



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

LEMBAGA PENGEMBANGAN DAN PEMBERDAYAAN PENDIDIK DAN TENAGA KEPENDIDIKAN

BIDANG KELAUTAN PERIKANAN TEKNOLOGI INFORMASI DAN KOMUNIKASI

(LPPPTK KPTK) GOWA

2018

RPL

REKAYASA
PERANGKAT LUNAK

Produce Server-side Script for Dynamic Web Pages

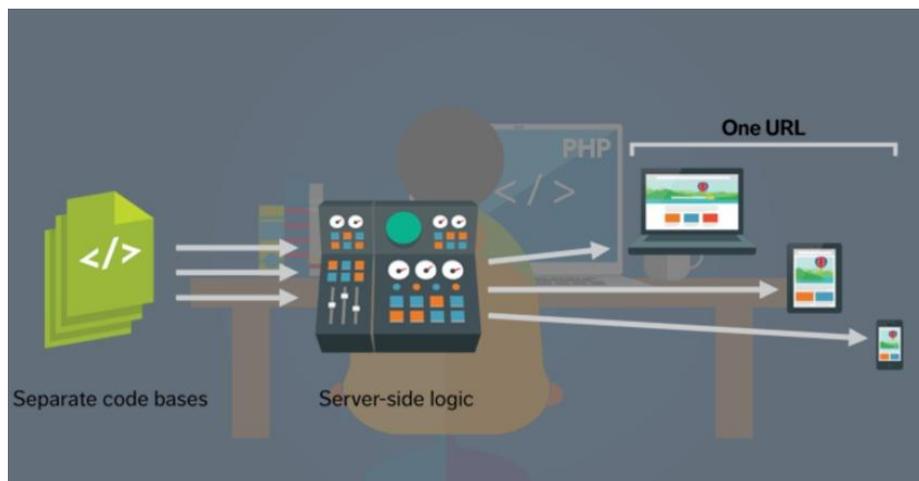
Penulis:

Deny Guslendra, S.Kom.

**Modul,
Pelatihan
Berbasis
Kompetensi**



MODUL
PENGEMBANGAN KEPROFESIAN
BERKELANJUTAN BERBASIS KOMPETENSI
MEMBUAT WEB DINAMIS MENGGUNAKAN
SCRIPT SERVER SIDE
ICTWEB415



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN R.I.
DIREKTORAT JENDERAL GURU DAN TENAGA KEPENDIDIKAN
LEMBAGA PENGEMBANGAN DAN PEMBERDAYAAN PENDIDIK DAN TENAGA KEPENDIDIKAN BIDANG
KELAUTAN, PERIKANAN, DAN TEKNOLOGI INFORMASI DAN KOMUNIKASI
GOWA
2018

MODUL PENGEMBANGAN KEPROFESIAN BERKELANJUTAN BERBASIS KOMPETENSI

BIDANG KEAHLIAN REKAYASA PERANGKAT LUNAK (RPL)

PROFESIONAL:

Produce Server-side Script for Dynamic Web Pages

Penulis:

Deny Guslendra, S.Kom. (guslendra@gmail.com))

Penelaah:

Novi Nurlaela, S.Kom. (java.alex@gmail.com)

Yohan Pribadi (Yue4hn@gmail.com)

Desain Grafis dan Ilustrasi:

Andi Arman Arsyad, S.Kom. (armand.pisces@gmail.com)

Copyright © 2018

Direktorat Pembinaan Guru Pendidikan Dasar Direktorat Jenderal Guru dan Tenaga
Kependidikan Kementerian Pendidikan dan Kebudayaan

Hak Cipta Dilindungi Undang-Undang Dilarang mengcopy sebagian atau keseluruhan isi
buku ini untuk kepentingan komersial tanpa izin tertulis dari Kementerian Pendidikan
Kebudayaan.

KATA PENGANTAR

Modul pengembangan keprofesian berkelanjutan (PKB) berbasis kompetensi merupakan salah satu media pembelajaran yang dapat digunakan sebagai media transformasi pengetahuan, keterampilan dan sikap kerja kepada peserta pelatihan untuk mencapai kompetensi tertentu berdasarkan program pelatihan yang mengacu kepada Standar Kompetensi.

Modul pelatihan ini berorientasi kepada pelatihan berbasis kompetensi (*Competence Based Training*) diformulasikan menjadi 3 (tiga) buku, yaitu Buku Informasi, Buku Kerja dan Buku Penilaian sebagai satu kesatuan yang tidak terpisahkan dalam penggunaannya sebagai referensi dalam media pembelajaran bagi peserta pelatihan dan instruktur, agar pelaksanaan pelatihan dapat dilakukan secara efektif dan efisien. Untuk memenuhi kebutuhan pelatihan berbasis kompetensi tersebut, maka disusunlah modul pelatihan berbasis kompetensi dengan judul "**Membuat Web Dinamis Menggunakan Script Server Side**".

Kami menyadari bahwa modul yang kami susun ini masih jauh dari sempurna. Oleh karena itu, kami sangat mengharapkan saran dan masukan untuk perbaikan agar tujuan dari penyusunan modul ini menjadi lebih efektif.

Demikian kami sampaikan, semoga Tuhan YME memberikan tuntunan kepada kita dalam melakukan berbagai upaya perbaikan dalam menunjang proses pelaksanaan pembelajaran di lingkungan direktorat guru dan tenaga kependidikan.

Gowa, April 2018
Kepala LPPPTK KPTK,

Prof. Dr. Irwan, M.Pd

DAFTAR ISI

KATA PENGANTAR.....	3
DAFTAR ISI.....	4
ACUAN STANDAR KOMPETENSI KERJA DAN SILABUS DIKLAT	5
A. Acuan Standar Kompetensi Kerja	5
B. Kemampuan yang Harus Dimiliki Sebelumnya	8
C. Silabus Diklat.....	9
LAMPIRAN	14
1. BUKU INFORMASI	14
2. BUKU KERJA	14
3. BUKU PENILAIAN.....	14

ACUAN STANDAR KOMPETENSI KERJA DAN SILABUS DIKLAT

A. Acuan Standar Kompetensi Kerja

Materi modul pelatihan ini mengacu pada unit kompetensi terkait yang disalin dari Standar Kompetensi Kerja dengan uraian sebagai berikut:

- Kode Unit** : **ICTWEB415**
- Judul Unit** : **Membuat Web Dinamis Menggunakan Script Server Side**
- Deskripsi Unit** : Unit kompetensi ini berhubungan dengan sikap, pengetahuan, dan keterampilan yang dibutuhkan dalam menggunakan *script server side* dalam membuat *web dinamis*.

ELEMEN KOMPETENSI		KRITERIA UNJUK KERJA	
1.	Menganalisa kebutuhan kebutuhan dokumentasi <i>web dinamis</i> sesuai interaksi <i>server side</i> .	1.1	Menentukan fungsionalitas dinamis dokumen web yang diperlukan.
		1.2	Menentukan Bahasa yang sesuai, untuk mencapai fungsionalitas tersebut.
		1.3	Menentukan persyaratan dokumen web.
2.	Merancang <i>Script Server Side</i>	2.1	Merancang dokumen web dan <i>skrip server side</i> , untuk diinteraksikan dengan sumber data eksternal.

		2.2	Dokumen web dan <i>script server side</i> dirancang, agar administrator dapat melakukan <i>insert</i> , <i>update</i> dan <i>delete</i> data pada sumber data external.
		2.3	Fitur keamanan didalam dokumen web diimplementasikan.
3	Membuat Dokumentasi Web	3.1	Menulis Bahasa <i>Extensible Hypertext Markup Language (XHTML)</i> yang dapat diperluas berdasarkan <i>aksesibilitas</i> .
		3.2	<i>Skrip Server Side</i> ditulis dengan standard XHTML.
4	Melakukan Uji dan <i>Debug Script</i>	4.1	Dokumen web diuji agar fungsionalitas tidak menjadi keharusan dan mengulanginya sampai benar.
		4.2	Dokumentasi dilengkapi dan dikirimkan ke personel yang tepat untuk mendapatkan

			persetujuan.
5	Melakukan Konfigurasi Keamanan Web	5.1	Menentukan ijin yang diperlukan untuk mencegah pesan kesalahan ditampilkan ke publik.
		5.2	Perangkat lunak <i>server</i> dikonfigurasi untuk meminimalisir potensi serangan terhadap <i>database</i> .

BATASAN VARIABEL

1. Konteks Variabel:

Unit ini berlaku untuk semua pengembang Bahasa pemrograman *server side* dalam menganalisa, merancang, membuat dan menguji web dokumen dinamis.

2. Peralatan dan perlengkapan

2.1 Peralatan

2.1.1 Perangkat lunak terkait

2.2 Perlengkapan

2.2.1 Petunjuk teknis konfigurasi

3. Peraturan yang diperlukan

3.1 Undang-Undang Nomor 11 Tahun 2008 tentang Informasi dan Transaksi Elektronik

3.2 Undang-Undang Nomor 14 Tahun 2008 tentang Keterbukaan Informasi Publik

4. Norma dan standar

4.1 Norma

4.1.1 Legalitas dan etika yang terkait dengan profesi bidang teknologi informasi

4.2 Standar

4.2.1 SNI ISO/IEC 20000-1:2009 Teknologi informasi Manajemen layanan Bagian 1: Spesifikasi

4.2.2 SNI ISO/IEC 20000-2:2009 Teknologi informasi Manajemen layanan Bagian 2: Aturan Praktik

4.2.3 Standar Pemrograman berorientasi objek yang ada

PANDUAN PENILAIAN

1. Konteks penilaian

1.1 Penilaian kompetensi dapat dilakukan dengan cara demonstrasi/praktik, dan/atau di tempat kerja.

2. Persyaratan kompetensi

2.1 ICTWEB415: Membuat Web Dinamis Menggunakan *Script Server Side*

3. Pengetahuan dan keterampilan yang dibutuhkan

3.1 Pengetahuan

3.1.1 Penggunaan perangkat lunak web dan cara konfigurasinya

3.2 Keterampilan

3.2.1 Mengoperasikan komputer

4. Sikap kerja yang diperlukan

4.1 Cekatan

4.2 Teliti

5. Aspek kritis

5.1 Ketepatan penggunaan memilih dan menggunakan perangkat lunak dengan maksimal.

B. Kemampuan yang Harus Dimiliki Sebelumnya

Ada pun kemampuan yang harus dimiliki sebelumnya sebagai berikut:

- Tidak ada

C. Silabus Diklat

Judul Unit Kompetensi : Membuat web dinamis menggunakan *script server side*.

Kode Unit Kompetensi : ICTWEB415

Deskripsi Unit Kompetensi : Unit ini meliputi pengetahuan, keterampilan, dan sikap kerja yang diperlukan untuk bekerja dalam pembuatan web dinamis dengan menggunakan *script server side*.

Perkiraan Waktu Pelatihan : JP @ 360 Menit

Tabel Silabus Unit Kompetensi :

Elemen Kompetensi	Kriteria Unjuk Kerja	Indikator Unjuk Kerja	Materi Diklat			Perkiraan Waktu Diklat (JP)	
			Pengetahuan (P)	Keterampilan (K)	Sikap (S)	P	K
Menganalisa kebutuhan kebutuhan dokumentasi <i>web dinamis</i> sesuai <i>interaksi server side</i> .	Menentukan fungsionalitas dokumen <i>web dinamis</i> yang diperlukan.	<ul style="list-style-type: none"> Mengetahui dan pemasangan kebutuhan perangkat keras dan lunak untuk pembuatan <i>web dinamis</i>. Dapat menjelaskan berbagai macam fungsi dalam penggunaan <i>web dinamis</i>. 	<ul style="list-style-type: none"> Prinsip kerja web server. Perangkat pendukung <i>web server</i>. 	<ul style="list-style-type: none"> Instalasi perangkat pendukung. 		20'	30'

	Menentukan Bahasa yang sesuai, untuk mencapai fungsionalitas tersebut.	<ul style="list-style-type: none"> Dapat memilih Bahasa pemrograman yang sesuai. Dapat memanfaatkan fungsi – fungsi dalam pemrograman <i>server side</i>. 	<ul style="list-style-type: none"> Sintak dasar dan fungsi PHP. <i>Variabel</i>, Tipe data dan penulisan operator dalam php. 	pengujian sintak dasar dengan memanfaatkan <i>variabel</i> , tipe data dan penulisan operator.		10'	10'
	Menentukan persyaratan dokumen web.	<ul style="list-style-type: none"> Dapat mengetahui syarat – syarat dalam dokumen web. 				15'	60'
Merancang <i>Script Server Side</i>	Dokumen web dan <i>skrip server side</i> dirancang, untuk diinteraksikan dengan sumber data <i>eksternal</i> .	<ul style="list-style-type: none"> Mampu merancang web dan sintak <i>server side</i> untuk dapat berinteraksi dengan <i>file external</i>. 	<ul style="list-style-type: none"> Jenis – jenis mode dan fungsi dalam interaksi dengan file. 	<ul style="list-style-type: none"> Mengatur file dengan memanfaatkan mode interaksi. 			
	Dokumen web dan <i>script server side</i> dirancang, agar administrator dapat melakukan <i>insert, update</i>	<ul style="list-style-type: none"> Mampu merancang web dan <i>sintak server side</i> agar dapat melakukan proses CRUT pada sumber basis data. 	<ul style="list-style-type: none"> <i>Review html</i> <i>Review MySQL query</i> Jenis dan fungsi metode pengiriman form. 	<ul style="list-style-type: none"> Membuat form dengan <i>html</i> dan php yang terkoneksi dengan basis data <i>mysql</i> 		30'	60'

	dan <i>delete</i> data pada sumber data <i>external</i> .						
	Fitur keamanan didalam dokumen web diimplementasikan.	<ul style="list-style-type: none"> Mampu mengimplementasikan fitur keamanan pada dokumen web. 	<ul style="list-style-type: none"> Fitur – fitur keamanan yang disediakan dalam sintak php. 	<ul style="list-style-type: none"> Membuat form pengiriman data dengan menerapkan keamanan data dalam sintak php. 		15'	15'
Membuat Dokumentasi Web	Menulis Bahasa <i>Extensible Hypertext Markup Language (XHTML)</i> yang dapat diperluas berdasarkan <i>aksesibilitas</i> .	<ul style="list-style-type: none"> Mampu mengembangkan Bahasa <i>xhtml</i> berdasarkan <i>aksesibilitas</i>. 	<ul style="list-style-type: none"> Tujuan dan manfaat dari XHTML. Aturan – aturan dalam penulisan <i>sintak xhtml</i>. 	<ul style="list-style-type: none"> mendesain dan menjalankan web dengan menggunakan standard <i>xhtml</i>. 		15'	15'
	<i>Skrip Server Side</i> ditulis dengan standard XHTML.	<ul style="list-style-type: none"> Mampu menerapkan <i>skrip server side</i> dengan kombinasi bahasa pemrograman <i>xhtml</i>. 		<ul style="list-style-type: none"> Mendesain dan menjalankan web dinamis PHP dengan menggunakan standard <i>xhtml</i>. 			25'
Melakukan Uji dan <i>Debug</i>	Dokumen web diuji agar	<ul style="list-style-type: none"> Mampu menguji fungsionalitas dokumen web untuk 	<ul style="list-style-type: none"> Metode – metode yang dipakai dalam melakukan 	<ul style="list-style-type: none"> Melakukan pengujian web dinamis dengan 		5'	5'

<i>Script</i>	<i>fungsi</i> alitas tidak menjadi keharusan dan mengulaginya sampai benar.	dapat bekerja sesuai dengan semestinya.	<i>debugging</i>	memanfaat <i>metode debugging.</i>			
	Dokumentasi dilengkapi dan dikirimkan ke personel yang tepat untuk mendapatkan persetujuan.	<ul style="list-style-type: none"> Dapat mendokumentasikan kebutuhan rekaman hasil debugging untuk dijadikan laporan. 		<ul style="list-style-type: none"> Memanfaatkan <i>fitur php</i> untuk merekam hasil <i>testing.</i> 			5'
Melakukan Konfigurasi Keamanan Web	Menentukan ijin yang diperlukan untuk mencegah pesan kesalahan ditampilkan ke publik.	<ul style="list-style-type: none"> Mampu menyembunyikan pesan kesalahan dari pengguna web. 	<ul style="list-style-type: none"> Fitur – fitur untuk mencegah pencurian data. 	<ul style="list-style-type: none"> Mengaktifkan fitur menyembunyikan dan merekam pesan kesalahan. 		5'	5'
	<i>Perangkat lunak server</i> dikonfigurasi untuk meminimalisir potensi serangan	<ul style="list-style-type: none"> Mampu mengkonfigurasi <i>web server</i> agar dapat menanggulangi serangan terhadap <i>database.</i> 	<ul style="list-style-type: none"> fitur konfigurasi keamanan dalam pada <i>web server.</i> 	<ul style="list-style-type: none"> Mengkonfigurasi 		5'	5'

	terhadap database.						
--	--------------------	--	--	--	--	--	--

LAMPIRAN

1. BUKU INFORMASI
2. BUKU KERJA
3. BUKU PENILAIAN



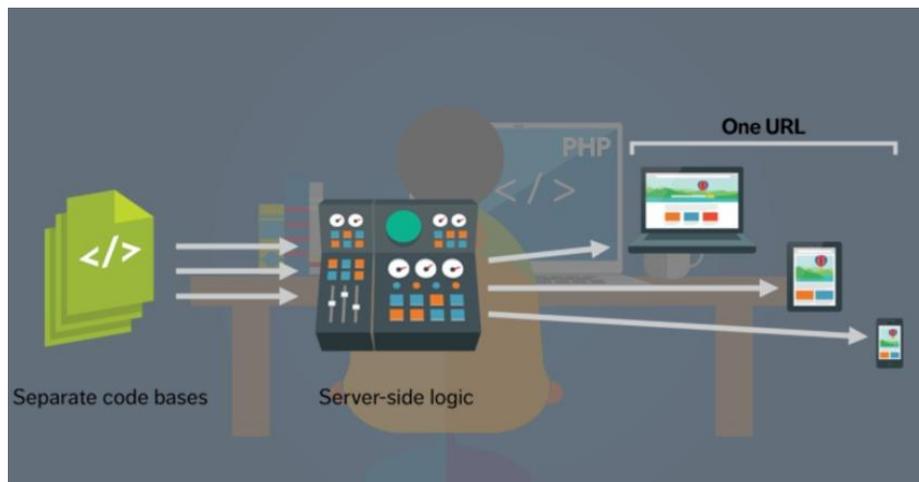
1

**LAMPIRAN
BUKU INFORMASI**



BUKU INFORMASI

MEMBUAT WEB DINAMIS MENGGUNAKAN SCRIPT SERVER SIDE ICTWEB415



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN R.I.
DIREKTORAT JENDERAL GURU DAN TENAGA KEPENDIDIKAN
LEMBAGA PENGEMBANGAN DAN PEMBERDAYAAN PENDIDIK DAN TENAGA
KEPENDIDIKAN BIDANG KELAUTAN, PERIKANAN, DAN TEKNOLOGI INFORMASI DAN
KOMUNIKASI
GOWA

DAFTAR ISI

DAFTAR ISI	2
DAFTAR GAMBAR.....	5
DAFTAR TABEL	6
BAB I PENDAHULUAN.....	7
A. Tujuan Umum	7
B. Tujuan Khusus	7
BAB II MENGANALISA KEBUTUHAN DOKUMENTASI WEB DINAMIS SESUAI INTERAKSI SERVER SIDE.....	8
A. Pengetahuan Yang Diperlukan Dalam Menganalisa Kebutuhan Dokumentasi Web Dinamis Sesuai Interaksi Server Side.	8
1. Web Dinamis.....	8
2. Web Server	8
3. Perangkat dan Aplikasi Pendukung Web Server	10
4. Instalasi XAMPP.....	12
5. Instalasi Text Editor Sublime Text	16
B. Keterampilan Yang Diperlukan Dalam Menganalisa Kebutuhan Dokumentasi Web Dinamis Sesuai Interaksi Server Side.	19
C. Sikap Kerja Yang Diperlukan Dalam Menganalisa Kebutuhan Dokumentasi Web Dinamis Sesuai Interaksi Server Side.	19
BAB III MERANCANG SCRIPT SERVER SIDE.....	20
A. Pengetahuan Yang Diperlukan Dalam Merancang Script Server Side	20
1. Pengenalan PHP	20
2. Sintak Dasar PHP.....	21
3. Komentar Dalam PHP	21
4. Variabel dan Aturan Penulisan.....	21
5. Tipe Data.....	22
6. Operator Matematika.....	23
7. Operator Penugasan	23
8. Operator Increment / Decrement	24
9. Operator Perbandingan	24
10. Operator Logika.....	24

Modul Pelatihan Berbasis Kompetensi Sektor Teknologi Informasi dan Komunikasi Sub Sektor Rekayasa Perangkat Lunak	Kode Modul ICTWEB415
11. Operator Array	25
12. File dan Direktori	25
13. Form HTML	33
14. Atribut action dan Atribut method	34
15. Elemen Kontrol di dalam Form.....	35
16. Latihan mengirimkan data menggunakan metode GET.....	37
17. Latihan mengirimkan data menggunakan metode POST.....	40
18. Kolaborasi dengan berbagai macam element.....	38
19. PHP dan Database MySQL.....	41
B. Keterampilan yang diperlukan dalam merancang script server side.....	51
C. Sikap kerja yang diperlukan dalam merancang script server side	51
BAB IV MEMBUAT DOKUMENTASI WEB.....	522
A. Pengetahuan Yang Diperlukan Dalam Membuat Dokumentasi Web	522
1. Pengenalan XHTML.....	52
2. XHTML dalam Sintak Server-side.....	544
B. Keterampilan Yang Diperlukan Dalam Membuat Dokumentasi Web	555
C. Sikap Kerja Yang Diperlukan Dalam Membuat Dokumentasi Web	555
BAB V MELAKUKAN UJI DAN DEBUG SCRIPT	566
A. Pengetahuan Yang Diperlukan Dalam Melakukan Uji Dan Debug Script.....	566
1. Pesan previous Error.....	566
2. Logging Error	57
3. Debugging dengan Assertion.....	58
4. Menangani error dokumen	600
B. Keterampilan Yang Diperlukan Dalam Melakukan Uji Dan Debug Script.....	611
C. Sikap Kerja Yang Diperlukan Dalam Melakukan Uji Dan Debug Script	611
BAB VI MELAKUKAN KONFIGURASI KEAMANAN WEB.....	622
A. Pengetahuan Yang Diperlukan Dalam Melakukan Konfigurasi Keamanan Web.	62
B. Keterampilan Yang Diperlukan Dalam Melakukan Konfigurasi Keamanan Web	69
Judul Modul: Membuat web dinamis menggunakan script server side Buku Informasi - Versi 2018	Halaman: 3 dari 72

C. Sikap Kerja Yang Diperlukan Dalam Melakukan Konfigurasi Keamanan Web..... 69

DAFTAR PUSTAKA..... 700

A. Buku Referensi 70

B. Referensi Lainnya 70

Daftar Alat Dan Bahan..... 71

Daftar Penyusun 72

DAFTAR GAMBAR

Gambar 2. 1 Alur kerja web server	8
Gambar 2. 2 Logo XAMPP	9
Gambar 2. 3 Logo Apache	10
Gambar 2. 4 Logo Microsoft.....	10
Gambar 2. 5 Web Browser.....	10
Gambar 2. 6 Logo Aplikasi Web Browser	11
Gambar 2. 7 Laman apachefriends.org	12
Gambar 2. 8 setup XAMPP	13
Gambar 2. 9 select component	13
Gambar 2. 10 installation folder	14
Gambar 2. 11 Bitnami for XAMPP	14
Gambar 2. 12 Tampilan install	15
Gambar 2. 13 Tampilan install selesai.....	15
Gambar 2. 14 XAMPP Control Panel.....	16
Gambar 2. 15 instalasi Sublime text	16
Gambar 2. 16 setup sublime text	17
Gambar 2. 17 destination location.....	17
Gambar 2. 18 ready to install.....	18
Gambar 2. 19 Completing Installation	19
Gambar 2. 20 Hasil instalasi.....	19

DAFTAR TABEL

Tabel 3. 1 Operator Matematika.....	24
Tabel 3. 2 Operator penugasan.....	23
Tabel 3. 3 Operator Increment / Decrement	24
Tabel 3. 4 Operator Perbandingan	24
Tabel 3. 5 Operator Logika	24
Tabel 3. 6 Operator Array.....	25
Tabel 3. 7 Mode dan Fungsi.....	25

BAB I PENDAHULUAN

A. Tujuan Umum

Setelah mempelajari modul ini peserta diharapkan mampu membuat web dinamis dengan menggunakan *skrip Server-Side* mulai dari tahap persiapan, rancangan, produksi, uji coba dan konfigurasi keamanan web.

B. Tujuan Khusus

Adapun tujuan mempelajari unit kompetensi melalui buku informasi Membuat web dinamis menggunakan *script server side* ini guna memfasilitasi peserta sehingga pada akhir diklat diharapkan memiliki kemampuan sebagai berikut:

1. Menganalisa kebutuhan kebutuhan dokumentasi web dinamis sesuai interaksi *server side*.
2. Merancang *Script Server Side*.
3. Membuat Dokumentasi Web.
4. Melakukan Uji dan *Debug Script*.
5. Melakukan Konfigurasi Keamanan Web.

BAB II

MENGANALISA KEBUTUHAN DOKUMENTASI WEB DINAMIS SESUAI INTERAKSI SERVER SIDE

A. Pengetahuan Yang Diperlukan Dalam Menganalisa Kebutuhan Dokumentasi Web Dinamis Sesuai Interaksi Server Side.

1. Web Dinamis

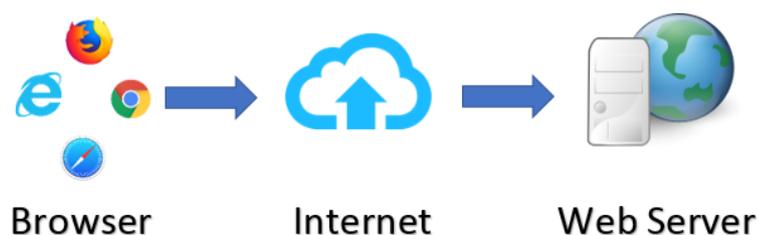
Web Dinamis adalah web yang konten / isi nya dapat berubah ubah setiap saat secara otomatis ataupun secara sistem yang telah di update pengguna (*user* atau *administrator*).

Perubahan konten / isi dokumen web dalam sebuah web dinamis tidak memerlukan keahlian programming dari sisi pengguna. Untuk membuat web dinamis diperlukan beberapa komponen yaitu *client-side scripting* (*HTML, JavaScript, Casing Style Sheet* atau *CSS*) dan *server-side scripting* seperti PHP dan program basis data seperti database MySQL untuk menyimpan data – data yang ada di web dinamis.

2. Web Server

Web Server merupakan *software* yang berfungsi untuk menerima permintaan HTTP dan HTTPS dari klien, yang dikenal dengan nama *web browser*, kemudian mengirimkan kembali hasilnya dalam bentuk halaman *web* yang secara umum berbentuk dokumen HTML.

Alur kerja *web server*



Gambar 2. 1 Alur kerja web server

Hubungan antara *web server* dan *browser* merupakan gabungan atau jaringan yang ada diseluruh dunia. Setelah terhubung secara fisik, *protocol* TCP/IP (*Networking Protocol*) yang memungkinkan semua komputer dapat berkomunikasi satu dengan yg lain. Pada saat *browser* meminta data *web page* ke *server* maka instruksi permintaan data oleh *browser* tersebut di kemas didalam TCP yang merupakan *protocol transport* dan dikirim ke alamat *protocol* tujuan yaitu *Hyper Text Transfer Protocol* (HTTP). **HTTP** merupakan *protocol* yang digunakan dalam *World Wide Web* (www) antar komputer yang terhubung dalam jaringan dunia. Protokol tersebut sudah sering digunakan ketika mengetik halaman *web* yang Anda tuju dengan mengetik (http://...).

Data yang di *passing* dari *browser* ke *web server* disebut sebagai **HTTP Request**, kemudian *web server* mencari data HTML yang ada dan dikemas dalam TCP *protocol* dan dikirim kembali ke *browser*. Pengiriman data dari *server* ke *browser* disebut sebagai **HTTP response**. Jika data yang diminta oleh *browser* tidak ditemukan oleh si *Web server* maka *browser* akan menampilkan pesan *error* seperti **Error : 404 Page Not Found**.

Fungsi utama dari *server web* adalah untuk mentransfer berkas atas permintaan pengguna melalui protokol komunikasi yang telah ditentukan. Sebuah halaman *web* dapat terdiri atas berkas *teks*, gambar, video, dan lainnya pemanfaatan *server web* berfungsi pula untuk mentransfer seluruh aspek pemberkasan dalam sebuah halaman *web* yang terkait, termasuk di dalamnya teks, gambar, video, atau lainnya.

Contoh web server:

- XAMPP (**X**, **A**pache, **M**ySQL, **P**HP, **P**erl) adalah perangkat lunak gratis yang berfungsi sebagai *server* yang berdiri sendiri (*localhost*) yang terdiri atas program *apache* HTTP *Server*, MySQL database, bahasa pemograman PHP dan Perl.



Gambar 2. 2 Logo XAMPP

- *Apache HTTP Server* adalah *web server* yang digunakan untuk melayani dan mengfungsikan situs *web*. Protokol yang digunakan untuk melayani fasilitas *web/ www* ini menggunakan *http*. *Web server* ini dapat dijalankan di banyak sistem operasi.



Gambar 2. 3 Logo Apache

- *IIS (Internet Information Services)* adalah *web server* yang digunakan dalam sistem operasi *Windows* yang berfungsi sebagai pendukung *protocol TCP/IP* yang berjalan pada lapisan aplikasi (*aplikasi layer*). *IIS* juga menjadi fondasi dari *platform Internet* dan *Intranet Microsoft*, yang mencakup *Microsoft Site Server*, *Microsoft Commercial Internet System* dan produk – produk *Microsoft BackOffice* lainnya.



Gambar 2. 4 Logo Microsoft

3. Perangkat dan Aplikasi Pendukung *Web Server*

a. Web Server

Web server merupakan sebuah komputer yang digunakan untuk yang melayani request halaman *web site*. Istilah *web server* biasanya dikenal sebagai *hosting* yaitu tempat data-data *website* diakses dan disimpan.



Gambar 2. 5 Web Browser

b. Web Browser

Web Browser adalah perangkat lunak yang digunakan untuk menerima dan menyajikan sumber informasi di internet.



Gambar 2. 6 Logo Aplikasi Web Browser

c. Jaringan Internet

Bandwith internet yang dimiliki harusnya besar dan yang lebih diperhatikan adalah *uploadnya* bukan *downloadnya* biasanya rasio *upload* dan *download* 1:1 semakin besar *bandwith* akan semakin banyak *web* yang bisa ditanamkan pada sebuah *server*. Jika dalam pengerjaannya menggunakan komputer lokal sebagai *web servernya*, maka jaringan *internet* tidak diperlukan.

d. Pemograman Server Side

Pemrograman *server side* merupakan teknologi *scripting* atau pemrograman web dimana *script* dikompilasi atau diterjemahkan di *server*. Contohnya seperti ASP, ColdFusion, Java Server Page, Perl, Python dan PHP. Pemrograman *web* yang paling umum digunakan adalah PHP.

e. Database

Database adalah kumpulan data yang tersimpan, tersusun dan saling terhubung satu dengan yang lain pada suatu komputer sehingga dapat digunakan untuk tujuan tertentu. Berikut ini adalah *database* yang sering digunakan antara lain: MySQL, Oracle, Microsoft Access, Microsoft SQL Server. Namun *database* yang paling banyak digunakan untuk pengembangan dokumen web dinamis adalah MySQL.

f. Text Editor

Perangkat lunak yang digunakan sebagai editor penulisan kode pemrograman seperti: HTML, PHP, CSS, JavaScript dll.

Contoh *software text editor*: *Adobe Dreamweaver, Notepad, Notepad++, Sublime Text, Brackets* dll.

g. FTP

Protocol internet yang digunakan untuk melakukan proses pengunduhan (*download*) dan pengunggahan (*upload*) berkas atau *file*. Proses *upload / download* berkas dilakukan jika telah memiliki *hosting*, berikut ini adalah penyedia *hosting* terkenal di Indonesia: *masterweb.com, jagoanhosting, niagahoster* dll.

4. Instalasi XAMPP

a. Unduh XAMPP

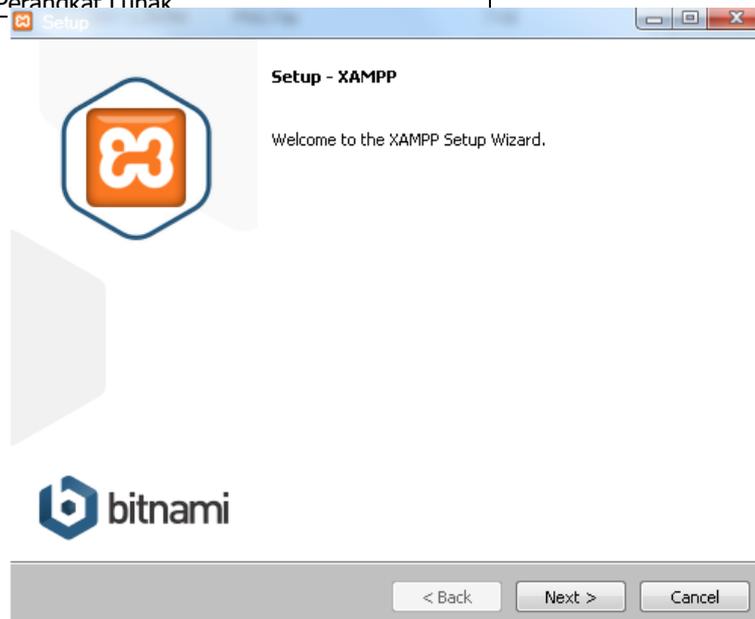
Download XAMPP melalui *website Apache Friends* pada link berikut ini <https://www.apachefriends.org/index.html>



Gambar 2. 7 Laman apachefriends.org

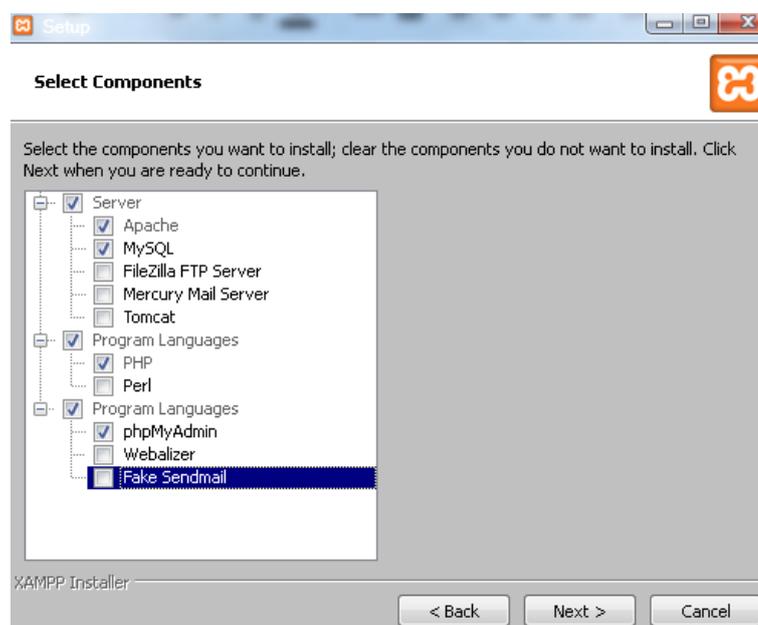
b. Install XAMPP

- Lakukan *instalasi* setelah Anda selesai mengunduh. Selama proses *instalasi* mungkin Anda akan melihat pesan yang menanyakan apakah Anda yakin akan menginstalnya. Silakan tekan *Yes* untuk melanjutkan *instalasi*.
- Klik tombol *NEXT*



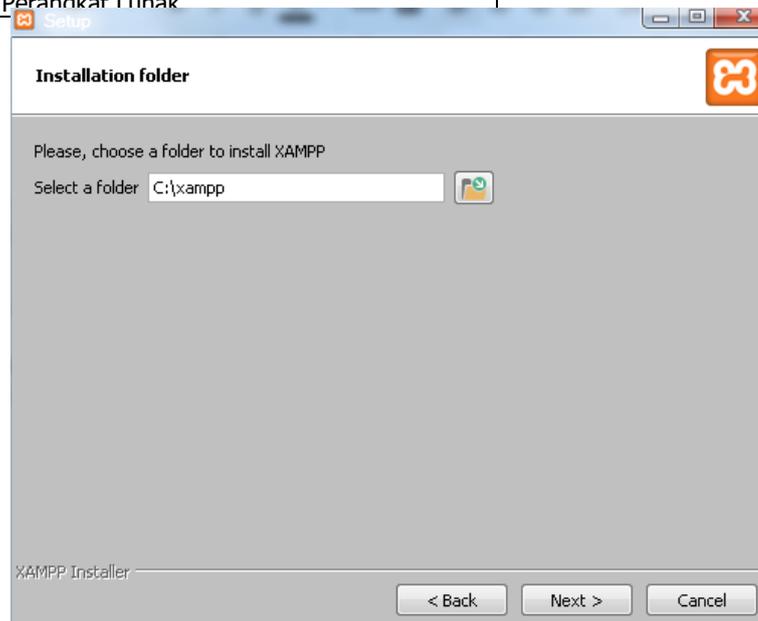
Gambar 2. 8 setup XAMPP

- Pada tampilan selanjutnya akan muncul pilihan mengenai komponen mana dari XAMPP yang ingin dan tidak ingin Anda *instal*. Beberapa pilihan seperti *Apache* dan *PHP* adalah bagian penting untuk menjalankan *website* dan akan otomatis diinstal. Silakan centang *MySQL* dan *phpMyAdmin*, untuk pilihan lainnya biarkan saja.



Gambar 2. 9 select component

- Berikutnya silakan pilih folder tujuan dimana XAMPP ingin Anda *instal*, pada tutorial ini pada direktori *C:|xampp*.



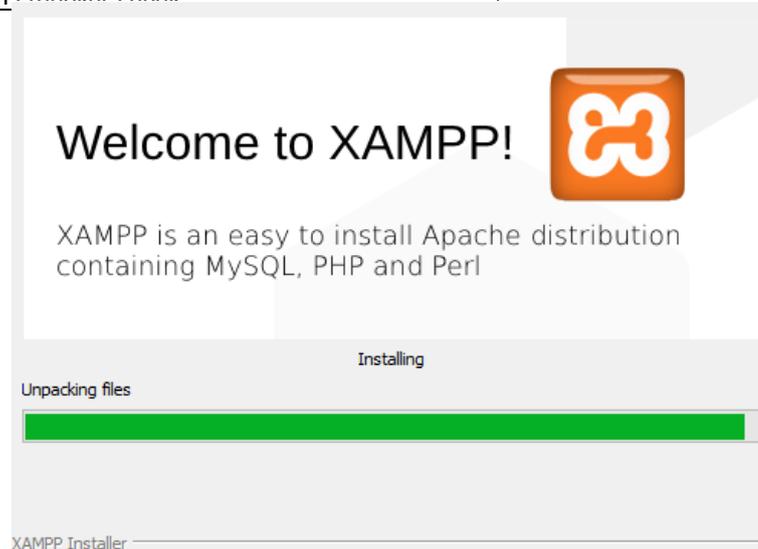
Gambar 2. 10 installation folder

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



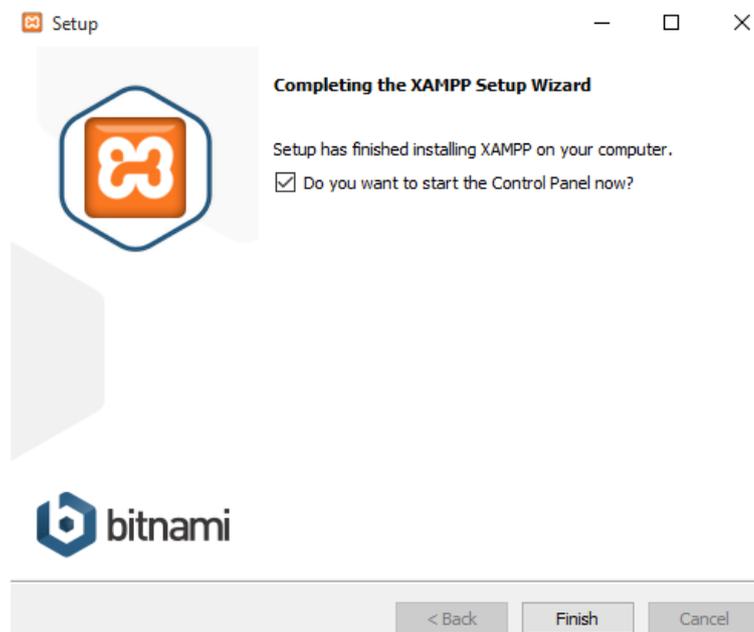
Gambar 2. 11 Bitnami for XAMPP

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



Gambar 2. 12 Tampilan install

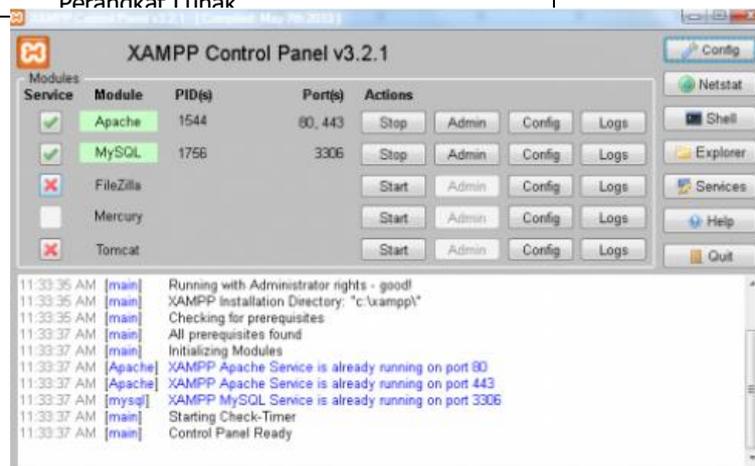
- Setelah berhasil diinstal, akan muncul notifikasi untuk langsung menjalankan *control panel*. Silakan klik *Finish*.



Gambar 2. 13 Tampilan install selesai

c. Jalankan XAMPP

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



Gambar 2. 14 XAMPP Control Panel

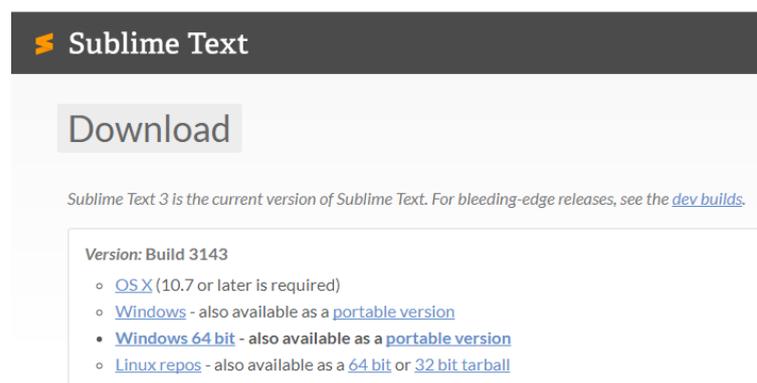
d. Pengecekan XAMPP

Untuk melakukan pengecekan, silakan akses *link* berikut melalui *browser* Anda <http://localhost>.

5. Instalasi *Text Editor Sublime Text*

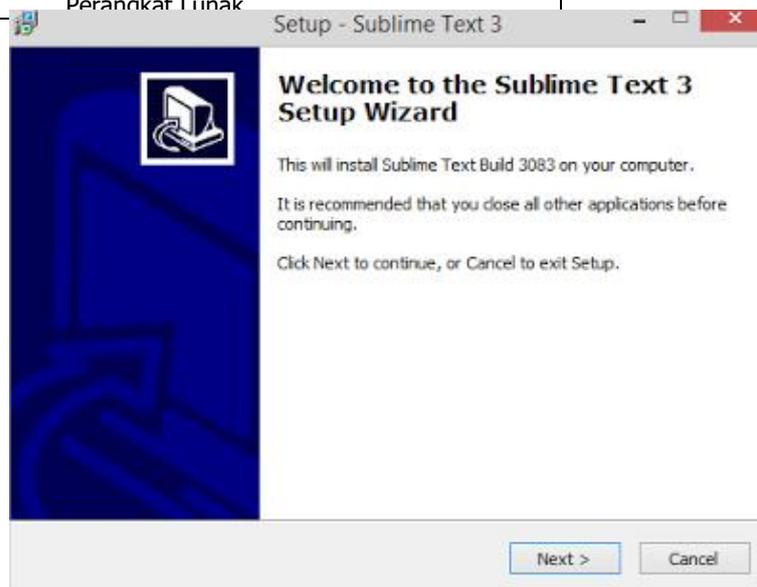
a. *Download aplikasi* melalui link berikut ini:

<https://www.sublimetext.com/3> . *Download* sesuai dengan sistem operasi yang dipakai.



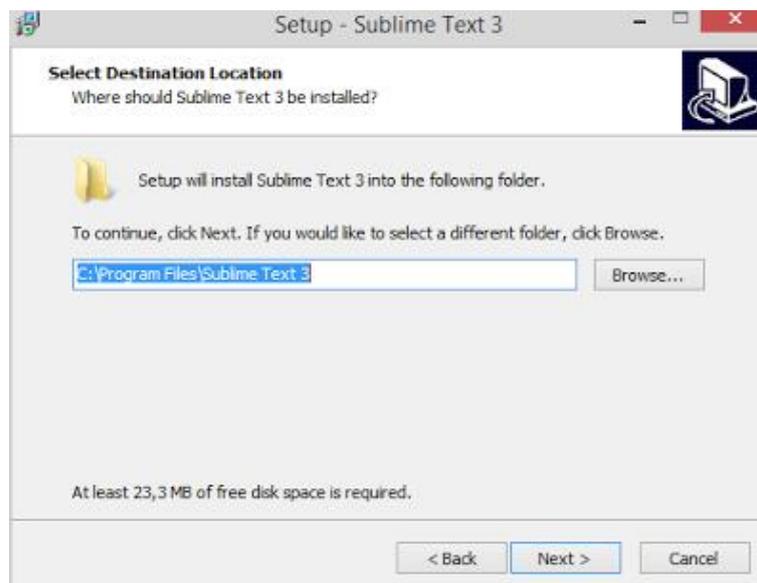
Gambar 2. 15 instalasi Sublime text

b. Klik 2x file hasil *download* tersebut dan klik tombol *next*



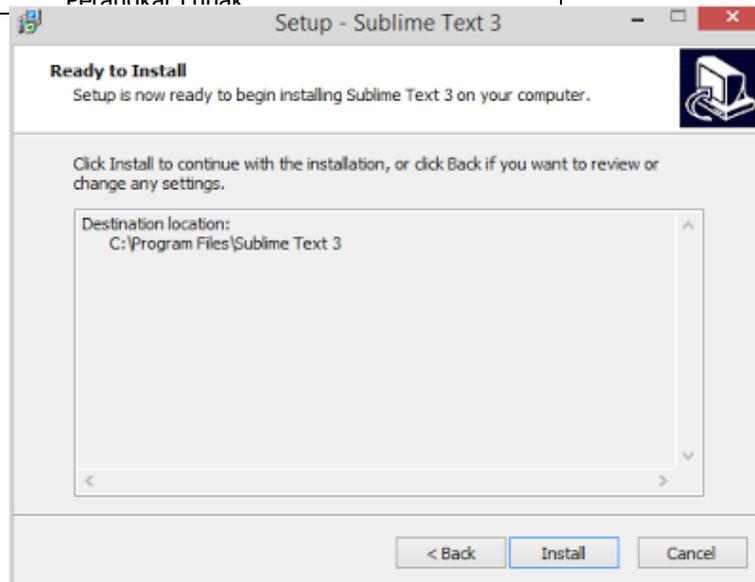
Gambar 2. 16 *setup sublime text*

c. Tentukan lokasi instalasinya, dan klik tombol *next*



Gambar 2. 17 *destination location*

d. Klik *next* sampai dengan tampilan *form Ready to Install*, klik tombol *Install*.



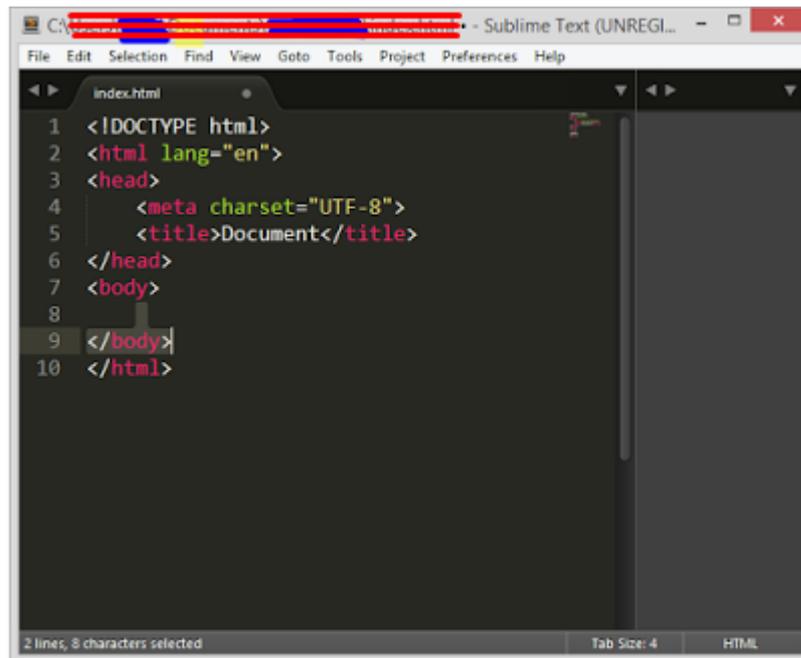
Gambar 2. 18 *ready to install*

- e. Tunggu hingga proses instalasi selesai.
- f. Jika proses instalasi selesai maka klik tombol *finish*.



Gambar 2. 19 *Completing Installation*

- g. Selesai dan jalankan aplikasi hasil instalasinya.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6 </head>
7 <body>
8
9 </body>
10 </html>
```

Gambar 2. 20 Hasil instalasi

B. Keterampilan Yang Diperlukan Dalam Menganalisa Kebutuhan Dokumentasi Web Dinamis Sesuai Interaksi Server Side.

1. Menginstall seluruh perangkat dan *aplikasi* pengembangan *web*.
2. Mengaktifkan fitur – fitur yang dibutuhkan dalam pembuatan *web*.

C. Sikap Kerja Yang Diperlukan Dalam Menganalisa Kebutuhan Dokumentasi Web Dinamis Sesuai Interaksi Server Side.

Harus bersikap secara:

1. Cermat dan teliti dalam menganalisis data;
2. Taat asas dalam mengaplikasikan langkah-langkah, panduan, dan pedoman yang dilakukan dalam menyusun tahapan penyajian;
3. Berpikir analitis serta *evaluatif* waktu melakukan analisis.

BAB III

MERANCANG SCRIPT SERVER SIDE

A. Pengetahuan Yang Diperlukan Dalam Merancang Script Server Side

1. Pengenalan PHP

Sebelum memulai kegiatan, sebaiknya kita mengenal terlebih dahulu *script server-side* yang kita gunakan yaitu PHP (*Hypertext Preprocessor*). PHP adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan *web* dinamis yang memproses data pada sisi *server*.

PHP merupakan singkatan dari **Personal Home Page**. Sesuai dengan namanya, PHP digunakan untuk membuat *website* pribadi. Dalam beberapa tahun perkembangannya, PHP menjadi bahasa pemrograman *web* yang *powerful*. PHP dapat digunakan dengan gratis (*free*) dan bersifat *Open Source*. PHP dirilis dalam lisensi *PHP License*, sedikit berbeda dengan lisensi *GNU General Public License (GPL)* yang biasa digunakan untuk proyek *Open Source*.

Apakah dokumen PHP itu?

- Dokumen PHP terdiri atas teks, tag dan *skrip HTML*.
- Dokumen PHP di proses dalam server dan kirim kembali ke *browser* sebagai plain HTML.
- Dokumen PHP berekstensi *.php*, *.php3* atau *.phtml*.

Beberapa kemudahan menggunakan PHP, diantaranya:

- PHP dapat berjalan pada bermacam *platform OS (Windows, Linux, Unix dll)*.
- PHP kompatibel dengan hampir semua *web server (apaceh, IIS dll)*.
- PHP dapat di *download* secara gratis melalui situs resminya.
- PHP mudah untuk dipelajari dan berjalan secara *efisien* pada *web server*.

2. Sintak Dasar PHP

Berikut ini beberapa sintak dasar pada PHP:

- Sintak php diawali dengan "<?php" dan diakhiri dengan "?>". Dokumen PHP dapat ditempatkan dimana saja dalam dokumen HTML.
- Sintak php dapat juga ditulis dengan "<?" dan diakhiri dengan "?>".
- Setiap baris sintak PHP harus diakhiri dengan titik koma (;). Titik koma adalah pemisah yang digunakan untuk membedakan satu *set instruksi* dengan *instruksi* lainnya.
- Dua sintak dasar untuk menampilkan teks dengan PHP yaitu dengan "*echo*" dan "*print*".

```
<?php
    echo "Semangat Belajar PHP";
?>
```

```
<?php
    print "Semangat Belajar PHP";
?>
```

3. Komentar Dalam PHP

Berikut ini penulisan sintak komentar:

- `//` = penulisan komentar dalam satu baris.
- `/*` dan `*/` = penulisan komentar lebih dari satu baris.

```
<?php
    // echo "Semangat Belajar PHP";
    // echo "Menampilkan baris ke 2";
    // echo "Menampilkan baris ke 3";
?>
```

```
<?php
    /*
    echo "Semangat Belajar PHP";
    echo "Menampilkan baris ke 2";
    echo "Menampilkan baris ke 3";
    */
?>
```

4. Variabel dan Aturan Penulisan

Variabel digunakan untuk menyimpan suatu nilai, seperti angka, string atau hasil dari suatu fungsi, sehingga dapat digunakan berulang kali pada program.

- Penulisan *variabel* dimulai dengan tanda "\$", di ikuti oleh nama *variabel*.
- Nama *variabel* harus dimulai dengan huruf atau karakter garis bawah (`_`) Sebuah nama *variabel* hanya dapat berisi karakter *alfa-numerik* dan garis bawah (`Az`, `0-9`, dan `_`)
- Nama *variabel* tidak boleh mengandung spasi. Jika *variabel* akan ditulis sebagai dua kata, maka dapat menggunakan garis bawah untuk

memisahkannya (contoh: `$my_variabel`), atau menggunakan huruf kapital (contoh: `$myVariabel`).

- Nama variabel adalah case sensitif ("y" dan "Y" adalah dua variabel yang berbeda)

Contoh penulisan variabel:

```
<?php
    $hasil = 1 + 5; // benar
    $HASIL = 1 + 5; // benar, berbeda dengan $hasil
    $_hasil = 1 + 5; // benar

    $1 = 1 + 5; // salah
    $1has = 1 + 5; // salah
    $#has = 1 + 5; // salah
    $hasil-ku = 1 + 5; // salah
?>
```

```
<?php
    $pekerjaan_ktp = "Guru";

    echo $pekerjaan_ktp;

?>
```

```
<?php
    $nama_ktp = "Agus Hendra";
    $pekerjaan_ktp = "Guru";

    echo $nama_ktp." adalah seorang ".$pekerjaan_ktp;

?>
```

5. Tipe Data

Di dalam PHP mengenal 3 macam tipe data, yaitu:

- Integer
Adalah bilangan bulat tanpa koma.

```
<?php
    $a = 1234 // desimal
    $b = - 1234 // negatif
    $c = 0123 // oktal
    $d = 0x12 // heksadesimal

?>
```

Pada sintak `$a = 1234`, berarti memberikan nilai 1234 pada variabel `$a`.

- Floating Point Number

Adalah bilangan pecahan. Terdapat tanda titik yang merupakan pemisah antara bilangan bulat dan pecahan.

```
<?php
    $a = 1.234 // bentuk biasa
    $b = 1.2e3 // bentuk eksponensial
?>
```

- String

Adalah teks yang merupakan gabungan huruf, angka, *whitespace* (spasi), dan berbagai karakter.

```
<?php
    $a = "ini adalah tipe data string";
?>
```

6. Operator Matematika

Tabel 3. 1 Operator Matematika

Operator	Nama	Contoh	Hasil
$x + y$	Penjumlahan	$2 + 2$	4
$x - y$	Pengurangan	$5 - 2$	3
$x * y$	Perkalian	$5 * 2$	10
x / y	Pembagian	$15 / 5$	3
$x \% y$	Modulus (Sisa hasil bagi)	$5 \% 210 \% 8$ $10 \% 2$	12 0
$-x$	Penyangkalan	-2	
$a . b$	Penggabungan	"Hi" . "Ha"	HiHa

7. Operator Penugasan

Tabel 3. 2 Operator penugasan

Assignment	Sama seperti	Keterangan
$x = y$	$x = y$	Nilai x sama dengan y
$x += y$	$x = x + y$	Penjumlahan
$x -= y$	$x = x - y$	Pengurangan
$x *= y$	$x = x * y$	Perkalian
$x /= y$	$x = x / y$	Pembagian
$x \% = y$	$x = x \% y$	Modulus
$a .= b$	$a = a . b$	Penggabungan

8. Operator Increment / Decrement

Tabel 3. 3 Operator Increment / Decrement

Operator	Nama	Keterangan
++ x	Pre-increment	Menambah x sebanyak satu, kemudian kembali ke x
x ++	Post-increment	Kembali ke x, kemudian menambah x sebanyak satu
— x	Pre-decrement	Mengurang x sebanyak satu, kemudian kembali ke x
x —	Post-decrement	Kembali ke x, kemudian mengurang x sebanyak satu

9. Operator Perbandingan

Tabel 3. 4 Operator Perbandingan

Operator	Nama	Keterangan	Contoh
x == y	Sama dengan	Benar jika x sama dengan y	5==8 Bernilai salah
x === y	Identik	Benar jika x sama dengan y, dan keduanya merupakan dari jenis yang sama	5==="5" Bernilai salah
x != y	Tidak sama dengan	Benar jika x tidak sama dengan	5!=8 Bernilai benar
x <> y	Tidak sama dengan	Benar jika x tidak sama dengan	5<>8 Bernilai benar
x !== y	Tidak identik	Benar jika x tidak sama dengan y, atau keduanya bukan dari jenis yang sama	5!== "5" Bernilai benar
x > y	Lebih besar dari	Benar jika x lebih besar dari y	5>8 Bernilai salah
x < y	Lebih kecil dari	Benar jika x lebih kecil dari y	5<8 Bernilai benar
x >= y	Lebih besar dari atau sama dengan	Benar jika x lebih besar dari atau sama dengan y	5>=8 Bernilai salah
x <= y	Lebih kecil dari atau sama dengan	Benar jika x kurang dari atau sama dengan y	5<=8 Bernilai benar

10. Operator Logika

Tabel 3. 5 Operator Logika

Operator	Nama	Keterangan	Contoh
x and y	And	Benar jika nilai x dan y bernilai benar	x=6y=3 (x < 10 and y > 1) Bernilai benar
x or y	Or	Benar jika salah satu atau keduanya (x dan y) bernilai benar	x=6y=3 (x==6 or y==5) Bernilai benar
x xor y	Xor	Benar jika x atau y bernilai benar, namun tidak keduanya	x=6y=3 (x==6 xor y==3) Bernilai salah
x && y	And	Benar jika kedua nilai x dan y adalah benar	x=6y=3 (x < 10 && y > 1) Bernilai benar
x y	Or	Benar jika salah satu atau keduanya (x dan y) bernilai benar	x=6y=3 (x==5 y==5) Bernilai salah
! x	Not	Benar jika nilai x tidak benar	x=6y=3 !(x==y) Bernilai benar

11. Operator Array

Tabel 3. 6 Operator Array

Operator	Name	Description
x + y	Penyatuan	Menyatukan x dan y
x == y	Persamaan	Benar jika x dan y memiliki nilai / pasangan yang sama
x === y	Identik	Benar jika x dan y memiliki nilai / pasangan yang sama dalam urutan yang sama dan jenis yang sama
x != y	Ketidaksamaan	Benar jika x tidak sama dengan y
x <> y	Ketidaksamaan	Benar jika x tidak sama dengan y
x !== y	Tidak identik	Benar jika x tidak identik dengan y

12. File dan Direktori

- Membuka dan menutup file

Sebelum mengakses suatu *file* melalui aplikasi yang dibuat untuk kemudian dibaca atau diisi data, pertama – tama kita harus membuka file tersebut menggunakan fungsi `fopen()`. Bentuk umum penggunaan fungsi `fopen()` adalah:

```
file_handler = fopen>NamaFile, Mode);
```

file handler adalah *pointer* ke *file* yang mengacu ke *file* yang kita buka, Nama File adalah nama *file* yang akan diakses (lengkap dengan *path*-nya jika *file* tidak berada dalam satu direktori dengan *file sintak program*).

Mode adalah jenis akses yang akan dilakukan, berikut ini penjelasan jenis *mode* dan fungsinya.

Tabel 3. 7 Mode dan Fungsi

Jenis Mode	Fungsi
r	Membuka file untuk proses pembacaan (<i>reading</i>) dengan standar <i>read only</i> dan tidak dapat menulis apapun ke dalam file.
r+	Membuka file untuk proses pembacaan (<i>reading</i>) dan penulisan. Jika file belum ada maka file akan dibuat. Sebaliknya, jika file telah ada, maka data baru akan ditambahkan pada bagian awal, yaitu dengan melakukan penimpaan (<i>overwriting</i>) terhadap data lama.
w	Membuka file untuk proses penulisan (<i>writing</i>). Jika file belum ada maka file akan dibuat. Sebaliknya, jika

Perangkat Lunak

	file yang didefinisikan sudah ada, maka file tersebut akan dihapus, selanjutnya akan dibuat file baru (dengan data kosong).
w+	Membuka file untuk proses penulisan dan pembacaan. Konsepnya sama seperti pada mode w
a	Membuka file untuk proses penambahan data (<i>appending</i>). Jika file belum ada maka file akan dibuat. Sebaliknya, jika file telah ada maka data yang dimasukkan ke dalam file akan ditambahkan pada bagian akhir dari data lama (<i>end of file</i>)
a+	Membuka file untuk proses pembacaan dan penambahan data. Konsepnya sama seperti pada mode a.

Jika file yang dibuka merupakan *file biner* (bukan *file teks*) maka perlu ditambahkan huruf "b" pada mode yang terdapat pada fungsi fopen(). Contoh:

```
$file_handle = fopen($binaryfile, "rb+");
```

Setelah selesai membaca atau menulis file melalui aplikasi yang dibuat, kita perlu menutup file tersebut menggunakan fungsi fclose(). Contoh:

```
Fclose($file_handle);
```

- Membaca isi file

PHP menyediakan sejumlah fungsi untuk melakukan pembacaan data dari file. Metode yang digunakan pun beragam, ada yang dibaca per baris, per *byte*, dan ada juga yang dibaca per karakter.

Sebelum melakukan praktikum ini, pertama-tama buat terlebih dahulu file teks, contoh dengan isi seperti berikut ini:

```
Chandra Eka
Thomas Edison
Indah Putri
Agus Pencos
Mamik Prakoso
```

Simpan file tersebut pada direktori modul dengan nama contoh.txt.

Praktikum 1: Membaca file per baris menggunakan fungsi fgets().

Nama file: baca_file_1.php

```

baca_file_1.php x
<!DOCTYPE html>
<html>
<head>
  <title>Membaca File</title>
</head>
<body>
  <h2>Demo membaca file (Bagian 1)</h2>
  <?php
    $file = "../modul/contoh.txt";
    $fp = fopen($file,"r");

    echo "Membaca file $file adalah:<br>";
    while (!feof($fp)) {
      $baris = fgets($fp, 1024);
      echo "<strong>$baris</strong>";
    }
    fclose($fp);
  ?>
</body>

```

Tampilan:



Demo membaca file (Bagian 1)

Membaca file ../modul/contoh.txt adalah:

Chandra Eka
Thomas Edison
Indah Putri
Agus Pencos
Mamik Prakoso

Pada *praktikum* di atas kita membaca *file* dengan cara membacanya **per-baris**, yaitu dengan menggunakan fungsi `fgets()`. Setiap baris yang dibaca akan ditampung ke dalam *variabel* `$baris` dan kemudian ditampilkan ke halaman web menggunakan perintah `echo`. Proses ini akan diulang (menggunakan *statement while*) sampai fungsi `feof()` mengembalikan nilai *true*. Fungsi `feof()` di atas berguna untuk memeriksa apakah proses pembacaan sudah sampai ke bagian akhir file atau belum. Nilai 1024 yang terdapat pada pemanggilan fungsi `fgets()` adalah nilai *byte* maksimal yang dapat dibaca pada satu baris data tertentu (nilai tersebut sesuai dengan kebutuhan).

Praktikum 2: Membaca file per *n* byte menggunakan fungsi `fread()`

Nama file: `baca_file_2.php`

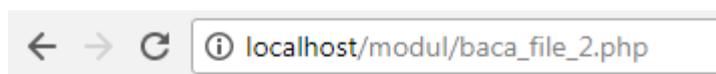
```

baca_file_2.php x
<!DOCTYPE html>
<html>
<head>
  <title>Membaca File</title>
</head>
<body>
  <h2>Demo membaca file (Bagian 2)</h2>
  <?php
    $file = "../modul/contoh.txt";
    $fp = fopen($file,"r");

    echo "Membaca file $file adalah:<br>";
    while (!feof($fp)) {
      $pecahan_data = fread($fp, 30);
      echo "<strong>$pecahan_data</strong><br>";
    }
    fclose($fp);
  ?>
</body>

```

Tampilan:



Demo membaca file (Bagian 2)

Membaca file `../modul/contoh.txt` adalah:

**Chandra Eka Thomas Edison In
dah Putri Agus Pencos Mamik
Prakoso**

Fungsi `fread()` akan membaca sejumlah *byte* dari isi file. Pada contoh di atas, kita memasukkan nilai 30 sebagai ukuran *byte* yang akan kita baca.

```
$pecahan_data = fread($fp, 30);
```

Dengan demikian, pada setiap pengulangan yang dilakukan di atas, `fread()` akan mengambil 30 *byte* data dari file `contoh.txt` dan menampilkannya ke layar *web browser*.

Praktikum 3: Membaca file per karakter menggunakan fungsi `fgetc()`

Nama file: `baca_file_3.php`

```

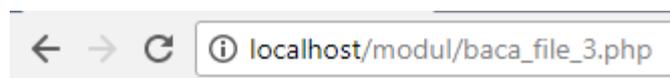
baca_file_3.php x
<!DOCTYPE html>
<html>
<head>
  <title>Membaca File</title>
</head>
<body>
  <h2>Demo membaca file (Bagian 2)</h2>
  <?php
    $file = "../modul/contoh.txt";
    $fp = fopen($file,"r");

    echo "Membaca file $file adalah:<br>";
    while (!feof($fp)) {
      $karakter = fgetc($fp);

      echo "<strong>$karakter</strong><br>";
    }
    fclose($fp);
  ?>
</body>

```

Tampilan:



Demo membaca file (Bagian 2)

Membaca file ../modul/contoh.txt adalah:

C
h
a
n
d
r
a

Pada contoh di atas, kita menangkap setiap karakter yang dibaca ke dalam variabel \$karakter, menggunakan kode berikut:

```
$karakter = fgetc($fp);
```

Proses ini akan diulang sampai fungsi feof() menghasilkan nilai true.

- Menulis dan Menambahkan Data ke dalam file

Proses penulisan (*write*) dan penambahan (*append*) data ke dalam suatu file sebenarnya merupakan proses yang sama. Perbedaannya hanya terletak pada saat pemanggilan fungsi fopen(). Pada saat akan menulis ke file, seharusnya menggunakan mode "w" ketika memanggil fungsi fopen().

```
$fp = fopen($file, "w");
```

Pada saat akan menambahkan data pada suatu file yang sebelumnya sudah ada datanya, kita tinggal menggunakan mode "a".

```
$fp = fopen($file, "a");
```

PHP menyediakan beberapa fungsi untuk keperluan penulisan data ke dalam file. Namun, saat ini kita hanya akan menggunakan fungsi fwrite() dan fputs().

Praktikum 3: Menulis data ke dalam file menggunakan fungsi fwrite() dan fputs().

Pada praktikum kali ini kita buat file baru dengan nama contoh_1.txt dan menempatkannya di direktori C:\xampp\htdocs\modul.

```
<!DOCTYPE html>
<html>
<head>
  <title>Menulis File</title>
</head>
<body>
  <h2>Demo menulis dan menambah data ke dalam file</h2>
  <?php

    $file = "../modul/contoh_1.txt";

    echo "<p>Menulis data pada file $file adalah:</p>";
    $fp = fopen($file,"w");
    fwrite($fp, "Berikut ini contoh penulisan teks ke dalam file.\n");
    fclose($fp);

    echo "<p>Menambah data pada file $file adalah:</p>";
    $fp = fopen($file,"a");
    fputs($fp, "Berikut ini contoh penambahan teks ke dalam file.\n");
    fclose($fp);

    echo "isi file $file adalah: <br>";

    $fp = fopen($file,"r");
    while (!feof($fp)) {
      $baris = fgets($fp, 1024);
      echo "<strong>$baris</strong><br>";
    }
  ?>
</body>
```

Tampilan:



Demo menulis dan menambah data ke dalam file

Menulis data pada file ../modul/contoh_1.txt adalah:

Menambah data pada file ../modul/contoh_1.txt adalah:

isi file ../modul/contoh_1.txt adalah:

Berikut ini contoh penulisan teks ke dalam file.

Berikut ini contoh penambahan teks ke dalam file.

Proses ini dibuat dengan cara melewati mode "w" pada saat pemanggilan fungsi fopen(). Selanjutnya, kita menulis data ke dalam file tersebut menggunakan fungsi fwrite().

```
fwrite($fp, "Berikut ini contoh penulisan teks ke dalam file.\n");
```

Ketika selesai menulis file, kita menutup koneksi dengan file tersebut menggunakan fungsi fclose().

Langkah selanjutnya, kita membuka kembali file yang telah dibuat dan ditulis di atas untuk ditambah data yang kita inginkan. Untuk keperluan ini, Anda perlu menggunakan mode "a" pada saat memanggil fungsi fopen(). Dalam contoh ini, kita menggunakan fungsi fputs() untuk menambah data ke dalam file.

```
fputs($fp, "Berikut ini contoh penambahan teks ke dalam file.\n");
```

Setelah selesai ditambah datanya, koneksi file kembali ditutup menggunakan fungsi fclose(). Untuk melihat isi file yang telah terbuat, kita dapat melakukan pembacaan terhadap file tersebut menggunakan fungsi fgets().

- Mengubah Nama File.

PHP menyediakan fungsi rename() yang dapat digunakan untuk mengubah nama file. Fungsi ini memiliki dua parameter. Parameter pertama adalah nama file yang akan diubah namanya dan parameter kedua adalah nama baru yang akan digunakan sebagai pengganti.

Pada praktikum kali ini kita buat file baru dengan nama hai.txt dan menempatkannya di direktori C:\xampp\htdocs\modul.

```
<!DOCTYPE html>
<html>
<head>
  <title>Rename File</title>
</head>
<body>
  <h2>Demo Mengganti Nama file</h2>
  <?php
    $file = "../modul/hai.txt";

    if(rename($file, "hallo.txt")){
      echo "Perubahan nama file BERHASIL";
    }else{
      echo "Perubahan nama file GAGAL";
    }

  ?>
</body>
```

Jika kode di atas dieksekusi, nama file hai.txt akan diubah menjadi hello.txt. Fungsi rename() akan mengembalikan nilai *true* jika proses perubahan nama file berhasil dilakukan. Jika sebaliknya, nilai yang akan dikembalikan adalah *false*.

- Menyalin File.

Untuk menyalin file di dalam PHP, kita dapat menggunakan fungsi copy(). Parameter pertama dari fungsi ini menunjukkan file yang akan disalin dan parameter kedua menunjukkan file tujuan.

```
<!DOCTYPE html>
<html>
<head>
  <title>Salin File</title>
</head>
<body>
  <h2>Demo Menyalin File</h2>
  <?php
    $file = "../modul/hai.txt";

    if(rename($file, "salinan.txt")){
      echo "file BERHASIL di salin";
    }else{
      echo "file GAGAL di salin";
    }

  ?>
</body>
```

- Menghapus File

Dalam PHP, file dihapus menggunakan fungsi `unlink()`. Fungsi ini hanya membutuhkan satu parameter, yaitu nama file yang akan dihapus. Sertakan lokasi (*path*) dari file bersangkutan jika file yang dihapus tidak berada satu direktori dengan file kode.

```
<!DOCTYPE html>
<html>
<head>
  <title>Hapus File</title>
</head>
<body>
  <h2>Demo Menghapus File</h2>
  <?php
    $file = "../modul/hai.txt";

    if(unlink($file)){
      echo "file BERHASIL di Hapus";
    }else{
      echo "file GAGAL di Hapus";
    }
  ?>
</body>
```

13. Form HTML

- Pemrosesan Form

Merupakan operasi mendasar yang sering digunakan dalam konteks pengembangan aplikasi web dinamis. Sebagai contoh: untuk menerima masukan dari user, tentu diperlukan sekali form isian dan cara pemrosesannya. Begitu pula halnya ketika administrator ingin masuk ke sistem, tentu memerlukan suatu antarmuka penghubung. Intinya, keberadaan form dan pemrosesannya mutlak diperlukan dalam membangun aplikasi web yang interaktif.

Teknik pengiriman form dapat dilakukan melalui tiga metode: POST, GET, dan kombinasi keduanya. Dalam konteks PHP, parameter – parameter GET dapat dibaca melalui superglobal `$_GET`, sedangkan POST melalui `$_POST`. Selain itu, keduanya juga dapat dibaca menggunakan `$_REQUEST`.

- Konsep penggunaan Tag `<form>` pada HTML

Tag HTML dalam pembuatan *form* adalah `<form> isian </form>`

Dalam bekerja dengan *form*, dibuat tombol *submit* yang digunakan untuk mengirimkan semua data yang ditulis oleh *user* melalui *form* ke *server*. Setiap *form* yang ditampilkan pada halaman web pasti akan memiliki tombol *submit*.

Tag `<form>` terdapat 2 atribut penting yaitu: *Action*, *Method*

14. Atribut action dan Atribut method

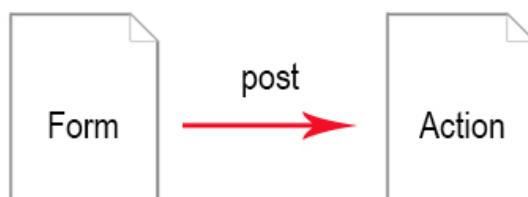
Atribut action digunakan untuk menunjukkan URL file PHP yang akan digunakan untuk memproses data. File tersebut dieksekusi ketika tombol *submit* di klik.

Atribut method digunakan untuk menentukan bagaimana cara informasi dari suatu *form* (komputer *klien*) dikirimkan ke komputer *server*. Kita dapat menggunakan salah satu dari dua cara: menggunakan *method get* atau *post*

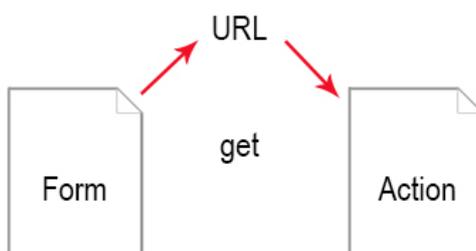
```
<form action="link ke URL" method="POST/GET"> </form>
```

Perbedaan *method GET* dan *POST*

- *Method POST* akan mengirimkan data atau nilai langsung ke action untuk ditampilkan, tanpa menampilkan pada URL.



- Sedangkan *method GET* akan menampilkan data/nilai pada URL, kemudian akan ditampilkan oleh action.



- *Method POST* menggunakan variable `$_POST` untuk menampung data/nilai.

```
// tampung data dengan variabel post  
$data = $_POST['nama'];  
// cetak  
echo $data;
```

- *Method GET* menggunakan variable \$_GET untuk menampung data/ nilai.

localhost/modul/form.php?nama=agus

```
// tampung data dengan variabel get  
$data = $_GET['nama'];  
// cetak  
echo $data;
```

15. Elemen Kontrol di dalam Form.

- Elemen *Text*

Adalah alat input *text* yang hanya mampu menampung satu baris. Contoh

sintak: `<input type="text">`

- Elemen *textarea*

Adalah alat input *text* yang mampu menampung lebih dari satu baris. Contoh

sintak: `<textarea rows="4"></textarea>`

- Elemen *checkbox*

Adalah alat inputan yang digunakan untuk memilih lebih dari satu opsi pilihan. Contoh sintak:

```
<input type="checkbox" name="vehicle" value="becak" checked> Becak<br>  
<input type="checkbox" name="vehicle" value="sepeda"> Sepeda Motor<br>  
<input type="checkbox" name="vehicle" value="mobil"> Mobil<br>
```

- Elemen *radio button*

Adalah alat inputan yang digunakan untuk memilih satu opsi dari opsi pilihan. Contoh sintak:

```
<input type="radio" name="gender" value="male"> Pria<br>  
<input type="radio" name="gender" value="female"> Perempuan<br>
```

- Elemen *listbox* dan *combobox*

Elemen list dibagi menjadi dua, yaitu sebagai *listbox* dan *combobox*.

Jika menggunakan elemen *list box*, maka kita perlu menentukan atribut *size* pada tag `<select>` dengan nilai LEBIH BESAR dari 1;

```
<select size="5">
  <option value="Islam">Islam</option>
  <option value="Katolik">Katolik</option>
  <option value="Kristen">Kristen</option>
  <option value="Hindu">Hindu</option>
  <option value="Budha">Budha</option>
</select>
```

Sedangkan untuk elemen *combo box*, kita dapat menentukan atribut *size* dengan nilai 1 atau tanpa menggunakan atribut *size*. Contoh sintak:

```
<select>
  <option value="Islam">Islam</option>
  <option value="Katolik">Katolik</option>
  <option value="Kristen">Kristen</option>
  <option value="Hindu">Hindu</option>
  <option value="Budha">Budha</option>
</select>
```

- Elemen *Hidden*

Adalah inputan yang tidak terlihat oleh user namun telah disisipkan data oleh programmer. Contoh sintak:

```
<input type="hidden" value="data tersembunyi">
```

- Elemen *Password*

Adalah inputan yang bersifat rahasia. Contoh sintak:

```
<input type="password">
```

16. Latihan mengirimkan data menggunakan metode GET.

Buat dokumen file dengan nama: latihan_get.html

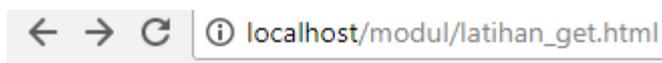
```
<!DOCTYPE html>
<html>
<head>
  <title>FORM File</title>
</head>
<body>
  <h2>Latihan Kirim Data GET</h2>

  <form action="get.php" method="get">
    Login:
    <input type="text" name="nama" > &nbsp;
    <input type="submit" value="Kirim">

  </form>

</body>
</html>
```

Hasil tampilan browser:



Latihan Kirim Data GET

Login:

Buat dokumen file dengan nama: get.php

```
<?php
  $nama = $_GET["nama"];
  if(isset($nama)){
    echo "Selamat Datang <strong>$nama</strong>, ".
      "Terima Kasih sudah mengunjungi website kami.";
  }
?>
```

Hasil tampilan browser:



Selamat Datang **Agus Pencos**, Terima Kasih sudah mengunjungi website kami.

17. Latihan mengirimkan data menggunakan metode POST.

Buat dokumen file dengan nama: latihan_post.html

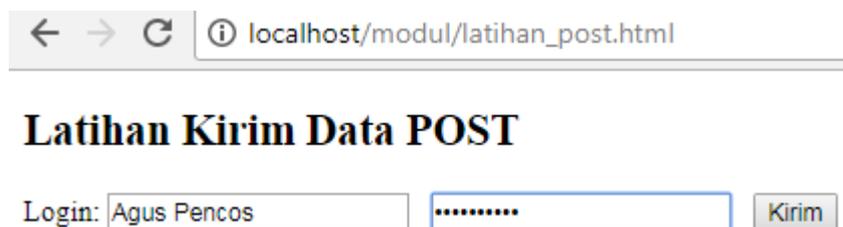
```
<!DOCTYPE html>
<html>
<head>
  <title>FORM File</title>
</head>
<body>
  <h2>Latihan Kirim Data POST</h2>

  <form action="post.php" method="POST">
    Login:
    <input type="text" name="nama_input"> &nbsp;
    <input type="password" name="pass_input"> &nbsp;
    <input type="submit" value="Kirim">

  </form>

</body>
</html>
```

Hasil tampilan browser:



← → ↻ ⓘ localhost/modul/latihan_post.html

Latihan Kirim Data POST

Login:

Buat dokumen file dengan nama: post.php

```
$nama = $_POST["nama_input"];
if(isset($nama)){
  echo "Selamat Datang <strong>$nama</strong>, ".
    "Terima Kasih sudah mengunjungi website kami.";
}
```

Hasil tampilan browser:



← → ↻ ⓘ localhost/modul/post.php

Selamat Datang Agus Pencos, Terima Kasih sudah mengunjungi website kami.

18. Kolaborasi dengan berbagai macam element.

Nama file: latihan_form_biografi.html

Deskripsi: Dokumen PHP form inputan data biografi.

Tampilan sintak:

```
<!DOCTYPE html>
<html>
<head>
  <title>Form Biodata</title>
</head>
<body>
  <h2>Input Biodata</h2>
  <form action="hasil_biografi.php" method="POST">
    <table>
      <tr>
        <td>Nama</td>
        <td>:</td>
        <td><input type="text" name="nama_input" placeholder="Nama lengkap Anda"></td>
      </tr>
      <tr>
        <td>Email</td>
        <td>:</td>
        <td><input type="text" name="email_input" placeholder="Email Anda"></td>
      </tr>
      <tr>
        <td>Jenis Kelamin</td>
        <td>:</td>
        <td>
          <input type="radio" name="jk_input" value="Laki-laki" checked="checked"> Laki-laki
          <input type="radio" name="jk_input" value="Perempuan"> Perempuan
        </td>
      </tr>
      <tr>
        <td>Agama</td>
        <td>:</td>
        <td>
          <select name="agama_input">
            <option value="Islam">Islam</option>
            <option value="Katolik">Katolik</option>
            <option value="Protestan">Protestan</option>
            <option value="Hindu">Hindu</option>
            <option value="Budha">Budha</option>
          </select>
        </td>
      </tr>
      <tr>
        <td>Keterangan</td>
        <td>:</td>
        <td>
          <textarea rows="4" name="keterangan_input"></textarea>
        </td>
      </tr>
      <tr>
        <td colspan="2"></td>
        <td>
          <input type="submit" value="Kirim">
        </td>
      </tr>
    </table>
  </form>
</body>
</html>
```

Hasil tampilan browser:

Input Biodata

Nama :

Email :

Jenis Kelamin: Laki-laki Perempuan

Agama :

Keterangan :

Nama file: hasil_biografi.php

Deskripsi: Dokumen PHP untuk merespon data dari dokumen latihan_form_biografi.html

Tampilan sintak:

```
<?php

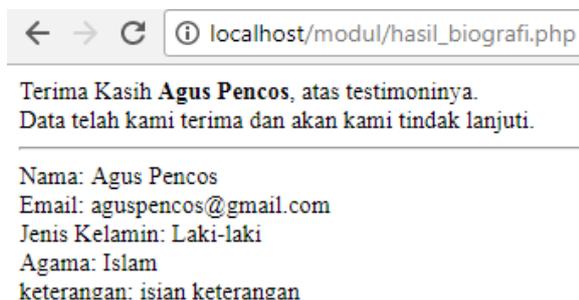
$name = $_POST["nama_input"];
$email = $_POST["email_input"];
$jk = $_POST["jk_input"];
$agama = $_POST["agama_input"];
$keterangan = $_POST["keterangan_input"];

if(!$name OR !$email OR !$agama ){
    echo "GAGAL menampilkan - Data kurang lengkap. ";
    echo "<a href='latihan_form_biografi.html'><button>Kembali</button></a>";
}else{
    echo "Terima Kasih <strong>$name</strong>, ".
        "atas testimoninya. <br> Data telah kami terima dan ".
        "akan kami tindak lanjuti.<hr>";

    echo "Nama: $name <br>";
    echo "Email: $email <br>";
    echo "Jenis Kelamin: $jk <br>";
    echo "Agama: $agama <br>";
    echo "keterangan: $keterangan <br>";
}

?>
```

Hasil tampilan browser:



19. PHP dan Database MySQL

- Merancang Struktur Database

Untuk membuat aplikasi berita secara sederhana, langkah pertama adalah merancang *database* dan tabel – tabel yang diperlukan. Dalam pembuatan *database* dapat menggunakan *phpMyAdmin* dengan mengakses <http://localhost/phpmyadmin> atau dengan menggunakan aplikasi *editor* database lainnya.

Buatlah *database* dengan nama "modul_db", kemudian buat tabel berikut:

Tabel berita:

Name	Type	Length	Decimals	Not null	
id_berita	int	5	0	<input checked="" type="checkbox"/>	 1
id_kategori	int	3	0	<input checked="" type="checkbox"/>	
judul	varchar	100	0	<input checked="" type="checkbox"/>	
headline	text	0	0	<input checked="" type="checkbox"/>	
isi	text	0	0	<input checked="" type="checkbox"/>	
pengirim	varchar	15	0	<input checked="" type="checkbox"/>	
tanggal	datetime	0	0	<input checked="" type="checkbox"/>	

Tabel berita query:

```
CREATE TABLE `berita` (
  `id_berita` int(5) unsigned NOT NULL AUTO_INCREMENT,
  `id_kategori` int(3) unsigned NOT NULL DEFAULT '0',
  `judul` varchar(100) NOT NULL DEFAULT '',
  `headline` text NOT NULL,
  `isi` text NOT NULL,
  `pengirim` varchar(15) NOT NULL DEFAULT '',
  `tanggal` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  PRIMARY KEY (`id_berita`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Tabel kategori:

Name	Type	Length	Decimals	Not null	
id_kategori	int	3	0	<input checked="" type="checkbox"/>	 1
nm_kategori	varchar	30	0	<input checked="" type="checkbox"/>	
deskripsi	varchar	200	0	<input checked="" type="checkbox"/>	

Tabel kategori query:

```
CREATE TABLE `kategori` (
  `id_kategori` int(3) unsigned NOT NULL AUTO_INCREMENT,
  `nm_kategori` varchar(30) NOT NULL DEFAULT '',
  `deskripsi` varchar(200) NOT NULL DEFAULT '',
  PRIMARY KEY (`id_kategori`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Catatan: Isikan sejumlah data ke tabel **kategori** langsung di database

- Membuat File Koneksi Database

Nama file: koneksi.php

Deskripsi: Dokumen PHP untuk koneksi ke database.

Tampilan sintak:

```
<?php
    $host = "localhost";
    $user = "root";      //user
    $pass = "";         //password
    $dbnm = "modul_db"; //nama database

    $conn = mysqli_connect ($host, $user, $pass);
    if ($conn) {
        $buka = mysqli_select_db ($conn, $dbnm);
        if (!$buka) {
            die ("Database tidak dapat dibuka");
        }
    } else {
        die ("Server MySQL tidak terhubung");
    }
?>
```

- Membuat Halaman Input Berita

Nama file: input_berita.php

Deskripsi: Dokumen PHP untuk memasukkan data berita.

Tampilan sintak:

Tampilan browser:

The screenshot shows a web browser window with the address bar displaying 'localhost/modul/input_berita.php'. Below the address bar are navigation links: 'Halaman Depan', 'Arsip Berita', and 'Input Berita'. The main content area is titled 'Input Berita' and contains a form with the following fields and controls:

- Judul Berita: A text input field.
- Kategori: A dropdown menu with 'Sport' selected.
- Headline Berita: A text input field.
- Isi Berita: A large text area for the main content.
- Pengirim: A text input field.

At the bottom of the form are two buttons: 'Input Berita' and 'Cancel'.

- Membuat Halaman Depan

Nama file: index.php

Deskripsi: Dokumen PHP untuk menampilkan 5 berita terbaru di halaman utama.

Tampilan sintak:

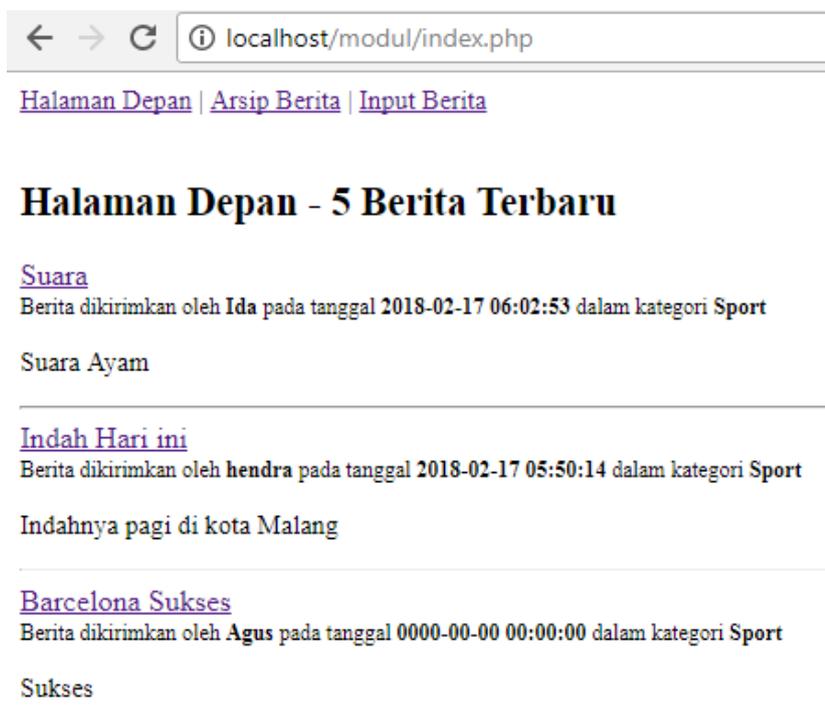
```
<?php
    include "koneksi.php";
?>
<html>
<head>
    <title>Index Berita</title>
</head>
<body>
    <a href="index.php">Halaman Depan</a> |
    <a href="arsip_berita.php">Arsip Berita</a> |
    <a href="input_berita.php">Input Berita</a>
    <br><br>
    <h2>Halaman Depan - 5 Berita Terbaru</h2>
    <?php
        $query = "SELECT A.id_berita, B.nm_kategori, A.judul, A.headline, A.pengirim, A.tanggal
        FROM berita A, kategori B WHERE A.id_kategori=B.id_kategori
        ORDER BY A.id_berita DESC LIMIT 0,5";

        //eksekusi perintah mysql
        $sql = mysqli_query($conn,$query);
        // $conn adalah variabel koneksi yg terdapat pada file: koneksi.php

        //mengeluarkan output hasil query mysql
        while ($hasil = mysqli_fetch_array ($sql)) {
            $id_berita = $hasil['id_berita'];
            $kategori = stripslashes ($hasil['nm_kategori']);
            $judul = stripslashes ($hasil['judul']);
            $headline = nl2br(stripslashes($hasil['headline']));
            $pengirim = stripslashes ($hasil['pengirim']);
            $tanggal = stripslashes ($hasil['tanggal']);

            //tampilkan berita
            echo "<font size='4'><a href='berita_lengkap.php?id=$id_berita'>$judul</a></font><br>";
            echo "<small>Berita dikirimkan oleh <b>$pengirim</b> pada tanggal <b>$tanggal</b>
            dalam kategori <b>$kategori</b></small>";
            echo "<p>$headline</p>";
            echo "<hr>";
        }
    ?>
</body>
</html>
```

Tampilan browser:



- Membuat Halaman Berita Lengkap

Nama file: berita_lengkap.php

Deskripsi: Dokumen PHP untuk menampilkan seluruh berita lengkap

Tampilan sintak:

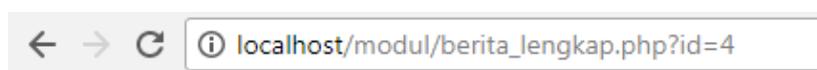
```
<?php
include "koneksi.php";

if (isset($_GET['id'])) {
    $id_berita = $_GET['id'];
} else {
    die ("Error. No Id Selected! ");
}
?>
<html>
<head><title>Berita Lengkap</title></head>
<body>
<a href="index.php">Halaman Depan</a> |
<a href="arsip_berita.php">Arsip Berita</a> |
<a href="input_berita.php">Input Berita</a>
<br><br>
<h2>Berita Lengkap</h2>
<?php
$query = "SELECT A.id_berita, B.nm_kategori, A.judul, A.isi, A.pengirim, A.tanggal
        FROM berita A, kategori B
        WHERE A.id_kategori=B.id_kategori && A.id_berita='id_berita'";
$sql = mysqli_query ($conn,$query);

$hasil = mysqli_fetch_array ($sql);
$id_berita = $hasil['id_berita'];
$kategori = stripslashes ($hasil['nm_kategori']);
$judul = stripslashes ($hasil['judul']);
$isi = nl2br(stripslashes ($hasil['isi']));
$pengirim = stripslashes ($hasil['pengirim']);
$tanggal = stripslashes ($hasil['tanggal']);
// fungsi stripslashes adalah untuk dapat menampilkan kata meskipun terdapat karakter khusus, contoh: ',',/,,* dst
// nl2br adalah Fungsi yang digunakan untuk menyisipkan HTML <br /> pada setiap baris baru dalam string

//tampilkan berita
echo "<font size=5 color=blue>$judul</font><br>";
echo "<small>Berita dikirimkan oleh <b>$pengirim</b> pada tanggal <b>$tanggal</b> dalam kategori <b>$kategori</b></small>";
echo "<p>$isi</p>";
?>
</body>
</html>
```

Tampilan browser:



[Halaman Depan](#) | [Arsip Berita](#) | [Input Berita](#)

Berita Lengkap

Suara

Berita dikirimkan oleh **Ida** pada tanggal **2018-02-17 06:02:53** dalam kategori **Sport**

berikut ini adalah suara ayam

- Membuat Halaman Arsip Berita

Nama file: arsip_berita.php

Deskripsi: Dokumen PHP untuk menampilkan keseluruhan berita.

Tampilan sintak:

```
<?php
    include "koneksi.php";
?>
<html>
<head>
    <title>Arsip Berita</title>
    <script language="javascript">
        function tanya() {
            if (confirm ("Apakah Anda yakin akan menghapus berita ini ?")) {
                return true;
            } else {
                return false;
            }
        }
    </script>
</head>
<body>
    <a href="index.php">Halaman Depan</a> |
    <a href="arsip_berita.php">Arsip Berita</a> |
    <a href="input_berita.php">Input Berita</a>
    <br><br>
    <h2>Arsip Berita</h2>
    <ol>
        <?php
            $query = "SELECT A.id_berita, B.nm_kategori, A.judul, A.pengirim, A.tanggal
                FROM berita A, kategori B WHERE A.id_kategori=B.id_kategori
                ORDER BY A.id_berita DESC";
            $sql = mysqli_query ($conn,$query);
            while ($hasil = mysqli_fetch_array ($sql)) {
                $id_berita = $hasil['id_berita'];
                $kategori = stripslashes ($hasil['nm_kategori']);
                $judul = stripslashes ($hasil['judul']);
                $pengirim = stripslashes ($hasil['pengirim']);
                $tanggal = stripslashes ($hasil['tanggal']);

                echo "<li><a href='berita_lengkap.php?id=$id_berita'>$judul</a><br>";
                echo "<small>Berita dikirimkan oleh <b>$pengirim</b> pada tanggal <b>$tanggal</b> dalam kategori
                    <b>$kategori</b><br>";
                echo "<b>Action : </b><a href='edit_berita.php?id=".base64_encode($id_berita)."'>Edit</a> | ";
                // base64_encode bertujuan untuk menyembunyikan data penting agar tidak diketahui
                // oleh orang lain.
                echo "<a href='delete_berita.php?id=$id_berita' onClick='return tanya()'>Delete</a>";
                echo "</small></li><br><br>";
            }
        ?>
    </ol>
</body>
</html>
```

Tampilan Browser:



- Membuat Halaman Edit Berita

Nama file: edit_berita.php

Deskripsi: Dokumen PHP untuk menampilkan halaman edit berita terpilih.

Tampilan sintak:

```
<?php
include "koneksi.php";

if (!(base64_decode($_GET['id']))) {
    die ("Error. No Id Selected! ");
} else {
    $id_berita = base64_decode($_GET['id']);
    // base64_decode untuk Membaca data penting yang telah disembunyikan.
    // sebelumnya (di file arsip_berita.php).
}

$query = "SELECT id_berita, id_kategori, judul, headline, isi, pengirim, tanggal
        FROM berita WHERE id_berita='$id_berita'";
$sql = mysqli_query ($conn,$query);
$hasil = mysqli_fetch_array ($sql);
$id_berita = $hasil['id_berita'];
$id_kategori = stripslashes ($hasil['id_kategori']);
$judul = stripslashes ($hasil['judul']);
$headline = stripslashes ($hasil['headline']);
$isi = stripslashes ($hasil['isi']);
$pengirim = stripslashes ($hasil['pengirim']);
$tanggal = stripslashes ($hasil['tanggal']);

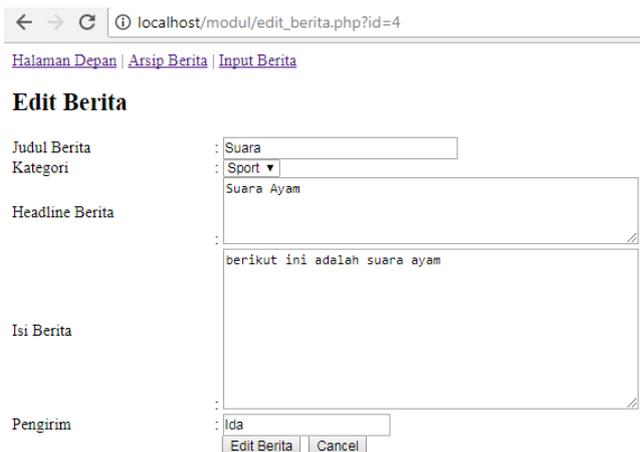
//proses edit berita
if (isset($_POST['Edit'])) {
    $id_berita = $_POST['hidberita'];
    $judul = addslashes (strip_tags ($_POST['judul']));
    $kategori = $_POST['kategori'];
    $headline = addslashes (strip_tags ($_POST['headline']));
    $isi_berita = addslashes (strip_tags ($_POST['isi']));
    $pengirim = addslashes (strip_tags ($_POST['pengirim']));
    //update berita
    $query = "UPDATE berita SET id_kategori='$kategori',judul='$judul',headline='$headline',
            isi='$isi_berita',pengirim='$pengirim'
            WHERE id_berita='$id_berita'";
    $sql = mysqli_query ($conn,$query);
    if ($sql) {
        echo "<h2><font color=blue>Berita telah berhasil diedit</font></h2>";
        header('Location: arsip_berita.php');
    } else {
        echo "<h2><font color=red>Berita gagal diedit</font></h2>";
    }
}
?>
```

```

<html>
<head>
<title>Edit Berita</title>
</head>
<body>
<a href="index.php">Halaman Depan</a> |
<a href="arsip_berita.php">Arsip Berita</a> |
<a href="input_berita.php">Input Berita</a>
<br><br>
<FORM ACTION="" METHOD="POST" NAME="input">
<table cellpadding="0" cellspacing="0" border="0" width="700">
<tr>
<td colspan="2"><h2>Edit Berita</h2></td>
</tr>
<tr>
<td width="200">Judul Berita</td>
<td><input type="text" name="judul" size="30" value="<?php echo $judul ?>"></td>
</tr>
<tr>
<td>Kategori</td>
<td><select name="kategori">
<?php
$query = "SELECT id_kategori, nm_kategori FROM kategori ORDER BY nm_kategori";
$sql = mysqli_query ($conn,$query);
while ($hasil = mysqli_fetch_array ($sql)) {
    $selected = ($hasil['id_kategori']==$id_kategori) ? "selected" : "";
    echo "<option value='".$hasil[id_kategori]'"$selected>$hasil[nm_kategori]</option>";
}
?>
</select>
</td>
</tr>
<tr>
<td>Headline Berita</td>
<td><textarea name="headline" cols="50" rows="4"><?php echo $headline?></textarea></td>
</tr>
<tr>
<td>Isi Berita</td>
<td><textarea name="isi" cols="50" rows="10"><?php echo $isi?></textarea></td>
</tr>
<tr>
<td>Pengirim</td>
<td><input type="text" name="pengirim" size="20" value="<?php echo $pengirim?>"></td>
</tr>
<tr>
<td>&nbsp;</td>
<td>&nbsp;<input type="hidden" name="hidberita" value="<?php echo $id_berita?>">
<input type="submit" name="Edit" value="Edit Berita">&nbsp;<input type="reset" name="reset" value="Cancel">
</td>
</tr>
</table>
</FORM>
</body>
</html>

```

Tampilan browser:



- **Membuat Halaman Hapus Berita**

Nama file: delete_berita.php

Deskripsi: Dokumen PHP untuk menampilkan halaman hapus berita terpilih.

Tampilan sintak:

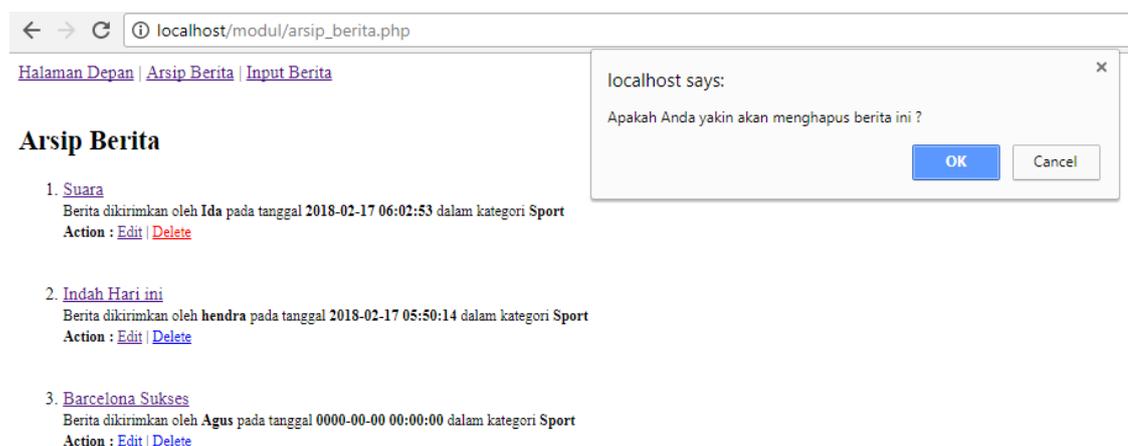
```
<?php
include "koneksi.php";

if (!(base64_decode($_GET['id']))) {
    die ("Error. No Id Selected! ");
} else {
    $id_berita = base64_decode($_GET['id']);
    // base64_decode untuk Membaca data penting yang telah disembunyikan.
    // sebelumnya (di file arsip_berita.php).
}

<html>
<head>
<title>Delete Berita</title>
</head>
<body>
<a href="index.php">Halaman Depan</a> |
<a href="arsip_berita.php">Arsip Berita</a> |
<a href="input_berita.php">Input Berita</a>
<br><br>
<?php
//proses delete berita
if (!empty($id_berita) && $id_berita != "") {

$query = "DELETE FROM berita WHERE id_berita='$id_berita'";
$sql = mysqli_query ($conn,$query);
if ($sql) {
    echo "<h2><font color=blue>Berita telah berhasil dihapus</font></h2>";
    header('Location: arsip_berita.php');
} else {
    echo "<h2><font color=red>Berita gagal dihapus</font></h2>";
}
echo "Klik <a href='arsip_berita.php'>di sini</a> untuk kembali ke halaman arsip berita";
} else {
    die ("Access Denied");
}
}
?>
</body>
</html>
```

Tampilan Browser:



B. Keterampilan Yang Diperlukan Dalam Merancang Script Server Side

1. Menganalisa dan memanfaatkan bahasa *server-side* (php) dengan mengkombinasikan dengan html dengan baik dan benar.
2. Menyusun tahapan langkah – langkah dalam mempersiapkan kebutuhan kerja dan memulai pekerjaan.

C. Sikap Kerja Yang Diperlukan Dalam Merancang Script Server Side

Harus bersikap secara:

1. Cermat dan teliti dalam mengidentifikasi dan mengelompokkan bahan/ perlengkapan pembelajaran
2. Taat asas dalam mengaplikasikan cara, langkah-langkah, panduan, dan pedoman yang dilakukan pada saat penyusunan rencana pembelajaran

BAB IV MEMBUAT DOKUMENTASI WEB

A. Pengetahuan Yang Diperlukan Dalam Membuat Dokumentasi Web

1. Pengenalan XHTML

XHTML merupakan singkatan dari (*Extensible Hypertext Markup Language*), XHTML merupakan gabungan dari HTML dan XML (*Extensible Markup Language*), dari segi peraturan XHTML lebih ketat dibandingkan dengan HTML, dan semua tag XHTML harus di tutup tidak sama dengan tag pada HTML.

a. Struktur Penulisan XHTML

Struktur Penulisan XHTML wajib mengunakan "doctype", contoh:

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

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title></title>
</head>
<body>
.....
</body>
</html>
```

b. Penulisan tag XHTML

tag yang tersusun pada XHTML haruslah selalu tersusun secara benar, contoh:

```
<p><i>contoh element yang berpasngan menggunakan XHTML yang benar</i></p>
<p><i>contoh element yang berpasngan menggunakan XHTML yang salah</p></i>
```

Setiap *tag single* yang terdapat pada XHTML haruslah selalu ditutup dengan tanda "/", contoh:

```
<link href="#" type="text/css" />
<br/>
<hr/>
```

Untuk menuliskan sebuah tag pada XHTML haruslah selalu menggunakan huruf kecil, contoh:

```
<!-- Benar -->
<p>benar</p>
<a href="#">benar</a>

<!-- Salah -->
<P>salah</P>
<A href="#">salah</A>
```

c. Atribut

Nama *atribut* harus digunakan pada XHTML harus menggunakan huruf kecil, contoh:

```
<input type="text" /> <!--benar-->
<input type="password" /> <!--benar-->

<input TYPE="text"> <!-- salah-->
<input TYPE="password"> <!-- salah-->
```

Setiap nilai atau *value* yang ada di dalam atribut harus menggunakan tanda kutip, contoh:

```
<input readonly="readonly">
<input disabled="disabled">
<input checked="checked">
<option selected="selected">
```

Tidak dibolehkan untuk menyingkat atribut yang akan kita gunakan, contoh:

```
<input readonly>
<input disabled>
<input checked>
<option selected>
```

Dan cara penulisan `img` harus menggunakan "alt", contoh:

```

```

2. XHTML dalam Sintak Server-side.

Di bawah ini adalah contoh sebuah file PHP sederhana, dengan mengkombinasikan antara sintak XHTML dengan sintak PHP:

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

<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>contoh xhtml-php</title>
  </head>
  <body>
    <?php
      echo "<h1>Buku Tamu</h1>";
      echo "<hr/>";
      echo "<form action='' method='POST'>";
      echo "<table>";
      echo "<tr>";
      echo "<td>Nama</td>";
      echo "<td></td>";
      echo "<td><input type='text'/></td>";
      echo "</tr>";
      echo "<tr>";
      echo "<td>Email</td>";
      echo "<td></td>";
      echo "<td><input type='text'/></td>";
      echo "</tr>";
      echo "<tr>";
      echo "<td>Komentar</td>";
      echo "<td></td>";
      echo "<td><textarea rows='4' readonly='readonly' style='overflow: hidden'>
        Tulisan ini tidak dapat diubah</textarea></td>";
      echo "</tr>";
      echo "<tr>";
      echo "<td></td>";
      echo "<td></td>";
      echo "<td><input type='submit' value='SUBMIT' /></td>";
      echo "</tr>";
      echo "</table>";
      echo "</form>";
      echo "<hr/>";
    ?>
  </body>
</html>
```

Tampilan Browser

Buku Tamu

Nama :

Email :

Komentar:

B. Keterampilan Yang Diperlukan Dalam Membuat Dokumentasi Web

1. Menganalisa dan memanfaatkan bahasa xhtml yang baik dan benar sesuai dengan aturan yang berlaku.
2. Menyusun tahapan langkah – langkah dalam mempersiapkan kebutuhan kerja dan memulai pekerjaan.

C. Sikap Kerja Yang Diperlukan Dalam Membuat Dokumentasi Web

Harus bersikap secara:

1. Cermat dan teliti dalam menganalisis data;
2. Taat asas dalam mengaplikasikan langkah-langkah, panduan, dan pedoman yang dilakukan dalam menyusun tahapan penyajian;
3. Berpikir analitis serta *evaluatif* waktu melakukan analisis.

BAB V

MELAKUKAN UJI DAN DEBUG SCRIPT

A. Pengetahuan Yang Diperlukan Dalam Melakukan Uji Dan Debug Script

Ketika kita melakukan pengembangan aplikasi perangkat lunak, sering kali kita menemui kesalahan-kesalahan (*errors*). Oleh karena itu kita harus mengurangi kesalahan pada aplikasi dengan mengimplementasikan pemrograman yang baik. Langkah ini tentu harus di dukung dengan menangani kesalahan (*error handling*) yang mungkin terjadi.

Debugging adalah sebuah metode yang dilakukan oleh para pemrogram dan pengembang perangkat lunak untuk mencari dan mengurangi bug, atau kerusakan di dalam sebuah program komputer atau perangkat keras sehingga perangkat tersebut bekerja sesuai dengan harapan.

Berikut beberapa kegunaan dan manfaat yang bisa kita dapatkan:

- Dengan debug dapat menghindari kesalahan penulisan kode seperti kode belum ditutup, fungsi yang salah dan lain-lain.
- Mengetahui fungsi yang tidak terpakai (*deprecated*) atau mengalami perubahan pada sistem terbaru.
- Menyesuaikan *addon* tambahan seperti *plugin* dan *theme* dengan sistem *core* terbaru.
- Debug berguna untuk menyeleraskan CMS yang dipakai (*mis. wordpress*) dengan *environment*-nya (*php, mysql, server dll.*).
- Menjaga dan meningkatkan keamanan sistem.

Berikut ini adalah metode – metode yang biasa digunakan dalam melakukan debugging:

1. Pesan previous Error

Pada saat kita bekerja dengan fungsi – fungsi *database*, dengan mudan kita bisa mendapatkan pesan kesalahan saat kode gagal diesksekusi. Sebagai contoh, kita bisa memanfaatkan fungsionalitas *sqlite_last_error()*, *mysqli_error()*, atau *pg_last_error()*. Akan tetapi, bagaimana jika kode tersebut

tidak terkait dengan interaksi *database*? Secara normal, kita hanya akan mengandalkan bantuan pesan kesalahan dari parser.

Apabila *parser* tidak diperkenankan untuk men-*generate* pesan kesalahan ke *browser*, kita bisa menggunakan variabel yang sudah didefinisikan, yaitu `$php_errormsg`. Variabel ini akan membantu kita untuk mendapatkan pesan kesalahan terakhir yang di-*generate* oleh PHP. Contoh penggunaannya seperti berikut:

```
error_reporting(0);

// aktifkan directive track_errors
if(!ini_get('track_errors')){
    ini_set('track_errors', TRUE);
}

if(!$fp = fopen('xxx', 'r')){
    // menampilkan previous error
    echo "error: " . $php_errormsg;
}
```

Catatan: variabel `$php_errormsg` hanya bisa digunakan jika konfigurasi directive `track_errors` diaktifkan.

2. Logging Error

Ketika *directive display_errors di-disable*, pesan kesalahan tidak akan ditampilkan ke browser. Ini bukan hanya berdampak bagi para pengunjung *web*, tetapi kita selaku pemilik *web* juga terkena dampaknya, dengan asumsi bahwa kita tidak memiliki izin mengubah `php.ini` secara langsung.

Agar kita tetap bisa mendapatkan informasi mengenai pesan kesalahan, kita tentu harus mencari solusi lain. Salah satu cara yang praktis adalah dengan memanfaatkan fungsi `error_log()`. Fungsi ini secara garis besar memiliki kegunaan sama seperti *directive error_log*, akan tetapi bias kita atur sendiri. Asalkan fungsi `error_log()` ini tidak *di-disable* secara permanen oleh admin, kita tetap bisa menerima pesan kesalahan tanpa memerlukan *intervensi* admin *web server*. Contoh penggunaannya seperti berikut:

```
<?php

$conn = @mysqli_connect('unknown','root','');

if (!$conn) {
    // kirim pesan ke browser
    echo "Maaf, koneksi gagal <br/>";
    // tulis pesan error ke file
    writeLog('koneksi gagal', mysqli_connect_error());
}

if (!$fp = @fopen('xxx', 'r')) {
    echo 'Tidak dapat membuka file';
    // tulis pesan error ke file
    writeLog('Gagal membuka file');
}

/*
    fungsi untuk menulis error log ke file.
    @param pesan kesalahan.
    @return fungsi error_log()
*/

function writelog($msg){
    $date = date('d/m/Y H:i:s');
    $file = 'C:/tmp/err.log';
    // tipe log= 3 (menulis ke file lokal)
    return error_log($date.$msg."\n",3,$file);
}

?>
```

Apabila aplikasi kita menggunakan jasa hosting pihak lain, sesuaikan lokasi *path file* ke direktori pribadi kita. Pastikan juga bahwa lokasi tersebut tidak mengizinkan akses file oleh *user* lain.

3. Debugging dengan Assertion

Fitur ini mengizinkan Anda untuk memeriksa apakah suatu ekspresi benar atau tidak. Apabila ekspresi salah, fungsi *assert()* akan mengembalikan aksi yang sudah didefinisikan. Sebaliknya jika ekspresi benar, *output* akan dicetak ke *browser*, seperti normalnya tanpa menggunakan *assertion*.

Seperti halnya ketika kita bekerja dengan *error handler*, disini kita mendapatkan informasi yang terkait dengan kesalahan kode program. Dalam implementasinya, *assert()* sering dikombinasikan dengan fungsi *assert_option()*. Contoh penggunaannya seperti berikut ini:

```
//enable evaluasi assert()
assert_options(ASSERT_ACTIVE,1);
// disable error reporting selama evaluasi
assert_options(ASSERT_WARNING,0);
assert_options(ASSERT_QUIET_EVAL,1);
// untuk memanggil fungsi jika assertion gagal
assert_options(ASSERT_CALLBACK,'assertHandler');

$x = 10;
$y = 0;

// periksa jika assertion FALSE
// melakukan pembagian dengan nol
assert('$z = $x / $y; //comment: $y harus lebih dari nol');

// jika TRUE, ini dieksekusi
echo 'Hasil pembagian = '.$z;

/*
fungsi untuk menangani assertion
jika evaluasi gagal, fungsi dipanggil
men-generate pesan kesalahan
*/
function assertHandler($file,$line,$code){
    echo 'error in:<b>',$file,'</b><br>';
    echo 'line: <b>',$line,'</b><br>';
    echo 'code: <b>',$code,'</b><br>';
    exit;
}
```

Apabila *assertion* diberikan dalam bentuk *string*, ia akan dievaluasi sebagai kode PHP. Keuntungan dari penggunaan *string* ini adalah dapat mengurangi *overhead* ketika pemrosesan dilakukan. Sangat disarankan, kita tidak menggunakan fitur *assertion* ini untuk operasi *runtime* normal. Dengan kata lain, gunakan *assertion* untuk mendukung tahap *debugging* aplikasi.

4. Menangani error dokumen

Secara normal, ketika kita mengakses halaman yang tidak ada, akan muncul pesan *Error 404 Not Found*. Sebagai contoh, jika *client* mengetikkan file yang tidak ada, kita bisa meresponnya, dengan menampilkan informasi atau langsung me-*redirect* ke lokasi yang tersedia. Untuk melakukan hal ini, kita cukup menulis satu baris konfigurasi di file *.htaccess*. Sebelumnya, pastikan bahwa *directive AllowOverride* di file *httpd.conf* bagian dokumen *root* kita bernilai minimal *FileInfo* atau *all*.

Langkah selanjutnya, buat file untuk menangani *error document*, misalnya bernama *404.php*. Setelah itu, ketikkan baris berikut di file *.htaccess*, dan letakkan pada direktori kerja Anda.

lokasi file *404.php* misalnya di

<http://namahost/modul/404.php>

ErrorDocument 404 /modul/404.php

Sampai disini, setiap kesalahan dokumen yang diakibatkan halaman tidak ada, akan direspon oleh kode *404.php*.

Hasilnya akan terlihat seperti dibawah ini:

HTTP Error 404

404 Not Found

The Web server cannot find the file or script you asked for. Please check the URL to ensure that the path is correct.

Please contact the server's administrator if this problem persists.

Perlu diperhatikan, khusus untuk *browser Internet Explorer*, halaman dokumen *handler* tidak akan ditampilkan jika ukurannya cukup kecil. Sebagai solusinya, tambahkan beberapa baris informasi untuk meningkatkan ukuran halaman. Sebagai tambahan, kita juga bisa menangani setiap *error document*, dengan menambahkan kode *error* dan menetapkan file yang akan menanganinya.

Berikut konfigurasinya di file `.htaccess`.

```
# Kode error lain di client
# Bad syntax
#ErrorDocument 400 /400.php
# Unauthorized
#ErrorDocument 401 /401.php
# Not used
#ErrorDocument 402 /402.php
# Forbidden
#ErrorDocument 403 /403.php

# kode error di server
# internal server error
#ErrorDocument 500 /500.php
# Not implemented
#ErrorDocument 501 /501.php
# Overloaded
#ErrorDocument 502 /502.php
# Gateway timeout
#ErrorDocument 503 /503.php
```

B. Keterampilan Yang Diperlukan Dalam Melakukan Uji Dan Debug Script

1. Menganalisa dan memanfaatkan fitur yang disediakan oleh PHP dalam melakukan uji dan *debug script*.
2. Menyusun tahapan langkah – langkah dalam menguji sintak php.

C. Sikap Kerja Yang Diperlukan Dalam Melakukan Uji Dan Debug Script

Harus bersikap secara:

1. Cermat dan teliti dalam menganalisis data;
2. Taat asas dalam mengaplikasikan langkah-langkah, panduan, dan pedoman yang dilakukan dalam menyusun tahapan penyajian;
3. Berpikir analitis serta *evaluatif* waktu melakukan analisis.

BAB VI MELAKUKAN KONFIGURASI KEAMANAN WEB

A. Pengetahuan Yang Diperlukan Dalam Melakukan Konfigurasi Keamanan Web.

Keamanan web atau sekuriti adalah salah satu atribut tambahan untuk mengukur *integritas* perangkat lunak. Jika seorang pengembang web mengabaikan keamanan suatu *website*, maka seorang *hacker* dapat mengambil data – data penting pada suatu *website*. Berikut ini beberapa tindakan keamanan untuk mencegah pencurian data / informasi penting, antara lain:

1. Mencegah Variabel Attack

Sebaiknya kita perlu berhati – hati terhadap variabel – variabel yang kita deklarasikan sendiri. Meskipun terlihat sepele, akan tetapi variabel bisa menjadi *boomerang* apabila tidak ditangani dengan baik dan benar. Sebagai contoh, perhatikan kode program berikut ini:

```
$res = verify($user, $pass);  
if ($res) {  
    $valid = 1;  
}  
  
if ($valid) {  
    // halaman sensitif  
    echo "valid user";  
}
```

Kode di atas berupaya melakukan verifikasi user, akan tetapi terdapat kecerobohan yang dapat berakibat fatal. Dalam kasus ini, nilai kembalian dari fungsi `verify()` akan di-assign ke variabel `$res`. Selanjutnya, jika nilai kembalian adalah *true*, maka *login* sesuai.

Masalah terdapat pada variabel `$valid`, dimana nilai variabel ini sebelumnya tidak diinisialisasi. Akibatnya, jika struktur kode diketahui, *client* bisa mengisi variabel `$valid` dengan nilai 1 atau *true*. Jadi, meskipun `verify()` gagal, *user* tetap bisa masuk ke halaman *sensitive*. Solusi yang digunakan adalah sebelum kode tersebut diproses, terlebih dahulu deklarasikan nilai

default untuk semua variabel yang sensitif dengan nilai 0 atau *false*, seperti contoh berikut ini:

```
$res = 0;
$valid = 0;

$res = verify($user, $pass);
if ($res) {
    $valid = 1;
}

if ($valid) {
    // halaman sensitif
    echo "valid user";
}
```

Pastikan juga bahwa kita selalu bekerja dengan register global dalam keadaan disable / off. Ini dapat membatasi agar nilai variabel harus dikirim melalui superglobal PHP.

2. Mencegah Command Injection

Sejumlah fungsi yang mengizinkan kita untuk mengeksekusi *system call* memang sangat membantu, terutama ketika kita bekerja di lingkungan jaringan secara *remote*. Akan tetapi, ini juga sangat membahayakan saat yang menggunakan adalah penyerang.

Untuk mengurangi resiko penyerangan, gunakan fungsi `escapeshellarg()` dan `escapeshellcmd()`. Fungsi `escapeshellarg()` akan membantu meloloskan argument ke shell dengan menambahkan tanda kutip. Sementara itu, `escapeshellcmd()` digunakan untuk menangani nama program. Contohnya seperti berikut:

```
// tanpa escape
echo "<b>Tanpa escape</b>";
echo "argumen" . $_POST['arg'] . "<p>";
system('dir' . $_POST['arg']);

// escape argumen
echo "<b>Tanpa escape</b>";
$args = escapeshellarg($_POST['arg']);
echo "argumen" . $_POST['arg'] . "<p>";
system('dir' . $_POST['arg']);

// escape command
echo "<b>Tanpa escape</b>";
$cmd = escapeshellcmd($_POST['cmd']);
echo "argumen" . $_POST['cmd'] . "<p>";
system('dir' . $_POST['cmd']);
```

Fungsi – fungsi yang harus diwaspadai dan ditangani dengan baik adalah: `system()`, `exec()`, `passthru()`, `popen()`, `proc_open()`, dan operator backtick(```). Operator *backtick* bukan tanda kutip tunggal. Bagaimanapun juga, fungsi – fungsi dan operator ini bisa sangat membahayakan kita. Oleh karena itu, jika kita tidak benar – benar memerlukan, sebaiknya kita men-disable-nya.

3. Enkripsi data

Dalam upaya untuk lebih meningkatkan proteksi kode program verifikasi, kita bisa menggabungkan *algoritma* hash dan *algoritma enkripsi*. Langkah ini memungkinkan dihasilkannya data terenkripsi (*chiphertext*) yang lebih sulit ditebak. Salah satu *algoritma enkripsi* PHP adalah *algoritma enkripsi* berbasis DES (*Data Encryption Standard*).

Berikut ini contoh kode program yang memanfaatkan algoritma hash dan enkripsi untuk melakukan verifikasi user.

```

if(!empty($_POST['nama'])) && (!empty($_POST['pass'])){
    if(strpos($_POST['nama'], ';')) die ('Incorrect input');

    $db = mysqli_connect('localhost','root','','databaseku')
        or die('Unable to connect...');

    $auth = false;
    $nama = Encrypt(myMagic($_POST['nama']));
    $pass = Encrypt(myMagic($_POST['pass']));

    $sql = "SELECT nama, password FROM uswer
            WHERE nama = ".$nama." AND password = ".$pass."";

    $res = mysqli_query($db,$sql) ;
    if (mysqli_num_rows($res) == 1) {
        $auth = true;
    }

    mysqli_free_result($res);

    if ($auth) {
        echo "Valid...";
    }else{
        exit("invalid account...");
    }

    // enkripsi plain text, return ciphertext
    function Encrypt($str){
        //di-hash kemudian dienkripsi
        $chiper = crypt(md5($str),md5($str));
        return $chiper;
    }
}
    
```

4. Menyembunyikan pesan error

PHP menyediakan *fitur* yang menarik dan praktis untuk menyembunyikan pesan kesalahan pada kode program. Fitur yang diimplementasikan melalui *operator suppression* (@) ini sangat tepat digunakan untuk menyembunyikan pesan kesalahan atau peringatan yang termasuk dalam kategori ringan.

Penggunaan *operator suppression* ini dapat digunakan hanya dengan menambahkan karakter @ di depan setiap ekspresi yang mungkin dapat mengakibatkan kesalahan. Hasilnya, jika ekspresi gagal dievaluasi, pesan kesalahan yang normalnya muncul akan diabaikan. Contoh penggunaannya seperti berikut:

```
<?php
    $file = "../modul/contoh.txt";

    // suppressing error
    $fp = @fopen($file,"r");

    // (perhatikan: kurang tanda kutip)
    $fp = @fopen($file,"r");

?>
```

Keterangan: pada kasus kode program pertama, kegagalan dalam operasi pembukaan file tidak akan menampilkan kesalahan. Adapun untuk kode kedua, Anda tidak perlu heran, ketika pesan kesalahan muncul di *browser*. Ini karena *parser* menemukan kesalahan sintak, dan *operator suppression* tidak menyembunyikan pesan yang disebabkan kesalahan saat *parsing* kode.

5. Mencegah tampilan error reporting

Untuk alasan keamanan, sangat disarankan untuk mencegah tampilan pesan kesalahan di *browser* ketika aplikasi sudah siap di-*deploy*. Langkah ini bisa dilakukan melalui konfigurasi *directive display_errors*, dan memberi nilai *boolean FALSE*. Contohnya seperti berikut:

```
<?php
// set directive display_errors disable
// letakkan di bagian atas kode program
ini_set('display_errors',FALSE);
?>
```

Meskipun konfigurasi *directive display_error* biasa diset saat *runtime* melalui fungsi `ini_set()`, akan lebih tepat jika kita set langsung secara permanen pada file `php.ini`. Hal ini karena untuk kasus kesalahan fatal, pengaturan saat *runtime* bias diabaikan.

```
// disable display_errors di php.ini
display_errors = off
```

sebagai ganti dari tampilan pesan kesalahan, aktifkan nilai *directive log_errors*, dan spesifikasikan lokasi di mana file log akan disimpan (melalui *directive error_log*).

```
log_errors = On
error_log = C:/tmp/php.log
```

6. Konfigurasi PHP

Berikut ini adalah penjelasan beberapa konfigurasi yang terdapat pada `php.ini` agar dapat diketahui fungsi – fungsinya.

Error handling dan logging

- `Error_reporting = E_ALL`

Baik dalam tahap pengembangan maupun *deployment*, sebaiknya kita mengeset semua *level error*.

- `Display_errors = Off`

Sangat disarankan, kita tidak menampilkan pesan *error* yang di-*generate* oleh PHP, kecuali saat tahap pengembangan.

- `Log_errors = On`

Directive ini menyatakan apakah *error* akan ditulis ke *file log* atau tidak. Sebaiknya kita mengaktifkan ini untuk mengganti pesan *error* di *browser*.

- `Error_log = "C:\tmp"`

Sesuaikan lokasi file dimana *log error* akan ditulis, ini memudahkan Anda men-
trace error.

Variabel

- Register_globals = Off

Meskipun dalam tahap pengembangan, sebaiknya kita jangan pernah mengaktifkan *directive* ini. Sebagai ganti untuk pengambilan nilai variabel, gunakan *superglobal*.

- Variabel_order = "GPCS"

Untuk menetapkan urutan variabel – variabel yang deregister PHP, yaitu: \$_GET, \$_POST, \$_COOKIE dan \$_SERVER.

- Register_argc_argv = Off

Jika *directive* ini diaktifkan, parameter – parameter URL beserta nilainya akan mudah ditangkap.

- Magic_quotes_gpc = Off

Untuk menangani data GPC dan sebaiknya tetap di-*disable* (*default*). Sebagai ganti untuk menangani tanda kutip, gunakan fungsi *dedicated database* atau fungsi *addslashes()*.

- Magic_quotes_runtime = Off

Ini berfungsi untuk menangani data yang di-*generate* saat *runtime*. Contohnya data dari SQL dan fungsi *exec()*.

Session dan Cookie

- Session.save_path = "C:\tmp"

Tentukan lokasi yang aman untuk menyimpan data *session*. Apabila platform sistem operasi kita mendukung *shared memory*, isikan nilai *mm*.

- Session.use_cookies = 1

Nilai 1 pada *directive* ini menyatakan bahwa modul akan menggunakan *cookie* untuk menyimpan *id session* di *client*.

- Session.use_only_cookies = 1

Untuk mencegah penyerangan melalui parameter URL ketika bekerja dengan *session*, aktifkan *directive* ini.

- Session.auto_start = 0

Disarankan, kita tidak menjalankan *session* secara otomatis, tetapi menggunakan `session_start()`.

Batasan Akses

- `Disable_functions = exec, system, popen, proc_open, passthru, fsockopen, ftp_connect, dl_open, mail, phpinfo.`

Directive ini digunakan untuk men-*disable* fungsi – fungsi PHP. Contoh fungsi – fungsi di atas merupakan fungsi yang sensitif dan berpotensi digunakan sebagai sarana penyerangan.

- `Disable_classes = namaClass`

Fungsi dari *directive* ini adalah untuk memproteksi *class* kita yang sifatnya sensitif. Cara mengisi nilainya seperti *disable_functions*, tanda koma merupakan pemisah.

- `Enable_dl = Off`

Directive ini mengizinkan ekstensi – ekstensi di-*load* saat *runtime*. Selain terkait masalah sekuriti, ini juga dapat mengurangi performansi.

- `Allow_url_fopen = Off`

Jika kita men-*disable directive* ini, maka *remote* file tidak diizinkan. Artinya, kita tidak bisa mengakses objek URL sebagai file.

- `Extension_dir = "C:\php5-1\ext"`

Tetapkan lokasi file ekstensi (*library*) secara permanen melalui *directive* ini.

- `File_uploads = Off`

Upload file ke *server* seringkali menimbulkan masalah dan disarankan kita tidak mengizinkan *client upload file*. Jika kita memerlukan, kita bisa memanfaatkan FTP *client*.

- `Safe_mode = On`

Safe mode merupakan solusi yang mencoba memecahkan masalah sekuriti *shared-server*. Jika ini diaktifkan, akses ke file – file sensitif akan diperiksa terlebih dahulu.

Performansi

- `Register_long_arrays = Off`

Nilai *Off* pada *directive* ini akan mengakibatkan superglobal lama (`$HTTP_*_VARS`) tidak dikenal.

- `Expose_php = Off`

Directive ini mengizinkan kita menulis kode PHP di file dengan ekstensi lain, seperti `.htm` atau `.html`. ini menjadikan kode bisa dieksekusi lebih cepat oleh server.

- `Session.gc_divisor = 1000`

Directive ini untuk menetapkan nilai pembagi, dimana GC (*Garbage Collection*) akan dijalankan pada tiap *request*.

- `Output_buffering = 4096`

Untuk meningkatkan *performansi*, gunakan *buffering output* dan tetapkan nilai *buffer 4096 byte* (4KB).

B. Keterampilan Yang Diperlukan Dalam Melakukan Konfigurasi Keamanan Web

1. Menganalisa dan memanfaatkan fungsi – fungsi sesuai dengan kebutuhan pembelajaran.
2. Menyusun skrip untuk konfigurasi keamanan *web*.

C. Sikap Kerja Yang Diperlukan Dalam Melakukan Konfigurasi Keamanan Web

Harus bersikap secara:

1. Cermat dan teliti dalam menganalisis data;
2. Taat asas dalam mengaplikasikan langkah-langkah, panduan, dan pedoman yang dilakukan dalam menyusun tahapan penyajian;
3. Berpikir analitis serta evaluatif waktu melakukan analisis.

DAFTAR PUSTAKA

A. Buku Referensi

- a. Prasetyo, Didik Dwi. 2006. *101 Tip & Trik Pemograman PHP*. Jakarta: Gramedia
- b. Hakim, Lukmanul. 2008. *Membongkar Trik rahasia Para Master PHP*. Yogyakarta: Lokomedia

B. Referensi Lainnya

- a. *Widigdo, Anon Kuncoro, 2003 Dasar Pemograman PHP dan MySQL, (www.ilmukomputer.com, diakses tanggal 12 Februari 2018)*

Daftar Alat Dan Bahan

A. Daftar Peralatan/Mesin

No.	Nama Peralatan/Mesin	Keterangan
1.	Laptop, infocus, laserpointer	Untuk di ruang teori
2.	Laptop	Untuk setiap peserta
3.	Koneksi internet	
4.		
5.		
6.		
7.		

B. Daftar Bahan

No.	Nama Bahan	Keterangan
1.	aplikasi XAMPP	Setiap peserta
2.	Text Editor	
3.		
4.		
5.		
6.		
7.		
8.		
9.		Setiap peserta

DAFTAR PENYUSUN

No.	Nama	Profesi
1.	Deny Guslendra	1. Instruktur IT 2. Software Developer



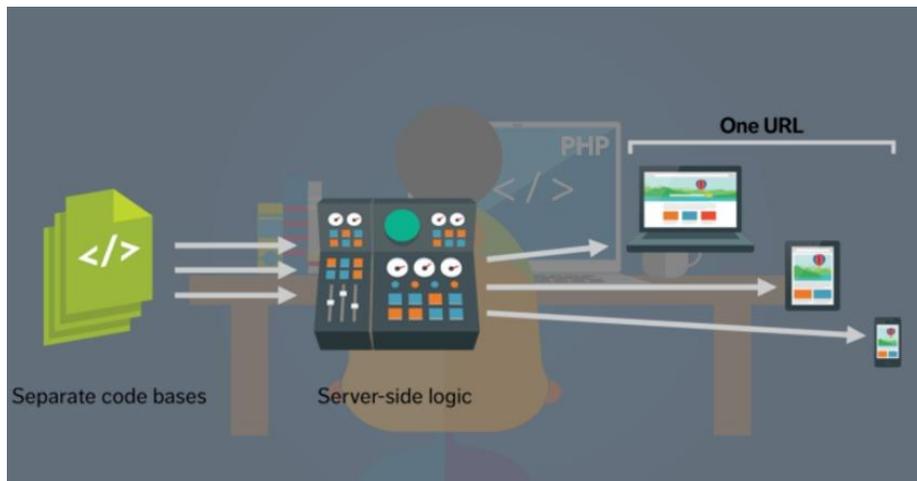
2

LAMPIRAN BUKU KERJA



BUKU KERJA

MEMBUAT WEB DINAMIS MENGGUNAKAN SCRIPT SERVER SIDE ICTWEB415



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN R.I.
DIREKTORAT JENDERAL GURU DAN TENAGA KEPENDIDIKAN
LEMBAGA PENGEMBANGAN DAN PEMBERDAYAAN PENDIDIK DAN TENAGA
KEPENDIDIKAN BIDANG KELAUTAN, PERIKANAN, DAN TEKNOLOGI INFORMASI DAN
KOMUNIKASI
GOWA

PENJELASAN UMUM

Pengembangan Keprofesian Berkelanjutan berbasis kompetensi mengharuskan proses pelatihan memenuhi unit kompetensi secara utuh yang terdiri atas pengetahuan, keterampilan, dan sikap kerja. Dalam buku informasi Membuat *Web Dinamis Menggunakan Script Server Side* telah disampaikan informasi apa saja yang diperlukan sebagai pengetahuan yang harus dimiliki untuk melakukan praktik/keterampilan terhadap unit kompetensi tersebut. Setelah memperoleh pengetahuan dilanjutkan dengan latihan-latihan guna mengaplikasikan pengetahuan yang telah dimiliki tersebut. Untuk itu diperlukan buku kerja Membuat *Web Dinamis Menggunakan Script Server Side* ini sebagai media praktik dan sekaligus mengaplikasikan sikap kerja yang telah ditetapkan karena sikap kerja melekat pada keterampilan. Adapun tujuan dibuatnya buku kerja ini adalah:

1. Prinsip pelatihan berbasis kompetensi dapat dilakukan sesuai dengan konsep yang telah digariskan, yaitu pelatihan ditempuh elemen kompetensi per elemen kompetensi, baik secara teori maupun praktik;
2. Prinsip praktik dapat dilakukan setelah dinyatakan kompeten teorinya dapat dilakukan secara jelas dan tegas;
3. Pengukuran unjuk kerja dapat dilakukan dengan jelas dan pasti.

Ruang lingkup buku kerja ini meliputi pengerjaan tugas-tugas teori dan praktik per elemen kompetensi dan kriteria unjuk kerja berdasarkan VET ICT. Ruang lingkup buku kerja ini meliputi pengerjaan tugas-tugas teori dan praktik per elemen kompetensi dan kriteria unjuk kerja berdasarkan VET ICT

DAFTAR ISI

PENJELASAN UMUM	2
DAFTAR ISI	3
BAB I TUGAS TEORI DAN PRAKTIK	4
A. Menganalisa kebutuhan dokumentasi web dinamis sesuai interaksi server side. .	4
1. Tugas Teori I	4
2. Tugas Praktik I.....	7
B. Merancang Script Server Side	10
1. Tugas Teori II	10
2. Tugas Praktik II.....	12
C. Membuat Dokumentasi Web	15
1. Tugas Teori III.....	15
2. Tugas Praktik III	17
D. Melakukan Uji dan Debug Script	20
1. Tugas Teori IV	20
2. Tugas Praktik IV	22
E. Melakukan Konfigurasi Keamanan Web	25
1. Tugas Teori V.....	25
2. Tugas Praktik V	27
BAB II CEK LIS TUGAS	30

BAB I
TUGAS TEORI DAN PRAKTIK

A. Menganalisa kebutuhan dokumentasi web dinamis sesuai interaksi server side.

1. Tugas Teori I

Perintah : Jawablah soal di bawah ini

Waktu Penyelesaian : 90 menit

Soal :

1. Apakah perbedaan antar *web statis* dan *web dinamis* itu?

Jawaban:

.....
.....
.....

2. Apakah fungsi utama dari *web server*?

Jawaban:

.....
.....
.....

3. Sebutkan aplikasi apasaja yang terinstall pada computer kita saat kita melakukan instalasi aplikasi XAMPP?

Jawaban:

.....
.....
.....

4. Sebutkan dan jelaskan kelebihan dan kekurangan dari beberapa *web server* yang ada?

Jawaban:

.....
.....
.....

5. Sebutkan bahasa pemrograman apa saja yang *script* nya dikompilasi lalu kemudian diterjemahkan di server?

Jawaban:

.....
.....
.....

Lembar Evaluasi Tugas Teori Menganalisa Kebutuhan Dokumentasi Web Dinamis Sesuai Interaksi Server Side.

Semua kesalahan harus diperbaiki terlebih dahulu sebelum ditandatangani.

No.	Benar	Salah
1.		
2.		
3.		
4.		
5.		

Apakah semua pertanyaan Tugas Teori Menganalisa kebutuhan dokumentasi *web dinamis* sesuai *interaksi server side* dijawab dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

2. Tugas Praktik I

- a. Elemen Kompetensi : Menganalisa kebutuhan kebutuhan dokumentasi *web dinamis* sesuai *interaksi server side*
- b. Waktu Penyelesaian : 180 menit
- c. Capaian Unjuk Kerja :
- Setelah menyelesaikan tugas membuat Menganalisa kebutuhan kebutuhan dokumentasi *web dinamis* sesuai interaksi server side peserta mampu:

- 1) Menyiapkan kebutuhan perangkat *web server*.
- 2) Melakukan instalasi kebutuhan pembangunan *web server*.

d. Daftar Alat/Mesin dan Bahan :

NO	NAMA BARANG	SPEKIFIKASI	KETERANGAN
A.	ALAT		
1.	Personal Komputer	standard	
2.			
B.	BAHAN		
1.	software XAMPP		
2.	software text editor		

e. Indikator Unjuk Kerja (IUK):

- 1) Mampu menyiapkan kebutuhan perangkat dan sistem.
- 2) Mampu menentukan bahasa pemrograman yang sesuai dan dibutuhkan.
- 3) Mampu menentukan persyaratan dokumen web.

b Keselamatan dan Kesehatan Kerja

Keselamatan dan kesehatan kerja yang perlu dilakukan pada waktu melakukan praktik kerja ini adalah:

- 1) Bertindak berdasarkan sikap kerja yang sudah ditetapkan sehingga diperoleh hasil seperti yang diharapkan, jangan sampai terjadi kesalahan karena ketidak-telitian dan tidak taat asas.
- 2) Waktu menggunakan komputer, printer, dan alat lainnya mengikuti petunjuknya masing-masing yang sudah ditetapkan.

c Standar Kinerja

- 1) Dikerjakan selesai tepat waktu, waktu yang digunakan tidak lebih dari yang ditetapkan.

d Tugas

Abstraksi Tugas Praktik I

SMK tempat bapak / ibu bekerja akan membutuhkan sebuah *website* pendataan peserta didik baru (PSB) agar dapat diakses dan dilengkapi syarat – syaratnya dimana saja dan kapan saja. Oleh karena kebutuhan tersebut, bapak / ibu berinisiatif membuat *website PSB*. Sesuai dengan penjelasan yang terdapat di Buku Informasi, maka persiapkan seluruh kebutuhan perangkat dan aplikasi pendukung dalam pembangunan *web server* agar dapat memulai pekerjaan dalam pembuatan web.

i. Instruksi Kerja

Setelah membaca abstraksi nomor **h** selanjutnya ikuti instruksi kerja sebagai berikut:

- 1) Mempersiapkan perangkat yang ada telah siap untuk digunakan.
- 2) Tentukan kebutuhan – kebutuhan pendukung yang diperlukan dipersiapkan.
- 3) Pemasangan perangkat lunak terbaru (*uptodate* dari alamat / *web* resmi) dipersiapkan.
- 4) Mempersiapkan check list kebutuhan yang telah di pasang atau belum.

j. Daftar Cek Unjuk Kerja Tugas I

NO	DAFTAR TUGAS/INSTRUKSI	POIN YANG DICEK	PENCAPAIAN		PENILAIAN	
			YA	TIDAK	K	BK
1.	Siapkan referensi kebutuhan	referensi yang dipakai				
2.	Sebutkan daftar perangkat <i>hardware</i> dan <i>software</i> yang dibutuhkan	kebutuhan perangkat				
3.	Mempersiapkan list dan confoh form inputan yang dibutuhkan.	daftar kebutuhan <i>input web</i>				
4.	Mempersiapkan list dan contoh form output dari <i>web PSB</i> .	daftar kebutuhan <i>output web</i>				
5.	<i>download</i> dan <i>instalasi software</i> pendukung yang <i>uptodate</i>	mempersiapkan perangkat pendukung sesuai standard.				

Apakah semua instruksi kerja tugas praktik Menganalisa kebutuhan kebutuhan dokumentasi *web dinamis* sesuai *interaksi server side* dilaksanakan dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

B. Merancang Script Server Side

1. Tugas Teori II

Perintah : Jawablah soal di bawah ini

Waktu Penyelesaian : 90 menit

Soal :

1. Sebutkan kemudahan – kemudahan dengan menggunakan PHP

Jawaban:

.....
.....
.....

2. Apakah fungsi dari variabel dalam bahasa pemograman

Jawaban:

.....
.....
.....

3. Tuliskan sintak PHP untuk menampilkan nama Anda ?

Jawaban:

.....
.....

4. Sebutkan dan jelaskan 3 tipe data dalam PHP ?

Jawaban:

.....
.....
.....

5. Tuliskan sintak PHP menghitung (24 + 2) dengan menggunakan operator aritmatika, yang akan menghasilkan tampilan "Hitungan penjumlahan dari 24 + 2 menghasilkan nilai 26"

Jawaban:

.....
.....
.....

Lembar Evaluasi Tugas Teori Merancang Script Server Side.

Semua kesalahan harus diperbaiki terlebih dahulu sebelum ditandatangani.

No.	Benar	Salah
1.		
2.		
3.		
4.		
5.		

Apakah semua pertanyaan Tugas Teori Menganalisa kebutuhan dokumentasi *web dinamis* sesuai *interaksi server side* dijawab dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

2. Tugas Praktik II

a. Elemen Kompetensi : Merancang *Script Server Side*

b. Waktu Penyelesaian : 180 menit

c. Capaian Unjuk Kerja :

Setelah menyelesaikan tugas Merancang *Script Server Side* peserta mampu:

- 1) Merancang dokumen *web* dan *skrip server side*
- 2) Melakukan *insert, update*, dan *delete* data
- 3) Menggunakan fitur keamanan pada web

d. Daftar Alat/Mesin dan Bahan :

NO	NAMA BARANG	SPEKIFIKASI	KETERANGAN
A.	ALAT		
1.	Personal Komputer	Core i3/i5/i7	
2.			
B.	BAHAN		
1.	software XAMPP		
2.	software text editor		

e. Indikator Unjuk Kerja (IUK):

- 1) Mampu menyiapkan dokumen php.
- 2) Mampu mengkomunikasikan antara php dengan *file external*.
- 3) Mampu mengkomunikasikan antara php dengan database MySQL.

f. Keselamatan dan Kesehatan Kerja

Keselamatan dan kesehatan kerja yang perlu dilakukan pada waktu melakukan praktik kerja ini adalah:

- 1) Bertindak berdasarkan sikap kerja yang sudah ditetapkan sehingga diperoleh hasil seperti yang diharapkan, jangan sampai terjadi kesalahan karena ketidak-telitian dan tidak taat asas.

- 2) Waktu menggunakan komputer, printer, dan alat lainnya mengikuti petunjuknya masing-masing yang sudah ditetapkan.

g. Standar Kinerja

- 1) Dikerjakan selesai tepat waktu, waktu yang digunakan tidak lebih dari yang ditetapkan.
- 2)

h. Tugas

Abstraksi Tugas Praktik II

Pada tugas praktik I bapak / ibu guru telah mempersiapkan kebutuhan *web server*. Terapkanlah setiap kebutuhan untuk perancangan dan pembangunan *web*, persiapkan file dan *databasenya* sesuai dengan kebutuhan *web PSB*, kemudian gunakan komunikasi php, file dan database sebagai transaksional sistem *web PSB* tersebut.

i. Instruksi Kerja

Setelah membaca abstraksi nomor **h** selanjutnya ikuti instruksi kerja sebagai berikut:

- 1) Desain form inputan disesuaikan dengan kebutuhan *web PSB*.
- 2) Desain *database* dengan tabel – tabel yang sesuai.
- 3) Letakkan dokumen php pada lokasi *web server*.
- 4) Mengkoneksikan dokumen php dengan *database*.
- 5) Jalankan bisnis proses *web PSB*.

k. Daftar Cek Unjuk Kerja Tugas I

NO	DAFTAR TUGAS/INSTRUKSI	POIN YANG DICEK	PENCAPAIAN		PENILAIAN	
			YA	TIDAK	K	BK
1.	Merancang form inputan	form kebutuhan disesuaikan				
2.	Merancang <i>database</i> dan tabel	tabel <i>database</i> disesuaikan				
3.	Letakkan dokumen php pada <i>web server</i>	uji coba tampilan di browser (localhost)				
4.	Mengkoneksikan dokumen php dengan <i>database</i>	pesan sukses terkoneksi <i>database</i>				
5.	jalankan bisnis proses PSB	cek CRUT				

Apakah semua instruksi kerja tugas praktik Merancang skrip *server side* dilaksanakan dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

C. Membuat Dokumentasi Web

1. Tugas Teori III

Perintah : Jawablah soal di bawah ini

Waktu Penyelesaian : 90 menit

Soal :

1. Apakah xhtml itu

Jawaban:

.....
.....
.....

2. Sebutkan perbedaan antara html dan xhtml

Jawaban:

.....
.....
.....

3. Sebutkan kelebihan penggunaan xhtml

Jawaban:

.....
.....

4. Tuliskan sintak xhtml untuk form login

Jawaban:

.....
.....
.....

5. Tuliskan sintak php pada *form login* dengan format xhtml

Jawaban:

.....
.....
.....

Lembar Evaluasi Tugas Teori Membuat Dokumentasi Web

Semua kesalahan harus diperbaiki terlebih dahulu sebelum ditandatangani.

No.	Benar	Salah
1.		
2.		
3.		
4.		
5.		

Apakah semua pertanyaan Tugas Teori membuat dokumentasi web dijawab dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

2. Tugas Praktik III

- a. Elemen Kompetensi : Membuat dokumentasi web
- b. Waktu Penyelesaian : 180 menit
- c. Capaian Unjuk Kerja :

Setelah menyelesaikan tugas membuat dokumentasi web peserta mampu:

- 1) Merancang dokumen web dengan skrip XHTML
- 2) Merancang dokumen web kombinasi skrip XHTML dengan PHP

- j. Daftar Alat/Mesin dan Bahan :

NO	NAMA BARANG	SPEKIFIKASI	KETERANGAN
A.	ALAT		
1.	Personal Komputer	Core i3/ i5/ i7	
2.			
B.	BAHAN		
1.	software XAMPP		
2.	software text editor		

- k. Indikator Unjuk Kerja (IUK):

- 1) Mampu merancang dokumen xhtml.
- 2) Mampu merancang 2 skrip (php & xhtml) dalam 1 dokumen web.

- l. Keselamatan dan Kesehatan Kerja

Keselamatan dan kesehatan kerja yang perlu dilakukan pada waktu melakukan praktik kerja ini adalah:

- 1) Bertindak berdasarkan sikap kerja yang sudah ditetapkan sehingga diperoleh hasil seperti yang diharapkan, jangan sampai terjadi kesalahan karena ketidak-telitian dan tidak taat asas.
- 2) Waktu menggunakan komputer, printer, dan alat lainnya mengikuti petunjuknya masing-masing yang sudah ditetapkan.

m. Standar Kinerja

- 1) Dikerjakan selesai tepat waktu, waktu yang digunakan tidak lebih dari yang ditetapkan.

n. Tugas

Abstraksi Tugas Praktik III

Merancang halaman daftar guru dengan fasilitas View, Tambah, Edit dan Hapus data guru terpilih.

o. Instruksi Kerja

Setelah membaca abstraksi nomor **h** selanjutnya ikuti instruksi kerja sebagai berikut:

- 1) Menampilkan data dari database.
- 2) Menggunakan latar belakang foto sekolah.
- 3) Menampilkan foto guru terpilih.
- 4) Menggunakan struktur skrip xhtml.

I. Daftar Cek Unjuk Kerja Tugas III

NO	DAFTAR TUGAS/INSTRUKSI	POIN YANG DICEK	PENCAPAIAN		PENILAIAN	
			YA	TIDAK	K	BK
1.	Menampilkan data dari <i>database</i>	koneksi <i>database</i>				
2.	Menggunakan latar belakang foto sekolah	sintak xhtml				
3.	Menampilkan foto guru terpilih	sintak xhtml dan php				
4.	Menggunakan stuktur XHTML	sintak xhtml yang baik dan benar				

Apakah semua instruksi kerja tugas praktik membuat dokumentasi *web* dilaksanakan dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

D. Melakukan Uji dan Debug Script

1. Tugas Teori IV

Perintah : Jawablah soal di bawah ini

Waktu Penyelesaian : 90 menit

Soal :

1. Apakah debugging itu?

Jawaban:

.....
.....
.....

2. Apakah manfaat dilakukannya *debugging*

Jawaban:

.....
.....
.....

3. Apakah fungsi dari *assertion*

Jawaban:

.....
.....

4. Apakah fungsi *error_log()*

Jawaban:

.....
.....
.....

5. Bagaimanakah cara menyembunyikan pesan kesalahan?

Jawaban:

.....
.....
.....

Lembar Evaluasi Tugas Teori Melakukan Uji dan Debug Script

Semua kesalahan harus diperbaiki terlebih dahulu sebelum ditandatangani.

No.	Benar	Salah
1.		
2.		
3.		
4.		
5.		

Apakah semua pertanyaan Tugas Teori membuat dokumentasi web dijawab dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

2. Tugas Praktik IV

d. Elemen Kompetensi : Melakukan Uji dan *Debug Script*

e. Waktu Penyelesaian : 180 menit

f. Capaian Unjuk Kerja :

Setelah menyelesaikan tugas Melakukan Uji dan *Debug Script* peserta mampu:

- 1) Melakukan pengujian web untuk menanggulangi kesalahan – kesalahan.
- 2) Mampu merekam seluruh pesan kesalahan secara tersembunyi dan aman.

p. Daftar Alat/Mesin dan Bahan :

NO	NAMA BARANG	SPESIFIKASI	KETERANGAN
A.	ALAT		
1.	Personal Komputer	Core i3/ i5/ i7	
2.			
B.	BAHAN		
1.	software XAMPP		
2.	software text editor		

q. Indikator Unjuk Kerja (IUK):

- 1) Mampu menguji dokumen web
- 2) Mampu mendokumentasikan pesan kesalahan dengan baik dan aman.

r. Keselamatan dan Kesehatan Kerja

Keselamatan dan kesehatan kerja yang perlu dilakukan pada waktu melakukan praktik kerja ini adalah:

- 1) Bertindak berdasarkan sikap kerja yang sudah ditetapkan sehingga diperoleh hasil seperti yang diharapkan, jangan sampai terjadi kesalahan karena ketidak-telitian dan tidak taat asas.
- 2) Waktu menggunakan komputer, printer, dan alat lainnya mengikuti petunjuknya masing-masing yang sudah ditetapkan.

s. Standar Kinerja

- 1) Dikerjakan selesai tepat waktu, waktu yang digunakan tidak lebih dari yang ditetapkan.

t. Tugas

Abstraksi Tugas Praktik IV

Melakukan pengujian debugging pada tugas praktik III dengan mengaplikasikan beberapa metode yang ada pada modul informasi agar dapat meminimalisir kesalahan.

u. Instruksi Kerja

Setelah membaca abstraksi nomor **h** selanjutnya ikuti instruksi kerja sebagai berikut:

- 1) Membuat halaman status web "dalam pengembangan (*under construction*)", sebagai *redirect* jika halaman yang diakses tidak ada.
- 2) Menyembunyikan pesan kesalahan dan mengaktifkan *log_errors*.

m. Daftar Cek Unjuk Kerja Tugas III

NO	DAFTAR TUGAS/INSTRUKSI	POIN YANG DICEK	PENCAPAIAN		PENILAIAN	
			YA	TIDAK	K	BK
1.	Membuat <i>error page</i>	<i>Konfigurasi .htaccess</i> atau <i>page error</i> <i>ready.</i>				
2.	Menyembunyikan pesan <i>error</i> dan menyimpan <i>log error</i> nya	mengaktifkan <i>log</i> <i>error</i> dan membacanya di lokasi <i>file log</i> nya.				

Apakah semua instruksi kerja tugas praktik Melakukan Uji dan *Debug Script* dilaksanakan dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

E. Melakukan Konfigurasi Keamanan Web

1. Tugas Teori V

Perintah : Jawablah soal di bawah ini

Waktu Penyelesaian : 90 menit

Soal :

1. Apakah tujuan kita memperhatikan masalah keamanan web.

Jawaban:

.....
.....
.....

2. Bagaimanakah cara agar mencegah dari serangan *variabel attack*

Jawaban:

.....
.....
.....

3. Apakah tujuan dari penggunaan fungsi *enkripsi* data

Jawaban:

.....
.....

4. Sebutkan jenis – jenis *enkripsi* di PHP

Jawaban:

.....
.....
.....

5. Jelaskan masing – masing fungsi yang harus diwaspadai dan ditangani dengan baik yaitu *system()*, *exec()*, *passthru()*, *popen()*, *proc_open()*, dan *operator backtick()*.

Jawaban:

.....
.....
.....

Lembar Evaluasi Tugas Teori Melakukan Konfigurasi Keamanan Web

Semua kesalahan harus diperbaiki terlebih dahulu sebelum ditandatangani.

No.	Benar	Salah
1.		
2.		
3.		
4.		
5.		

Apakah semua pertanyaan Tugas Teori membuat dokumentasi web dijawab dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:

2. Tugas Praktik V

a. Elemen Kompetensi : Melakukan *Konfigurasi* Keamanan Web

b. Waktu Penyelesaian : 180 menit

c. Capaian Unjuk Kerja :

Setelah menyelesaikan tugas Melakukan konfigurasi keamanan web peserta mampu:

1) Mencegah pesan kesalahan tampil pada *website*.

2) *Mengkonfigurasi web server* untuk proteksi terhadap serangan *database*.

d. Daftar Alat/Mesin dan Bahan :

NO	NAMA BARANG	SPESIFIKASI	KETERANGAN
A.	ALAT		
1.	Personal Komputer	Core i3/ i5/ i7	
2.			
B.	BAHAN		
1.	software XAMPP		
2.	software text editor		

e. Indikator Unjuk Kerja (IUK):

1) Mampu mencegah pesan kesalahan

2) Mampu mengkonfigurasi *web server* untuk keamanan.

f. Keselamatan dan Kesehatan Kerja

Keselamatan dan kesehatan kerja yang perlu dilakukan pada waktu melakukan praktik kerja ini adalah:

1) Bertindak berdasarkan sikap kerja yang sudah ditetapkan sehingga diperoleh hasil seperti yang diharapkan, jangan sampai terjadi kesalahan karena ketidak-telitian dan tidak taat asas.

2) Waktu menggunakan komputer, printer, dan alat lainnya mengikuti petunjuknya masing-masing yang sudah ditetapkan.

g. Standar Kinerja

- 1) Dikerjakan selesai tepat waktu, waktu yang digunakan tidak lebih dari yang ditetapkan.

h. Tugas

Abstraksi Tugas Praktik V

Membuat *form login* dengan memanfaatkan *fitur enkripsi* data yang terhubung dengan database. Hak akses web terdiri atas *admin* dan *member*.

i. Instruksi Kerja

Setelah membaca *abstraksi* nomor **h** selanjutnya ikuti instruksi kerja sebagai berikut:

- 1) Membuat form login dengan menerapkan proses *enkripsi* data.
- 2) Menyembunyikan pesan kesalahan dan mengaktifkan *log_errors*.
- 3) Mengkonfigurasi PHP sebelum dan sesudah tugas praktik dibuat.

j. Daftar Cek Unjuk Kerja Tugas V

NO	DAFTAR TUGAS/INSTRUKSI	POIN YANG DICEK	PENCAPAIAN		PENILAIAN	
			YA	TIDAK	K	BK
1.	Membuat <i>form login</i> dengan menerapkan <i>enkripsi</i> data	kode <i>enkripsi</i> digunakan				
2.	Menyembunyikan pesan error dan menyimpan log error nya	mengaktifkan log error dan membacanya di lokasi <i>file log</i> nya.				
3.	Konfigurasi PHP sebelum dan sesudah tugas dilaksanakan.	perubahan konfigurasi sebelum dan sesudah tugas dilaksanakan				
4.						
5.						
6.						
7.						
8.						
9.						
10.						

Apakah semua instruksi kerja tugas praktik Melaksanakan konfigurasi keamanan web dilaksanakan dengan benar dengan waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai

BAB II CEK LIS TUGAS

NO	TUGAS UNJUK KERJA	PENILAIAN		TANGGAL
		K	BK	
1.	Menganalisa kebutuhan kebutuhan dokumentasi web dinamis sesuai interaksi <i>server side</i> .			
2.	Merancang <i>Script Server Side</i>			
3.	Membuat Dokumentasi Web			
4.	Melakukan Uji dan <i>Debug Script</i>			
5.	Melakukan Konfigurasi Keamanan Web			

Apakah semua tugas unjuk kerja Membuat web dinamis menggunakan *script server-side* telah dilaksanakan dengan benar dan dalam waktu yang telah ditentukan?

YA

TIDAK

	NAMA	TANDA TANGAN
PESERTA
PENILAI

Catatan Penilai:



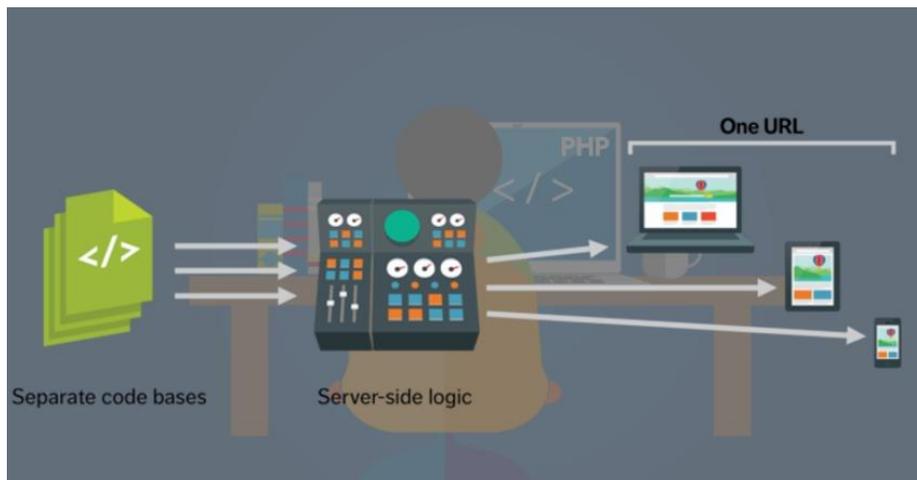
3

**LAMPIRAN
BUKU PENILAIAN**



BUKU PENILAIAN

MEMBUAT WEB DINAMIS MENGGUNAKAN SCRIPT SERVER SIDE ICTWEB415



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN R.I.
DIREKTORAT JENDERAL GURU DAN TENAGA KEPENDIDIKAN
LEMBAGA PENGEMBANGAN DAN PEMBERDAYAAN PENDIDIK DAN TENAGA
KEPENDIDIKAN BIDANG KELAUTAN, PERIKANAN, DAN TEKNOLOGI INFORMASI DAN
KOMUNIKASI
GOWA

PENJELASAN UMUM

Buku penilaian untuk unit kompetensi Membuat *web dinamis* menggunakan *script server side* dibuat sebagai konsekuensi logis dalam pelatihan berbasis kompetensi yang telah menempuh tahapan penerimaan pengetahuan, keterampilan, dan sikap kerja melalui buku informasi dan buku kerja. Setelah latihan-latihan (*exercise*) dilakukan berdasarkan buku kerja maka untuk mengetahui sejauh mana kompetensi yang dimilikinya perlu dilakukan uji komprehensif secara utuh per unit kompetensi dan materi uji komprehensif itu ada dalam buku penilaian ini.

Adapun tujuan dibuatnya buku penilaian ini, yaitu untuk menguji kompetensi peserta pelatihan setelah selesai menempuh buku informasi dan buku kerja secara komprehensif dan berdasarkan hasil uji inilah peserta akan dinyatakan kompeten atau belum kompeten terhadap unit kompetensi Membuat web dinamis menggunakan *script server side*. Metoda Penilaian yang dilakukan meliputi penilaian dengan opsi sebagai berikut:

1. Metoda Penilaian Pengetahuan

- a. Tes Tertulis

Untuk menilai pengetahuan yang telah disampaikan selama proses pelatihan terlebih dahulu dilakukan tes tertulis melalui pemberian materi tes dalam bentuk tertulis yang dijawab secara tertulis juga. Untuk menilai pengetahuan dalam proses pelatihan materi tes disampaikan lebih dominan dalam bentuk *obyektif tes*, dalam hal ini jawaban singkat, menjodohkan, benar-salah, dan pilihan ganda. *Tes essay* bisa diberikan selama tes *essay* tersebut *tes essay* tertutup, tidak *tes essay* terbuka, hal ini dimaksudkan untuk mengurangi faktor subyektif penilai.

- b. Tes Wawancara

Tes wawancara dilakukan untuk menggali atau memastikan hasil tes tertulis sejauh itu diperlukan. Tes wawancara ini dilakukan secara perseorangan antara penilai dengan peserta uji/peserta pelatihan. Penilai sebaiknya lebih dari satu orang.

2. Metoda Penilaian Keterampilan

a. Tes Simulasi

Tes simulasi ini digunakan untuk menilai keterampilan dengan menggunakan media bukan yang sebenarnya, misalnya menggunakan tempat kerja tiruan (bukan tempat kerja yang sebenarnya), obyek pekerjaan disediakan atau hasil rekayasa sendiri, bukan obyek kerja yang sebenarnya.

b. Aktivitas Praktik

Penilaian dilakukan secara sebenarnya, di tempat kerja sebenarnya dengan menggunakan obyek kerja sebenarnya.

3. Metoda Penilaian Sikap Kerja

a. Observasi

Untuk melakukan penilaian sikap kerja digunakan metoda *observasi* terstruktur, artinya pengamatan yang dilakukan menggunakan lembar penilaian yang sudah disiapkan sehingga pengamatan yang dilakukan mengikuti petunjuk penilaian yang dituntut oleh lembar penilaian tersebut. Pengamatan dilakukan pada waktu peserta uji/peserta pelatihan melakukan keterampilan kompetensi yang dinilai karena sikap kerja melekat pada keterampilan tersebut.

DAFTAR ISI

PENJELASAN UMUM	2
DAFTAR ISI	4
BAB I PENILAIAN TEORI	5
A. Lembar Penilaian Teori	5
B. Ceklis Penilaian Teori.....	10
BAB II PENILAIAN PRAKTIK	11
A. Lembar Penilaian Praktik	11
B. Ceklis Aktivitas Praktik	13
BAB III CEKLIS PENILAIAN SIKAP KERJA	15
A. Penilaian Sikap Kerja	15
LAMPIRAN	16
Lampiran 1. Kunci Jawaban	16

BAB I PENILAIAN TEORI

A. Lembar Penilaian Teori

Unit Kompetensi : Membuat web dinamis menggunakan *Script Server Side*

Diklat :

Waktu : 60 menit

PETUNJUK UMUM

1. Jawablah materi tes ini pada lembar jawaban/kertas yang sudah disediakan.
2. Modul terkait dengan unit kompetensi agar disimpan.
3. Bacalah materi tes secara cermat dan teliti.

Isian

Lengkapilah kalimat di bawah ini dengan cara mencari jawabannya pada kolom sebelah kanan dan tuliskan jawabannya saja pada kertas yang tersedia.

- | | |
|---|--|
| <ol style="list-style-type: none">1. <i>Web server</i> keluaran <i>windows</i> adalah ...2. ... adalah <i>text editor alternatif</i> yang memiliki fungsi yang sama yaitu sebagai tempat penulisan sintak dalam pembuatan web.3. FTP yang merupakan protokol <i>internet</i> yang dipakai untuk melakukan proses <i>upload</i> dan <i>download</i> berkas/ file, akan berfungsi jika kita memiliki ...4. <i>Wordpress, Drupal, dan Joomla</i> merupakan ...5. ... adalah teks yang merupakan gabungan huruf, angka, <i>whitespace (spasi)</i>, dan berbagai karakter.6. PHP memiliki fungsi dalam pembacaan data / file secara per baris dengan menggunakan fungsi ... | <ol style="list-style-type: none">1. cms2. hosting3. fgets4. integer5. <?php6. IIS7. String8. \$9. Delete10. Brackets11. @12. <?13. unlink |
|---|--|

7. Fungsi mengapus file menggunakan fungsi ...
8. Salah satu cara untuk menyembunyikan pesan error adalah dengan tanda ...

Benar-Salah

Nyatakan pernyataan di bawah ini benar atau salah dengan cara menulis huruf B jika Benar dan huruf S jika Salah.

B	S	1. Pengiriman data dari <i>server</i> ke <i>browser</i> disebut sebagai HTTP <i>Request</i> .
B	S	2. Apache adalah <i>web server</i> yang gratis.
B	S	3. Jenis mode akses file dengan "a" berarti membuka file untuk proses penambahan data.
B	S	4. Ketika kita selesai menulis file, kita dapat menutup koneksi dengan file menggunakan fungsi <i>fclose()</i> ;
B	S	5. <i>Attribute form</i> yang digunakan untuk menuju pada halaman penerima <i>variabel</i> adalah <i>url=""</i> .
B	S	6. Perintah php untuk mengeksekusi perintah <i>MySQL</i> adalah <i>mysqli_query</i>
B	S	7. Fungsi <i>stripslashes</i> adalah untuk dapat menyembunyikan kata meskipun terdapat karakter khusus.
B	S	8. Mencegah tampilan pesan kesalahan di browser dapat dilakukan dengan konfigurasi <i>directive display_hidden = FALSE</i> ;

Pilihan Ganda

Jawablah pertanyaan/pernyataan di bawah ini dengan cara memilih pilihan jawaban yang tepat dan menuliskan huruf A/B/C/D yang sesuai dengan pilihan tersebut.

1. Yang bukan termasuk dalam web server adalah
 - A. Apache
 - B. IIS
 - C. XAMPP
 - D. Linux
2. Perangkat lunak yang digunakan untuk menerima dan menyajikan sumber informasi di *internet* adalah...
 - A. FTP
 - B. Browser
 - C. Download
 - D. web-server

3. Penyewaan tempat untuk menampung data-data yang diperlukan oleh sebuah *website*, sehingga dapat diakses melalui internet adalah...
- A. FTP
B. Browser
C. Domain
D. Hosting
4. Penamaan unik yang diberikan untuk mengidentifikasi alamat (*IP Address*) *server* komputer seperti *web server* atau *email server* di internet adalah...
- A. FTP
B. Browser
C. Domain
D. Hosting
5. Tipe data pada PHP yang berisi hanya bilangan bulat tanpa koma adalah...
- A. Integer
B. Number
C. Complete
D. String
6. Membaca file dengan cara per-baris menggunakan fungsi...
- A. fgets()
B. fopen()
C. fread()
D. feof()
7. Metode pengiriman data pada form yang akan menampilkan data/ nilai pada URL, kemudian akan ditampung oleh action adalah...
- A. HREF
B. POST
C. URL
D. GET
8. *Elemen form* yang mampu menampung inputan lebih dari satu baris adalah...
- A. <input type="checkbox">
B. <input type="radio">
C. <textarea>
D. <input type="text">
9. *Atribut action* digunakan untuk...
- A. menentukan nama form
B. menentukan metode pengiriman yang dipakai
C. menentukan alamat halaman web yang akan memproses masukan
D. menerima masukan berupa pilihan
10. Perintah yang digunakan untuk membuat garis lurus pada pembuatan situs HTML
- A. HR
B. LI
C. BR
D. OL
11. Penggunaan fungsi *isset()* adalah untuk...
- A. menyatakan variabel sudah diset atau tidak.
B. menyatakan data telah tersimpan.

20. Berikut ini adalah *fasilitas enkripsi* yang disediakan oleh php, kecuali...

- A. md5
- B. base64_encode
- C. keycode
- D. hash

Essay

Jawablah pertanyaan-pertanyaan di bawah ini dengan jelas dan benar!

1. Sebutkan pengertian PHP?
2. PHP juga mampu *lintas platform*, apa artinya?
3. Penggunaan bahasa PHP untuk apa saja?
4. Sebutkan salah satu keunggulan yang dimiliki oleh PHP?
5. Sebutkan beberapa kelemahan PHP?
6. PHP mempunyai konektivitas yang baik dengan beberapa kasus basis data, sebutkan 5 diantaranya?
7. Script PHP termasuk dalam HTML-*embedded*, apa artinya?
8. Operator dapat dikelompokkan dalam 4 kategori, sebut dan jelaskan
9. PHP mendukung 8 buah tipe data primitif, apa saja?
10. Sebutkan jenis operator aritmatika?

B. Ceklis Penilaian Teori

NO. KUK	NO. SOAL	KUNCI JAWABAN	JAWABAN PESERTA	PENILAIAN		KETERANGAN
				K	BK	
	Isian					
	A.1	IIS				
	A.2	Brackets				
	A.3	hosting				
	A.4	cms				
	A.5	string				
	A.6	fgets				
	A.7	unlink				
	A.8	@				
	B-S					
	B.1	S				
	B.2	B				
	B.3	B				
	B.4	B				
	B.5	S				
	B.6	B				
	B.7	S				
	B.8	B				
	PG					
	C.1	D				
	C.2	B				
	C.3	D				
	C.4	C				
	C.5	A				
	C.6	A				
	C.7	D				
	C.8	C				
	C.9	C				
	C.10	A				
	C.11	A				
	C.12	B				
	C.13	A				
	C.14	C				
	C.15	B				
	C.16	A				
	C.17	C				
	C.18	A				
	C.19	D				
	C.20	C				
	Essay					
	D.1	Terlampir				
	D.2	Terlampir				
	D.3	Terlampir				
	D.4	Terlampir				
	D.5	Terlampir				
	D.6	Terlampir				
	D.7	Terlampir				
	D.8	Terlampir				
	D.9	Terlampir				
	D.10	Terlampir				

BAB II PENILAIAN PRAKTIK

A. Lembar Penilaian Praktik

Tugas Unjuk Kerja Membuat web dinamis menggunakan *script server side*

1. Waktu : 120 menit
2. Alat : PC, laptop, printer
3. Bahan : *Software web-server, database, text editor*
4. Indikator Unjuk Kerja
 - a. Mampu menyiapkan dan memasang alat dan bahan.
 - b. Mampu merancang web dinamis dengan memakai sintak XHTML dan PHP yang terhubung dengan basis data MySQL.
 - c. Mampu menguji / *debugging web*.
 - d. Mampu memanfaatkan konfigurasi keamanan web.
5. Standar Kinerja
 - a. Selesai dikerjakan tidak melebihi waktu yang telah ditetapkan.
 - b. Toleransi kesalahan 5% (*lima persen*), tetapi tidak pada aspek kritis.
6. Instruksi Kerja

Abstraksi tugas:

Buatlah aplikasi penilaian siswa berbasis web dengan menggunakan bahasa pemrograman *xhtml dan php*.

Untuk menyelesaikan tugas ini, ikuti instruksi selanjutnya di bawah ini.

- a. Aplikasi *terkoneksi* dengan *database MySQL*.
- b. Halaman yang terdapat pada aplikasi, antara lain:
 - View daftar nilai siswa.
 - Tambah nilai siswa.
 - Edit nilai siswa.
 - Hapus nilai siswa.

c. Struktur database terdiri atas tabel berikut ini:

- Tabel Kelas

Name	Type	Length	Decimals	Not null	
id_kelas	int	11	0	<input checked="" type="checkbox"/>	 1
nama	varchar	255	0	<input type="checkbox"/>	

- Tabel Mapel

Name	Type	Length	Decimals	Not null	
id_mapel	int	11	0	<input checked="" type="checkbox"/>	 1
nama	varchar	40	0	<input type="checkbox"/>	

- Tabel Siswa

Name	Type	Length	Decimals	Not null	
id_siswa	int	11	0	<input checked="" type="checkbox"/>	 1
nis	varchar	30	0	<input type="checkbox"/>	
nama	varchar	30	0	<input type="checkbox"/>	
jns_kelamin	enum	0	0	<input type="checkbox"/>	
id_kelas	varchar	11	0	<input type="checkbox"/>	

- Tabel Nilai

Name	Type	Length	Decimals	Not null	
id_nilai	int	11	0	<input checked="" type="checkbox"/>	 1
id_siswa	int	11	0	<input type="checkbox"/>	
id_mapel	int	11	0	<input type="checkbox"/>	
nilai_harian	int	11	0	<input type="checkbox"/>	
nilai_tugas	int	11	0	<input type="checkbox"/>	
nilai_ujian	int	11	0	<input type="checkbox"/>	
nilai_final	int	11	0	<input type="checkbox"/>	

- d. Field nilai_final yang terdapat pada tabel nilai didapat dari rata-rata nilai_harian, nilai_tugas dan nilai_ujian.
- e. Terapkan fungsi keamanan data.
- f. Pesan kesalahan ditampilkan jika terdapat inputan yang belum terisi.
- g. Pesan proses berhasil jika inputan berhasil di simpan / *update* / *delete*.
- h. Lakukan uji *debugging web*.

B. Ceklis Aktivitas Praktik

Kode Unit Kompetensi : ICTWEB415

Judul Unit Kompetensi : Membuat Web Dinamis Menggunakan Script Server Side.

Nama Peserta/Asesi :

INDIKATOR UNJUK KERJA	TUGAS	HAL-HAL YANG DIAMATI	PENILAIAN	
			K	BK
1. Mampu menentukan fungsionalitas dinamis dokumen web yang diperlukan.	1.1 Mempersiapkan dan memastikan perangkat pendukung hardware dan software yang dibutuhkan telah siap 1.2 Melakukan instalasi perangkat.	<ul style="list-style-type: none"> Cek kebutuhan <i>hardware requirement</i>. Cek <i>uptodate software</i>. <i>Download</i> jika <i>software</i> belum tersedia. 		
2. Mampu menentukan bahasa yang sesuai, untuk mencapai fungsionalitas tersebut.	2.1 Menjalankan service program agar dapat berjalan normal.	<ul style="list-style-type: none"> Membuka dan menjalankan service sesuai dengan kebutuhan. 		
3. Mampu menentukan persyaratan dokumen web.	3.1 Mempersiapkan dokumen web dan testing pada jaringan lokal atau internet	<ul style="list-style-type: none"> Uji testing dokumen web sederhana. 		
4. Mampu merancang dokumen web dan skrip server side, untuk diinteraksikan dengan sumber data external.	4.1 Mempersiapkan file sumber. 4.2 Mengkoneksikan skrip dengan file sumber. 4.3 Menulis rekaman log pada file sumber.	<ul style="list-style-type: none"> Perintah php dalam <i>connect</i> dan penulisan data. 		
5. Mampu merancang dokumen web dan skrip server side, agar administrator dapat melakukan insert, update dan delete data pada sumber data external.	5.1 Merancang dokumen web dengan menambahkan fasilitas view, insert, update dan delete pada web.	<ul style="list-style-type: none"> Perintah php yang digunakan (tipe data, variabel dan operator) Perintah koneksi pada database dan perintah <i>query</i> nya. 		
6. Mampu mengimplementasikan Fitur keamanan didalam dokumen web	6.1 Memilih metode dan fitur keamanan data yang akan dikirimkan.	<ul style="list-style-type: none"> Metode yang digunakan untuk mengamankan data. 		
7. Mampu Menulis Bahasa Extensible Hypertext Markup Language (XHTML)	7.1 Merancang setiap elemen dengan menggunakan standard xhtml.	<ul style="list-style-type: none"> Skrip yang sesuai dengan aturan XHTML. 		

yang dapat diperluas berdasarkan aksesibilitas.				
8. Mampu menulis skrip server side dengan standard xhtml	8.1 Mengkolaborasikan antara skrip php dengan <i>standard xhtml</i> .	<ul style="list-style-type: none"> • Penulisan sintak yang baik dan benar sesuai dengan aturan standard. 		
9. Mampu menguji dokumen web agar fungsionalitas tidak menjadi keharusan dan mengulanginya sampai benar.	9.1 Melakukan proses uji dan debugging pada halaman web dan terhadap <i>database</i> .	<ul style="list-style-type: none"> • Sintak uji dan debugging pada web dan <i>database</i>. 		
10. Mampu melengkapi dan mengirimkan dokumentasi ke personel yang tepat untuk mendapatkan persetujuan	10.1 Menggunakan fitur php dengan menjalankan fitur penulisan log.	<ul style="list-style-type: none"> • Sintak penulisan dan cek log. 		
11. Mampu menentukan ijin yang diperlukan untuk mencegah pesan kesalahan ditampilkan ke publik.	11.1 Menjalankan fitur server side untuk menyembunyikan pesan kesalahan setelah web telah dipublish.	<ul style="list-style-type: none"> • Cek Pengaturan konfigurasi yang digunakan. 		
12. Mampu mengkonfigurasi perangkat lunak server untuk meminimalisir potensi serangan terhadap database.	12.1 Mengkonfigurasi variabel, session cookie, Batasan akses dan performansi setelah web telah dipublish.	<ul style="list-style-type: none"> • Cek Pengaturan konfigurasi yang digunakan. 		

Catatan :

.....

.....

.....

.....

Tanda Tangan Peserta Pelatihan :

Tanda Tangan Instruktur :

BAB III
PENILAIAN SIKAP KERJA

CEKLIS PENILAIAN SIKAP KERJA

Menyiapkan membuat web dinamis menggunakan script server side

INDICATOR UNJUK KERJA	NO. KUK	K	BK	KETERANGAN
1. Harus bertindak cermat, teliti, berpikir aktif inisiatif	1.1			
2. Harus bertindak cermat, teliti, berpikir evaluatif	1.2			
3. Harus bertindak teliti, akurat, dan memperhatikan SOP	1.3			
4. Harus bertindak teliti, akurat, dan memperhatikan SOP	2.1			
5. Harus bertindak cermat, teliti, berpikir analitis dan evaluatif	2.2			
6. Harus bertindak teliti, akurat, dan memperhatikan SOP	2.3			
7. Harus bertindak teliti, akurat, dan memperhatikan SOP	3.1			
8. Harus bertindak teliti, akurat, dan memperhatikan SOP	3.2			
9. Harus bertindak cermat, teliti, berpikir evaluatif	4.1			
10. Harus bertindak cermat, teliti, berpikir <i>analitis</i> dan <i>evaluatif</i>	4.2			
11. Harus bertindak cermat, teliti, berpikir evaluatif	5.1			
12. Harus bertindak teliti, akurat, dan memperhatikan SOP	5.2			

Catatan:

.....

Tanda Tangan Peserta :

Tanda Tangan Instruktur :

LAMPIRAN-LAMPIRAN

LAMPIRAN 1

Kunci Jawaban Penilaian Teori

NO. KUK	NO. SOAL	KUNCI JAWABAN	JAWABAN PESERTA	PENILAIAN		KETERANGAN
				K	BK	
	Isian					
	A.1	IIS				
	A.2	Brackets				
	A.3	hosting				
	A.4	cms				
	A.5	string				
	A.6	fgets				
	A.7	unlink				
	A.8	@				
	B-S					
	B.1	S				
	B.2	B				
	B.3	B				
	B.4	B				
	B.5	S				
	B.6	B				
	B.7	S				
	B.8	B				
	PG					
	C.1	D				
	C.2	B				
	C.3	D				
	C.4	C				
	C.5	A				
	C.6	A				
	C.7	D				
	C.8	C				
	C.9	C				
	C.10	A				
	C.11	A				
	C.12	B				
	C.13	A				
	C.14	C				
	C.15	B				
	C.16	A				
	C.17	C				
	C.18	A				
	C.19	D				
	C.20	C				

Jawaban Soal Essay

1. PHP (*PHP : Hypertext Preprocessor*) adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis.
2. Artinya PHP dapat berjalan di Sistem *Operasi Windows* dan beberapa *versi Linux*, dan PHP dapat dibangun sebagai modul pada *web server Apache* dan sebagai binary yang dapat berjalan sebagai CGI (*Common Gateway Interface*)
3. –
 - a. PHP digunakan sebagai landasan operasi pada pemrograman jaringan berbasis web.
 - b. PHP digunakan juga untuk pemrograman *database*.
 - c. PHP digunakan untuk membuat aplikasi web.
4. keunggulan yang dimiliki oleh PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam software basis data, sehingga dapat menciptakan suatu halaman web yang dinamis
5. *Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server*
6. artinya kode PHP dapat disisipkan pada sebuah halaman HTML



kptk.or.id



[instagram.com/lp3tk](https://www.instagram.com/lp3tk)



[facebook.com/lp3tk](https://www.facebook.com/lp3tk)



twitter.com/lp3tk



[youtube.com/lp3tk](https://www.youtube.com/lp3tk)