

IDENTITAS BUKU

APLIKASI TEKNOLOGI INFORMASI PENDUKUNG TUGAS KEPOLISIAN

Penyusun:

Tim Penyusun Kurikulum dan Hanjar Dikbangspes
Bintara/Gol.II PNS Polri Teknologi Informasi Dasar
Lemdiklat Polri T.A. 2022

Editor:

1. Kombes Pol. Nirboyo, S.I.K.
2. Pembina Rudiyanto, S.Pd.
3. Pembina Nolik Dwi Atmoko, S.E., M.E.
4. Kompol Yon Helmi
5. AKP I Gede Eka Irawan, S.T.
6. Penda Tk I Haryati
7. Penda Paramita Rahmadani, A.Md.
8. Briptu Ginawan Rahmat Permana, S.Kom.

Hanjar Pendidikan Polri
Pendidikan Pengembangan Spesialisasi
Bintara/Gol.II PNS Polri Teknologi Informasi Dasar n

Diterbitkan oleh:

Bagian Kurikulum dan Bahan Ajar Pendidikan Pengembangan Spesialisasi
Biro Kurikulum
Lembaga Pendidikan dan Pelatihan Polri
Tahun 2022

Hak cipta dilindungi Undang-Undang
Dilarang memperbanyak dan/atau mengutip sebagian atau seluruh isi Hanjar
Pendidikan Polri ini, tanpa izin tertulis dari Kalemdiklat Polri.

DAFTAR ISI

Cover	i
Sambutan Kalemdiklat Polri.....	ii
Keputusan Kalemdiklat Polri	iv
Lembar identitas buku.....	vi
Daftar isi	vii
 APLIKASI TEKNOLOGI INFORMASI PENDUKUNG TUGAS KEPOLISIAN	
Pendahuluan.....	1
Standar Kompetensi	1
 MODUL 1 APLIKASI TEKNOLOGI INFORMASI DALAM MENDUKUNG TUGAS POLRI	
Pengantar	2
Kompetensi Dasar	2
Materi Pelajaran.....	3
Metode Pembelajaran.....	4
Alat, Media, Bahan dan Sumber Belajar	5
Kegiatan Pembelajaran	5
Tagihan/Tugas.....	7
Lembar Kegiatan	7
Bahan Bacaan	8
 POKOK BAHASAN 1	
WEB POLRI	
1. Pengertian Web Polri	8
2. Kegunaan Web Polri.	8
3. Menu Informasi yang bisa diakses dalam Web Polri.....	8
4. Cara Mengakses Web Polri	8

POKOK BAHASAN 2**WEBMAIL POLRI**

1. Pengertian Webmail Polri.....	10
2. Kegunaan Webmail Polri.....	10
3. Kewajiban, Larangan dan Etika dalam menggunakan Email Polri.....	10
4. Cara Menggunakan Webmail Polri.....	12

POKOK BAHASAN 3**APLIKASI POLISIKU PRESISI**

1. Pengertian Aplikasi Polisiku Presisi	16
2. Fitur Utama Aplikasi Polisiku Presisi	16
3. Cara Akses Aplikasi Polisiku	16

POKOK BAHASAN 4**CALL CENTER 110**

1. Pengertian Call center 110.....	18
2. Tujuan Call center 110	18
3. Tugas Operator Layanan Polri 110	18
Rangkuman	19
Soal Latihan.....	19


MODUL 2 PEMBUATAN APLIKASI WEB


Pengantar	20
Kompetensi Dasar	20
Materi Pelajaran.....	21
Metode Pembelajaran.....	22

Alat, Media, Bahan dan Sumber Belajar	23
Kegiatan Pembelajaran	23
Tagihan/Tugas.....	25
Lembar Kegiatan	25
Bahan Bacaan	26
POKOK BAHASAN 1	
PERANCANGAN APLIKASI	
1. Pengertian Perancangan Aplikasi	26
2. Pengertian Algoritma Pemrograman	26
3. Perbedaan Algoritma dan Flowchart.....	26
4. Contoh Algoritma Pemrograman.....	27
5. Cara Kerja Algoritma Pemrograman	27
6. Fungsi Algoritma Pemrograman.....	27
POKOK BAHASAN 2	
PEMOGRAMAN APLIKASI WEB	
1. Pengertian Program dan Pemograman.....	29
2. Jenis-Jenis Bahasa Pemograman.....	29
POKOK BAHASAN 3	
CLIENT SERVER SEDERHANA	
1. Pengenalan Local XAMPP	34
2. Prosedur Instalasi XAMPP	34
3. Operasionalisasi XAMPP	38
4. Mengatasi Error Pada XAMPP	39
POKOK BAHASAN 4	
CARA MEMBUAT APLIKASI SEDERHANA	
1. Pembuatan Algoritma dan Flowchart Aplikasi.....	42
2. Pembuatan Aplikasi Menghitung Kecepatan.....	44


3.	Konfigurasi dan Menjalankan Xampp (Server Side).....	45
4.	Setting dan Koneksi DB PHP Myadmin	47
5.	Instalasi Sublime Text	52
6.	Pembuatan Web Sederhana dengan Sublime Text	55
7.	Perawatan Aplikasi Web yang Dibuat	61
	Rangkuman	67
	Soal Latihan.....	68


MODUL	APLIKASI TEKNOLOGI INFORMASI PENDUKUNG TUGAS KEPOLISIAN
	 40 JP (1.800 menit)

	<p>PENDAHULUAN</p> <p>Perkembangan teknologi digital di era 4.0. berdampak terhadap tingginya ekspektasi dan kecepatan layanan yang diberikan Polri. Dalam hal ini Polri dituntut berani melakukan adaptasi dan perubahan menuju keamanan dan ketertiban masyarakat baru. Secara Individu, dibutuhkan SDM Polri yang smart, profesional dan mampu menciptakan inovasi layanan digital serta menggunakannya dalam pelayanan terhadap masyarakat.</p> <p>Dalam menjawab tuntutan tersebut di atas, Polri menyediakan aplikasi pendukung tugas kepolisian untuk mempermudah dan mengefisienkan pelayanan terhadap masyarakat. Dalam hal ini setiap anggota Polri dituntut memiliki kemampuan mengoperasikan berbagai aplikasi pendukung pelaksanaan tugas kepolisian seperti Web Polri, Email Polri, Aplikasi Polisiku Presisi, Call center 110.</p> <p>Dalam upaya mewujudkan anggota Polri yang memiliki kemampuan dalam pengoperasian aplikasi pendukung tugas kepolisian, disiapkan modul Aplikasi Pendukung Tugas Kepolisian sebagai bahan rujukan bagi pendidik dan peserta didik dalam pendidikan spesialisasi.</p>
---	--


	<p>STANDAR KOMPETENSI</p> <p>Terampil membuat, mengoperasikan dan merawat aplikasi Teknologi Informasi (TI) pendukung tugas kepolisian.</p>
---	--

MODUL 01	APLIKASI TEKNOLOGI INFORMASI DALAM Mendukung TUGAS POLRI
	 10 JP (450 menit)


	<p>PENGANTAR</p> <p>Modul ini berisi materi pengertian Web Polri, kegunaan Web Polri, menu informasi yang bisa diakses dalam Web Polri, cara mengakses Web Polri, Pengertian Webmail Polri, Cara menggunakan Webmail Polri, Pengertian Aplikasi Polisiku Presisi, Fitur Aplikasi Polisiku Presisi, Cara mengakses Aplikasi Polisiku Presisi, Pengertian Call center 110, Tujuan Call center 110 dan Tugas Operator layanan Call center 110.</p> <p>Tujuannya agar peserta didik terampil mengoperasikan aplikasi Polri.</p>
---	--


	<p>KOMPETENSI DASAR</p> <ol style="list-style-type: none"> 1. Memahami Web Polri. <ul style="list-style-type: none"> Indikator Hasil Belajar: <ol style="list-style-type: none"> a. Menjelaskan pengertian Web Polri. b. Menjelaskan kegunaan Web Polri. c. Menjelaskan menu informasi yang bisa diakses dalam Web Polri. d. Menjelaskan cara mengakses Web Polri. 2. Memahami Webmail Polri. <ul style="list-style-type: none"> Indikator Hasil Belajar: <ol style="list-style-type: none"> a. Menjelaskan pengertian Webmail Polri. b. Menjelaskan kegunaan webmail Polri. c. Menjelaskan kewajiban, larangan dan etika dalam menggunakan Webmail Polri. d. Menjelaskan cara menggunakan Webmail Polri. 3. Memahami Aplikasi Polisiku Presisi. <ul style="list-style-type: none"> Indikator Hasil Belajar: <ol style="list-style-type: none"> a. Menjelaskan pengertian Aplikasi Polisiku Presisi. b. Menjelaskan fitur Aplikasi Polisiku Presisi. c. Menjelaskan cara mengakses Aplikasi Polisiku Presisi.
---	--


	<p>4. Memahami Call Center 110.</p> <p>Indikator Hasil Belajar:</p> <ol style="list-style-type: none"> a. Menjelaskan pengertian Call Center 110. b. Menjelaskan tujuan Call Center 110. c. Menjelaskan tugas operator layanan Call Center 110. <p>5. Terampil mengoperasikan aplikasi kepolisian.</p> <p>Indikator Hasil Belajar:</p> <ol style="list-style-type: none"> a. Mempraktikkan cara mengakses Web Polri. b. Mempraktikkan cara mengakses Email Polri. c. Mempraktikkan cara mengakses Aplikasi Polisiku Presisi. d. Mempraktikkan cara mengakses Call Center 110.
--	--

	<p>MATERI PELAJARAN</p> <p>1. Pokok Bahasan 1:</p> <p>Web Polri.</p> <p>Subpokok Bahasan:</p> <ol style="list-style-type: none"> a. Pengertian Web Polri. b. Kegunaan Web Polri. c. Menu informasi yang bisa diakses dalam Web Polri d. Cara mengakses Web Polri. <p>2. Pokok Bahasan 2:</p> <p>Webmail Polri.</p> <p>Subpokok Bahasan:</p> <ol style="list-style-type: none"> a. Pengertian Webmail Polri. b. Kegunaan webmail Polri. c. Kewajiban, Larangan dan Etika dalam menggunakan Email Polri. d. Cara menggunakan Webmail Polri. <p>3. Pokok Bahasan 3:</p> <p>Aplikasi Polisiku Presisi.</p> <p>Subpokok Bahasan:</p> <ol style="list-style-type: none"> a. Pengertian Aplikasi Polisiku Presisi. b. Fitur Aplikasi Polisiku Presisi. c. Cara mengakses Aplikasi Polisiku Presisi.
---	--


	<p>4. Pokok Bahasan 4:</p> <p>Call Center 110.</p> <p>Subpokok Bahasan:</p> <ol style="list-style-type: none"> a. Pengertian Call Center 110. b. Tujuan Call Center 110. c. Tugas operator layanan Call Center 110.
--	--


	<p>METODE PEMBELAJARAN</p> <ol style="list-style-type: none"> 1. Metode Ceramah Metode ini digunakan untuk menyampaikan materi aplikasi pendukung tugas kepolisian. 2. Metode Tanya Jawab Metode ini digunakan untuk memperdalam pemahaman materi dan untuk mengetahui tingkat penguasaan peserta didik tentang aplikasi pendukung tugas kepolisian. 3. Metode Demonstrasi Metode ini digunakan untuk memberikan contoh: <ol style="list-style-type: none"> a. Cara mengakses Web Polri. b. Cara mengakses Email Polri. c. Cara mengakses Aplikasi Polisiku Presisi. d. Cara mengakses Call Center 110. 4. Metode Praktik Metode ini digunakan untuk melatih keterampilan: <ol style="list-style-type: none"> a. Cara mengakses Web Polri. b. Cara mengakses Email Polri. c. Cara mengakses Aplikasi Polisiku Presisi. d. Cara mengakses Call Center 110. 5. Metode Penugasan Metode ini digunakan untuk memperdalam pemahaman materi, diberikan dalam bentuk meresume materi aplikasi pendukung tugas kepolisian.
---	---


	<h2 style="text-align: center;">ALAT, MEDIA, BAHAN DAN SUMBER BELAJAR</h2>
	<ol style="list-style-type: none"> 1. Alat, Media dan Bahan: <ol style="list-style-type: none"> a. <i>Whiteboard.</i> b. Papan <i>Flipchart.</i> c. Komputer/laptop. d. LCD dan <i>screen.</i> e. Alat tulis. 2. Sumber Belajar: <ol style="list-style-type: none"> a. Peraturan Kepala Divisi Teknologi Informasi dan Komunikasi Kepolisian Negara Republik Indonesia Nomor 1 Tahun 2017 Tentang Tata Kelola Email Kepolisian Negara Republik Indonesia. b. Peraturan Kapolri No 1 Tahun 2011 tentang Penyelenggaraan Sistem Telekomunikasi di Lingkungan Polri. c. Undang-undang No 36 Tahun 1999 tentang Telekomunikasi.


	<h2 style="text-align: center;">KEGIATAN PEMBELAJARAN</h2>
	<ol style="list-style-type: none"> 1. Tahap Awal: 10 menit <ol style="list-style-type: none"> a. Pendidik melaksanakan apersepsi: <ol style="list-style-type: none"> 1) Pendidik melaksanakan perkenalan. 2) Pendidik menyampaikan tujuan pembelajaran. 3) Pendidik menciptakan suasana pembelajaran yang kondusif. b. Peserta didik menyimak, menanggapi dan melaksanakan instruksi pendidik. 2. Tahap Inti: 430 menit <p>Tahap inti 1: penyampaian materi aplikasi pendukung tugas kepolisian (90 menit)</p> <ol style="list-style-type: none"> a. Pendidik menggali pendapat peserta didik tentang pemahaman awal materi yang akan dibahas. b. Pendidik menyampaikan materi aplikasi pendukung tugas kepolisian. c. Peserta didik menyimak, mencatat hal-hal yang penting dan menanggapi materi yang disampaikan pendidik. d. Pendidik memberikan waktu kepada peserta didik untuk bertanya tentang materi yang belum dipahami.

	<p>Tahap inti 2: demonstrasi dan praktik (340 menit)</p> <ul style="list-style-type: none">a. Pendidik memberikan contoh:<ul style="list-style-type: none">1) Cara mengakses Web Polri.2) Cara mengakses Email Polri.3) Cara mengakses Aplikasi Polisiku Presisi.4) Cara mengakses Call Center 110.b. Peserta didik memperhatikan dan menanyakan contoh demonstrasi yang belum dipahami.c. Pendidik menjawab pertanyaan yang diajukan oleh peserta didik.d. Pendidik menginstruksikan kepada peserta didik untuk melaksanakan praktik:<ul style="list-style-type: none">1) Cara mengakses Web Polri.2) Cara mengakses Email Polri.3) Cara mengakses Aplikasi Polisiku Presisi.4) Cara mengakses Call Center 110.e. Peserta didik melakukan praktik:<ul style="list-style-type: none">1) Cara mengakses Web Polri.2) Cara mengakses Email Polri.3) Cara mengakses Aplikasi Polisiku Presisi.4) Cara mengakses Call Center 110.f. Pendidik memfasilitasi pelaksanaan praktik.g. Pendidik memberikan koreksi dan memastikan hasil praktik sesuai ketentuan. <p>3. Tahap Akhir: 10 menit</p> <ul style="list-style-type: none">a. Pendidik mengecek penguasaan materi dengan cara bertanya secara lisan dan acak kepada peserta didik.b. Pendidik memberikan kesimpulan dan penguatan materi aplikasi pendukung tugas kepolisian.c. Pendidik melakukan evaluasi pembelajaran dan menutup pembelajaran.
--	---

	<p>TAGIHAN/TUGAS</p>
	<p>Peserta didik mengumpulkan resume tentang aplikasi teknologi informasi pendukung tugas kepolisian yang telah diterima dan dikumpulkan satu hari setelah pelajaran selesai.</p>

	<p>LEMBAR KEGIATAN</p>
	<p>-----</p>

	BAHAN BACAAN
	<p style="text-align: center;">POKOK BAHASAN WEB POLRI</p> <ol style="list-style-type: none"> 1. Pengertian Web Polri Web Polri adalah Situs atau Portal milik Polri yang berisikan informasi tentang institusi Polri baik internal maupun eksternal. Isi dari web Polri diperbaharui secara berkala oleh admin Div TIK Polri. 2. Kegunaan Web Polri Memberikan Informasi mengenai Polri maupun satuan tugas fungsinya dapat diakses melalui Web Polri www.polri.go.id sehingga masyarakat mudah memperoleh informasi tersebut. 3. Menu Informasi yang Bisa Diakses dalam Web Polri <ol style="list-style-type: none"> a. Tentang Polri; b. Informasi Publik; c. Layanan Polri; d. Hubungi Polri; e. Fitur Polri; f. Internal Polri. 4. Cara Mengakses Web Polri <ol style="list-style-type: none"> a. Pastikan komputer/laptop dalam keadaan tersambung ke internet. b. Buka web browsing apapun yang ada pada komputer/laptop pengguna (<i>Mozilla, google chrome, internet explorer, dll</i>). c. Klik menu pencarian, lalu tuliskan www.polri.go.id. d. Lalu akan muncul tampilan sebagai berikut:



e. Web Polri siap untuk diakses

POKOK BAHASAN 2

WEBMAIL POLRI

1. Pengertian Webmail Polri

Webmail adalah salah satu fasilitas akses yang diberikan untuk mengakses email Polri. Selain menggunakan webmail, email Polri juga dapat diakses melalui email *client* seperti *Microsoft Outlook, Mozilla Thunderbird, Apple Mail, Kmail, Evolution* dan lain-lain, termasuk akses menggunakan perangkat *mobile* berupa handphone, smartphone dan tablet.

Webmail bisa dipilih sebagai pilihan utama karena memiliki fleksibilitas dan kemudahan akses, antara lain bisa diakses dari berbagai lokasi komputer, dari berbagai browser dan dengan berbagai model tampilan. Webmail juga lebih mudah dipakai karena kita tidak perlu melakukan setting dan konfigurasi dimasing-masing klien. Proses pengelolaan email dan proses backup dapat lebih mudah karena data disatukan di satu tempat dengan tetap mempertahankan perlindungan keamanan dan kerahasiaan data.

Bagi yang sudah terbiasa menggunakan *Gmail* atau *Yahoo Mail*, tampilan webmail Polri secara umum mirip dengan keduanya sehingga mudah digunakan dan tidak perlu terlalu lama beradaptasi. Peruntukan email Polri ini untuk personil Polri seluruhnya termasuk ASN Polri, sehingga dalam pembuatan akun Polri hanya bisa dibuat oleh admin yang ditunjuk oleh Div TIK Polri agar dalam pembuatan akun baru menyesuaikan nomenklatur yang sudah ada.

2. Kegunaan Webmail Polri

Webmail Polri digunakan untuk:

- a. Sarana komunikasi antar personil Polri maupun instansi lain secara resmi.
- b. Sarana pengiriman data dan laporan kepolisian lebih aman.
- c. *Broadcasting* pesan email lebih mudah.

3. Kewajiban, Larangan dan Etika dalam menggunakan Email Polri

a. Kewajiban

Pengguna Email Polri berkewajiban untuk

- 1) Menjaga kerahasiaan, keamanan dan etika dalam menggunakan Email Polri;
- 2) Mengingat atau menyimpan kata sandi Email Polri pada tempat yang aman;

	<p>3) Email Polri hanya untuk kepentingan dinas; dan</p> <p>4) Menggunakan kata kunci dengan mengacu pada aspek keamanan, dan melakukan penggantian kata kunci secara berkala.</p> <p>b. Larangan</p> <p>Email Polri dilarang digunakan untuk:</p> <ol style="list-style-type: none"> 1) Kegiatan politik, kegiatan kriminal, mengandung pornografi, pesan/lampiran ilegal, hal yang tidak senonoh, memfitnah atau menghina; 2) Diskriminasi terhadap usia, gender, ras, orientasi seksual, agama atau politik; 3) Promosi atau publikasi seseorang, pandangan politik atau agama, pengoperasian bisnis atau apapun kegiatan untuk kepentingan pribadi; 4) Pendaftaran atau identitas diri pada aplikasi sosial media; 5) Membalas, meneruskan email berantai dan email berkategori sampah atau email lain yang belum dapat dikonfirmasi kebenarannya; 6) Mengungkapkan rahasia Polri atau informasi Polri yang bersifat rahasia; 7) Menyalin dan mendistribusikan materi yang dilindungi hak cipta; 8) Mengunduh atau menyalin berkas data dari sistem eksternal dan internet kecuali setelah memperoleh izin dari Kepala Satuan Kerja terkait; 9) Diberikan untuk mengakses dan mengirimkan email kepada orang lain; 10) Kegiatan yang dapat mengganggu kinerja jaringan, dan mengurangi kehandalan sistem informasi atau mengganggu operasionalisasi layanan TIK Polri <p>c. Etika</p> <ol style="list-style-type: none"> 1) Email Polri hanya digunakan untuk kepentingan resmi komunikasi institusi Polri dan tidak menggunakannya untuk kegiatan yang dapat merusak nama baik institusi Polri; 2) Mengirim lampiran email (attachment) dalam jumlah yang wajar; 3) Tidak mengirim email dalam bentuk huruf besar semua; dan 4) Tidak mengirim email dalam ukuran huruf yang sangat besar dan warna-warni.
--	---

4. Cara Menggunakan Webmail Polri

Berikut adalah cara menggunakan webmail Polri untuk penggunaan email sehari-hari. Cara penggunaan mencakup pengiriman dan penerimaan email termasuk tips-tips dalam membalas, memforward, menambah *attachment* dan menggunakan fitur lain yang disertakan pada Zimbra.

a. Membuka Webmail menggunakan web browser

Untuk membuka webmail, kita menggunakan browser dan menyetikkan alamat <https://webmail.polri.go.id>, <https://mail.polri.go.id> atau bisa melalui web Polri di buka di menu Internal.

Zimbra akan memunculkan halaman login sebagai berikut:

Masukkan nama dan password pada *form* isian. Jika kita ingin menyimpan nama dan *password* agar lain kali tidak usah mengetikkannya, klik pada pilihan **Stay Sign In**. Bagi pengguna layanan email sebelumnya, *user name* dan *password* bisa menggunakan *user name* dan *password* yang lama. Jika *user name* dan *password* yang lama tidak dapat digunakan, silakan mengkonfirmasi pada bagian layanan email Divisi Teknologi Informasi POLRI.

Pilih pilihan *Default* pada bagian **version**. Pilihan ini akan menampilkan Zimbra dalam modus **Advanced** yang mendukung berbagai fungsi tambahan, misalnya fungsi *drag drop*. Selain pilihan *Default*, terdapat juga pilihan untuk standar dalam format HTML yang lebih cepat diakses. Format standar ini menghilangkan beberapa fungsi dan pilihan yang hanya ada pada pilihan *Advanced*.

Format advanced berbasis AJAX dan multi fungsi sedangkan format HTML bersifat standar. Akses format HTML jauh lebih cepat, terutama jika *bandwidth* yang digunakan terbatas, namun tampilannya lebih sederhana dibandingkan format *advanced*.

Selain 2 pilihan di atas, ada juga pilihan *mobile* jika ingin mengakses Zimbra Webmail melalui perangkat handphone atau PDA. Jika modus ini yang dipilih, Zimbra hanya akan

menampilkan webmail dalam format sederhana dan disesuaikan dengan ukuran HP/PDA. Selain itu, pilihan ini juga menghilangkan berbagai fitur agar lebih cepat diakses dan bisa menghemat pulsa jika diakses menggunakan jaringan GSM.

Perlu diingat bahwa pilihan *advanced* memerlukan resolusi layar monitor minimal 1024 X 768. Setting layar kurang dari ukuran tersebut biasanya akan secara otomatis masuk kedalam modus Standar HTML. Setelah kita login, kita akan menjumpai tampilan utama webmail POLRI sebagai berikut:

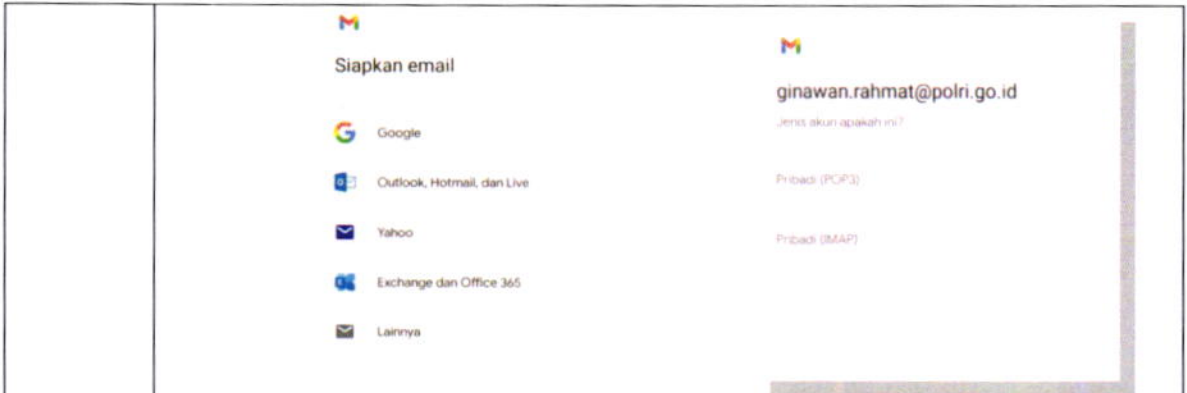
Tampilan utama webmail POLRI terbagi dalam beberapa bagian, antara lain:

- 1) Bagian Kiri: Logo, informasi folder email dan kalender.
- 2) Bagian Atas 1: Berisi kotak pencarian dan pilihan tampilan informasi account, termasuk menu *sign out*, profile dan pilihan personal lainnya.
- 3) Bagian Atas 2: Berisi tab email, *Address Book*, Kalender, *Tasks*, *Documents*, *Briefcase* dan *Preferences*.
- 4) Bagian Atas 3: Berisi pilihan *New* (untuk membuat email, *task*, *address book* dll), *reply* email, *forward* email, menandai email sebagai *spam (junk)* dan posisi chat.
- 5) Bagian Atas 4: Berisi keterangan *flag* email (dalam bentuk gambar bendera, email penting dilambungkan dalam bentuk *flag* tanda seru, *flag attachment* dan keterangan label judul email).
- 6) Bagian tengah: Berisi daftar email yang diterima atau dikirim.
- 7) Bagian bawah/samping: Merupakan isi dari email yang sedang dipilih dari daftar yang ada.

Secara *default*, semua pengguna yang baru pertama kali menggunakan webmail Polri akan secara otomatis mengikuti format standard tampilan isi email disisi samping kanan dan email dikelompokkan dalam kelompok percakapan (*group conversation*). Konfigurasi ini bisa diubah dengan menggunakan tombol **View** pada sisi kanan tampilan email.

b. Membuka email melalui Smartphone

Untuk membuka email Polri menggunakan aplikasi pada Smartphone (Android) dengan membuka aplikasi Gmail, klik pojok kanan atas dan pilih **Tambahkan akun lainnya**. Tampilan sebagai berikut:

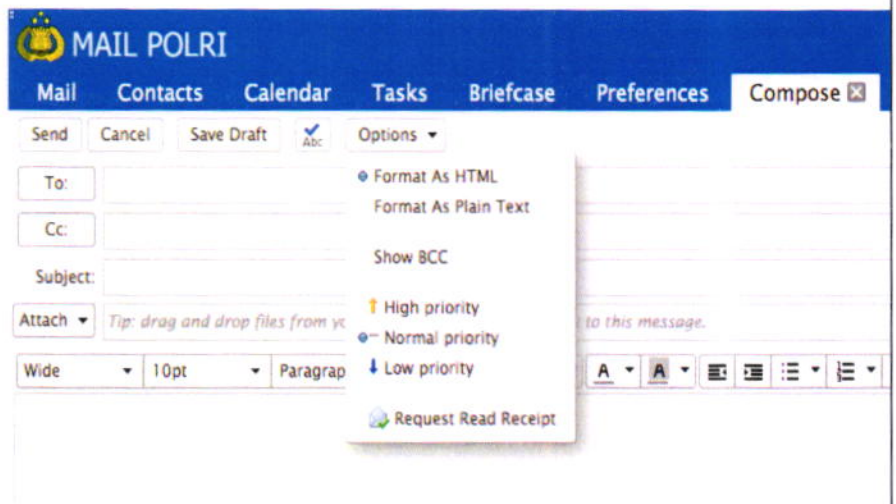


- 1) Pada bagian **Siapkan Email** pilih Lainnya;
- 2) Jenis akun terdapat 2 jenis protokol penerimaan email yaitu POP3 (*offline*) setiap email masuk harus *refresh* ulang ke *server*, IMAP (*online*) setiap email baru otomatis masuk. Di sarankan untuk menggunakan IMAP agar lebih mudah;
- 3) *Input* alamat email Polri dan password;
- 4) *Setting server* menggunakan **mail.polri.go.id** sedangkan pada smtp server menggunakan **smtp.polri.go.id**;
- 5) Tahap terakhir membuat *display name*.

Dengan melakukan login email Polri pada smartphone akan memudahkan dalam pengelolaan email masuk dan keluar tanpa harus membuka pada web browser.

c. Membuat dan Mengirim Email

Untuk membuat email, klik menu *New* dan pilih *New Email*. Lihat gambar berikut:



Setelah kita memilih menu *New Email*, kita dapat langsung mengetikkan email yang akan dikirimkan.

Isikan data berikut:

- 1) To: Alamat email yang dituju
- 2) Cc: Alamat email yang ingin mendapatkan salinan/tembusan email yang dikirimkan
- 3) Bcc: Klik pada pilihan **Option | Show Bcc** . Bcc merupakan singkatan dari *Blind Carbon Copy*, artinya si penerima pada posisi To dan Cc tidak akan tahu kalau email tersebut dikirimkan pada si bcc.

Contoh kasus : Salah satu user mengirim email ke (to) vivian@polri.go.id di cc-kan ke admin@polri.go.id dan di bcc ke atasan@polri.go.id maka vivian akan menerima email sebagai penerima asli, admin akan menerima email sebagai penerima email salinan atau tembusan sedangkan atasan akan menerima salinan email tanpa diketahui oleh vivian dan admin. Bcc biasanya diperlukan jika ingin mengirim email kategori rahasia dan antara penerima yang satu dengan penerima yang lain diharapkan tidak tahu bahwa email yang sama dikirimkan kepada pihak lain.

- 1) Subject: Ketikkan subyek dari email yang mencerminkan isi email. Hindari pengiriman email tanpa subyek dan penggunaan subyek email yang tidak jelas. Jangan menulis subject dalam huruf besar semua, karena subject dalam huruf kapital biasanya akan diberi penalti oleh layanan anti spam sebagai email mencurigakan.
- 2) Untuk mengirim *attachment*, klik tombol **Attach**. Ketikkan isi email pada bagian isi dan setelah dipastikan benar, klik tombol *Send*.
- 3) Pada bagian isi email harus dituliskan pesan walupun singkat untuk menghindari email yang dikirim dianggap spam.

POKOK BAHASAN 3

APLIKASI POLISIKU PRESISI

1. Pengertian Aplikasi Polisiku Presisi

Dalam rangka lebih cepat memberikan pelayanan kepada masyarakat, Polri telah melaunching aplikasi Polisiku Presisi, yaitu aplikasi perantara bantuan polisi kepada masyarakat yang dapat didownload di android *play store* dan apple *app store*.

2. Fitur Utama Aplikasi Polisiku Presisi

Aplikasi Polisiku Presisi memiliki fitur utama yaitu untuk mencari pos Polisi terdekat dari posisi masyarakat. Selain itu terdapat fitur lain antara lain:

- a. Melakukan panggilan telepon call center 110 melalui jaringan seluler Voice.
- b. Mencari pos Polisi dan teleponnya di seluruh Indonesia.
- c. Melakukan pengaduan masyarakat dengan terintegrasi Dumas Presisi.
- d. Serta sebagai sarana penyaluran informasi dari Humas Polri kepada masyarakat.
- e. Memberikan aspirasi melalui fitur Halo Polisiku.
- f. Fitur layanan publik seperti SKCK Online dan SIM Online.

Kedepannya Polri akan mengembangkan aplikasi Superapss Presisi sebagai mal layanan publik Polri secara digital yang bisa diakses kapan saja dan dimana saja. Sehingga masyarakat tidak perlu menginstall banyak aplikasi untuk mengakses layanan-layanan publik milik Polri.

3. Cara Akses Aplikasi Polisiku

- a. Pastikan handphone dalam keadaan tersambung ke internet.
- b. Download aplikasi polisiku di *play store* atau *app store* pada ponsel pengguna.
- c. Lalu akan muncul tampilan sebagai berikut:



PolisiKu 
PRECISI



d. Aplikasi siap untuk dijalankan.

POKOK BAHASAN 4

CALL CENTER 110

1. Pengertian Call Center 110

Call center 110 merupakan pusat berbagai pengaduan gangguan Kamtibmas, tindak pidana kriminalitas, musibah kecelakaan lalu lintas mau pun kejadian lainnya. Prinsipnya, apa pun gejala di tengah masyarakat yang berpotensi menimbulkan kerugian.

2. Tujuan Call Center 110

Kehadiran Layanan Polri 110 ditujukan untuk memenuhi harapan dan kebutuhan masyarakat terhadap terselenggaranya layanan keamanan publik. Dalam penyelenggaraan layanan contact center, telah disiapkan sebuah sistem aplikasi yang dapat memungkinkan pencatatan/perekaman setiap interaksi Polri dan masyarakat, sehingga dimungkinkan pengendalian response kebutuhan masyarakat terhadap Polri.

Sistem tersebut direncanakan akan membuka saluran via : telepon, SMS, email, fax dan media sosial yang didukung oleh jaringan Telkom Group di Indonesia.


Masyarakat yang nantinya melakukan panggilan ke nomor akses 110 akan langsung terhubung ke agen yang akan memberikan layanan berupa informasi, pelaporan (kecelakaan, bencana, kerusakan, dll) dan pengaduan (penghinaan, ancaman, tindak kekerasan dll).


Masyarakat bisa menggunakan layanan Polri 110 secara gratis. Namun demikian, Polri menghimbau agar layanan Polri 110 ini tidak dibuat main-main, karena jika nantinya terjadi seperti itu, pihak Polri tentu akan melacak masyarakat yang membuat laporan bohong.


3. Tugas Operator Layanan Polri 110


- a. Melakukan verifikasi data pelapor/pengadu.
- b. Melakukan klarifikasi kebutuhan pelapor/pengadu.
- c. Mencatat Laporan/Pengaduan secara singkat kedalam aplikasi Layanan Polisi 110.
- d. Memberitahukan nomor Laporan/Pengaduan kepada pelapor/pengadu secara lisan dan notifikasi melalui *Short Message Service* dengan *sender* Polisi_110.
- e. Meneruskan Laporan/Pengaduan ke SPKT yang akan dilanjutkan ke Petugas Piket fungsi terkait.
- f. Memonitor tindak lanjut dari satuan fungsi kepolisian terkait dan/atau petugas di lapangan.


	<p>g. Menerima laporan perkembangan penanganan Laporan/Pengaduan dari petugas di lapangan atau unit patrol.</p> <p>h. Memutakhirkan data status Laporan/Pengaduan sudah selesai ditindaklanjuti ke dalam aplikasi Layanan Polisi 110.</p>
--	---

	<p>RANGKUMAN</p> <p>Aplikasi pendukung tugas kepolisian terdiri dari: (1) Web Polri adalah Situs atau Portal milik Polri yang berisikan informasi tentang institusi Polri baik internal maupun eksternal. (2) Email Polri. email Polri dapat diakses melalui email <i>client</i> seperti <i>Microsoft Outlook</i>, <i>Mozilla Thunderbird</i>, <i>Apple Mail</i>, <i>Kmail</i>, <i>Evolution</i> dan lain-lain. (3) Aplikasi Polisiku Presisi. Aplikasi Polisiku memiliki fitur utama yaitu untuk mencari pos Polisi terdekat dari posisi masyarakat. (4) Call center 110. Kehadiran Layanan Polri 110 ditujukan untuk memenuhi harapan dan kebutuhan masyarakat terhadap terselenggaranya layanan keamanan publik.</p>
---	--


	<p>SOAL LATIHAN</p> <ol style="list-style-type: none"> 1. Jelaskan pengertian Web Polri! 2. Jelaskan kegunaan Web Polri! 3. Jelaskan menu informasi yang bisa diakses dalam Web Polri! 4. Jelaskan cara mengakses Web Polri! 5. Jelaskan Pengertian Webmail Polri! 6. Jelaskan kegunaan webmail Polri! 7. Jelaskan kewajiban, larangan dan etika dalam menggunakan Webmail Polri! 8. Jelaskan cara menggunakan Webmail Polri! 9. Jelaskan pengertian Aplikasi Polisiku Presisi! 10. Jelaskan fitur Aplikasi Polisiku Presisi! 11. Jelaskan cara mengakses Aplikasi Polisiku Presisi! 12. Jelaskan pengertian Call Center 110! 13. Jelaskan tujuan Call Center 110! 14. Jelaskan tugas operator layanan Call Center 110!
--	--

MODUL 02	PEMBUATAN APLIKASI WEB
	 30 JP (1.350 menit)


	<p>PENGANTAR</p> <p>Modul ini berisi materi pengertian perancangan aplikasi, pengertian algoritma pemrograman, perbedaan algoritma dan flowchart, contoh algoritma pemrograman, cara kerja algoritma pemrograman, fungsi algoritma pemrograman, pengertian program dan pemrograman, jenis-jenis bahasa pemrograman, pengenalan local XAMPP, prosedur Instalasi XAMPP, operasionalisasi XAMPP, mengatasi error pada XAMPP, pembuatan algoritma dan flowchart aplikasi, pembuatan aplikasi menghitung kecepatan, konfigurasi dan menjalankan XAMPP (Server Side), setting dan koneksi DB PHP MyAdmin, instalasi sublime text, pembuatan web sederhana dengan sublime text dan perawatan aplikasi web yang dibuat.</p> <p>Tujuannya agar peserta didik mampu membuat dan mengoperasikan aplikasi web.</p>
---	---

	<p>KOMPETENSI DASAR</p> <ol style="list-style-type: none"> 1. Memahami perancangan aplikasi web. <ul style="list-style-type: none"> Indikator Hasil Belajar: <ol style="list-style-type: none"> a. Menjelaskan pengertian perancangan aplikasi. b. Menjelaskan pengertian algoritma pemrograman. c. Menjelaskan perbedaan algoritma dan flowchart d. Menjelaskan contoh algoritma pemrograman. e. Menjelaskan cara kerja algoritma pemrograman. f. Menjelaskan fungsi algoritma pemrograman. 2. Memahami pemrograman aplikasi web. <ul style="list-style-type: none"> Indikator Hasil Belajar: <ol style="list-style-type: none"> a. Menjelaskan pengertian program dan pemrograman. b. Menjelaskan jenis-jenis bahasa pemrograman. 3. Memahami <i>client server</i> sederhana. <ul style="list-style-type: none"> Indikator Hasil Belajar: <ol style="list-style-type: none"> a. Menjelaskan pengenalan local XAMPP. b. Menjelaskan prosedur instalasi XAMPP.
---	--


	<p>c. Menjelaskan operasionalisasi XAMPP.</p> <p>d. Menjelaskan mengatasi error pada XAMPP.</p> <p>4. Memahami cara membuat aplikasi web sederhana.</p> <p>Indikator Hasil Belajar</p> <p>a. Menjelaskan pembuatan algoritma dan flowchart aplikasi.</p> <p>b. Menjelaskan pembuatan aplikasi menghitung kecepatan.</p> <p>c. Menjelaskan konfigurasi dan menjalankan XAMPP (Server Side).</p> <p>d. Menjelaskan setting dan koneksi DB PHP MyAdmin.</p> <p>e. Menjelaskan instalasi <i>sublime text</i>.</p> <p>f. Menjelaskan pembuatan web sederhana dengan <i>sublime text</i>.</p> <p>g. Menjelaskan perawatan aplikasi web yang dibuat.</p> <p>5. Mampu membuat dan mengoperasikan aplikasi web.</p> <p>Indikator Hasil Belajar:</p> <p>a. Mempraktikkan pembuatan algoritma dan flowchart aplikasi.</p> <p>b. Mempraktikkan pembuatan aplikasi menghitung kecepatan.</p> <p>c. Mempraktikkan konfigurasi dan menjalankan XAMPP (Server Side).</p> <p>d. Mempraktikkan setting dan koneksi DB PHP MyAdmin.</p> <p>e. Mempraktikkan instalasi <i>sublime text</i>.</p> <p>f. Mempraktikkan pembuatan web sederhana dengan <i>sublime text</i>.</p> <p>g. Mempraktikkan perawatan aplikasi yang dibuat.</p>
--	---


	<p>MATERI PELAJARAN</p> <p>1. Pokok Bahasan 1:</p> <p>Perancangan aplikasi</p> <p>Subpokok Bahasan:</p> <p>a. Pengertian perancangan aplikasi.</p> <p>b. Pengertian algoritma pemrograman.</p> <p>c. Perbedaan algoritma dan flowchart</p> <p>d. Contoh algoritma pemrograman.</p> <p>e. Cara kerja algoritma pemrograman.</p> <p>f. Fungsi algoritma pemrograman.</p> <p>2. Pokok Bahasan 2:</p> <p>Pemrograman aplikasi web.</p> <p>Subpokok Bahasan:</p> <p>a. Pengertian program dan pemrograman.</p> <p>b. Jenis-jenis bahasa pemrograman.</p>
---	--

	<p>3. Pokok Bahasan 3:</p> <p><i>Client server sederhana.</i></p> <p>Subpokok Bahasan:</p> <ol style="list-style-type: none"> a. Pengenalan Local XAMPP. b. Prosedur Instalasi XAMPP. c. Operasionalisasi XAMPP. d. Mengatasi error pada XAMPP. <p>4. Pokok Bahasan 4:</p> <p>Cara membuat aplikasi sederhana.</p> <p>Subpokok Bahasan:</p> <ol style="list-style-type: none"> a. Pembuatan algoritma dan flowchart aplikasi. b. Pembuatan aplikasi menghitung kecepatan. c. Konfigurasi dan menjalankan XAMPP (Server Side). d. Setting dan koneksi DB PHP MyAdmin. e. Instalasi sublime text. f. Pembuatan web sederhana dengan sublime text. g. Perawatan aplikasi web yang dibuat.
--	---

	<p>METODE PEMBELAJARAN</p> <ol style="list-style-type: none"> 1. Metode Ceramah Metode ini digunakan untuk menyampaikan materi pembuatan aplikasi web. 2. Metode Tanya Jawab Metode ini digunakan untuk memperdalam pemahaman materi dan untuk mengetahui tingkat penguasaan peserta didik tentang pembuatan aplikasi web. 3. Metode Demonstrasi Metode ini digunakan untuk memberikan contoh: <ol style="list-style-type: none"> a. Pembuatan algoritma dan flowchart aplikasi. b. Pembuatan aplikasi menghitung kecepatan. c. Konfigurasi dan menjalankan XAMPP (server side). d. Setting dan koneksi DB PHP MyAdmin. e. Instalasi sublime text. f. Pembuatan web sederhana dengan sublime text. g. Perawatan aplikasi yang dibuat. 4. Metode Praktik Metode ini digunakan untuk melatih keterampilan. <ol style="list-style-type: none"> a. Pembuatan algoritma dan flowchart aplikasi. b. Pembuatan aplikasi menghitung kecepatan.
---	--

	<ul style="list-style-type: none"> c. Konfigurasi dan menjalankan XAMPP (Server Side). d. Setting dan koneksi DB PHP MyAdmin. e. Instalasi sublime text. f. Pembuatan web sederhana dengan sublime text. g. Perawatan aplikasi yang dibuat. <p>5. Metode Penugasan</p> <p>Metode ini digunakan untuk memperdalam pemahaman materi, diberikan dalam bentuk meresume materi pembuatan aplikasi sederhana.</p>
--	---

	<p>ALAT, MEDIA, BAHAN DAN SUMBER BELAJAR</p>
	<ol style="list-style-type: none"> 1. Alat, Media dan Bahan: <ul style="list-style-type: none"> a. <i>Whiteboard.</i> b. <i>Papan Flipchart.</i> c. <i>Komputer/laptop.</i> d. <i>LCD dan screen.</i> e. <i>Alat tulis.</i> 2. Sumber Belajar: <ul style="list-style-type: none"> a. Peraturan Kapolri Nomor 1 Tahun 2011 tentang Penyelenggaraan Sistem Telekomunikasi di Lingkungan Polri. b. Undang-undang Nomor 36 Tahun 1999 tentang Telekomunikasi.

	<p>KEGIATAN PEMBELAJARAN</p>
	<ol style="list-style-type: none"> 1. Tahap Awal: 20 menit <ul style="list-style-type: none"> a. Pendidik melaksanakan apersepsi: <ul style="list-style-type: none"> 1) Pendidik merefleksikan materi sebelumnya. 2) Pendidik menyampaikan tujuan pembelajaran. 3) Pendidik menciptakan suasana pembelajaran yang kondusif. b. Peserta didik menyimak, menanggapi dan melaksanakan instruksi pendidik. 2. Tahap Inti : 1.230 menit <p>Tahap inti 1: penyampaian materi pembuatan aplikasi web sederhana (270 menit)</p>

- a. Pendidik menggali pendapat peserta didik tentang pemahaman awal materi yang akan dibahas.
- b. Pendidik menyampaikan materi pembuatan aplikasi web sederhana.
- c. Peserta didik menyimak, mencatat hal-hal yang penting dan menanggapi materi yang disampaikan pendidik.
- d. Pendidik memberikan waktu kepada peserta didik untuk bertanya tentang materi yang belum dipahami.


Tahap inti 2: demonstrasi pembuatan aplikasi web sederhana (270 menit)


- a. Pendidik memberikan contoh:
 - 1) Pembuatan algoritma dan flowchart aplikasi.
 - 2) Pembuatan aplikasi menghitung kecepatan.
 - 3) Konfigurasi dan menjalankan XAMPP (Server Side).
 - 4) Setting dan koneksi DB PHP MyAdmin.
 - 5) Instalasi sublime text.
 - 6) Pembuatan web sederhana dengan sublime text.
 - 7) Perawatan aplikasi yang dibuat.
- b. Peserta didik memperhatikan dan menanyakan contoh demonstrasi yang belum dipahami.
- c. Pendidik menjawab pertanyaan yang diajukan oleh peserta didik.

Tahap inti 3: praktik pembuatan aplikasi web sederhana (690 menit)

- a. Pendidik menginstruksikan kepada peserta didik untuk melaksanakan praktik:
 - 1) Pembuatan algoritma dan flowchart aplikasi.
 - 2) Pembuatan aplikasi menghitung kecepatan.
 - 3) Konfigurasi dan menjalankan XAMPP (Server Side).
 - 4) Setting dan koneksi DB PHP MyAdmin.
 - 5) Instalasi sublime text.
 - 6) Pembuatan web sederhana dengan sublime text.
 - 7) Perawatan aplikasi yang dibuat.
- b. Peserta didik melakukan praktik:
 - 1) Pembuatan algoritma dan flowchart aplikasi.
 - 2) Pembuatan aplikasi menghitung kecepatan.
 - 3) Konfigurasi dan menjalankan XAMPP (Server Side).
 - 4) Setting dan koneksi DB PHP MyAdmin.
 - 5) Instalasi sublime text.
 - 6) Pembuatan web sederhana dengan sublime text.
 - 7) Perawatan aplikasi yang dibuat.
- c. Pendidik memfasilitasi pelaksanaan praktik.
- d. Pendidik memberikan koreksi dan memastikan hasil praktik sesuai ketentuan.

	<p>3. Tahap Akhir: 10 menit</p> <ol style="list-style-type: none"> Pendidik mengecek penguasaan materi dengan cara bertanya secara lisan dan acak kepada peserta didik. Pendidik memberikan kesimpulan dan penguatan materi pembuatan aplikasi sederhana. Pendidik melakukan evaluasi pembelajaran dan menutup pembelajaran. <p>4. Tes Kemampuan Aplikasi Kepolisian: 90 menit</p>
--	---

	<p>TAGIHAN/TUGAS</p> <hr/> <p>Peserta didik mengumpulkan hasil praktik berupa aplikasi web</p>
---	---

	<p>LEMBAR KEGIATAN</p> <hr/> <hr style="border-top: 1px dashed black;"/>
--	---



BAHAN BACAAN

POKOK BAHASAN 1 PERANCANGAN APLIKASI

1. Pengertian Perancangan Aplikasi

Dalam merancang sebuah aplikasi diperlukan metode atau langkah-langkah yang diperlukan atau dikenal dengan algoritma. Pengertian algoritma pemrograman adalah suatu alur yang dipergunakan dalam suatu perhitungan atau pemecahan suatu masalah secara sistematis, serta dalam kegiatan pemrograman algoritma biasanya dianggap sebagai sebuah logika untuk menentukan program yang akan dibuat.

Dalam definisi lain, algoritma pemrograman adalah serangkaian proses yang wajib diikuti dalam suatu perhitungan pemecahan masalah yang lain, terutama pada program komputer. Algoritma pemrograman adalah suatu pemecahan masalah dengan suatu susunan yang logis berdasarkan sistematika tertentu.

2. Pengertian Algoritma Pemograman

Algoritma pemrograman adalah suatu hal yang paling dasar, oleh karenanya kamu harus bisa memahami dan mengetahui tahap awal dari pembuatan sebuah program. Untuk menyelesaikan algoritma di pemrograman dapat dilakukan menggunakan dua cara yaitu tulisan dan gambar, namun dari dua metode itu kembali lagi kepada seorang programmer, ingin menggunakan metode yang mana.

3. Perbedaan Algoritma dan Flowchart

Algoritma pemrograman adalah sebuah proses secara sistematis dan bertujuan untuk menyelesaikan masalah dengan salah satu metode penyampaiannya yaitu melalui flowchart. Flowchart adalah sebuah gambaran atau diagram yang memiliki satu aliran atau lebih, namun hal itu hanya dapat berlaku secara sekuensial atau saling berkesinambungan. Flowchart juga memiliki simbol-simbol tertentu yang bisa menjadi sebuah gambaran dari setiap alur yang saling terhubung dengan anak panah.

Jadi, sekarang sudah jelas bukan, bahwa kedua istilah tersebut berbeda. Meski begitu, algoritma dan flowchart tidak bisa dipisahkan, sebab flowchart menggambarkan desain program yang akan dibuat dengan berpedoman pada algoritma atau pola pikir.

4. Contoh Algoritma Pemrograman

Sekali lagi, algoritma pemrograman adalah suatu alur proses sistematis untuk memecahkan masalah. Contoh sederhananya ialah ada data berupa nama dan nilai siswa, jika nilai siswa lebih besar atau sama dengan 80 maka siswa tersebut dinyatakan lulus, tetapi jika nilai siswa di bawah nilai 80 maka dinyatakan tidak lulus. Maka algoritmanya seperti berikut:

- a. Baca nama dan nilai siswa
- b. Jika nilai ≥ 80 maka
- c. Keterangan = lulus
- d. Tetapi jika
- e. Keterangan = tidak lulus
- f. Tulis nama dan keterangan

5. Cara Kerja Algoritma Pemrograman

Dari contoh di atas, dapat kita simpulkan cara kerja algoritma pemrograman adalah deskripsi dari sebuah proses untuk mengerjakan sesuatu yang disusun dalam sederet aksi. Sederhananya prinsip kerja algoritma di pemrograman terbagi menjadi tiga, yaitu input \Rightarrow proses \Rightarrow output.

Dalam kehidupan sehari-hari, prinsip tersebut dapat kita pahami dengan proses sederhana seperti merebus air. Pertama-tama, mari kita definisikan input dan output. Berdasarkan proses merebus air, yang menjadi input adalah air mentah dan outputnya adalah air matang. Jadi, susunan algoritma pemrogramannya adalah:

- a. Ambil panci
- b. Isi dengan air
- c. Letakkan pada kompor
- d. Nyalakan api kompor
- e. Tunggu hingga keluar gelembung
- f. Air sudah matang

6. Fungsi Algoritma Pemrograman

Fungsi utama algoritma pemrograman adalah untuk menyelesaikan suatu masalah. Algoritma ini merupakan suatu hal yang penting dalam aktivitas membuat suatu program. Berikut fungsi algoritma pada pemrograman lainnya:

- a. Dapat menyelesaikan permasalahan rumit dalam suatu program yang melibatkan perhitungan tingkat tinggi.
- b. Mampu menyederhanakan suatu program sehingga penggunaannya lebih efektif dan efisien.
- c. Dapat digunakan secara berulang sehingga meminimalkan

	<p>penulisan program yang berulang.</p> <ul style="list-style-type: none">d. Mempermudah pencarian kesalahan serta dapat diperbaiki dalam waktu yang singkat.e. Adanya alur proses yang jelas sehingga kamu dapat dengan mudah mencari kesalahan jika terjadi error.
--	---

POKOK BAHASAN 2

PEMOGRAMAN APLIKASI WEB

1. Pengertian Program dan Pemrograman

a. Program

Program dapat diartikan:

- 1) untuk mendeskripsikan instruksi-instruksi tersendiri, disebut *source code*, yang dibuat oleh programmer;
- 2) untuk mendeskripsikan keseluruhan bagian dari perangkat lunak yang *executable*.

Atau secara umum bisa diartikan sebagai himpunan atau kumpulan instruksi tertulis yang dibuat oleh programmer atau suatu bagian *executable* dari suatu perangkat lunak.

b. Pemrograman

Pemrograman adalah cara membuat program menjadi sebuah program komputer, atau bisa diartikan sebagai suatu kumpulan urutan perintah ke komputer untuk mengerjakan sesuatu.

Bahasa adalah suatu kumpulan simbol-simbol atomic yang terbatas atau berbentuk alfabet. Untaian yang menjadi bagian dari suatu bahasa disebut kata atau kalimat. Bahasa yang dapat dimengerti oleh komputer disebut bahasa pemrograman. Dengan bahasa pemrograman seseorang dapat berkomunikasi dengan suatu alat dan memberikan perintah tertentu. Aplikasi Microsoft Office merupakan hasil dari sekumpulan intruksi dari Bahasa pemrograman yang sudah dibuat dengan *user interface* sehingga *user* bisa berkomunikasi untuk melakukan pengetikan, pengeditan serta pencetakan lembar dokumen.

2. Jenis-Jenis Bahasa Pemrograman

Berdasarkan tingkat kedekatannya dengan logika komputer, bahasa pemrograman terbagi menjadi tiga jenis sebagai berikut:

a. Bahasa pemrograman tingkat rendah

Bahasa tingkat rendah adalah jenis bahasa pemrograman yang paling dekat hubungannya dengan logika komputer. Bahasa jenis ini bisa dijalankan di komputer tanpa perlu diterjemahkan terlebih dahulu dengan Compiler atau Interpreter. Nah, bahasa pemrograman tingkat rendah dibagi lagi menjadi dua, yaitu:

1) Bahasa mesin

Bahasa mesin adalah bahasa asli yang dipahami komputer. Bahasa ini juga dikenal sebagai kode mesin atau kode objek. Umumnya, bahasa mesin ditampilkan dalam kode biner (0 dan 1).

```
10100001 10111100 10010011 00000100
00001000 00000011 00000101 11000000
10010011 00000100 00001000 10100011
11000000 10010100 00000100 00001000
```

Oleh karena itu, bahasa mesin cenderung sulit dipahami oleh manusia. Namun positifnya, program yang ditulis dengan bahasa mesin punya kinerja yang sangat gesit, karena tidak perlu diterjemahkan lagi.

2) Bahasa assembly

Bahasa assembly atau bahasa rakitan adalah jenis bahasa tingkat rendah untuk membangun sistem yang berhubungan langsung dengan perangkat keras. Contohnya untuk membuat sistem operasi atau driver.

```
section      .text
global      _start

_start:

    mov     edx, len
    mov     ecx, msg
    mov     ebx, 1
    mov     eax, 4
    int     0x80

    mov     eax, 1
    int     0x80

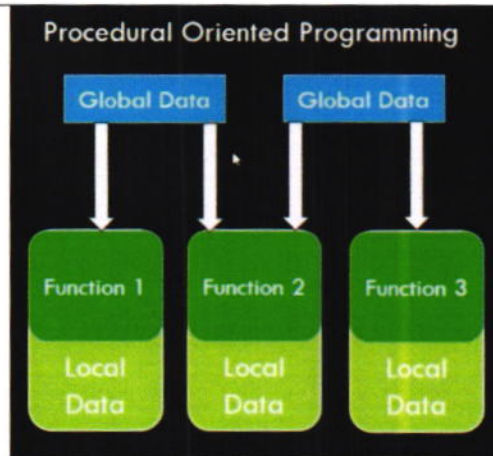
section      .data

msg         db  'Hello, world!', 0xa
len         equ $ - msg
```

Bahasa assembly memanfaatkan mnemonik, atau singkatan kata-kata dalam bahasa Inggris untuk membentuk perintah tertentu. Karenanya, bahasa ini perlu dikonversi lagi menjadi bahasa mesin.

Nah, proses penerjemahan ini dilakukan sebuah tools bernama assembler. Meski harus diterjemahkan dulu, performa program berbahasa assembly terbilang

	<p>cepat. Sebab, ia tidak membutuhkan banyak memori ketika dijalankan.</p> <p>b. Bahasa pemrograman tingkat menengah</p> <p>Bahasa tingkat menengah adalah jenis bahasa pemrograman yang menjadi penghubung antara bahasa tingkat rendah dan bahasa tingkat tinggi. Makanya, bahasa ini sering disebut dengan pseudo-language atau bahasa semu.</p> <p>Sebagai bahasa penghubung, bahasa tingkat menengah menggunakan perintah yang mendekati bahasa manusia, sehingga mudah dipahami. Di sisi lain, kompleksitasnya lumayan tinggi, mendekati bahasa assembly.</p> <p>Oleh sebab itu, bahasa tingkat menengah juga sering dipakai untuk membangun sistem operasi. Bahkan, ia biasa digunakan untuk membuat program yang dapat mengelola program lain, seperti terminal. Beberapa contoh dari bahasa tingkat menengah antara lain bahasa C, C++, dan C#.</p> <p>c. Bahasa pemrograman tingkat tinggi</p> <p>Bahasa tingkat tinggi adalah jenis bahasa pemrograman dengan perintah yang menyerupai bahasa manusia. Dengan kata lain, perintah tersebut sangat mudah ditulis, dibaca, dan dikelola untuk keperluan lebih lanjut. Maka dari itu, bahasa tingkat tinggi sangat direkomendasikan bagi Anda yang baru belajar pemrograman. Mengingat, tingkat kesulitannya relatif lebih rendah dibanding dua jenis bahasa yang telah disebutkan sebelumnya.</p> <p>Terlepas dari itu, bahasa tingkat tinggi sering dimanfaatkan untuk membangun <i>website</i> dan aplikasi. Bahkan, hampir semua <i>website</i> dan aplikasi yang Anda install dibuat menggunakan bahasa tingkat tinggi.</p> <p>Seperti yang sudah dibahas, bahasa tingkat tinggi perlu diterjemahkan dulu menjadi kode mesin. Tapi untungnya, setiap bahasa sudah mengikutsertakan Compiler atau Interpreternya masing-masing, sehingga nyaman digunakan.</p> <p>Nah, bahasa tingkat tinggi masih dibagi lagi menjadi dua jenis, yaitu:</p> <ol style="list-style-type: none"> 1) Pemrograman berorientasi prosedural (<i>procedural oriented programming</i>) <p>Pemrograman berorientasi prosedural adalah jenis bahasa pemrograman yang membagi perintah program menjadi bagian-bagian yang lebih kecil. Nantinya, bagian tersebut akan diurutkan secara sistematis. Jika sudah, komputer akan melaksanakan perintah sesuai dengan urutan yang sudah ditetapkan.</p>
--	---

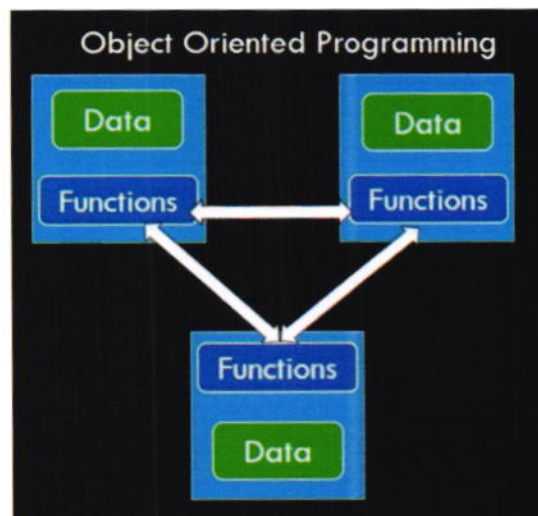


Kelebihannya, programmer akan lebih mudah memahami alur program. Namun kekurangannya, akan lebih sulit memodifikasi satu bagian program tanpa mempengaruhi bagian yang lainnya.

Nah, contoh bahasa pemrograman berorientasi prosedural adalah Pascal, Basic, dan FORTRAN.

2) Pemrograman berorientasi objek (*object oriented programming*)

Pemrograman berorientasi objek adalah jenis bahasa pemrograman yang mengelola perintah program menjadi objek-objek tertentu. Nah, masing-masing objek akan memiliki karakteristiknya tersendiri. Kemudian, komputer akan mengeksekusi perintah yang diberikan sesuai dengan karakteristik masing-masing objek.



Keuntungan dari pemrograman berorientasi objek adalah kemudahan dari segi modifikasi, tanpa menyebabkan error di bagian program lain. Kekurangannya, program jadi lebih sulit dipahami alurnya. Contoh bahasa pemrograman berorientasi objek antara lain **Java, PHP, Python, dan Ruby**

Setelah mengetahui apa itu bahasa pemrograman, fungsi, dan jenisnya, sekarang kita bahas mengenal beberapa contoh bahasa pemrograman.

Contoh bahasa pemrograman yang perlu diketahui:

Semua sistem dan program yang ada saat ini tercipta berkat bahasa pemrograman. Maka dari itu, Anda perlu mengenal beberapa diantaranya. Nah, 15+ contoh bahasa pemrograman adalah sebagai berikut:

- a) JavaScript;
- b) Python;
- c) SQL;
- d) Java;
- e) TypeScript22;
- f) C#;
- g) C++;
- h) PHP;
- i) C;
- j) Go;
- k) Kotlin;
- l) Ruby;
- m) Dart;
- n) Swift;
- o) R;
- p) HTML;
- q) CSS.

POKOK BAHASAN 3

CLIENT SERVER SEDERHANA

1. Pengenalan Local XAMPP

XAMPP adalah aplikasi yang memungkinkan Anda untuk membuat web server lokal untuk melakukan pengetesan *website*.

Banyak orang yang senang menggunakan XAMPP karena aplikasi ini ringan untuk dijalankan. Selain itu, XAMPP juga bisa digunakan di berbagai platform, mulai dari Apache, MySQL, PHP, hingga Perl.

Kelebihan XAMPP karena mampu bekerja secara offline dan dapat digunakan untuk mencoba fitur-fitur dalam *website*, melihat kinerja apakah sudah sesuai dengan yang diharapkan, mengevaluasi konten-konten yang akan dipasang pada *website* dan lain sebagainya sebelum *website* /aplikasi dihosting secara online. Jadi dengan kata lain, XAMPP merupakan prototipe penampilan *website* offline tanpa harus terhubung dengan internet dan dapat berdiri sendiri dalam sistem operasi atau localhost.

Menggunakan XAMPP untuk membuat *website* bukanlah hal yang sulit. Namun sebelum itu, perlu mengetahui bagian-bagian terpenting dalam XAMPP, sebagai berikut:

- a. htdocs adalah sebuah folder yang berada pada instalasi xampp. Fungsi dari folder htdocs ini untuk meletakkan berkas atau file *website*. File *website* dapat berupa skrip PHP, HTML, CSS, Javascript.
- b. phpMyadmin adalah bagian yang berfungsi untuk mengelola database MYSQL. Sehingga bila *website* menggunakan database, Anda perlu membuat dahulu akun database di phpMyadmin. Pengelolaan database juga dilakukan dari phpMyadmin. Untuk menggunakan phpMyadmin, Anda dapat membuka url berikut melalui browser <http://localhost/phpMyAdmin>.
- c. Kontrol Panel adalah bagian yang berfungsi untuk pengelolaan service XAMPP (apache, mysql, php, perl). Anda dapat melakukan stop ataupun restart service dari kontrol panel.

2. Prosedur Instalasi XAMPP

Kebutuhan instalasi XAMPP, untuk memulai langkah pada instalasi yang diperlukan: komputer dengan sistem operasi Windows dan File installer XAMPP.

Berikut adalah dua langkah mudah untuk menginstal XAMPP di komputer:

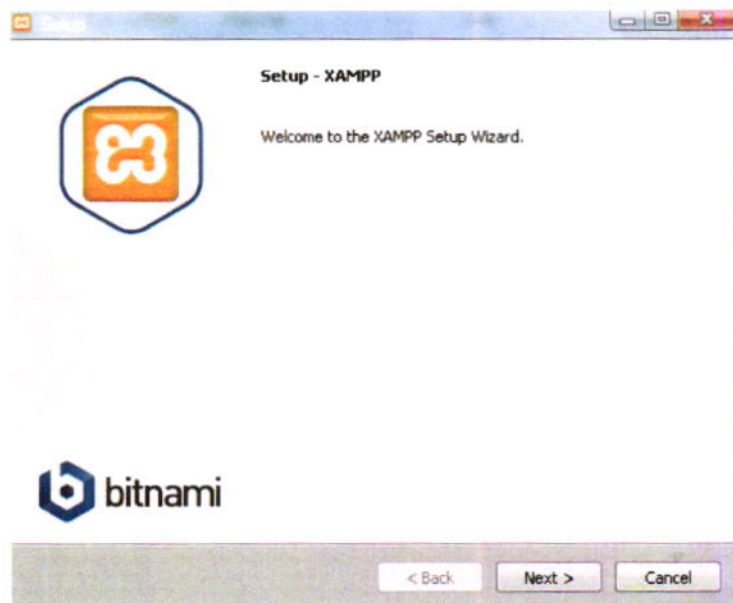
a. Download XAMPP

Download XAMPP terbaru melalui *website* Apache Friends berikut ini:

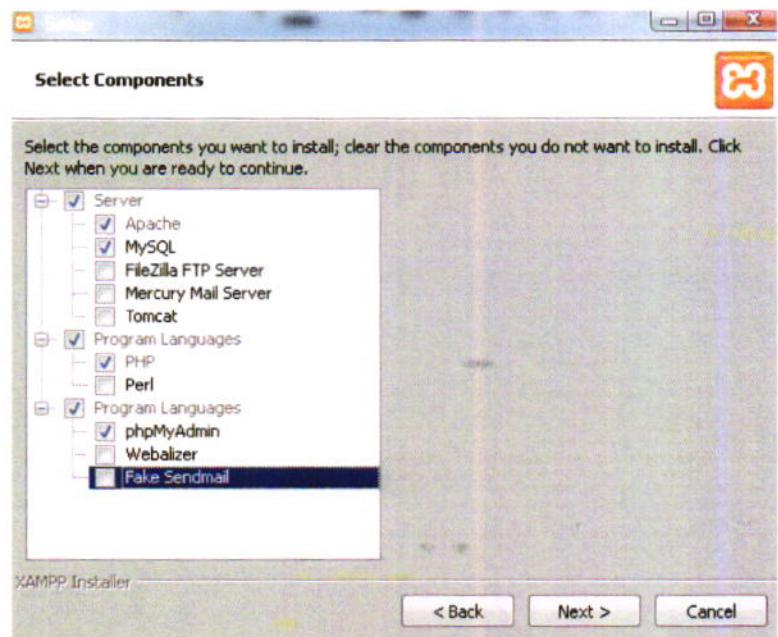


b. Instal XAMPP

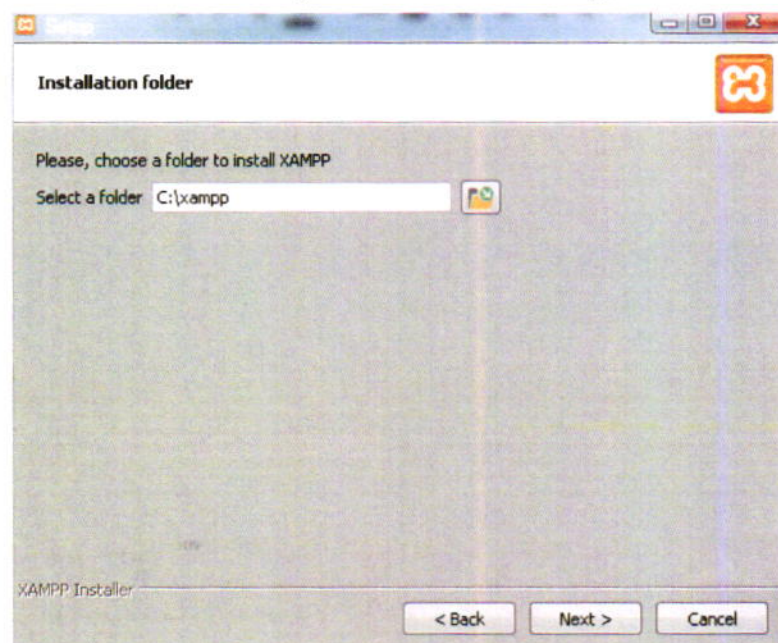
- 1) Lakukan instalasi setelah Anda selesai mengunduh. Selama proses instalasi mungkin Anda akan melihat pesan yang menanyakan apakah Anda yakin akan menginstalnya. Silakan tekan **Yes** untuk melanjutkan instalasi.
- 2) Klik tombol **Next**.



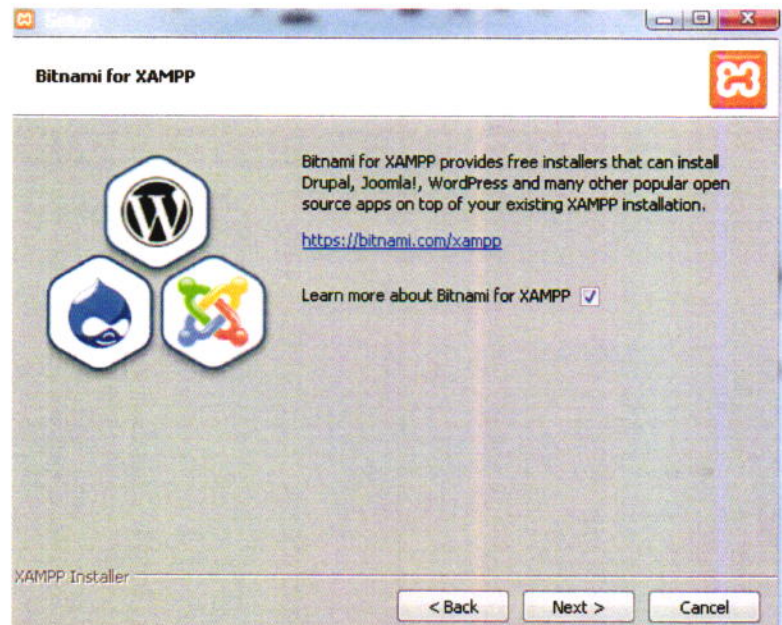
- 3) Pada tampilan selanjutnya akan muncul pilihan mengenai komponen mana dari XAMPP yang ingin dan tidak ingin Anda instal. Beberapa pilihan seperti Apache dan PHP adalah bagian penting untuk menjalankan *website* dan akan otomatis diinstal. Silakan centang MySQL dan phpMyAdmin, untuk pilihan lainnya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



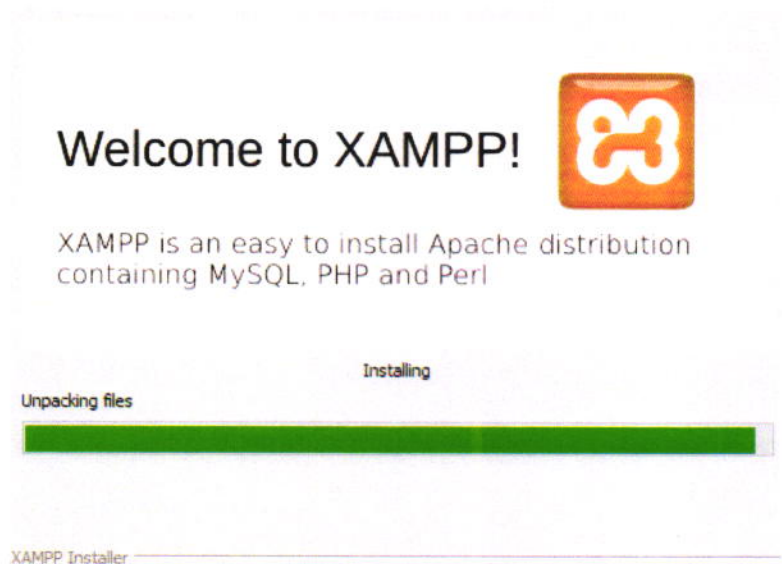
- 4) Berikutnya silakan pilih folder tujuan dimana XAMPP ingin Anda instal. Contohnya di direktori *C:\xampp*.



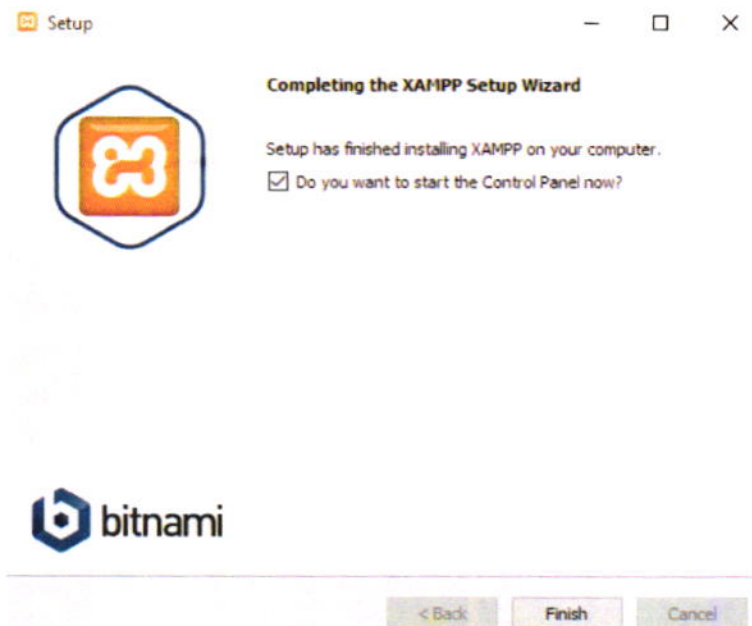
- 5) Pada halaman selanjutnya, akan ada pilihan apakah Anda ingin menginstal Bitnami untuk XAMPP, dimana nantinya dapat Anda gunakan untuk [install WordPress](https://bitnami.com/xampp), Drupal, dan Joomla secara otomatis.



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

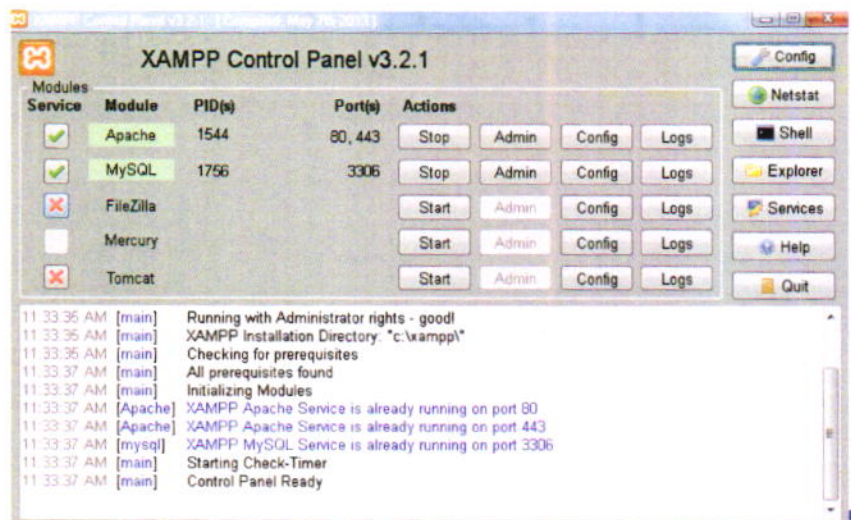


7) Setelah berhasil diinstal, akan muncul notifikasi untuk langsung menjalankan control panel. Silakan klik **Finish**



3. Operasionalisasi XAMPP

Untuk menjalankan 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:



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

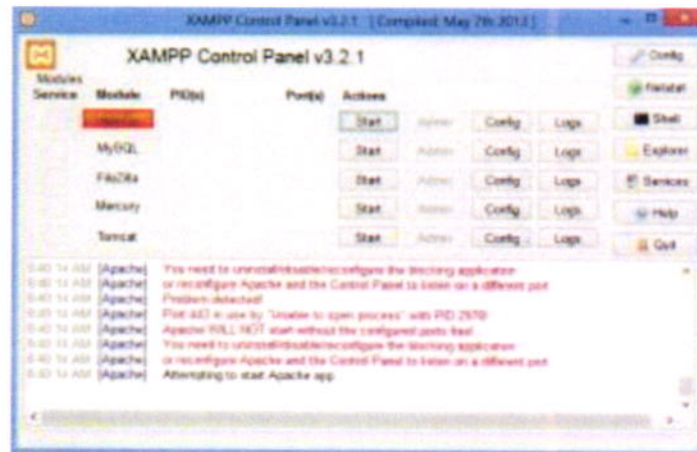
4. Mengatasi Error Pada XAMPP

a. Apache error

Secara default Apache akan aktif menggunakan port 80, sedangkan tool Mysql menggunakan port 3306, Untuk beberapa kasus, ada kalanya ketika tool-tool tersebut mencoba untuk diaktifkan, program tidak bisa running, bahkan muncul pesan informasi ERROR dengan tulisan berwarna merah di bawah ini:

*"Error: Apache shutdown unexpectedly
This may be due to a blocked port, missing dependencies, improper privileges, a crash, or a shutdown by another method. Press the log button to view error logs and check the window event Viewer for more clues. If you need more help, copy and post this entire log window on the forums"*

Kurang lebih akan terlihat akan seperti ini:



Jika masalahnya seperti itu, maka kemungkinan disebabkan oleh faktor-faktor berikut:

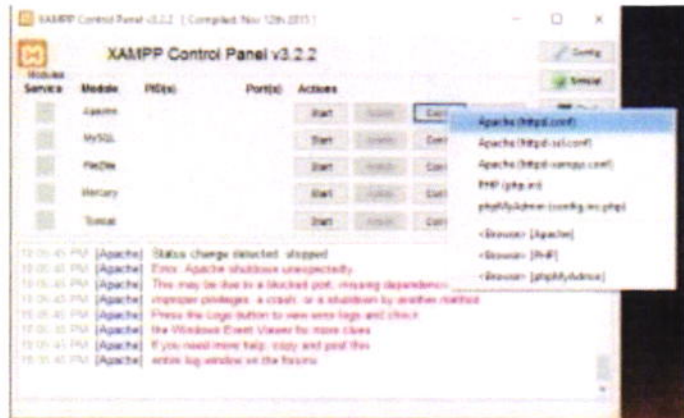
- 1) Adanya aplikasi yang menggunakan port 80 dan port 3306, sehingga terjadi crash. Ini bisa terjadi ketika anda menggunakan window 8.x atau ketika menggunakan window 10, atau jika di dalam komputer anda terinstal VM ware, atau aplikasi Skype sedang berjalan, Apache tidak bisa dijalankan karena adanya aplikasi lain yang lebih dulu berjalan menggunakan port yang sama yang digunakan oleh Apache dan juga Mysql.

Cara mengatasinya:

Jika anda mengalami hal serupa, apache pada xampp anda tidak bisa running atau berjalan di port 80, cara mengatasinya anda bisa mencoba mengganti default port

untuk Apache dan juga mysql dengan port yang belum digunakan, langkah-langkahnya adalah sebagai berikut:

- Aktifkan xampp control panel yang sudah anda instal.
- Klik button Config untuk Apache, kemudian pilih menu Apache (httpd.conf), seperti terlihat pada gambar di bawah ini:



- Pada dokument text yang terbuka, silahkan cari angka 80, kemudian ganti menjadi 8080, untuk text berikut:

Listen 80 menjadi *Listen* 8080
ServerName localhost:80 menjadi *ServerName localhost:8080*

- Simpan perubahan, dan silahkan anda coba restart kembali apache xampp anda. Jika masih tidak bisa berjalan, exit xampp control-panel, kemudian aktifkan kembali dengan klik kanan pilih “**open as administrator**” bisa jadi di block oleh firewall. Jika masih tidak bisa berjalan silahkan **uninstall xampp** di komputer anda, kemudian **download xampp versi terbaru**, dan **install ulang xampp** pada komputer anda.

Cara menjalankan port:

Karena port default dari apache sudah diganti **dari sebelumnya 80 menjadi port 8080**, ini berpengaruh juga ketika kita memanggil localhost, ketika port xampp masih menggunakan port bawaan (port 80), untuk memanggil halaman localhost cukup dengan membuka browser kemudian pada address bar tinggal di masukan localhost maka halaman localhost akan terbuka dan bisa diakses, namun ketika port sudah diganti menjadi 8080, maka ketika

anda ingin mengakses halaman localhost perintahnya sedikit berbeda, caranya:

- (1) Buka browser
- (2) Panggil di address bar dengan perintah **localhost:8080**

Dengan perintah di atas maka localhost akan kembali diakses, jika anda ingin menjalankan aplikasi web yang sudah anda buat sebelumnya, maka untuk memanggil halaman utama aplikasi web tersebut (*misalkan saja aplikasi web yang anda buat anda disimpan di folder latihan di htaccess*) gunakan perintah **localhost:8080/latihan**.

b. MySQL XAMPP Error

Banyak sekali kemungkinan terjadi error ketika anda start Mysql pada xampp control panel, sehingga Mysql sulit untuk di start dan tidak bisa running, salah satu error yang banyak sekali terjadi adalah seperti error di bawah ini:

```
12:19:12 PM [mysql] Attempting to start MySQL app...
12:19:12 PM [mysql] Status change detected: running
12:19:13 PM [mysql] Status change detected: stopped
12:19:13 PM [mysql] Error: MySQL shutdown unexpectedly.
12:19:13 PM [mysql] This may be due to a blocked port, missing
dependencies,
12:19:13 PM [mysql] improper privileges, a crash, or a
shutdown by another method
12:19:13 PM [mysql] Press the Logs button to view error logs
and check
12:19:13 PM [mysql] the Windows Event Viewer for more clues
12:19:13 PM [mysql] If you need more help, copy and post this
12:19:13 PM [mysql] entire log window on the forums
```

Cara mengatasi error di atas bisa dicoba cara-cara berikut:

- a. *Stop* semua server yang sedang berjalan di xampp, dll, kemudian exit.
- b. Masuk ke *folder* instalasi xampp, kemudian masuk ke folder data, misal C:\xampp\mysql\data
- c. Cari file **ibdata1**, silahkan ganti misal menjadi **ibdata1.bak**, atau **hapus**.
- d. *Restat* kembali xampp anda, kemudian coba kembali untuk start Mysql.
- e. Jika masih tidak bisa berjalan silahkan **uninstall xampp** di komputer anda, kemudian **download xampp versi terbaru**, dan **install ulang xampp** pada komputer anda

POKOK BAHASAN 4

CARA MEMBUAT APLIKASI SEDERHANA

1. Pembuatan Algoritma dan Flowchart Aplikasi

Membuat aplikasi sederhana untuk menghitung kecepatan

Apa itu kecepatan?

Sebelum lebih lanjut membicarakan tentang Algoritma, Flowchart dan Contoh Pemrograman Untuk Menghitung Kecepatan, kalian harus tahu pengertian dari kecepatan. Seperti yang saya kutip dari Wikipedia, tentang pengertian kecepatan.

Kecepatan besaran vektor yang menunjukkan seberapa cepat benda berpindah. Besar dari vektor ini disebut dengan kelajuan dan dinyatakan dalam satuan meter per sekon (m/s).

Rumus kecepatan:

Sebelum masuk kepada pembahasan Algoritma, kalian harus tau rumus untuk menentukan kecepatan rata-rata. Adapun rumus untuk menentukan kecepatan adalah sebagai berikut:

$$v = s/t$$

Keterangan:

v = kecepatan (km/jam)

s = jarak (km)

t = waktu tempuh (jam)

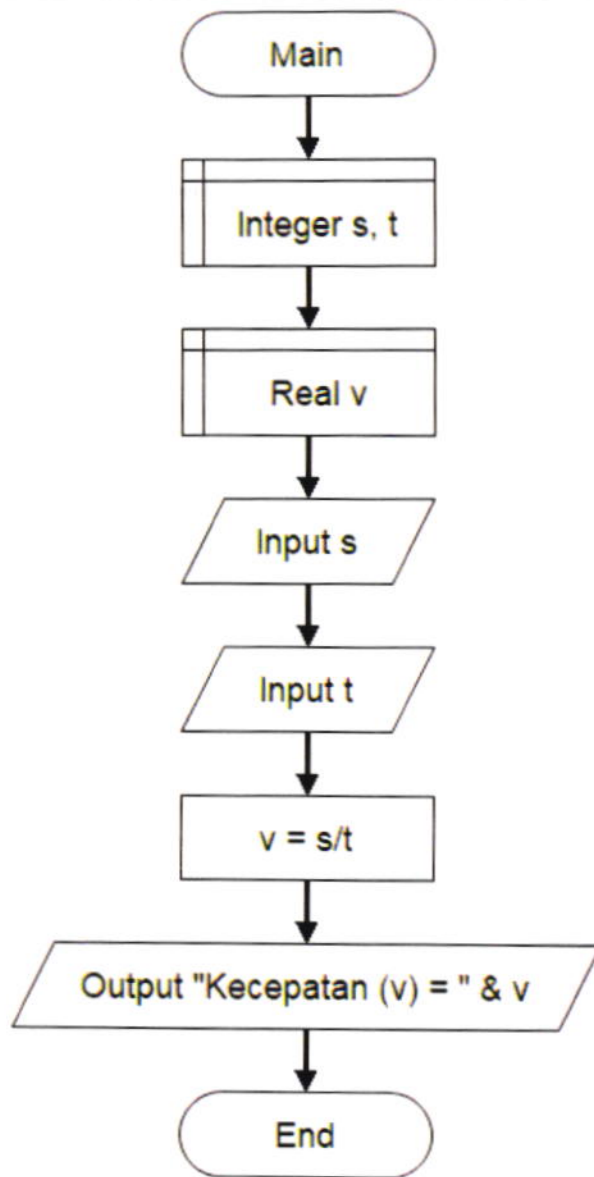
Algoritma Menghitung atau Mencari Nilai Kecepatan

Setelah kalian tau rumus untuk mencari kecepatan, sekarang tinggal kita susun algoritma untuk menghitung atau mencari kecepatan. Adapun Algoritma untuk menghitung atau mencari kecepatan adalah sebagai berikut:

- a. *Mulai*
- b. *Deklarasi variabel kecepatan (v), jarak (s) dan waktu (t)*
- c. *Input jarak (s) dan waktu (t)*
- d. *Hitung kecepatan (v) dengan rumus $v = s/t$*
- e. *Tampilkan output kecepatan*
- f. *Selesai*

Flowchart menghitung atau mencari nilai kecepatan

Dari algoritma untuk menghitung atau mencari kecepatan yang telah dibuat di atas, selanjutnya kita akan membuat flowchartnya. Flowchart untuk menghitung atau mencari kecepatan dapat digambarkan sebagai berikut:



Flowchart untuk menghitung atau mencari kecepatan

2. Pembuatan Aplikasi Menghitung Kecepatan

Berikut ini adalah Code Program untuk Menghitung kecepatan dalam bahasa pemrograman PHP:

```

1 <form method="post" action="">
2   Input jarak (km) : <input type="text" name="s"><br>
3   Input waktu (jam) : <input type="text" name="t"><br>
4   <input type="submit" name="submit" value="Hitung">
5 </form>
6
7 <?php
8 if(isset($_POST['submit'])){
9     $s = $_POST['s'];
10    $t = $_POST['t'];
11
12    $v = $s / $t;
13    echo "Kecepatan (v) = " . $v . " km/jam";
14 }
15 ?>
16

```

Penulisan dalam Bahasa PHP

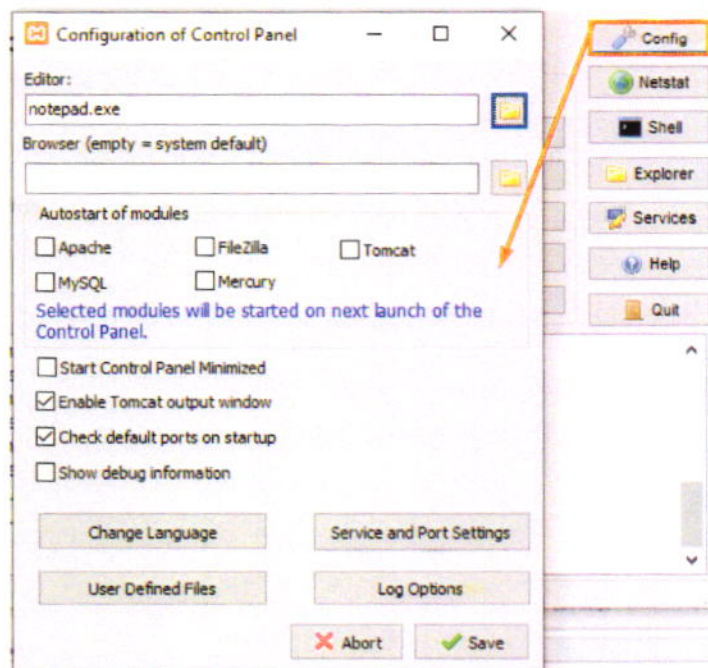
Output

3. Konfigurasi dan Menjalankan Xampp (Server Side)

a. Config

Pada menu Config, Anda bisa melakukan beberapa konfigurasi dasar untuk penggunaan XAMPP seperti mengganti:

- 1) **Editor**: yang akan digunakan untuk mengubah default text editor.
- 2) **Browser**: browser default yang akan digunakan untuk akses web server.



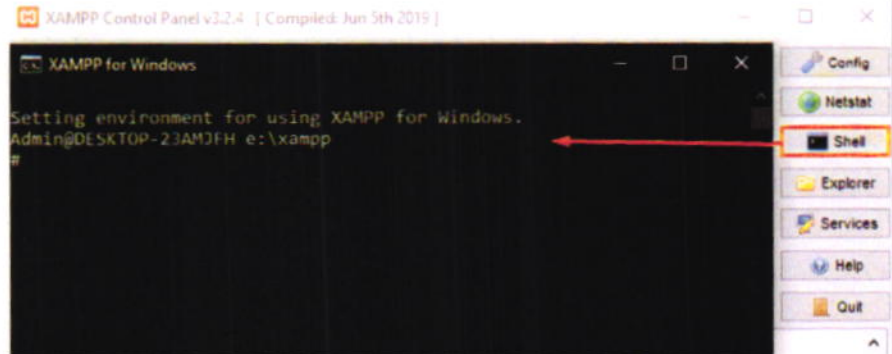
b. Netstat

Netstat sering digunakan untuk memastikan apakah ada program/ aplikasi lain yang menggunakan port default XAMPP.



c. Shell

Menu shell pada XAMPP, digunakan untuk menampilkan command prompt pada Windows. Fungsinya untuk melakukan konfigurasi web server atau memperbaiki error pada web server.



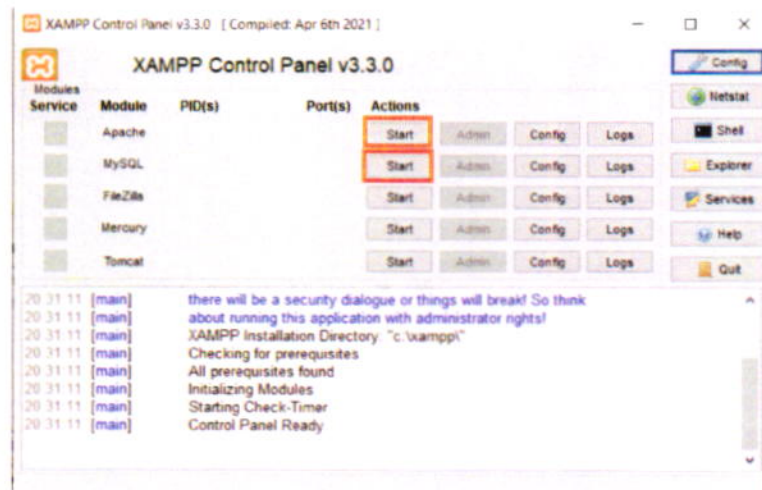
d. Explorer

Explorer merupakan tombol shortcut untuk menuju ke lokasi instalasi XAMPP. Ketika Anda klik icon **Explorer** maka akan diarahkan ke lokasi file instalasi XAMPP

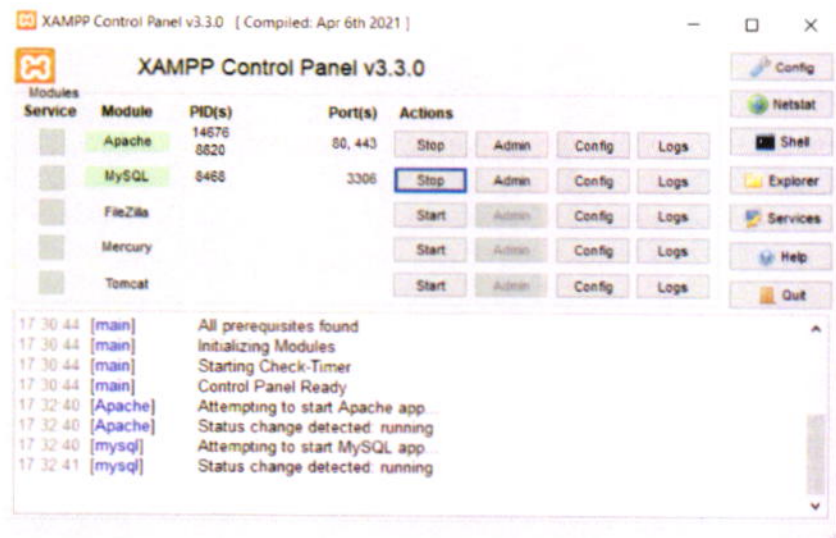


Jika XAMPP sudah terinstall, bukanlah aplikasi tersebut. Anda akan melihat lima module yang terinstall, yaitu: **Apache, MySQL, FileZilla, Mercury** dan **Tomcat**.

Biasanya, untuk membuat sebuah *website*, Anda hanya perlu menjalankan dua module saja, Apache sebagai web server dan MySQL untuk database.



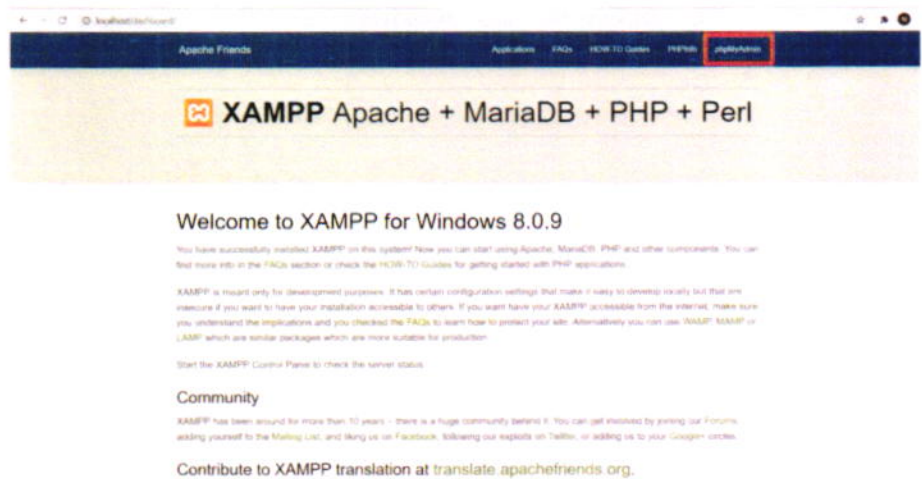
Klik tombol Start pada Apache dan MySQL agar Anda dapat mengakses localhost/phpMyAdmin untuk membuat database. Jika sudah berhasil dijalankan, tampilan XAMPP akan berubah seperti gambar di bawah ini:



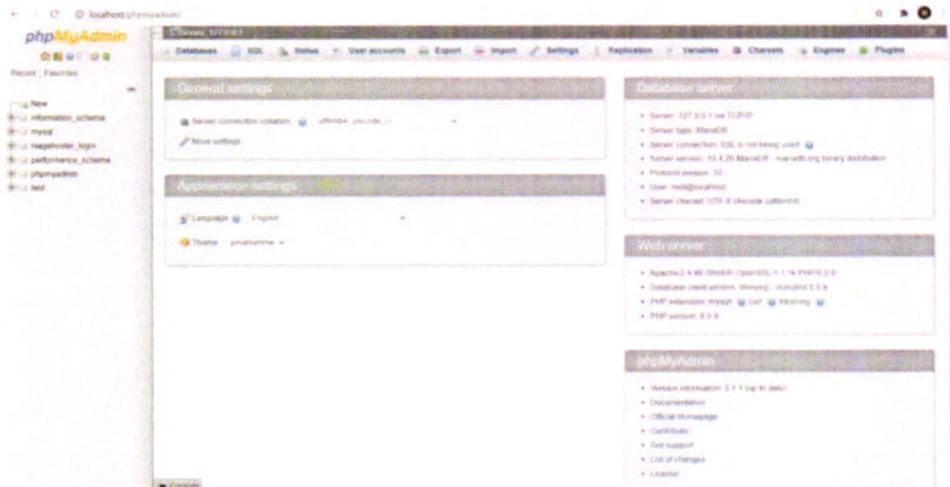
4. Setting dan Koneksi DB PHP Myadmin

a. Membuka akses PHPMyAdmin

Anda bisa mengakses phpMyAdmin dengan dua cara. **Pertama**, melalui dashboard localhost. Caranya, ketik **localhost** pada kolom pencarian browser, lalu tekan **Enter**. Kemudian, pilih **phpMyAdmin** seperti yang ditunjukkan pada gambar berikut ini:



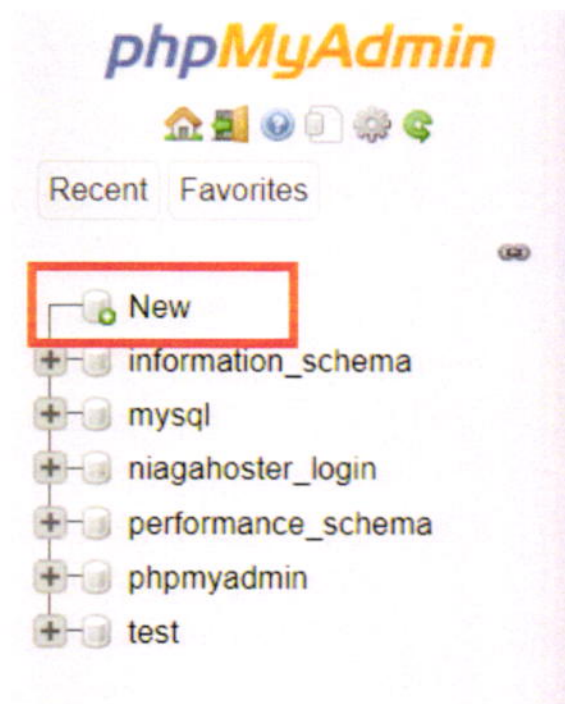
Kedua, langsung mengakses **localhost /phpMyAdmin** pada kolom pencarian web browser. Jangan lupa tekan **Enter**. Maka akan muncul tampilan phpMyAdmin seperti ini:



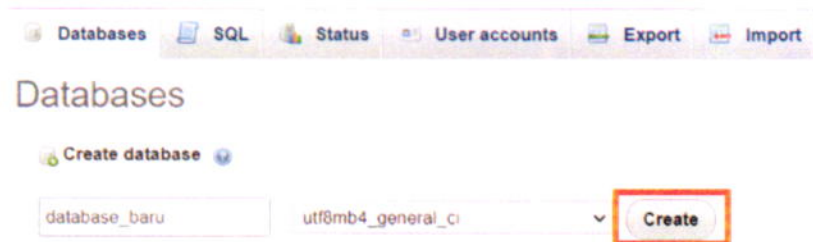
Pada halaman inilah, Anda dapat melakukan pengelolaan **database** untuk **website** Anda. Misalnya, membuat database, mengedit, menghapus, bahkan melakukan query data.

b. Membuat database di phpMyAdmin

Pada bagian kiri localhost/phpMyAdmin, silahkan klik **New** untuk membuat database baru.



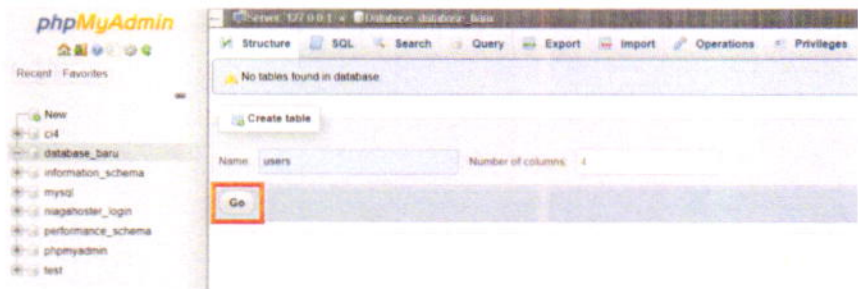
Masukkan nama database pada kolom yang tersedia. Pada tutorial ini, kami menggunakan **database_baru** sebagai nama database yang akan dibuat. Jika sudah diberi nama, klik **Create**.



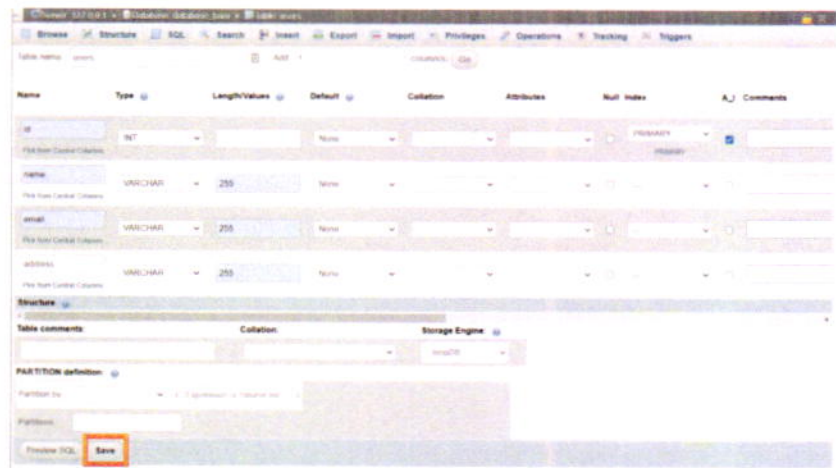
Nantinya, database baru yang Anda buat akan tampil di bagian kiri halaman phpMyAdmin. Sekarang, Anda sudah bisa menambahkan tabel sebagai tempat penyimpanan data pada database baru.

c. Membuat tabel pada database localhost/phpMyAdmin

Kali ini, mari membuat tabel dengan empat kolom untuk data pengguna dengan nama tabel **user s**. Kemudian, klik tombol **Go**.



Di bagian ini, Anda harus memasukkan nama kolom dan tipe datanya. Untuk tabel *user s*, kami membuat kolom **id**, **name**, **email**, dan **address**. Jika sudah terisi seperti gambar di bawah ini, klik tombol **Save**



Sekarang, Anda sudah memiliki database dengan tabel *user s* yang siap dipakai dan dihubungkan dengan *website* Anda



d. Menjalankan *Website* dengan XAMPP

Untuk menjalankan *website* dengan XAMPP, Anda harus menempatkan project *website* Anda di dalam direktori **C:\xampp\htdocs**. Kalau ditempatkan di luar folder *htdocs*, project *website* Anda tidak akan bisa diakses di localhost.

Misalnya, Anda memiliki project *website* yang disimpan di dalam folder *website_baru*. Maka folder *website_baru* harus ditempatkan di dalam folder *htdocs* seperti ini:

Name	Date modified	Type	Size
ci_phpmailer	16/08/2021 10:01	File folder	
ci4	07/09/2021 8:20	File folder	
dashboard	13/08/2021 6:55	File folder	
img	13/08/2021 6:55	File folder	
niagahoster_login	13/08/2021 13:55	File folder	
tutor_composer	01/09/2021 12:46	File folder	
webalizer	13/08/2021 6:55	File folder	
website_baru	23/09/2021 8:02	File folder	
xampp	13/08/2021 6:55	File folder	
applications	27/08/2019 21:02	Microsoft Edge HT...	
bitnami	27/08/2019 21:02	Cascading Style Sh...	
favicon	16/07/2015 22:32	Icon	
index.php	16/07/2015 22:32	PHP File	

Dengan begitu, Anda bisa menjalankan project *website* dengan mengakses **localhost/website_baru** pada web browser.

e. Menghubungkan *website* dengan database

Langkah berikutnya dari cara membuat database di phpmyadmin adalah menghubungkan *website* dengan database yang sudah Anda buat di localhost/phpMyAdmin. Dengan begitu, pengelolaan database dapat dilakukan langsung pada *website*.

Anda dapat membuat koneksi ke database dengan melakukan sinkronisasi database. Biasanya, sinkronisasi ini dilakukan menggunakan script khusus.

Caranya, buatlah script php dengan nama koneksi.php, config.php, database.php atau nama lainnya pada project *website* Anda. Kemudian, salinlah kode berikut ini:

```

1      <?php
2      $server = "localhost";
3      $user = "root";
4      $pass = "";
5      $database = "database_baru";
6
6      $conn = mysqli_connect($server, $user , $pass,
7      $database);
8
9      if (!$conn) {
10         die("<script>alert('Gagal tersambung dengan
11         database.')</script>");
12     }
13
14     ?>

```

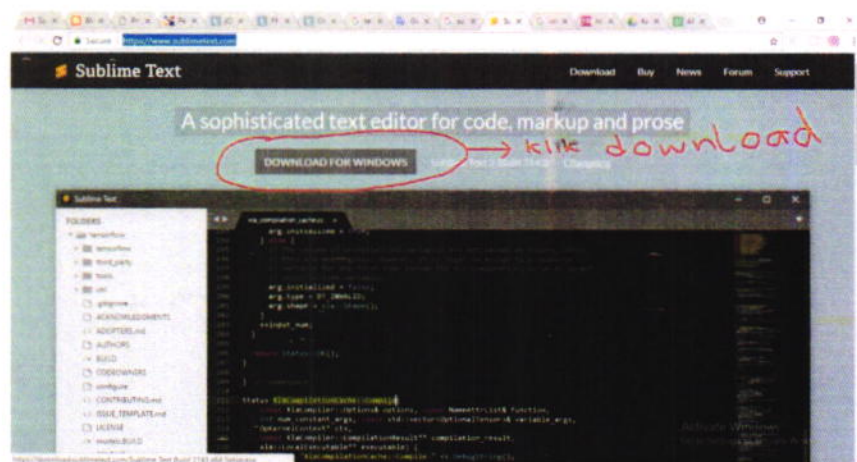
Pada bagian server, tuliskan **localhost**. Kemudian, masukkan **root** sebagai *user*. Sedangkan untuk password, kosongkan saja. Jangan lupa tulis nama database yang Anda buat pada baris kode **\$database**.

Simpan script PHP tersebut. Coba lakukan pengecekan. Seharusnya *website* Anda sudah terhubung dengan database di phpMyAdmin.

5. Instalasi Sublime Text

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

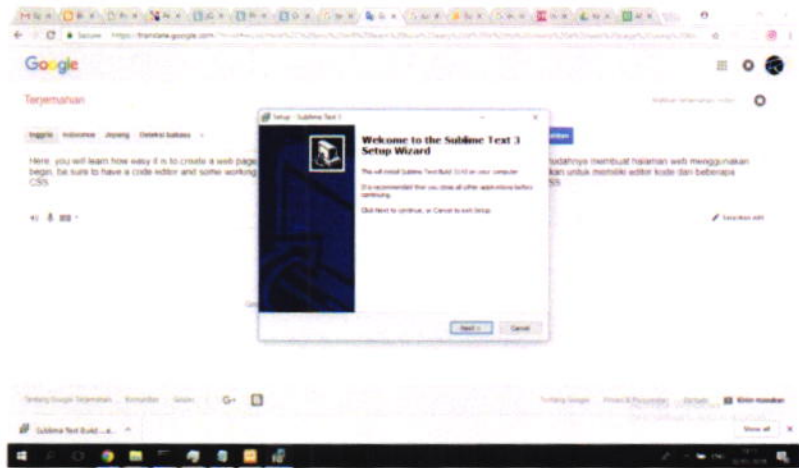
- a. Step 1 download terlebih dahulu file **.exe** di situs resminya sublime



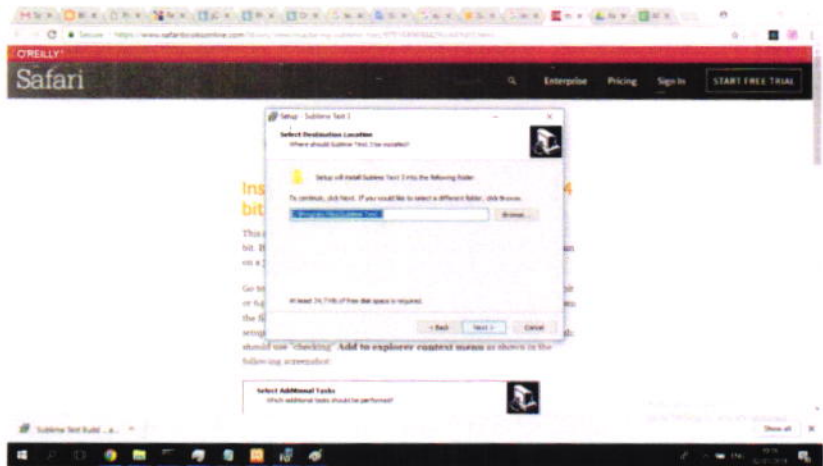
- b. Setelah download selesai maka double klik file **.exe** pada hasil download tadi



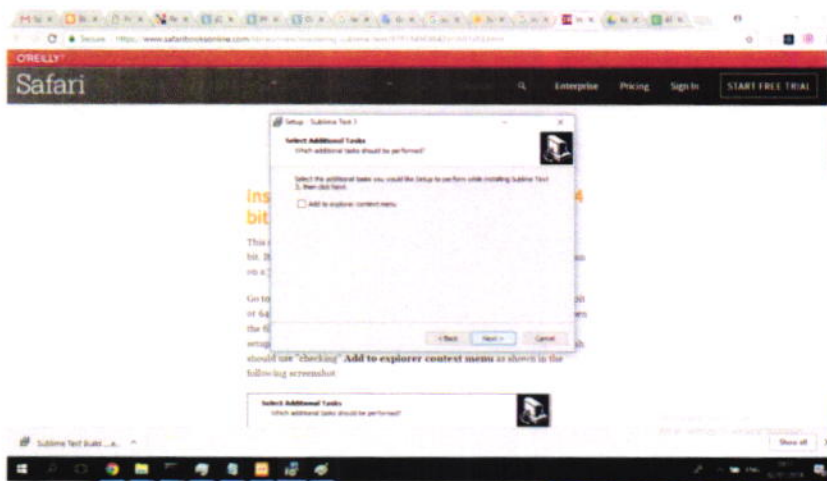
c. Kemudian klik **next**



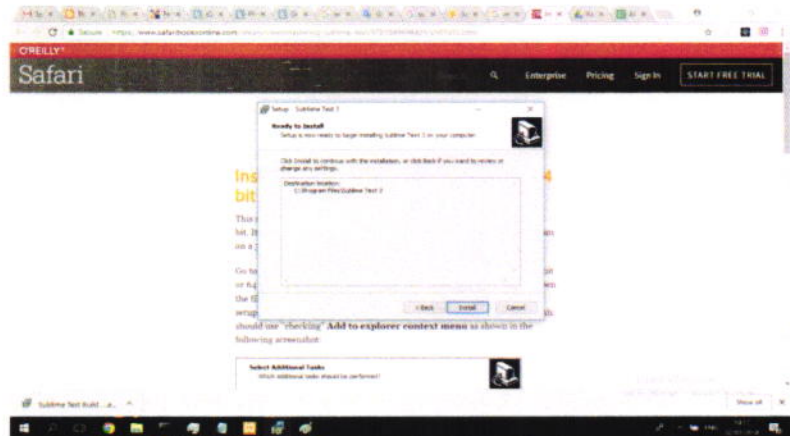
d. Kemudian klik **next**



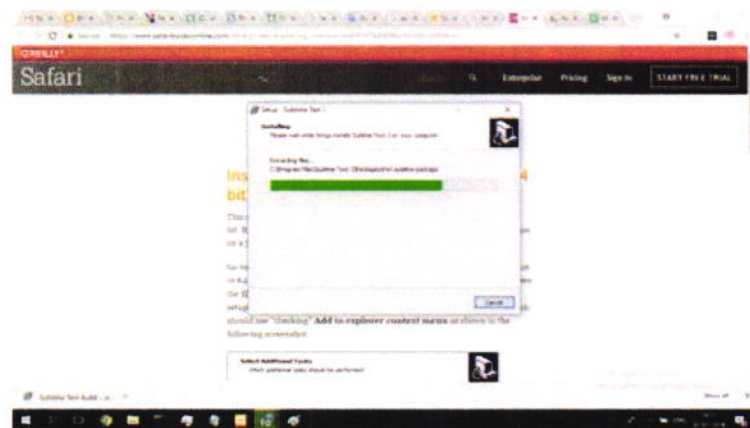
e. Kemudian klik **next**



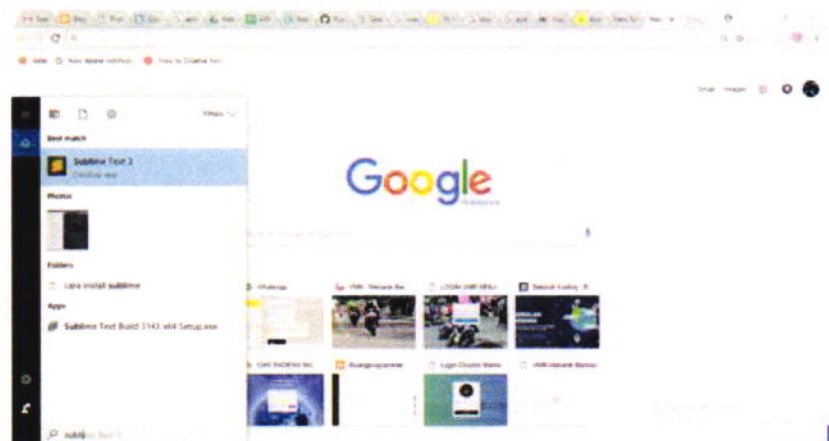
f. Kemudian klik **next**



g. Kemudian tunggu sampai proses install selesai



h. Setelah proses instal selesai kemudian bukan pada di menu windows 10 seperti gambar dibawah ini



i. Sublime siap untuk membuat aplikasi

6. Pembuatan Web Sederhana dengan Sublime Text

a. Sintaks dasar PHP

Secara umum, [sintaks dasar PHP](#) dibuka dengan tag `<?php` dan diakhiri dengan tag `?>` seperti berikut:

```
<?php [code]; ?>
```

Nah, bagian `;` bisa Anda isi dengan perintah apapun. Misalnya, kami ingin menampilkan teks “Selamat Datang di Niagahoster”, maka sintaksnya adalah:

```
<?php echo “Selamat Datang di Niagahoster”; ?>
```

Berikut hasilnya ketika ditampilkan di browser:



Jangan lupa, simpan file PHP tersebut dengan ekstensi **.php**.

b. Struktur dasar *website*

Sebelum membuat *website*, penting untuk paham struktur dasarnya terlebih dahulu. Secara umum, struktur dasar *website* adalah sebagai berikut:

- 1) **Header** - Terletak di bagian paling atas *website* dan biasanya memuat informasi singkat seputar *website*, seperti logo, judul, dan tagline *website*.
- 2) **Menu** - Bagian *website* yang berfungsi sebagai jalan pintas menuju konten-konten yang ada di *website*, sering juga disebut dengan navigasi.
- 3) **Isi/Konten** - Bagian utama pada *website* yang berisi artikel atau informasi yang ingin disampaikan pemilik *website* kepada pengunjung.
- 4) **Footer** - Terletak di bagian paling bawah *website* dan biasanya terdiri dari informasi pemilik *website* dan copyright

Ini adalah contoh penerapan struktur *website* di atas:



c. Cara membuat *website* dengan PHP dari awal

Untuk membuat *website*, dasar [HTML](#) dan [CSS](#) perlu Anda kuasai untuk membangun struktur yang baik dan tampilan yang menarik.

Cara membuat *website* dengan PHP dari awal bisa dilakukan dengan langkah berikut:

- 1) Mempersiapkan software yang digunakan
- 2) Membuat file `index.php`
- 3) Membuat file `style.css`
- 4) Membuat file `home.php`
- 5) Membuat file `about.php`
- 6) Membuat file `contact.php`
- 7) Mencoba *website* di browser

Cara membuat *website* PHP dengan langkah di atas:

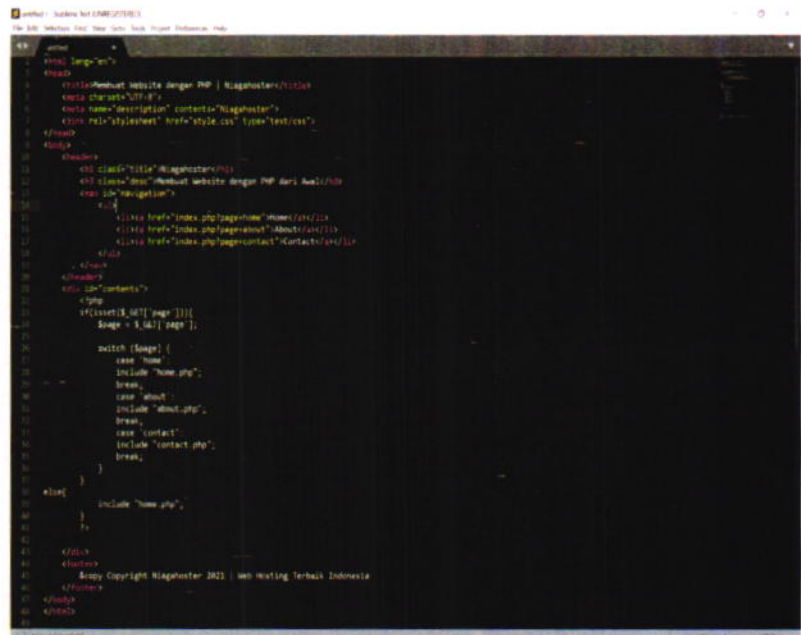
1) Langkah 1: persiapan

Ini adalah berbagai tools yang perlu Anda persiapkan sebelum membuat *website* PHP

- a) **PHP Editor** - Digunakan untuk membuat file *website* dan menulis kodenya. Kali ini kami menggunakan [Sublime Text](#).
- b) **Web Browser** - Berguna untuk mengecek tampilan *website*. Kami menggunakan [Google Chrome](#) pada tutorial kali ini.
- c) **Web Server** - Sebagai tempat untuk menyimpan folder *website*. Disini kami memakai [XAMPP](#) lalu membuat folder bernama ***website php*** di `c:\xampp\htdocs`.

2) Langkah 2: membuat file index.php

Bukalah Sublime Text, lalu klik **File > New File**. Kemudian, masukkan source code di contoh berikut atau source code Anda sendiri sesuai keinginan:



```

1 <?php
2 <html lang="en">
3 <head>
4 <title>Membuat website dengan PHP | NlagaHoster</title>
5 <meta charset="UTF-8">
6 <meta name="description" content="NlagaHoster">
7 <link rel="stylesheet" href="style.css" type="text/css">
8 </head>
9 <body>
10 <div class="container">
11 <div class="title">NlagaHoster</div>
12 <div class="desc">Membuat website dengan PHP dari A-Z</div>
13 <div class="navigation">
14 <ul>
15 <li><a href="index.php/pagehome">Home</li>
16 <li><a href="index.php/pageabout">About</li>
17 <li><a href="index.php/pagecontact">Contact</li>
18 </ul>
19 </div>
20 <div class="content">
21 <div id="content">
22 <h2><?php
23 <isset($_GET['page'])
24 <page = $_GET['page']>
25 </h2>
26 <div>
27 <switch (<?php
28 <page>
29 </switch>
30 <div>
31 <case "home">
32 <include "home.php">
33 <break;
34 <case "about">
35 <include "about.php">
36 <break;
37 <case "contact">
38 <include "contact.php">
39 <break;
40 </div>
41 </div>
42 </div>
43 </div>
44 </div>
45 </div>
46 </div>
47 </div>
48 </div>
49 </div>
50 </div>
51 </div>
52 </div>
53 </div>
54 </div>
55 </div>
56 </div>
57 </div>
58 </div>
59 </div>
60 </div>
61 </div>
62 </div>
63 </div>
64 </div>
65 </div>
66 </div>
67 </div>
68 </div>
69 </div>
70 </div>
71 </div>
72 </div>
73 </div>
74 </div>
75 </div>
76 </div>
77 </div>
78 </div>
79 </div>
80 </div>
81 </div>
82 </div>
83 </div>
84 </div>
85 </div>
86 </div>
87 </div>
88 </div>
89 </div>
90 </div>
91 </div>
92 </div>
93 </div>
94 </div>
95 </div>
96 </div>
97 </div>
98 </div>
99 </div>
100 </div>

```

Jika sudah, simpan file dengan nama **index.php** pada folder **website php**. Penjelasan kode di atas sebagai berikut:

- a href=" ... "** digunakan untuk memasukkan menu navigasi.
- \$_GET ['...']** digunakan untuk memanggil method GET.

Pada PHP, method GET berfungsi untuk mengambil data dari variabel ke halaman lain. Untuk itu, diperlukan variabel baru bernama page.

Method GET nantinya akan memanggil isian data dari variabel page. Kemudian, isian data tersebut akan ditampilkan sesuai dengan halaman menu yang dituju

- switch (...)** {case '...'} digunakan untuk berpindah-pindah halaman.
- include "home.php";** digunakan untuk mengatur file home.php sebagai halaman default.

3) Langkah 3: membuat file style.css

Cara membuat *website* PHP tidak lengkap tanpa langkah menambahkan file CSS. Tujuannya, agar tampilan *website* semakin menarik. Untuk itu, buat file


```
<div class="page">
```

```
<h2>Tentang Niagahoster</h2>
```

```
<p>Telah dipercaya lebih dari 500.000 klien yang terdaftar dari seluruh Indonesia, Niagahoster selalu berkomitmen menyajikan teknologi terbaik untuk kesuksesan online Anda. Bergabunglah dan temukan kemudahan menciptakan website impian Anda bersama Niagahoster!</p>
```

```
</div>
```

6) Langkah 6: membuat file `contact.php`

Terakhir, buat halaman Contact dengan memilih menu **File** lalu klik **New File**. Kemudian salin source code berikut:

```
<div class="page">
```

```
<h2>Hubungi Kami</h2>
```

```
<p>Telp: 0274-2885822
```

```
<br>WA: 0895395186038
```

```
<br>Senin - Minggu
```

```
24 Jam Non Stop</p>
```

```
<p>Jl. Palagan Tentara Pelajar
```

```
No 81 Jongkang, Sariharjo, Ngaglik, Sleman
```

```
Daerah Istimewa Yogyakarta
```

```
55581</p>
```

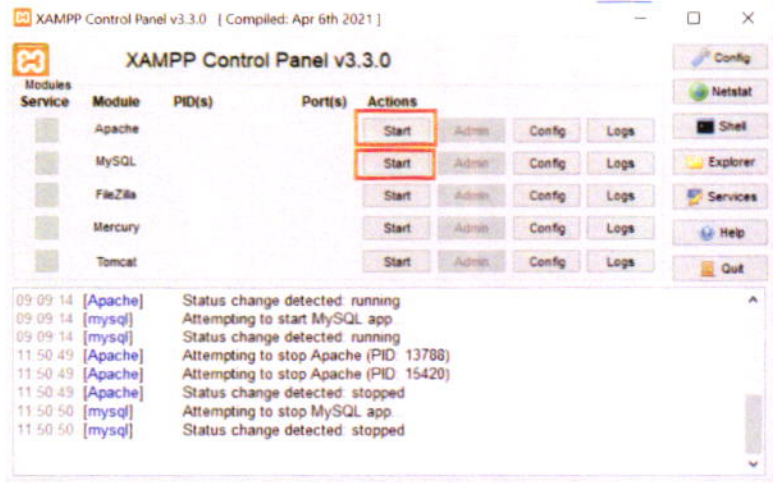
```
</div>
```

Jika sudah, simpan file dengan nama **contact.php**.

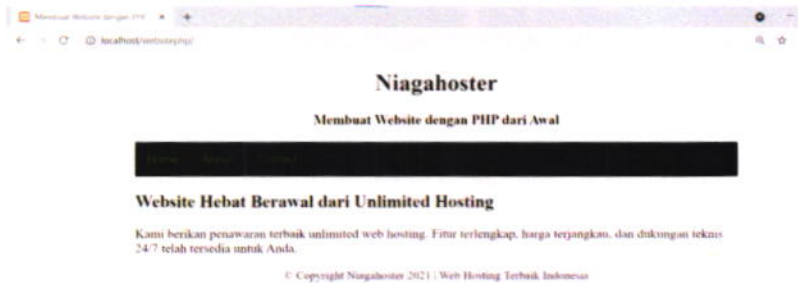
7) Langkah 7: mencoba *website* di browser

Nah, semua proses coding telah selesai dan *website* sudah siap. Tapi, apakah *website* dapat diakses dengan lancar? Mari lakukan pengecekan.

Caranya, buka XAMPP Control Panel, lalu klik **Start** pada modul **Apache** dan **MySQL**.

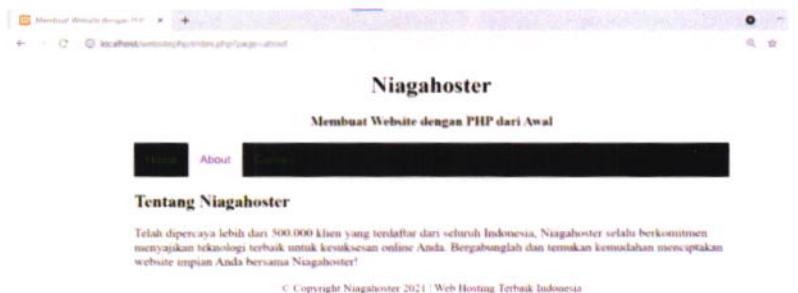


Kemudian buka browser, lalu ketikkan **localhost/website php**. Jika berhasil, begini tampilannya:

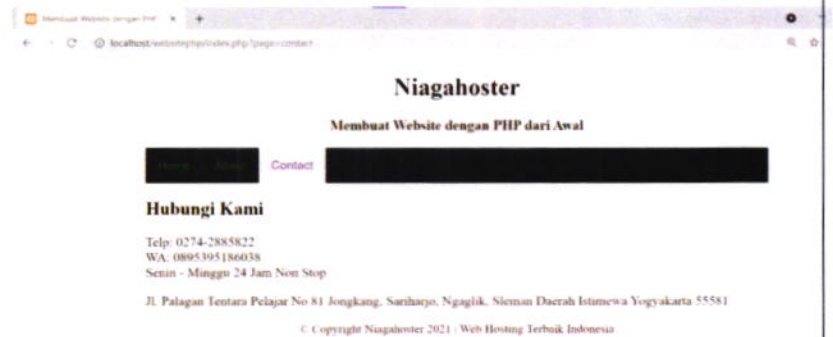


Secara default, *website* akan menampilkan halaman Home seperti skrip yang sudah Anda tulis pada file `index.php`.

Sekarang coba buka halaman About dengan cara klik menu navigasi **About**. Begini kira-kira hasilnya:



Sedangkan untuk halaman Contact, tampilan yang dihasilkan seperti berikut:



Anda telah berhasil menyelesaikan cara membuat *website* dengan PHP dan CSS!

Kalau *coding* Anda benar, *website* akan berjalan lancar. Anda bisa terus berkreasi seperti mengganti header dan footer, menambahkan menu baru, mengganti halaman default, sampai mempercantik tampilan melalui CSS.

7. Perawatan Aplikasi Web yang Dibuat

Kebutuhan setiap orang dalam membuat *website* memang berbeda-beda. Namun meski kebutuhan pemilik *website* itu variatif, semuanya sama-sama membutuhkan pemeliharaan.

a. Perawatan mingguan

Pemeliharaan *website* sebaiknya dilakukan dalam periode tertentu. Jangan setahun sekali, atau bahkan tidak pernah sama sekali. Setidaknya, lakukanlah setiap minggu.

Hal itu dilakukan untuk mencegah 'tabungan masalah' yang bisa menumpuk karena Anda tidak pernah memonitor *website* Anda.

- 1) Pastikan seluruh halaman dan tautan masih menjalankan fungsinya
- 2) Pastikan semua halaman *website* Anda menampilkan informasi yang seharusnya

Berikut cara memeriksanya:

- a) Instal plugin [Broken Link Checker](#) untuk menemukan tautan halaman yang rusak.
- b) Lakukan [setting plugin](#) Broken Link Checker.
- c) Perbaiki atau hapus URL yang rusak.

3) Periksa seluruh update komponen *website*

Hal penting yang sering terlupakan oleh pemilik *website* adalah memperbarui seluruh komponen *website*. Mereka, bahkan mungkin Anda, terkadang **mengira *website* akan baik-baik saja walau tanpa di-update.**

Jika Anda berpikir *website* tetap aman meski tanpa diperbarui, tidak usah heran jika tiba-tiba malware menyerang. Bukan cuma malware saja, tapi juga berbagai gangguan yang merugikan *website* Anda.



Mungkin, efek dari pembaruan tersebut tidak terlihat dan barangkali tidak terasa bagi Anda. Sehingga kesannya, itu hanya remeh temeh belaka.

Namun yang namanya **pembaruan, meski kecil, akan membuat seluruh komponen *website* bekerja lebih oke.** Karena itu, rajin-rajinlah [memeriksa notifikasi update](#) Anda

4) Hapus seluruh komentar spam

Setahu Anda, mungkin komentar spam hanyalah komentar-komentar yang tidak relevan dengan isi konten. Alhasil karena menganggapnya sepele, Anda memilih membiarkan mereka mengisi halaman komentar Anda.

Sayangnya, komentar spam tidak sesederhana itu. Mengabaikannya adalah cara yang salah. Orang jahat mana pun bisa bersembunyi di balik komentar spam untuk menggiring pengunjung *website* menuju halaman berbahaya.

Gampang saja, mereka tinggal meletakkan pancingan link pada kolom komentar Anda. Menunggu kail itu disambut pengunjung atau bahkan Anda sendiri.

Selain itu, membiarkan komentar spam terpampang pada *website* Anda akan memberi kesan ke pengunjung bahwa Anda terlalu cuek. Jadi, rutinlah menghapus seluruh komentar spam pada *website* Anda.

b. Perawatan bulanan

Pasti Anda memiliki tujuan tertentu dari membuat sebuah *website*. Entah itu melakukan penjualan, memiliki blog yang diakses banyak orang, atau tujuan lainnya. Namun, mungkin ada saatnya strategi untuk meningkatkan traffic, penjualan, atau apapun itu kurang berhasil.

Nah, perawatan *website* secara bulanan perlu dilakukan untuk membantu Anda mendapatkan hasil analisis yang cukup merepresentasikan kenyataan. Tidak diambil terlalu cepat, namun juga tidak terlalu lambat.

Agar semakin jelas, yuk simak 3 cara maintenance *website* secara bulanan berikut ini:

1) Tinjau performa SEO

SEO adalah kunci jika ingin *website* Anda mudah ditemukan di Google. Orang-orang lebih mempercayai hasil pencarian yang tampil pada halaman pertama Search Engine Results Page (SERP).

Memiliki **performansi SEO yang bagus tidak hanya melejitkan ranking *website*, tetapi juga meningkatkan traffic**. Semakin suatu *website* berada di peringkat teratas, kemungkinan untuk di-klik juga semakin besar.



Sudahkah Anda mengecek traffic *website* Anda? Puaskah dengan hasilnya?

Sebaiknya, jangan cepat berpuas diri atau mungkin malah putus asa karena traffic tidak kunjung melonjak. Lakukan manuver dengan menerapkan tips dan trik SEO yang terjamin ampuh.

2) Review konten dalam *website*

Salah satu hal yang Anda pertimbangkan dalam membuat konten adalah tren. Sayangnya, tidak ada tren yang berlangsung abadi. Tren juga bisa berganti dengan cepat.

Terkadang, saat membuat konten, Anda terjebak dalam zona nyaman. Rasanya cocok dan nyaman dengan topik yang sedang trending sekarang. Padahal, nanti atau bahkan besok, topik itu bisa jadi tidak dilirik lagi.



Padahal, **konten yang relevan dengan kebutuhan user sangat berpengaruh ke performa SEO**. Itu karena yang orang-orang cari adalah informasi, bukan *website* Anda.

Karena itu, rutinalah melakukan update konten dengan meletakkan informasi yang relevan. Meski membutuhkan waktu lama, hal ini adalah salah satu cara profesional untuk melakukan *maintenance website*.

3) Tes kecepatan *loading* halaman *website*

Semua orang tidak suka menunggu, termasuk Anda. Orang tidak akan menyia-nyiakan waktu demi menunggu halaman *website* yang tidak kunjung menampilkan informasi.

Mungkin ada pemilik *website* yang berpikir, ya sudahlah tidak masalah lama, pasti ada saja orang yang mau bersabar. Faktanya, 57% orang memilih meninggalkan *website* yang memiliki page load lebih dari 3 detik.

Dengan page load yang lambat, *website* Anda dengan mudah akan tenggelam dalam halaman pencarian di Google. Itu karena Google menganggap *website* Anda memberikan *user experience* yang buruk. Otomatis, performansi SEO *website* pun terjun bebas.

Jadi, sudah sadar kan bahwa kecepatan *loading* halaman *website* cukup penting? Kalau sudah, gunakanlah platform untuk menguji *page load website* Anda. Lalu supaya kecepatan *loading* halaman *website* Anda semakin maksimal, pasanglah plugin yang mendukung.

c. Perawatan tiap kuartal

Perawatan ini dilakukan setiap tiga bulan. Kenapa harus menunggu tiga bulan? Tidak setiap hari saja sekalian?

Tidak semua hal bisa dilakukan dalam jarak waktu pendek. Jika Anda memaksakan diri ingin mengerjakan semuanya sekaligus, pasti akan pusing sendiri.

Nah, berikut adalah maintenance *website* yang perlu Anda lakukan setidaknya setiap tiga bulan sekali.

1) Periksa semua info kontak aktif

Pernahkah Anda menghubungi seseorang dan yang terdengar di ujung telepon adalah, "Nomor yang Anda hubungi tidak terdaftar"? Anda pun memeriksa catatan nomor kembali, mencoba menelepon lagi, tapi mendapat respon yang sama.

Alhasil, informasi yang ingin Anda tanyakan atau katakan pun gagal tersampaikan. Dan ternyata, penyebabnya adalah orang yang ingin Anda hubungi telah mengganti nomornya.



Pastikan siapa pun bisa menjangkau Anda melalui alamat yang tertera di *website* Anda. Jika ada perubahan, segeralah memperbarui informasi kontak Anda. Terdengar sepele, tapi bisa membanjiri Anda dengan berbagai kerugian jika Anda memilih menggampangkannya.

2) Perbaiki dan perbarui struktur desain

Website bisa diibaratkan seperti rumah. Rumah memang bisa berdiri dalam jangka waktu yang lama, meski diterpa angin dan hujan. Namun jika dibiarkan terus menerus, perlahan tapi pasti akan ada atap yang bocor, dinding yang berlumut, dan komponen lain yang mulai merapuh.

Nah, sama halnya dengan *website*. Mau desain Anda sebagus dan sekeren apapun, akan percuma jika Anda tidak pernah melakukan perawatan.

Kenapa? Serupa dengan konten, desain memiliki trennya sendiri. Hal tersebut disebabkan oleh *user experience* pengguna internet yang berubah-ubah.




Contohnya, dulu Anda menikmati tampilan *website* yang serba putih dan terang. Tapi lama-lama, justru tampilan seperti itu memunculkan masalah baru: keluhan mata lelah. Akhirnya, lahirlah tren dark mode yang membuat *website* beserta kontennya lebih ramah di mata.


Itu hanya satu contoh saja, lho. Yang jelas, melakukan maintenance *website* dengan memperbaiki struktur desain itu penting. Maka dari itu, **ikuti info seputar navigasi, warna tampilan, dan desain responsif yang update dengan tren Google.**

Jika perlu, perbarui tampilan *website* Anda. Tapi, jangan lupa juga untuk menyediakan **halaman**

	<p><u>sementara</u> supaya pengunjung tahu bahwa <i>website</i> Anda sedang di-maintenance.</p> <p>3) Periksa durasi masa aktif domain Anda</p> <p>Gara-gara sibuk mengurus SEO, struktur <i>website</i>, konten, dan lain-lain, Anda jadi melupakan durasi masa aktif domain Anda. Akibatnya, Anda harus rela ketika orang-orang jadi tidak bisa mengakses <i>website</i> Anda. Parahnya lagi, domain Anda akan lenyap secara otomatis. Pasti akan sayang sekali bukan jika usaha yang Anda bangun susah payah runtuh gara-gara melupakan tanggal kadaluarsa <i>website</i> ? Nah, sebelum Anda kehilangan itu semua, pastikan periksa durasi masa aktif domain Anda saat melakukan maintenance <i>website</i></p>
--	---

	<h2>RANGKUMAN</h2>
	<ol style="list-style-type: none"> 1. Dalam merancang aplikasi web harus memahami:(1) pengertian perancangan aplikasi. merancang sebuah aplikasi diperlukan metode atau langkah-langkah yang diperlukan atau dikenal dengan algoritma. (2) pengertian algoritma pemrograman. suatu hal yang paling dasar, oleh karenanya kamu harus bisa memahami dan mengetahui tahap awal dari pembuatan sebuah program. (3) perbedaan algoritma dan flowchart. algoritma dan flowchart tidak bisa dipisahkan, sebab flowchart menggambarkan desain program yang akan dibuat dengan berpedoman pada algoritma atau pola pikir. (4) contoh algoritma pemrograman. (5) cara kerja algoritma pemrograman. adalah deskripsi dari sebuah proses untuk mengerjakan sesuatu yang disusun dalam sederet aksi. (6) fungsi algoritma pemrograman. adalah untuk menyelesaikan suatu masalah. Algoritma ini merupakan suatu hal yang penting dalam aktivitas membuat suatu program. 2. Dalam pemrograman aplikasi web harus memahami: (1) pengertian program dan pemrograman. Program dapat diartikan sebagai himpunan atau kumpulan instruksi tertulis yang dibuat oleh programmer atau suatu bagian executable dari suatu perangkat lunak sedangkan pemrograman adalah cara membuat program menjadi sebuah program komputer, atau bisa diartikan sebagai suatu kumpulan urutan perintah ke komputer untuk mengerjakan sesuatu. (2) jenis-jenis bahasa pemrograman. antara lain Bahasa Pemrograman Tingkat Rendah, Menengah dan tinggi. 3. Membuat <i>Client server</i> sederhana harus memahami: (1) pengenalan local XAMPP. XAMPP adalah aplikasi yang memungkinkan Anda untuk membuat web server lokal untuk melakukan pengetesan <i>website</i>. (2) prosedur instalasi XAMPP.

	<p>Yaitu dengan cara mendownload dan menginstall nya. (3) operasionalisasi XAMPP. Untuk menjalankan XAMPP silakan buka aplikasi XAMPP kemudian klik tombol Start pada Apache dan MySQL. Jika berhasil dijalankan, Apache dan MySQL akan berwarna hijau. (4) mengatasi error pada XAMPP. Error pada XAMPP antara lain Apache error dan MySQL XAMPP Error.</p> <p>4. Membuat aplikasi web sederhana harus mengerti:(1) pembuatan algoritma dan flowchart aplikasi. (2) pembuatan aplikasi menghitung kecepatan. (3) konfigurasi dan menjalankan XAMPP (Server Side). (4) setting dan koneksi DB PHP MyAdmin. (5) instalasi <i>sublime text</i>. (6) pembuatan web sederhana dengan <i>sublime text</i>. (7) perawatan aplikasi web yang dibuat.</p>
--	---

	<h3>SOAL LATIHAN</h3>
	<ol style="list-style-type: none"> 1. Jelaskan pengertian perancangan aplikasi! 2. Jelaskan pengertian algoritma pemrograman! 3. Jelaskan perbedaan algoritma dan flowchart! 4. Jelaskan contoh algoritma pemrograman! 5. Jelaskan cara kerja algoritma pemrograman! 6. Jelaskan fungsi algoritma pemrograman! 7. Jelaskan pengertian program dan pemrograman! 8. Jelaskan jenis-jenis bahasa pemrograman! 9. Jelaskan pengenalan local XAMPP! 10. Jelaskan prosedur instalasi XAMPP! 11. Jelaskan operasionalisasi XAMPP! 12. Jelaskan mengatasi error pada XAMPP! 13. Jelaskan pembuatan algoritma dan flowchart aplikasi! 14. Jelaskan pembuatan aplikasi menghitung kecepatan! 15. Jelaskan konfigurasi dan menjalankan XAMPP (Server Side)! 16. Jelaskan setting dan koneksi DB PHP MyAdmin! 17. Jelaskan instalasi <i>sublime text</i>! 18. Jelaskan pembuatan web sederhana dengan <i>sublime text</i>! 19. Jelaskan perawatan aplikasi web yang dibuat!