalam array gunakan square-bracket ("[]") sebagai element array penampung value. Square-bracket ("[]") bisa memiliki "key" ataupun tidak, jika tanpa disertai "key" maka key akan tercipta dengan sendirinya.

Jika ingin membuang element array (pasangan key/value), maka lakukan perintah unset() terhadap array tersebut.

```

1 <?php $arr = array(5 => 1, 12 => 2);
2
3 $arr[] = 56; // Ini sama dengan $arr[13] = 56;
4 // key diambil dari nilai key terbesar+1
5
6 $arr["x"] = 42; // Menambahkan element baru
7 // ke dalam array dengan key "x"
8 unset($arr[5]); // Membuang element array I key-5
9 unset($arr); // Membuang array seluruhnya
10 ?>

```

Ketika dilakukan perintah unset() terhadap array maka index element array tidak akan ter-reindex (penataan ulang terhadap key index). Untuk mengindex ulang seluruh element array maka gunakan perintah array_values()

```

1 <?php $a = array(1 => 'satu', 2 => 'dua',
2 3 => 'tiga'); unset($a[2]); /* array yang
3 dihasilkan dari modifikasi di atas adalah
4 $a = array(1 => 'satu', 3 => 'tiga');
5 dan BUKAN $a = array(1 => 'satu', 2 => 'tiga'); */
6
7 $b = array_values($a); // Sekarang $b adalah array(0 => '
8 satu',
9 1 => 'tiga')
10 ?>

```

2.2.6 Struktur Kontrol

Script *PHP* dibangun oleh serangkaian pernyataan, dimana pernyataan tersebut bisa berupa pemberi nilai (*assignment*), pemanggil fungsi, perulangan (*loop*),

pernyataan kondisi, atau apapun termasuk pernyataan kosong. Dan pernyataan-pernyataan tersebut bisa diatur alur kerjanya dengan adanya stuktur kontrol. Agar lebih jelas, berikut ini akan dijelaskan jenis struktur kontrol yang ada dalam *PHP* :

2.2.6.1 If

Pernyataan *If* merupakan percabangan bersyarat yang berfungsi untuk melewati suatu proses, jika syarat terpenuhi maka akan dilakukan proses yang berikutnya. Proses bisa berupa satu instruksi atau beberapa instruksi dalam satu kelompok.

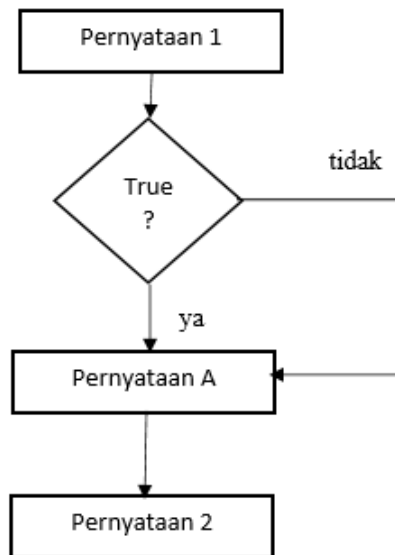


Figure 2.12: Struktur IF

Struktur if satu instruksi : If (syarat) instruksi;

```
1 <?php
2 $panjang = 30;
3 $lebar = 80;
4 $luas = $panjang * $lebar;
5 $maks = 100;
```

```
6 if ( $luas > $maks)
7 echo "Luas lebih dari $maks";
8 ?>
```

Struktur if dengan sekumpulan instruksi:

```
1 If (syarat)
2 {
3 instruksi;
4 instruksi;
5 instruksi;
6 }
7 <?php
8 // Nilai $panjang dan $lebar bisa diganti,
9 // agar lebih mudah memahami.
10 $panjang = 30;
11 $lebar = 80;
12 $luas = $panjang * $lebar;
13 $maks = 100;
14 if ( $luas > $maks)
15 {
16 echo "Panjang = $panjang <br>";
17 echo "Lebar = $lebar <br>";
18 echo "Luas Yang Dihasilkan = $luas <br>";
19 echo "Luas Maksimal = $maks <br>";
20 echo "Luas lebih dari $maks";
21 }
22 ?>
```

2.2.6.2 If Else

Struktur if else memiliki dua alur percabangan, sehingga mempunyai alternatif. If else bisa menuju ke proses berikutnya walaupun syarat terpenuhi atau tidak terpenuhi.

Struktur If Else dengan satu instruksi :

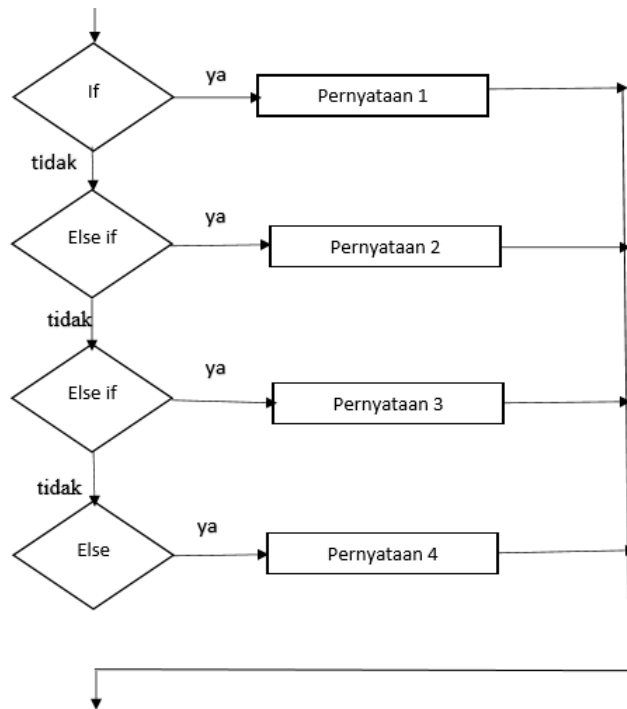


Figure 2.13: if else

If (syarat) Instruksi; Else Instruksi;

Contoh:

```

1 <?php $angka = 4; $sisia = $angka % 2;
2 // mencari sisa bagi if ($sisia == 0) echo
3 "$angka merupakan bilangan genap"; else echo
4 "$angka merupakan bilangan ganjil"; ?>
5
6
7 Struktur If Else dengan sekumpulan instruksi:
8
9 If (syarat) { instruksi; instruksi; }
10 else { instruksi; instruksi; }
11
12 <?php $angka = 4; $sisia = $angka % 2;
13 // mencari sisa bagi if ($sisia == 0) echo
14 "$angka merupakan bilangan genap"; else echo
15 "$angka merupakan bilangan ganjil"; ?>

```

2.2.6.3 If, Elseif, Else

Pernyataan if else if else merupakan pengembangan dari pernyataan if else namun memiliki percabangan lebih dari dua. if else if else sesuai untuk memecahkan masalah yang membutuhkan banyak percabangan karena banyak alternatif yang bisa diperoleh. Pernyataan if else if else boleh dikatakan juga sebagai if di dalam if.

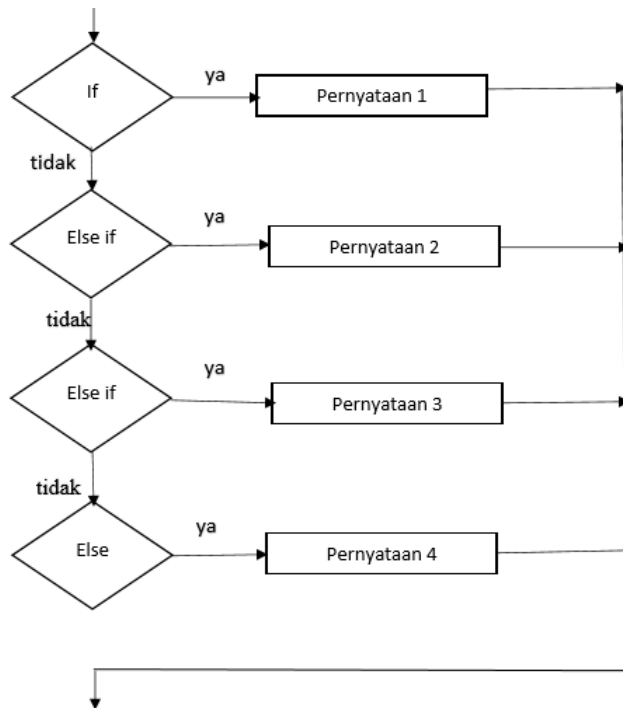


Figure 2.14: if elseif else

Struktur if else if else adalah:

```

1 If (syarat) { instruksi; instruksi; } elseif (syarat)
2 { instruksi; instruksi; } elseif (syarat)
3 { instruksi; instruksi; } else { instruksi;
4 instruksi; }
5
6 <?php $a = 9; // silahkan dirubah nilainya untuk mengetahui
7 hasil yang diperoleh $b = 3; if ($a > $b)
8 { echo "a is lebih besar dari b"; }
  
```

```

9 elseif ($a == $b) { echo "a sama dengan b"; }
10 else { echo "a lebih kecil dari b"; } ?>

```

2.2.6.4 Operator Kondisi Ternary

Untuk menyederhanakan pernyataan if ... else, PHP memiliki operator kondisi ternary yang cukup singkat. Dengan struktur sebagai berikut :

```

1 variable = ekspresi logik atau relasi ? ekspresi B:ekspresi S

```

Penjelasan struktur Operator Kondisi :

1. Variable, variable yang akan berisi nilai ekspresi 1 atau ekspresi 2.
2. Ekspresi logik atau relasi, merupakan ekspresi yang dijalankan untuk memeriksa kondisi, sehingga menghasilkan nilai benar atau salah.
3. Ekspresi B, akan dijalankan jika ekspresi logika atau relasi bernilai benar (true).
4. Ekspresi S, akan dijalankan jika ekspresi logika atau relasi bernilai salah (false)

```

1 <?php $umur = 5; // bagaimana jika $umur lebih dari 5?
2
3 If ($umur <= 5) $usia = "Balita"; else $usia =
4 "Remaja atau dewasa"; echo $usia . "<br>";
5 // Script di atas dapat dituliskan seperti berikut ini
6 : $usia = $umur <= 5 ? "Balita" : "Remaja atau dewasa";
7 echo $usia;
8 ?>

```

2.2.6.5 Struktur Kontrol Alternatif

PHP menawarkan sintaks alternatif untuk struktur kontrol if, while, for, foreach, dan switch yang berguna untuk mengelompokkan beberapa instruksi yang akan dieksekusi. Sintaks alternatif tersebut diawali dengan tanda titik dua (":")

dan diakhiri dengan `endif;`, `endwhile;`, `endfor;`, `endforeach;`, atau `endswitch;` sesuai dengan struktur kontrol-nya. Fungsinya mirip dengan penanda pasangan kurawal

```
1 ( { } )
```

yang biasa digunakan untuk mengelompokkan beberapa instruksi.

```
1 <?php if ($a == 5): echo "a sama dengan 5";
2 echo "..."; elseif ($a == 6):
3 echo "a sama dengan 6"; echo "!!!"; else:
4 echo "a bukan 5 ataupun 6"; endif; ?>
```

2.2.6.6 Whlie

Perulangan `while` akan melakukan pengecekan syarat di awal blok perulangan. Selama syarat bernilai benar maka perulangan terus berlanjut dan sebaliknya jika syarat bernilai salah maka perulangan akan dihentikan.

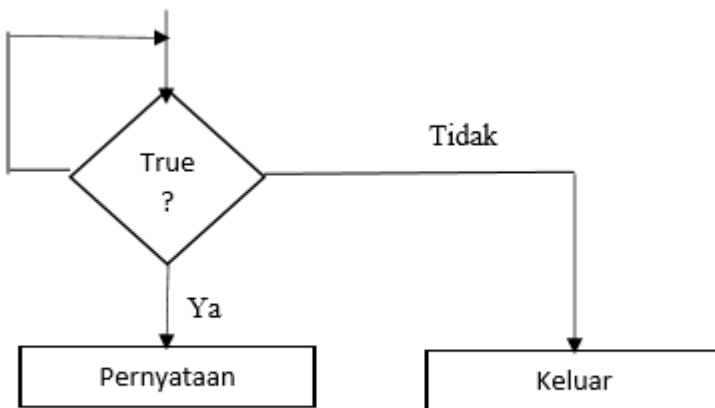


Figure 2.15: While

Struktur While untuk instruksi tunggal :

```
while (syarat) instuksi;
```

Struktur While dengan sekumpulan instruksi :

```
while (syarat)  instuksi; instuksi;
```

atau juga bisa dalam bentuk berikut ini :

```
while (syarat) instuksi; instuksi; endwhile;
```

```
1 <?php /* Contoh 1 */
2
3 $i = 1; while ($i <= 10) { echo $i++;
4 //12345678910 } echo "<br>";
5
6 /* Contoh 2 */
7
8 $i = 1; while ($i <= 10): echo $i;
9 //12345678910 $i++; endwhile;
10 ?>
```

2.2.6.7 Do-While

Perulangan do while merupakan kebalikan dari perulangan for dan while, karena perulangan do while melakukan pengecekan terhadap syarat diakhir blok perulangan. Jadi do while paling sedikit akan melakukan satu kali proses perulangan.

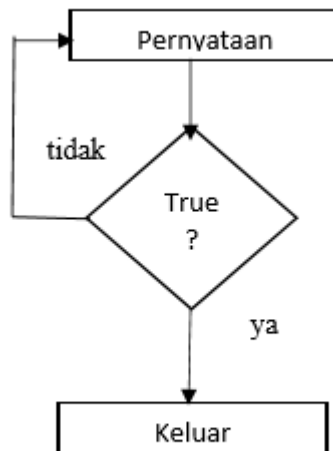


Figure 2.16: Do-While

Struktur do while adalah sebagai berikut:

do instruksi; instruksi; while (syarat)

Berikut contoh mencetak bilangan 1 sampai dengan 10, termasuk nilai totalnya:

```

1 <?php $bil = 1; $total = 0;
2 do {
3   $total += $bil; echo $bil . " " . $total . "<br>"; $bil++; }
4 while ($bil <= 10) /*
5 Hasilnya 1 1 2 3 3 6 4 10 5 15 6 21 7 28 8 36 9 45 10 55 */
6 ?>

```

2.2.6.8 For

Struktur kontrol For lebih cocok untuk perulangan dengan jumlah pencacah yang pasti atau sudah diketahui.

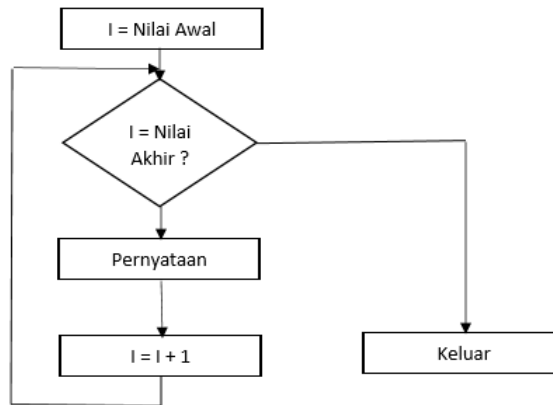


Figure 2.17: For

Struktur For dengan instruksi tunggal :

for (inisialisasi; syarat ; pencacah) instuksi;

Struktur For dengan sekumpulan instruksi :

for (inisialisasi; syarat ; pencacah) instuksi; instuksi;

Penjelasan struktur for :

1. inisialisasi, merupakan nilai awal saat perulangan for dilakukan.
2. syarat, untuk mengevaluasi setiap kali perulangan dilakukan, berhenti tidaknya suatu perulangan ditentukan oleh syarat tersebut. Syarat bisa berupa ekspresi relasional atau ekspresi logika. Jika syarat bernilai benar maka perulangan dilanjutkan dan jika syarat bernilai salah maka perulangan akan dihentikan.
3. pencacah, untuk mengatur perubahan nilai variable pencacah yang nilainya bisa diatur menaik atau menurun sesuai dengan kebutuhan.
4. inisialisasi, syarat dan pencacah dalam perulangan for disebut dengan argumen dan argumen tidak harus diisi lengkap. Meskipun argumen

tidak harus lengkap, tanda titik koma (;) harus tetap dicantumkan di setiap argumen yang kosong.

```
1 <?php // Contoh perulangan for dengan argumen lengkap.
2
3 for ($i=1;$i<=10;$i++)
4 { $total += $i; echo $i . " " . $total . "<BR>"; }
5
6
7 // Contoh perulangan for dengan argumen tidak lengkap.
8 // Argumen syarat yang kosong, tetapi diganti dengan
9 syarat di bawahnya.
10
11 for ($i = 1;;$i++) { if ($i > 10) { break; } echo $i; }
12
13 // Perulangan for tanpa argumen sama sekali,
14 // menyebabkan perulangan dilakukan terus menerus.
15
16 // Perulangan dihentikan secara paksa dengan instruksi break.
17
18 $i = 1; for (;;) { if ($i > 10) { break; } echo $i; $i++; }
19 ?>
```

2.2.6.9 For Each

Perulangan foreach mulai digunakan pada PHP versi 4 ke atas. Perulangan foreach banyak digunakan bersamaan dengan data array, karena untuk mengakses data array akan jauh lebih mudah dan praktis.

Struktur foreach adalah sebagai berikut :

```
1 foreach(array_expression as $value)
2 statement foreach(array_expression as $key => $value) statement
```

Untuk lebih jelasnya, lihat contoh berikut :

Contoh 1: foreach yang hanya menampilkan value saja.

```
1 <?php $a = array (1, 2, 3, 17);
2
3 foreach ($a as $v) { echo "Nilai \ $a saat ini adalah: $v.<br>"; }
4
```



```
5 /* Nilai $a saat ini adalah: 1.  
6 Nilai $a saat ini adalah: 2. Nilai  
7 $a saat ini adalah: 3. Nilai $a saat ini adalah: 17. */  
8 ?>
```

Contoh 2: foreach yang menampilkan value dengan key.

```
1 <?php $a = array (1, 2, 3, 17);  
2  
3 $i = 0;  
4  
5 foreach($a as $v) { echo "\$a[$i] => $v.<br>"; $i++; }  
6  
7 /* Hasilnya $a[0] => 1. $a[1] => 2. $a[2] => 3. $a[3] => 17. */  
8 ?>
```

Contoh 3: foreach yang menampilkan key dan value.

```
1 <?php $a = array ("one" => 1, "two" => 2,  
2 "three" => 3, "seventeen" => 17);  
3  
4 foreach($a as $k => $v) { echo "\$a[$k] => $v.<br>"; }  
5  
6 /* Hasilnya $a[one] => 1. $a[two] => 2.  
7 $a[three] => 3. $a[seventeen] => 17.  
8 /*  
9 ?>
```

Contoh 4: foreach yang menampilkan array multi-dimensi

```
1 <?php $a[0][0] = "a"; $a[0][1] = "b"; $a[1][0] = "y";  
2 $a[1][1] = "z";  
3  
4 foreach($a as $v1) { foreach ($v1 as $v2) { echo "$v2"; } }  
5  
6 // Hasilnya abyz  
7 ?>
```

2.2.6.10 Break

Pada contoh-contoh sebelumnya pernyataan break sudah pernah disinggung sedikit, yaitu bertujuan untuk menghentikan dan keluar dari struktur perulangan.

gan for, foreach, while, do...while atau struktur switch kemudian melanjutkan eksekusi program ke baris perintah berikutnya.

Break bisa ditambahkan dengan argumen angka, yang menunjukkan kelompok struktur seberapa yang akan dihentikan.

Contoh break pada struktur for :

```
1 <?php for ($i=1;$i<=10;$i++) { if ($i==6) break; echo $i . " "; }
2 echo "Akhir pengulangan "; ?>
```

Contoh break yang dikenakan pada struktur while dan switch, dimana break memiliki dua tingkat stuktur :

```
1 <?php $i = 0; while (++$i) { switch ($i)
2 { case 5: echo "Di 5<br>"; break 1;
3 /* break 1, keluar hanya di stuktur switch. */
4 case 10: echo "Di 10, keluar<br>"; break 2;
5 /* break 2, keluar dari struktur switch dan
6 struktur while. */ default: break;
7 }
8 }
9 ?>
```

2.2.6.11 Continue

Pernyataan continue digunakan pada struktur perulangan. Bertujuan untuk melewati suatu tahap perulangan dan melanjutkan kembali ke proses perulangan selanjutnya.

Continue bisa ditambahkan argumen angka yang berarti memberitahukan, struktur seberapa yang akan dihentikan.

```
1 <?php for ($i=1;$i<=10;$i++) { if ($i == 7)
2 continue; echo "&nbsp;"; //1 2 3 4 5 6 8 9 10 } ?>
```

2.2.6.12 Switch

Pernyataan switch hampir sama dengan pernyataan if ... elseif ... else tetapi dengan pendekatan yang berbeda. Keduanya sama-sama yang memiliki banyak percabangan, perbedaannya adalah, variable yang dijadikan syarat harus bernilai pasti dan bukan lagi sebagai ekspresi.

Strukturnya Switch :

```
1 switch ($variable) { case nilai_variable_1:   instruksi;
2 instruksi;   break;
3
4 case nilai_variable_2:   instruksi;   instruksi;   break;
5
6 case nilai_variable_3:   instruksi;   instruksi;   break;
7
8 default:   instruksi;   instruksi; }
9 \end{verbatim}
10 Penjelasan switch : \par
```

1. Switch diawali dengan tanda ““ dan diakhiri dengan tanda ””.
2. Setiap case harus diakhiri dengan break, fungsi break adalah untuk menghentikan proses pencarian jika syarat sudah terpenuhi. Jika tidak menggunakan break maka meskipun nilai sudah ditemukan sesuai syaratnya, proses akan tetap berjalan sampai pada alternatif terakhir dan nilai yang dihasilkan adalah nilai case paling akhir walaupun nilai yang sesuai berada diposisi sebelumnya.
3. Default bertipe pilihan (optional), bisa disertakan boleh juga tidak. Default disertakan jika dari sekian banyak alternatif case tidak satu pun yang memenuhi syarat, maka instruksi default akan dilakukan.

Contoh kemiripan penggunaan if dan switch :

```
1 <?php // Percabangan dengan if if ($i == 0)
2 { echo "i sama dengan 0"; } if ($i == 1)
3 { echo "i sama dengan 1"; } if ($i == 2)
4 { echo "i sama dengan 2"; }
5
6 // Percabangan dengan switch switch ($i)
```

```
7 { case 0: echo "i sama dengan 0"; break;  
8 case 1: echo "i sama dengan 1"; break;  
9 case 2: echo "i sama dengan 2"; break; }  
10 ?>
```

Contoh switch disertai default :

```
1 <?php switch ($i) { case 0: echo "i sama dengan 0";  
2 break; case 1: echo "i sama dengan 1"; break; case 2:  
3 echo "i sama dengan 2"; break; default:  
4 echo "i tidak sama dengan 0, 1 atau 2"; } ?>
```

2.2.7 Predefine Variable

PHP menyediakan banyak sekali penanganan seputar variable, baik itu saat mengirim atau mengambil nilai variable, kita sebut saja sebagai predefined variable. Karena predefined variable yang disediakan cukup banyak, maka tentu saja tidak bisa didokumentasikan secara lengkap disini.

PHP juga memiliki sekumpulan predefined variable tambahan yang bisa berasal dari web server, environment, atau input user (berasal dari form). Kumpulan predefined tersebut memiliki sifat khusus sesuai dengan peruntukannya yang juga bersifat global, karena secara otomatis memiliki jangkuan/scope yang luas. Hal ini sering disebut juga dengan superglobal.

PHP SUPERGLOBAL

1. `$GLOBALS` Merupakan rujukan bagi variable yang memiliki lingkup global disemua script. `$GLOBALS` mulai tersedia di PHP mulai versi 3 ke atas.
2. `$_SERVER` Merupakan sekumpulan variable yang dihasilkan oleh web server. Sehingga semua variable yang berhubungan dengan penanganan server, misalnya untuk mengetahui alamat "IP host" dan "nama host" yang terhubung dengan server atau fungsi-fungsi lainnya yang berhubungan dengan server ditangani oleh `$_SERVER`. Analogi `$_SERVER` untuk PHP versi sebelumnya adalah `$HTTP_SERVER_VARS`.
3. `$_GET` Merupakan variable yang dihasilkan oleh query string URL atau melalui HTTP GET. `$_GET` sangat berhubungan dengan penerimaan

data yang berasal dari halaman website diluar PHP, terutama variable yang berasal dari FORM HTML yang menggunakan method GET. Analogi `$_GET` untuk PHP versi sebelumnya adalah `$HTTP_GET_VARS`.

Pernyataan GET akan dibahas lebih lanjut pada bab “PENANGANAN FORM SESSION”, karena GET sangat penting untuk diketahui oleh pembaca.

4. `$_POST` Merupakan variable yang dihasilkan melalui HTTP POST. `$_POST` sangat berhubungan dengan penerimaan data yang berasal dari halaman website diluar PHP, terutama variable yang berasal dari FORM HTML yang menggunakan method POST. Analogi `$_POST` untuk PHP versi sebelumnya adalah `$HTTP_POST_VARS`.

POST akan dibahas lebih lanjut pada bab “PENANGANAN FORM SESSION”, karena POST sangat penting untuk diketahui oleh pembaca.

5. `$_COOKIE` Merupakan variable yang dihasilkan melalui HTTP COOKIE. Cookie bisa dianggap sebagai variable tampungan berisi data yang ditempatkan pada komputer client. `$_COOKIE` bisa digunakan dengan syarat layanan cookie pada browser client diaktifkan. Analogi `$_COOKIE` untuk PHP versi sebelumnya adalah `$HTTP_COOKIE_VARS`.
6. `$_FILES` Merupakan variable yang dihasilkan melalui HTTP POST UPLOAD FILE, `$_FILES` berguna untuk meng-upload file dari komputer client menuju ke komputer server dengan menggunakan FORM HTML. Analogi `$_FILES` untuk PHP versi sebelumnya adalah `$HTTP_POST_FILES`.
7. `$_ENV` Merupakan variable yang dihasilkan melalui environment. Analogi `$_ENV` untuk PHP versi lebih lama adalah `$HTTP_ENV_VARS`.
8. `$_REQUEST` Merupakan variable yang dihasilkan melalui mekanisme input GET, POST dan COOKIE (bisa menerima variable dari form dengan method POST atau GET), tetapi cara ini kurang bisa dipercaya kebenaran hasil yang diperoleh, sehingga kurang disarankan penggunaannya.
9. `$_SESSION` Merupakan variable yang telah diregister-kan (disimpan dalam file sementara). `$_SESSION` memiliki fungsi yang hampir sama dengan `$_COOKIE`, perbedaanya hanya pada masalah penempatan variable tampungannya. Session meletakkan variable tampungannya di server, sedangkan cookie menempatkan variable tampungannya di client. Cara

ini jauh lebih aman daripada menggunakan cookie. Analogi `$_SESSION` untuk PHP versi sebelumnya adalah `$HTTP_SESSION_VARS`.

Session akan dibahas lebih lanjut dalam bab “PENANGANAN FORM SESSION”, karena session sangat penting untuk diketahui oleh pembaca.

2.2.8 Penangann Session

Gambaran session adalah sebagai berikut : Misalkan sebuah website memiliki banyak halaman, dan salah satu halamannya terdapat form yang harus diisi, kemudian bagaimanakah caranya agar variable data yang terdapat pada form tersebut bisa tercatat disetiap halaman website?, meskipun kita berpindah dari halaman satu ke halaman yang lain. Dengan kata lain, variable tersebut harus memiliki lingkup yang bersifat global dan bisa dikenali di setiap halaman website yang dibuka. Tentu tidak efektif jika selalu dibuat form inputan baru disetiap halaman untuk pendefinisian ulang varibale data, agar bisa dikenali disetiap halaman. Oleh karena itu diperlukan adanya varibale yang memiliki lingkup/scope global yang bisa dikenali disetiap halaman, tanpa harus men-submit ulang formnya. Salah satu caranya adalah dengan menggunakan session.

Contoh umum penggunaan session dalam website adalah :

1. Form login user

Beberapa website terkadang membutuhkan login user, agar user bisa mengakses halaman-halaman website yang memerlukan verifikasi hak akses disetiap halaman tertentu.

2. Website Shopping Cart

Website perbelanjaan biasanya menyimpan nama barang dan jumlah yang akan dibeli di dalam session, sehingga meskipun pengunjung berpindah-pindah halaman, data perbelanjaan akan tetap terjaga.

3. Hit Counter hit counter

Untuk menghitung jumlah pengunjung website, saat pertama kali pengunjung membuka website maka sessionid-nya akan disimpan di variable gloabal session, jika pengunjung membuka halaman lain maka hit counter tetap terhitung satu kali. Hal ini untuk mencegah terhitungnya hit

counter beberapa kali, jika pengunjung membuka halaman lain dalam satu alamat website.

Session bisa dikonfigurasi sesuai dengan keinginan, dimana pengaturan session diletakkan di file `php.ini`. Walaupun pada kenyataannya untuk merubah konfigurasi session tidak mudah untuk dilakukan jika website terletak pada komputer server (virtual hosting), karena pemilik website tidak memiliki hak akses penuh terhadap komputer server, tetapi alangkah baiknya jika mengetahui konfigurasi apa saja yang ada pada session.

Beberapa pengaturan session antara lain :

1. `session.name` Pemberian nama session terdiri dari karakter alphanumeric, nama standarnya adalah `PHPSESSID`.
2. `session.auto_start` Penggunaan session harus diawali dengan `session_start()`, jika `session.auto_start` bernilai 1 maka secara otomatis `session_start()` akan dijalankan saat start up atau komputer dinyalakan, jika bernilai 0 maka session harus diaktifkan secara manual, sehingga `session_start()` harus dideklarasikan terlebih dahulu.
3. `session.cookie_lifetime` Untuk menentukan umur atau durasi session, jika bernilai 0, maka session akan dihapus secara otomatis saat keluar dari browser internet.
4. `session.cookie_path` Untuk menentukan letak path file-file `session_cookie`.

Fungsi built-in PHP yang berhubungan dengan session, antara lain :

1. `SESSION_START()`

Agar bisa menggunakan fungsi-fungsi session, maka disetiap halaman website yang mengandung fungsifungsi session harus diawali dengan `session_start()`.

```
1 <?php session_start(); ?>
```

2. SESSION_DESTROY()

`session_destroy()` berguna untuk menghapus dan mengakhiri session, sekaligus menghapus semua elemen yang ada. Jika browser telah ditutup maka secara otomatis session akan diakhiri walaupun fungsi `session_destroy()` tidak dituliskan secara eksplisit.

```
1 <?php session_start(); session_destroy(); ?>
```

3. SESSION_UNSET()

Untuk menghapus elemen-elemen dari session, tetapi tanpa membuang atau mengakhiri session itu sendiri maka bisa digunakan fungsi `session_unset()`.

```
1 <?php session_start(); session_unset(); ?>
```

4. SESSION_ID()

Untuk mendapatkan atau memberi nilai id pada session, dimana setiap kali pengunjung membuka website maka akan diberikan identifikasi session yang bersifat unik atau berbeda disetiap session yang telah tercipta.

```
1 <?php session_start(); echo "Session ID yang terbentuk : " .  
  session_id() . "<br>"; session_destroy(); ?>
```

5. SESSION_NAME()

Fungsi `session_name()` adalah untuk memperoleh atau memberi nilai terhadap nama sebuah session. Nama session secara standar adalah PHPSESSID. Jika diinginkan nama lain bisa dituliskan sebagai berikut :

```
1 <?php session_start();  
2  
3 // menampilkan nama session standar echo  
4 "Nama session standar : " . session_name() . "<br>";  
5 // memberi nama session sendiri  
6 session_name("SessionKu"); echo "Pemberian nama  
7 session baru : " . session_name();  
8 ?>
```


2.3 PENGENALAN PHPMYADMIN

2.3.1 Pengertian phpMyAdmin

phpMyAdmin adalah sebuah software gratis berbasis scripting language PHP yang bertujuan untuk memudahkan kita mengelola database MySQL. Tanpa phpMyAdmin, Anda perlu menggunakan terminal untuk mengelola database anda. Sementara, dengan phpMyAdmin Anda tidak perlu susah lagi mengelola database MySQL Anda karena phpMyAdmin memiliki user interface grafis. phpMyAdmin mendukung banyak operasional MySQL, MariaDB, dan Drizzle sehingga Anda bisa menggunakannya untuk mengelola database, columns, tables, indexes, users, dan masih banyak lagi. Di saat yang sama, Anda juga bisa meng-execute SQL statement apapun secara langsung. Sampai saat ini, phpMyAdmin sudah diterjemahkan ke 72 bahasa. PhpMyAdmin sendiri sudah berumur 15 tahun dan memenangkan beberapa penghargaan. Seperti yang sudah disebutkan di atas, Anda juga tidak perlu khawatir karena phpMyAdmin mendukung fitur-fitur MySQL seperti MySQL users dan privileges management, stored procedures and triggers management, maintenance server, tables, dan databases, mengeksekusi, mengedit, dan meng-bookmark statement SQL apapun, dan masih banyak lagi.[?]

2.3.2 Sejarah phpMyAdmin

phpMyAdmin pertama kali didirikan oleh Tobias Ratschiller. Saat itu, Ratschiller adalah seorang konsultan IT. Ratschiller juga kemudian mendirikan perusahaan perangkat lunak Maguma. Ratschiller mulai mengerjakan front-end berbasis PHP ke MySQL pada tahun 1998, terinspirasi oleh MySQL-Webadmin Peter Kuppelwieser. Dia sempat berhenti mengerjakan proyek tersebut dan juga phpAdsNew pada tahun 2000 karena kurangnya waktu.

Pada saat itu, phpMyAdmin sudah menjadi salah satu aplikasi PHP dan alat administrasi MySQL yang paling populer, dengan komunitas pengguna dan kontributor yang besar. Untuk mengkoordinasikan semakin banyak patches, tiga orang pengembang, Olivier Müller, Marc Delisle dan Loïc Chapeaux, mendaftarkan proyek phpMyAdmin di SourceForge.net dan mengambil alih pembangunan pada tahun 2001. Sejak tahun 2015 pembangunan sepenuhnya didasarkan pada GitHub.[2]

2.3.3 Fitur-Fitur phpMyAdmin

Berikut adalah beberapa fitur phpMyAdmin yang membuat software satu ini banyak digunakan:

1. phpMyAdmin memiliki interface yang user-friendly dan intuitive yang mudah dipelajari. Dengan begitu, orang-orang dapat mempelajari dan membiasakan diri dengan fitur-fiturnya dengan mudah dan cepat.
2. phpMyAdmin memperbolehkan user memanfaatkan kebanyakan fungsi MySQL, termasuk mengelola database, menjalankan queries MySQL, mengeksekusi statement MySQL, mengelola user dan permission dan masih banyak lagi.
3. Anda juga bisa mengimport dan mengexport data dari dan ke berbagai format.
4. Dengan pre-defined fuctions, Anda bisa mengubah format data-data yang sudah tersimpan
5. phpMyAdmin bisa membuat graphics database dalam bentuk PDF, bersamaan dengan beberapa queries yang kompleks dibantu dengan query-by-example.
6. Melakukan pencarian pada database

2.4 PENGENALAN MySQL

2.4.1 Pengertian MySQL

MySQL adalah sebuah database management system (manajemen basis data) menggunakan perintah dasar SQL (Structured Query Language) yang cukup terkenal. Database management system (DBMS) MySQL multi pengguna dan multi alur ini sudah dipakai lebih dari 6 juta pengguna di seluruh dunia.

MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu Free Software (perangkat lunak bebas) dan Shareware (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah database server yang gratis dengan lisensi GNU General Public License (GPL) sehingga dapat Anda

pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada.

Seperti yang sudah disinggung di atas, MySQL masuk ke dalam jenis RDBMS (Relational Database Management System). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada MySQL. Contohnya di dalam MySQL sebuah database terdapat satu atau beberapa tabel.

SQL sendiri merupakan suatu bahasa yang dipakai di dalam pengambilan data pada relational database atau database yang terstruktur. Jadi MySQL adalah database management system yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server.[6]

2.4.2 Sejarah MySQL

MySQL adalah pengembangan lanjutan dari proyek UNIREG yang dikerjakan oleh Michael Monty Widenius dan TcX (perusahaan perangkat lunak asal Swedia).[2]

Sayangnya, UNIREG belum terlalu kompatibel dengan database dinamis yang dipakai di website. TcX kemudian mencari alternatif lain dan menemukan perangkat lunak yang dikembangkan oleh David Hughes, yaitu miniSQL atau mSQL. Namun, ditemukan masalah lagi karena mSQL tidak mendukung indexing sehingga belum sesuai dengan kebutuhan TcX.

Pada akhirnya muncul kerjasama antara pengembang UNIREG (Michael Monty Widenius), mSQL (David Hughes), dan TcX. Kerjasama ini bertujuan untuk mengembangkan sistem database yang baru, dan pada 1995 dirilis MySQL seperti yang dikenal saat ini. Saat ini pengembangan MySQL berada di bawah Oracle.

2.4.3 Mengenal SQL

Sesuai dengan pengertian MySQL yang sudah dibahas sebelumnya, bahwa sistem manajemen server ini menggunakan SQL sebagai bahasa pemrogramannya.

SQL merupakan bahasa pemrograman yang perlu Anda pahami karena dapat merelasikan antara beberapa tabel dengan database maupun antar database.

Ada tiga bentuk SQL yang perlu Anda ketahui, yaitu Data Definition Language (DDL), Data Manipulation Language(DML), dan Data Control Language (DCL).

1. Data Definition Language (DDL)

DDL berguna pada saat Anda ingin mendefinisikan data di dalam database. Terdapat beberapa query yang dikelompokkan ke dalam DDL, yaitu:

CREATE	Dipakai untuk membuat tabel dan database
DROP	Dipakai untuk menghapus database dan tabel Dipakai untuk mengubah struktur tabel yang sudah ada
ALTER	Alter dapat mengganti field menggunakan perintah “Change”, menambahkan field menggunakan perintah “Add”, atau menghapus field menggunakan perintah “drop”, dan mengubah Namanya menggunakan perintah “Rename”

Figure 2.18: Data Definition Langaue (DDL)

Argumen DDL di atas perlu Anda pahami karena merupakan dasar penggunaan SQL di bagian awal pembuatan database. Contohnya saja jika belum menjalankan perintah “CREATE”, Anda belum bisa melanjutkan penggunaan argumen yang lainnya.

2. Data Manipilation Language (DML)

DML dapat Anda pakai setelah menjalankan perintah DDL. DML berfungsi untuk memanipulasi, mengubah, atau mengganti isi dari database (tabel) yang sudah ada.

Terdapat beberapa perintah DML yang perlu Anda ketahui, yaitu:

INSERT	Dipakai untuk memasukan data ke dalam tabel database
UPDATE	Dipakai untuk mengubah data yang ada di dalam tabel pada database
DELETE	Dipakai untuk menghapus data di dalam tabel pada database

Figure 2.19: Data Menipulation Language (DML)

3. Data Control Language (DCL)

DCL berguna untuk memberikan hak akses database, mendefinisikan space, mengalokasikan space, dan melakukan audit penggunaan database.

Terdapat beberapa perintah DCL yang perlu Anda ketahui, yaitu:

GRANT	Dipakai untuk memberikan izin user untuk mengakses database
REVOKE	Dipakai untuk membatalkan izin user untuk mengakses database
COMMIT	Dipakai untuk menetapkan penyimpanan pada database
ROLLBACK	Dipakai untuk membatalkan penyimpanan pada database

Figure 2.20: Data Control Language (DCL)

2.4.4 Pengertian MySQL

Menurut Adi Nugroho (2011) MySQL (My Structured Query Language) adalah: “ Suatu sistem basis data relation atau Relational Database managemnt System (RDBMS) yang mampu bekerja secara cepat dan mudah digunakan MySQL juga merupakan program pengakses database yang bersifat jaringan, sehingga sapat digunakan untuk aplikasi multi user (banyak pengguna). MySQL didistribusikan gratis dibawah lisensi GPL (General Public License). Dimana setiap program bebas menggunakan MySQL namun tidak bisa dijadikan produk turunan yang dijadikan closed source atau komersial”. [1]

2.4.5 Fungsi-Fungsi di MySQL Server

Jika MySQL sudah dapat berjalan dengan baik di server atau perangkat, ada beberapa fungsi yang bisa Anda jalankan menggunakan teks perintah (command prompt). Untuk masuk ke dalam MySQL server buka ‘CMD’ di Windows dan ‘Terminal’ di Linux.

2.4.6 Login dan Logout MySQL Server

Pengertian MySQL adalah database manajemen server yang cukup aman sehingga mempunyai aturan hak akses yang ketat. Jadi ketika ingin mengakses MySQL, Anda harus menggunakan password yang sudah diatur sebelumnya. Jika sebelumnya belum mengatur user untuk masuk ke dalam MySQL, Anda dapat menggunakan user root dengan mengetikkan perintah berikut.

```
$ mysql -u root -p
```

Opsi ‘-u’ merupakan tanda jika ingin login menggunakan user dengan menggunakan password yang dinyatakan dengan opsi ‘-p’. Jika ingin keluar, Anda hanya perlu mengetikkan perintah “quit” atau “”.

2.4.7 Opsi di MySQL Server

MySQL server menyediakan beberapa bantuan yang bisa Anda akses untuk mengubah dan mendokumentasikan server, yaitu dengan mengetikkan perintah “” atau “” ke dalam koneksi MySQL yang sedang aktif. Perhatian! Semua koneksi harus diakhiri tanda titik koma (;). Tanda ini sebagai petunjuk bahwa perintah atau query yang dimasukkan sudah selesai dan siap dieksekusi.

Bantuan	\h atau \?	Dipakai untuk menampilkan opsi bantuan yang tersedia di dalam MySQL
Clear	\c	Dipakai untuk mengakses atau membatalkan semua perintah yang berjalan pada satu perintah
Connect	\r	Dipakai untuk merefresh koneksi ke database yang ada di dalam Server Host
Ego	\G	Dipakai untuk menampilkan data di dalam database server secara horizontal
Go	\g	Dipakai untuk mengeksekusi perintah yang sudah dimasukkan
Tee	\T	Dipakai untuk mengatur lokasi file untuk perintah yang ingin didokumentasikan
Note	\t	Dipakai untuk mengakhiri perintah \T yang berguna untuk mendokumentasikan semua perintah yang sudah dijalankan
Print	\p	Dipakai untuk menampilkan seluruh perintah yang sudah dijalankan ke layar
Prompt	\R	Dipakai untuk mengubah ‘prompt’ sesuai dengan keinginan
Source	\.	Dipakai untuk mengeksekusi perintah dari luar yang berbentuk file.sql
Use	\u	Dipakai untuk masuk ke dalam database (mirip dengan perintah “cd” di dalam bach script)

Figure 2.21: Opsi di MySQL Server

2.4.8 Perintah Administrasi di MySQL Server

Pengertian MySQL server lainnya yaitu sebuah database manajemen sistem yang bersifat client/server. Jadi sangat penting untuk mengatur hak akses setiap user. Pengaturan hak akses ini supaya semua user mempunyai wewenang yang berbeda. Hak akses juga dapat mengatur user supaya hanya dapat mengakses database tertentu saja.

Pengaturan hak akses di MySQL tentu akan sangat berguna apalagi karena

perangkat lunak ini berjalan di jaringan publik yang kemungkinan semua user bisa mengaksesnya melalui internet. Sebelum mengatur hak akses, Anda harus membuat terlebih dahulu user MySQL selain 'root'.

2.5 PENGENALAN CSS

2.5.1 Pengertian CSS

Oke pertama-tama saat mulai belajar CSS kamu tentu harus paham betul pengertiannya. Sebenarnya apa itu CSS? CSS adalah singkatan dari Cascading Style Sheets. Jika diterjemahkan ke bahasa Indonesia secara harfiah, CSS berarti “lembar penataan menurun”.

Namun secara konteks, Cascading Style Sheets adalah kumpulan perintah yang digunakan untuk menjelaskan tampilan sebuah halaman situs web dalam mark-up language. Mark-up language atau bahasa markah adalah bahasa pemrograman yang biasanya digunakan membuat website.

Kalau kamu sudah mengenal HTML, nah itulah salah satu contoh bahasa mereka. HTML atau Hypertext Mark Up Language tergolong sebagai bahasa pemrograman yang paling standar dan lazim digunakan dalam pembuatan halaman web. HTML ini istilahnya ada di “balik layar” tampilan halaman web yang kamu akses. Tatanan sebuah situs web akan berantakan tanpa adanya bahasa pemrograman seperti HTML.[2]

2.5.2 Cara Kerja CSS

Untuk cara kerjanya sendiri kamu bisa melihat pada Gambar 1. CSS beroperasi melalui tag `<style>` dengan atribut `class` warna. Dengan adanya CSS pada HTML tersebut maka pengaturan warna teks akan menjadi lebih mudah.

Saat kamu ingin mengganti warna teks cukup mengetikkan tag `` tanpa harus menulis ulang perintah. Jadi bisa disimpulkan bahwa CSS akan menghemat waktumu dengan perintah-perintah yang efisien.

Hal ini bisa terjadi karena CSS sendiri dikembangkan untuk bisa mengubah tampilan laman website tanpa harus mengganti isi konten. Jika kembali pada perumpamaan manusia dan pakaian di poin sebelumnya, dengan CSS kamu

tidak mengubah bentuk manusianya tapi hanya mengganti pakaiannya. Dengan begitu untuk mengubah dan memprogram ulang tampilan website pun bisa dilakukan dalam waktu cepat.

2.5.3 Peran CSS

Dari sini kamu bisa menyimpulkan bahwa peran CSS untuk website sangatlah penting. Tanpa adanya CSS, tampilan website akan membosankan atau bahkan membutuhkan waktu lama untuk loading. Bayangkan saja kamu hanya bisa bergantung pada HTML untuk membuat sebuah situs. Bukan cuma tampilan situs akan “hambar” tapi kamu juga butuh waktu lebih lama karena harus berulang kali mengetikkan perintah.

2.5.4 Macam-Macam CSS

Saat belajar CSS kamu perlu mengetahui bahwa CSS sendiri dibagi menjadi beberapa macam dengan penggunaan yang berbeda-beda. Hal ini dimaksudkan agar kamu mencoba mempraktikkan teori yang sudah dipelajari tidak mengalami kekeliruan. Setidaknya ada tiga macam CSS yang sering digunakan oleh web designer. Berikut penjelasannya.

1. Inline Style Sheet

Pertama ada inline style sheet. Sederhananya, CSS model ini adalah CSS dengan perintah pemrograman yang letaknya ada pada objek. Misalnya kamu ingin mengubah sebuah tulisan pada laman tertentu di website milikmu, inline style sheet CSS harus menempel pada elemen tulisan tersebut. Kamu cukup menambahkan tag `<style>` saja untuk menerapkan CSS ini.

2. External Style Sheet

Selain inline style sheet ada external style sheet. CSS ini letaknya berbeda dengan laman yang akan diubah. Cara ini lebih praktis daripada inline style sheet karena bisa menghemat ruang dan bisa digunakan berulang-ulang untuk laman web yang berbeda. Kamu bisa mengenali CSS tipe ini lewat tag `<link rel>`. Tag ini akan menghubungkan halaman coding pada external style sheet CSS yang terpisah.

3. Embedded Style Sheet

Terakhir ada embedded style sheet. CSS model ini sama seperti inline style sheet, sama-sama berada pada satu laman coding. Maka tidak mengherankan jika embedded style sheet terkadang disebut dengan internal style sheet. Biasanya CSS ini diapit oleh tag `<head>` `</head>` dan diawali dengan tag `<style>`. Embedded style sheet sering digunakan untuk mengatur laman web dengan tampilan yang unik. Misalnya dalam satu paragraf tulisan ada kalimat yang berbeda dan hal tersebut terus berulang.

Sampai di bagian ini kamu bisa menyimpulkan bahawa CSS punya andil yang besar dalam pembuatan website. Tanpa adanya CSS tampilan halaman web akan terlihat membosankan. Dari sisi web designer sendiri, CSS akan membantu mereka untuk menghemat waktu karena tidak harus lagi mengetikkan perintah yang sama.[2]

2.6 HUBUNGAN CSS, HTML, PHP DAN MySQL

Lalu, apa hubungannya HTML dengan CSS? Keduanya sangat berhubungan erat. CSS adalah kode-kode yang dipakai untuk mendesain sebuah laman HTML. Jika HTML diibaratkan sebagai seorang manusia, maka CSS adalah pakaian yang membuat penampilan menjadi semakin menarik. CSS akan membantu para web designer untuk mengubah tampilan teks (baik dari bentuk dan ukuran font maupun warnanya), menambahkan gambar, hingga mengubah latar belakang sebuah halaman HTML.

Keberadaan CSS bisa terlihat dengan adanya atribut warna teks. Di sini CSS memberi perintah berupa teks berwarna biru melalui tag `` dengan atribut `class="warna"`. Jadi setiap tag `` muncul, teks yang mengikutinya akan berwarna biru. Pun demikian saat kamu ingin mengganti warnanya. Tinggal mengganti CSS pada tag `<style>` dari "blue" ke warna lain, maka teks yang awalnya berwarna biru akan berubah warna.

Melihat dari pengertian – pengertian di atas dapat disimpulkan bahwa HTML dan CSS satu paket bahasa pemrograman yang digunakan untuk membuat tampilan sebuah website, sedangkan PHP dan MYSQL digunakan untuk membuat halaman website yang dibuat HTML + CSS menjadi dinamis.

PHP berfungsi sebagai penghubung HTML dan MYSQL. Peran PHP MYSQL dalam membaut web dinamis adalah MYSQL sebagai penampung datanya seperti konten – konten yang dapat berubah – ubah dalam web dinamis maka mysql lah yang manampung datanya, sedangkan PHP digunakan untuk menghubungkan query – query yang ada dalam mysql baik untuk menampilkan data dalam web atau menyimpan data dalam mysql.

2.7 PENGENALAN XAMPP

2.7.1 Pengertian XAMPP

XAMPP adalah sebuah paket perangkat lunak (software) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / MariaDB (sekarang), PHP, dan Perl. Sementara imbuhan huruf “X” yang terdapat pada awal kata berasal dari istilah cross platform sebagai simbol bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti OS Linux, OS Windows, Mac OS, dan juga Solaris.

Sejarah mencatat, software XAMPP pertama kali dikembangkan oleh tim proyek bernama Apache Friends dan sampai saat ini sudah masuk dalam rilis versi 7.3.9 yang bisa didapatkan secara gratis dengan label GNU (General Public License).

Jika dijabarkan secara gamblang, masing-masing huruf yang ada di dalam nama XAMPP memiliki arti sebagai berikut ini:

X = Cross Platform

Merupakan kode penanda untuk software cross platform atau yang bisa berjalan di banyak sistem operasi.

A = Apache

Apache adalah aplikasi web server yang bersifat gratis dan bisa dikembangkan oleh banyak orang (open source).

M = MySQL / MariaDB

MySQL atau MariaDB merupakan aplikasi database server yang dikembangkan oleh orang yang sama. MySQL berperan dalam mengolah, mengedit, dan menghapus daftar melalui database. P = PHP

Huruf “P” yang pertama dari akronim kata XAMPP adalah inisial untuk menunjukkan eksistensi bahasa pemrograman PHP. Bahasa pemrograman ini biasanya digunakan untuk membuat website dinamis, contohnya dalam website berbasis CMS WordPress.

P = Perl

Sementara itu, untuk huruf P selanjutnya merupakan singkatan dari bahasa pemrograman Perl yang kerap digunakan untuk memenuhi berbagai macam kebutuhan. Perl ini bisa berjalan di dalam banyak sistem operasi sehingga sangat fleksibel dan banyak digunakan.

2.7.2 Fungsi XAMPP

Program aplikasi XAMPP berfungsi sebagai server lokal untuk mengampu berbagai jenis data website yang sedang dalam proses pengembangan. Dalam prakteknya, XAMPP bisa digunakan untuk menguji kinerja fitur ataupun menampilkan konten yang ada didalam website kepada orang lain tanpa harus terkoneksi dengan internet, atau istilahnya website offline.

XAMPP bekerja secara offline layaknya web hosting biasa namun tidak bisa diakses oleh banyak orang. Maka dari itu, XAMPP biasanya banyak digunakan oleh para mahasiswa maupun pelajar untuk melihat hasil desain website sebelum akhirnya dibuat online menggunakan web hosting yang biasa dijual dipasaran.

2.7.3 Bagian-Bagian Penting di Dalam XAMPP

1. HTDOCS

Htdocs merupakan nama sebuah folder bagian dari XAMPP yang berfungsi untuk menyimpan berbagai file dan dokumen yang akan ditampilkan kedalam website. Kapasitas penyimpanan di dalam menu htdocs sendiri bergantung kepada harddisk yang terpasang di perangkat komputer.

2. Control Panel

Seperti namanya, Control Panel memberikan kita akses lebih leluasa dalam mengatur database, mengunggah file, atau melakukan setting lebih detail terkait jeroan website. Dari dalam Control Panel kita juga bisa mengatur stop dan start aplikasi XAMPP agar tidak memberatkan kinerja komputer.

3. PhpMyAdmin

Sama seperti fungsi PhpMyAdmin pada web hosting sesungguhnya, peran PhpMyAdmin dalam aplikasi XAMPP juga diplot sebagai pengatur konfigurasi MySQL. Untuk membuka phpMyAdmin di XAMPP caranya Anda bisa langsung mengetikkan URL `http://localhost/phpMyAdmin`.

2.8 PENGENALAN DATABASE

2.8.1 Pengertian Database



Figure 2.22: Pengertian Database

Sumber : <https://www.nesabamedia.com/pengertian-database-dan-fungsinya/>

Seperti yang dijelaskan sebelumnya bahwa suatu database pada hakikatnya terdiri dari kumpulan data yang sudah diperoleh sebelumnya. Jika dilihat dari segi Bahasa maka database juga dapat disebut basis data.[5]

Suatu data dapat dimaksudkan informasi yang diperoleh dan disimpan sedangkan basis adalah semacam perkumpulan atau tempat berkumpul. Maka dapat diambil kesimpulan bahwa arti dari database adalah kumpulan data atau informasi yang diperoleh dan selanjutnya disimpan dalam suatu media, umumnya adalah di komputer.

Pengolahan database dalam media komputer ditujukan untuk mempermudah dan tentunya mengikuti perkembangan zaman yang semakin menerapkan era komputerisasi. Suatu pengelolaan sistem database dalam dunia IT biasa dikenal dengan istilah DBMS (Database Management System). Suatu database juga dapat didefinisikan terdiri dari kumpulan tabel – tabel yang menyimpan data serta informasi.

Namun pada hakikatnya penerapan database tidak hanya terdapat dalam lingkup IT saja, namun lebih dari itu. Contohnya pada sekolah atau universitas terdapat database mahasiswa, murid, tenaga pengajar, sarana prasarana dan lain lain. Dalam lingkungan perusahaan juga pastinya terdapat data – data perusahaan mencakup database karyawan, keuangan, dan lain – lain.[?]

2.8.2 Fungsi Database

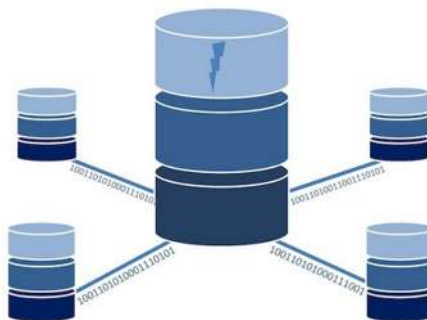


Figure 2.23: Fungsi Database

Sumber : <https://www.nesabamedia.com/pengertian-database-dan-fungsinya/>

Bagi orang awam mungkin suatu database hanyalah kumpulan data dan informasi yang hanya perlu di backup untuk keamanan. Namun dibalik itu terdapat fungsi – fungsi lain dari penggunaan database, yakni sebagai berikut :

1. Suatu data dapat dikelompokkan dengan tujuan mempermudah proses identifikasi data, pengelompokkan dapat dilakukan dengan berbagai macam cara seperti membuat beberapa tabel atau dengan field yang berbeda – beda. Sebagai contoh suatu DBMS pada perbankan dapat mencari informasi user dengan lebih cepat karena sudah dikelompokkan masing – masing.
2. Menghindari data ganda yang tersimpan. Suatu software DBMS dapat di setting agar mampu mengenali duplikasi data yang terjadi saat diinput. Hal ini dikarenakan sifat database yang dapat diakses oleh lebih dari satu pengguna. Salah satu cara yang dilakukan adalah dengan menerapkan sistem kata kunci atau Primary Key.
3. Mempermudah penggunaan hampir di semua sisi seperti memasukkan data baru, mengupdate atau bahkan menghapus data yang sudah tidak diperlukan lagi. Didukung dengan tampilan atau tata muka yang sudah disediakan menggunakan aplikasi tertentu.
4. Menjadi solusi terbaik dari penggunaan kertas sebagai media penyimpanan yang kurang efektif dan banyak memakan ruang. Dengan adanya database maka file dapat disimpan secara digital.
5. Suatu database juga dapat menjadi alternatif lain terkait masalah penyimpanan ruang dalam suatu aplikasi. Hal ini dikarenakan keterbatasan dari media penyimpanan oleh kebanyakan aplikasi komputer.

2.8.3 Jenis-Jenis

Database

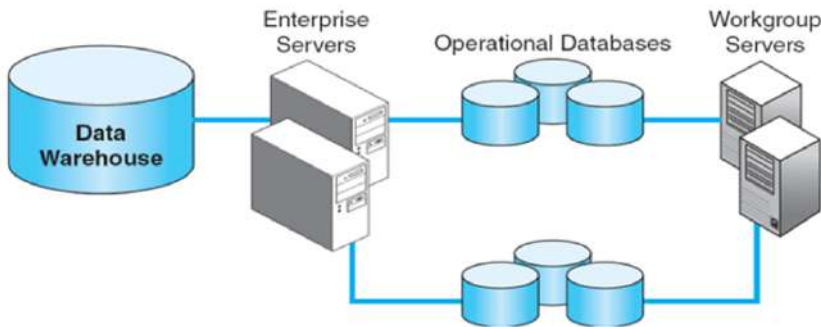


Figure 2.24: Jenis-Jenis DB

Sumber : <HTTPS://WWW.NESABAMEDIA.COM/PENGERTIAN-DATABASE-DAN-FUNGSIONYA/>

Operational Database

Suatu database pada umumnya terdiri dari data yang digunakan oleh banyak user atau pengguna, dari masing – masing user tersebut tentunya memiliki keperluan yang berbeda – beda juga. Dengan adanya hal ini suatu sistem manajemen database juga dapat dikelompokkan menjadi beberapa jenis dengan fungsi dan kegunaan masing – masing. Dibawah ini adalah jenis – jenis database yaitu :

1. Operational Database

Database jenis ini dapat menyimpan data dengan rinci agar dapat dioperasikan dari seluruh organisasi. Dalam hal ini mereka juga dapat disebut Subject Area Database (SADB), transaksi database atau produksi database. Contoh dari Operational Database adalah database yang digunakan pelanggan, database akuntansi dan lain – lain.

2. Relational Database

Jenis database yang satu ini bisa dibilang paling populer, hal ini dikarenakan dengan menggunakan Relational Database, user dapat mengakses atau mencari informasi dalam tabel yang berbeda – beda. Query yang dilakukan juga dapat melibatkan beberapa tabel karena fungsi relasi ini.

3. Distributed Database

Sesuai dengan namanya database yang satu ini dapat mendistribusikan data – data secara tersebar namun saling berhubungan serta dapat diakses secara bersama – sama. Database jenis ini biasanya digunakan pada suatu lembaga atau perusahaan yang terdiri dari beberapa cabang, agar dapat memudahkan user mengakses data dari tempat satu ke tempat lainnya.

4. External Database

Database ini mampu menyediakan akses ke bagian eksternal, data yang disimpan nantinya akan digunakan untuk keperluan komersial. Akses ke dalam database ini akan lebih mudah karena diperuntukkan bagi publik dan disamping itu juga lebih efisien karena tidak perlu mencari informasi dari internet.

Selain jenis – jenis diatas sebenarnya masih banyak contoh database lain dengan fungsi yang berbeda – beda seperti End – User Database, Navigation Database, In Memory Database, Real Time Database, Document Oriented Database serta masih banyak lagi. Namun jenis yang kami jelaskan mungkin yang paling sering digunakan.

Itulah pembahasan mengenai pengertian database beserta fungsi dan jenis-jenis database yang paling umum digunakan. Setelah membaca artikel diatas, sekarang anda pasti sudah mengerti apa yang dimaksud dengan database, fungsinya untuk apa dan ketika ingin membuat sebuah database, anda sudah mempunyai gambaran mengenai jenis database apa yang ingin anda gunakan, entah itu operational database, relational database atau jenis database lain.

2.8.4 Pembagian Basis Data

Menurut jenisnya, basis data dapat dibagi menjadi:

1. Basis data flat-file.

Basis data ini ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Pada dasarnya, basis data flat-file tersusun dari sekumpulan string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. Basis data flat-file cocok untuk menyimpan

daftar atau data yang sederhana dan dalam jumlah kecil. Basis data flat-file akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk itu.

Beberapa kendala dalam menggunakan basis data jenis ini adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi dan juga adanya duplikasi data yang mungkin sulit dihindari. Salah satu tipe basis data flat-file adalah file CSV yang menggunakan pemisah koma untuk setiap nilainya.

2. Basis data relasional.

Basis data ini mempunyai struktur yang lebih logis terkait cara penyimpanannya. Kata "relasional" berasal dari kenyataan bahwa tabel-tabel yang ada di basis data relasional dihubungkan satu dengan lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing-masing tabel tersusun atas baris (tupel) dan kolom (atribut).

Untuk membuat hubungan antara dua atau lebih tabel, digunakan key (atribut kunci) yaitu primary key di salah satu tabel dan foreign key di tabel yang lain. Saat ini, basis data relasional menjadi pilihan utama karena keunggulannya. Program aplikasi untuk mengakses basis data relasional menjadi lebih mudah dibuat dan dikembangkan dibandingkan dengan penggunaan basis data flat-file.

Beberapa kekurangan yang mungkin dirasakan di basis data jenis ini adalah implementasi yang lebih sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi. Selain itu, proses pencarian informasi juga menjadi lebih lambat karena perlu menghubungkan tabel-tabel terlebih dahulu apabila datanya tersebar di beberapa tabel.

Namun, terlepas dari beberapa kekurangannya, basis data relasional telah digunakan secara luas. Saat ini, basis data relasional telah banyak dimanfaatkan oleh perusahaan-perusahaan dari skala kecil, menengah hingga besar. Beberapa basis data ternama yang ada saat ini, baik yang berasal dari sumber terbuka (open source) atau yang komersil, adalah juga basis data relasional.

2.8.5 Contoh Beberapa Basis Data

Berikut ini adalah beberapa basis data relasional populer dan paling banyak digunakan untuk saat ini beserta sejarahnya:

1. Basis data MySQL.



Figure 2.25: DB MySQL

Sumber : <https://www.termasmedia.com/lainnya/software/69-pengertian-database.html>

MySQL merupakan basis data sumber terbuka yang paling populer dan banyak digunakan untuk aplikasi berbasis web seperti website dinamis dan e-commerce. Tahun 2013, MySQL merupakan basis data kedua yang paling banyak digunakan di dunia dan yang pertama untuk basis data sumber terbuka.

Dilihat dari sejarahnya, MySQL dibuat tahun 1995 dan disponsori oleh perusahaan Swedia, MySQL AB. Pengembang platform MySQL adalah Michael Widenius, David Axmark dan Allan Larsson. MySQL dibuat untuk menyediakan opsi pengelolaan data yang efisien, terpercaya dan handal. Pada tahun 2000, platform MySQL berubah menjadi sumber terbuka dan mengikuti ketentuan GPL.

Penggunaan MySQL sebagai basis data utama untuk aplikasi web sering dipadukan dengan PHP sebagai bahasa skrip berorientasi obyek. MySQL adalah salah satu komponen penting dari web service solution stack LAMP (Linux, Apache, MySQL and PHP) yaitu platform pengembangan web sumber terbuka dimana Linux sebagai sistem operasi, Apache sebagai Web Server, MySQL sebagai basis data dan PHP sebagai bahasa skrip.

Apabila Anda membuat blog atau website menggunakan CMS seperti Joomla, Wordpress, Drupal atau Magento, Anda sedang menggunakan MySQL sebagai solusi basis datanya. MySQL juga banyak digunakan oleh

perusahaan-perusahaan besar dunia seperti Facebook, Google, Adobe, Alcatel Lucent dan juga Zappos.

Pada Januari 2008, MySQL diakuisisi oleh Sun Microsystems. Pada April 2009, terjadi pencapaian kesepakatan antara Sun Microsystems dan Oracle Corporation terkait pembelian Sun Microsystems beserta hak cipta (copyright) dan merek dagang (trademark) MySQL oleh Oracle. Namun baru pada Januari 2010, MySQL secara resmi diakuisisi oleh Oracle.

Di bawah naungan Oracle Corporation, MySQL tersedia melalui skema lisensi ganda. Anda dapat menggunakan opsi lisensi sumber terbuka (GPL) selama masih mematuhi aturan lisensi tersebut. Jika Anda ingin mendistribusikan aplikasi non-GPL dimana terdapat MySQL di dalamnya, Anda dapat membeli lisensi komersial sebagai gantinya.[2]

Basis data Oracle



Figure 2.26: DB Oracle

Sumber : <https://www.termasmedia.com/lainnya/software/69-pengertian-database.html>

Oracle merupakan basis data relasional terkemuka yang dimiliki oleh Oracle Corporation. Oracle telah dianggap sebagai basis data terbaik untuk versi basis data komersial. Oracle sendiri tersedia dalam berbagai konfigurasi dengan cakupan tool yang dapat disesuaikan untuk perusahaan skala kecil, menengah hingga besar yang membutuhkan solusi yang terbaik dan tepat dari sebuah basis data untuk keperluan bisnisnya. Oracle dianggap lebih baik untuk masalah kinerja dan skalabilitas dibandingkan dengan basis data komersial lainnya.

Sejarah Oracle dimulai untuk pertama kali pada tahun 1977 ketika versi pertamanya dikembangkan oleh SDL (Software Development Laboratories) yang di dalamnya terdapat Larry Ellison dan dua orang temannya, Bob Miner dan Ed Oates. Nama Oracle sendiri berasal dari nama kode (code-name) sebuah proyek yang didanai oleh CIA ketika Larry Ellison masih bekerja di pekerjaan sebelumnya di Ampex.

Pada tahun 1979, versi komersial Oracle tersedia untuk pertama kali, sedangkan versi terkini Oracle adalah Oracle 12c, dimana "c" mengacu pada cloud computing (komputasi awan). Cloud computing merupakan refleksi kerja Oracle untuk memperluas basis datanya yang memungkinkan perusahaan mengkonsolidasi dan mengelola basis data sebagai cloud service.[2]

Basis data Microsoft SQL Server.



Figure 2.27: DB Microsoft SQL Server

Sumber : <https://www.termasmedia.com/lainnya/software/69-pengertian-database.html>

Dari namanya, Anda bisa menebak dengan mudah siapa pemilik basis data ini. Benar, Microsoft tidak hanya terkenal dengan perangkat lunak sistem operasi Microsoft Windows dan Microsoft Windows Server, aplikasi perkantoran Microsoft Office dan programming language suite Microsoft Visual Studio, tetapi juga dengan basis datanya, Microsoft SQL Server.

Microsoft SQL Server adalah basis data relasional yang bersifat komersial. Tidak seperti Microsoft Office Access yang peruntukannya untuk

komputer dekstop, Microsoft SQL Server untuk komputer dengan Windows Server yang menyediakan layanan pengelolaan basis data kelas perusahaan dan juga alat intelijen bisnis terpadu (integrated business intelligence (BI) tools).

Cukup banyak perusahaan skala kecil, menengah maupun besar dalam mengelola sistem informasi perusahaan menggunakan sistem operasi Microsoft Windows Server. Microsoft SQL Server tentu menjadi pilihan bagi pengguna Microsoft Windows Server karena selain mudah digunakan dan faktor ketersediaan, juga telah terintegrasi dengan Microsoft Windows Server.

Versi terbaru basis data ini yaitu Microsoft SQL Server 2016 tersedia untuk platform on-premises (yang dikelola di server lokal) dan juga sebagai cloud database. Versi terbaru ini juga membawa fitur-fitur baru dan sejumlah peningkatan yang dapat memberi terobosan kinerja, keamanan lanjutan serta pelaporan dan analisis terpadu.

Awal sejarah Microsoft SQL Server dimulai ketika Microsoft dan Sybase mengembangkan sistem manajemen basis data yang diberi nama Sybase SQL Server untuk bisa berkompetisi dengan basis data yang sudah ada saat itu seperti IBM dan Oracle. Sybase akan mendapatkan hak penjualan dan keuntungan dari versi produk basis data yang tidak didesain untuk platform Microsoft. Pada tahun 1989, basis data pertama dirilis. Kemudian Sybase menjual hak ciptanya tersebut ke Microsoft dan akhirnya Microsoft memberi nama ulang basis data tersebut menjadi Microsoft SQL Server.[2]

Basis data MariaDB.

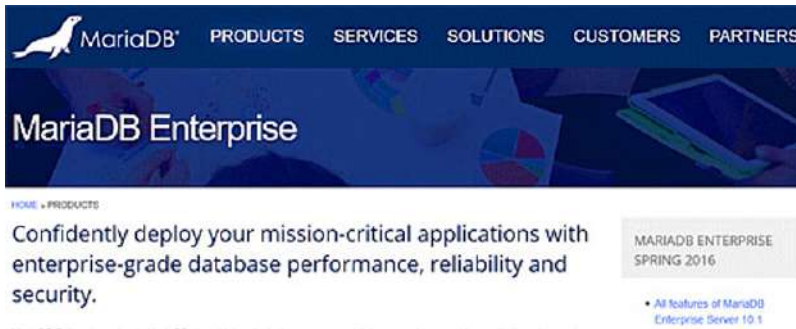


Figure 2.28: DB MariaDB Enterprise

Sumber : <https://www.termasmedia.com/lainnya/software/69-pengertian-database.html>

Mungkin sebagian dari Anda masih asing atau pernah mendengar nama basis data ini. MariaDB merupakan sistem basis data relational yang sepenuhnya sumber terbuka. Semua kode sumber basis data MariaDB dirilis di bawah lisensi GPL, LGPL atau BSD. MariaDB sebenarnya adalah fork dari basis data MySQL.

Fork di sini dapat diartikan sebagai proyek terkait yang dapat dianggap sebagai versi lain (mini) dari MySQL standar. MariaDB merupakan pengembangan versi lain MySQL oleh Michael Widenius yang telah meninggalkan Sun Microsystems setelah diakuisisi secara resmi oleh Oracle Corporation. Baca sejarah dan fork MySQL di artikel Mengenal Database MySQL Dan Keunggulannya.

Kepopuleran basis data ini terus merangkak naik. Ini terbukti pada tahun 2013, Red Hat Enterprise Linux (RHEL) telah menyingkirkan MySQL dan kemudian menggantinya dengan MariaDB. Demikian juga dengan Fedora yang lebih memilih MariaDB dibandingkan MySQL untuk rilis Fedora 19.

Rilis stabil terbaru MariaDB adalah versi 10.1 yang juga dikenal dengan MariaDB Enterprise 2016. Di rilis stabil terbarunya ini, MariaDB memiliki proteksi yang lebih baik dengan pengamanan basis data pada setiap lapisan. MariaDB dapat mendeteksi dan mencegah serangan dengan menggunakan access control, Key Management Services dan juga authentication.[2]

2.9 PERANGKAT KERAS

2.9.1 Pengertian Perangkat Keras

Perangkat Keras – Pada seperangkat komputer terdapat beberapa komponen yang bekerja saling mendukung. Komponen tersebut dikenal dengan istilah hardware atau perangkat keras. Setiap komputer dirancang dengan spesifikasi perangkat keras tersendiri.[2]

Oleh karena itu performa, kecepatan dan keunggulan dari setiap komputer berbeda-beda. Semakin tinggi spesifikasi hardware-nya, maka semakin bagus pula kinerjanya.

Kalau perangkat lunak atau software tidak berwujud fisik, maka perangkat keras atau hardware memiliki pengertian sebaliknya. Oleh karena perangkat keras berbentuk fisik, sehingga bisa dilihat, diraba dan diketahui bentuknya. Keberadaan hardware pada sebuah komputer tentunya sangatlah vital. Setiap perangkat keras memiliki fungsi tersendiri.

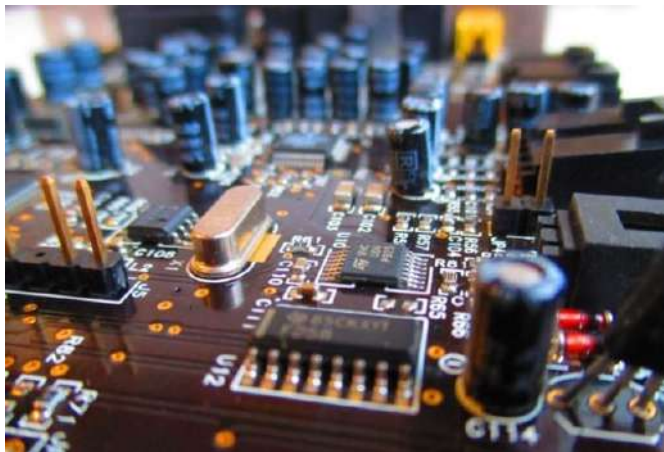


Figure 2.29: Pengertian Perangkat Keras

Sumber : <https://salamadian.com/pengertian-perangkat-keras-hardware/>

Dengan begitu, setiap hardware tidak bisa berdiri sendiri. Jadi dalam seperangkat komputer harus terdiri atas beberapa perangkat keras, seperti motherboard, harddisk, kartu memori, CPU dan sebagainya.

Semua komponen tersebut saling mendukung, sehingga komputer bisa beroperasi. Kemudian semua komponen perangkat keras tersebut membutuhkan perangkat lunak agar komputer bisa dihidupkan dan difungsikan.

2.9.2 Fungsi Perangkat Keras

Setidaknya terdapat lima macam perangkat keras pada sebuah perangkat komputer. Jenis hardware tersebut berfungsi sebagai pendukung kelistrikan, penyimpanan, output, proses dan input. Seluruh jenis perangkat keras itu nantinya terhubung dalam sistem yang terintegrasi. Sehingga bisa difungsikan untuk input, pemrosesan dan output data.

Performa perangkat keras sangat ditentukan oleh spesifikasinya. Semakin tinggi maka kinerjanya akan semakin bagus. Untuk itu, anda bisa menyesuaikan spek komputernya sesuai dengan kebutuhan dalam penggunaannya.



Figure 2.30: Fungsi Perangkat Keras

Sumber : <https://salamadian.com/pengertian-perangkat-keras-hardware/>

Kalau komputer hanya difungsikan untuk mengetik, menyimpan foto dan menonton video tidak harus menggunakan spek tinggi. Namun, kalau komputer digunakan untuk kebutuhan desain grafis, bermain video berat, maka harus menggunakan komputer dengan spek tinggi.

2.9.3 Macam-Macam Perangkat Keras

Seperti disinggung di atas, perangkat keras terdiri atas bermacam jenis dengan fungsinya masing-masing. Nah, agar lebih jelas fungsi dari setiap macamnya, berikut ini pemaparannya.

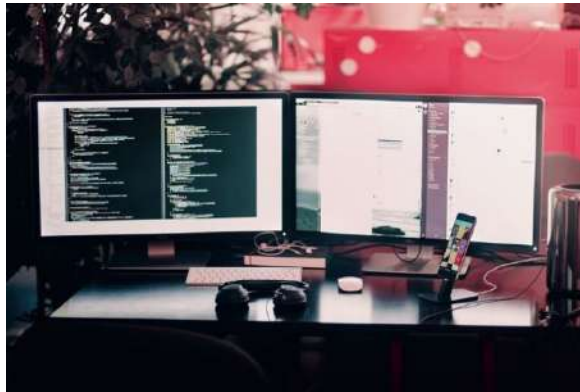


Figure 2.31: Macam-macam Perangkat Keras

Sumber : <https://salamadian.com/pengertian-perangkat-keras-hardware/>

1. Perangkat keras input. Fungsinya adalah untuk input atau memasukkan data pada memori komputer. Keyboard adalah hardware utama untuk fungsi ini. Sedangkan perangkat peripheral atau pendukungnya adalah mouse dan scanner.
2. Perangkat keras untuk pemrosesan data. Hardware ini memiliki fungsi dalam olah data atau untuk mengeksekusi perintah apabila ada data yang dimasukkan. Komponen utama dari hardware ini adalah CPU (Central Processing Unit), sound card (kartu suara), VGA, dan motherboard.
3. Perangkat keras output. Jenis hardware ini memiliki fungsi sebagai penampil data atau mengeluarkan data yang sudah diproses sebelumnya. Monitor dan speaker adalah perangkat utama untuk fungsi tersebut. Komponen pendukungnya seperti printer, proyektor, dan sebagainya.
4. Perangkat keras untuk penyimpanan. Fungsi dari perangkat keras ini adalah sebagai penyimpan data dari pengguna. Terdapat dua hardware komputer untuk fungsi ini, yakni RAM (untuk penyimpanan volatile), dan harddisk (untuk media penyimpanan non volatile internal). Ada

pula Read Only Memory (ROM) sebagai media simpan dengan sifat non volatile. Jadi datanya tidak akan lenyap ketika listrik mati. Sesuai namanya maka sifatnya hanya read only. ROM umumnya dimanfaatkan sebagai Firmware atau chip BIOS.

5. Perangkat keras kelistrikan. Agar sebuah komputer bisa beroperasi, pastinya membutuhkan pasokan energi listrik. Adanya hardware ini berfungsi untuk menstabilkan tegangan atau daya listrik yang mengalir pada perangkat komputer. Perangkat keras seperti Power Supply bermanfaat untuk mengelola sistem kelistrikan bagi setiap perangkat kerasnya. Tidak hanya itu, biasanya komputer juga dikoneksikan dengan stavolt atau stabilizer voltage. Fungsinya untuk menstabilkan daya listrik, sebelum nantinya dikelola oleh Power Supply. Uninterruptible Power Supply (UPS) bisa juga ditambahkan pada perangkat komputer. Tujuannya sebagai hardware untuk backup daya listrik. Sehingga saat mati listrik, maka perangkat komputer tetap bisa hidup untuk beberapa waktu. Pemasangan UPS sangat penting, agar komputer tidak tiba-tiba mati ketika aliran listriknya terputus. Dengan begitu, sekaligus mencegah risiko kerusakan perangkat keras dari komputernya.

2.9.4 Contoh Perangkat Keras



Figure 2.32: Contoh Perangkat Keras

Sumber : <https://salamadian.com/pengertian-perangkat-keras-hardware/>

Sangat banyak contoh perangkat keras untuk mendukung performa komputer. Supaya kinerja komputer anda lebih cepat, nyaman saat multitasking atau membuka beberapa software sekaligus, tidak mudah panas, dan sebagainya.

Maka harus pandai dalam memilih spesifikasi hardwarenya. Nah, berikut ini sejumlah contoh dari perangkat lunak yang lazim digunakan di perkantoran atau pribadi.

1. Keyboard. Perangkat keras ini tersusun dari banyak tombol, sehingga memudahkan anda saat mengoperasikan komputer. Input data sangat didukung dengan keberadaan hardware ini. Anda bisa menggunakan keyboard dengan kabel atau nirkabel.
2. Mouse. Perangkat keras pendukung ini dimanfaatkan sebagai input data, yang bisa mengendalikan kursor, fungsi klik dan scroll. Anda juga bisa menggunakan mouse dengan kabel dan yang tanpa kabel.
3. Scanner. Hardware ini bekerja layaknya mesin fotocopy. Jadi bisa memindai data berupa teks pada buku, foto dan sebagainya. Sehingga bisa menciptakan softcopy atau file dari berbagai media tersebut.
4. CPU (Central Processing Unit) atau Processor. Perangkat keras ini adalah otak dari komputer. Sehingga seluruh proses yang berlangsung pada seperangkat komputer dikendalikan oleh prosesor. Semakin canggih prosesornya sudah pasti semakin bagus pula performa komputernya.
5. Motherboard. Perangkat keras ini merupakan papan sirkuit yang mengkoneksikan seluruh hardware pada perangkat komputer. Guna mendukung kinerjanya, motherboard dilengkapi chip BIOS (Basic Input Output System).
6. Video Graphic Array atau VGA. Tugas dari perangkat keras ini adalah memproses visualisasi data berbentuk grafis. Perkembangan hardware ini sangat pesat, guna mendukung industri desain grafis, game dan multimedia.
7. Sound Card. Kartu suara ini berfungsi dalam pemrosesan data dengan format suara. Perangkat keras ini bermanfaat saat komputer mengeluarkan suara atau melakukan perekaman suara.

8. Monitor. Media output ini berguna dalam menayangkan hasil dari proses data grafisnya yang didukung oleh VGA.
9. Speaker. Perangkat ini berfungsi untuk mengeluarkan suara, yang merupakan hasil pemrosesan dari kartu suaranya.
10. Printer. Perangkat keras ini berguna untuk mencetak hasil kerja pada komputer. Kertas biasanya media yang lazim digunakan. Saat ini ada pula produk yang menggabungkan ketiga fungsi pada sebuah printer. Sehingga perangkat ini bisa untuk mencetak, fotocopy dan scan.
11. Proyektor. Biasanya perangkat ini digunakan sebagai penunjang presentasi, proses belajar mengajar atau menonton film. Anda tinggal mengkoneksikan pada komputer, dan menembakkan cahaya proyektor pada media yang sesuai.
12. RAM (Random Access Memory) dan ROM (Read Only Memory). RAM adalah media simpan data ketika diproses oleh prosesor. Sedangkan ROM merupakan media simpan yang sifatnya hanya read only.
13. Harddisk. Perangkat keras ini tersusun dari piringan sebagai media penyimpanan sistem operasi, software dan data pengguna komputer.

INSTALASI DAN CARA PENGGUNAAN TOOLS

3.1 CARA MENGINSTALL MYSQL DI WINDOWS DAN LINUX

MySQL server dapat digunakan dan mendukung berbagai macam sistem operasi (*cross-platform*), terlebih lagi sistem operasi Linux dan Windows. Proses instalasinya pun sangat mudah. Jika menggunakan sistem operasi Windows, Anda bisa menginstall XAMPP untuk menjalankan MySQL server yang di dalamnya sudah terdapat juga modul untuk menjalankan Apache, PHP, FileZilla, dan Tomcat. Sedangkan untuk menginstall sistem operasi Linux, Anda dapat menginstall MySQL secara terpisah atau menginstall LAMP (Linux, Apache, MySQL, PHP) yang sudah ada modul Apache dan PHP juga. Selain itu, XAMPP juga sudah tersedia di Linux. Sehingga proses instalasi dapat anda lakukan dengan proses yang paling mudah menurut anda.

1. Instalasi MySQL di sistem operasi Windows menggunakan XAMPP sangat mudah.
 - Langkah 1. Pertama-tama Download file instalasi XAMPP . file instalasi XAMPP dapat di download pada halaman website apachefriends.org.
 - Langkah 2. selanjutnya Install XAMPP Setelah selesai proses download, silakan eksekusi file instalasi XAMPP untuk menjalankan proses instalasi. Kemudian ikuti instruksi yang ada.
 - Langkah 3. Kemudian Aktifkan MySQL server menggunakan GUI XAMPP Saat Anda selesai menjalankan proses instalasi akan muncul panel kontrol XAMPP. Klik “Start” pada modul Apache dan MySQL.

Sampai di titik ini MySQL server sudah dapat berjalan dan sudah dapat dipakai untuk menjalankan perintah SQL.

2. Menginstal MySQL di sistem operasi Linux

Pada kali ini panduan cara install MySQL Server di Linux akan menggunakan Terminal sebagai contohnya. Anda bisa menggunakan cara ini atau anda ingin cara yang lebih mudah, dengan cara langsung mengunduh file instalasi XAMPP dan menginstallnya seperti contoh Windows di atas.

- Langkah 1. Hal yang pertama yang harus dilakukan adalah Install Server Web Apache

Sebelum proses instalasi MySQL dijalankan, ada baiknya untuk memperbarui repositorinya terlebih dahulu. Untuk memperbarui repositori pada Linux caranya cukup mudah, Anda tinggal membuka Terminal kemudian mengetikkan baris komentar berikut:

```
$ sudo apt-get update
```

Kemudian jalankan perintah berikut untuk menginstall Apache.

```
$ sudo apt-get install apache2
```

- Langkah 2, Mengecek Proses Instalasi Apache

Jika proses instalasi Apache sudah selesai dan berhasil dijalankan, Anda bisa mengeceknya dengan mengakses web browser kemudian menuliskan alamat IP server atau alamat domain.

```
http://<alamat_ip_address_server>
```

- Langkah 3, Instalasi MySQL

Proses instalasi MySQL server di Linux cukup menggunakan satu perintah saja.

```
$ sudo apt-get install mysql-server
```

Proses instalasi ini akan meminta Anda untuk memasukkan username dan password untuk akses ke dalam MySQL server.

- Langkah 4, Pengecekan Instalasi MySQL

Anda dapat memastikan apakah proses instalasi MySQL sudah berjalan baik atau belum dengan mengetikkan perintah di bawah ini.

```
$ sudo systemctl status mysql
```

Perintah di atas untuk mengecek apakah MySQL sudah berjalan atau belum. Jika belum, Terminal akan menampilkan status 'Error' atau 'Failed'. Anda dapat mengikuti artikel Cara Instal LAMP di Linux untuk melihat panduan lengkapnya.

3.2 CARA INSTALLASI SUBLIME

Sublime merupakan sebuah software text editor yang digunakan oleh programmer untuk membuat aplikasi. misalnya kita mau membuat aplikasi berbasis web maka yang dibutuhkan sebuah text editor untuk mengedit kode. langsung saja untuk cara menginstal sublime maka ikuti langkah-langkah sebagai berikut :

1. step 1 download terlebih dahulu file .exe di situs resminya sublime. Anda dapat mendownload sublime pada link ini <https://www.sublimetext.com/>.



Figure 3.1: Download Sublime

2. Klik save untuk mendownload file sublime

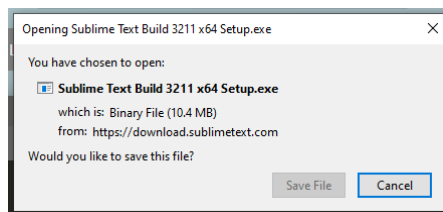


Figure 3.2: Klik Save

3. Setelah download selesai maka double klik file .exe pada hasil download tadi

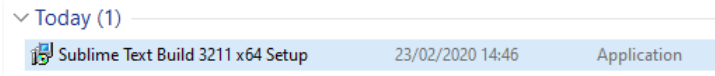


Figure 3.3: Klik File

4. Pilih folder untuk menyimpan file sublime

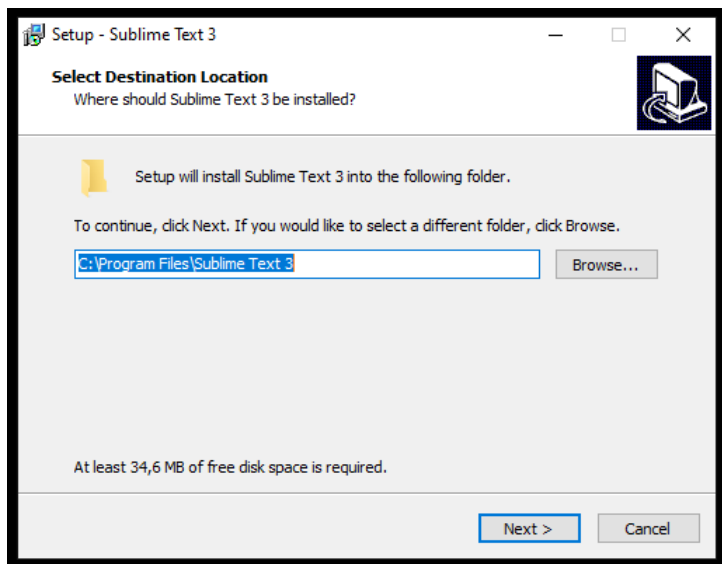


Figure 3.4: pilih folder penyimpanan file sublime

5. Centang tanda kotak seperti pada gambar

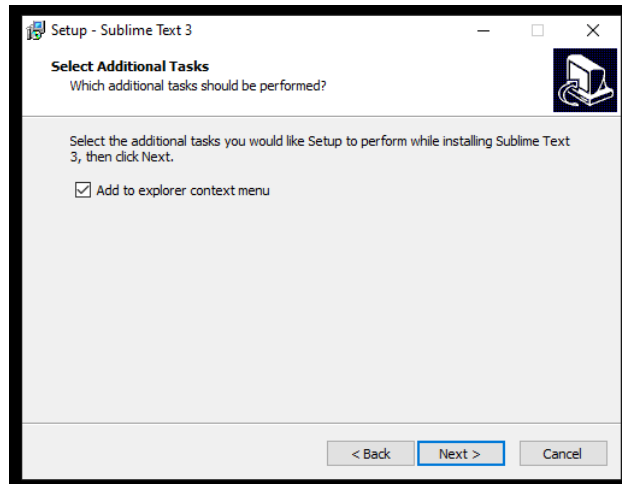


Figure 3.5: Centang kotak

6. Kemudian klik install

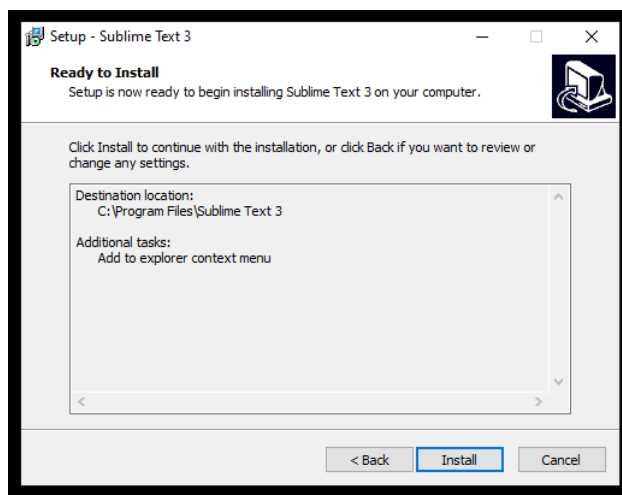


Figure 3.6: Klik Install

7. Sublime sudah bisa dijalankan

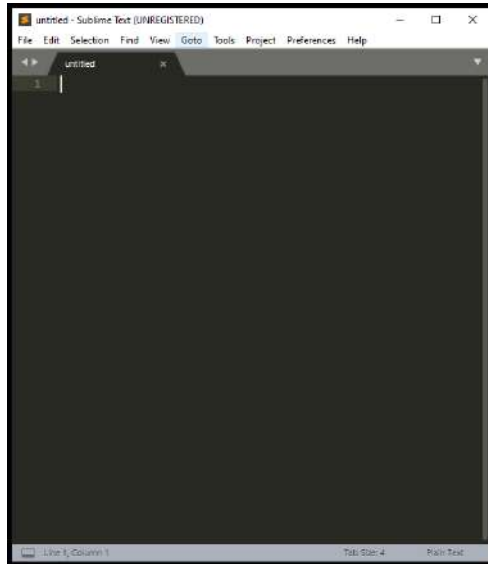


Figure 3.7: Install Selesai

3.3 INSTALLASI XAMPP

3.3.1 Proses Instalasi XAMPP di Windows

1. langkah pertama adalah Download aplikasi XAMPP terbaru

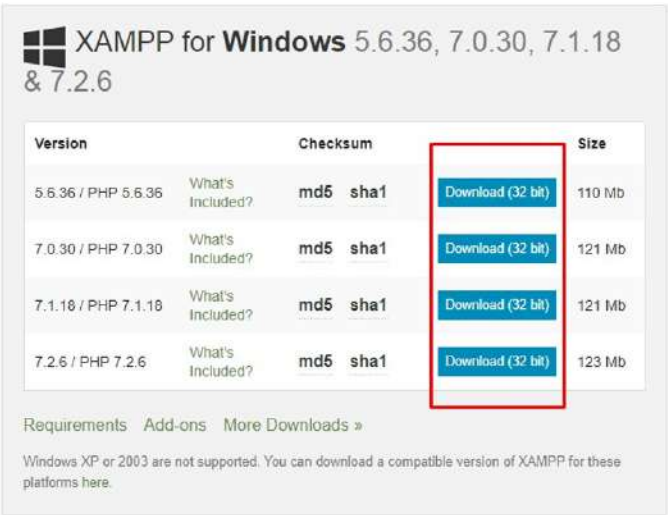


Figure 3.8: Download XAMPP

2. Setelah itu, klik file XAMPP yang sudah Anda download, kemudian akan muncul jendela “installer language” seperti di bawah ini:



Figure 3.9: Installer Language

3. Proses selanjutnya pilih bahasa. Pilih yang Bahasa Inggris (English). Klik OK.
4. pada suwaktu-waktun pada proses ini muncul pesan error. Jika ada, anda tidak perlu khawatir dan lanjutkan dengan klik OK dan YES.

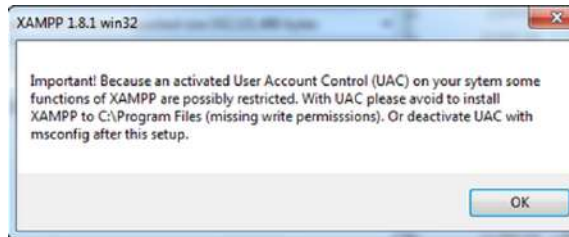


Figure 3.10: Eror 1

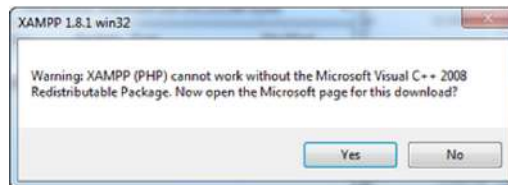


Figure 3.11: Error 2

5. Berikutnya akan muncul jendela yang isinya meminta Anda untuk menutup semua aplikasi yang sedang berjalan. Jika semua aplikasi sudah ditutup, maka klik tombol Next.



Figure 3.12: Klik next

6. Anda akan diminta untuk memilih aplikasi yang akan diinstal. Centang saja semua pilihan dan klik tombol Next.

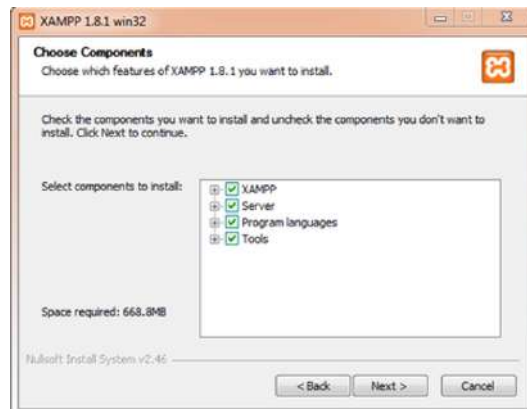


Figure 3.13: Centang App

7. Kemudian adalah proses untuk menentukan lokasi folder penyimpanan file-file dan folder XAMPP. Secara default akan diarahkan ke lokasi c:xampp. Namun jika Anda ingin menyimpannya di folder lain bisa klik browse dan tentukan secara manual folder yang ingin digunakan. Jika sudah selesai, lanjutkan dan klik tombol Install.

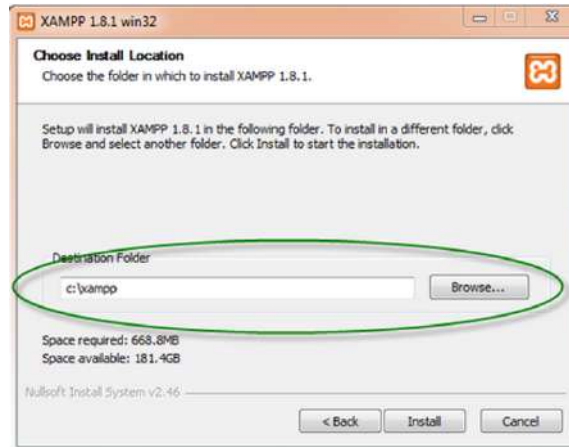


Figure 3.14: Penyimpanan folder

8. Tunggu beberapa saat hingga proses instalasi selesai. Apabila sudah muncul jendela seperti di bawah ini, klik tombol Finish untuk menyelesaikannya.



Figure 3.15: Klik finish

9. Berikutnya, akan muncul jendela dialog seperti gambar di bawah ini yang memastikan Anda apakah mau langsung menjalankan aplikasi XAMPP atau tidak. Jika ya, maka klik YES.



Figure 3.16: Klik yes

3.3.2 Menjalankan XAMPP

1. yang harus anda lakukan adalah Buka aplikasi XAMPP, anda dapat melakukannya melalui Start Menu atau Desktop, dan klik icon XAMPP. Atau, jika Anda membukanya begitu proses instalasi selesai maka klik Yes seperti yang terlihat pada gambar di atas.
2. Setelah XAMPP terbuka, silahkan klik tombol Start pada kolom Action sehingga tombol tersebut berubah menjadi Stop. Dengan mengklik tombol tersebut, artinya itulah aplikasi yang dijalankan, artinya aplikasi XAMPP sudah aktif. Biasanya jika saya menggunakan XAMPP, yang saya start

hanyalah aplikasi Apache dan MySQL, karena saya tidak memerlukan aplikasi seperti Filezilla, dan lain-lain.



Figure 3.17: Mengaktifkan XAMPP

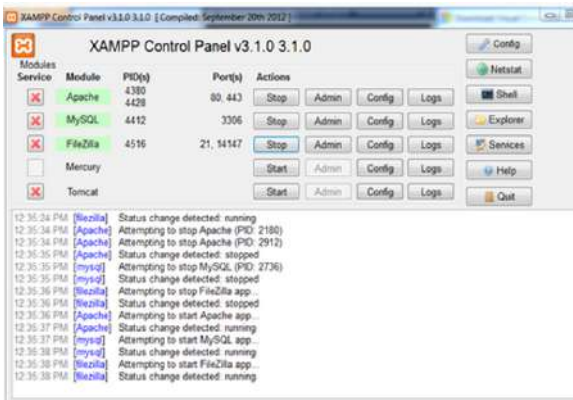


Figure 3.18: XAMPP aktif

3. Untuk melihat apakah XAMPP sudah berjalan atau belum, Sekarang bukalah browser, dan coba ketikkan `http://localhost/xampp` di address bar. Jika muncul tampilan seperti gambar di bawah ini, instalasi telah berhasil.

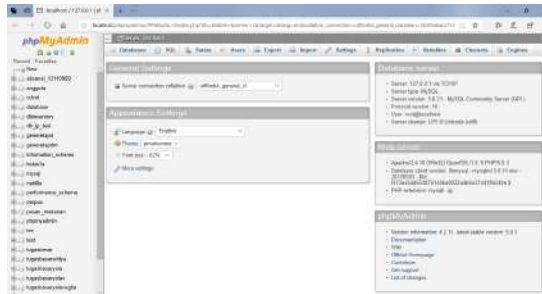


Figure 3.19: Tampilan Localhost

3.3.3 Membuat Database dengan XAMPP

Jika Anda telah menginstal XAMPP di computer anda, maka dapat dipastikan anda sudah bisa membuat database. Berikut tahapan untuk membuat database.

1. Klik dua kali pada menu XAMPP, XAMPP Control Panel akan terbuka. untuk mengaktifkan XAMPP, Anda bisa klik pada tombol Start yang ada di sebelah Apache. Setelah muncul tulisan 'Running' di sebelah Apache, Anda bisa klik tombol Start di sebelah MySQL. Tunggu MySQL menyala. Kalau kata 'Running' sudah ada di sebelah MySQL, berarti Apache dan MySQL sudah aktif.
2. Ada 2 cara untuk mengunjungi halaman selanjutnya, yaitu dengan mengeklik admin atau dengan membuka browser Anda dan ketik Jika Anda sudah mengkonfigurasinya dengan benar, Anda akan secara otomatis dibawa ke localhost/xampp.
3. Setelah itu, ketik localhost/phpmyadmin pada browser Anda dan Anda akan dibawa ke admin area phpMyAdmin.
4. Lalu Pilih Database di bagian atas dan Anda akan melihat pilihan Create New Database. Masukkan nama yang akan anda gunakan pada database baru Anda dan klik Create.

5. Setelah Anda memasukkan nama database baru, Anda akan melihat database tersebut muncul di bagian kiri admin area phpMyAdmin Anda. Jika Anda ingin menambahkan data pada database Anda, Anda tinggal klik pada nama database tersebut dan lakukan penambahan dengan cara meng-import database yang sudah ada atau menambahkan table dan columns Anda sendiri.

3.4 MEMBUAT DATABASE MENGGUNAKAN MySQL DATABASE MELALUI cPANEL

Salah satu fungsi phpMyAdmin adalah untuk membuat database sebuah website. Karena phpMyAdmin merupakan salah satu fitur pada cPanel, Anda bisa langsung mengaksesnya melalui cPanel website Anda.

1. Langkah pertama Login ke akun cPanel Anda dengan mengakses domainanda.com/cpanel. Kemudian scroll ke bawah dan pilih MySQL Databases di bagian Databases.

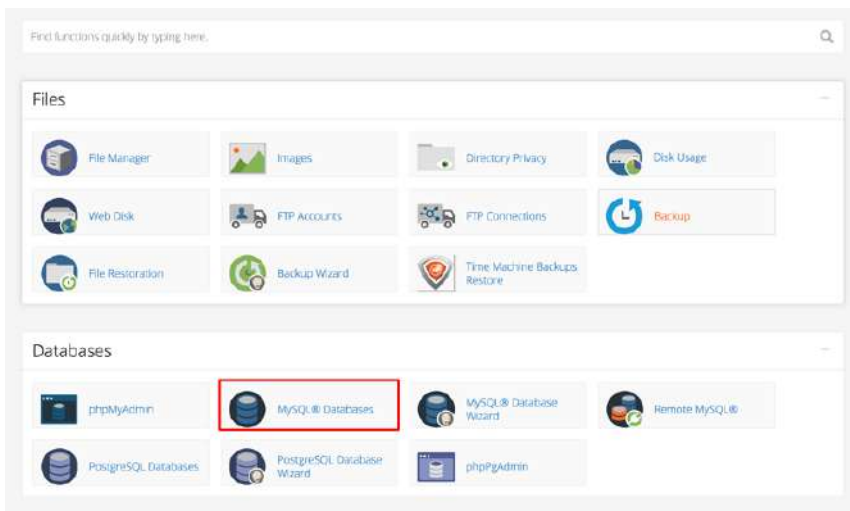


Figure 3.20: Membuat database baru dengan MySQL

2. Setelah itu, Anda akan melihat layar seperti di bawah ini. Pada tahap ini anda akan diminta untuk mengisi data New Database. Setelah Anda

mengisi nama database baru Anda, klik Create New Database.

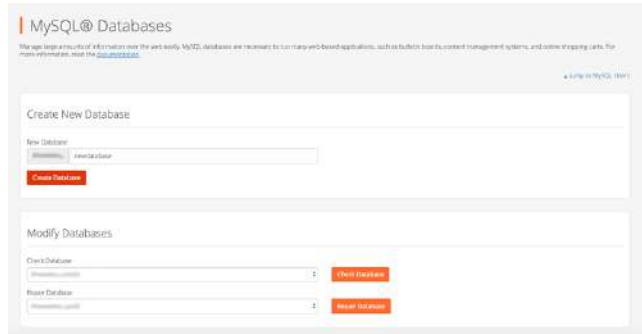


Figure 3.21: Database Baru MySQL

3. Apabila Anda sudah melihat layar seperti di bawah ini, berarti database Anda berhasil dibuat.



Figure 3.22: Database berhasil dibuat

4. langkah selanjutnya, Setelah Anda selesai membuat database dan user baru, scroll lagi ke bawah dan Anda akan menemukan Add User to Database. Lalu Pilih nama user dan database yang tadi Anda buat, kemudian klik Add.

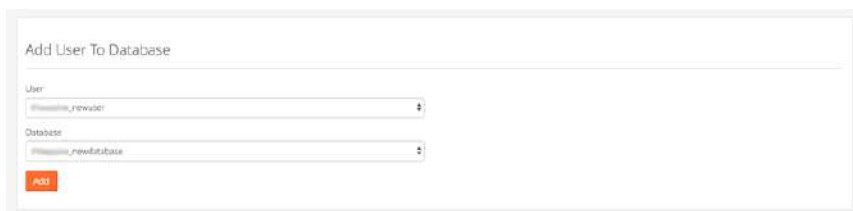


Figure 3.23: Add User to Database

5. Kemudian Anda akan melihat layar seperti di bawah ini. Agar lebih mudah, centang pilihan All Privileges. Setelah itu klik Make Changes.

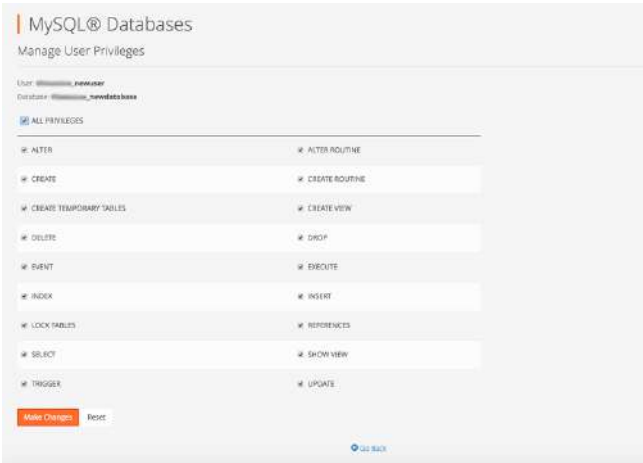


Figure 3.24: Manage Privilage

- 6. Apabila berhasil, Anda akan menemukan kotak hijau di bagian kanan atas layar Anda. Maka Anda berhasil membuat database baru Anda.
- 7. terakhir, Anda tinggal kembali ke area admin cPanel Anda. Kemudian klik phpMyAdmin di bagian Databases.
- 8. Anda dapat melihat area admin phpMyAdmin Anda.

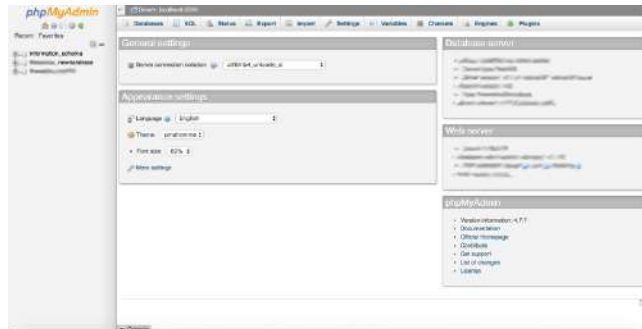


Figure 3.25: PhpMyAdmin

9. Anda dapat menghandle berbagai opsi untuk mengelola database Anda menggunakan phpMyAdmin di cPanel Anda. terdapat bebrapa opsi sebagai berikut:

(a) Structure

Menu ini berguna untuk mengorganisir schema, tabel, kolom, baris, dan lain lain.

(b) SQL

Menu ini digunakan untuk menjalankan query atau kueri pada database.

(c) Search

Untuk mencari kata-kata atau value di dalam tabel database.

(d) Query

SQL mendefinisikan seperangkat perintah, seperti SELECT, INSERT, UPDATE, DELETE, CREATE TABLE dan lain-lain.

(e) Export

Anda bisa meng-export database dengan format berbeda-beda seperti CSV, PDF, SQL, XML, Text, dan masih banyak lagi.

(f) Import

Anda bisa meng-import database dengan format berbeda-beda seperti OpenDocument Spreadsheet, CSV, SQL, ESRI Shape file, Mediawiki Table, XML.

(g) Operations

Ada beberapa jenis operations yang bisa Anda eksekusi di seluruh database dan tabel terpisah.

(h) Triggers

Sebuah trigger dikenal sebagai objek basis data yang terhubung dengan tabel, dan akan diaktifkan kapan pun terjadi suatu event. Misalnya insert, update atau delete terjadi pada tabel.

10. Anda sudah bisa melakukan import, export database serta mengosongkan dan menghapus database jika Anda inginkan. Kami akan membahasnya di bawah ini. [?]

3.4.1 Export database menggunakan phpMyAdmin

1. Klik Export di bagian atas. Kemudian Anda akan melihat layar seperti di bawah ini.

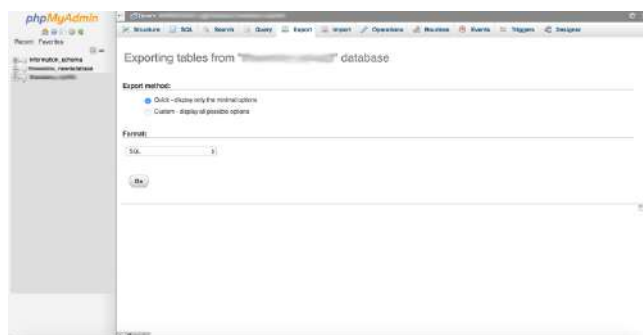


Figure 3.26: Export Database dari PhpMyAdmin

2. Anda bisa langsung mengklik Go untuk mengexport database Anda. Tetapi Anda juga bisa merubah formatnya terlebih dahulu. Klik saja kolom yang bertuliskan SQL dan itu akan mengeluarkan pilihan formatnya. [?]

3.4.2 Import database menggunakan phpMyAdmin

1. Pilih Import di bagian atas



Figure 3.27: Import Database dari PhpMyAdmin

2. Pilih Choose File dan pilih database yang ingin Anda import ke website Anda. File boleh dikompres ataupun tidak. Jika Anda memilih untuk mengupload file yang sudah dikompres, nama filenya harus berakhir dengan `.[format].[compression]`. Misalnya `namafile.sql.zip`. Setelah itu Anda tinggal scroll ke bawah dan click Go. [?]

3.4.3 Cara Mengelola Database di phpMyAdmin

Kegunaan utama dari phpmyadmin adalah untuk mengelola database yang akan anda buat. Berikut ini adalah beberapa fungsi phpMyAdmin yang dapat membantu Anda untuk melakukan itu. Ketika Anda membuka admin area phpMyAdmin, pilih database yang ingin Anda kelola di bagian kiri layar Anda. Di halaman yang terbuka Anda akan melihat daftar dengan tabel database, allowed actions, jumlah catatan, mesin penyimpanan, pemeriksaan, ukuran tabel dan overheadnya. Hal-hal yang bisa dilakukan yang dapat Anda lakukan pada tabel yang dipilih adalah:

1. Browse

Hanya tabel yang memiliki record atau catatan yang dapat Anda akses. Setelah Anda mengklik tombol Browse, sebuah halaman baru dengan daftar record di dalam tabel akan dibuka. Jika Anda ingin mengedit salah satu record yang ada, Anda bisa mengklik tombol Edit. Anda akan melihat struktur record dan Anda dapat mengubah nilai dari record tersebut. Setelah selesai dengan perubahan, Anda bisa mengirimkannya dengan mengklik tombol Go di bagian bawah halaman.

2. Structure

Dengan mengklik tombol Structure maka halaman baru yang menunjukkan struktur tabel database akan terbuka. Anda akan melihat nama field, jenis, collations, atribut, informasi tambahan, nilai default dan apakah nilai field dapat berupa NULL. Anda dapat mencari nilai yang berbeda dengan mengklik ikon tindakan yang sesuai. Selain itu, Anda dapat mengedit struktur lapangan atau menghapus bidang. Anda dapat menentukan indeks yang berbeda: Primary, Unique, Index dan Fulltext.

3. Search

Dengan tombol search Anda dapat membuat kueri penelusuran untuk tabel yang dipilih. Anda dapat menggunakan fungsi Query by example untuk melakukan pencarian. Cukup gunakan ke berbagai bidang untuk menyusun kueri penelusuran Anda dan klik tombol Go untuk menjalankannya. Misalnya, jika Anda ingin memvisualisasikan semua catatan dengan nilai bidang yang dimulai dengan "a" sebaiknya pilih dulu bidang yang ingin Anda tampilkan. Pilih operator LIKE dari menu drop-down dan masukkan nilai field yang sesuai "a%" (% adalah singkatan dari wildcard string). Klik tombol Go untuk melihat hasilnya.

4. Insert

Dengan menggunakan tombol Insert, Anda dapat memasukkan record di tabel database Anda. Setelah Anda mengisi nilai klik yang sesuai pada tombol Go dan record baru akan dimasukkan.

5. Empty

Tombol empty memperbolehkan Anda mengosongkan sebuah tabel database, menghapus data, dan menyimpan tabel kosong.

6. Drop

Tombol drop akan menghapus seluruh tabel dan isi recordnya.[?]

3.4.4 Menghapus Database di phpMyAdmin

1. Empty Database di phpMyAdmin untuk menghapus database terdapat dua cara yang dapat Anda gunakan, yaitu dengan empty (mengosongkan) dan drop (menghapus) data tersebut.

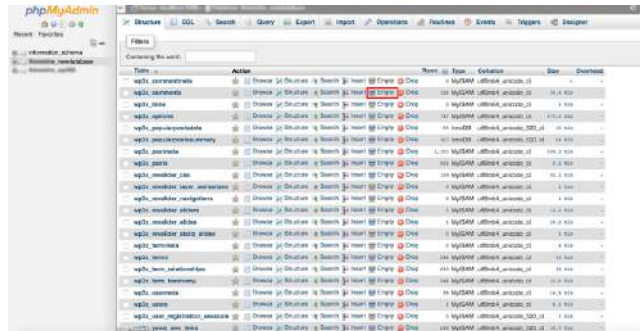


Figure 3.28: Empty Database di phpMyAdmin

- Klik pada database yang ingin Anda edit.
 - Kemudian di row yang ingin Anda kosongkan
 - klik pilihan Empty
 - Setelah Anda mengklik itu, mereka akan menanyakan apakah Anda ingin benar-benar mengosongkan data tersebut
 - Klik OK
 - Jika Anda memperhatikan angka yang ada kolom Rows
 - setelah Anda klik Empty dan OK, maka angka tersebut akan berubah menjadi 0.
2. Menghapus dengan cara Drop Database di phpMyAdmin Apabila Anda ingin menghapus sebuah data, Anda dapat memilih pilihan Drop. Pada data yang akan Anda hapus, klik pilihan Drop. Sama seperti sebelumnya, mereka akan menanyakan apakah Anda yakin ingin menghapus data tersebut. Klik Ok. Setelah itu, row tersebut akan hilang dari daftar data Anda.

3.4.5 Mengganti Nama Database di phpMyAdmin

Apabila anda melakukan kesalahan pada saat memberi nama database anda dan Anda ingin mengganti nama database Anda, Anda bisa melakukannya dengan phpMyAdmin. Anda tinggal pilih database mana yang ingin Anda ganti namanya di bagian kiri layar. Setelah itu klik Operations di bagian atas. Anda akan melihat tampilan layar seperti di bawah ini.

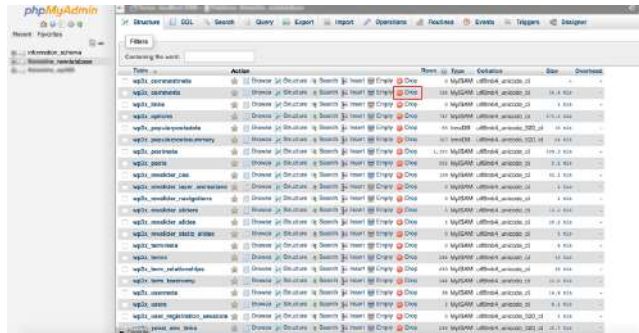


Figure 3.29: Drop Database di phpMyAdmin

caranya dengan memasukkan nama database yang ingin Anda gunakan dan klik Go. Anda tidak perlu mencentang pilihan adjust privilege.

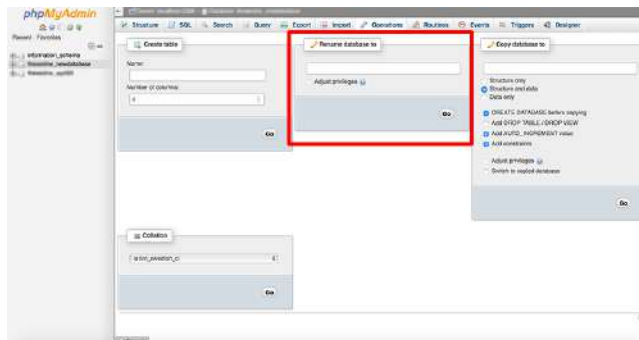


Figure 3.30: Rename Database

3.4.6 Membuat Database Baru di phpMyAdmin

Tidak hanya melalui cPanel, Untuk membuat database anda bisa secara langsung pada phpmyadmin. Yang akan anda perlukan adalah XAMPP.

XAMPP adalah sebuah cross-platform. Nama itu sendiri merupakan singkatan dari Cross-Platform (X), Apache (A), MariaDB (M), PHP (P), dan Perl (P). Tool yang satu ini mempermudah developer untuk membuat sebuah server web local untuk testing dan keperluan deployment. Semua yang dibutuhkan untuk membuat sebuah web server, yaitu aplikasi server (Apache), database

(MariaDB), dan bahasa pemrograman (PHP) sudah termasuk dalam satu file. Karena XAMPP adalah sebuah cross platform, tool ini bekerja di Mac, Linux, dan Windows.[?]

3.4.7 Upload Database ke phpMyAdmin

Apabila Database telah dibuat, Berikut adalah langkah-langkah untuk mengeksport databasw ke phpmyadmin:

1. Export database yang ada di localhost Anda
2. Kemudian import database tersebut ke phpMyAdmin cPanel Anda

Anda bisa melihat cara export dan import database di bagian atas artikel ini.

3.4.8 Membuat Table

1. Untuk membuat table pada database, tentukan terlebih dahulu database yang akan digunakan. Sehingga muncul pesan "No tables found in database, pesan ini muncul karena belum ada satupun table yang pernah dibuat di database "mydatabase" .

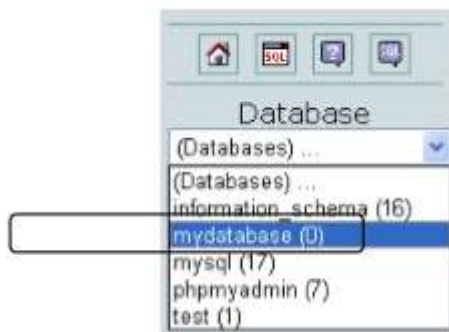


Figure 3.31: Memilih database mydatabase

2. Kemudian tentukan nama table beserta jumlah field/kolom yang akan dibuat, misalnya nama tablenya adalah mahasiswa_tb dan jumlah kolomnya ada 5, lalu tekan tombol go. dan lebar tipe data yang dibutuhkan.

Kemudia tentukan jenis storage enginenya, dalam contoh berikut ini digunakan jenis MyISAM. Tentukan nama field, tipe data, dan lebar tipe data yang dibutuhkan. Tentukan pula jenis storage enginenya, dalam contoh berikut ini digunakan jenis MyISAM.



Figure 3.32: membuat table siswa

- 3. Langkah selanjutnya tekan tombol Save, atau tekan tombol Go jika masih ingin menambah beberapa field baru.

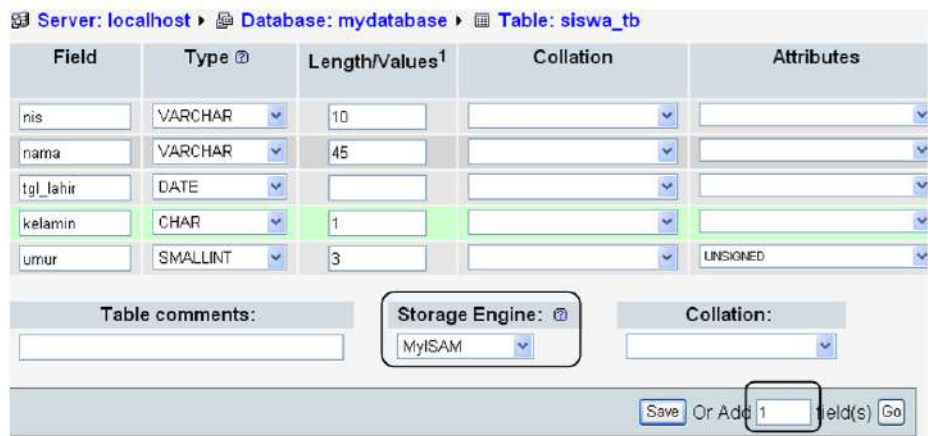


Figure 3.33: membuat Struktur Table

4. Kemudian akan muncul konfirmasi seperti berikut

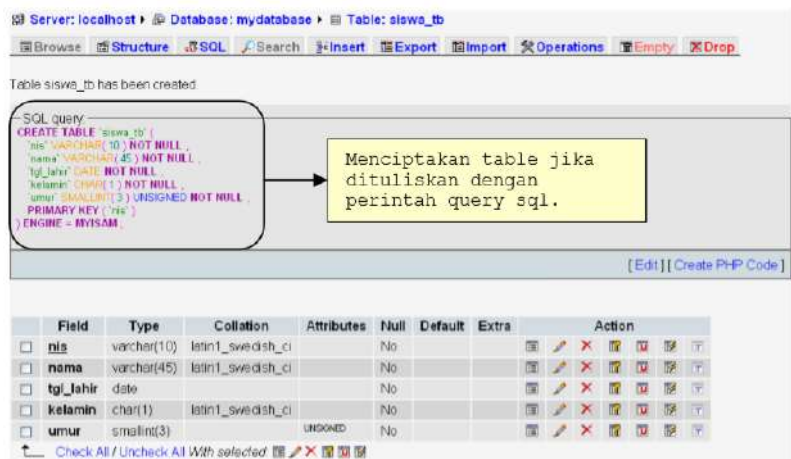


Figure 3.34: Konfirmasi Creat Table

3.4.9 Insert Table

Insert table adalah proses memasukkan data (baris-baris record) ke dalam table.

- 1. Saat anda akan memasukkan data ke dalam table, pilih terlebih dahulu salah satu nama database dimana table yang akan dipilih berada, kemudian klik pada salah satu table yang akan ditambahkan datanya dengan perintah insert.

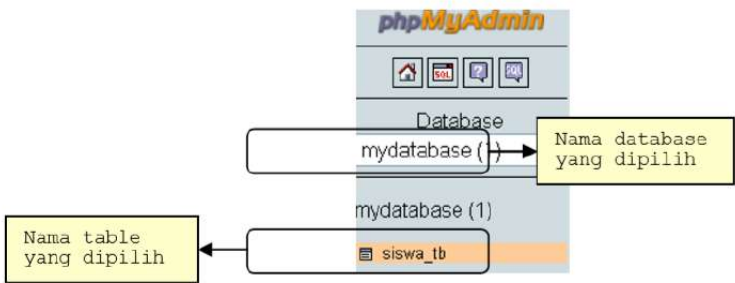


Figure 3.35: Memilih Database dan Table

2. setelah keluar struktur tablenya kemudian tekan tombol insert seperti tampak berikut ini.

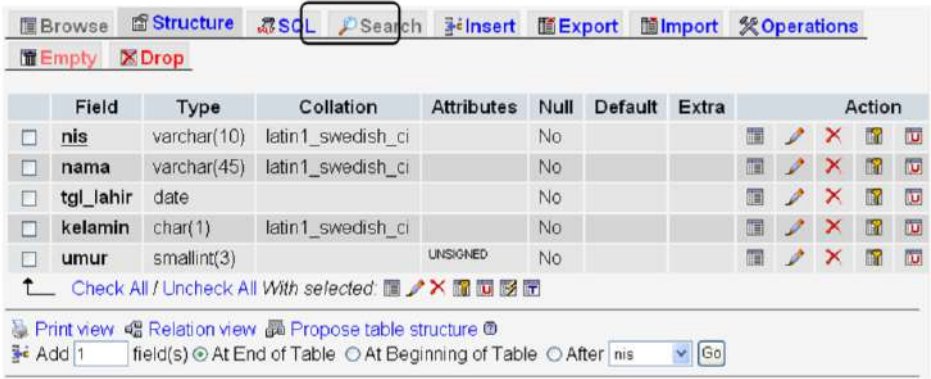


Figure 3.36: Memilih Insert

3. Isi lengkap datanya sesuai form inputan yang telah diberikan. Setelah masing-masing field terisi lengkap, silahkan tekan tombol Go. Terdapat tombol Reset yang berfungsi untuk membersihkan atau membatalkan isian data field yang pernah dituliskan.

The screenshot shows the cPanel database interface for editing a table named 'siswa_tb'. It features two tables for data entry. The first table has fields: nis (varchar(10)), nama (varchar(45)), tgl_lahir (date), kelamin (char(1)), and umur (smallint(3) unsigned). The second table is identical. At the bottom, there are options to 'Insert as new row' and 'Go back to previous page', along with 'Go' and 'Reset' buttons.

Field	Type	Function	Null	Value
nis	varchar(10)			1234567890
nama	varchar(45)			Dodit Suprianto
tgl_lahir	date			2007-07-26
kelamin	char(1)			L
umur	smallint(3) unsigned			32

Field	Type	Function	Null	Value
nis	varchar(10)			2583691478
nama	varchar(45)			Rini Agustina
tgl_lahir	date			2007-07-27
kelamin	char(1)			P
umur	smallint(3) unsigned			28

Insert as new row and then Go back to previous page
Go Reset

Figure 3.37: Mengisi Data Pada Tabel

4. Apabila data telah dimasukkan maka akan muncul konfirmasi seperti berikut ini:

The screenshot shows the cPanel database interface after executing an SQL query. A yellow banner at the top indicates 'Inserted rows: 2'. Below this, the SQL query is displayed in a text area. A callout box with an arrow points to the query, containing the text 'Memasukkan data dengan perintah Query SQL INSERT.'.

Server: localhost Database: MyDatabase Table: siswa_tb

Inserted rows: 2

```
SQL query
INSERT INTO 'MyDatabase' 'siswa_tb' (
  'nis',
  'nama',
  'tgl_lahir',
  'kelamin',
  'umur'
) VALUES (
  '1234567890', 'Dodit Suprianto', '2007-07-26', 'L', '32'
), (
  '2583691478', 'Rini Agustina', '2007-07-27', 'P', '28'
);
```

Memasukkan data dengan perintah Query SQL INSERT.

[Edit] [Create PHP Code]

Figure 3.38: input data

3.4.10 Menampilkan Table

- 1. Untuk menampilkan isi table (baris-baris record), bisa menggunakan tombol browse, tapi sebelumnya aktifkan terlebih dahulu database serta nama tablenya, seperti cara sebelumnya .

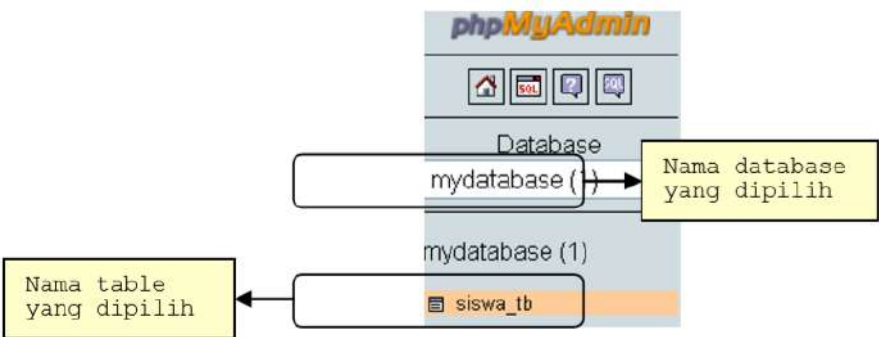


Figure 3.39: Memilih Database dan Table

- 2. Setelah memilih nama table, kemudian tekan tombol browse , sehingga yang tampak adalah seperti berikut ini :

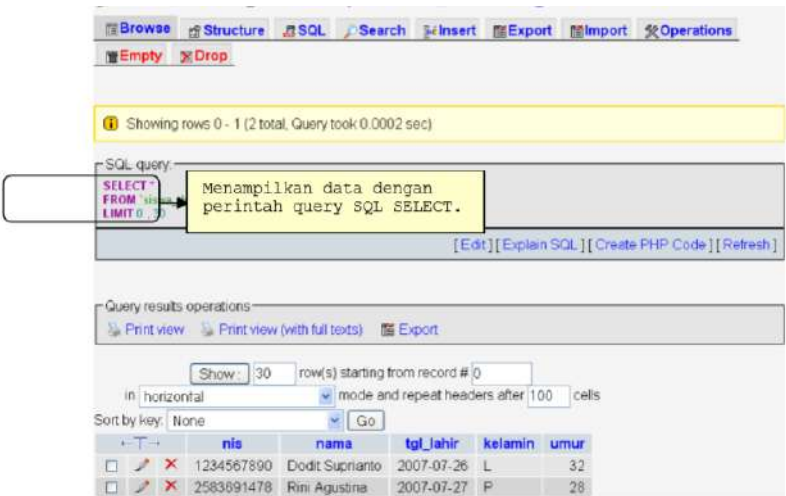


Figure 3.40: Konfirmasi Perintah Select

3.4.11 Ekspor Struktur Database

PHPMyAdmin menyediakan fasilitas ekspor, yaitu membuat struktur database secara menyeluruh ataupun sebagian, termasuk isi tablenya. Strukturnya, ditulis dengan menggunakan bahasa SQL.

Hasil ekspor bisa disimpan dalam file dengan berbagai format misalnya: CSV, CSV for MS Excel, Microsoft Excel 2000, Microsoft Word 2000, PDF, SQL, atau XML. Struktur database dan table juga dapat ditampilkan dalam format web tanpa harus disimpan ke dalam file.

1. Untuk meng-ekpor keseluruhan database, maka pilih ikon home.



Figure 3.41: Menuju Ke Halaman Home

2. Ikon home akan membawa anda menuju ke halaman web utama PHP-MyAdmin, kemudian lanjutkan dengan memilih menu Export, seperti berikut :

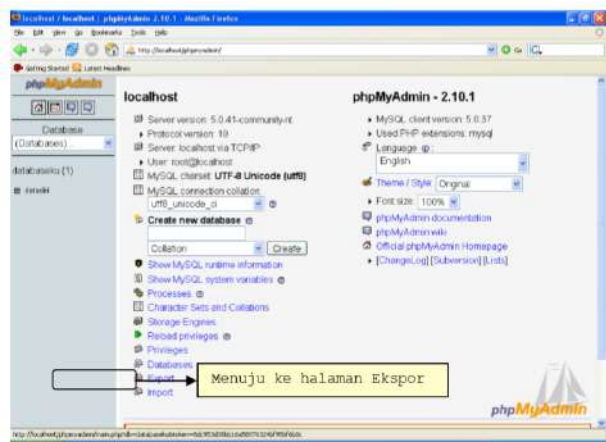


Figure 3.42: Menu Ke Halaman Ekspor

3. Pilih salah satu database yang akan diekspor, dan tentukan juga format yang akan digunakan (penulis menggunakan format standarnya yaitu format SQL).

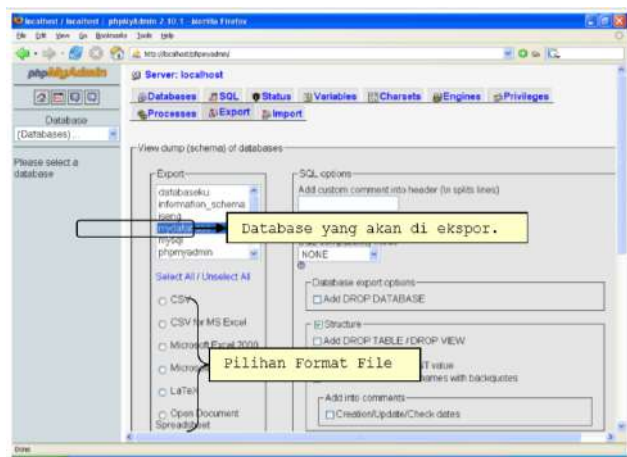


Figure 3.43: Caption

4. Adapun Yang perlu diperhatikan sebelum mengekspor database

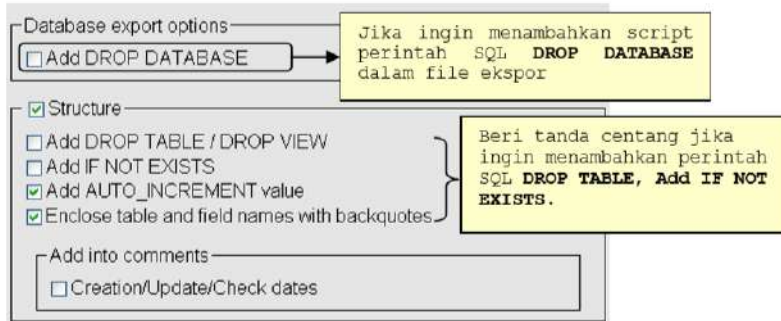


Figure 3.44: Properti Tambah Sebelum Ekspor

5. Apabila hasil ekspor disimpan dalam bentuk file, maka beri tanda centang pada pernyataan Save as file. Tetapi jika hanya menampilkan script SQL-nya saja, maka pernyataan Save as file jangan diberi tanda centang. Kalau sudah memilih salah satu pilihan tersebut maka tekan tombol Go. Seperti tampak berikut ini :

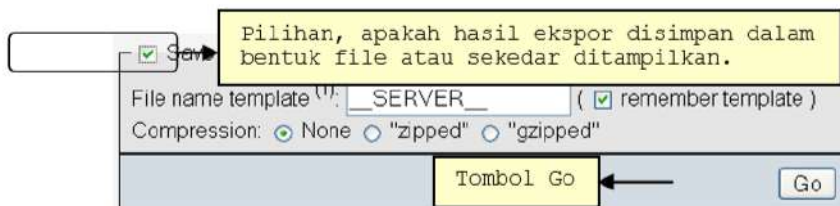


Figure 3.45: Pilihan Apakah Diekspor Ke File

6. Apabila hasil ekspor hanya ditampilkan dalam halaman web saja tanpa perlu disimpan dalam bentuk file, maka hasilnya akan tampak seperti berikut ini :

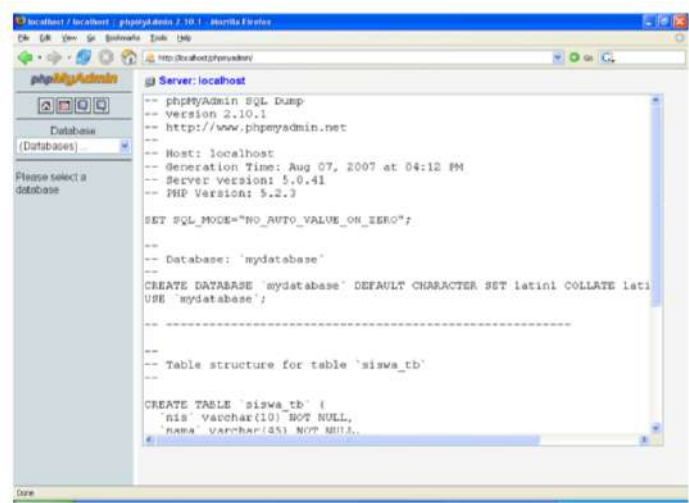


Figure 3.46: Jika hasil ekspor sekedar ditampilkan saja.

7. Apabila hasil ekpor disimpan dalam bentuk file maka akan muncul form yang meminta kita untuk menyimpan file tersebut. Seperti tampak berikut ini :

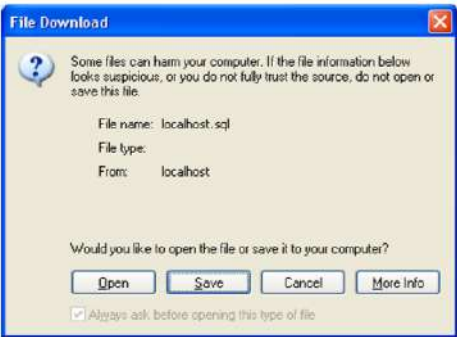


Figure 3.47: Konfirmasi Jika Hasil Ekspor Disimpan Ke File

3.4.12 Impor Struktur Database

1. Selain fasilitas ekspor, PHPMyAdmin juga menyediakan fasilitas impor yaitu mengambil struktur database dan table yang berasal dari sebuah

3.4 MEMBUAT DATABASE MENGGUNAKAN MySQL DATABASE MELALUI cPANEL

123

file (terutama file yang berekstensi *.sql). Untuk melakukan proses impor pilih ikon home.



Figure 3.48: Kembali Ke Menu Utama

2. Lalu klik sekali pada menu Import untuk menuju ke halaman import.

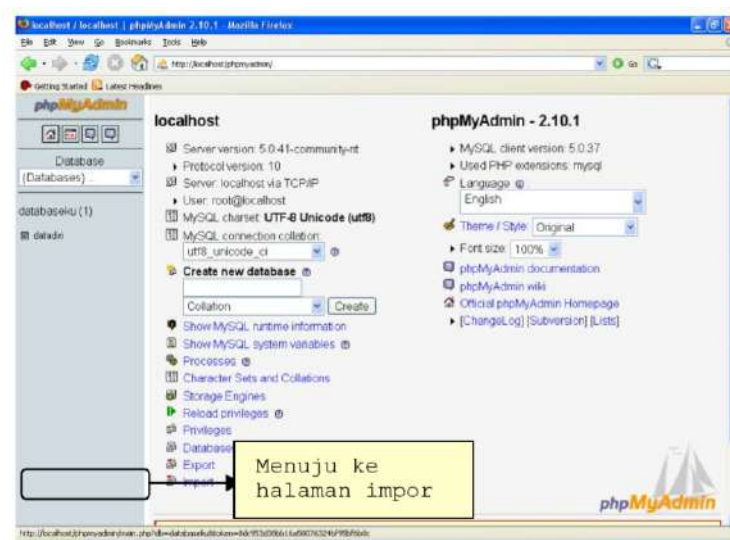


Figure 3.49: Menu Ke Halaman Import

3. Untuk meng-impor struktur database dan table yang berasal dari file luar sangatlah mudah, anda cukup menekan tombol browse untuk mencari di mana lokasi file yang akan di-impor berada (file berekstensi *.sql), setelah ditemukan, lanjutkan dengan menekan tombol Go untuk mengeksekusi file tersebut.

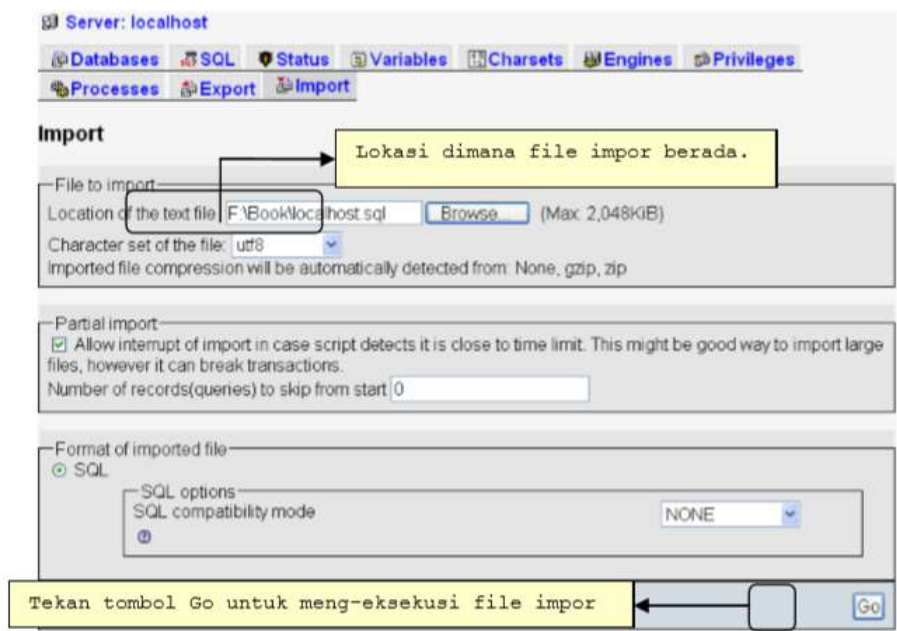


Figure 3.50: Membuka file ekspor

4. Apabila proses impor berhasil, maka akan keluar pesan konfirmasi bahwa proses impor telah sukses, seperti tampak berikut ini :

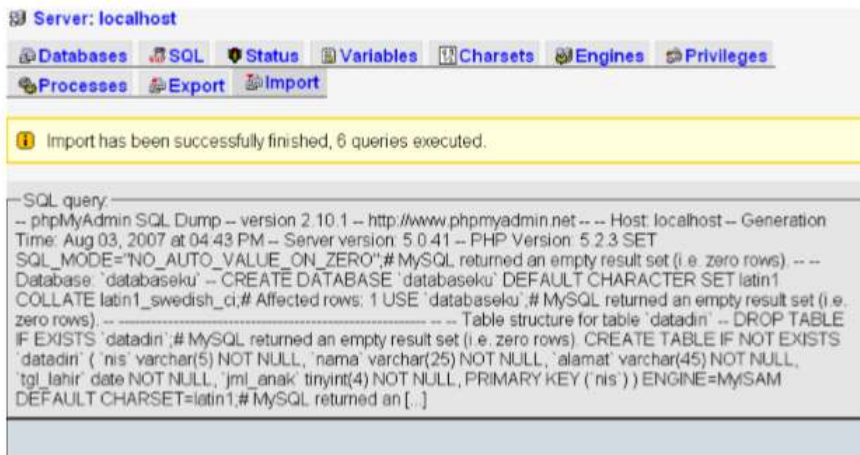


Figure 3.51: Konfirmasi hasil impor.

3.4.13 Menghapus Table dan Database

menghapus beberapa table, semua table, bahkan menghapus database sekalipun dapat dilakukan sangat mudah, yaitu dengan menggunakan perintah Query SQL Drop.

Untuk menghapus salah satu database maka pilih terlebih dahulu salah satunya, lalu pilih menu Drop. Jika mau menghapus table maka pilih salah satu table yang akan dihapus kemudian pilih menu drop.

Sebagai contoh akan dihapus database yang bernama "databaseku". Pilih terlebih dahulu database "databaseku" kemudian tekan menu "Drop".

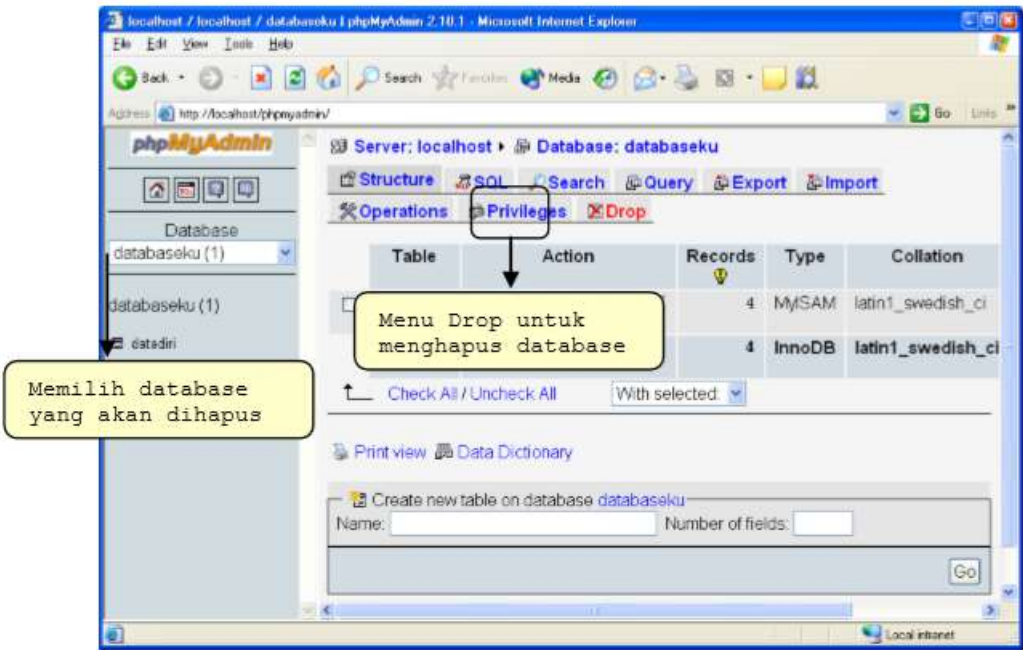


Figure 3.52: Menghapus Database dengan menu Drop

Setelah itu tekan tombol "Go", sehingga muncul jendela peringatan untuk memastikan apakah anda akan benar-benar menghapus database, Jika yakin makan lanjutkan dengan menekan tombol "OK".

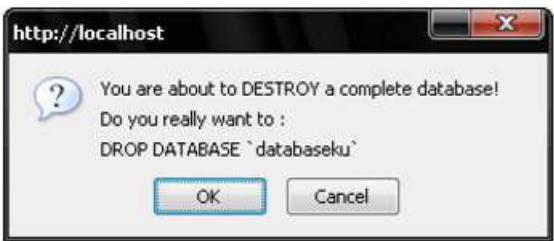


Figure 3.53: Konfirmasi Penghapusan Database

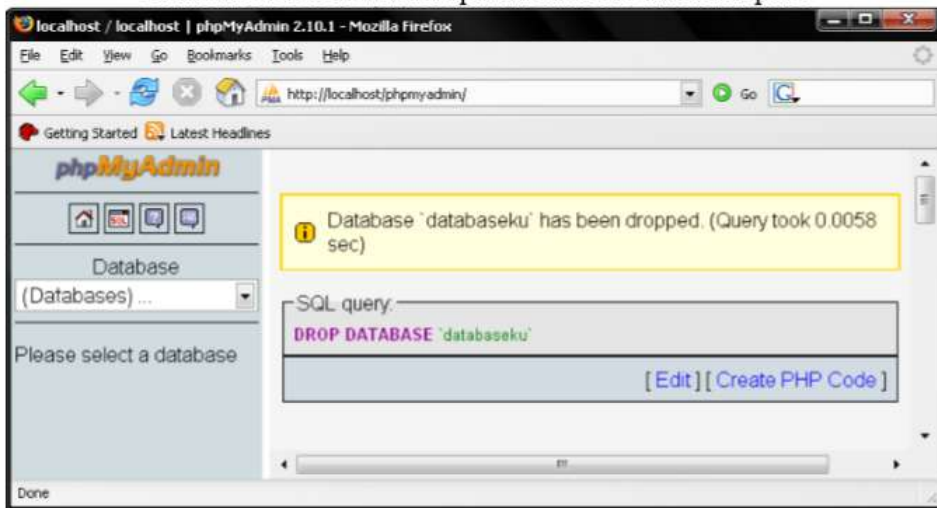


Figure 3.54: Konfirmasi bahwa penghapusan database berhasil

3.4.14 Membuat User dan Hak Akses

1. Untuk membuat user account baru beserta batasan aksesnya dapat menggunakan menu privileges. Langkah pertama adalah menuju ke halaman utama PHPMyAdmin dengan cara memilih ikon home seperti sebelumnya. Kemudian pilih menu privileges, seperti tampak gambar di bawah ini:

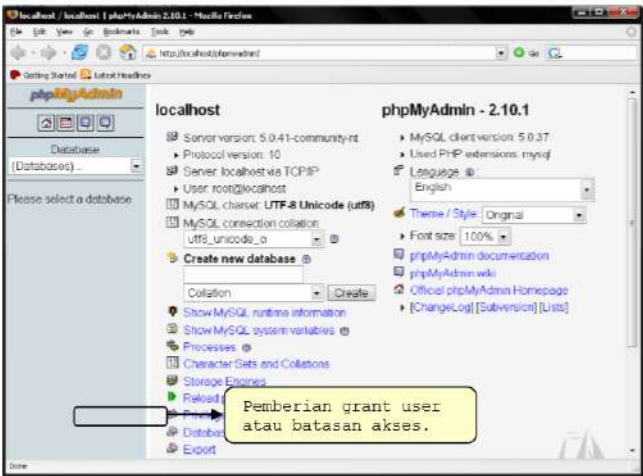


Figure 3.55: Menu Privillage

2. Kemudian pilih menu Add New User untuk menambah user baru ke server MySQL.

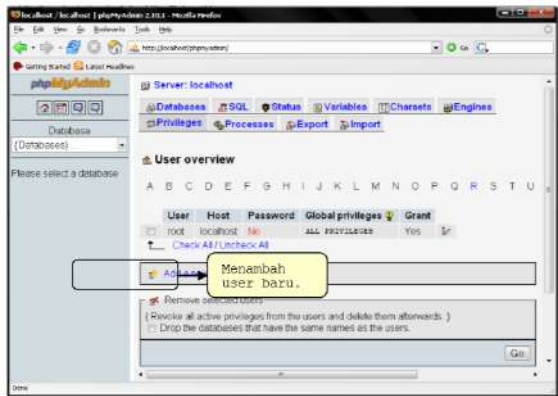


Figure 3.56: Menu Add new User Untuk Menambah User Baru

3. Selanjutnya Menentukan User name, Host dan Password yang akan digunakan sebagai sarana login ke database MySQL. Field password boleh dikosongkan yang berarti untuk mengakses MySQL tidak membutuhkan password.

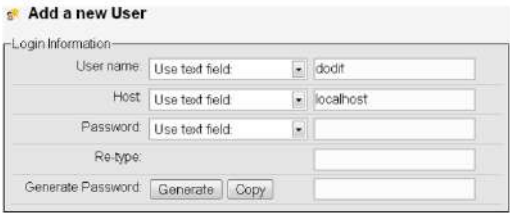


Figure 3.57: Menentukan User name, Host dan Password MySQL

4. Memberikan batasan hak akses kepada user bersangkutan, antara lain akses Data (SELECT, INSERT, UPDATE, DELETA dan FILE), akses Structure (CREATE, ALTER, INDEX, DROP, dan lain-lain). Sebagai contoh dipilih beberapa item hak akses seperti tampak gambar di bawah ini, kemudian tekan tombol Go untuk melanjutkan proses :

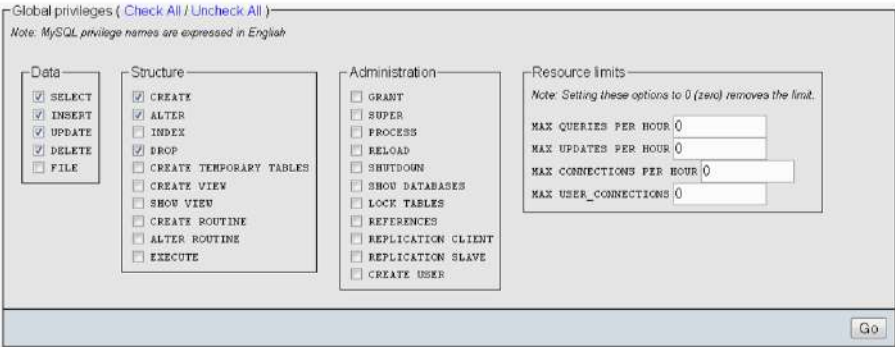


Figure 3.58: Isin Hak Akses User

5. Konfirmasi pembuatan account user baru disertai dengan batasan aksesnya.

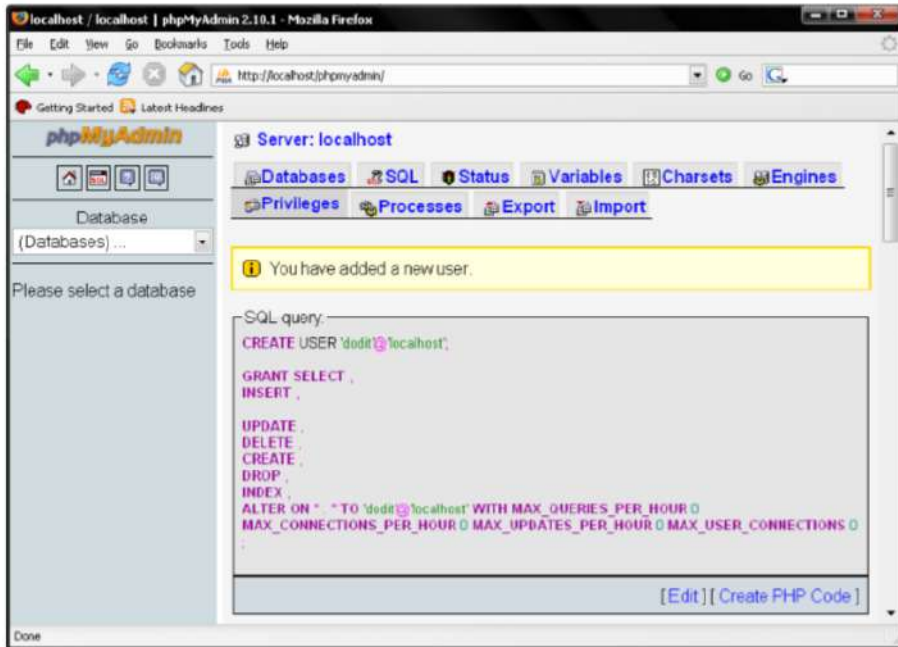


Figure 3.59: Konfirmasi Pembuatan User Account Baru

Masih banyak fitur-fitur PHPMyAdmin yang bisa digunakan, contohnya seperti cara merubah struktur table atau database, namun tidak sempat dipaparkan di buku ini [10].

PERANCANGAN APLIKASI

4.1 PERANCANGAN APLIKASI

4.1.1 Use Case

4.1.1.1 Pengertian Use Case

Use case atau diagram use case merupakan pemodelan untuk melakukan (*behaviour*) sistem informasi yang akan dibuat. Sedangkan untuk penjabaran alur kinerja dari setiap use case akan ditulis di dalam skenario. [3]

4.1.1.2 Kelebihan dan Kekurangan Use Case

1. Kelebihan

- Use Case dapat digunakan ulang di dalam sebuah proyek
- Use Case dapat digunakan sebagai dasar untuk pengukuran, penjadwalan atau validasi
- Use Case tidak menggunakan bahasa yang khusus
- Use Case telah terbukti mudah dimengerti oleh pebisnis, jadi merupakan sebuah jembatan penghubung bagi pengembang *software* dengan pemakai

2. Kekurangan

- Template Use Case tidak menjamin kejelasan sistem, hal ini tergantung dari kemampuan yang dimiliki penulisnya.

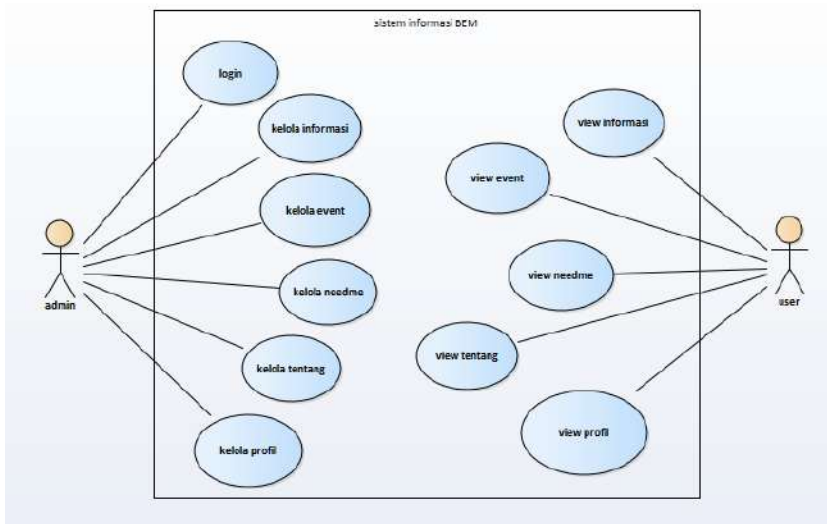


Figure 4.1: Use Case

4.1.2 Activity Diagram

Activity diagram menggambarkan *work flow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Diagram ini memperlihatkan aliran dari suatu aktivitas lainnya dalam suatu sistem. [9]

bahasa pemodelan UML termasuk bahwa diagram ini biasanya mudah dipahami baik untuk analis dan pemangku kepentingan. Dalam UML untuk IT Business Analyst, "Diagram aktivitas adalah yang paling berguna untuk menggambarkan alur kerja karena mudah dimengerti. baik untuk pengguna akhir". Karena mereka termasuk di antara diagram yang paling mudah digunakan, mereka umumnya dianggap sebagai alat penting dalam repertoar analis. Selain itu, seperti yang disebutkan di atas, diagram aktivitas memungkinkan seorang analis untuk menampilkan berbagai kondisi dan aktor dalam alur kerja melalui penggunaan *swimlane*. Namun demikian, *swimlanes* bersifat opsional karena kondisi tunggal atau aktor biasanya ditampilkan tanpa mereka. Perubahan tambahan memungkinkan diagram untuk lebih mendukung perilaku berkelanjutan dan berkelanjutan aliran data. Spesifikasi UML secara signifikan memperpanjang fitur dan skala diagram aktivitas di luar klasifikasi awal mereka sebagai kasus khusus diagram negara.

diagram ini berguna dalam metode berikut:

- Aturan Bisnis
- Fungsi yang terjadi secara paralel
- Rantai rumit dari beberapa kasus penggunaan
- Aliran perangkat lunak dan konfigurasi
- kontrol logika dengan poin penilaian dan aliran alternatif
- Kasus penggunaan tunggal.

1. Activity Diagram Login

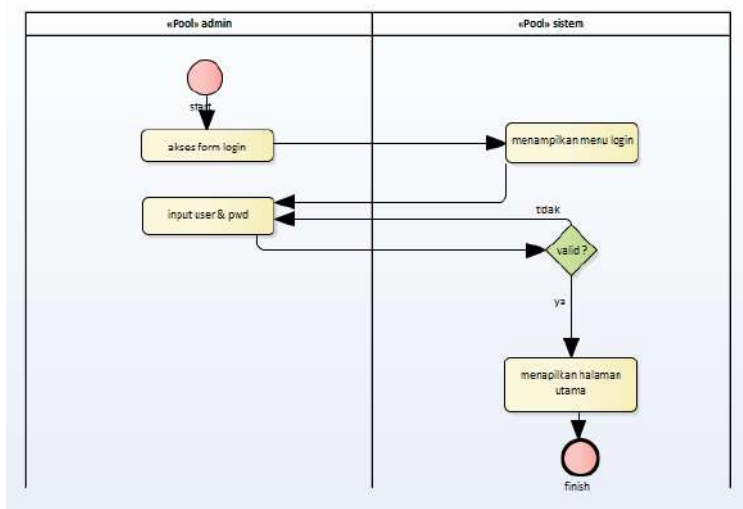


Figure 4.2: Activity Diagram Login

2. Actifty Diagram Kelola Info

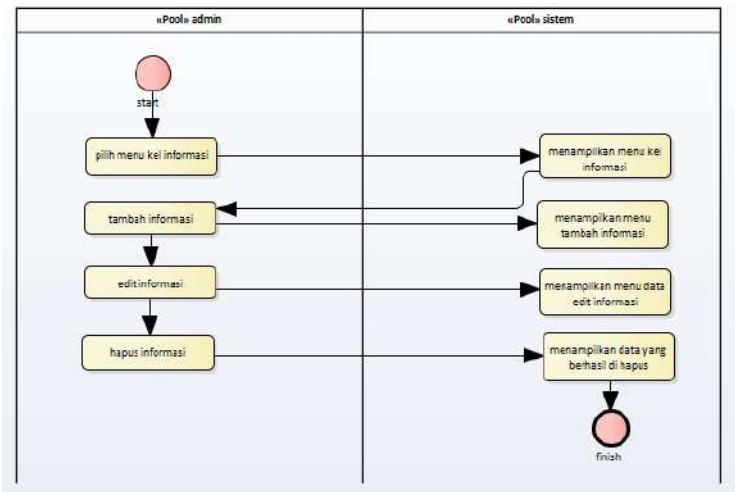


Figure 4.3: Actifty Diagram Kelola Info

3. Actifty Diagram Kelola Event

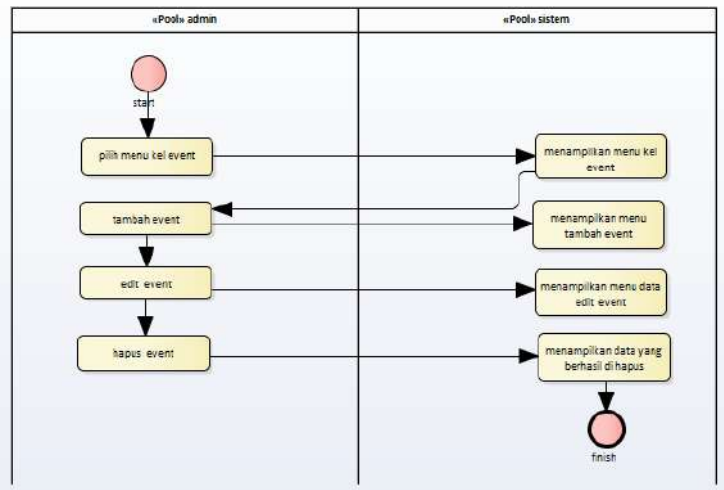


Figure 4.4: Actifty Diagram Kelola Event

4. Activity Diagram Kelola Needme

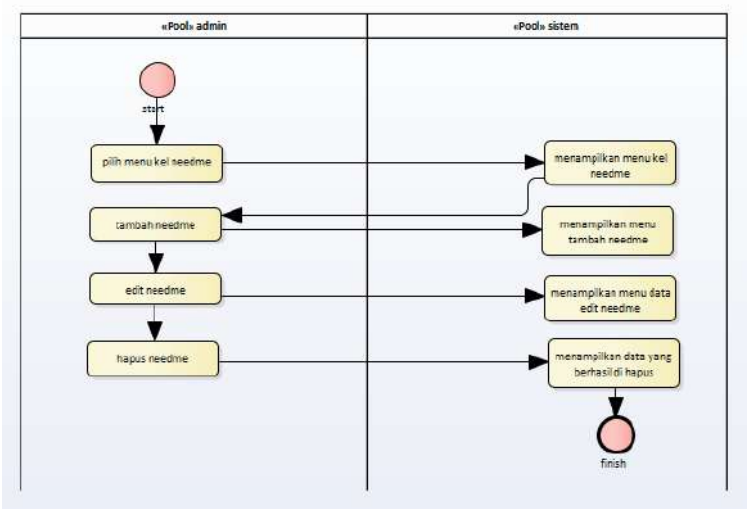


Figure 4.5: Actiftiy Diagram Kelola Needme

5. Activity Diagram Kelola Tentang

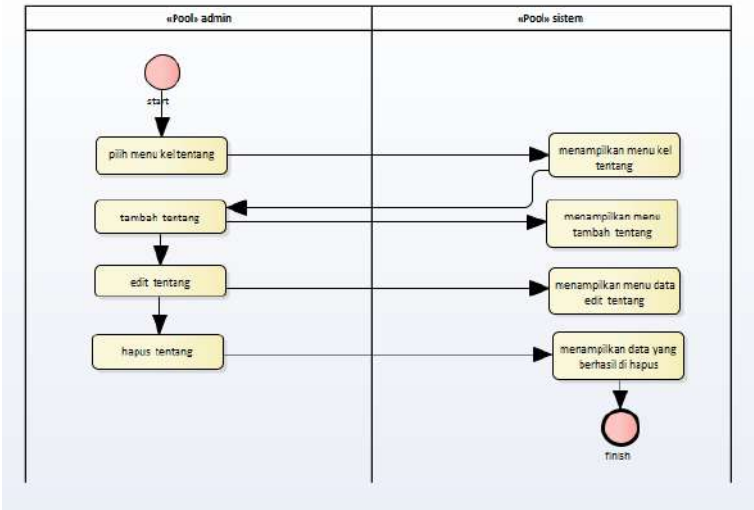


Figure 4.6: Actiftiy Diagram Kelola Tentang

6. Actifty Diagram Kelola Profil

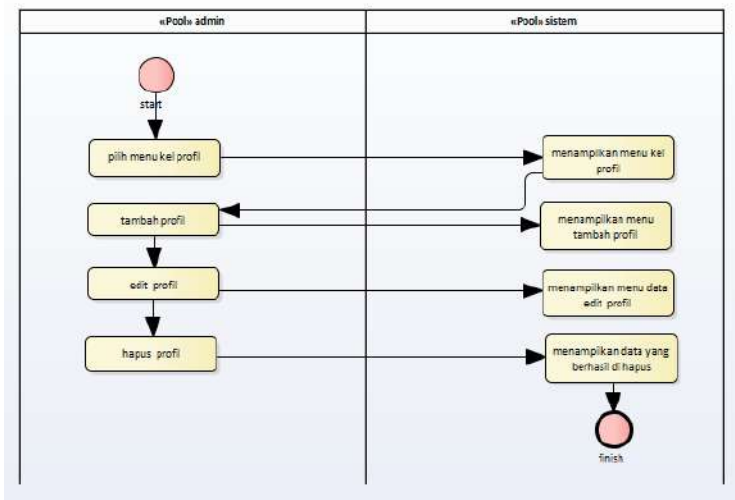


Figure 4.7: Actifty Diagram Kelola Profil

7. Actifty Diagram View Informasi

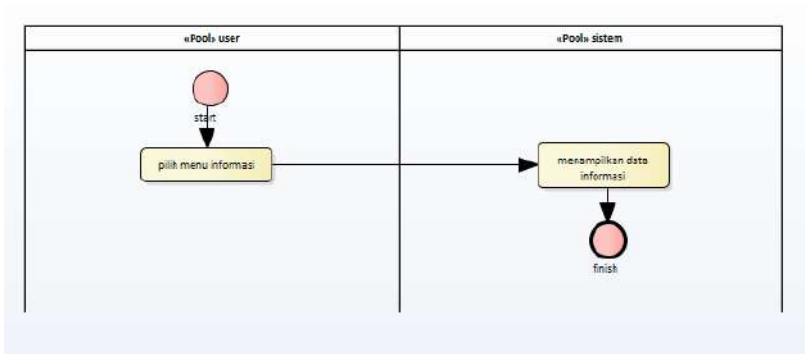


Figure 4.8: Actifty Diagram View Informasi

8. Actifty Diagram View Event

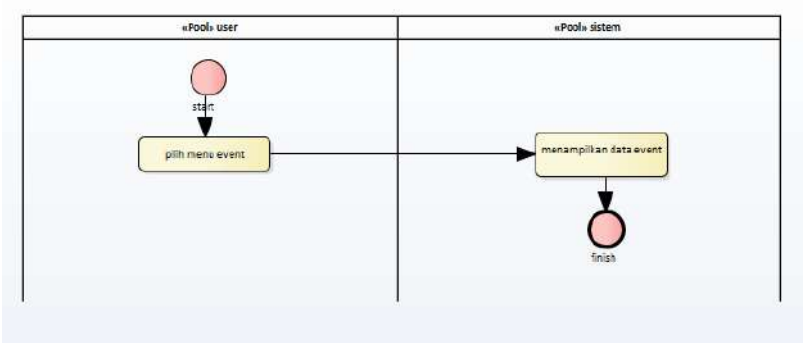


Figure 4.9: Actifty Diagram View Event

9. Actifty Diagram View Needme

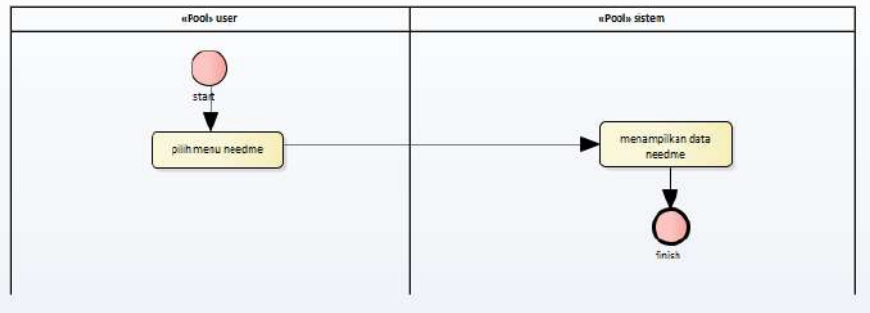


Figure 4.10: Actifty Diagram View Needme

10. Activiity Diagram View Tentang

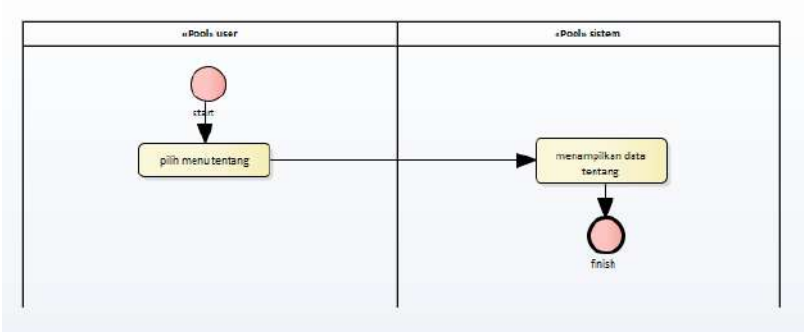


Figure 4.11: Activiity Diagram View Tentang

11. Actifity Diagram View Profil

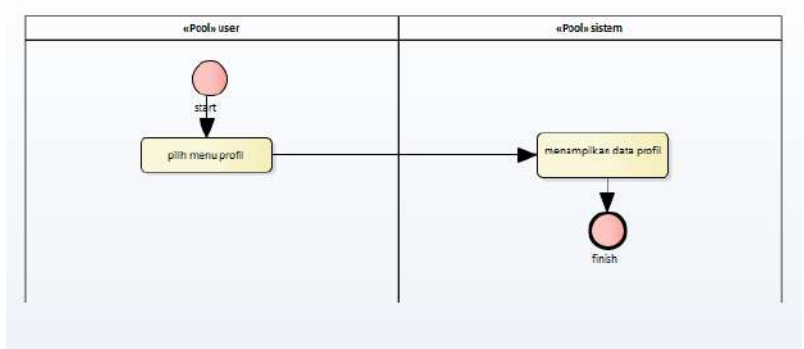


Figure 4.12: Actifity Diagram View Profil

4.1.3 Sequence Diagram

Sequence diagram adalah suatu diagram yang memperlihatkan/menampilkan interaksi-interaksi antar objek di dalam sistem yang disusun pada sebuah urutan atau rangkaian waktu. Interaksi antar objek tersebut termasuk pengguna, display, dan sebagainya berupa "pesan/message". [9]

1. Sequence Diagram Login

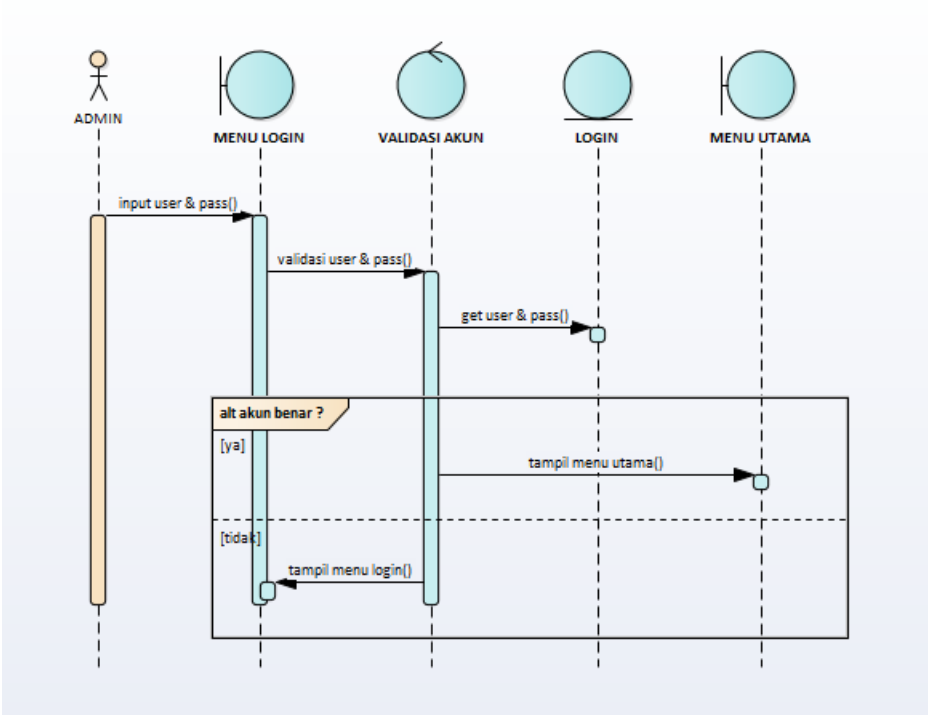


Figure 4.13: Sequence Diagram Login

2. Sequence Diagram Kelola Info

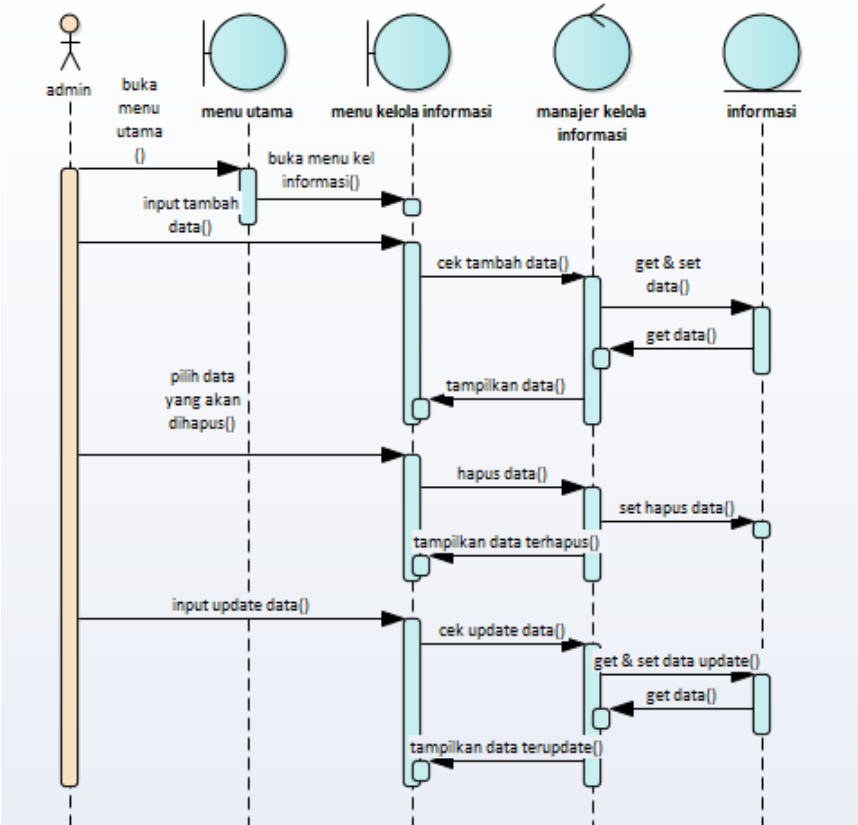


Figure 4.14: Sequence Diagram Kelola Info

3. Sequence Diagram Kelola Event

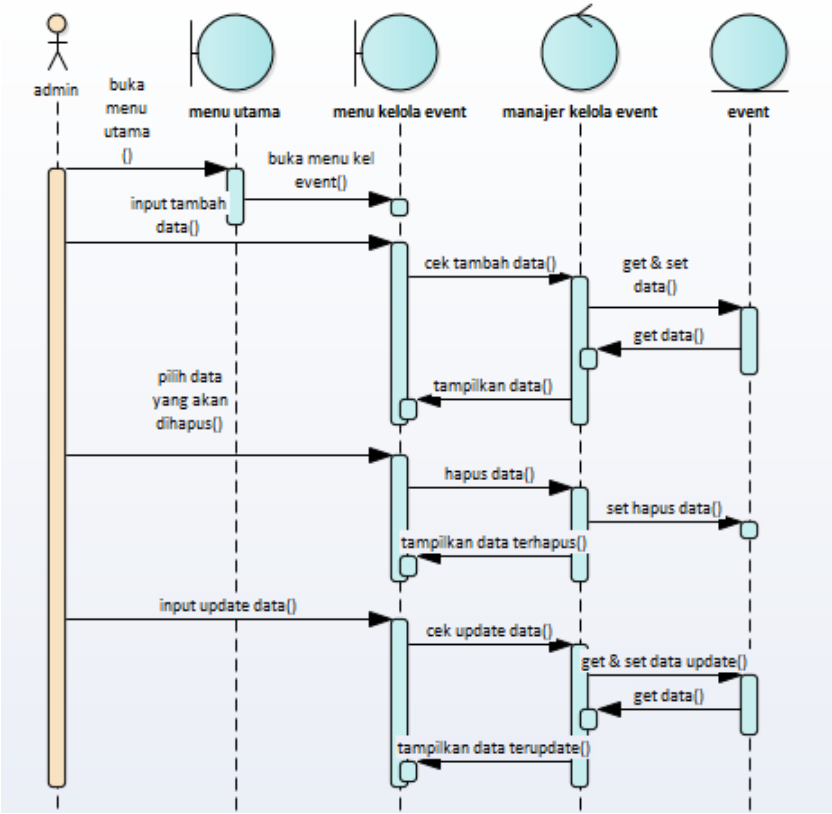


Figure 4.15: Sequence Diagram Kelola Event

4. Sequence Diagram Kelola Needme

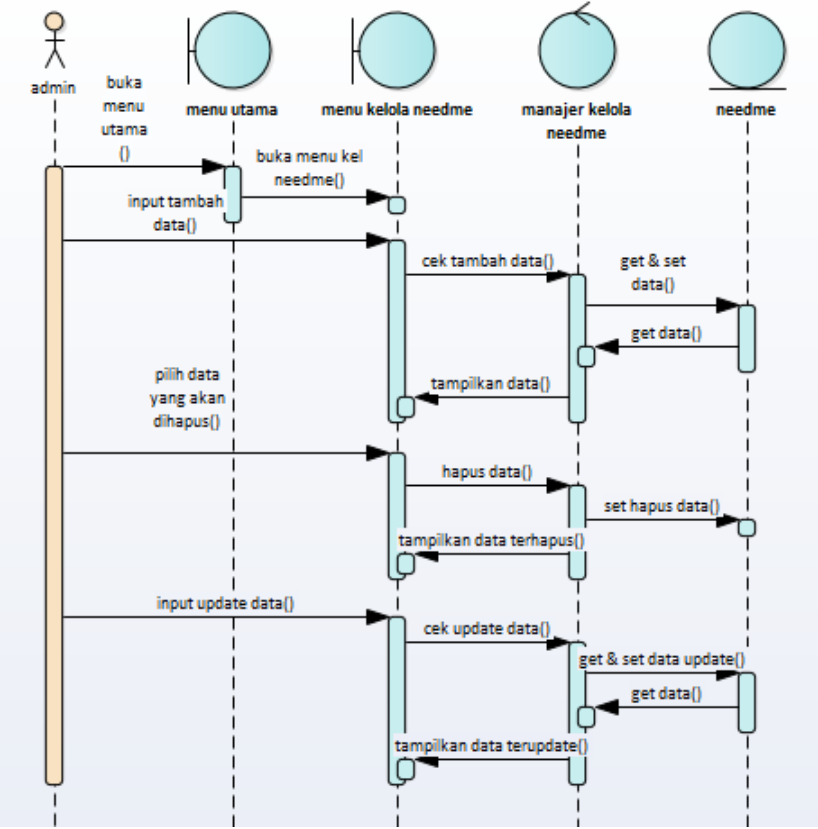


Figure 4.16: Sequence Diagram Kelola Needme

5. Sequence Diagram Kelola Tentang

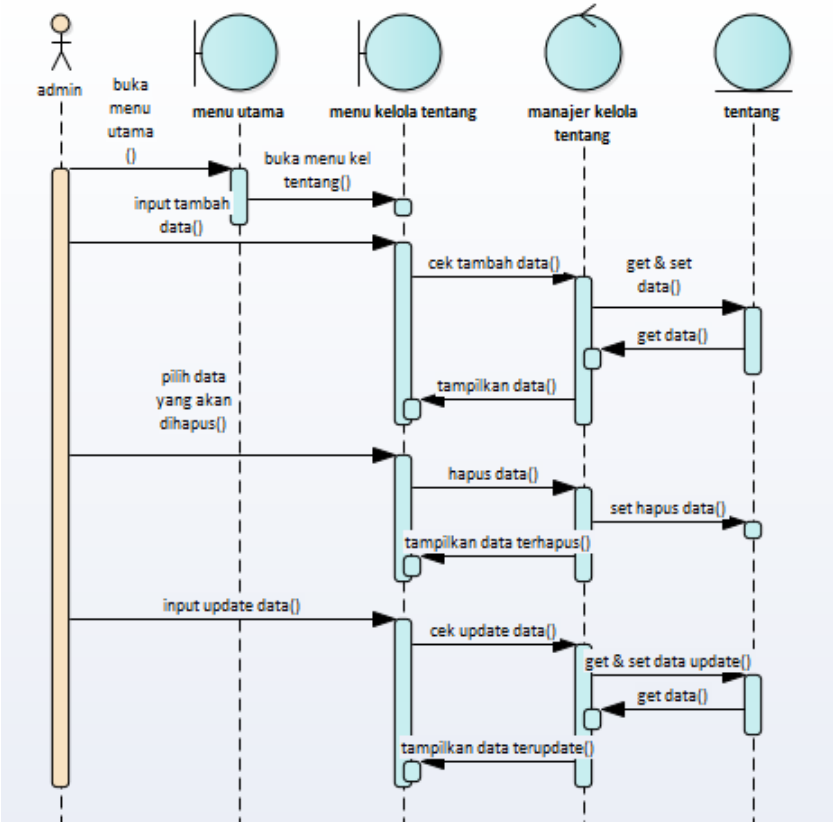


Figure 4.17: Sequence Diagram Kelola Tentang

6. Sequence Diagram Kelola Profil

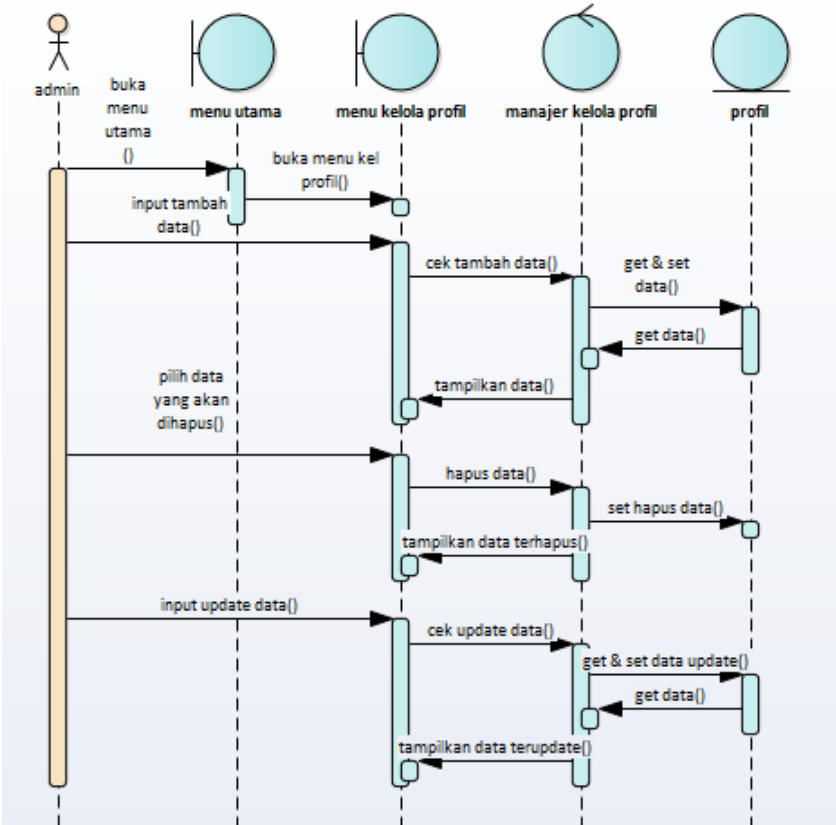


Figure 4.18: Sequence Diagram Kelola Profil

7. Sequence Diagram View Info

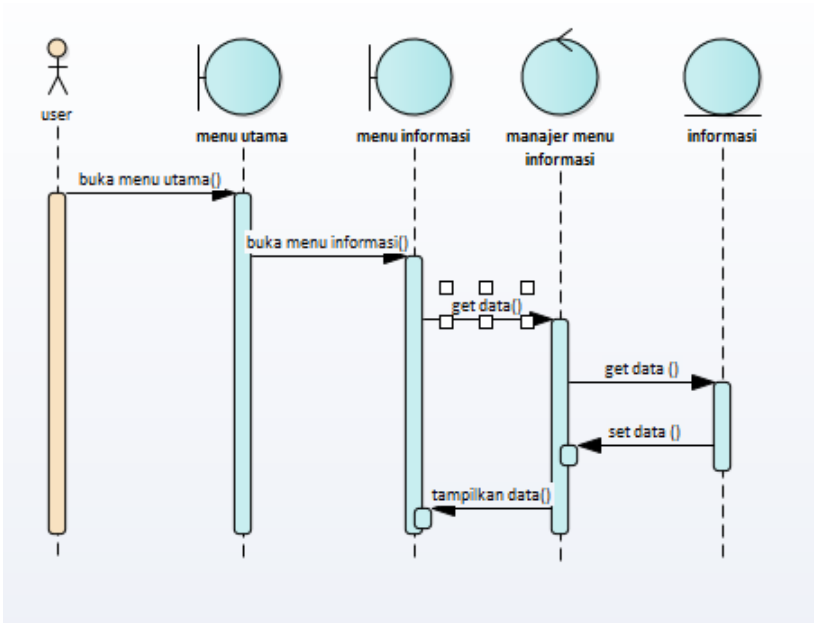


Figure 4.19: Sequence Diagram View Info

8. Sequence Diagram View Event

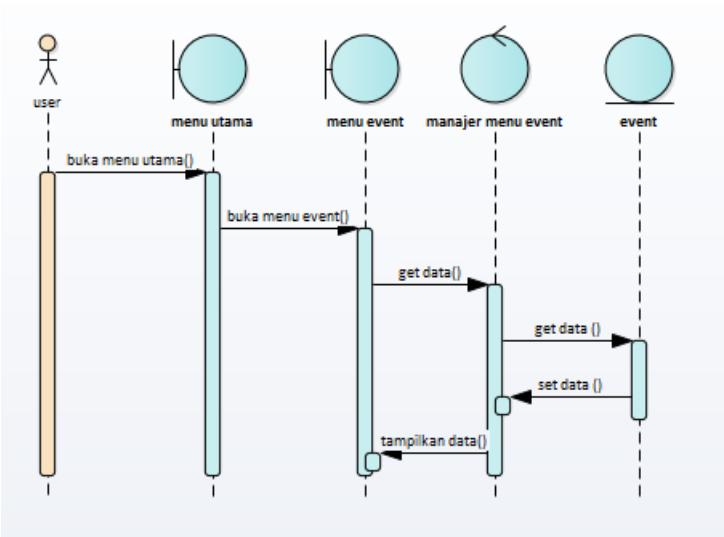


Figure 4.20: Sequence Diagram View Event

9. Sequence Diagram View Needme

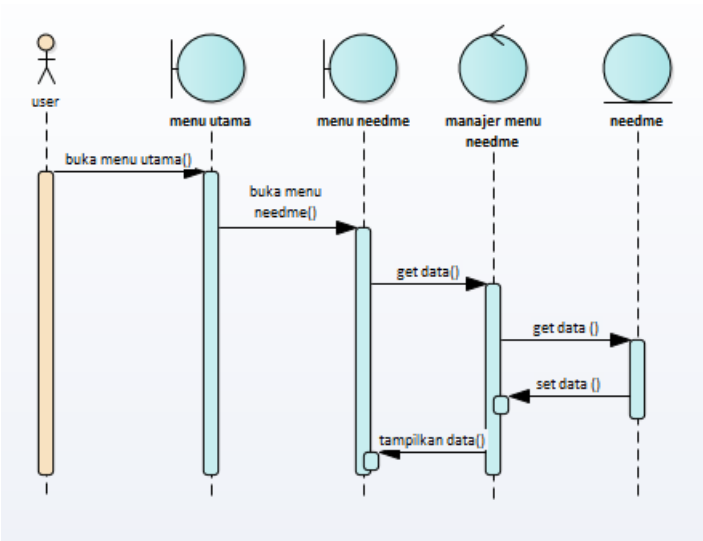


Figure 4.21: Sequence Diagram View Needme

10. Sequence Diagram View Tentang

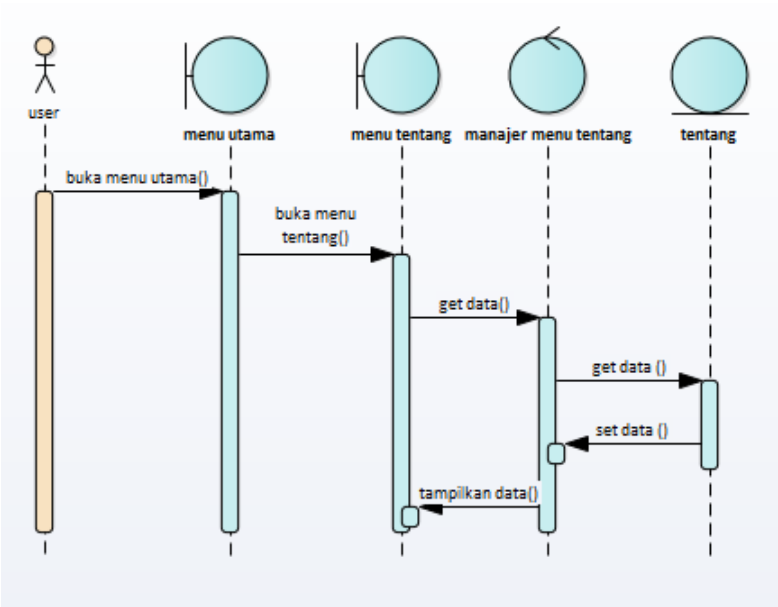


Figure 4.22: Sequence Diagram View Tentang

11. Sequence Diagram View Profil

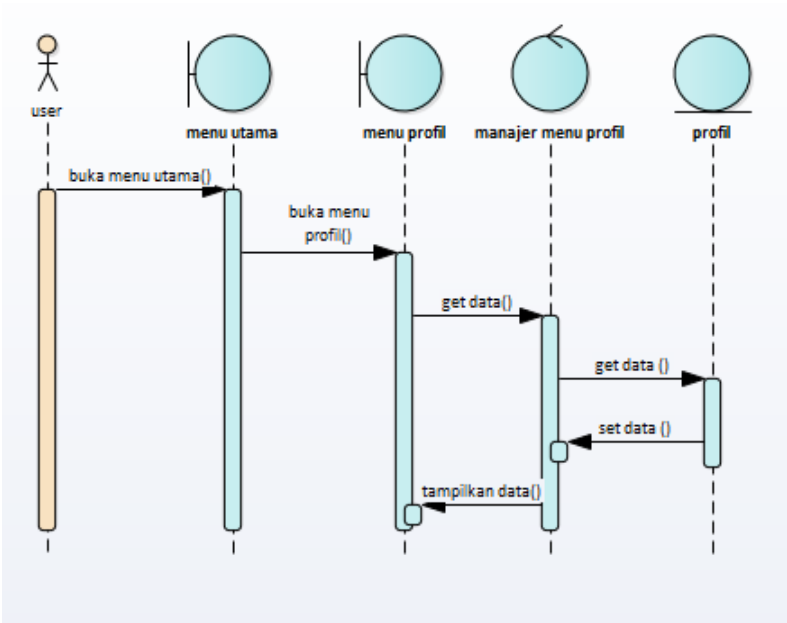


Figure 4.23: Sequence Diagram View Profil

4.1.4 Class Diagram

Class adalah kumpulan objek-objek yang mempunyai struktur umum, behavior umum, relasi umum, dan semantic atau kata yang umum (Henderi, 2007) [9]. Class diagram menggambarkan keadaan sistem fungsi-fungsi dan kebutuhan yang akan berkaitan dengan menu utama dan koneksi database.

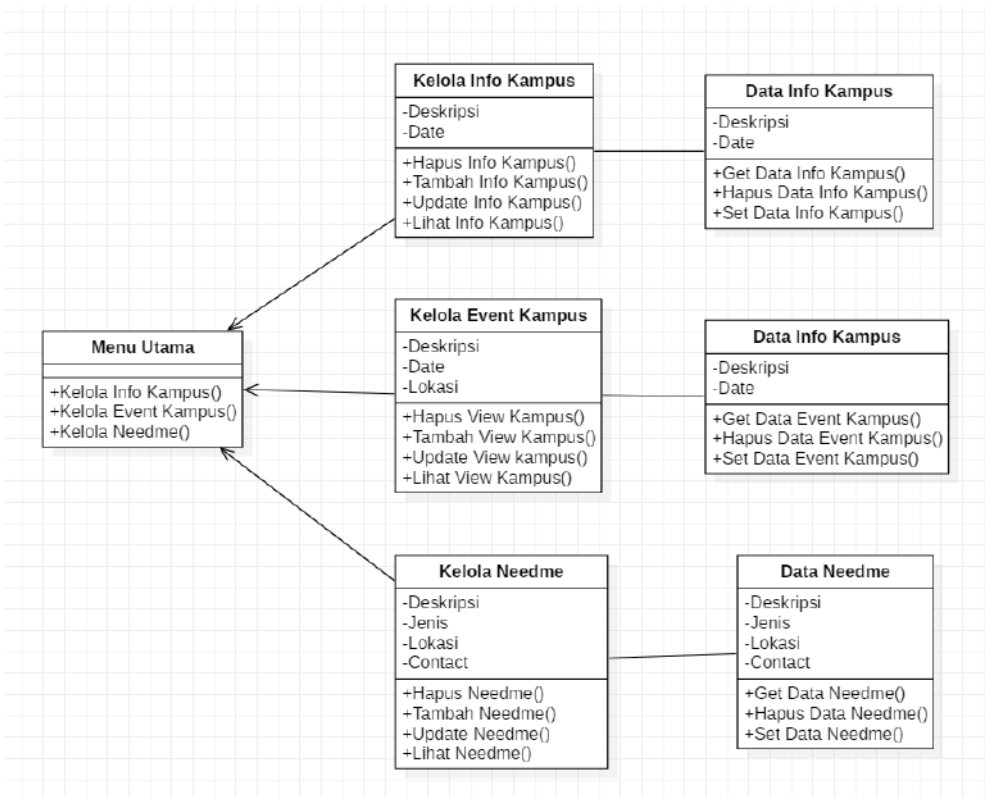


Figure 4.24: Class Diagram

4.1.5 Model "Air Terjun" (*Waterfall*)

Metode yang digunakan pada perancangan ini yaitu metode *Waterfall* atau Air Terjun. Sering juga disebut model *Sequential Linier*. Metode pengembangan sistem yang paling tua dan yang paling sederhana. Cocok untuk pengembangan perangkat lunak dan dengan mempunyai spesifikasi yang tidak akan berubah-ubah. Model ini menyediakan pendekatan alur hidup perangkat lunak secara *sequential* atau terurut dimulai dari analisa, desain, pengkodean, pengujian dan tahap pendukung.

4.1.6 Tahap-tahap metode *Waterfall*

1. Analisis Kebutuhan Perangkat Lunak

Pengumpulan kebutuhan untuk menspesifikasikan kebutuhan perangkat lunak sehingga dapat dipahami kebutuhan dari *user*.

2. Desain

Desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antar muka dan prosedur pengkodean.

3. Pembuatan Kode Program

Hasil tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak dari segi logik dan fungsional serta memastikan bahwa semua bagian sudah diuji sehingga keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung atau Pemeliharaan

Dikarenakan adanya perubahan ketika sudah dikirimkan ke *user*. Perubahan dapat terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian.

4.1.7 Keunggulan dan Kelemahan Metode *Waterfall*

Metode pengembangan *waterfall* mempunyai keunggulan dalam membangun dan mengembangkan suatu sistem, antara lain;

1. Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
2. Dokumen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.

Dalam proses membangun dan mengembangkan suatu sistem, metode waterfall mempunyai beberapa kelemahan, antara lain;

1. Diperlukan manajemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk..
2. Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan.
3. Pelanggan sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.

4.2 CARA PEMBUATAN APLIKASI

Sistem Informasi Badan Eksekutif Mahasiswa XYXY Berbasis *Website* ini memiliki *actor* dan 2 tampilan *interface*. Yaitu *Admin* sebagai pengelola *website* dan *User* sebagai mahasiswa umum yang mengakses *website* tersebut. Berikut ini penulis menyantumkan kode program atau kodingan dalam pembuatan Sistem Informasi Badan Eksekutif Mahasiswa XYXY Berbasis *Website*.

4.2.1 *Cascading Style Sheet (CSS)*

CSS adalah suatu cara untuk membuat format atau layout halaman web menjadi lebih menarik dan mudah dikelola. *CSS* muncul karena sulitnya mengatur layout tampilan dokumen yang dibuat dengan *HTML* murni meskipun telah menggunakan berbagai kombinasi format. Style ini digunakan untuk mengatur tata letak yang ada ditampilkan *user*. setiap file yang berhubungan dengan *CSS* akan mengakses *CSS* dengan cara menuliskan link pemanggilan fungsi *CSS* didalam file tersebut.

1. Style.css

```
1 .container {
2     width: 100%;
3     height: 2000px;
4 }
5 .head{
6     width: 100%;
7     height: 75px;
8     padding: 15px;
9 }
10 /**/
11 .menu{
12     width: 100%;
13     margin-bottom: 100px;
14 }
15 .menu a{
16     text-decoration: none;
17     color: #fff;
18 }
19 /**/
20 .menu li{
21     list-style-type: none;
22     float: left;
23     width: 110px;
```

```
24     height: 45px;
25     background: #333;
26     padding: 10px;
27     text-align: center;
28     cursor: pointer;
29     line-height: 30px;
30     transition-duration: 0.6s;
31 }
32 .menu li:hover {
33     background: #fb090a;
34 }
35 .menu li ul li {
36     position: relative;
37     right: 50px;
38     height: 65px;
39     top: 5px;
40 }
41 .menu li ul {
42     display: none;
43 }
44 .menu ul li span.arrow {
45     border: 4px solid transparent;
46     border-top: 9px solid #fff;
47     position: absolute;
48     top: 114px;
49     left: 230px;
50 }
51 .menu li:hover ul {
52     display: block;
53     animation: drop 0.3s;
54     position: relative;
55 }
56 .konten {
57     width: 100%;
58     height: 500px;
59     margin-top: 50px;
60 }
61 .left-konten {
62     width: 40%;
63     min-height: 400px;
64     background-color: #fff;
65     float: left;
66     margin: 10px;
67 }
68 .right-konten {
69     width: 55%;
70     min-height: 400px;
```

```
71 background-color: #fff;
72 float: left;
73 margin: 10px;
74 }
75 @keyframes drop {
76   0%{
77     opacity: 0;
78     left: 250%
79   }
80   50%{
81     opacity: 1;
82     left: 100px;
83   }
84   100%{
85     opacity: 1;
86     left: 5px;
87   }
88 }
89 .back {
90   margin-left: 75px;
91   margin-top: 35px;
92 }
93 .gambar {
94   margin-top: 20px;
95   margin-left: 35px;
96   float: left;
97 }
98 .pilihan{
99   width: 300px;
100  height: 200px;
101  float: left;
102  padding: 10px;
103  margin: 5px;
104  margin-bottom: 30px;
105 }
106 .tulisan{
107   width: 300px;
108   margin-bottom: 50px;
109   margin-left: 25px;
110 }
111 .garis{
112   width: 97%;
113   height: 10px;
114   margin: 25px;
115   position: center;
116 }
117 .medsos{
```

```
118 width: 100%;
119 height: 150px;
120 float: left;
121 margin-top: 50px;
122 }
123 .logobem{
124   margin-left: 75px;
125   margin-top: 10px;
126 }
127 .body-login{
128   margin: 0;
129   padding: 0;
130   font-family: sans-serif;
131   background: url(img/bem_gema.jpeg) no-repeat;
132   background-size: cover;
133 }
134 .login-box{
135   width: 280px;
136   position: absolute;
137   top: 50%;
138   left: 50%;
139   transform: translate(-50%, -50%);
140   color: white;
141 }
```

2. Style2.css

```
1 body{
2   margin: 0;
3   padding: 0;
4   font-family: sans-serif;
5   background: url(img/bem_gema.jpeg) no-repeat;
6   background-size: cover;
7 }
8 .login-box{
9   width: 280px;
10  position: absolute;
11  top: 50%;
12  left: 50%;
13  transform: translate(-50%, -50%);
14  color: white;
15 }
16 .login-box h1{
17   float: left;
18   font-size: 40px;
19   border-bottom: 6px solid #4caf50;
20   margin-bottom: 50px;
21   padding: 13px 0;
```

```
22 }
23 .textbox{
24     width: 100%;
25     overflow: hidden;
26     font-size: 20px;
27     padding: 8px 0;
28     margin: 8px 0;
29     border-bottom: 1px solid #4caf50;
30 }
31 .textbox input{
32     border: none;
33     outline: none;
34     background: none;
35     color: white;
36     font-size: 18px;
37     width: 80%;
38     float: left;
39     margin: 0 10px;
40 }
41 .btn{
42     width: 100%;
43     background: none;
44     border: 2px solid #4caf50;
45     color: white;
46     padding: 5px;
47     font-size: 18px;
48     cursor: pointer;
49     margin: 12px 0;
50 }
```

3. Style3.css

```
1 .container {
2     width: 100%;
3     height: 1000px;
4 }
5 .full{
6     width: 100%;
7     height: 2000px;
8 }
9 .head{
10     width: 100%;
11     height: 75px;
12     padding: 15px;
13 }
14 .left {
15     width: 30%;
16     min-height: 1000px;
```

```
17 background-color: #fff;
18 float: left;
19 margin-top: 10px;
20 }
21 .right{
22     width: 69%;
23     min-height: 1000px;
24     background-color: #fff;
25     float: left;
26     margin: 10px;
27     margin-right: 0px;
28 }
29 .gambar {
30     margin-top: 50px;
31     margin-left: 35px;
32 }
33 .garis{
34     width: 90%;
35     height: 8px;
36     margin: 25px;
37     position: center;
38 }
39 .menu{
40     width: 100%;
41     margin-bottom: 100px;
42 }
43 .menu a{
44     text-decoration: none;
45     color: #fff;
46 }
47 .menu li{
48     list-style-type: none;
49     width: 110px;
50     height: 45px;
51     padding: 10px;
52     text-align: center;
53     cursor: pointer;
54     line-height: 20px;
55     transition-duration: 0.6s; /*untuk animasi (delay warna)
        saat kursor mengarah ke menu*/
56     margin-left: 0px;
57 }
58 /*untuk membuat saat menu diarahi oleh kursor warnanya
        berubah*/
59 .menu li:hover {
60     background: #fb090a;
61 }
```

```
62 /*pengaturan menu turunan ke 1*/
63 .menu li ul li{
64     position: relative;
65     width: 150px;
66     left: 60px;
67     height: 50px;
68     top: -10px;
69 }
70
71 /*untuk menghilangkan menu turunan kedua dari tampilan*/
72 .menu li ul{
73     display: none;
74 }
75 /*untuk memunculkan turunan menu ke 1*/
76 .menu li:hover ul {
77     display: block;
78     animation:drop 0.3s;
79     position: relative;
80 }
81 /*untuk membuat menu turunan 1 bergerak dari kiri ke kanan ke
      kanan lagi*/
82 @keyframes drop {
83     0%{
84         opacity: 0;
85         left: 250%
86     }
87     50%{
88         opacity: 1;
89         left: 100px;
90     }
91     100%{
92         opacity: 1;
93         left: 5px;
94     }
95 }
```

4. Style4.css

```
1 body{
2     margin: 0;
3     padding: 0;
4     font-family: sans-serif;
5     background: url(img_kosan/bem_gema.jpeg) no-repeat;
6     background-size: cover;
7 }
8 .left-i{
9     width: 49%;
10    min-height: 450px;
```



```
11 background-color: #fff;
12 float: left;
13 margin-top: 10px;
14 margin-left: 10px;
15
16 }
17 .right-i{
18     width: 49%;
19     min-height: 450px;
20     background-color: #fff;
21     float: left;
22     margin: 10px;
23     margin-right: 0px;
24 }
25 .layar{
26     width: 100%;
27     height: 300px;
28     padding: 20px;
29     margin-top: 30px;
30 }
31 .layar-i{
32     width: 100%;
33     height: 450px;
34     padding: 20px;
35     margin-top: 30px;
36 }
37 }
38 .layar-a{
39     width: 100%;
40     height: 600px;
41     padding: 20px;
42     margin-top: 30px;
43 }
44 .layar-1{
45     width: 24%;
46     min-height: 420px;
47     background-color: #fff;
48     float: left;
49     margin-top: 10px;
50     margin-left: 10px;
51     padding: 10px;
52 }
53 .layar-2{
54     width: 24%;
55     min-height: 420px;
56     background-color: #fff;
57     float: left;
```

```
58     margin-top: 10px;
59     margin-left: 10px;
60     padding: 10px;
61 }
62 .layar-3{
63     width: 24%;
64     min-height: 420px;
65     background-color: #fff;
66     float: left;
67     margin-top: 10px;
68     margin-left: 10px;
69     padding: 10px;
70 }
71 .layar-4{
72     width: 24%;
73     min-height: 420px;
74     background-color: #fff;
75     float: left;
76     margin-top: 10px;
77     margin-left: 10px;
78     padding: 10px;
79 }
80 .container-1{
81     width: 100%;
82     height: 2000px;
83 }
84 .garis-nama{
85     width: 70%;
86     height: 10px;
87 }
88 .garis-nama1{
89     width: 35%;
90     height: 10px;
91 }
```

4.2.2 Tampilan Admin

- Login / Form Login (form_login.php)

```

1 <?php
2     session_start();
3     if (isset($_SESSION['npm'])) {
4         header("location:index_adminn.php");
5     } else {
6 ?>
7
8 <!DOCTYPE html>
9 <html>
10 <head>
11     <meta charset="utf-8">
12     <title>Login</title>
13     <link rel="stylesheet" type="text/css" href="style2.css">
14     <link rel="stylesheet" type="text/css" href="w3.css">
15 </head>
16 <body>
17 <div>
18     <form action="" method="post" name="form_input">
19     <script language="javascript"></script>
20     <div class="login-box">
21         <h1>Login</h1>
22         <div class="textbox">
23             <input type="text" placeholder="Username" name="
24             username" value="">
25         </div>
26
27         <div class="textbox">
28             <input type="password" placeholder="Password" name="
29             password" value="">
30         </div>
31
32         <button class="btn" type="submit" name="masuk" value="
33         Sign In">Sign In</button>
34
35     <?php
36         include "config.php";
37         if (isset($_POST['masuk'])) {
38             $cek = mysqli_query($conn, "SELECT * FROM admin WHERE
39             username = '". $_POST['username'] . "' AND password = '".

```

```

40         if ($count > 0) {
41             session_start();
42             $_SESSION['npm'] = $npm_user;
43             header("location:index_adminn.php");
44         } else {
45             echo "Gagal Masuk";
46         }
47     }
48     ?>
49
50 </div>
51 </div>
52 </body>
53 </html>
54 <?php } ?>

```

5. Koneksi (config.php)

```

1 <?php
2
3 /*
4 $dbhost = "localhost";
5 $dbname = "admin";
6 $dbuser = "root";
7 $dbpassword = "";
8 */
9
10 $conn = mysqli_connect('localhost','root','','webbem');
11 if (!$conn) {
12     echo 'Koneksi Gagal';
13 }
14
15 ?>

```

6. Tampilan Halaman Awal Admin (index-adminn.php)

```

1 <?php
2     session_start();
3     if (!isset($_SESSION['npm'])) {
4         echo "Anda Belum Masuk! Silahkan <a href='form_login.php'
5         '> masuk dulu!</a>";
6     } else {
7         ?>
8     <!DOCTYPE html>
9     <html>
10    <head>
11        <meta charset="utf-8">
12        <title>Halaman Utama</title>

```

```

12 <link rel="stylesheet" type="text/css" href="style3.css">
13 <link rel="stylesheet" type="text/css" href="w3.css">
14 </head>
15 <body>
16 <div class="head w3-green">
17 <center>
18 <font face="rockwell" size="5" color="black">Badan
  Eksekutif Mahasiswa </font>
19 </center>
20
21 </div>
22
23 <div class="left w3-green">
24 <div class="menu">
25 <ul><font face="rockwell" size="2">
26
27 <li><a href="index_adminn.php">Home</a></li>
28 <li><a href="H_Event.html">Event</a>
29 <ul>
30 <li><a href="t_hiburan/info_hiburan.php">Hiburan</
  li></a>
31 <li><a href=" ">Lomba</li></a>
32 <ul>
33 <li><a href="t_lakademik/info_la.php">Akademik
  </a>
34 <li><a href="t_lolahraga/info_lo.php">Olahraga
  </a>
35 <li><a href="t_lseni/info_ls.php">Seni</a>
36 </li>
37 </ul>
38 </ul>
39 </li>
40 <li><a href=" ">NeedMe</a>
41 <ul>
42 <li><a href="t_kosan/menuuu.php">Info Kosan</a>
43 <li><a href="t_kantin/info_kantin.php">Info Kantin
  </a>
44 <li><a href="t_laundry/info_laundry.php">Info
  Laundry</a>
45 <li><a href="t_tailor/info_tailor.php">Info Tailor
  </a>
46 <li><a href="t_catering/info_catering.php">Info
  Catering</a>
47 </ul>
48 </li>
49 <li><a href="keluar.php">Keluar</a></li>
50

```

```

51     </font>
52 </ul>
53 </div>
54 </div>
55
56 <div class="right">
57     <center>
58         <div><font face="rockwell" size="10" color="orange">
SELAMAT DATANG</font></div>
59
60         <div><font face="rockwell" size="7" color="green">
Berikan yang terbaik , Informasi Valid No Hoax</font></div>
61         <div class="garis w3-red"></div>
62         <div class="gambar">
63             <center>
64                 </div>
65             </center>
66         </center>
67     </div>
68 </body>
69 </html>
70
71 <?php } ?>
72

```

7. Halaman Menu-Menu Pada Tampilan Admin

(a) Koneksi (fungsi.php)

```

1 <?php
2
3 $conn = mysqli_connect("localhost","root","","webbem");
4
5 function query($query){
6     global $conn;
7     $data = mysqli_query($conn,$query);
8     $rows=[];
9     while( $row = mysqli_fetch_assoc($data)){
10         $rows[]=$row;
11     }
12     return $rows;
13 }
14
15 function tambah($data){
16     global $conn;
17     $nama = $data["nama"];
18     $hari_tgl= $data["hari_tgl"];

```

```

19     $jam = $data["jam"];
20     $lokasi = $data["lokasi"];
21     $keterangan = $data["keterangan"];
22     $foto = $data["foto"];
23
24     $query = "INSERT INTO hiburan VALUES (','','$nama','',
25     $hari_tgl','$jam','$lokasi','$keterangan','$foto')";
26     mysqli_query($conn, $query);
27     return mysqli_affected_rows($conn);
28 }
29 ?>

```

(b) Halaman Menu

```

1 <?php
2 session_start();
3 if(!isset($_SESSION['npm'])){
4     echo "Anda Belum Masuk! Silahkan <a href='../
5     form_login.php'> masuk dulu!</a>";
6 } else {
7
8 ?>
9 <!DOCTYPE html>
10 <html>
11 <head>
12     <meta charset="utf-8">
13     <title>Halaman Info Kantin</title>
14     <link rel="stylesheet" type="text/css" href="../style3.
15     css">
16     <link rel="stylesheet" type="text/css" href="../w3.css"
17     >
18 </head>
19 <body>
20 <div class="full">
21     <div class="head w3-green">
22         <center>
23             <font face="rockwell" size="5" color="black">Badan
24             Eksekutif Mahasiswa
25             </font>
26         </center>
27
28     </div>
29     <center>
30
31         <div><font face="rockwell" size="10" color="orange"
32         >SELAMAT DATANG</font></div>
33         <div><font face="rockwell" size="7" color="green">

```

```

29     INFO HIBURAN</font></div>
30     <div class="garis w3-red"></div>
31     <button><a href="tambah.php">Tambah</a></button>
32 <table border="1">
33
34     <tr>
35         <th>id Hiburan</th>
36         <th>Foto</th>
37         <th>Nama</th>
38         <th>Date</th>
39         <th>Jam</th>
40         <th>Lokasi</th>
41         <th>Keterangan</th>
42         <th>Aksi</th>
43     </tr>
44
45     <?php
46     include 'fungsi.php';
47     $no = 1;
48     $data = mysqli_query($conn, "select * from hiburan");
49     while( $d = mysqli_fetch_assoc($data)){
50         ?>
51         <tr>
52             <td><?php echo $no++; ?></td>
53             <td></td>
54             <td><?php echo $d['nama'];?></td>
55             <td><?php echo $d['hari_tgl'];?></td>
56             <td><?php echo $d['jam'];?></td>
57             <td><?php echo $d['lokasi'];?></td>
58             <td><?php echo $d['keterangan'];?></td>
59             <td>
60                 <button><a href="edit.php?id_hiburan=<?php echo
$d['id_hiburan']; ?>">EDIT</a></button>
61                 <button><a href="hapus.php?id_hiburan=<?php
echo $d['id_hiburan']; ?>">HAPUS</a></button>
62             </td>
63         </tr>
64     <?php
65     }
66     ?>
67     <p><button><a href="../index_adminn.php">Home</a></
button>
68     <br></p>
69 </center>
70 </table>

```



```

71     </table>
72 </center>
73 </div>
74 </body>
75 </html>
76 <?php }
77 ?>
78

```

(c) Tambah Data (tambah.php)

```

1     <?php
2     require 'fungsi.php';
3     $conn = mysqli_connect("localhost","root","","webbem"
4 );
5     if (isset($_POST["OK"])){
6         if (tambah($_POST)>0){
7             echo "
8                 <script>
9                     alert('Sedang Menyiapkan Informasi ');
10                    document.location.href='info_hiburan.php
11                ';
12                </script>
13            ";
14        }else{
15            echo "
16                <script>
17                    alert('Pesanan Gagal !');
18                </script>
19            ";
20        }
21    }
22    $webbem = query("SELECT * FROM hiburan");
23    ?>
24    <center>
25    <h2> Tambah Info Hiburan </h2>
26    <div>
27        <form action="" method="post" name="form_input"
28        enctype="multipart/form-data">
29            <script language="javascript">
30            </script>
31            <table>
32            <title>Tambah Info Kantin</title>
33            <tr>
34                <td>Nama</td>
35                <td>: <input type="text" name="nama" id="nama"
36                placeholder="TIDAK BOLEH KOSONG !" size="20"

```

```

34         required=""></td>
35     </tr>
36     <tr>
37         <td>Date</td>
38         <td>: <input type="text" name="hari_tgl" id="
hari_tgl" placeholder="TIDAK BOLEH KOSONG !" size="20"
required="">
39     </td>
40 </tr>
41 <tr>
42     <td>Jam</td>
43     <td> : <input type="text" name="jam" id="jam"
placeholder="TIDAK BOLEH KOSONG !" size="20" required
=""></td>
44 </tr>
45 <tr>
46     <td>Lokasi</td>
47     <td> : <input type="text" name="lokasi" id="
lokasi"
48     placeholder="TIDAK BOLEH KOSONG !" size="20"
required=""></td>
49 </tr>
50 <tr>
51     <td>Keterangan</td>
52     <td>: <input type="text" name="keterangan" id
="keterangan" placeholder="TIDAK BOLEH KOSONG !" size
="20" required=""></td>
53 </tr>
54 <tr>
55     <td>Foto</td>
56     <td>: <input type="file" name="foto" id="foto
" ></td>
57 </tr>
58 <td>
59 <td>
60     <button type="submit" name="OK" >Simpan</
button>
61     <input type="reset" name="Cancel" value="
Cancel">
62     <button><a href="info_hiburan.php">
Kembali</a></button></td>
63 </tr>
64 </table>
65 </form>
66 </body>
67 </html>

```

(d) Update Data (update.php)

```

1 <?php
2 // koneksi database
3 include 'fungsi.php';
4
5 // menangkap data yang di kirim dari form
6 $id_hiburan = $_POST['id_hiburan'];
7 $nama = $_POST['nama'];
8 $hari_tgl = $_POST['hari_tgl'];
9 $jam = $_POST['jam'];
10 $lokasi = $_POST['lokasi'];
11 $keterangan = $_POST['keterangan'];
12 $foto = $_POST['foto'];
13 // update data ke database
14 mysqli_query($conn, "update hiburan set nama='$nama',
    hari_tgl='$hari_tgl', jam='$jam', lokasi='$lokasi',
    keterangan='$keterangan', foto='$foto' where id_hiburan
    ='$id_hiburan' ");
15
16 // mengalihkan halaman kembali ke index.php
17 header("location:info_hiburan.php");
18
19 ?>
20

```

(e) Edit Data (edit.php)

```

1 <html>
2 <head>
3 </head>
4 <body>
5 <center>
6 <h3>Edit</h3>
7
8 <?php
9 include 'fungsi.php';
10 error_reporting(E_ALL ^ (E_NOTICE | E_WARNING));
11 $id_hiburan = $_GET['id_hiburan'];
12 $data = mysqli_query($conn, "select * from hiburan where
    id_hiburan ='$id_hiburan' ");
13 while($d = mysqli_fetch_array($data)){
14     ?>
15     <form method="post" action="update.php" name="">
16         <table>
17             <tr>
18                 <td>Nama</td>
19                 <td> :

```

```

20         <input type="text" name="nama" value="<?php
        echo $d['nama']; ?>" required><br>
21         <input type="hidden" name="id_hiburan"
        value="<?php echo $d['id_hiburan']; ?>"></td>
22     </tr>
23     <tr>
24
25         <td>Date</td>
26         <td> : <input type="text" name="hari_tgl"
        value="<?php echo $d['hari_tgl']; ?>" required></td>
27     </tr>
28     <tr>
29         <td>Jam</td>
30         <td> : <input name="jam" value="<?php echo $d['
        jam']; ?>" required>
31     </td>
32 </tr>
33 <tr>
34         <td>Lokasi</td>
35         <td> : <input name="lokasi" value="<?php echo
        $d['lokasi']; ?>" required>
36     </td>
37 </tr>
38 <tr>
39         <td>Keterangan</td>
40         <td> : <input type="text" name="keterangan"
        value="<?php echo $d['keterangan']; ?>" required></td>
41     </tr>
42     <tr>
43         <td>Foto</td>
44         <td> :
45         <input type="file" name="foto" value="<?php
        echo $d['foto']; ?>">
46     </td>
47 </tr>
48
49     <tr>
50         <td></td>
51         <td><input type="submit" value="SIMPAN">
52         <button><a href="info_hiburan.php">KEMBALI</a
        ></td></button>
53     </tr>
54 </table>
55 </form>
56 <?php
57 }
58 ?>

```

```
59 </center>
60 </body>
61 </html>
62
```

(f) Hapus Data (hapus.php)

```
1 <?php
2 // koneksi database
3 include 'fungsi.php';
4
5 // menangkap data id yang di kirim dari url
6 $id_hiburan= $_GET['id_hiburan'];
7
8
9 // menghapus data dari database
10 mysqli_query($conn,"delete from hiburan where id_hiburan
    ='$id_hiburan'");
11
12 // mengalihkan halaman kembali ke index.php
13 header("location:info_hiburan.php");
14
15 ?>
16
```

(g) Keluar (keluar.php)

```
1 <?php
2 session_start();
3 session_destroy();
4 header("location:form_login.php");
5 ?>
6
```

4.2.3 Tampilan User

1. Tampilan Awal User (Halaman-1.html)

```
1 <!doctype html>
2 <html>
3 <head>
4 <title>Website BEM </title>
5 <link rel="stylesheet" type="text/css" href="style.css">
6 <link rel="stylesheet" type="text/css" href="w3.css">
7 </head>
8 <body>
9 <div class="container w3-white">
```

```

10
11 <div class="head w3-green">
12
13   <font face="rockwell" size="5" color="black"><center>Badan
      Eksekutif Mahasiswa </center></font>
14 </div>
15
16 <div class="menu w3-red">
17
18   <ul><font face="rockwell" size="2"></font>
19
20   <center>
21
22     <li><a href="Halaman_1.html">Home</a></li>
23     <li><a href=" ">Informasi<span class="arrow"></span></a>
24       <ul>
25         <li><a href="info_doang/m_dan_b.html">Minat dan Bakat
              (UKM)</a>
26         <li><a href="info_doang/komunitas.html">Komunitas</a>
27       </ul>
28     </li>
29     <li><a href="H_Event.html">Event</a>
30       <ul>
31         <li><a href="t_hiburan/view_hiburan.php">Hiburan</li>
32       </a>
33         <li><a href=" ">Lomba</li></a>
34         <ul>
35           <li><a href="t_lakademik/view_la.php">Akademik</a>
36         >
37           <li><a href="t_lolahraga/view_lo.php">Olahraga</a>
38         >
39           <li><a href="t_lseni/view_seni.php">Seni</a>
40         </li>
41       </ul>
42     </li>
43     <li><a href=" ">NeedMe</a>
44       <ul>
45         <li><a href="t_kosan/view_kosan.php">Info Kosan</a>
46         <li><a href="t_kantin/view_kantin.php">Info Kantin
47       </a>
48         <li><a href="t_laundry/view_laundry.php">Info
              Laundry</a>
49         <li><a href="t_tailor/view_tailor.php">Info Tailor
50       </a>
51         <li><a href="t_catering/view_catering.php">Info
              Catering</a>

```

```

48         </ul>
49     </li>
50     <li><a href="H_Tentang_Bem.html">Tentang Kami</a>
51         <ul>
52             <li><a href="info_doang/h_logo_bem.html">Logo BEM</a>
53             <li><a href="info_doang/h_visi_misi.html">Visi & Misi
54             </a>
55             <li><a href="info_doang/h_struktur_org_arga.html">
56             Struktur Organisasi</a>
57             </ul>
58         </li>
59     <li><a href="H_Profil.html">Profil</a>
60         <ul>
61             <li><a href="info_doang/h_pengurus_inti.html">
62             Pengurus Inti</a>
63             <li><a href="info_doang/h_internal.html">Internal
64             Kampus</a>
65             <li><a href="info_doang/h_kema.html">Kesejahteraan
66             Mahasiswa</a>
67             <li><a href="info_doang/h_eksternal.html">Eksternal
68             Kampus</a>
69             </ul>
70         </li>
71     </center>
72 </ul>
73 </div>
74 <div class="konten">
75     <center>
76     <div class="left-konten">
77         <div></div>
78         <div><font face="rockwell" size="5" color="green">"
79         Semangat Pergerakan Demi Kesejahteraan"</font></div>
80         <div class="logobem"></div>
82         <font face="rockwell" size="3" color="black">~~~BEM
83         POLITEKNIK POS INDONESIA~~~</font>
84     </div>
85     <div class="right-konten">

```

```

86     <div></div>
87 </div>
88
89 </div>
90
91 <div class="garis w3-red"></div>
92
93 <div class="left-konten">
94
95     <center>
96     <div class="tulisan">
97
98     </div>
99
100     <div class="gambar">
101         
102         <br><font face="rockwell" size="3" color="black">
Pakaian Adat Sulawesi</font><br>
103
104         
105
106         
107         <div class="tulisan"><font face="rockwell" size="3"
color="black">Pakaian Adat Minang ~ Bazar Komunitas</font
></div>
108     </div>
109 </center>
110 </div>
111
112 <div class="right-konten">
113     <center>
114
115     <div class="tulisan">
116
117     </div>
118     <div class="gambar">
119         
120
121         
122
123         <div class="tulisan"><font face="rockwell" size="3"
color="black">BEM dan KSR</font></div>
124
125         
126     </div>
127 </center>
128 </div>

```



```

129 <div class="medsos">
130   <center>
131     <a href="link facebook"></a>
132     <font face="rockwell" size="2" color="purple">BEM
      Politeknik Pos Indonesia</font>
133
134     <a href="link instagram"></a>
135     <font face="rockwell" size="2" color="purple">
      bempoltekpos</font>
136
137     <a href="link line"></a>
138     <font face="rockwell" size="2" color="purple">@wlu2012n</
      font>
139
140     <a href="link twitter "></a>
141     <font face="rockwell" size="2" color="purple">
      @bempoltekpos</font>
142
143     <a href="link youtube"></a>
144     <font face="rockwell" size="2" color="purple">BEM</font>
145
146   </center>
147 </div>
148 </body>
149 </html>
150

```

2. Halaman Menu Pada Tampilan User

(a) Halaman Info Minat dan Bakat (m-dan-b.html)

```

1      <!doctype html>
2 <html>
3 <head>
4   <title>Website BEM</title>
5   <link rel="stylesheet" type="text/css" href="../style.
      css">
6   <link rel="stylesheet" type="text/css" href="../w3.css"
      >
7   <link rel="stylesheet" type="text/css" href="../style4.
      css">
8 </head>
9 <body>

```

```

10 <div class="container-1 w3-white">
11
12 <div class="head w3-green">
13
14   <font face="rockwell" size="5" color="black"><center>
15     Badan Eksekutif Mahasiswa </center></font>
16 </div>
17 <div class="menu w3-red">
18
19   <ul><font face="rockwell" size="2"></font>
20
21   <center>
22
23     <li><a href="../Halaman_1.html">Home</a></li>
24     <li><a href="">Informasi<span class="arrow"></span></a>
25       <ul>
26         <li><a href="m_dan_b.html">Minat dan Bakat (UKM)
27           </a>
28         <li><a href="komunitas.html">Komunitas</a>
29         </li>
30         <li><a href="H_Event.html">Event</a>
31         <ul>
32           <li><a href="../t_hiburan/view_hiburan.php">
33             Hiburan</li></a>
34           <li><a href="">Lomba</li></a>
35           <ul>
36             <li><a href="../t_lakademik/view_la.php">
37               Akademik</a>
38             <li><a href="../t_lolahraga/view_lo.php">
39               Olahraga</a>
40             <li><a href="../t_lseni/view_seni.php">Seni</a>
41             </li>
42             </ul>
43           </li>
44           <li><a href="">NeedMe</a>
45           <ul>
46             <li><a href="../t_kosan/view_kosan.php">Info
47               Kosan</a>
48             <li><a href="../t_kantin/view_kantin.php">Info
49               Kantin</a>
50             <li><a href="../t_laundry/view_laundry.php">
51               Info Laundry</a>

```

```

47         <li><a href=" ../t_tailor/view_tailor.php">Info
Tailor</a>
48         <li><a href=" ../t_catering/view_catering.php">
Info Catering</a>
49     </ul>
50 </li>
51 <li><a href="H_Tentang_Bem.html">Tentang Kami</a>
52     <ul>
53         <li><a href="h_logo_bem.html">Logo BEM</a>
54         <li><a href="h_visi_misi.html">Visi & Misi</a>
55         <li><a href="h_struktur_org_arga.html">Struktur
Organisasi</a>
56     </ul>
57 </li>
58 <li><a href="H_Profil.html">Profil</a>
59     <ul>
60         <li><a href="h_pengurus_inti.html">Pengurus Inti
</a>
61         <li><a href="h_internal.html">Internal Kampus</a>
62         <li><a href="h_kema.html">Kesejahteraan Mahasiswa
</a>
63         <li><a href="h_eksternal.html">Eksternal Kampus</
a>
64     </ul>
65 </li>
66
67 </center>
68 </ul>
69 </div>
70 <center>
71 <center><div class="garis-namal w3-red"></div></center>
72 <div><center><font face="forte" size="7" color="black">
Minat Dan Bakat (Unit Kegiatan Mahasiswa – UKM)</font>
></center></div>
73 <center><div class="garis-nama w3-red"></div></center><br>
<br>
74
75 <div class="layar-1">
76     <center>
77     <br><br>
78     <font face="forte" size="3" color="black">
79     – K-RADIO – <br>
80     Sesuia dengan namanya UKM K-Radio rutin melakukan
siaran di kampus politeknik pos Indonesia. Lewat K-
Radio juga Ormawa atau UKM lain dapat menyampaikan
berita dengan cara pengajuan permintaan media partner.

```

```

81     </font>
82   </center>
83 </div>
84
85 <div class="layar-2">
86   <center>
87     <br>
88     <font face="forte" size="3" color="black">
89       - ELC - <br>
90       Nah kalo kalian suka banget belajar bahasa Inggris ,
91       UKM ini adalah organisasi yang tepat. Di UKM ini kita
92       bisa sharing , mulai dari grammer , tensis , speaking dan
93       juga mengikuti perlombaan antar kampus.
94     </font>
95   </center>
96 </div>
97 <div class="layar-3">
98   <center>
99     <br>
100     <font face="forte" size="3" color="black">
101       - HIPMI - <br>
102       UKM HIPMI ini lebih ke kegiatan berbisnis. UKM HIPMI
103       telah menciptakan banyak produk usaha dan memiliki
104       relasi yang luas terutama antar kampus di Bandung.
105     </font>
106   </center>
107 </div>
108 <div class="layar-4">
109   <center>
110     <br>
111     <font face="forte" size="3" color="black">
112       - IMA - <br>
113       Kalo HIPMI menciptakan produk , nah kalo IMA ini
114       adalah UKM yang mempelajari Marketing pemasarannya.
115       Sama halnya dengan HIPMI , UKM IMA juga mempunyai banyak
116       relasi diluar kampus.
117     </font>
118   </center>
119 </div>
120 <div class="layar-1">
121   <center>
122     <br><br>

```

```

116 <font face="forte" size="3" color="black">
117   - COMPOSER - <br>
118   UKM composer, adalah UKM kesenian dalam bidang musik.
      Sudah banyak band-band bagus yang dibentuk dalam
      composer. Composer juga rutin mengadakan acara Musik
      Senja(konser kecil)
119 </font>
120 </center>
121 </div>
122
123 <div class="layar-2">
124   <center>
125     <br><br>
126     <font face="forte" size="3" color="black">
127       - BASKET -<br>
128       Basket adalah UKM olahraga yang sudah mengikuti
      perlombaan di dalam kampus maupun luar kampus. UKM
      Basket memiliki jadwal rutin latihan pada hari Jumat
      dan hari Minggu
129     </font>
130     </center>
131 </div>
132
133 <div class="layar-3">
134   <center>
135     <br><br>
136     <font face="forte" size="3" color="black">
137       - TENNIS CLUB - <br>
138       Tennis Club adalah UKM tennis lapangan. UKM ini juga
      sudah banyak mengikuti perlombaan diluar kampus, dan
      tidak jarang pulang dengan membawa penghargaan.
139     </font>
140     </center>
141 </div>
142 <div class="layar-4">
143   <center>
144     <br><br>
145     <font face="forte" size="3" color="black">
146       - TAEKWONDO - <br>
147       Sudah terlihat jelas dari nama UKM ini. UKM beladiri
      jenis taekwondo dapat kamu ikuti untuk menjaga diri
      kamu dari tindak laku kriminal
148     </font>
149     </center>

```

```

150 </div>
151
152 <div class="medsos">
153   <center>
154     <a href="https://m.facebook.com/PoliteknikPosIndonesia/"></a>
156     <font face="rockwell" size="2" color="purple">BEM
157       Politeknik Pos Indonesia</font>
158
159     <a href="http://instagram.com/bempoltekpos?igshid=r0noadrmme"></a>
161     <font face="rockwell" size="2" color="purple">
162       bempoltekpos</font>
163
164     <a href="http://line.me/ti/p/%40wlu2012n"></a>
165     <font face="rockwell" size="2" color="purple">
166       @wlu2012n</font>
167
168     <a href="https://twitter.com/bempoltekpos"></a>
170     <font face="rockwell" size="2" color="purple">
171       @bempoltekpos</font>
172
173     <a href="https://www.youtube.com/channel/UC46gaN1c6uqOii-EMAuDX8Q"></a>
174     <font face="rockwell" size="2" color="purple">BEM
175       poltekpos</font>
176
177   </center>
178 </div>
179

```

(b) Halaman Info Tailor (view-tailor.php)

```

1   <!DOCTYPE html>
2   <html>
3   <head>
4     <meta charset="utf-8">
5     <title>Halaman Info Tailor</title>
6     <link rel="stylesheet" type="text/css" href="../style3.css">
7     <link rel="stylesheet" type="text/css" href="../w3.css">

```

```

8
9 </head>
10 <body>
11 <div class="full">
12   <div class="head w3-green">
13     <center>
14       <font face="rockwell" size="5" color="black">Badan
15       Eksekutif Mahasiswa </font>
16     </center>
17   </div>
18   <center>
19
20     <div><font face="rockwell" size="10" color="orange"
21     >SELAMAT DATANG</font></div>
22     <div><font face="rockwell" size="7" color="green">
23     INFO LOMBA TAILOR</font></div>
24     <div class="garis w3-red"></div>
25
26 <table border="1">
27   <tr>
28     <th>id Tailor</th>
29     <th>Foto</th>
30     <th>Nama</th>
31     <th>Jenis Jahitan</th>
32     <th>Open-Close</th>
33     <th>Alamat</th>
34     <th>Contact</th>
35   </tr>
36
37   <?php
38   include 'fungsi.php';
39   $no = 1;
40   $data = mysqli_query($conn,"select * from tailor");
41   while( $d = mysqli_fetch_assoc($data)){
42     ?>
43     <tr>
44       <td><?php echo $no++; ?></td>
45       <td></td>
47       <td><?php echo $d['nama'];?></td>
48       <td><?php echo $d['jenis_jahitan'];?></td>
49       <td><?php echo $d['open_close'];?></td>
50       <td><?php echo $d['alamat'];?></td>
51       <td><?php echo $d['contact'];?></td>

```

```
51     </tr>
52     <?php
53     }
54     ?>
55     <p><button><a href=" ../ Halaman_1 .html ">Home</a></
      button>
56     <br></p>
57 </center>
58 </table>
59 </table>
60 </center>
61 </div>
62 </body>
63 </html>
64
```

Dibawah ini ada link code program, kalian bisa cek dibawah ini ;

https://drive.google.com/open?id=1mCzJFUUI5_

[qReeWI793qLwKfIhXd3Erq](https://drive.google.com/open?id=1mCzJFUUI5_qReeWI793qLwKfIhXd3Erq)

CARA MENJALANKAN APLIKASI

5.1 TAMPILAN ADMIN

1. Menu Login Admin

Untuk mengakses halaman Admin, *actor* harus memiliki akun untuk dapat masuk ke halaman selanjutnya. Caranya dengan memasukkan *user* dan *password* yang telah dibuat.



Figure 5.1: Halaman Login Admin

2. Halaman Awal Admin

Pada halaman utama ini terdapat menu menu informasi yang dapat dikelola oleh admin.



Figure 5.2: Halaman Awal Admin

3. Halaman Kelola Info Hiburan



Figure 5.3: Halaman Kelola Info Hiburan

4. Halaman Kelola Info Lomba Akademik



Figure 5.4: Halaman Kelola Info Lomba Akademik

5. Halaman Kelola Info Lomba Olahraga



Figure 5.5: Halaman Kelola Info Lomba Olahraga

6. Halaman Kelola Info Lomba Seni



Figure 5.6: Halaman Kelola Info Lomba Seni

7. Halaman Kelola Info Kosan



Figure 5.7: Halaman Kelola Info Kosan

8. Halaman Kelola Info Kantin



Figure 5.8: Halaman Kelola Info Kantin

9. Halaman Kelola Info Laundry



Figure 5.9: Halaman Kelola Info Laundry

10. Halaman Kelola Info Tailor



Figure 5.10: Halaman Kelola Info Tailor

11. Halaman Kelola Info Catering



Figure 5.11: Halaman Kelola Info Catering

12. Halaman Awal User



Figure 5.12: Halaman Awal User

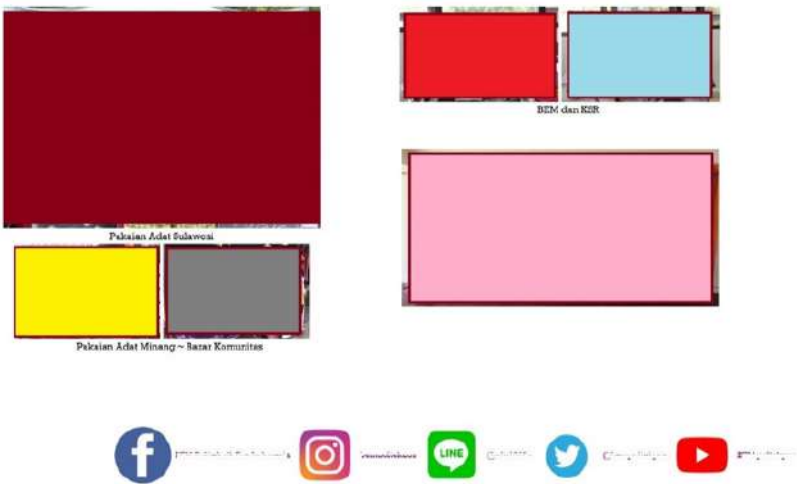


Figure 5.13: Halaman Awal User2

13. Halaman Info Minat Dan Bakat

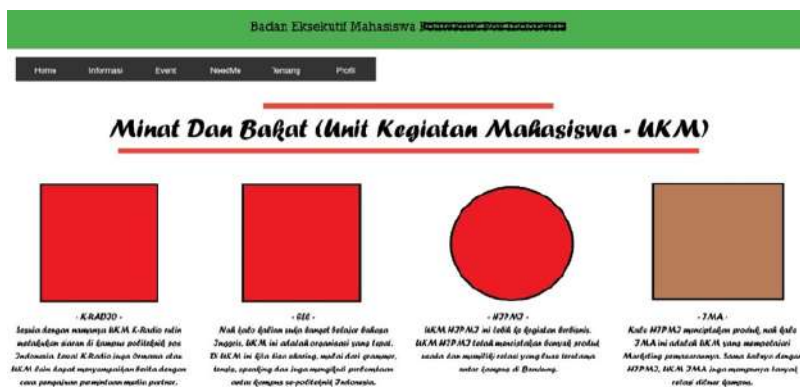


Figure 5.14: Halaman Info Minat Dan Bakat

14. Halaman Info Komunitas

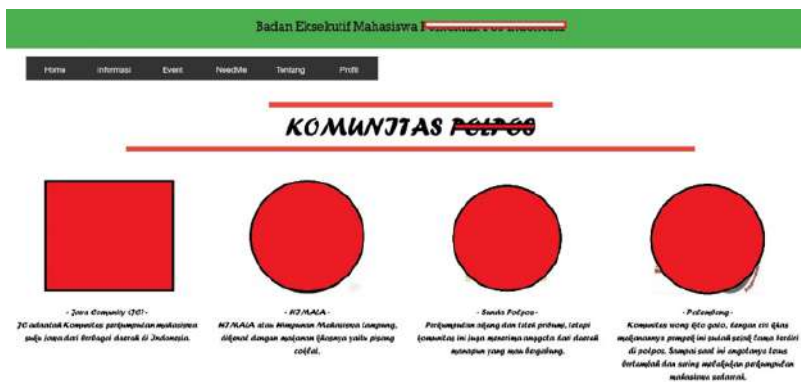


Figure 5.15: Halaman Info Komunitas

15. Halaman Info Hiburan



Figure 5.16: Halaman Info Hiburan

16. Halaman Info Lomba Akademik



Figure 5.17: Halaman Info Lomba Akademik

17. Halaman Info Lomba Olahraga



Figure 5.18: Halaman Info Lomba Olahraga

18. Halaman Info Lomba Seni



Figure 5.19: Halaman Info Lomba Seni

19. Halaman Info Kosan



Figure 5.20: Halaman Info Kosan

20. Halaman Info Kantin



Figure 5.21: Halaman Info Kantin

21. Halaman Info Laundry



Figure 5.22: Halaman Info Laundry

22. Halaman Info Tailor



Figure 5.23: Halaman Info Tailor

23. Halaman Info Catering



Figure 5.24: Halaman Info Catering

24. Halaman Info Logo BEM

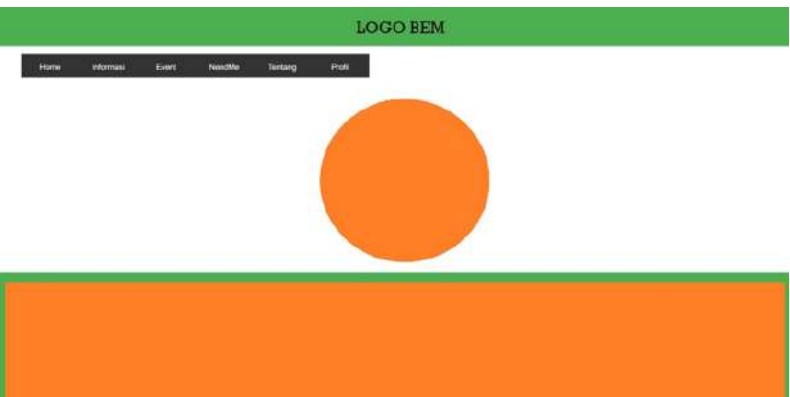


Figure 5.25: Halaman Info Logo BEM

25. Halaman Info Visi dan Misi



Figure 5.26: Halaman Info Visi dan Misi

26. Halaman Info Struktur Organisasi



Figure 5.27: Halaman Info Struktur Organisasi

27. Halaman Info Pengurus Inti

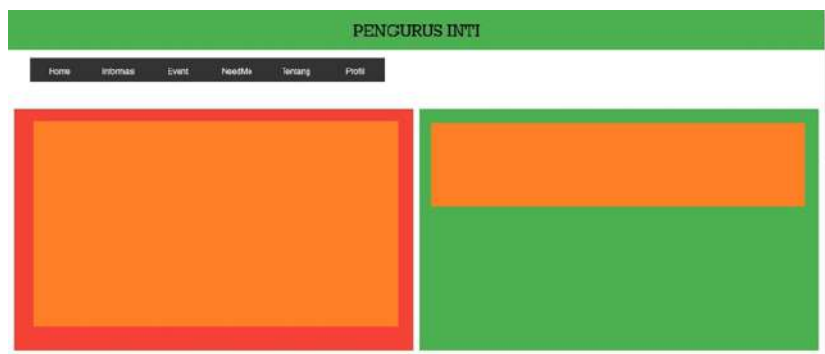


Figure 5.28: Halaman Info Pengurus Inti

28. Halaman Info Pengurus Internal Kampus

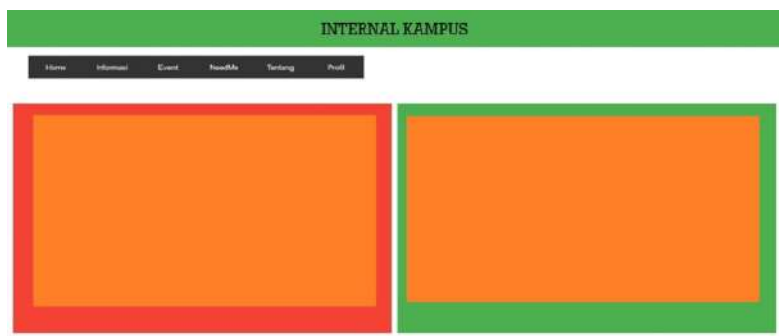


Figure 5.29: Halaman Info Pengurus Internal Kampus

29. Halaman Info Pengurus Kesejahteraan Mahasiswa



Figure 5.30: Halaman Info Pengurus Kesejahteraan Mahasiswa

30. Halaman Info Pengurus Eksternal Kampus

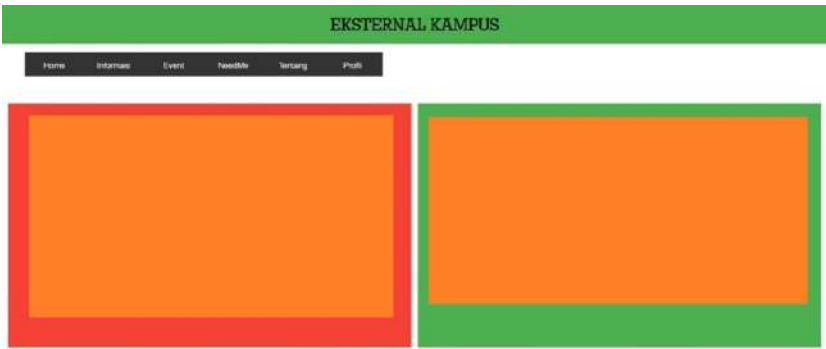


Figure 5.31: Halaman Info Pengurus Eksternal Kampus

DAFTAR PUSTAKA

- [1] Mara Destiningrum and Qadhli Jafar Adrian. Sistem informasi penjadwalan dokter berbasis web dengan menggunakan framework codeigniter (studi kasus: Rumah sakit yukum medical centre). *Jurnal Teknoinfo*, 11(2):30–37, 2017.
- [2] Harri Hidayat, Hartono Hartono, and Sukiman Sukiman. Pengembangan learning management system (lms) untuk bahasa pemrograman php. *Jurnal Ilmiah Core IT: Community Research Information Technology*, 5(1), 2017.
- [3] Tri A Kurniawan. Pemodelan use case (uml): Evaluasi terhadap beberapa kesalahan dalam praktik. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, 5(1):77–86, 2018.
- [4] Higia Agustina Mawadah, Ivan Yuliansyah, Alifah Novita Rohimanto, Tri Brotoharsono, and Indra Azimi. Kencolepot aplikasi pencarian tempat wisata kuliner di bandung berbasis android. *eProceedings of Applied Science*, 1(2), 2015.
- [5] Gulda Patria and Nanda Nugraha. Membangun cloud repository library berbasis web menggunakan owncloud, php, apache, css, mysql dan sistem operasi debian 9.0. *JUPITER*, 5(1), 2019.
- [6] Dana Pranata, Hamdani Hamdani, and Dyna Marisa Khairina. Rancang bangun website jurnal ilmiah bidang komputer (studi kasus: Program studi ilmu komputer universitas mulawarman). *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, 10(2):25–29, 2015.
- [7] Syaifudin Ramadhani, Urifatun Anis, and Siti Tazkiyatul Masruro. Ran-

- cang bangun sistem informasi geografis layanan kesehatan di kecamatan lamongan dengan php mysql. *Jurnal Teknik*, 5(2), 2013.
- [8] Anggiani Septima Riyadi, Eko Retnadi, and Asep Deddy Supriatna. Perancangan sistem informasi berbasis website subsistem guru di sekolah pesantren persatuan islam 99 rancabango. *Jurnal Algoritma*, 9(2):327–337, 2012.
- [9] Yosua Simaremare, Apol Pribadi, and Radityo Prasetyanto Wibowo. Perancangan dan pembuatan aplikasi manajemen publikasi ilmiah berbasis online pada jurnal sisfo. *Jurnal Teknik ITS*, 2(3):A470–A475, 2013.
- [10] Rahmawati Erma Standsyah and Intannia Sari Restu. Implementasi phpmyadmin pada rancangan sistem pengadministrasian. *Unisda Journal of Mathematics and Computer Science (UJMC)*, 3(2):38–44, 2017.