

# **UNIVERSITAS GUNADARMA**

**FAKULTAS ILMU KOMPUTER UNIVERSITAS GUNADARMA**



## **TULISAN ILMIAH**

**PEMBUATAN WEBSITE E-LEARNING PROGRAMMING  
MENGUNAKAN PHP NATIVE DAN FRAMEWORK BOOTSTRAP**

**Nama : Kristovel Adi Sucipto**  
**NPM : 16117987**  
**Jurusan : Sistem Informasi**  
**Pembimbing : IMAM AHMAD TRINUGROHO, ST., MMSI**

**Diajukan Guna Melengkapi Sebagian Syarat Dalam Mencapai Gelar Setara  
Sarjana Muda**

**Jakarta**

**2023**

## **LEMBAR PENGESAHAN**

Judul PI : PEMBUATAN WEBSITE E-LEARNING  
PROGRAMMING MENGGUNAKAN PHP NATIVE  
DAN FRAMEWORK BOOTSTRAP  
Nama : Kristovel Adi Sucipto  
NPM : 16117987  
Tanggal Sidang : 21 Juli 2023  
Tanggal Lulus : 21 Juli 2023

Menyetujui

Pembimbing

Kasubag. Sidang PI

(Imam Ahmad Trinugroho, ST., MMSI)

(Dr. Sri Nawangsari, SE., MM.)

Ketua Jurusan Sistem Informasi

(Dr. Setia Wirawan, SKom, MMSI.)

## ABSTRAK

**Kristovel Adi Sucipto, 16117987**

**PEMBUATAN E-LEARNING PROGRAMMING WEB MENGGUNAKAN  
PHP NATIVE DAN BOOTSTRAP**

Tulisan Ilmiah. Jurusan Sistem Informasi. Fakultas Ilmu Komputer dan Teknologi Informasi. Universitas Gunadarma. 2023.

Kata Kunci: Bootstrap, *Code Editor*, *E-Learning*, Web  
(ix + 91 + Lampiran)

Pada masa pandemi covid-19, banyak sekali kegiatan yang dilakukan secara *online* untuk menghambat mata rantai virus tersebut salah satunya yaitu belajar. Materi belajar yang sangat mumpuni untuk dipelajari yaitu *programming web* karena materi ini banyak diminati untuk pekerjaan yang banyak digantikan oleh teknologi komputer. Maka dibuatlah *e-learning* yang berisikan materi HTML, Javascript, dan PHP sebagai bahasa program dasar pembuatan website serta kuis dan code editor sebagai praktik *coding*. *E-learning* tersebut dapat dikunjungi dengan URL <http://belajar-code.infinityfreeapp.com/>.

Daftar Pustaka (2011-2023)

## KATA PENGANTAR

Puji dan syukur kepada Tuhan Yesus, karena atas anugrah dan kasih-Nya, penulis dapat menyusun dan menyelesaikan penulisan ilmiah ini.

Penulisan Ilmiah ini disusun guna melengkapi sebagian syarat dalam mencapai gelar Setara Sarjana Muda pada jurusan Sistem informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Gunadarma. Adapun judul penulisan ini adalah “Pembuatan Website E-Learning Programming Menggunakan PHP Native dan Framework Bootstrap”.

Dalam penyusunan Tulisan Ilmiah ini, penulis tidak jarang mendapat hambatan, tantangan, dan kesulitan, hal ini disebabkan karena banyak faktor dari keterbatasan dan kemampuan yang penulis miliki.

Penulis juga sadar sepenuhnya bahwa tanpa bantuan, bimbingan, petunjuk serta dorongan dari berbagai pihak, tidak mungkin Tulisan Ilmiah ini dapat diselesaikan, sehubungan dengan hal tersebut, maka penulis dalam kesempatan ini dengan segala kerendahan hati mengucapkan terimakasih, kepada :

1. Prof. Dr. E.S. Margianti, SE., MM. Selaku Rektor Universitas Gunadarma.
2. Prof. Dr.rer.nat. Achmad Benny Mutiara, SSI, SKom Selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi.
3. Dr. Setia Wirawan, SKom, MMSI Selaku Ketua Jurusan Sistem Informasi.
4. Dr. Sri Nawangsari, SE., MM Selaku Kasubag Penulisan Ilmiah Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Gunadarma
5. Imam Ahmad Trinugroho, ST., MMSI Selaku Dosen Pembimbing yang telah memberikan bimbingan untuk Penulisan Ilmiah ini dari awal sampai akhir.
6. Bapak dan Ibu Dosen serta Staff Pengajar dan Karyawan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Gunadarma yang telah membantu baik secara langsung maupun tidak langsung.
7. Kepada Lemeria Siburian selaku single mother yang saya sayangi telah memberikan motivasi dan dukungan secara moril dan materil sehingga penulis dapat menyelesaikan Penulisan Ilmiah ini.
8. Kepada Yusuf Adi Ariyanto selaku adik saya yang telah membantu penulis dalam pembuatan Penulisan Ilmiah ini.
9. Kepada jemaat gereja GEPKIN Cilincing yang telah mendoakan, memberikan motivasi dan mendukung penulis secara moril sehingga penulis dapat mengerjakan Penulisan Ilmiah ini dengan semangat.

10. Terima kasih juga kepada anggota AMXL yang selalu support saya, teman-teman terdekat penulis yang telah membantu belajar pembuatan website dan memberikan dukungan secara moril dan materil sehingga penulis dapat menyelesaikan Penulisan Ilmiah ini dengan baik.
11. Semua pihak yang ikut membantu dalam Penulisan Ilmiah ini baik langsung maupun tidak langsung yang tidak dapat penulis sebutkan satu-persatu dalam penulisan ini.

Akhir kata, dengan segala harapan dan kerendahan hati penulis berharap Tulisan Ilmiah ini dapat diterima dan bermanfaat bagi semua pihak yang memerlukan dan dapat menambah pengetahuan penulis juga pembaca.

Jakarta, 19 Juni 2023

Kristovel Adi Sucipto

## DAFTAR ISI

<b>LEMBAR PENGESAHAN .....</b>	<b>i</b>
<b>ABSTRAK .....</b>	<b>ii</b>
<b>KATA PENGANTAR.....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>v</b>
<b>DAFTAR TABEL .....</b>	<b>viii</b>
<b>DAFTAR LAMPIRAN .....</b>	<b>ix</b>
<b>1. PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Ruang Lingkup .....	2
1.3. Tujuan Penulisan .....	3
1.4. Metode Penelitian.....	3
1.5. Sistematika Tulisan Ilmiah .....	4
<b>2. TINJAUAN PUSTAKA .....</b>	<b>5</b>
2.1 <i>E-Learning</i> .....	5
2.1.1 Pengertian <i>E-Learning</i> .....	5
2.1.2 Manfaat <i>E-Learning</i> .....	5
2.2 Internet.....	7
2.3 World Wide Web (WWW).....	7
2.3.1 Pengertian WWW .....	7
2.3.2 Jenis-jenis <i>Website</i> .....	8
2.4 HTML 5.....	8
2.4.1 Kelebihan dan kekurangan HTML.....	9
2.4.2 <i>Tag</i> HTML .....	9
2.5 PHP.....	11
2.5.1 Kelebihan PHP .....	11
2.5.2 <i>PHP Native</i> .....	12

2.6	Javascript .....	12
2.6.1	Kelebihan Javascript .....	13
2.7	Bootstrap .....	14
2.7.1	CSS.....	14
2.7.3	<i>Framework</i> Bootstrap .....	14
2.7.4	Kelebihan <i>Framework</i> Bootstrap .....	14
2.8	Codemirror .....	15
2.9	<i>Database</i> .....	15
2.9.1	Pengertian <i>Database</i> .....	15
2.9.2	Perintah MySQL .....	16
2.9.3	MySQL.....	16
2.10	Visual Studio Code.....	16
2.11	XAMPP .....	17
2.12	<i>Software Development Life Cycle</i> .....	17
2.12.1	Model <i>Waterfall</i> .....	18
2.13	<i>Unified Modeling Language</i> .....	19
2.13.1	Pengertian UML.....	19
2.13.2	Use Case Diagram.....	20
2.13.3	Activity Diagram.....	22
2.14	Struktur Navigasi.....	23
2.14.1	Navigasi Linier .....	24
2.14.2	Navigasi Non-Linier.....	24
2.14.3	Navigasi Hirarki .....	25
2.14.4	Navigasi Campuran.....	25
<b>3.</b>	<b>PERANCANGAN DAN IMPLEMENTASI .....</b>	<b>26</b>
3.1	GAMBARAN UMUM SISTEM.....	26
3.2	ANALISIS SISTEM.....	26
3.2.1	KEBUTUHAN FUNGSIONAL .....	27
3.2.2	Kebutuhan Non Fungsional.....	27
3.2.3	Pengumpulan Data Materi.....	28
3.3	PERANCANGAN .....	31
3.4.1	Rancangan Struktur Navigasi.....	31
3.4.2	Unified Model Language (UML).....	32

3.4.3	Rancangan Tampilan Antarmuka.....	43
3.4.4	Rancangan <i>Database</i> .....	53
3.4	IMPLEMENTASI .....	55
3.4.1	Halaman User.....	55
3.4.2	Halaman Admin .....	70
3.5	UJI COBA .....	74
3.5.1	Hosting .....	74
3.5.2	Black Box.....	78
3.6	PEMELIHARAAN .....	87
<b>4.</b>	<b>PENUTUP</b> .....	<b>88</b>
4.1	Kesimpulan.....	88
4.2	Saran .....	88
	<b>DAFTAR PUSTAKA</b> .....	<b>89</b>
	<b>LAMPIRAN</b> .....	<b>L-1</b>



## DAFTAR TABEL

Table 2.1 Tag HTML .....	9
Table 2.2 Simbol Use Case .....	21
Table 2.3 Simbol Activity Diagram .....	22
Table 3.1 Data Materi HTML .....	28
Table 3.2 Data Materi Javascript .....	29
Table 3.3 Data Materi PHP .....	30
Table 3.4 Rancangan Tabel User .....	53
Table 3.5 Rancangan Tabel Materi .....	54
Table 3.6 Rancangan Tabel Soal.....	54
Table 3.7 Table Pengujian Halaman Home .....	78
Table 3.8 Table Pengujian Halaman Materi .....	79
Table 3.9 Table Pengujian Halaman Code Editor.....	81
Table 3.10 Table Pengujian Login dan Registrasi .....	83
Table 3.11 Table Pengujian Admin .....	84

## **DAFTAR LAMPIRAN**

LEMBAR PERNYATAAN UJI COBA APLIKASI.....	L-1
OUTPUT PROGRAM.....	L-2

# 1. PENDAHULUAN

## 1.1. Latar Belakang

Perkembangan teknologi yang sangat maju di era *modern* dan globalisasi memungkinkan berbagai kegiatan dilakukan secara cepat dan efisien, salah satunya dengan adanya internet. Internet adalah sebuah sistem informasi global yang menghubungkan komputer-komputer dan jaringan-jaringan komputer di seluruh dunia. Banyak manfaat yang telah dirasakan oleh berbagai masyarakat dengan adanya internet saat ini. Salah satunya adalah dalam bidang pendidikan dengan penggunaan *e-learning* dalam kegiatan pembelajaran di sekolah, perguruan tinggi, tempat — tempat kursus bahkan komunitas — komunitas online sudah mulai menggunakan konsep seperti ini.

*E-learning* atau *electronic learning* merupakan konsep yang merubah paradigma model proses pembelajaran dengan menggunakan berbagai jaringan media elektronik (audio/visual) berbasis website yang memiliki keterhubungan dengan teknologi internet. Seiring dengan perkembangan teknologi informasi dan tuntutan globalisasi pendidikan serta pembelajaran jarak jauh, berbagai konsep telah dikembangkan untuk menggantikan metode pembelajaran tradisional, salah satunya adalah konsep *e-learning*. *E-learning* dapat digunakan sebagai alternatif atas permasalahan dalam bidang pendidikan, baik sebagai tambahan, pelengkap maupun pengganti atas kegiatan pembelajaran yang sudah ada.

Pada masa covid-19 ini, banyak sekali sekolah dan kampus menerapkan pembelajaran atau kegiatan belajar mengajar dari rumah melalui media online seperti Whatsapp, G-meet, Zoom dll. Pada saat melakukan kegiatan belajar online tersebut, dosen maupun pengajar lainnya kerap kali memberikan materi secara kurang menyeluruh atau tidak memberikan dasar praktik yang mudah dipahami mahasiswa maupun pelajar lainnya sehingga menyebabkan kegiatan belajar online tersebut menjadi tidak efektif terhadap pengembangan kemampuan pelajar manapun.

Pada saat ini banyak sekali website pembelajaran tentang *coding* atau *programming* yang tersedia dan sudah dikenal oleh banyak orang seperti CodePolitan,

W3School, Udemy, Programmer Zaman Now, dan Web Programming Unpas yang hanya tersedia di Youtube yang mana sangat membantu mahasiswa dan pelajar lainnya dalam mempelajari *coding* atau *programming*.

Oleh karena itu penulis ingin melakukan suatu analisis dan perancangan terhadap suatu sistem yang akan dibuat agar sesuai dengan tujuan pembuatan sistem dengan judul “*Pembuatan Website E-Learning Programming Menggunakan PHP Native dan Framework Bootstrap*”. Dengan website ini, dapat menyediakan materi pembelajaran dengan harapan dapat mudah dimengerti dan terarah oleh mahasiswa dan pelajar lainnya serta code editor dimana mahasiswa dan pelajar lainnya dapat melakukan praktik langsung pada saat selesai membaca materi tersebut.

## **1.2. Ruang Lingkup**

Berdasarkan latar belakang di atas, batasan masalah yang dapat ditentukan dalam pembuatan *Website E-learning* adalah, batasan pertama yaitu pembuatan website ini dibuat dengan bahasa pemrograman PHP dengan Framework Bootstrap sebagai desain web utama dan MySQL adalah sebagai basis datanya.

Batasan kedua yaitu materi pembelajaran yang dimuat pada *website e-learning* adalah materi yang diambil dari website W3School. Bahasa pemrograman yang akan dimuat pada materi adalah HTML, Javascript, dan PHP.

Batasan ketiga yaitu pembuatan *code editor* pada *e-learning* adalah menggunakan *web component* Codemirror. Tidak hanya code editor saja yang dibuat, tetapi beberapa ekstensi tambahan sebagai pendukung penulisan *coding* yang rapi serta fitur dukungan lainnya dalam mengkode.

### 1.3. Tujuan Penulisan

Berdasarkan permasalahan di atas, maka tujuan penulisan ini dapat dijabarkan dalam beberapa hal. Pertama, merancang dan membangun suatu aplikasi *e-learning* berbasis *website*.

Kedua, *website e-learning* ini tidak hanya ditujukan pada mahasiswa saja, anak-anak yang bersekolah dan orang dewasa dapat menggunakan *website* tersebut bagi yang ingin belajar.

Ketiga, menjadi *website e-learning* alternatif dan memberikan materi yang dapat mudah dipahami oleh pengguna.

### 1.4. Metode Penelitian

Metode penelitian dalam penulisan ini dibuat dengan menggunakan metode SDLC *waterfall*. Dalam metode ini terbagi dalam beberapa tahapan, yaitu :

1. Analisis Sistem

Pada tahap ini akan melakukan analisis sistem berupa pengumpulan data berupa materi dari beberapa sumber dan kuis serta kebutuhan hardware dan software dalam pembuatan *website e-learning*.

2. Perancangan

Pada tahap ini akan melakukan perancangan dalam pembuatan *website e-learning* yaitu struktur navigasi, alur diagram, tampilan antarmuka, dan *database*.

3. Implementasi

Pada tahap ini merupakan inti dari penelitian, dimana program akan dibangun dengan menggunakan data yang telah diolah serta divisualisasikan pada tahap perancangan.

4. Uji Coba

Pada tahap ini dilakukan hosting aplikasi untuk mengetahui apakah aplikasi sudah berjalan dengan baik sesuai dengan kebutuhan.

5. Pemeliharaan

Pada tahap ini merupakan tahap akhir yaitu memperbaiki permasalahan website atau memodifikasi website supaya lebih baik dari sebelumnya.

### **1.5. Sistematika Tulisan Ilmiah**

Dalam sistematika tulisan ilmiah ini merupakan gambaran keseluruhan dari penulisan yang berisi empat bab. Bab pertama merupakan pendahuluan yang berisikan latar belakang, ruang lingkup, tujuan penelitian, metode penelitian, dan sistematika tulisan ilmiah. Bab kedua merupakan tinjauan pustaka yang berisikan teori-teori yang digunakan dalam penulisan. Bab ketiga merupakan perancangan dan implementasi yang menjelaskan cara menguraikan mengenai informasi yang dihasilkan, struktur navigasi yang digunakan dan struktur tahapan-tahapan pembuatan *website*. Bab keempat merupakan penutup yang berisikan kesimpulan dan saran dari uraian pada bab sebelumnya dan merupakan jawaban dari tujuan penulisan ilmiah ini.

## **2. TINJAUAN PUSTAKA**

### **2.1 *E-Learning***

#### **2.1.1 Pengertian *E-Learning***

Menurut Rusman, Deni, & Cepi (2012: 265), WBL merupakan salah satu bentuk *e-learning* yang materi maupun cara penyampaianya melalui internet (*web*). Melalui pengertian tersebut akan tercipta sebuah pembelajaran berbasis web. Pembelajaran berbasis *web* tersebut melakukan kegiatan belajar mengajar dengan memanfaatkan jaringan internet. *E-learning* berbasis *web* ini juga memberikan kesempatan para penggunanya untuk berkomunikasi dan bertukar informasi dimana saja dan kapan saja.

Menurut Rusman (2012: 293) *e-learning* merupakan segala aktivitas belajar yang menggunakan bantuan teknologi elektronik. Melalui *e-learning*, pemahaman siswa tentang sebuah materi tidak tergantung pada guru/instruktur tetapi dapat diperoleh dari media elektronik. Teknologi elektronik yang banyak digunakan misalnya internet, intranet, tape video atau audio, penyiaran melalui satelit, televisi interaktif serta CD-ROM (Rusman, 2012: 291).

#### **2.1.2 Manfaat *E-Learning***

Menurut Made Wena (2009: 213-214) manfaat *e-learning* dapat dikategorikan berdasar 3 sudut pandang, yaitu :

##### **1) Sudut pandang siswa**

E-learning dapat membuat aktivitas belajar siswa menjadi lebih fleksibel, siswa dapat mengakses pembelajaran setiap saat dan berulang-ulang. Selain itu siswa juga dapat berinteraksi dengan guru setiap saat, jadi ketika ada pertanyaan ataupun merasa kurang jelas siswa dapat langsung bertanya pada gurunya. Hal ini sangat berbeda dengan pembelajaran konvensional, di mana proses kegiatan belajar mengajar dilakukan harus tepat waktu dan tempatnya. Selain itu

menurut Seok (Lantip Diat Prasajo & Riyanto, 2011: 231) ada beberapa kelebihan *e-learning* dibanding pembelajaran konvensional apabila fasilitas infrastruktur yang mendukung terpenuhi menjangkau daerah pedesaan, yaitu:

- Sekolah-sekolah kecil di pedesaan dapat mengakses atau mempelajari mata pelajaran yang tidak diajarkan di sekolahnya
- Bagi siswa yang mengikuti *home schoolers* (program pendidikan keluarga) dapat mengikuti pembelajaran yang tidak dapat diajarkan orangtuanya seperti bahasa asing atau komputer.
- *E-learning* dapat diakses oleh siswa yang fobi (phobia) terhadap sekolah, siswa yang di rawat di rumah sakit atau di rumah, yang putus sekolah tapi berminat melanjutkan sekolahnya, siswa yang berada di luar daerah maupun di luar negeri.
- Siswa yang tidak tertampung di sekolah konvensional dapat memanfaatkan *e-learning* untuk mendapatkan pendidikan.

## 2) Sudut pandang guru/pendidik

Dari sudut pandang seorang pendidik menurut Soekartawi (Made Wena, 2009: 213) manfaat *e-learning* antara lain:

- Lebih mudah dalam melakukan kemutakhiran bahan-bahan belajar sesuai dengan perkembangan ilmu yang ada.
- Dapat mengembangkan diri atau melakukan penelitian dengan waktu luang lebih banyak.
- Mengontrol kebiasaan belajar siswa, guru dapat mengetahui kapan siswanya belajar.
- Mengecek siswa dalam mengerjakan soal-soal latihan.
- Memeriksa jawaban siswa dan memberitahukan hasil penilaian kepada siswa.



### 3) Sudut pandang sekolah

Bagi sekolah, *e-learning* memiliki beberapa manfaat antara lain:

- Akan tersedia bahan ajar yang telah divalidasi oleh ahli sesuai dengan bidangnya sehingga setiap guru dapat menggunakan dengan mudah, efektif, dan efisien.
- Pengembangan isi pembelajaran akan sesuai dengan berbagai pokok bahasan.
- Dapat menjadi pedoman praktis implementasi pembelajaran sesuai kondisi dan karakteristik pembelajaran.
- Menumbuhkan sikap kerjasama baik antara sesama guru maupun antara guru dengan siswa dalam memecahkan masalah pembelajaran.

## 2.2 Internet

Menurut Sibero(2011), internet merupakan Interconnected Network, yaitu sebuah jaringan komputer yang saling menghubungkan antar komputer secara global. Internet mampu bekerja karena adanya jaringan komputer, baik jaringan lokal maupun jaringan dengan jangkauan global yang lebih luas. Selain itu, internet juga memakai protokol komunikasi yang sama, yaitu IP atau TCP.

Menurut Berners Lee, internet yaitu sebuah jaringan yang memiliki beberapa jaringan di dalamnya. Konsep dari Berners Lee tersebut dapat diartikan bahwa adanya jaringan komputer lokal yang terhubung dengan jaringan lainnya.

## 2.3 World Wide Web (WWW)

### 2.3.1 Pengertian WWW

Menurut Gregorius(2000), *website* adalah kumpulan halaman *web* yang saling terhubung dan seluruh *file* saling terkait. *Web* terdiri dari *page* atau halaman dan

kumpulan halaman yang dinamakan *homepage*. *Homepage* berada pada posisi teratas dengan halaman-halaman terkait berada di bawahnya. Biasanya, setiap halaman di bawah *homepage* (*child page*) berisi *hyperlink* ke halaman lain dalam *web*.

*Website* adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses di seluruh dunia selama terkoneksi dengan jaringan internet. *Website* merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara, dan animasi sehingga menarik untuk dikunjungi (Sholechul Azis, 2013).

### 2.3.2 Jenis-jenis *Website*

Pernyataan dari sumber katadata.co.id, berdasarkan buku CMM Website Interaktif MCMS Joomla(CMS), jenis *website* dibagi berdasarkan sifat adalah:

- 1) *Website* dinamis, yaitu sebuah *website* yang menyediakan konten atau isi yang selalu berubah setiap saat. Contoh website dinamis adalah media berita daring.
- 2) *Website* statis, merupakan website yang kontennya sangat jarang diubah. Misalnya, profil organisasi dan sebagainya.

## 2.4 HTML 5

HTML atau Hyper-Text Markup Language merupakan bahasa *markup* standar dalam pembuatan website. Umumnya, penggunaan HTML adalah untuk menyusun bagian paragraf, heading, maupun link pada halaman web. Tapi, meskipun susunannya seperti coding, HTML bukanlah bahasa pemrograman.

HTML tidak dianggap sebagai bahasa pemrograman karena tidak bisa memberikan fungsi yang dinamis. Sekarang, bahasa markup ini dianggap sebagai standar web resmi, dikelola oleh World Wide Web Consortium (W3C), yang juga bertugas merilis pembaruan rutin HTML.

### 2.4.1 Kelebihan dan kekurangan HTML

Kelebihan yang ada pada bahasa *markup* HTML :

- Digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar. Dijalankan secara alami di setiap *web browser*.
- Mudah dipelajari.
- *Open-source* dan sepenuhnya gratis.
- Rapi dan konsisten.
- Menjadi standar resmi *web*, dikelola oleh (W3C).
- Integrasi mudah dengan bahasa *backend*, seperti PHP dan Node.js

Kekurangan yang ada pada bahasa *markup* HTML :

- Umumnya digunakan untuk halaman *web* statis. Untuk fitur dinamis, perlu menggunakan JavaScript atau bahasa *backend*, seperti PHP.
- Tidak bisa menjalankan *logic*. Alhasil, semua halaman *web* harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti *header* dan *footer*.
- Fitur baru tidak bisa digunakan dengan cepat di sebagian *browser*.
- Perilaku *browser* terkadang sulit diprediksi. Misalnya, *browser* lama tidak selalu bisa me-render *tag* yang lebih baru.

### 2.4.2 Tag HTML

Tag HTML yang lazim dipakai pada HTML 5, yaitu:

Table 2.1 Tag HTML

TAG	KETERANGAN
<!DOCTYPE>	Menentukan tipe dokumen

<html>	Membuat sebuah dokumen HTML
<head>	Membuat informasi pada dokumen
<meta>	Membuat metadata pada dokumen
<title>	Membuat judul dari halaman
<body>	Membuat isi konten pada halaman
<h1> to <h6>	Membuat heading
<p>	Membuat paragraf
 	Membuat baris baru pada halaman
<hr>	Membuat garis lurus panjang secara horizontal
<!--...-->	Membuat komentar
<b>	Membuat huruf cetak tebal
<form>	Membuat form pada HTML
<input>	Membuat kolom input pada form
<textarea>	Membuat input teks dengan banyak baris
<button>	Membuat tombol
<select>	Membuat input pilihan ( <i>dropdown</i> )
<option>	Membuat daftar pilihan dari tag <select>
<label>	Membuat label dari sebuah elemen <input>
<iframe>	Membuat jendela baru yang berisikan konten web lain
<img>	Menampilkan gambar
<a>	Membuat <i>hyperlink</i>
<ul>	Membuat urutan list dengan simbol atau karakter spesial pada HTML
<ol>	Membuat urutan list dengan nomor, abjad, dan angka romawi
<li>	Membuat sebuah item daftar
<table>	Membuat <i>table</i>
<th>	Membuat sel <i>header</i> pada <i>table</i>

<tr>	Membuat baris pada <i>table</i>
<td>	Membuat sel dalam <i>table</i>
<style>	Membuat style atau desain pada halaman web
<div>	Mengelompokkan tag atau elemen ( <i>block-line</i> )
<span>	Mengelompokkan tag atau elemen ( <i>inline</i> )
<script>	Membuat <i>script</i> program pada <i>client side</i>

## 2.5 PHP

PHP atau singkatan dari Hypertext Preprocessor, merupakan bahasa pemrograman yang berjalan pada *server side (backend)* atau sisi server. Bahasa pemrograman tersebut juga termasuk salah satu bahasa *scripting*. Bahasa *scripting* adalah bahasa yang mengotomatiskan eksekusi *task* (tugas) dalam *environment runtime* khusus. Tugas ini mencakup menginstruksikan halaman statis (dibuat dengan HTML dan CSS) untuk melakukan tindakan tertentu dengan aturan yang sudah ditetapkan.

Seperti, kita dapat menggunakan script untuk memvalidasi apakah semua kolom dalam sebuah *form* sudah diisi sebelum *form* dikirim kembali ke server. *Script* tersebut akan berjalan, lalu memeriksa semua kolom ketika user mengirimkan form. Jika ada kolom yang masih kosong, akan muncul teks peringatan untuk memberi tahu user tentangnya.

### 2.5.1 Kelebihan PHP

Kelebihan bahasa pemrograman PHP, yaitu :

- Mudah dipelajari – Dapat mempelajari PHP dengan mudah karena tersedia dokumentasi lengkap tentang fungsinya beserta contoh.
- Banyak digunakan – PHP umum digunakan untuk membuat berbagai jenis platform seperti *e-commerce*, blog, media sosial, dan lain-lain.

- Hemat biaya – PHP adalah bahasa pemrograman/skrip yang *open-source*, dengan kata lain bahasa tersebut dapat digunakan secara gratis.
- Ada banyak komunitasnya – Jika menjumpai masalah terkait PHP, banyak situs atau blog yang secara spesifik membahas PHP di internet.
- Terintegrasi dengan *database* – beberapa contoh program *database* yang terintegrasi dengan PHP adalah MySQL, Oracle, Sybase, DB2, dll.

### 2.5.2 PHP Native

PHP *Native* merupakan bahasa program PHP murni. Yang berarti program tersebut dibuat dari nol tanpa bantuan dari library eksternal maupun memiliki *styling* program sedari awal.

Walaupun dibuat dari nol, PHP *Native* atau PHP murni memiliki kelebihan tersendiri. Kelebihannya adalah program yang dapat dibaca karena program tersebut hanya bisa mudah dibaca oleh pembuat program tersebut.

Penggunaan PHP *Native* tersebut cukup diperuntukkan untuk *website* statis, yang berarti untuk pembuatan *website* yang jarang berubah dan tetap. Dan juga tidak cocok untuk penggunaan bisnis karena mudah sekali banyak melakukan perubahan *website*. Itulah kekurangan dari PHP *Native*.

## 2.6 Javascript

Javascript merupakan bahasa pemrograman yang berjalan pada *client-side* atau sisi klien dan juga termasuk bahasa *scripting*. Javascript digunakan oleh banyak programmer untuk pembuatan *website* yang sangat dinamis dan juga menampilkan interaksi kepada pengguna. Javascript juga merupakan bahasa pemrograman yang paling populer pada tahun 2022 bila dilihat berdasarkan *survey* pada *website* stackoverflow sebanyak 65,36% dari 71.547 responden.

Javascript adalah bahasa pemrograman dijalankan menggunakan *interpreter* dan dibantu dengan *web browser* seperti Google Chrome, Mozilla Firefox dan lain-lain. Pengkodingannya dibuat dengan menyisipkan *tag* `<script>` yang ada pada HTML atau bisa dipanggil secara eksternal dengan menaruh sumber direktori file Javascript tersebut.

### 2.6.1 Kelebihan Javascript

Kelebihan yang ada pada bahasa Javascript adalah:

- **Struktur yang sederhana.** Strukturnya yang sederhana membuat JavaScript lebih mudah dipelajari dan diterapkan, serta lebih cepat daripada sejumlah bahasa lain. Galat juga mudah diidentifikasi dan diperbaiki.
- **Eksekusi lebih cepat.** JavaScript mengeksekusi *script* langsung di *browser web* tanpa harus terhubung ke *server* atau menggunakan *compiler*. Selain itu, sebagian besar *browser* memungkinkan JavaScript meng-*compile* kode pada saat eksekusi program.
- **Fungsi yang serbaguna.** JavaScript kompatibel dengan bahasa pemrograman lain seperti PHP, Perl, dan Java. Bahasa ini juga menjadikan *data science* dan *machine learning* bisa diakses oleh developer.
- **Cukup populer dan banyak digunakan.** Ada berbagai sumber dan forum yang bisa membantu pemula mempelajari bahasa *scripting* ini.
- **Mengurangi beban server.** Karena berjalan di sisi klien, JavaScript bisa mengurangi permintaan yang dikirim ke *server*. Validasi data bisa dilakukan melalui *browser*, dan update hanya berlaku pada bagian halaman *web* tertentu.
- **Selalu diperbarui dan dikembangkan.** Tim *developer* JavaScript dan ECMA International terus memperbarui serta merancang *framework* dan *library* baru sehingga bisa terus relevan.

## 2.7 Bootstrap

### 2.7.1 CSS

CSS atau singkatan dari Cascading Style Sheet merupakan bahasa *markup* yang berfungsi untuk membuat desain tampilan pada website agar menjadi rapi dan menarik. Tidak hanya itu, CSS dapat membuat animasi sederhana dan tampilan gerak bila dipadukan dengan bahasa Javascript sebagai interaksinya.

Bahasa *markup* CSS tersebut dibuat oleh Hakon Wium Lie. Hakon lahir pada tahun 1965 di Norwegia. Dia juga pernah bekerja bersama Tim Berners-Lee di CERN kemudian mengusulkan CSS pada tahun 1999. Dan pada saat itu juga menjadi CTO Opera hingga kini.

### 2.7.3 Framework Bootstrap

*Framework* Bootstrap merupakan *framework* CSS yang berfungsi untuk mendesain *website responsive* dengan cepat dan mudah. *Framework* Bootstrap dibuat oleh Mark Otto dan Jacob Thornton dari Twitter pada tahun 2011. *Framework* ini berbasis *open-source* yang berarti dapat dipakai siapa saja secara gratis.

### 2.7.4 Kelebihan Framework Bootstrap

Kelebihan yang ada pada *Framework* Bootstrap, yaitu :

- **Ramah untuk pemula.** *Framework* ini dibuat dengan mengakses elemen dan class yang sudah siap pakai. Dan juga menyediakan *component design* yang lengkap lalu tinggal melakukan *copy paste* pada pembuatan program *web*.
- **Grid style yang canggih.** Semua elemen di dalam website bisa dibuat menyesuaikan dengan perangkat yang digunakan pengunjung. Artinya, tampilan Bootstrap dapat dimuat baik PC maupun *mobile*.



- **Kompatibilitas dengan *web browser* terbaru.** Bootstrap dapat mendukung semua *web browser* versi terbaru seperti Firefox, Google Chrome, dan Safari.
- **Bersifat *open-source*.** Bootstrap merupakan *framework* gratis yang memerlukan biaya apapun. Jadi, Bootstrap dapat diakses maupun digunakan oleh siapa saja.
- **Kebebasan kustomisasi.** Artinya *framework* ini dapat diubah sesuai selera pengguna.
- **Rutin diperbaharui.**
- **Memiliki dokumentasi yang lengkap.** Bootstrap memiliki panduan yang lengkap mengenai pengunduhan, pemasangan serta penggunaan *framework* tersebut.
- **Memiliki komunitas yang besar.** Bila memiliki kendala akan penggunaan *framework* tersebut, dapat ditanyakan melalui forum stackoverflow.

## 2.8 Codemirror

Codemirror adalah teks *editor* serbaguna yang diimplementasikan dalam bahasa Javascript. Diperuntukkan untuk mengedit *code*, dan dilengkapi dengan sejumlah *mode* bahasa dan fitur tambahan yang mengimplementasikan fungsi pengeditan yang canggih. Codemirror merupakan komponen *code editor* yang disematkan pada halaman *web*.

## 2.9 Database

### 2.9.1 Pengertian Database

Menurut C.J. Date, *database* merupakan koleksi data operasional yang sengaja disimpan dan juga dipakai oleh sistem aplikasi dari suatu organisasi. Terdapat 3 jenis data yang disimpan dalam *database* yaitu data *input*, *output* dan operasional.

- Data *input* yakni data yang masuk dari luar sistem.
- Data *output* yakni data yang dihasilkan oleh sistem.
- Data operasional yakni data yang tersimpan pada sistem.

Menurut Anhar (2010:45), *database* atau basis data adalah sekumpulan tabel-tabel yang berisi data dan merupakan kumpulan dari *field* atau *kolom*. Struktur *file* yang menyusun sebuah *database* adalah *Data Record* dan *Field*.

### 2.9.2 Perintah MySQL

Perintah dalam MySQL dibagi menjadi 3 bagian :

- **DDL (*Data Definition Language*)**. Digunakan untuk menentukan skema dari *database* dan memodifikasi struktur objek dari *database* di dalam *database* yang ada. Perintah yang termasuk pada DDL adalah : *Create*, *Drop*, *Alter* dan lain-lain.
- **DML (*Data Manipulation Language*)**. Digunakan untuk mengambil dan memanipulasi data dalam *database*. Perintah yang termasuk pada DML adalah : *Insert*, *Update*, *Delete*.
- **DCL (*Data Control Language*)**. Digunakan untuk pengontrolan data yang berhubungan dengan izin, hak dan control lainnya pada sistem *database*. Perintah yang termasuk pada DCL adalah : *Grant* dan *Revoke*.

### 2.9.3 MySQL

Menurut Arief (2011d:152) “MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi web yang menggunakan database sebagai sumber dan pengolahan datanya”.

## 2.10 Visual Studio Code

VS Code merupakan *text editor* buatan Microsoft. *Text editor* ini bersifat *open source*, dengan kata lain dapat digunakan oleh siapapun dan gratis.

VS Code mempunyai fitur yang lengkap dalam penulisan *code*. Bila dirasa kurang memuaskan, fitur tambahan dapat dilakukan dengan mengunduh dan memasang pada menu Extension.

### 2.11 XAMPP

XAMPP merupakan *software web server* yang bersifat *open-source*. Kegunaan XAMPP yaitu menjalankan *web server* itu sendiri secara *local* dan *offline*.

Fungsi yang ada pada XAMPP, ialah:

- Memiliki penerjemah bahasa pemrograman yaitu PHP dan Perl
- Memiliki DBMS (*Database Management System*) yaitu MySQL dan PHPMyAdmin
- Fungsi tersedia lainnya yaitu Tomcat, Filezilla dan lain-lain

### 2.12 *Software Development Life Cycle*

Ian Sommerville (2011) menjelaskan bahwa ada lima tahapan pada model *waterfall*, yaitu *Requirements Analysis and Definition*, *Sytem and Software Design*, *Implementation and Unit Testing*, *Integration and System Testing*, dan *Operationa and Maintenance*. Tahapan pada SDLC tersebut, yaitu :

1. ***Requirement Analysis***. Pada tahap ini mengumpulkan informasi yang dapat diperoleh dengan berbagai macam cara diantaranya, diskusi, observasi, survei, wawancara, dan sebagainya. Informasi yang diperoleh kemudian diolah dan dianalisa sehingga didapatkan data atau informasi yang lengkap mengenai spesifikasi kebutuhan pengguna akan perangkat lunak yang akan dikembangkan.

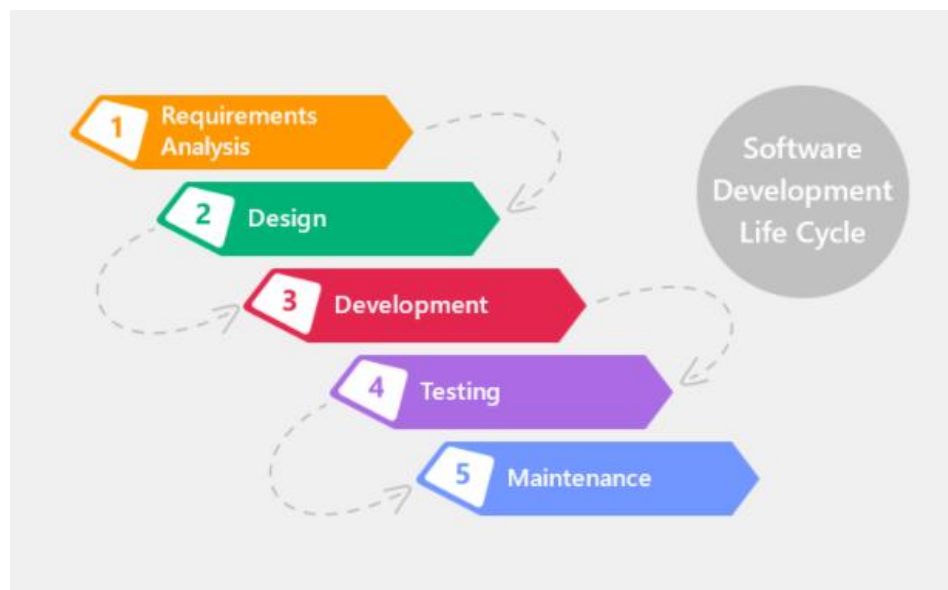
2. ***System and Software Design.*** Meneruskan dari tahapan pertama, pada tahap ini menyusun sebuah perancangan desain untuk dapat memberikan gambaran lengkap mengenai apa yang harus dikerjakan. Tahapan ini juga akan membantu pengembang untuk menyiapkan kebutuhan hardware dalam pembuatan arsitektur sistem perangkat lunak yang akan dibuat secara keseluruhan.
3. ***Implementation and Unit Testing.*** Tahapan SDLC untuk membuat model cara kerja aplikasi. Ada beberapa aspek yang perlu diperhatikan pada tahap ini, yaitu:
4. ***Integration and System Testing.*** Setelah seluruh unit atau modul yang dikembangkan dan diuji di tahap implementasi selanjutnya diintegrasikan dalam sistem secara keseluruhan. Setelah proses integrasi selesai, selanjutnya dilakukan pemeriksaan dan pengujian sistem secara keseluruhan untuk mengidentifikasi kemungkinan adanya kegagalan dan kesalahan sistem.
5. ***Operation and Maintenance.*** Pada tahap terakhir dalam metode *waterfall*, perangkat lunak yang sudah jadi dioperasikan pengguna dan dilakukan pemeliharaan. Pemeliharaan memungkinkan pengembang untuk melakukan perbaikan atas kesalahan yang tidak terdeteksi pada tahap-tahap sebelumnya. Pemeliharaan meliputi perbaikan kesalahan, perbaikan implementasi unit sistem, dan peningkatan dan penyesuaian sistem sesuai dengan kebutuhan.

#### **2.12.1 Model Waterfall**

Model *waterfall* merupakan metode pengembangan perangkat lunak tertua sebab sifatnya yang natural. Metode Waterfall merupakan pendekatan SDLC paling awal yang digunakan untuk pengembangan perangkat lunak. Urutan dalam model

*waterfall* bersifat serial yang dimulai dari proses perencanaan, analisa, desain, dan implementasi pada sistem.

Metode ini dilakukan dengan pendekatan yang sistematis, mulai dari tahap kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding*, *testing/verification*, dan *maintenance*. Langkah-langkah yang dilalui harus diselesaikan satu per satu (tidak dapat meloncat ke tahap berikutnya) dan berjalan secara berurutan, oleh sebab itu disebut *waterfall* (air terjun).



Gambar 2.1 Model Waterfall

## 2.13 *Unified Modeling Language*

### 2.13.1 Pengertian UML

Menurut Yadanur (2012), UML merupakan bahasa standar yang digunakan untuk membangun dan memvisualisasi sistem *software*.





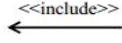
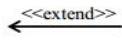
Menurut Rosa Dan Shalahuddin (2015), UML adalah salah satu bahasa dasar yang banyak digunakan dalam dunia industri guna mendefinisikan kebutuhan, membuat desain dan analisis, dan menggambarkan arsitektur dalam pemrograman yang berorientasikan pada objek.

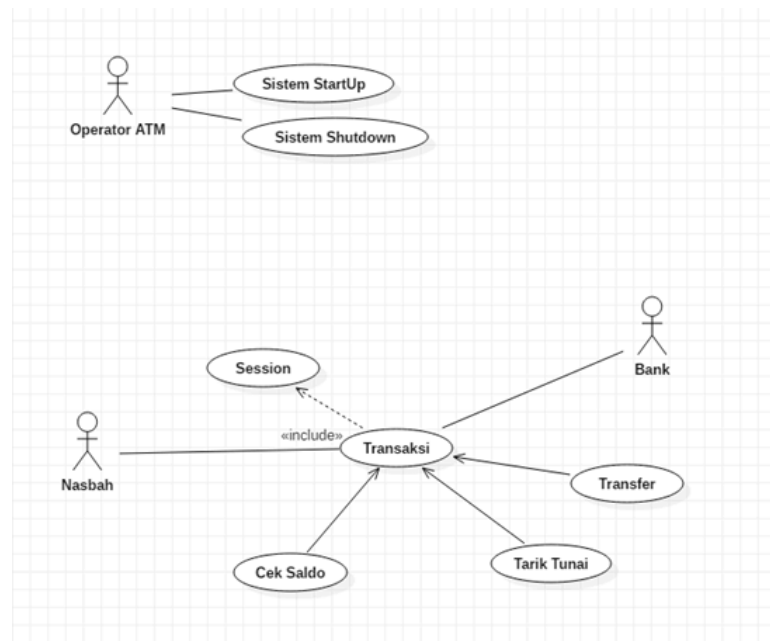
Menurut Joomla, UML merupakan *standard modeling language* yang terdiri dari kumpulan diagram-diagram, dikembangkan untuk membantu para pengembang sistem dan perangkat lunak agar bisa menyelesaikan tugas-tugas tertentu, seperti spesifikasi, visualisasi, desain arsitektur, konstruksi, simulasi, *testing*, dan dokumentasi.

### **2.13.2 Use Case Diagram**

*Use case* adalah komponen gambaran fungsional dalam sebuah sistem. Sehingga konsumen maupun pembuat saling mengenal dan mengerti mengenai alur sistem yang akan dibuat.

Table 2.2 Simbol Use Case

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi









Gambar 2.2 Use Case pada ATM

### 2.13.3 Activity Diagram

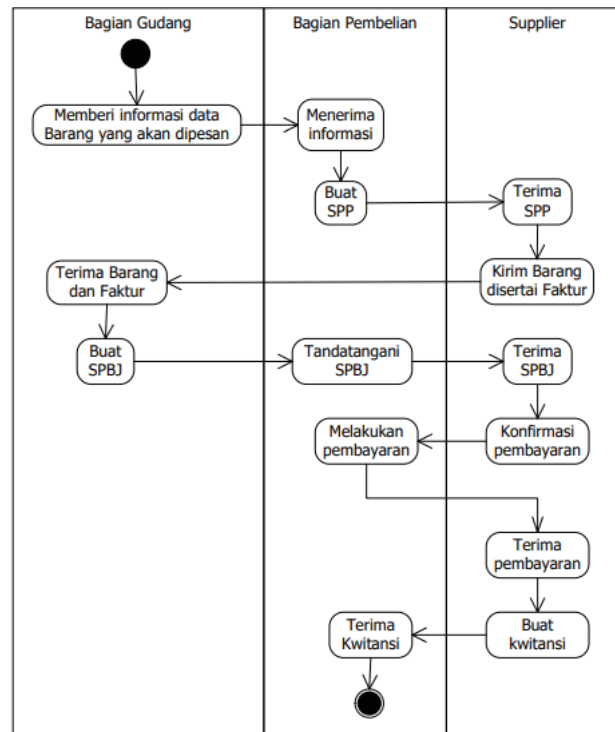
*Activity diagram*, dalam bahasa Indonesia diagram aktivitas, yaitu diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem. Runtutan proses dari suatu sistem digambarkan secara vertikal. *Activity diagram* merupakan pengembangan dari Use Case yang memiliki alur aktivitas.

Alur atau aktivitas berupa bisa berupa runtutan menu-menu atau proses bisnis yang terdapat di dalam sistem tersebut. Dalam buku *Rekayasa Perangkat Lunak* karangan Rosa A.S mengatakan, “Diagram aktivitas tidak menjelaskan kelakuan aktor. Dapat diartikan bahwa dalam pembuatan *activity diagram* hanya dapat dipakai untuk menggambarkan alur kerja atau aktivitas sistem saja.”

Table 2.3 Simbol Activity Diagram

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi





Gambar 2.3 Activity Diagram Penjualan

## 2.14 Struktur Navigasi

Alur dari sebuah program termasuk terpenting dalam pembuatan aplikasi halaman *web* dan gambaranya harus ada pada tahap perencanaan. Menentukan struktur navigasi merupakan halaman yang sebaiknya dilakukan sebelum membuat *website*.

Beberapa dasar struktur pembuatan halaman *web* yaitu linier, non linier, hirarki dan campuran.

### 2.14.1 Navigasi Linier

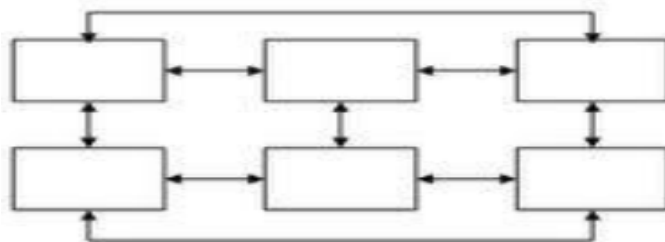
Struktur navigasi linier hanya mempunyai satu rangkaian cerita yang berurut, yang menampilkan satu demi satu rangkaian cerita yang berurut menurut urutannya.



*Gambar 2.4 Navigasi Linier*

### 2.14.2 Navigasi Non-Linier

Struktur navigasi Non-linier merupakan pengembangan dari struktur navigasi linier. Percabangan pada struktur non linier ini berbeda dengan percabangan pada struktur hirarki. Pada percabangan ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama yaitu tidak ada Master Page dan Slave Page.

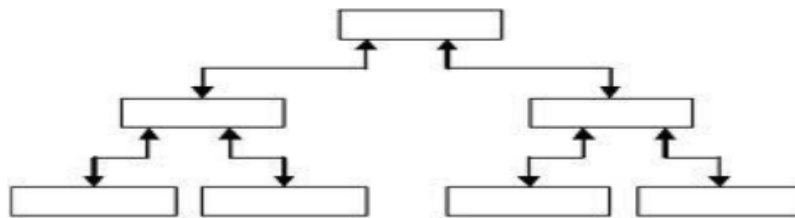


*Gambar 2.5 Navigasi Non-Linier*

### 2.14.3 Navigasi Hirarki

Struktur navigasi hirarki biasanya disebut struktur bercabang, merupakan struktur yang mengandalkan percabangan untuk menampilkan data berdasarkan kriteria tertentu.

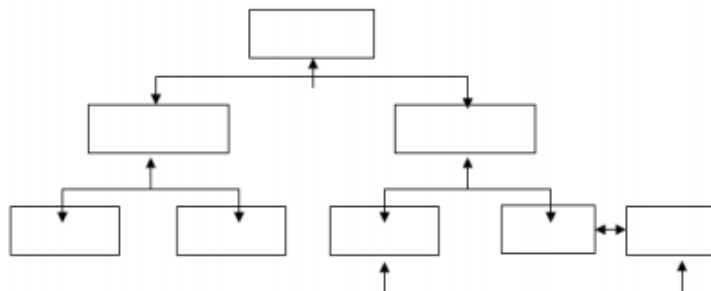
Tampilan pada menu satu akan disebut sebagai *Master Page* (halaman utama pertama), halaman ini mempunyai halaman percabangan yang disebut *slave page* (halaman pendukung).



Gambar 2.6 Navigasi Hirarki

### 2.14.4 Navigasi Campuran

Struktur navigasi *composite* (campuran) disebut juga struktur navigasi bebas yang merupakan gabungan dari ketiga struktur yang ada. Struktur navigasi ini biasa digunakan dalam pembuatan *multi media* karena dapat memberikan keinteraksian yang lebih tinggi.



Gambar 2.7 Navigasi Campuran

### **3. PERANCANGAN DAN IMPLEMENTASI**

#### **3.1 GAMBARAN UMUM SISTEM**

*Website e-learning* ini dibuat untuk sarana pembelajaran bahasa pemrograman PHP, Javascript, serta bahasa utama *web* yaitu HTML. Pembuatan *website* ini memiliki 2 user yaitu admin dan user.

Pada akun admin, disediakan halaman yaitu pembuatan materi dan pembuatan kuis. Admin bertugas membuat materi mengenai bahasa pemrograman sesuai penyebutan di atas dan dapat berimprovisasi bila admin tersebut ingin membuat materi dengan bahasa pemrograman selain di atas. Admin juga bertugas membuat kuis mengenai bahasa pemrograman.

Pada akun user, disediakan halaman materi, *code editor*, dan kuis. Halaman materi berfungsi untuk pembelajaran si *user*, halaman code editor berfungsi untuk mencoba mempraktikan pemrograman atau coding, halaman kuis berfungsi untuk menjawab kuis sebagai *refreshing* ingatan setelah belajar dari halaman materi. Halaman materi dan *code editor* dapat dimasuki langsung tetapi halaman kuis harus melakukan *login* akun terlebih dahulu. Bila tidak ada akun dapat mendaftar pada halaman registrasi akun.

#### **3.2 ANALISIS SISTEM**

Pada langkah analisis merupakan analisa terhadap kebutuhan dalam pembuatan *website e-learning*. Kebutuhan analisa tersebut yaitu kebutuhan fungsional dan non-fungsional, serta pengumpulan data materi.

### 3.2.1 KEBUTUHAN FUNGSIONAL

Analisa kebutuhan fungsional dibutuhkan untuk mengetahui hal-hal yang bisa dikerjakan oleh sistem. Kebutuhan fungsional sistem yang akan dirancang sebagai berikut:

- Fitur Materi. Fitur ini berisikan materi tentang *programming*.
- Fitur Kuis. Fitur ini berisikan kuis.
- Fitur Registrasi. Fitur ini berfungsi untuk registrasi akun.
- Fitur Login. Fitur ini berfungsi untuk *login* akun user dan juga admin yang dibuat dengan perbedaan level user.
- Fitur Dashboard. Fitur ini berfungsi menampilkan grafik/*chart* banyaknya materi dan kuis yang tersimpan pada *database*.
- Fitur Tambah Materi. Fitur ini berfungsi menambahkan dan mengedit materi oleh admin.
- Fitur Tambah Kuis. Fitur ini berfungsi membuat atau menambahkan kuis oleh admin.
- Fitur Edit Kuis. Fitur ini berfungsi mengedit dan menghapus kuis oleh admin.

### 3.2.2 Kebutuhan Non Fungsional

Analisa kebutuhan non fungsional dibutuhkan sebuah kebutuhan yang dilakukan untuk mengetahui spesifikasi kebutuhan dalam pembuatan dari sebuah sistem atau aplikasi yang telah dibuat. Spesifikasi kebutuhan seperti perangkat keras (*hardware*) dan perangkat lunak (*software*).

- a. Kebutuhan spesifikasi perangkat keras :
  1. Laptop ASUS X441NA.
  2. Processor Intel Celeron N3350
  3. *Random Access Memory* 4 GB
- b. Kebutuhan spesifikasi perangkat lunak :

1. Sistem operasi Windows
2. Server Web Local XAMPP
3. PhpMyAdmin
4. Visual Studio Code sebagai *text editor*
5. Web browser yaitu Microsoft Edge

### 3.2.3 Pengumpulan Data Materi

Penelitian ini dilakukan untuk mengumpulkan data-data tentang materi yang akan dimuat pada *website e-learning*. Materi tersebut diambil berdasarkan *website* yang banyak dikunjungi oleh orang yaitu w3school.com sebagai patokan panduan materi web ini serta MDN Web Docs. Tak hanya itu saja, Channel Youtube Web Programming Unpas juga menjadi bahan penelitian pendukung dalam pembuatan materi tersebut.

#### 3.2.3.1 Data Materi HTML

Data materi HTML yang dapat diuraikan sebagai berikut :

*Table 3.1* Data Materi HTML

No	Nama Materi	Deskripsi Materi
1	HTML Intro	Materi ini menjelaskan mengenai pengenalan HTML yaitu, Apa itu HTML, <i>tag</i> utama dalam HTML dasar, struktur halaman HTML.
2	HTML Block & Inline	Materi ini menjelaskan mengenai <i>tag-tag</i> yang dikelompokkan sebagai <i>tag block</i> dan <i>tag inline</i> .
3	HTML Atribut	Materi ini menjelaskan atribut yang ada pada <i>tag</i> HTML.
4	HTML Id	Materi ini menjelaskan mengenai atribut Id.
5	HTML Class	Materi ini menjelaskan mengenai atribut Class.

<b>6</b>	HTML Style dan CSS	Materi ini menjelaskan mengenai pembuatan <i>style</i> dalam elemen HTML serta penggunaan CSS.
<b>7</b>	HTML List	Materi ini menjelaskan mengenai pembuatan list.
<b>8</b>	HTML Table	Materi ini menjelaskan mengenai pembuatan table.
<b>9</b>	HTML Form	Materi ini menjelaskan pembuatan form input.
<b>10</b>	HTML Javascript	Materi ini menjelaskan pembuatan Javascript dalam HTML.

### 3.2.3.2 Data Materi Javascript

Data materi Javascript yang dapat diuraikan sebagai berikut :

*Table 3.2 Data Materi Javascript*

<b>No</b>	<b>Nama Materi</b>	<b>Deskripsi Materi</b>
<b>1</b>	Javascript Intro	Materi ini menjelaskan mengenai sejarah adanya Javascript dan penempatan <i>script</i> Javascript.
<b>2</b>	Javascript Output	Materi ini menjelaskan mengenai sintaks <i>output</i> Javascript.
<b>3</b>	Javascript Variabel dan Tipe Data	Materi ini menjelaskan mengenai variabel dan tipe datanya pada Javascript.
<b>4</b>	Javascript Aritmatika dan Operator	Materi ini menjelaskan mengenai bilangan aritmatika dan operator.
<b>5</b>	Javascript String	Materi ini menjelaskan <i>string</i> pada Javascript.
<b>6</b>	Javascript Array	Materi ini menjelaskan mengenai array
<b>7</b>	Javascript Iterasi	Materi ini menjelaskan mengenai perulangan.
<b>8</b>	Javascript Percabangan	Materi ini menjelaskan mengenai kondisi percabangan.

<b>9</b>	Javascript Events	Materi ini menjelaskan mengenai <i>event</i> pada Javascript.
<b>10</b>	Javascript Function	Materi ini menjelaskan penggunaan Function.

### 3.2.3.3 Data Materi PHP

Data materi PHP yang dapat diuraikan sebagai berikut :

*Table 3.3 Data Materi PHP*

<b>No</b>	<b>Nama Materi</b>	<b>Deskripsi Materi</b>
<b>1</b>	PHP Intro	Materi ini menjelaskan mengenai Apa itu PHP dan cara menginstall PHP.
<b>2</b>	PHP Sintaks	Materi ini menjelaskan mengenai sintaks pada PHP.
<b>3</b>	PHP Variabel	Materi ini menjelaskan mengenai variabel pada PHP.
<b>4</b>	PHP Echo/Print	Materi ini menjelaskan mengenai output pada PHP.
<b>5</b>	PHP Tipe Data	Materi ini menjelaskan mengenai tipe data pada PHP.
<b>6</b>	PHP Operators	Materi ini menjelaskan mengenai operator pada PHP.
<b>7</b>	PHP Percabangan	Materi ini menjelaskan mengenai percabangan IF dan Switch Case pada PHP.
<b>8</b>	PHP Loop	Materi ini menjelaskan mengenai perulangan pada PHP.
<b>9</b>	PHP Function	Materi ini menjelaskan mengenai <i>function</i> pada PHP.



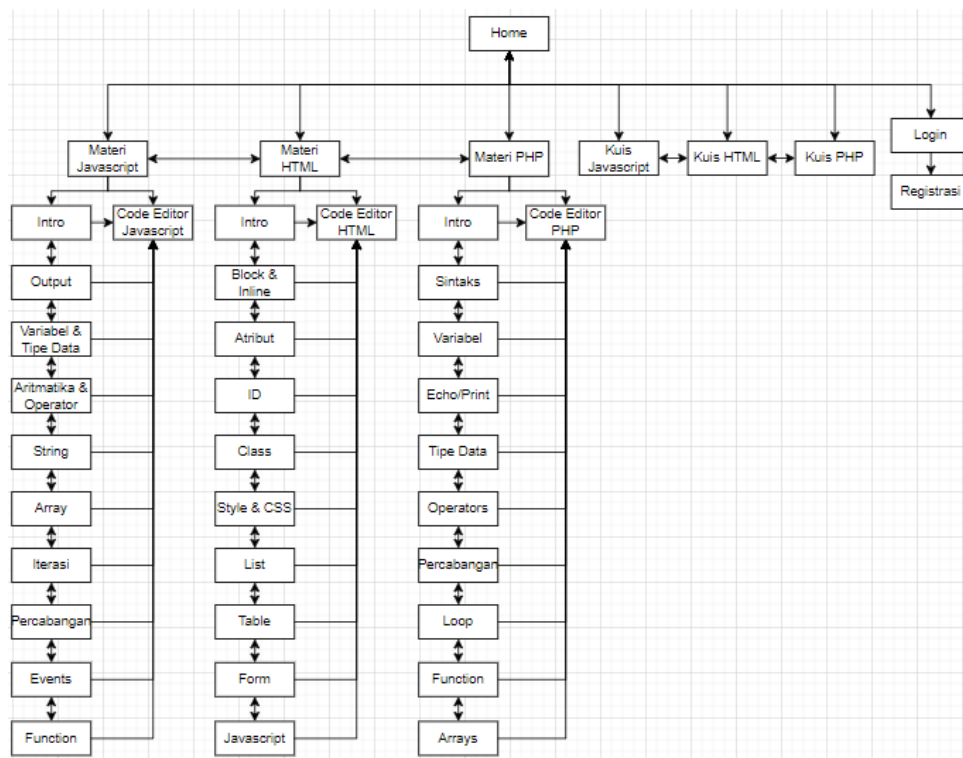
10	PHP Arrays	Materi ini menjelaskan mengenai <i>array</i> pada PHP.
----	------------	--

### 3.3 PERANCANGAN

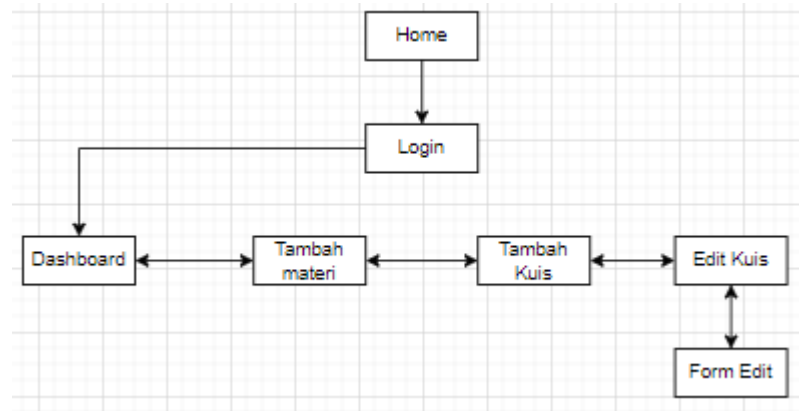
Perancangan *website* merupakan proses untuk mengaplikasikan kebutuhan ke dalam sebuah alur kerja sistem. Pada tahap ini menjelaskan struktur navigasi, *Unified Modelling Language* (UML) yang terdiri dari *use case diagram* dan *activity diagram*, tampilan antarmuka serta *database*.

#### 3.4.1 Rancangan Struktur Navigasi

Pada perancangan struktur navigasi pada *website e-learning* tersebut, dibuat dengan menggunakan struktur navigasi campuran atau *composite*. Perancangan struktur navigasi ini dibagi dua berdasarkan level user-nya, yaitu user dan admin. Struktur navigasi pada user dan admin seperti di bawah berikut.



Gambar 3.1 Struktur Navigasi User



*Gambar 3.2 Struktur Navigasi Admin*

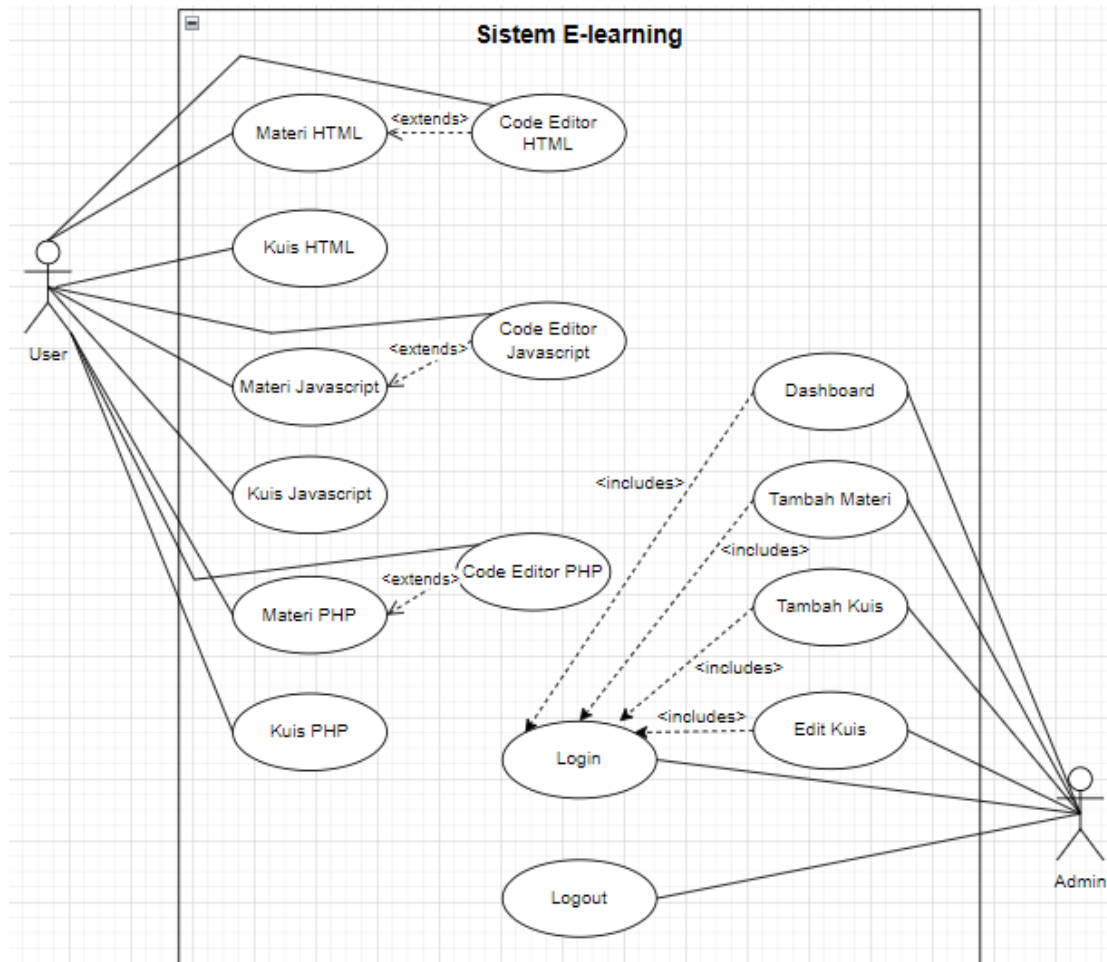
Penjelasan pada struktur navigasi user, diawali dengan halaman Home. Lalu di halaman Home tersebut dapat beralih ke halaman Login, Registrasi, Materi HTML, Materi Javascript, Materi PHP, Kuis HTML, Kuis Javascript dan Kuis PHP. Pada saat memasuki halaman Materi HTML dapat beralih ke halaman materi yang diinginkan. Setiap memasuki halaman materi yang dipilih dapat memasuki halaman Code Editor yang disediakan berupa *button*, begitu juga halaman Materi Javascript dan PHP.

Sedangkan untuk penjelasan struktur navigasi admin yaitu, diawali dengan halaman Home. Lalu diarahkan ke halaman Login, melakukan login dengan akun admin lalu langsung diarahkan ke halaman Dashboard. Saat tiba di halaman Dashboard dapat melakukan peralihan ke banyak halaman seperti halaman Tambah Materi, Tambah Kuis, dan Edit Kuis. Pada saat di halaman Edit Kuis, kuis yang dipilih untuk diedit akan dialihkan ke halaman form edit.

### **3.4.2 Unified Model Language (UML)**

UML yang digunakan pada perancangan *website e-learning* yaitu *use case diagram* dan *activity diagram*.

#### **a. Use Case Diagram**



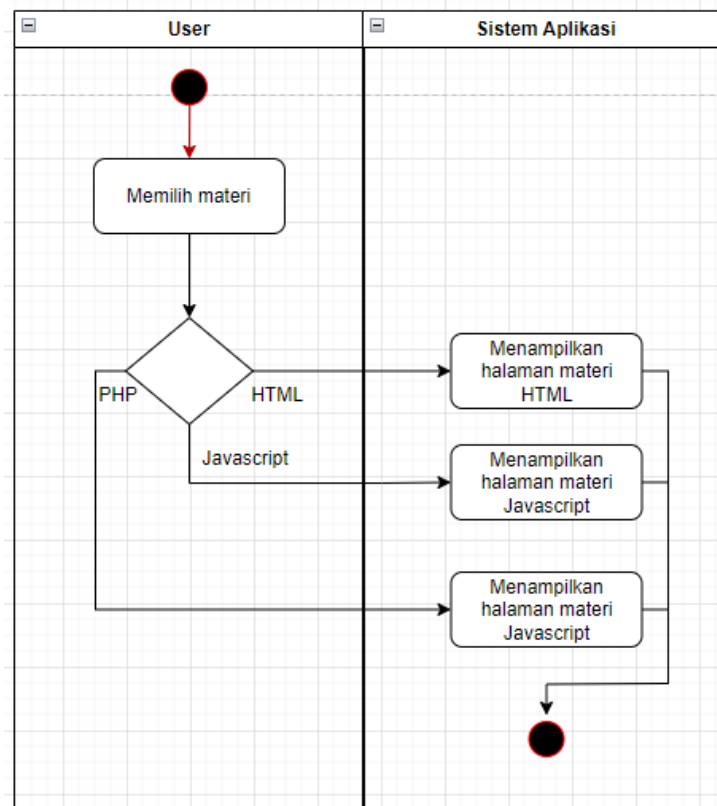
Gambar 3.3 Use Case Diagram

Penjelasan *use case diagram* pada gambar 3.3, terdapat dua aktor yaitu user dan admin. Pada posisi user bisa memasuki materi HTML, Javascript dan PHP. Ketika sudah memasuki materi bahasa yang dipilih, setiap halaman materi yang dipilih baru dapat memasuki Code Editor. Kemudian user dapat memasuki kuis HTML, Javascript dan PHP.

Pada posisi admin, melakukan login terlebih dahulu. Setelah *login* akan diarahkan ke *dashboard*. Pada halaman *dashboard* dapat melihat sejumlah data berisi

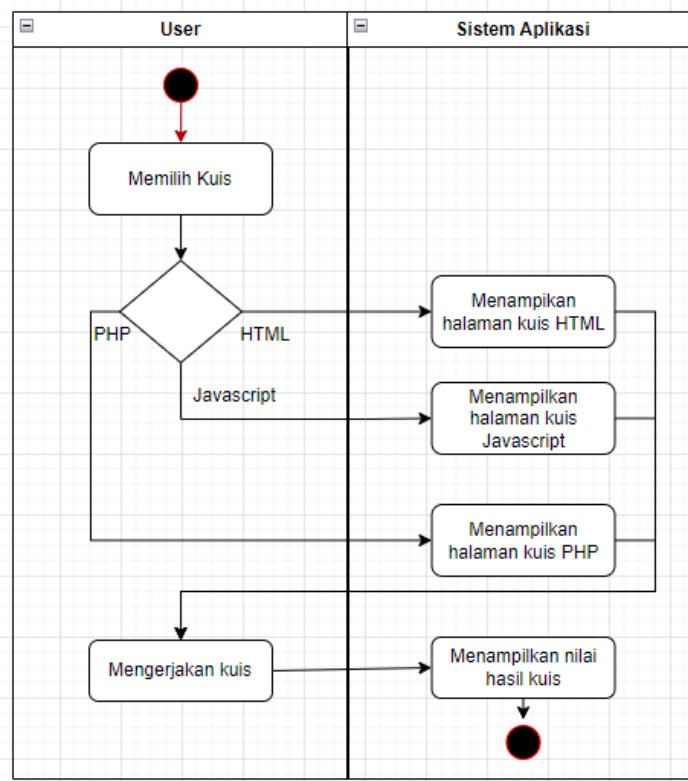
banyak materi dan kuis yang ditampilkan. Admin dari posisi *Dashboard* dapat beralih ke Tambah materi, Tambah kuis, dan Edit kuis.

### b. Activity Diagram



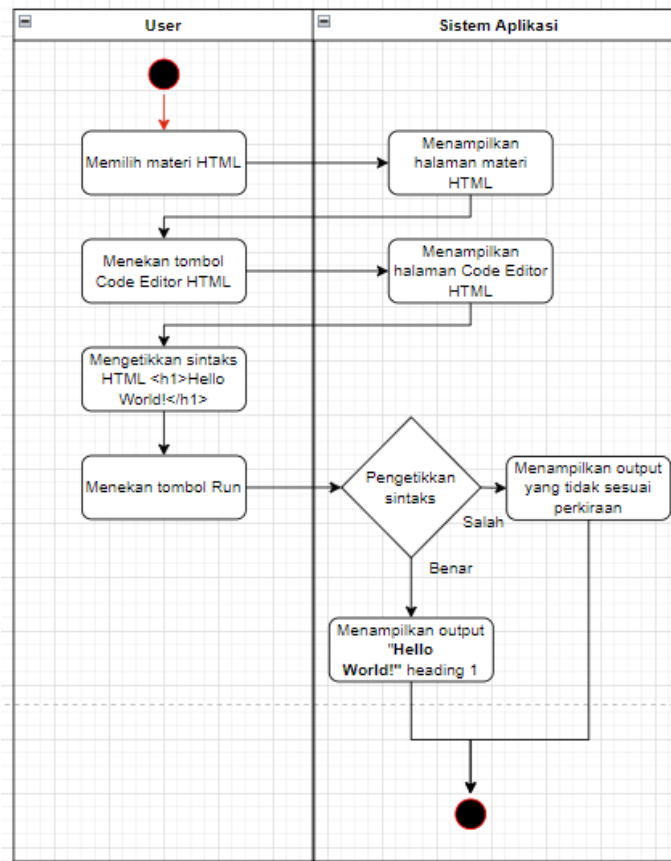
Gambar 3.4 Activity Diagram memilih materi

Penjelasan pada gambar 3.4, merupakan langkah-langkah memasuki halaman materi. Pada halaman Home, disediakan pilihan 3 materi yaitu HTML, Javascript dan PHP. Misalnya, ketika memilih materi HTML akan diarahkan ke halaman intro/default HTML. Lalu setelah itu dapat memilih materi HTML yang tersedia sesuai pada gambar 3.1. Begitu juga dengan Javascript dan PHP.



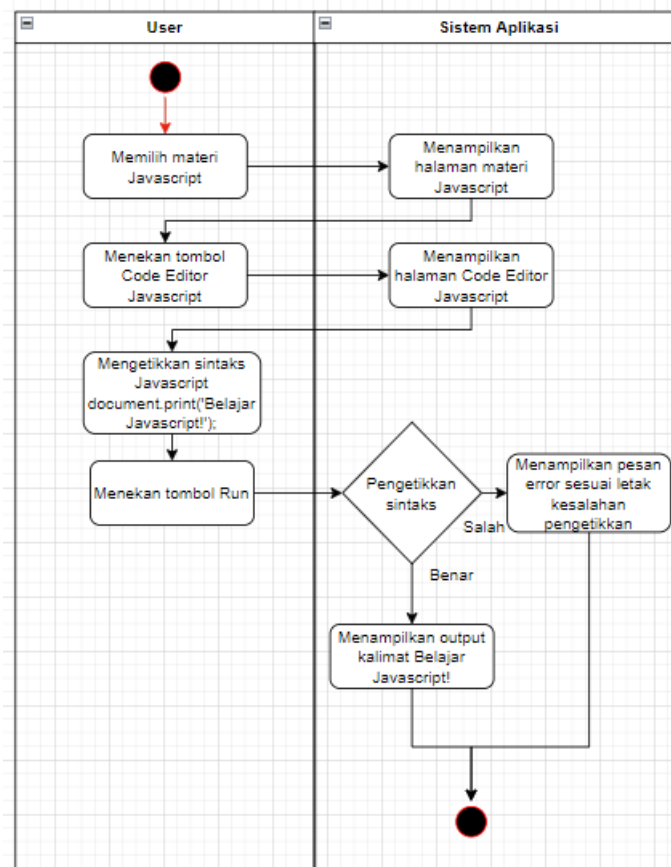
*Gambar 3.5 Activity Diagram kuis*

Penjelasan pada gambar 3.5, merupakan langkah-langkah memasuki halaman kuis. Pada halaman Home, disediakan 3 kuis yaitu HTML, Javascript dan PHP. Misalnya, ketika memilih kuis HTML akan diarahkan ke halaman kuis HTML. Begitu juga dengan Javascript dan PHP.



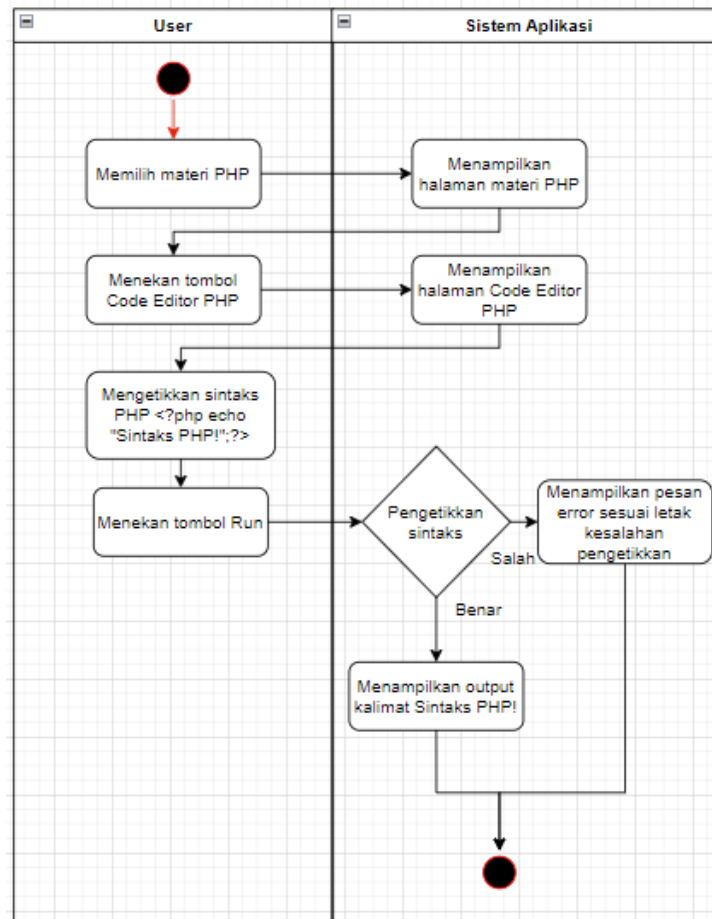
Gambar 3.6 Diagram Activity Code Editor HTML

Penjelasan pada gambar 3.6, merupakan menggunakan *Code Editor* HTML. Pada halaman Home memilih materi HTML, lalu ketika tiba di halaman materi menekan tombol Code Editor HTML. Setelah tiba di halaman Code Editor, mengetikkan sintaks/kode HTML yang sudah dipelajari. Setelah pengetikkan sintaks/kode selesai, menjalankan kode tersebut dengan menekan tombol Run. Setelah itu, sistem memeriksa apakah kode tersebut dibuat secara benar atau tidak. Jika salah, maka akan memunculkan *output* seadanya sesuai pengetikkan kode. Jika benar, maka akan menghasilkan *output* yang sesuai dengan perkiraan.



Gambar 3.7 Diagram Activity Code Editor Javascript

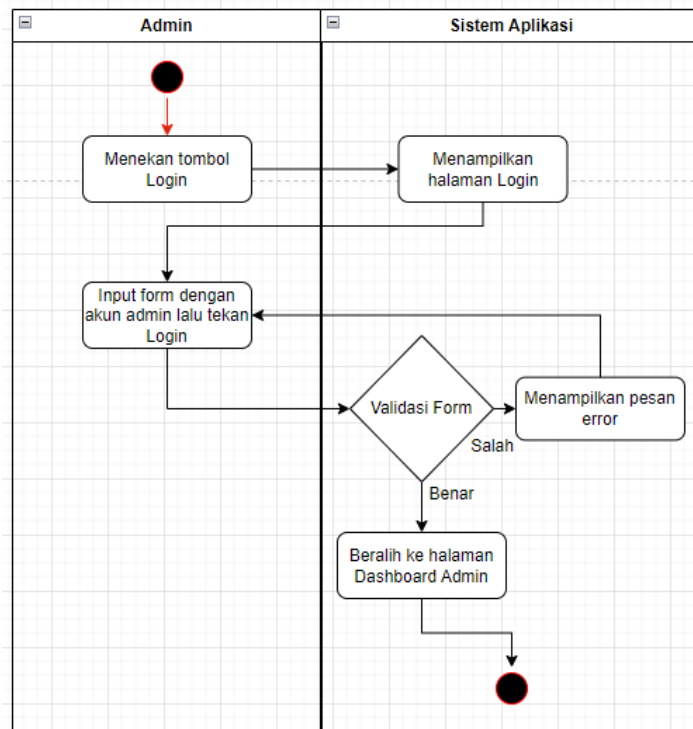
Penjelasan pada gambar 3.7, merupakan menggunakan *Code Editor Javascript*. Pada halaman Home memilih materi Javascript, lalu ketika tiba di halaman materi menekan tombol Code Editor Javascript. Setelah tiba di halaman Code Editor, mengetikkan sintaks/kode Javascript yang sudah dipelajari. Setelah pengetikkan sintaks/kode selesai, menjalankan kode tersebut dengan menekan tombol Run. Setelah itu, sistem memeriksa apakah kode tersebut dibuat secara benar atau tidak. Jika benar, maka akan menghasilkan output yang sesuai dengan perkiraan. Jika salah, maka akan memunculkan pesan *error* sesuai letak kesalahan penulisan kode. Pesan *error* dapat ditemukan pada *output*, jika tidak tampil pada *output* bisa dapat ditemukan pada konsol browser dengan menekan tombol Ctrl+Shift+J.



Gambar 3.8 Activity Diagram Code Editor PHP

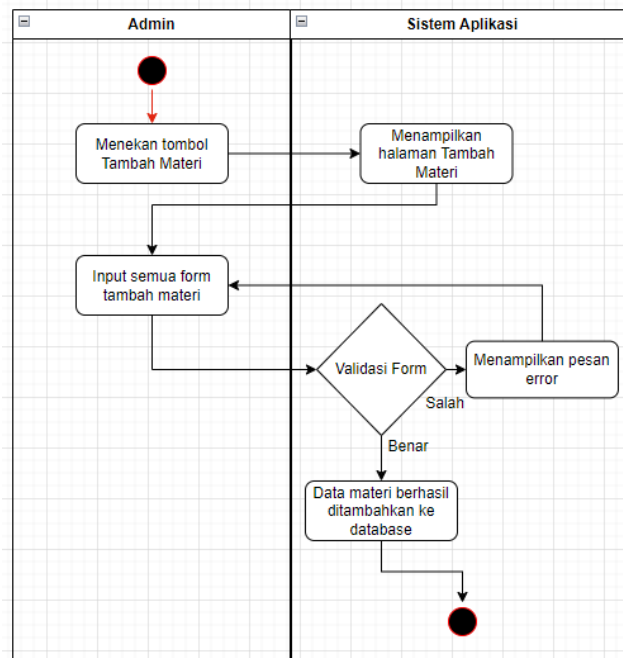
Penjelasan pada gambar 3.8, merupakan menggunakan *Code Editor* PHP. Pada halaman Home memilih materi PHP, lalu ketika tiba di halaman materi menekan tombol Code Editor PHP. Setelah tiba di halaman Code Editor, mengetikkan sintaks/kode PHP yang sudah dipelajari. Setelah pengetikkan sintaks/kode selesai, menjalankan kode tersebut dengan menekan tombol Run. Setelah itu, sistem memeriksa apakah kode tersebut dibuat secara benar atau tidak. Jika salah, maka akan memunculkan pesan *error* pada *output* sesuai letak kesalahan penulisan kode. Jika benar, maka akan menghasilkan *output* yang sesuai dengan perkiraan.





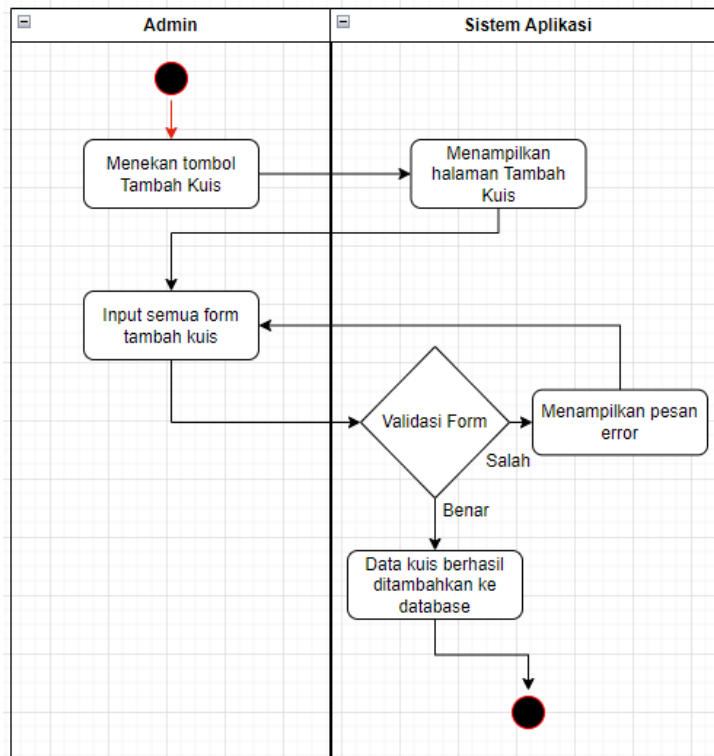
Gambar 3.9 Activity Diagram Login Admin

Penjelasan pada gambar 3.9, merupakan langkah awal pada admin yaitu *login*. Pada saat di halaman Home, menekan tombol *login* lalu akan dialihkan ke halaman Login. Mengisi form *login* dengan akun admin, jika salah maka akan menampilkan pesan *error* dan mengulang pengisian form hingga benar, jika benar maka akan dialihkan ke halaman *default* admin yaitu Dashboard.



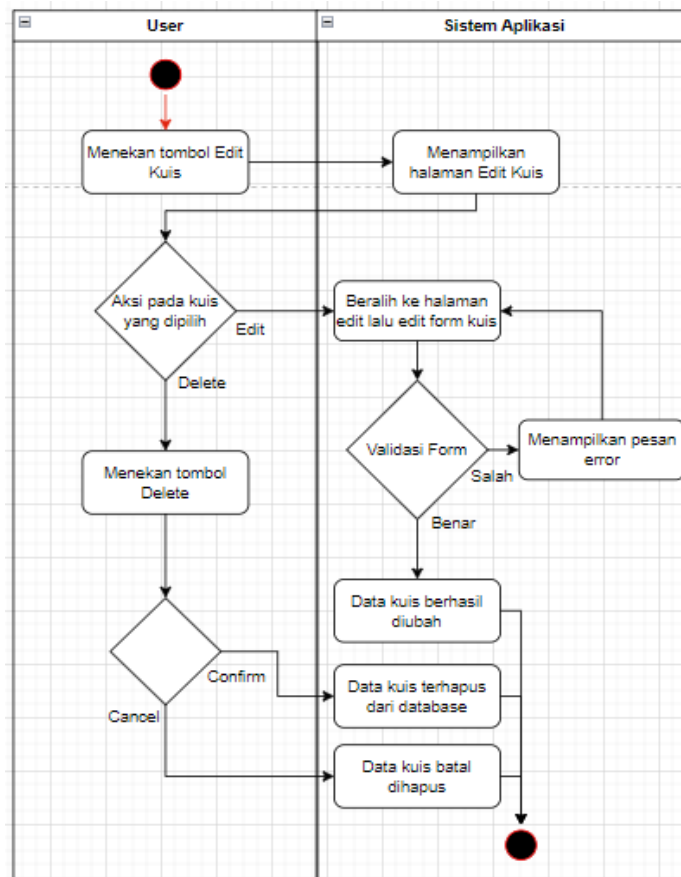
Gambar 3.10 Activity Diagram Tambah Materi

Penjelasan pada gambar 3.10, setelah admin melakukan *login* sesuai gambar *activity diagram* 3.9, menekan tombol tambah materi lalu akan dialihkan ke halaman Tambah Materi. Pada halaman Tambah Materi akan disediakan form tambah materi. Ketika pengisian form tersebut selesai, jika validasi form salah maka akan menampilkan pesan *error* dan mengulang pengisian form tersebut. Jika validasi benar maka menampilkan pesan berhasil dan data tersebut bertambah pada *database*.



Gambar 3.11 Activity Diagram Tambah Kuis

Penjelasan pada gambar 3.11, setelah admin melakukan *login* sesuai gambar *activity diagram* 3.9, menekan tombol tambah kuis lalu akan dialihkan ke halaman Tambah Kuis. Pada halaman Tambah Kuis akan disediakan form tambah kuis. Ketika pengisian form tersebut selesai, jika validasi form salah maka akan menampilkan pesan *error* dan mengulang pengisian form tersebut. Jika validasi benar maka menampilkan pesan berhasil dan data tersebut bertambah pada *database*.



Gambar 3.12 Activity Diagram Edit Kuis

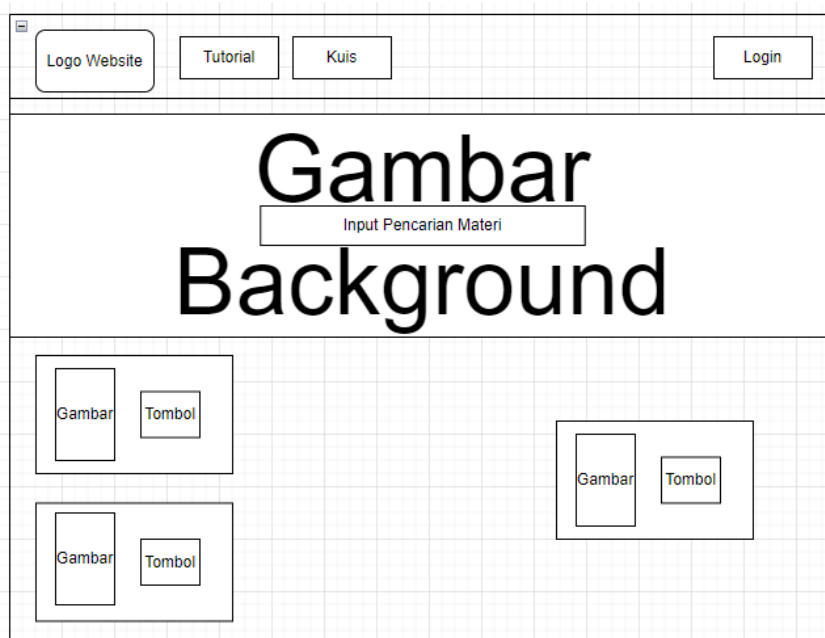
Penjelasan pada gambar 3.12, setelah admin melakukan *login* sesuai gambar *activity diagram* 3.9, menekan tombol *edit* kuis lalu akan dialihkan ke halaman Edit Kuis. Pada halaman Edit Kuis akan menampilkan list kuis sudah disimpan dalam *database*. Setiap kuis tersebut memiliki dua aksi, yaitu *Edit* dan *Delete*. Jika memilih aksi *Delete*, akan menampilkan *pop-up confirm* yang mana jika memilih *Confirm* maka kuis yang dipilih akan terhapus dari *database* dan jika memilih *Cancel* maka kuis batal dihapus. Jika memilih aksi *Edit*, akan diarahkan ke halaman Edit pada kuis yang dipilih. Ketika selesai mengedit form tersebut, jika validasi salah maka akan menampilkan pesan error dan mengulang pengeditan form tersebut. Jika validasi benar maka akan dialihkan ke halaman Edit Kuis dan menampilkan pesan berhasil.

### 3.4.3 Rancangan Tampilan Antarmuka

Perancangan *website* ini dilakukan ini membahas mengenai rancangan dari setiap halaman *website*. Gambaran desain setiap halaman *website* dibuat dengan tujuan untuk mempermudah *programmer* untuk membuat tampilan *website* yang diinginkan.

- **Rancangan Halaman Utama Website**

Rancangan tersebut merupakan halaman utama saat pertama kali memasuki *website e-learning*. Pada *navbar* berisi tombol *dropdown* Materi, *dropdown* Kuis, dan Login. Pada bagian *body website* terdapat input pencarian materi. Di bawah input pencarian terdapat beberapa *section* yang berisikan tombol yang berfungsi sama dengan tombol Materi.

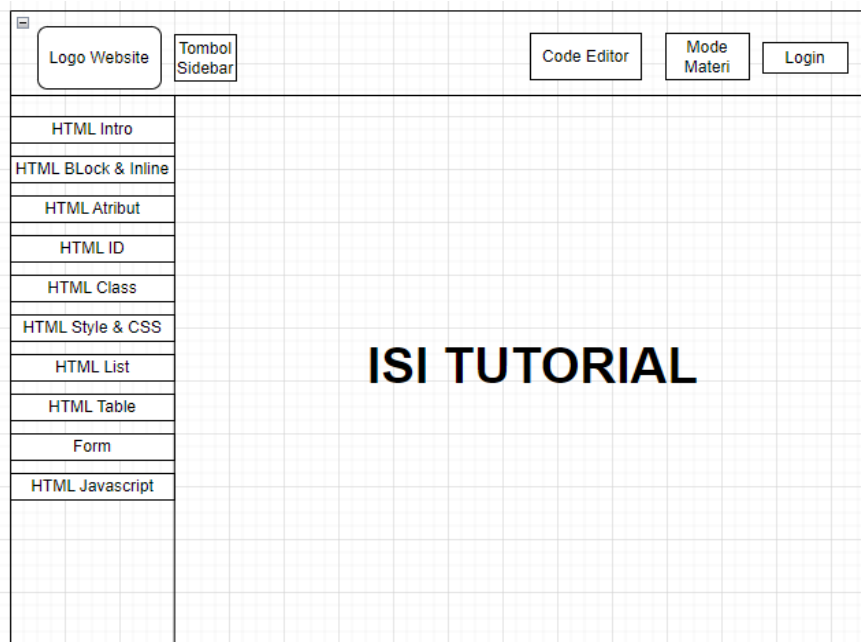


Gambar 3.13 Rancangan Tampilan Halaman Utama

- **Rancangan Tampilan Materi HTML**

Rancangan ini menampilkan materi HTML, pada *navbar* tombol *Sidebar* untuk menutup dan membuka *sidebar*, tombol Code Editor HTML, serta

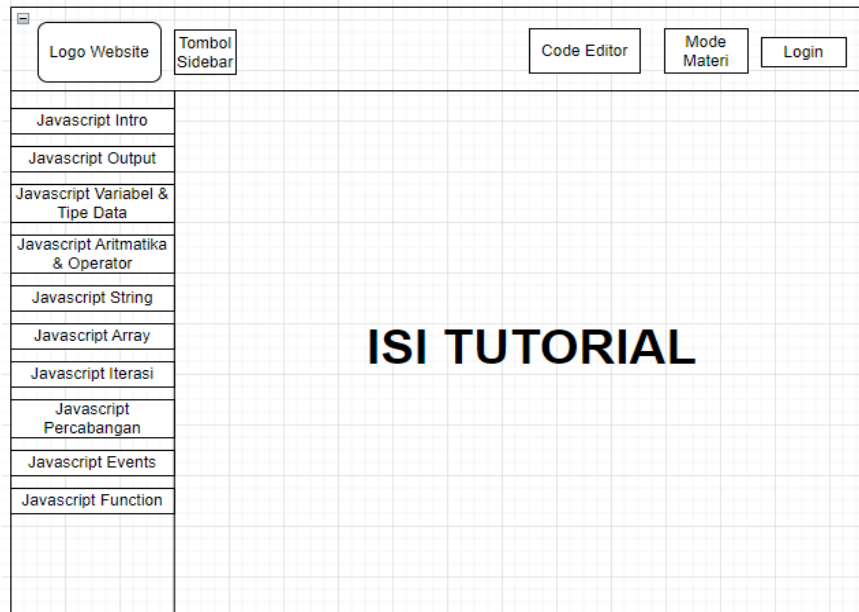
tombol *dropdown* Materi lainnya dan tombol *Login*. Pada bagian sidebar berisikan tombol materi HTML yang sudah ditentukan.



Gambar 3.14 Rancangan Tampilan Materi HTML

- **Rancangan Tampilan Materi Javascript**

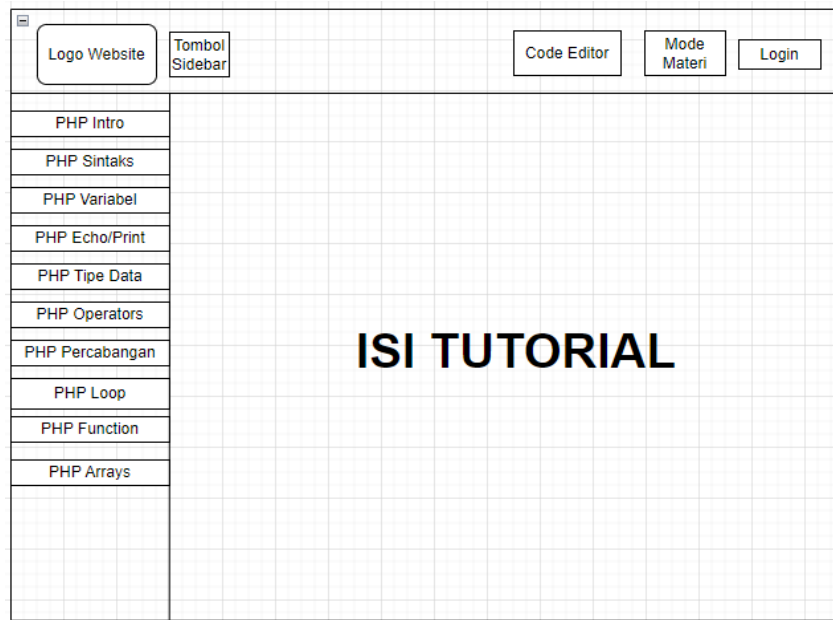
Rancangan ini menampilkan materi Javascript, pada *navbar* tombol *Sidebar* untuk menutup dan membuka *sidebar*, tombol Code Editor Javascript, serta tombol *dropdown* Materi lainnya dan tombol *Login*. Pada bagian sidebar berisikan tombol materi Javascript yang sudah ditentukan.



*Gambar 3.15 Rancangan Tampilan Materi Javascript*

- **Rancangan Tampilan Materi PHP**

Rancangan ini menampilkan materi PHP, pada *navbar* tombol *Sidebar* untuk menutup dan membuka *sidebar*, tombol *Code Editor* PHP, serta tombol *dropdown* Materi lainnya dan tombol *Login*. Pada bagian sidebar berisikan tombol materi PHP yang sudah ditentukan.

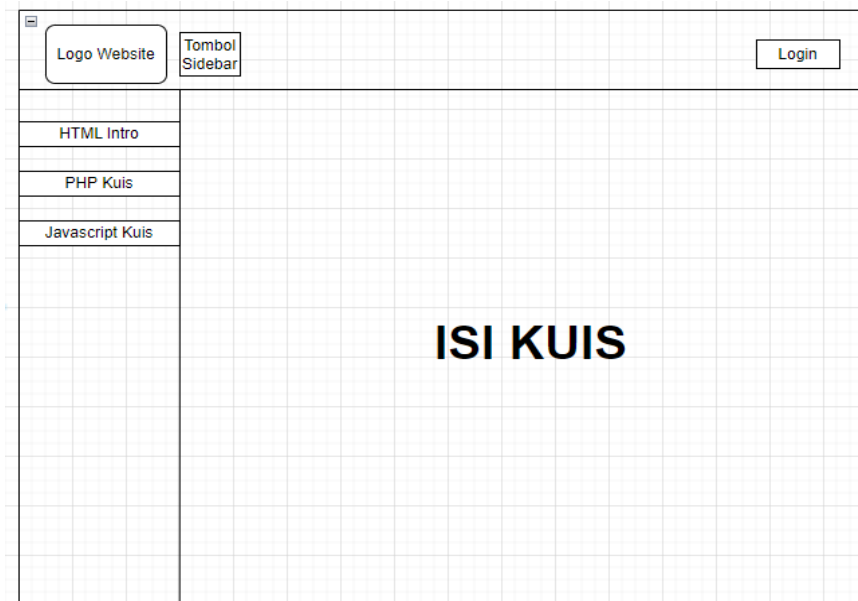


*Gambar 3.16 Rancangan Tampilan Materi PHP*

- Rancangan Tampilan Kuis HTML, Javascript, dan PHP

Rancangan ini menampilkan halaman kuis HTML, Javascript, dan PHP. Namun halaman kuis tersebut dipisahkan pada tombol kuis masing-masing yang terletak di *sidebar* agar dapat berpindah kuis secara langsung ketika ingin berganti kuis bahasa pemrograman.





*Gambar 3.17 Rancangan Tampilan Kuis*

- **Rancangan Tampilan Halaman Registrasi**

Rancangan ini menampilkan form registrasi yang berisi nama, username, email, dan password. Memiliki tombol Daftar dan tombol Login.

The wireframe shows a registration form layout on a grid background. In the top-left corner, there is a box labeled "Logo Website". The word "Gambar" is centered on the left side of the form area. On the right side, there are four input fields stacked vertically: "Input Nama", "Input Username", "Input Email", and "Input Password". Below these fields are two buttons: "Tombol Login" and "Tombol Daftar".

*Gambar 3.18* Rancangan Tampilan Registrasi

- **Rancangan Tampilan Halaman Login**

Rancangan ini menampilkan form login yang berisi username dan password. Memiliki tombol Login dan tombol Daftar jika ingin daftar akun.

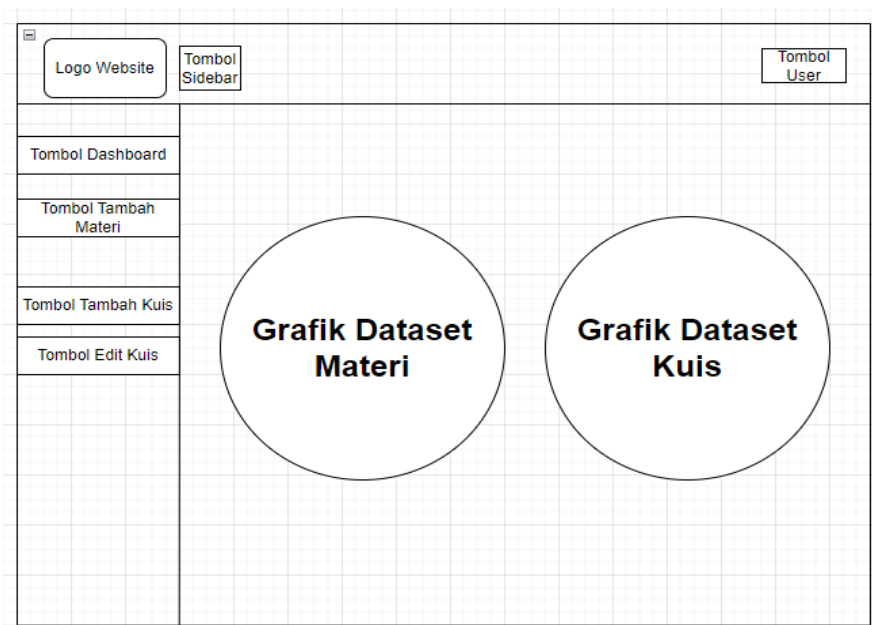
The wireframe shows a login form layout on a grid background. In the top-left corner, there is a box labeled "Logo Website". The word "Gambar" is centered on the left side of the form area. On the right side, there are two input fields stacked vertically: "Input Username" and "Input Password". Below these fields are two buttons: "Tombol Login" and "Tombol Daftar".

*Gambar 3.19* Rancangan Tampilan Login

Pembahasan rancangan halaman kali ini dikhususkan untuk admin. Pada rancangan tampilan admin ini terdapat *navbar* yang berisikan logo, tombol *Navbar* dan tombol User yang berisikan nama admin dan tombol Logout. Bagian kiri menampilkan *sidebar* yang berisikan tombol *Dashboard*, Tambah Materi, Tambah Kuis dan Edit Kuis.

- **Rancangan Tampilan Halaman Dashboard**

Halaman ini paling awal dimasuki setelah admin melakukan *login*. Rancangan ini hanya menampilkan grafik *pie* yang menunjukkan jumlah materi dan kuis yang tersimpan dalam *database*.



Gambar 3.20 Rancangan Tampilan Dashboard

- **Rancangan Tampilan Halaman Tambah Materi**

Rancangan ini menampilkan form tambah materi yang berisikan input Mode, input Judul Materi, tombol Referensi yang bertuliskan tanda + bila ditekan akan menampilkan input Sumber dan input URL, input Isi Materi, Preview Materi, dan tombol Submit.

Logo Website	Tombol Sidebar	Tombol User
Tombol Dashboard	Input Mode Bahasa	Input Judul Materi
Tombol Tambah Materi	+	
	Input Sumber	Input URL
Tombol Tambah Kuis	Input Isi Materi	
Tombol Edit Kuis	Preview Isi Materi	
	Tombol Submit	

*Gambar 3.21* Rancangan Tampilan Tambah Materi

- **Rancangan Tampilan Halaman Tambah Kuis**

Rancangan ini menampilkan form tambah kuis yang berisikan input Soal Mode, input Soal Kuis, input Kunci Jawaban, input Pilihan A, input Pilihan B, input Pilihan C, input Pilihan D, dan tombol Submit.

*Gambar 3.22 Rancangan Tampilan Tambah Kuis*

- **Rancangan Tampilan Halaman Edit Kuis**

Rancangan ini menampilkan tombol filter HTML, PHP, Javascript dan Reset. Di bawah tombol tersebut tersedia *table* yang menampilkan *list* nomor, mode, soal, jawaban, dan aksi. Pada *list* aksi tersebut, tiap data akan menampilkan status ON/OFF, tombol *Edit*, dan tombol *Delete*.

Nomor	Mode	Soal	Jawaban	Aksi
1	HTML	Isi Soal	Isi Jawaban	ON Tombol Edit Tombol Delete

*Gambar 3.23 Rancangan Tampilan Edit Kuis*

- Rancangan Tampilan Halaman Edit

Halaman akan ditampilkan bila menekan tombol *Edit* dari salah satu kuis pada *table* tersebut. Rancangan ini menampilkan form *edit* yang berisikan edit Soal Kuis, edit Kunci Jawaban, edit Pilihan A, edit Pilihan B, edit Pilihan C, edit Pilihan D, tombol ON/OFF Kuis Aktif, dan tombol Submit.

The wireframe shows a web interface for editing a quiz question. It features a sidebar with navigation links and a main content area with a title and several action buttons.

Gambar 3.24 Rancangan Tampilan Edit

#### 3.4.4 Rancangan Database

Rancangan *database* yang akan digunakan untuk menampung semua data yang akan diinput terdiri dari beberapa tabel. Di antaranya tabel user, materi, dan soal.

Table 3.4 Rancangan Tabel User

Nama Field	Type Data	Panjang	Keterangan
id	int	2	Primary Key, Auto Increment
level	int	2	
email	varchar	50	
username	varchar	50	
password	char	128	
name	varchar	50	

*Table 3.5 Rancangan Tabel Materi*

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Panjang</b>	<b>Keterangan</b>
id	int	2	Primary Key, Auto Increment
href	char	20	
page	int	2	
mode	char	20	
judul	char	100	
isi	text	-	
sumber	char	100	
url	text	-	

*Table 3.6 Rancangan Tabel Soal*

<b>Nama Field</b>	<b>Tipe Data</b>	<b>Panjang</b>	<b>Keterangan</b>
id	int	2	Primary Key, Auto Increment
mode	char	20	
soal	varchar	500	
a	varchar	500	
b	varchar	500	
c	varchar	500	
d	varchar	500	
jawaban	varchar	500	
aktif	char	2	



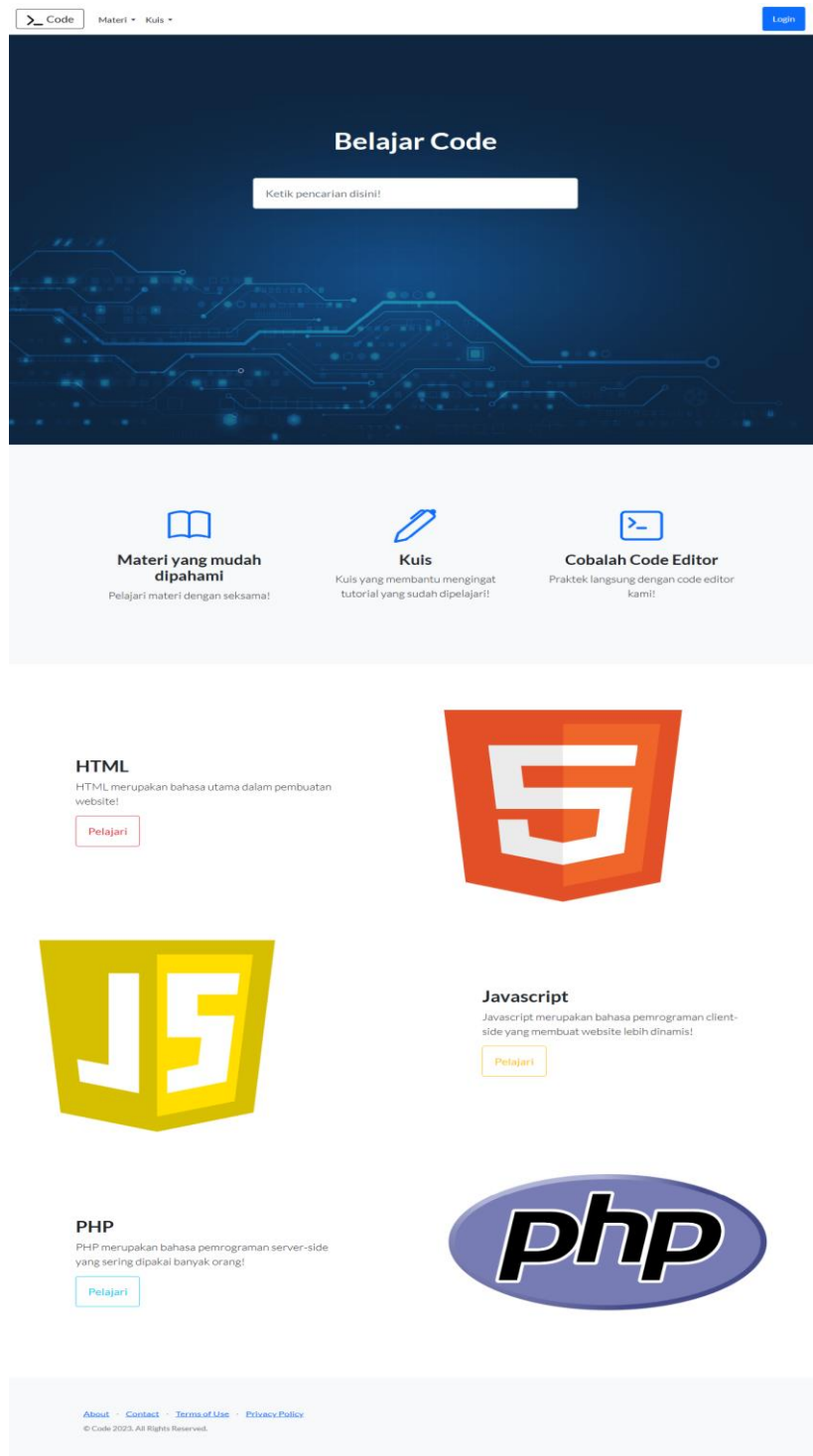
### **3.4 IMPLEMENTASI**

Pada tahap implementasi merupakan memvisualisasikan rancangan yang sudah dibuat pada tahap sebelumnya. Implementasi akan dibahas menjadi 2 bagian halaman, yaitu user dan admin.

#### **3.4.1 Halaman User**

##### **3.4.1.1 Pembuatan Halaman Utama / Home**

Pembahasan ini mengenai pembuatan halaman web yang akan dikunjungi oleh *user* saja dengan eksekusi kode program. Pembuatan halaman tersebut antara lain Home, Materi (HTML, Javascript, dan PHP), Kuis (HTML, Javascript, dan PHP), Code Editor (HTML, Javascript, dan PHP), Registrasi dan Login.



Gambar 3.25 Tampilan Home

Penjelasan fitur yang ada pada halaman Home, pada bagian *Navbar* berisikan tombol Materi yang berisikan 3 bahasa yaitu HTML, PHP, dan Javascript. Ketika menekan dari salah satu bahasa tersebut akan dialihkan ke halaman Materi tersebut sesuai bahasa yang dipilih.

Di samping tombol Materi, terdapat tombol Kuis yang berisikan 3 bahasa juga yaitu HTML, PHP, dan Javascript. Ketika menekan dari salah satu bahasa tersebut akan dialihkan ke halaman Kuis tersebut sesuai bahasa yang dipilih.

Pada bagian bawah *Navbar* terdapat fitur pencarian yang berfungsi untuk menampilkan materi yang sesuai dengan pengetikkan *user* lalu ketika pencarian ditampilkan, dapat dipilih dan ditekan akan langsung diarahkan ke halaman materi yang dituju.

#### **3.4.1.2 Pembuatan Halaman Materi HTML**

Pada halaman Materi HTML di bagian *Navbar* memiliki tombol sidebar yang berfungsi membuka dan menutup *sidebar* yang berisikan judul materi. Dan juga memiliki tombol *Code Editor* HTML bila ditekan akan dialihkan ke halaman *Code Editor* HTML. Lalu di sampingnya memiliki tombol mode lainnya seperti PHP dan Javascript bila ingin berganti bahasa pemrograman yang ingin dipelajari.

Pada bagian kiri adalah *sidebar* yang berisikan judul materi yang bila ditekan akan alihkan ke halaman yang sesuai dengan judul materi yang dipilih. Pada bagian samping *sidebar* yaitu *content* utama yang berisikan materi tersebut.

Code

HTML Tutorial

HTML

HTML Intro

HTML Block & Inline

HTML Atribut

HTML ID

HTML Class

HTML Style & CSS

HTML List

HTML Table

HTML Form

HTML Javascript

Coba HTML

Mode Materi

## HTML Intro

HTML merupakan bahasa markup yang terdiri atas banyak banyak elemen. HTML juga merupakan bahasa standar dalam pembuatan sebuah halaman web.

### Sejarah

HTML (Hypertext Markup Language) adalah bahasa pemrograman web yang pertama kali dikembangkan pada tahun 1990 oleh seorang ilmuwan bernama Tim Berners-Lee di CERN, Swiss. Pada saat itu, HTML hanya memiliki sedikit tag dan fungsionalitas yang sangat terbatas, hanya digunakan untuk membuat dokumen sederhana yang terdiri dari teks dan gambar. Seiring dengan berkembangnya teknologi web, HTML juga mengalami perkembangan yang signifikan. Berikut ini adalah beberapa tahap perkembangan HTML hingga saat ini:

1. HTML 2.0 HTML 2.0 adalah versi pertama dari HTML yang secara resmi diterbitkan pada tahun 1995. Versi ini menambahkan banyak fitur baru seperti tabel, formulir, dan frame.
2. HTML 3.2 HTML 3.2 diterbitkan pada tahun 1997 dan menambahkan beberapa elemen baru seperti bidang input, tombol, dan daftar.
3. HTML 4.01 HTML 4.01 diterbitkan pada tahun 1999 dan merupakan versi HTML yang paling banyak digunakan pada saat itu. Versi ini menambahkan beberapa tag baru seperti label, fieldset, dan legend. Selain itu, HTML 4.01 juga mendukung Cascading Style Sheets (CSS) untuk memformat dokumen web.
4. XHTML (HTML 5) XHTML (Extensible Hypertext Markup Language) diperkenalkan pada tahun 2000 sebagai pengganti HTML 4.01. Versi ini menggunakan aturan sintaks yang lebih ketat dan mendukung XML. HTML 5 secara resmi dirilis pada tahun 2014 dan menambahkan beberapa fitur baru seperti audio, video, dan geolocation.
5. HTML 5.1 dan 5.2 HTML 5.1 dirilis pada tahun 2016 dan menambahkan beberapa fitur baru seperti semantik, aksesibilitas, dan performa yang lebih baik. HTML 5.2 dirilis pada tahun 2017 dan menambahkan beberapa tag baru seperti dialog, menyempurnakan semantik, serta menyempurnakan fitur-fitur yang sudah ada di HTML 5.

### Struktur

Struktur HTML setiap dokumen HTML harus dimulai dengan tag `<html>` dan diakhiri dengan tag `</html>`. Semua elemen HTML harus diletakkan di antara tag ini. Di dalam tag `<html>`, terdapat dua bagian utama yaitu `<head>` dan `<body>`. Bagian `<head>` digunakan untuk menambahkan meta data dan informasi penting lainnya, sedangkan bagian `<body>` digunakan untuk menambahkan konten halaman web.

```
<!DOCTYPE html>
<html>
<head>
  <title>Judul Halaman Web</title>
</head>
<body>
  <h1>Selamat Datang</h1>
  <p>Ini adalah paragraf.</p>
</body>
</html>
```

Hasil

**Selamat Datang.**

Ini adalah paragraf.

Penjelasan elemen HTML di atas:

```
<!DOCTYPE html> Mendeklarasikan atau mendefinisikan dokumen tersebut dengan HTML5.
<html> Elemen yang menampung semua elemen HTML.
<head> Menyimpan meta informasi pada halaman web seperti author, description, pengaturan bahasa yang digunakan serta penggunaan karakter encoding dan lain-lain.
<title> Judul pada halaman web.
<body> Berisikan tubuh atau konten web.
<h1> Pembuatan heading.
<p> Pembuatan kalimat paragraf.
```

Elemen HTML terdiri dari tag pembuka dan tag penutup. Tag pembuka dimulai dengan tanda kurung siku `<` dan diakhiri dengan tanda kurung siku `>`. Tag penutup dimulai dengan tanda kurung siku `<` dan diakhiri dengan tanda kurung siku `/` dan `>`. Beberapa elemen HTML tidak membutuhkan tag penutup seperti `<img>`, `<br>` dan sebagainya.

### Semantik

HTML juga memiliki tag elemen semantik yang membantu dalam mengorganisir konten halaman web dengan lebih baik. Tag elemen semantik meliputi `header`, `nav`, `section`, `article`, `aside`, `footer`, dan lainnya. Setiap tag elemen semantik memiliki arti dan tujuan tertentu.

```
<header>
  <nav>
    <section>
      <article>
        <footer>
```

Hasil

**Logo dan judul halaman web**

- Beranda
- Tentang
- Kontak

**Judul Artikel**

Isi artikel...

**Iklan**

Isi iklan...

Hak Cipta © 2023 Halaman Web Saya

Referensi

W3schools (Intro)

W3schools (Semantic)

Copyright © Code 2023

Privacy Policy Terms & Conditions

Gambar 3.26 Tampilan Materi HTML Halaman 1

### 3.4.1.3 Pembuatan Halaman Materi Javascript

Pada halaman Materi Javascript di bagian *Navbar* memiliki tombol sidebar yang berfungsi membuka dan menutup *sidebar* yang berisikan judul materi. Dan juga memiliki tombol *Code Editor* Javascript bila ditekan akan dialihkan ke halaman *Code Editor* Javascript. Lalu di sampingnya memiliki tombol mode lainnya seperti HTML dan PHP bila ingin berganti bahasa pemrograman yang ingin dipelajari.

Pada bagian kiri adalah *sidebar* yang berisikan judul materi yang bila ditekan akan alihkan ke halaman yang sesuai dengan judul materi yang dipilih. Pada bagian samping *sidebar* yaitu *content* utama yang berisikan materi tersebut.

Code

JavaScript Tutorial

JavaScript

JavaScript Intro

JavaScript Output

JavaScript Variabel & Tipe Data

JavaScript Aritmatika & Operator

JavaScript String

JavaScript Array

JavaScript Iterasi

JavaScript Percabangan

JavaScript Events

JavaScript Function

Coba JS

Mode Materi

## Javascript Output

JavaScript memiliki beberapa cara untuk menghasilkan output atau tampilan ke layar. Output di JavaScript dapat digunakan untuk menampilkan informasi, memberikan feedback kepada pengguna, atau memecah bagian-bagian kode agar lebih mudah dipahami. Berikut ini beberapa cara untuk menghasilkan output di JavaScript.

### Console Log

Metode `console.log()` adalah cara paling umum untuk menghasilkan output di JavaScript. Metode ini digunakan untuk mencetak informasi ke konsol browser atau terminal. Kita dapat menggunakan `console.log()` untuk mencetak nilai variabel atau pesan teks.

```
console.log(5+7);
```

Hasil

12

Untuk dapat membuka konsol atau terminal pada browser cukup tekan tombol F12 pada keyboard anda.

Console Browser

### Document Write

Metode `document.write()` digunakan untuk menulis HTML atau teks ke dalam dokumen web. Metode ini dapat digunakan untuk menampilkan pesan teks di halaman web.

```
<h1>Document Write</h1>
<p>Ini paragraf.</p>

<script>
document.write("Hello World!");
</script>
```

Hasil

## Document Write

Ini paragraf.

Hello World!

`document.write` dijalankan setelah seluruh kode HTML dimuat.

### Inner HTML

Metode `innerHTML` digunakan untuk menetapkan isi dari elemen HTML. Metode ini dapat digunakan untuk mengganti isi elemen HTML yang sudah ada atau menambahkan elemen HTML baru ke halaman web.

```
<h1>Inner HTML</h1>
<p id="test"></p>

<script>
document.getElementById("test").innerHTML = "Paragraf ini menggunakan inner HTI";
</script>
```

Hasil

## Inner HTML

Paragraf ini menggunakan inner HTML

### Pop Up (Alert)

Metode `alert()` digunakan untuk menampilkan pesan pop-up di jendela browser. Metode ini sering digunakan untuk memberikan feedback kepada pengguna atau memberikan peringatan.

```
<h1>Alert Output</h1>
<p>Tekan tombol di bawah!</p>
<button onclick="popup()">Klik disini!</button>

<script>
function popup(){
alert("Username atau Password anda salah!");
}
</script>
```

Hasil

## Alert Output

Tekan tombol di bawah!

Klik disini!

Referensi

[W3schools](#)

Gambar 3.27 Tampilan Materi Javascript Halaman 2

#### **3.4.1.4 Pembuatan Halaman Materi PHP**

Pada halaman Materi PHP di bagian *Navbar* memiliki tombol sidebar yang berfungsi membuka dan menutup *sidebar* yang berisikan judul materi. Dan juga memiliki tombol *Code Editor* PHP bila ditekan akan dialihkan ke halaman *Code Editor* PHP. Lalu di sampingnya memiliki tombol mode lainnya seperti HTML dan Javascript bila ingin berganti bahasa pemrograman yang ingin dipelajari.

Pada bagian kiri adalah *sidebar* yang berisikan judul materi yang bila ditekan akan alihkan ke halaman yang sesuai dengan judul materi yang dipilih. Pada bagian samping *sidebar* yaitu content utama yang berisikan materi tersebut.

Code

PHP Tutorial

PHP

PHP Intro

PHP Sintaks

PHP Variabel

PHP Echo/Print

PHP Tipe Data

PHP Operators

PHP Percabangan

PHP Loop

PHP Function

PHP Arrays

Coba PHP

Mode Materi

## PHP Variabel

Variabel merupakan salah satu konsep dasar dalam pemrograman. Variabel digunakan untuk menyimpan nilai yang dapat diakses dan diubah selama proses eksekusi program. Dalam PHP, Anda dapat membuat variabel untuk menyimpan berbagai jenis data, termasuk angka, teks, array, dan objek.

### Penulisan Variabel

- Variabel dalam PHP diawali dengan tanda dolar (\$), diikuti dengan nama variabel.
- Variabel harus diawali dengan huruf atau underscore (\_).
- Variabel tidak bisa diawali dengan angka.
- Nama variabel bersifat case-sensitive, yang berarti \$nama dan \$Nama dianggap sebagai dua variabel yang berbeda.
- Contoh penulisan variabel yang valid: \$nama, \$umur, \$jumlah\_barang.

### Variabel Scope

Variabel scope merujuk pada wilayah atau bagian program di mana sebuah variabel dapat diakses dan digunakan. Dalam PHP, terdapat beberapa jenis scope yang mempengaruhi ketersediaan variabel, yaitu scope global, scope lokal, dan scope statis.

#### Scope Global

```
// Scope global
$name = "John Doe"; // Variabel global

function tampilkanNama() {
    global $name; // Memanggil variabel global di dalam fungsi
    echo $name;
}

tampilkanNama();
```

Hasil

John Doe

#### Scope Lokal

Variabel lokal adalah variabel yang hanya dapat diakses di dalam blok kode tertentu di mana variabel tersebut dideklarasikan. Variabel lokal dideklarasikan di dalam fungsi atau metode, dan tidak dapat diakses dari luar blok tersebut.

```
// Scope lokal
// Variabel lokal
echo $umur;
}

tampilkanUmur();
// Jika $umur dipanggil di luar scope function tidak akan menghasilkan output
?>
```

Hasil

25

#### Scope Statis

Variabel statis adalah variabel yang mempertahankan nilainya di antara pemanggilan fungsi. Variabel statis dideklarasikan di dalam fungsi dan nilainya akan diingat setelah fungsi selesai dieksekusi. Ketika fungsi dipanggil kembali, variabel statis akan menyimpan nilainya dari pemanggilan sebelumnya.

```
// Scope static
// Variabel static
$jumlah++;
echo $jumlah;
}

hitungJumlah(); // Output : 1
echo "
"; // Membuat baris ke bawah
hitungJumlah(); // Output : 2
echo "
";
hitungJumlah(); // Output : 3
?>
```

Hasil

1
2
3

Referensi

W3schools

Scope Variabel

PHP

Copyright © Code 2023

[Privacy Policy](#) · [Terms & Conditions](#)

Gambar 3.28 Tampilan Data Materi Materi PHP Halaman 3



#### **3.4.1.5 Pembuatan Halaman Kuis**

Pada halaman ini memiliki fitur kuis sederhana pada bagian layar utama monitor. Cukup memilih pada option yang dianggap benar di setiap soal tetapi harus diisi semua agar dapat disubmit oleh tombol Submit dan dapat menampilkan nilai kuis. Pada bagian samping atau *sidebar* memiliki tombol kuis bahasa pemrograman yang dapat dipilih.

Code

HTML Kuis

PHP Kuis

Javascript Kuis

## HTML Kuis

- 1.Tag HTML yang digunakan untuk membuat heading adalah <...>
  - ☐ html
  - ☐ head
  - ☐ title
  - ☐ h1
- 2.Tag yang menyimpan informasi halaman web adalah <...>
  - ☐ script
  - ☐ body
  - ☐ html
  - ☐ head
- 3.Tag <title> berfungsi untuk
  - ☐ Tag untuk memberi judul pada dokumen HTML
  - ☐ Tag untuk membuat link
  - ☐ Tag untuk membuat heading
  - ☐ Tag untuk membuat paragraf
- 4.Tag yang berfungsi untuk pembuatan list huruf atau angka adalah <...>
  - ☐ li
  - ☐ ul
  - ☐ ol
  - ☐ select
- 5.Tag yang berfungsi untuk pembuatan list berupa simbol adalah <...>
  - ☐ ol
  - ☐ ul
  - ☐ li
  - ☐ select
- 6.<a href="">, Yang merupakan atribut adalah...
  - ☐ <a href="">
  - ☐ a
  - ☐ href
  - ☐ ""
- 7.Salah satu jenis type input pada HTML 5, kecuali...
  - ☐ chekckbox
  - ☐ text
  - ☐ password
  - ☐ submit
- 8.Pembuatan design halaman web dilakukan pada tag <...>
  - ☐ script
  - ☐ img
  - ☐ style
  - ☐ iframe
- 9.Selector class pada CSS dan tag style berupa...
  - ☐ #
  - ☐ \$
  - ☐ .
  - ☐ !
- 10.Pembuatan bahasa pemrograman Javascript dibuat pada tag <...>
  - ☐ link
  - ☐ iframe
  - ☐ script
  - ☐ head

Submit

Copyright © Code 2023

[Privacy Policy](#) · [Terms & Conditions](#)

*Gambar 3.29* Tampilan Halaman Kuis HTML

Code

HTML Kuis

PHP Kuis

Javascript Kuis

## Javascript Kuis

- Penulisan code Javascript pada HTML ditempatkan pada tag <...>
  - ☐ script
  - ☐ noscript
  - ☐ code
  - ☐ button
- Deklarasi variabel pada Javascript, kecuali...
  - ☐ var
  - ☐ let
  - ☐ const
  - ☐ string
- Output pada Javascript adalah...
  - ☐ console.log();
  - ☐ print;
  - ☐ print\_r
  - ☐ echo;
- ```
var x = 5;
var y = 10;
console.log(x + y);
```

Output yang ditampilkan adalah...

  - ☐ 510
  - ☐ Error
  - ☐ 15
  - ☐ 5
- Fungsi dari typeof () adalah...
  - ☐ Menampilkan Output
  - ☐ Membuat alert
  - ☐ Pengecekan tipe data pada variabel
  - ☐ Sebuah variabel
- ```
var x = 10;
if (x > 5) {
  console.log("x is greater than 5");
} else{
  console.log("x is less than or equal to 5");
}
```

Output yang ditampilkan adalah...

  - ☐ Error
  - ☐ x is greater than 5
  - ☐ x is less than or equal to 5
  - ☐ undefined
- ```
function Hello(){
  console.log('Hello World');
}
```

Pemanggilan function yang benar adalah...

  - ☐ Hello;
  - ☐ hello();
  - ☐ Hello();
  - ☐ function Hello()
- ```
arr = ["Budi"];

console.log(arr.length);
```

Output yang ditampilkan adalah...

  - ☐ 1
  - ☐ undefined
  - ☐ 0
  - ☐ null
- ```
for (let i = 0; i < 5; i++) {
  console.log(1);
}
```

Output yang ditampilkan adalah...

  - ☐ 1, 2, 3, 4, 5
  - ☐ 0, 1, 2, 3, 4
  - ☐ 1, 3, 5, 7, 9
  - ☐ 1, 2, 3, 4, 5
- Sintaks yang digunakan untuk mendapatkan elemen HTML di JavaScript adalah...
  - ☐ document.getElementsByClassName()
  - ☐ document.querySelector()
  - ☐ document.getElementById()
  - ☐ document.getElementsByTagName()

Submit

Copyright © Code 2023

[Privacy Policy](#) · 
 [Terms & Conditions](#)

*Gambar 3.30 Tampilan Halaman Kuis Javascript*

Code

HTML Kuis

PHP Kuis

Javascript Kuis

## PHP Kuis

1. Kepanjangan dari PHP pada awalnya adalah...

☐ Hypertext Preprocessor  
☐ Person Home Page  
☐ Personal Home Page  
☐ Hyperlink Preprocessor

2. Sintaks `echo` digunakan untuk...

☐ Membaca File  
☐ Menampilkan Output  
☐ Menampilkan tanggal  
☐ Mengambil Output

3. `= 'Budi';` Penulisan variabel nama pada PHP yang benar adalah...

☐ nama  
☐ \$nama  
☐ \_nama  
☐ ?nama

4. `$a = 1; $b = 2; $c = $a*$b; echo $c;`

Output yang dihasilkan...

☐ 12  
☐ 1  
☐ 2  
☐ Error

5. `$nilai = 75; if ($nilai >= 100){ echo 'A'; } elseif ($nilai >= 75){ echo 'B'; } elseif ($nilai >= 50){ echo 'C'; } elseif ($nilai >= 25){ echo 'D'; }`

Output yang ditampilkan adalah...

☐ A  
☐ B  
☐ C  
☐ D

6. `switch ($pilihan){ case "A": echo 80; break; case "B": echo 70; break; default: echo 'Pilihan tidak valid'; break; }`

Jika \$pilihan bernilai 85, outputnya adalah...

☐ 80  
☐ undefined  
☐ 70  
☐ Pilihan tidak valid

7. Berikut adalah contoh syntax untuk melakukan looping pada PHP, kecuali:

☐ while  
☐ switch  
☐ for  
☐ foreach

8. `$i=0; while($i<5){ echo $i; $i++; }`

Output yang ditampilkan adalah...

☐ 1234  
☐ 12345  
☐ 01234  
☐ 012345

9. `$colors = array("red", "green", "blue"); echo count($color);`

Output yang dihasilkan...

☐ 1  
☐ 2  
☐ 3  
☐ 4

10. `$numbers = array(1, 3, 5, 7, 9); echo $numbers[2];`

Output yang dihasilkan...

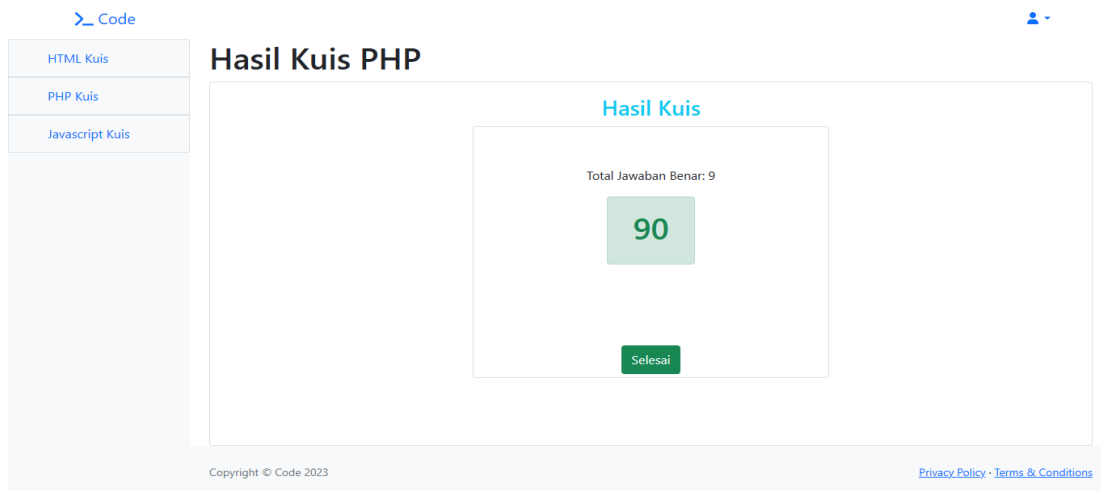
☐ 1  
☐ 3  
☐ 5  
☐ 7

Submit

Copyright © Code 2023

[Privacy Policy](#) · [Terms & Conditions](#)

*Gambar 3.31* Tampilan Halaman Kuis PHP



*Gambar 3.32* Tampilan Halaman Result

Halaman ini dapat ditampilkan ketika kuis yang sudah dikerjakan.

#### **3.4.1.6 Pembuatan Halaman Code Editor**

Pada halaman *Code Editor*, di bagian *Navbar* berisikan tombol Run yang berfungsi menjalankan kode program ketika selesai mengetik kode program. Di samping tombol Run ada tombol pilihan yang berfungsi mengubah tema latar pengetikkan kode program.

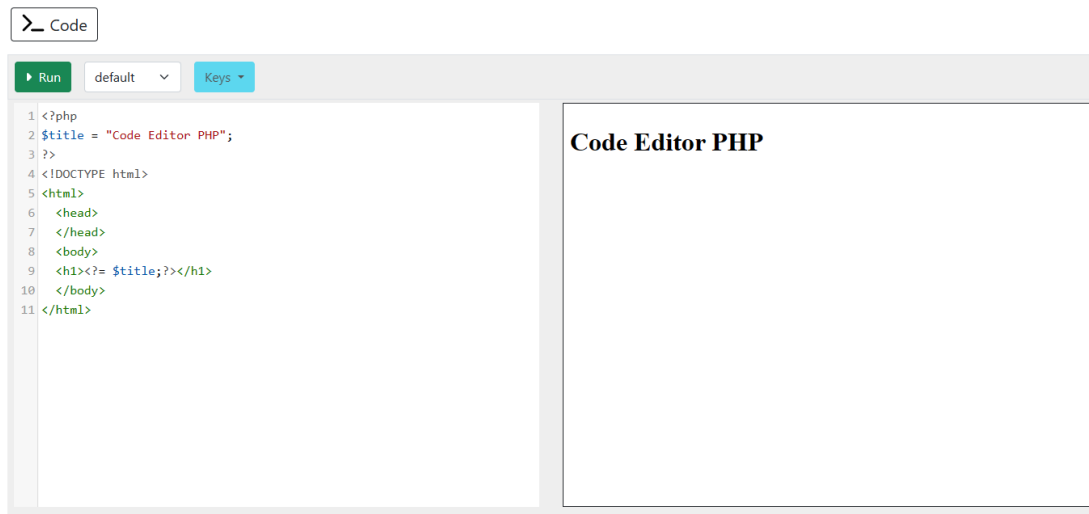
Di bagian bawah *Navbar* berisikan pengetikkan kode program atau Text Editor. Di samping *Text Editor* terdapat output program yang berfungsi menampilkan hasil dari kode program yang dijalankan oleh tombol Run.



*Gambar 3.33 Tampilan Halaman Code Editor HTML*



*Gambar 3.34 Tampilan Halaman Code Editor Javascript*



Gambar 3.35 Tampilan Halaman *Code Editor PHP*

#### 3.4.1.7 Pembuatan Halaman Register

Pada halaman Register, berisikan form untuk daftar akun. Di bawah form terdapat tombol Daftar bila *form* sudah diisi semua lalu menekan tombol Daftar, maka akun tersebut bisa dipakai untuk kebutuhan fitur yang akan mendatang.

The screenshot shows a web application interface for a registration page. At the top, there is a button labeled 'Code' with a code icon. Below it, the title 'DAFTAR AKUN!' is displayed in a large, bold, black font. To the left of the form is an illustration of a person sitting cross-legged, using a laptop, with a gear icon and a code icon nearby. Below the illustration is the text 'Coding Vectors by Vecteezy'. To the right of the illustration is a registration form with the following fields:

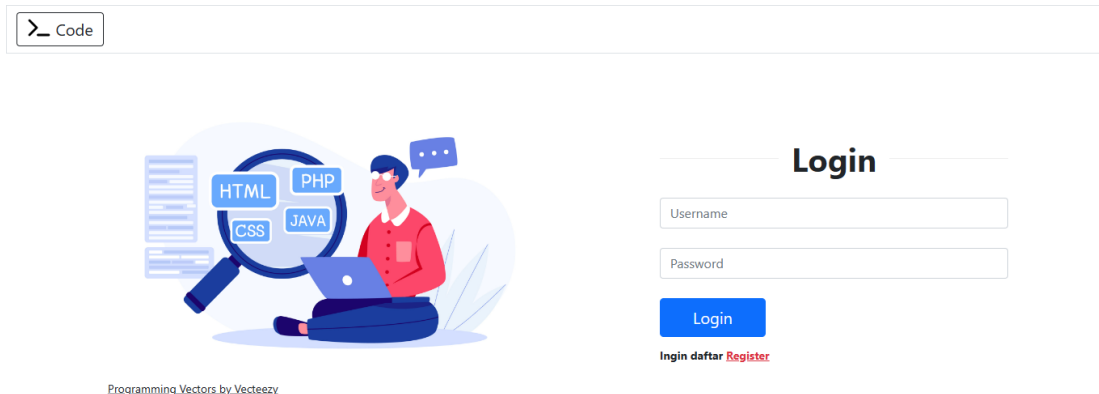
- Nama
- Username
- Email
- Password

Below the form fields, there is a link that says 'Login jika sudah punya akun!'. At the bottom of the form is a blue button labeled 'Daftar'.

Gambar 3.36 Tampilan Halaman Register

### 3.4.1.8 Pembuatan Halaman Login

Pada halaman Login berisikan *form username* dan *password*. *Form* tersebut diisi dengan benar sesuai dengan pendaftaran akun sebelumnya lalu menekan tombol Login, maka akan langsung diarahkan ke halaman Home.



Gambar 3.37 Tampilan Halaman Login

### 3.4.2 Halaman Admin

Pembahasan ini mengenai pembuatan web yang akan dikunjungi oleh admin dengan eksekusi kode program. Pembuatan halaman tersebut lain Dashboard, Tambah Materi, Tambah Kuis dan Edit Kuis.

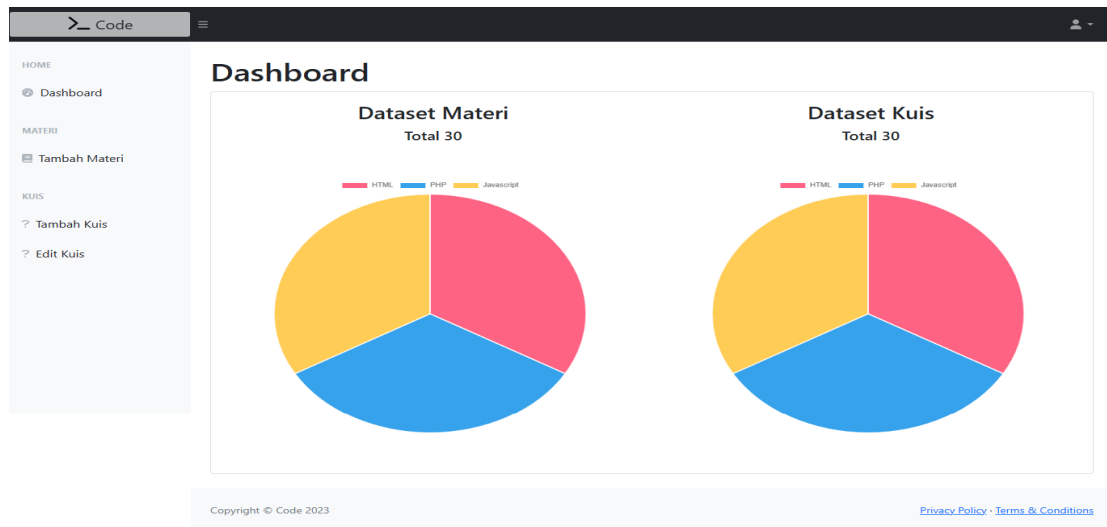
Pada tiap halaman yang sudah disebutkan di atas, dihubungkan pada folder *template* yang bertujuan mengurangi penggunaan kode program yang berulang. Folder *template* tersebut berisikan *head*, *footer* dan *sidebar*.

### 3.6.1 Pembuatan Halaman Dashboard

Pada halaman Dashboard, di bagian halaman tengah hanya menampilkan jumlah dataset Materi dan Kuis yang sudah disimpan pada *database*.



Pada bagian *sidebar*, berisikan tombol halaman Tambah Materi, Tambah Kuis dan Edit Kuis. *Sidebar* ini akan muncul terus pada halaman lainnya.



Gambar 3.38 Tampilan Halaman Dashboard

### 3.6.2 Pembuatan Halaman Tambah Materi

Pada halaman ini berisikan form untuk membuat materi baru.

The screenshot shows the 'Tambah Materi' page. The sidebar is identical to the dashboard. The main content area has a title 'Tambah Materi' and a form with the following elements:
 

- Mode Bahasa:** A dropdown menu with 'Bahasa...' selected.
- Judul Materi:** A text input field.
- Referensi:** A text input field with a '+' icon for adding more references.
- Isi Materi:** A rich text editor with a toolbar containing icons for bold, italic, underline, link, unlink, list, and other formatting options.
- Preview Materi:** A section with a 'Preview Materi' icon and a large text area for previewing the content.
- Tambah Materi:** A green button at the bottom of the form.

 The footer is the same as the dashboard page.

Gambar 3.39 Tampilan Halaman Tambah Materi

### 3.6.3 Pembuatan Halaman Tambah Kuis

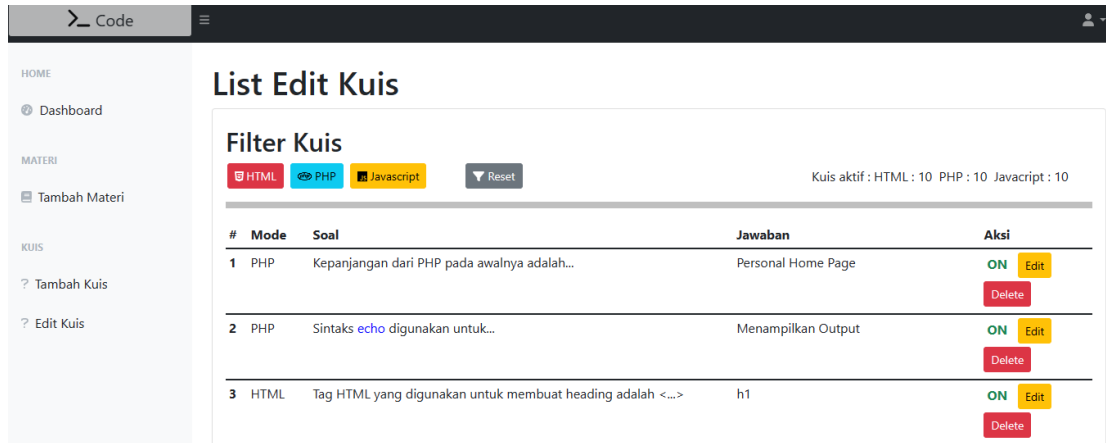
Pada halaman ini berisikan form untuk membuat atau menambahkan kuis baru.

Gambar 3.40 Tampilan Halaman Tambah Kuis

### 3.6.4 Pembuatan Halaman Edit Kuis

Pada halaman ini, berisikan 3 tombol Filter untuk menyaring kuis yang sudah dibuat berdasarkan bahasa yang dipilih serta 1 tombol Reset untuk menghilangkan filter sebelumnya.

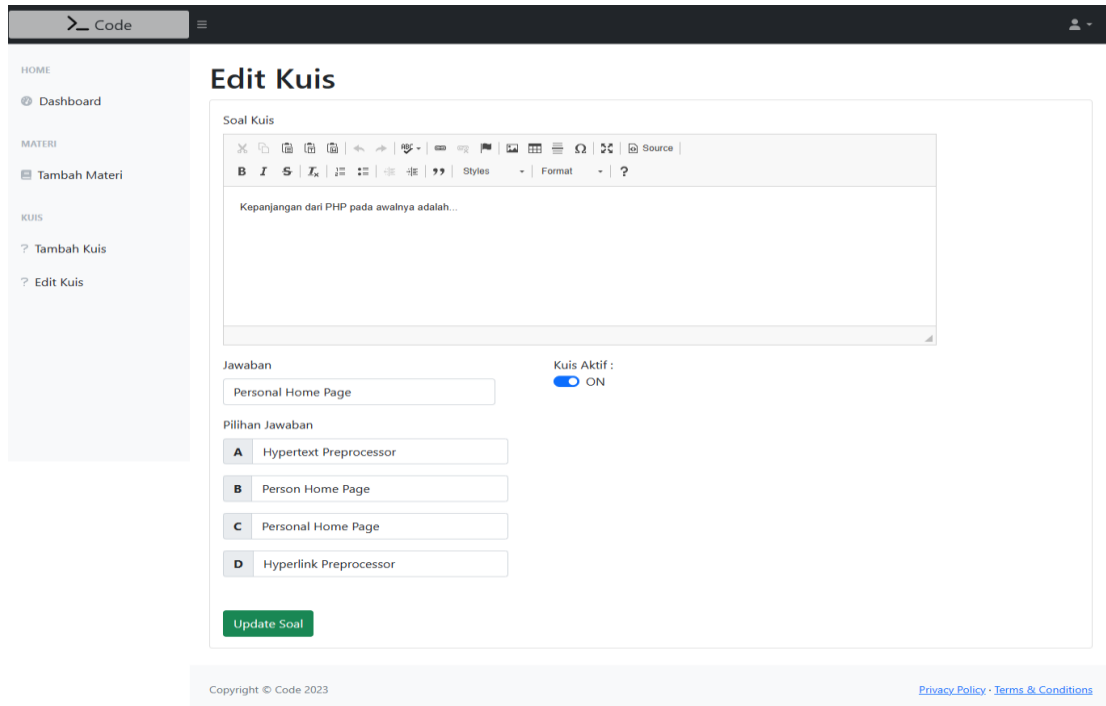
Lalu di bagian table berisikan kuis yang sudah dibuat sebelumnya, dan pada bagian baris Aksi ada 2 buah tombol yaitu Edit dan Delete. Tombol Edit berfungsi untuk mengedit kuis yang dipilih sedangkan tombol Delete berfungsi menghapus kuis yang dipilih.



Gambar 3.41 Tampilan Halaman Edit Kuis

### 3.6.5 Pembuatan Halaman Edit

Pada pembahasan ini, ditujukan untuk membuat halaman Edit bila kuis tersebut dipilih untuk diedit.



Gambar 3.42 Tampilan Halaman Edit

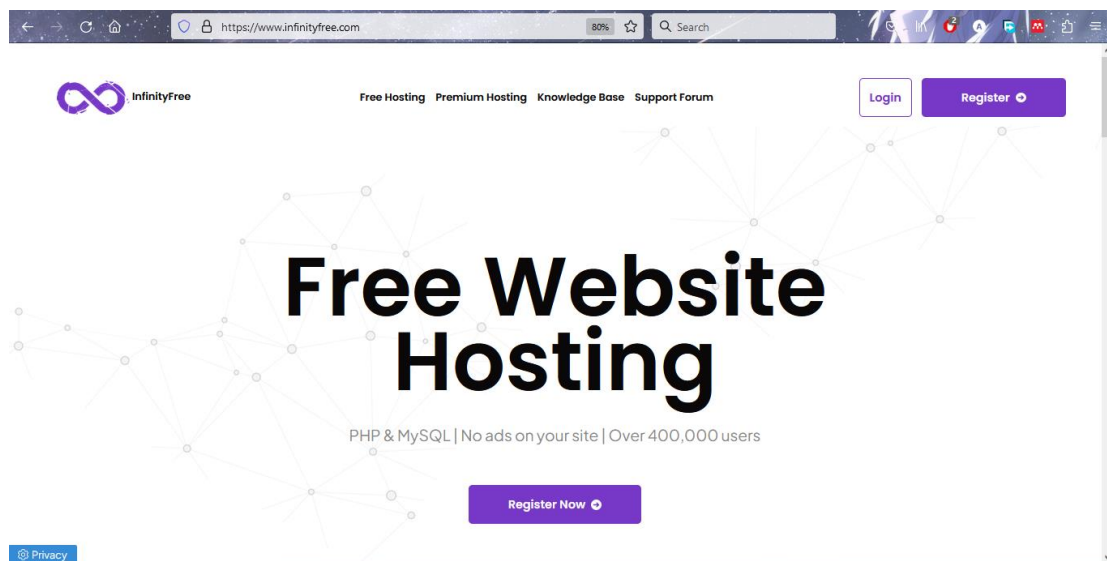
### 3.5 UJI COBA

Pada tahap ini merupakan pengujian terhadap aplikasi yang sudah selesai dibuat. *Website e-learning* tersebut akan dihosting secara online lalu dilanjutkan melalui pengujian dengan menggunakan *black box testing*.

#### 3.5.1 Hosting

Menurut Aliyun (2014:72), *hosting* memiliki arti layanan berbasis internet sebagai tempat penyimpanan data atau tempat menjalankan aplikasi ditempat terpusat yang disebut dengan server dan dapat diakses melalui jaringan internet.

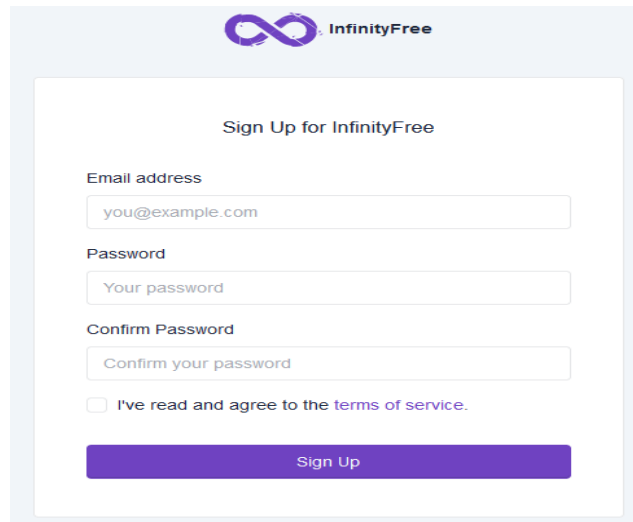
*Hosting* tersebut berfungsi membuat website yang selesai dibuat dapat dijalankan lalu dipakai banyak orang. *Hosting* tersedia secara berbayar dan gratis. *Hosting* yang akan digunakan pada *website e-learning* adalah infinityfree.com. Keunggulan pada hosting ini adalah gratis dan bebas iklan terhadap website yang sudah dihosting, dan juga *hosting* ini memiliki opsi berbayar untuk penggunaan yang lebih kompleks dan keamanan yang terjamin.



Gambar 3.43 InfinityFree Hosting

Langkah-langkah menggunakan infinityfree *hosting* :

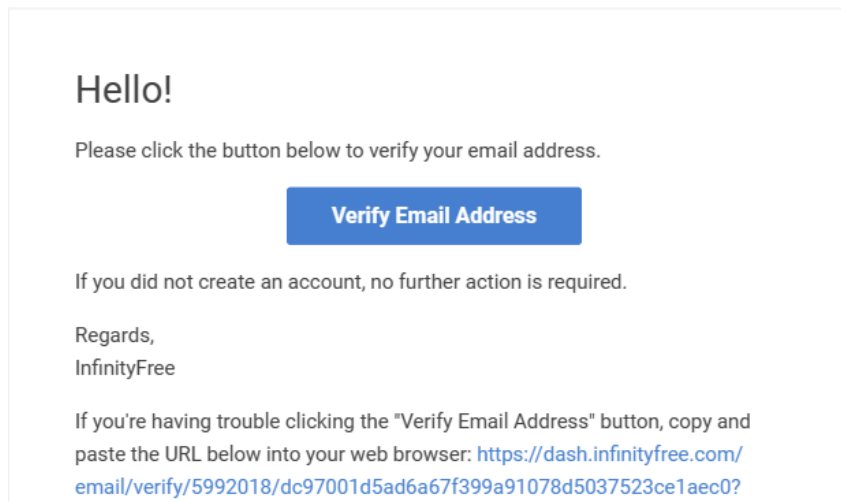
- Melakukan registrasi.



The image shows the 'Sign Up for InfinityFree' form. At the top is the InfinityFree logo. The form has three input fields: 'Email address' with the placeholder 'you@example.com', 'Password' with the placeholder 'Your password', and 'Confirm Password' with the placeholder 'Confirm your password'. Below these fields is a checkbox labeled 'I've read and agree to the [terms of service](#)'. At the bottom is a purple 'Sign Up' button.

*Gambar 3.44* Registrasi akun InfinityFree

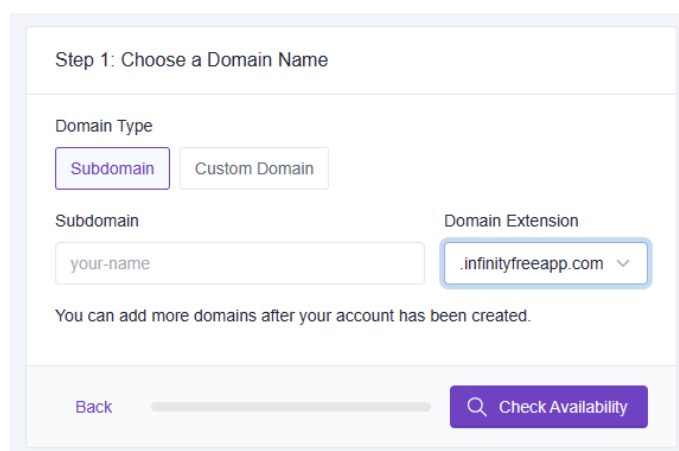
- Setelah selesai registrasi, melakukan verifikasi *email* yang sama pada saat registrasi *hosting*.



*Gambar 3.45* Verifikasi *Email* InfinityFree

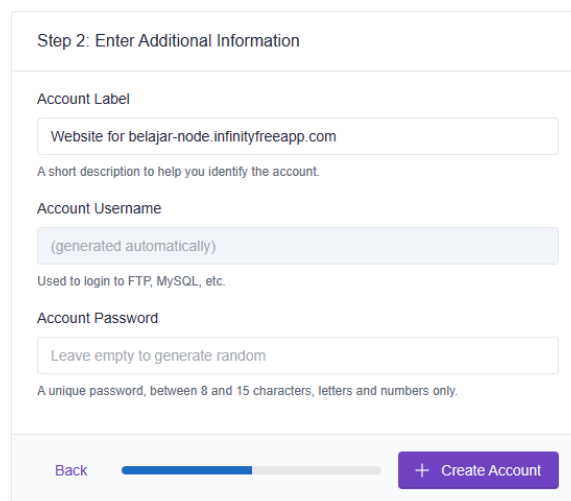
- Selesai verifikasi *email* otomatis akan login akun infinityfree. Lalu buat akun hosting. Dalam registrasi akun hosting saya sarankan memilih Domain Type yang Subdomain, lalu pada *Step 2* pengisian *password* disarankan kosong saja.

Alamat URL yang digunakan pada *website e-learning* ini adalah <http://belajar-code.infinityfreeapp.com>.



The screenshot shows the 'Step 1: Choose a Domain Name' registration form. It has a title bar 'Step 1: Choose a Domain Name'. Below it, there's a 'Domain Type' section with two buttons: 'Subdomain' (highlighted with a blue border) and 'Custom Domain'. Under 'Subdomain', there's a text input field containing 'your-name'. To the right, there's a 'Domain Extension' dropdown menu showing '.infinityfreeapp.com'. Below these fields, a note says 'You can add more domains after your account has been created.' At the bottom, there's a 'Back' link, a progress bar, and a 'Check Availability' button with a magnifying glass icon.

*Gambar 3.46 Registrasi Akun Hosting Step 1*

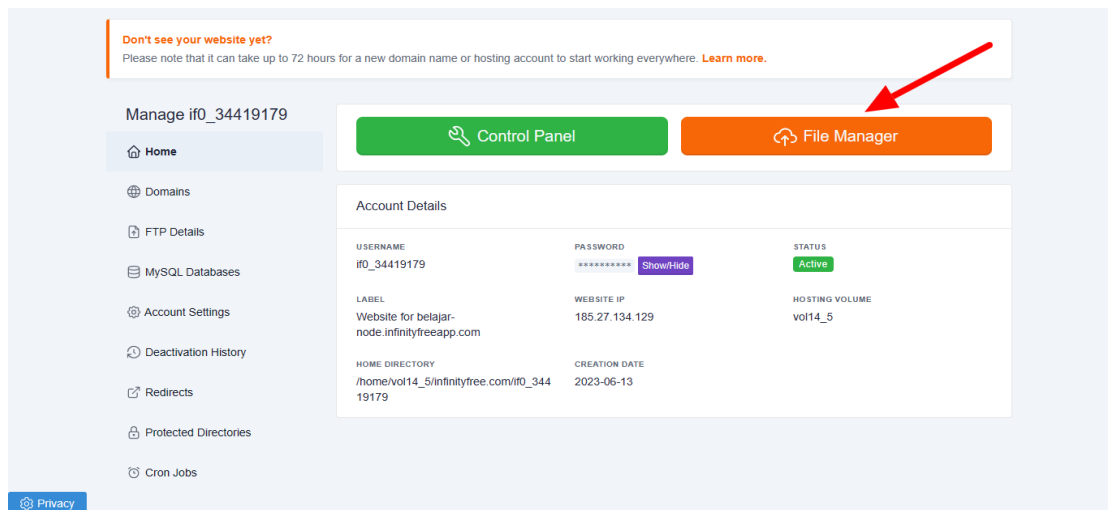


The screenshot shows the 'Step 2: Enter Additional Information' registration form. It has a title bar 'Step 2: Enter Additional Information'. Below it, there's an 'Account Label' section with a text input field containing 'Website for belajar-node.infinityfreeapp.com' and a note 'A short description to help you identify the account.' Underneath is the 'Account Username' section with a text input field containing '(generated automatically)' and a note 'Used to login to FTP, MySQL, etc.' Below that is the 'Account Password' section with a text input field containing 'Leave empty to generate random' and a note 'A unique password, between 8 and 15 characters, letters and numbers only.' At the bottom, there's a 'Back' link, a progress bar, and a 'Create Account' button with a plus icon.

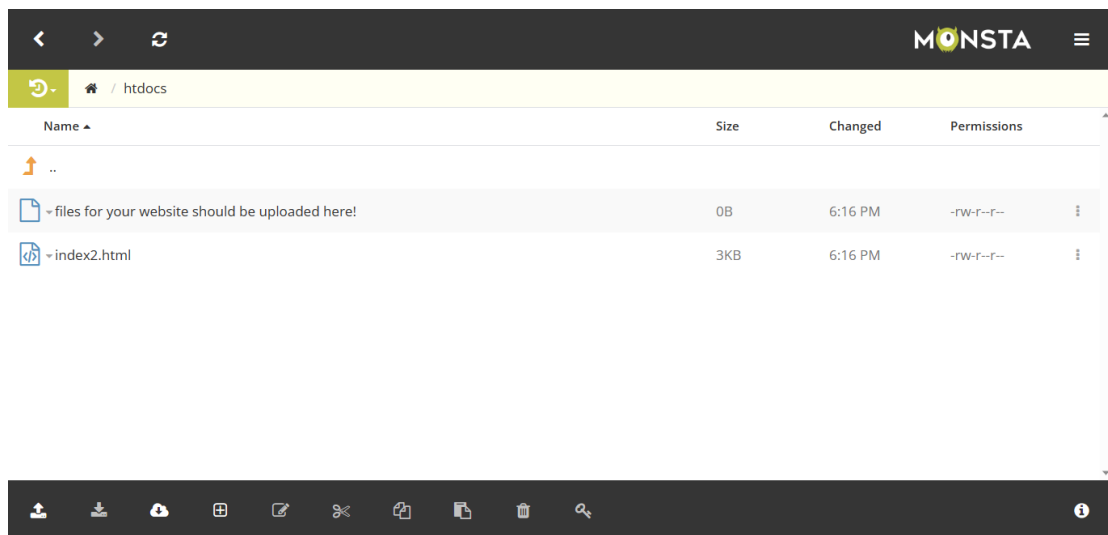
*Gambar 3.47 Registrasi Akun Hosting Step 2*

- Selesai buat akun *hosting*, langsung klik *Finish*. Kemudian masuk ke *File Manager*. Kemudian arahkan direktori ke folder *htdocs* untuk menaruh *website* yang sudah dibuat.

\*Catatan : Ada peringatan bertuliskan *website* dapat ditampilkan secara umum setelah selesai melakukan registrasi akun *hosting* selama 72 jam atau 3 hari.

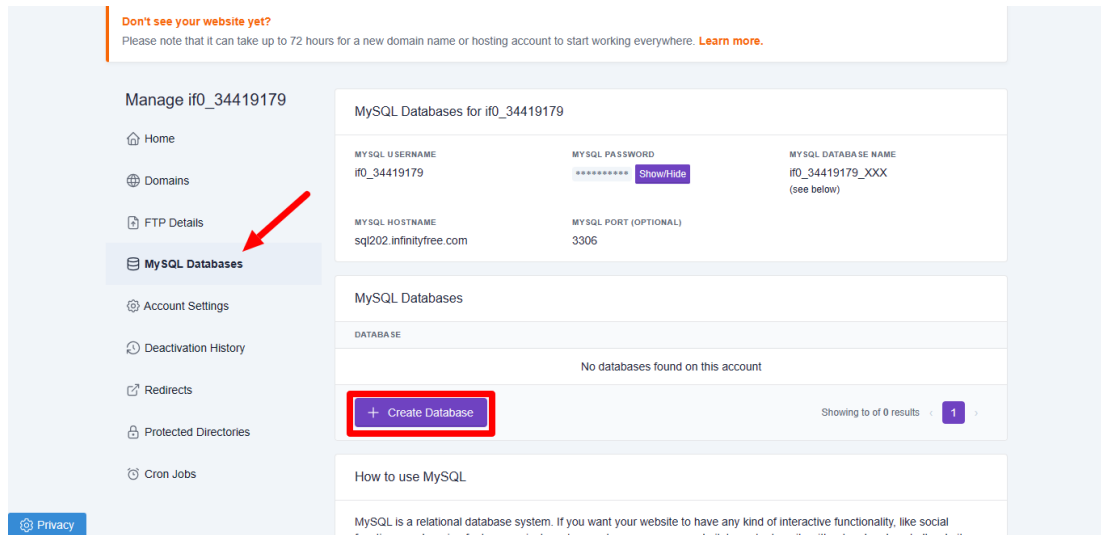


Gambar 3.48 Tampilan Home Akun *Hosting* yang sudah Dibuat



Gambar 3.49 File Manager dalam Direktori *htdocs*

- Jika memiliki *database*, pilih tab MY SQL Database, lalu klik *Create Database*. Import saja *database* pada PHPMyAdmin tersebut.



Gambar 3.50 Pengaturan Database pada Akun Hosting

### 3.5.2 Black Box

Pengujian ini bertujuan mengetahui fungsi-fungsi pada *website* berjalan sesuai yang diharapkan dan mengetahui kesalahan-kesalahan atau *error*. Oleh karena itu, program harus diuji coba terlebih dahulu untuk melihat kesalahan pada *website*, pengujian ini menggunakan metode *black box*. Pengujian terhadap metode ini *black box* berfokus pada persyaratan fungsional perangkat lunak.

Table 3.7 Table Pengujian Halaman Home

| Kelas Uji              | Test Case               | Hasil yang Diharapkan       | Hasil Pengujian                  | Kesimpulan |
|------------------------|-------------------------|-----------------------------|----------------------------------|------------|
| Input pencarian materi | Mengetikkan kata "HTML" | Menampilkan hasil pencarian | Menampilkan hasil pencarian yang | Berhasil   |



|                 |                         |                                                                                 |                                                                    |          |
|-----------------|-------------------------|---------------------------------------------------------------------------------|--------------------------------------------------------------------|----------|
|                 |                         | terkait kata yang diketik                                                       | mengandung kata HTML                                               |          |
|                 | Mengetikkan kata “hlmt” | Menampilkan peringatan <i>error</i>                                             | Menampilkan pesan <i>error</i> “Keyword tidak tersedia!”           | Berhasil |
| Tombol navigasi | Menekan tombol kuis     | Menampilkan kuis sesuai dengan bahasa program yang dipilih                      | Beralih ke halaman kuis sesuai bahasa program yang dipilih         | Berhasil |
|                 | Menekan tombol materi   | Menampilkan materi ke halaman pertama sesuai dengan bahasa program yang dipilih | Beralih ke halaman materi Intro sesuai bahasa program yang dipilih | Berhasil |
|                 | Menekan tombol login    | Menampilkan form login                                                          | Berpindah ke halaman login                                         | Berhasil |

Table 3.8 Table Pengujian Halaman Materi

| Kelas Uji | Test Case | Hasil yang Diharapkan | Hasil Pengujian | Kesimpulan |
|-----------|-----------|-----------------------|-----------------|------------|
|-----------|-----------|-----------------------|-----------------|------------|

|                   |                                                             |                                                            |                                                 |          |
|-------------------|-------------------------------------------------------------|------------------------------------------------------------|-------------------------------------------------|----------|
| Materi HTML       | Memilih judul materi HTML “Block & Inline”                  | Menampilkan materi HTML sesuai judul yang dipilih          | Beralih ke halaman materi HTML “Block & Inline” | Berhasil |
|                   | Memilih materi bahasa pemrograman lain, misalnya Javascript | Menampilkan materi “Intro” bahasa pemrograman yang dipilih | Beralih ke halaman materi Javascript “Intro”    | Berhasil |
|                   | Menekan tombol navigasi halaman Code Editor HTML            | Menampilkan halaman Code Editor HTML                       | Beralih ke halaman Code Editor HTML             | Berhasil |
| Materi Javascript | Memilih judul materi Javascript “Output”                    | Menampilkan materi Javascript sesuai judul yang dipilih    | Beralih ke halaman materi Javascript “Output”   | Berhasil |
|                   | Memilih materi bahasa pemrograman lain, misalnya PHP        | Menampilkan materi “Intro” bahasa pemrograman yang dipilih | Beralih ke halaman materi PHP “Intro”           | Berhasil |

|            |                                                        |                                                            |                                           |          |
|------------|--------------------------------------------------------|------------------------------------------------------------|-------------------------------------------|----------|
|            | Menekan tombol navigasi halaman Code Editor Javascript | Menampilkan halaman Code Editor Javascript                 | Beralih ke halaman Code Editor Javascript | Berhasil |
| Materi PHP | Memilih judul materi PHP “Loop”                        | Menampilkan materi PHP sesuai judul yang dipilih           | Beralih ke halaman materi PHP “Loop”      | Berhasil |
|            | Memilih materi bahasa pemrograman lain, misalnya HTML  | Menampilkan materi “Intro” bahasa pemrograman yang dipilih | Beralih ke halaman materi HTML “Intro”    | Berhasil |
|            | Menekan tombol navigasi halaman Code Editor PHP        | Menampilkan halaman Code Editor PHP                        | Beralih ke halaman Code Editor PHP        | Berhasil |

*Table 3.9 Table Pengujian Halaman Code Editor*

| Kelas Uji | Test Case | Hasil yang Diharapkan | Hasil Pengujian | Kesimpulan |
|-----------|-----------|-----------------------|-----------------|------------|
|-----------|-----------|-----------------------|-----------------|------------|

|                                         |                                                                                                                                  |                                                            |                                                                                  |          |
|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------|----------------------------------------------------------------------------------|----------|
| Halaman<br>Code<br>Editor<br>HTML       | Mengetikkan sintaks HTML<br>“<h1>Hello World!</h1>” lalu menekan tombol Run                                                      | Menampilkan <i>output</i> heading bertuliskan Hello World! | Kotak <i>preview</i> menampilkan kalimat “Hello World!” dengan cetak tebal besar | Berhasil |
|                                         | Memilih tema warna “dracula”                                                                                                     | Tema warna Code editor yang berbeda sesuai yang dipilih    | Tema warna menjadi hitam dan warna tulisan menjadi ungu dan putih                | Berhasil |
| Halaman<br>Code<br>Editor<br>Javascript | Mengetikkan sintaks Javascript<br>“document.write(“<p>Bahasa Javascript</p>”);” pada tag elemen <script> lalu menekan tombol Run | Menampilkan <i>output</i> kalimat “Bahasa Javascript”      | Kotak <i>preview</i> menampilkan kalimat paragraf Bahasa Javascript              | Berhasil |
|                                         | Memilih tema warna                                                                                                               | Merubah warna tema Code editor                             | Tema warna menjadi abu-abu                                                       | Berhasil |

|                         |                                                            |                                                                 |                                                           |          |
|-------------------------|------------------------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------|----------|
|                         | “base16-light”                                             | yang berbeda sesuai yang dipilih                                | dan warna tulisan menjadi coklat dan hitam                |          |
| Halaman Code Editor PHP | Mengetikkan sintaks PHP “<?php echo “Hello World!”?>”      | Menampilkan kalimat “Hello World!”                              | Kotak preview menampilkan kalimat Hello World!            | Berhasil |
|                         | Memilih tema warna “dracula” dan “base16-light” bergantian | Merubah warna tema Code editor yang berbeda sesuai yang dipilih | Warna latar dan tulisan berubah sesuai warna yang dipilih | Berhasil |

Table 3.10 Table Pengujian Login dan Registrasi

| Kelas Uji     | Test Case                                | Hasil yang Diharapkan          | Hasil Pengujian                                           | Kesimpulan |
|---------------|------------------------------------------|--------------------------------|-----------------------------------------------------------|------------|
| Halaman Login | Mengosongkan satu atau semua <i>form</i> | Menampilkan pesan <i>error</i> | Muncul pesan <i>error</i> “Username atau Password salah!” | Berhasil   |
|               | Mengisi <i>form</i> dengan salah         | Menampilkan pesan <i>error</i> | Muncul pesan <i>error</i> “Username atau Password salah!” | Berhasil   |

|                    |                                                                    |                                                                                                    |                                                                                                 |          |
|--------------------|--------------------------------------------------------------------|----------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|----------|
|                    | Mengisi semua <i>form</i> dengan benar bagi user                   | Pindah ke halaman <i>Home</i>                                                                      | <i>Form</i> berhasil disubmit dan beralih ke halaman <i>Home</i>                                | Berhasil |
| Halaman Registrasi | Mengosongkan satu atau semua <i>form</i>                           | Menampilkan pesan <i>error</i>                                                                     | Muncul pesan <i>error</i> “Lengkapi semua form!”                                                | Berhasil |
|                    | Mengisi <i>form</i> dengan benar                                   | Data <i>user</i> baru ditambahkan ke <i>database</i> lalu beralih otomatis ke halaman <i>Login</i> | Data <i>user</i> berhasil ditambahkan ke <i>database</i> lalu dialihkan ke halaman <i>Login</i> | Berhasil |
|                    | Mengisi <i>form</i> dengan email yang sudah ada di <i>database</i> | Menampilkan pesan <i>error</i>                                                                     | Muncul pesan <i>error</i> “Sudah dipakai. Silahkan gunakan <i>email</i> yang lain”              | Berhasil |

Table 3.11 Table Pengujian Admin

| Kelas Uji             | Test Case                                | Hasil yang Diharapkan                         | Hasil Pengujian                                                | Kesimpulan |
|-----------------------|------------------------------------------|-----------------------------------------------|----------------------------------------------------------------|------------|
| Halaman tambah materi | Mengosongkan satu atau semua <i>form</i> | Menampilkan pesan <i>error</i> lalu tidak ada | Muncul pesan <i>error</i> “Semua input harus diisi!” lalu data | Berhasil   |

|                     |                                                                                         |                                                                     |                                                                                                  |          |
|---------------------|-----------------------------------------------------------------------------------------|---------------------------------------------------------------------|--------------------------------------------------------------------------------------------------|----------|
|                     |                                                                                         | data baru di <i>database</i>                                        | pada <i>database</i> tidak bertambah                                                             |          |
|                     | Mengisi <i>form</i> dengan benar                                                        | Menampilkan pesan lalu data baru bertambah di <i>database</i>       | Muncul pesan “Berhasil menambahkan materi!” lalu data pada <i>database</i> bertambah             | Berhasil |
| Halaman tambah kuis | Mengosongkan satu atau semua <i>form</i>                                                | Menampilkan pesan error lalu tidak ada data baru di <i>database</i> | Muncul pesan error “Semua input harus diisi!” lalu data tidak bertambah di <i>database</i>       | Berhasil |
|                     | Mengisi <i>form</i> dengan benar                                                        | Menamipkan pesan dan data baru ditambahkan ke <i>database</i>       | Muncul pesan “Berhasil menambahkan kuis!” lalu data pada <i>database</i> bertambah               | Berhasil |
| Halaman edit kuis   | Menekan tombol <i>edit</i> pada salah satu list kuis, misalnya kuis nomor urut 1 dengan | Masuk ke halaman edit pada kuis yang dipilih                        | Saat menekan tombol edit pada kuis yang dipilih akan berpindah ke halaman edit kuis yang dipilih | Berhasil |

|  |                                                                                                                                                                                               |                                                                                             |                                                                                                                   |          |
|--|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|----------|
|  | pertanyaan<br>“Kepanjangan<br>dari PHP pada<br>awalnya<br>adalah...”                                                                                                                          |                                                                                             |                                                                                                                   |          |
|  | Menekan<br>tombol <i>delete</i><br>pada salah satu<br>list kuis,<br>misalnya kuis<br>dengan nomor<br>urut 5 dengan<br>pertanyaan<br>“Deklarasi<br>variabel pada<br>Javascript,<br>kecuali...” | Data kuis yang<br>dipilih akan<br>terhapus dari<br><i>database</i>                          | Soal yang dipilih<br>dari list edit<br>menghilang dan<br>data pada <i>database</i><br>juga terhapus               | Berhasil |
|  | Menekan<br>tombol filter<br>HTML,<br>Javascript atau<br>PHP                                                                                                                                   | Menyaring<br>kuis yang ingin<br>ditampilkan<br>berdasarkan<br>tombol filter<br>yang ditekan | Saat menekan filter<br>HTML hanya<br>menampilkan kuis<br>HTML, begitu juga<br>dengan filter<br>Javascript dan PHP | Berhasil |



### 3.6 PEMELIHARAAN

Pada tahap pemeliharaan atau *maintenance* merupakan tahapan akhir pada pengembangan model *waterfall*. Tahapan ini melakukan pengecekan terhadap *bug* atau kesalahan-kesalahan yang terdeteksi (*debugging*) pada saat aplikasi siap rilis ataupun *go public*. Setelah melakukan *debugging* akan segera dilakukan *maintenance*.

*Maintenance* tidak hanya melakukan *debugging* tetapi melakukan pengoptimalan terhadap UI/UX dan *updating* fitur yang mendukung aplikasi tersebut. Fitur yang disarankan perlu ditambahkan dalam *maintenance* yang akan datang yaitu forum diskusi. Forum diskusi sangat relevan dalam membantu pembelajaran bahasa pemrograman bagi user.

Fitur login dan registrasi akan bisa digunakan sebagaimana fungsi dan tujuan fitur tersebut dengan adanya forum diskusi.

## 4. PENUTUP

### 4.1 Kesimpulan

*Website* ini dibuat untuk pembelajaran pemrograman alternatif secara *online* yang mana dapat dipelajari oleh semua kalangan atau bagi yang tidak memiliki pengetahuan akan IT, dan juga *website e-learning* dapat diakses di berbagai *device* manapun.

Mengingat pembahasan BAB III sebelumnya, *website e-learning* dapat diakses dengan alamat <http://belajar-code.infinityfreeapp.com>. Alamat *website* dapat diakses selama setahun sesuai ketentuan jasa *hosting* dan dapat berjalan 24 jam penuh.

*Website* ini sudah diuji coba menggunakan *blackbox testing*. Hasilnya adalah semua fungsi dapat berjalan dengan baik.

### 4.2 Saran

*Webiste e-learning* ini sudah diuji coba dengan berbagai perangkat termasuk *gadget* atau *smartphone*. Demi kenyamanan dalam belajar dan juga praktek *coding*, penulis menyarankan mengakses *website e-learning* menggunakan PC atau laptop.

## DAFTAR PUSTAKA

- Adica. 2022. *Teori E-Elearning Menurut Beberapa Cendekiawan*. Diakses dari URL <https://www.silabus.web.id/e-learning/>. 27 Desember 2022.
- ADMINLP2M. 2022. *Metode Waterfall – Definisi dan Tahap-tahap Pelaksanaannya*. Diakses dari URL [https://lp2m.uma.ac.id/2022/06/07/metode-waterfall-definisi-dan-tahap-tahap-pelaksanaannya/#:~:text=Ian%20Sommerville%20\(2011\)%20menjelaskan%20bahwa,Testing%2C%20dan%20Operationa%20and%20Maintenance.](https://lp2m.uma.ac.id/2022/06/07/metode-waterfall-definisi-dan-tahap-tahap-pelaksanaannya/#:~:text=Ian%20Sommerville%20(2011)%20menjelaskan%20bahwa,Testing%2C%20dan%20Operationa%20and%20Maintenance.) 19 Agustus 2023.
- Amira K. 2021. *Pengertian Internet: Sejarah, Manfaat, dan Dampak Negatifnya*. Diakses dari URL <https://www.gramedia.com/literasi/internet/>. 4 Januari 2023.
- Ariata C. 2023. *Apa Itu JavaScript? Apa Perbedaannya dengan Java?*. Diakses dari URL <https://www.hostinger.co.id/tutorial/apa-itu-javascript>. 7 Januari 2023.
- Arief & M.Rudianto. 2011. *Pemrograman Web Dinamis Menggunakan Php dan Mysql*. Yogyakarta:ANDI.
- Dian Wahyuningsih. & Rakhmat Makmur. 2017. *E-Learning Teori dan Aplikasi*. Bandung:Informatika.
- Dicoding Intern. 2021 .“*Apa itu Activity Diagram? Beserta Pengertian, Tujuan, Komponen*. Diakses dari URL <https://www.dicoding.com/blog/apa-itu-activity-diagram/>. 9 Januari 2023.

Dicoding Intern. 2021. *Contoh Use Case Diagram Lengkap dengan Penjelasannya*. Diakses dari URL <https://www.dicoding.com/blog/contoh-use-case-diagram/>. 9 Januari 2023.

Faradilla A. 2023. *Apa Itu HTML? Fungsi dan Cara Kerja HTML*. Diakses dari URL <https://www.hostinger.co.id/tutorial/apa-itu-html>. 6 Januari 2023.

Faradilla A. 2023. *Apa Itu PHP? Pengertian PHP untuk Pemula*. Diakses dari URL <https://www.hostinger.co.id/tutorial/apa-itu-php/>. 6 Januari 2023.

Iftitah Nurul Laily. 2022. *Pengertian Website Menurut Para Ahli, Beserta Jenis dan Fungsinya*. Diakses dari URL <https://katadata.co.id/safrezi/berita/6200a2a9697ec/pengertian-website-menurut-para-ahli-beserta-jenis-dan-fungsinya>. 3 Januari 2023.

LamanIT. 2022. *13 Pengertian UML Menurut Para Ahli dan Jenisnya*. Diakses dari URL <https://lamanit.com/pengertian-uml-menurut-para-ahli/>. 9 Januari 2023.

Mirza M.Haikal. 2021. *Bootstrap: Pengertian, Kegunaan, Kelebihan, dan Kekurangannya*. Diakses dari URL <https://www.niagahoster.co.id/blog/bootstrap-adalah/>. 7 Januari 2023.

Ridwan Fajar. 2016. *Mengenal Sosok Hakon Wium Lie: Pencipta CSS*. Diakses dari URL <https://codepolitan.com/blog/mengenal-sosok-hakon-wium-lie-pencipta-css>. 7 Januari 2023.

Rozi, Zaenal A. and SmitDev Community. 2015. *Bootstrap Design Framework*. Jakarta:PT. Elex Media Komputindo.

Setiawan, Didik. 2017. *Buku Sakti Pemrograman WEB: HTML, CSS, PHP, MySQL & JavaScript*. Yogyakarta:Start UP.

Sidik, B. 2017. *Pemrograman Web dengan PHP 7*. Bandung:Informatika.

Sommerville, Ian. 2011. *Software Engineering (Rekayasa Perangkat Lunak)*. Jakarta:Erlangga.

Syafnidawaty. 2020 . *DATABASE*. Diakses dari URL  
<https://raharja.ac.id/2020/04/25/database/>. 7 Januari 2023.

## LEMBAR PERNYATAAN UJI COBA APLIKASI

Saya yang bertanda tangan di bawah ini:

Nama : Kristovel Adi Sucipto

NPM : 16117987

Judul PI : PEMBUATAN WEBSITE E-LEARNING PROGRAMMING  
MENGUNAKAN PHP NATIVE DAN FRAMEWORK  
BOOTSTRAP

Menyatakan bahwa aplikasi dalam penulisan ilmiah ini telah selesai dan diujicobakan. Semua fungsi telah berjalan dengan baik.

Demikian pernyataan ujicoba ini dibuat dengan sebenar-benarnya dan dengan penuh kesadaran.

Jakarta, 16– Oktober – 2023

Mahasiswa

Pembimbing



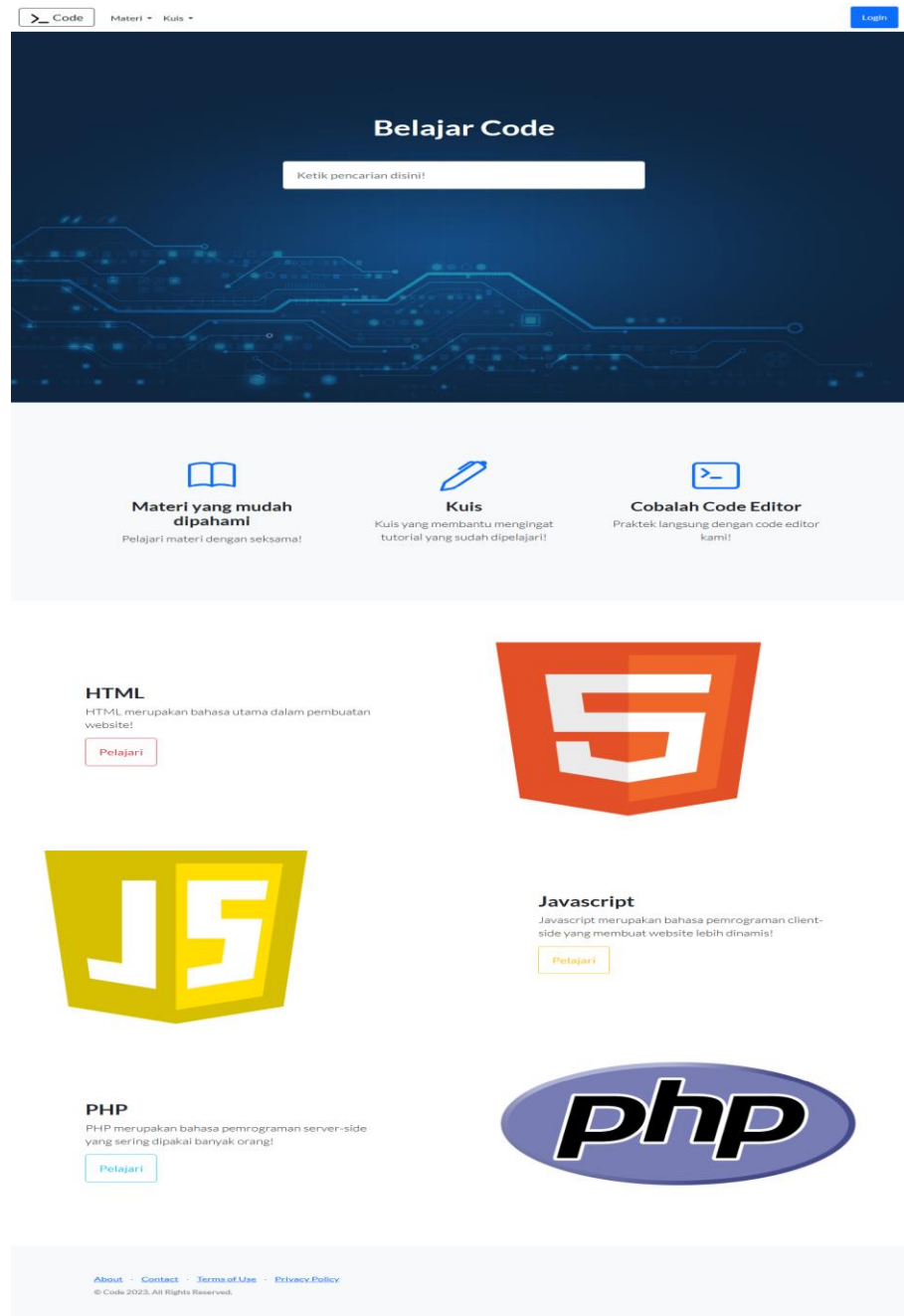
(Kristovel A.S.)



(Imam Ahmad Trinugroho, ST., MMSI)

# OUTPUT PROGRAM

## 1. Halaman Home



## 2. Halaman Tutorial HTML

Code

HTML Tutorial

HTML Intro

HTML Block & Inline

HTML Atribut

HTML ID

HTML Class

HTML Style & CSS

HTML List

HTML Table

HTML Form

HTML Javascript

Coba HTML

Mode Materi

# HTML Intro

HTML merupakan bahasa markup yang terdiri atas banyak banyak elemen. HTML juga merupakan bahasa standar dalam pembuatan sebuah halaman web.

## Sejarah

HTML (Hypertext Markup Language) adalah bahasa pemrograman web yang pertama kali dikembangkan pada tahun 1990 oleh seorang ilmuwan bernama Tim Berners-Lee di CERN, Swiss. Pada saat itu, HTML hanya memiliki sedikit tag dan fungsionalitas yang sangat terbatas, hanya digunakan untuk membuat dokumen sederhana yang terdiri dari teks dan gambar. Seiring dengan berkembangnya teknologi web, HTML juga mengalami perkembangan yang signifikan. Berikut ini adalah beberapa tahap perkembangan HTML hingga saat ini:

1. HTML 2.0 HTML 2.0 adalah versi pertama dari HTML yang secara resmi diterbitkan pada tahun 1995. Versi ini menambahkan banyak fitur baru seperti tabel, formulir, dan frame.
2. HTML 3.2 HTML 3.2 diterbitkan pada tahun 1997 dan menambahkan beberapa elemen baru seperti bidang input, tombol, dan daftar.
3. HTML 4.01 HTML 4.01 diterbitkan pada tahun 1999 dan merupakan versi HTML yang paling banyak digunakan pada saat itu. Versi ini menambahkan beberapa tag baru seperti label, fieldset, dan legend. Selain itu, HTML 4.01 juga mendukung Cascading Style Sheets (CSS) untuk memformat dokumen web.
4. XHTML (HTML 5) XHTML (Extensible Hypertext Markup Language) diperkenalkan pada tahun 2000 sebagai pengganti HTML 4.01. Versi ini menggunakan aturan sintaks yang lebih ketat dan mendukung XML. HTML 5 secara resmi dirilis pada tahun 2014 dan menambahkan beberapa fitur baru seperti audio, video, dan geolocation.
5. HTML 5.1 dan 5.2 HTML 5.1 dirilis pada tahun 2016 dan menambahkan beberapa fitur baru seperti semantik, aksesibilitas, dan performa yang lebih baik. HTML 5.2 dirilis pada tahun 2017 dan menambahkan beberapa tag baru seperti dialog, menyempurnakan semantik, serta menyempurnakan fitur-fitur yang sudah ada di HTML 5.

## Struktur

Struktur HTML setiap dokumen HTML harus dimulai dengan tag `<html>` dan diakhiri dengan tag `</html>`. Semua elemen HTML harus diletakkan di antara tag ini. Di dalam tag `<html>`, terdapat dua bagian utama yaitu `<head>` dan `<body>`. Bagian `<head>` digunakan untuk menambahkan meta data dan informasi penting lainnya, sedangkan bagian `<body>` digunakan untuk menambahkan konten halaman web.

```
<!DOCTYPE html>
<html>
<head>
  <title>Judul Halaman Web</title>
</head>
<body>
  <h1>Selamat Datang.</h1>
  <p>Ini adalah paragraf.</p>
</body>
</html>
```

### Hasil

Selamat Datang.

Ini adalah paragraf.

Penjelasan elemen HTML di atas:

```
<!DOCTYPE html> Mendeklarasikan atau mendefinisikan dokumen tersebut dengan HTML5.
<html> Elemen yang menampung semua elemen HTML.
<head> Menyimpan meta informasi pada halaman web seperti author, description, pengaturan bahasa yang digunakan serta penggunaan karakter encoding dan lain-lain.
<title> Judul pada halaman web.
<body> Berisikan tubuh atau konten web.
<h1> Pembuatan heading.
<p> Pembuatan kalimat paragraf.
```

Elemen HTML terdiri dari tag pembuka dan tag penutup. Tag pembuka dimulai dengan tanda kurung siku `<` dan diakhiri dengan tanda kurung siku `>`. Tag penutup dimulai dengan tanda kurung siku `<` dan diakhiri dengan tanda kurung siku `/` dan `>`. Beberapa elemen HTML tidak membutuhkan tag penutup seperti `<img>`, `<br>` dan sebagainya.

## Semantik

HTML juga memiliki tag elemen semantik yang membantu dalam mengorganisir konten halaman web dengan lebih baik. Tag elemen semantik meliputi `<header>`, `<nav>`, `<section>`, `<article>`, `<aside>`, `<footer>`, dan lainnya. Setiap tag elemen semantik memiliki arti dan tujuan tertentu.

```
<header>
  <nav>
    <section>
      <article>
    <aside>
  <footer>
```

### Hasil

Logo dan judul halaman web

- Beranda
- Tentang
- Kontak

Judul Artikel

Isi artikel...

Iklan

Isi iklan...

Hak Cipta © 2023 Halaman Web Saya

Referensi

[W3schools \(Intro\)](#)

[W3schools \(Semantic\)](#)

Copyright © Code 2023

[Privacy Policy](#) [Terms & Conditions](#)



### 3. Halaman Tutorial Javascript

Code

JavaScript Tutorial

JavaScript

JavaScript Intro

JavaScript Output

JavaScript Variabel & Tipe Data

JavaScript Aritmatika & Operator

JavaScript String

JavaScript Array

JavaScript Iterasi

JavaScript Percabangan

JavaScript Events

JavaScript Function

Coba JS

Mode Materi

## Javascript Output

JavaScript memiliki beberapa cara untuk menghasilkan output atau tampilan ke layar. Output di JavaScript dapat digunakan untuk menampilkan informasi, memberikan feedback kepada pengguna, atau memecah bagian-bagian kode agar lebih mudah dipahami. Berikut ini beberapa cara untuk menghasilkan output di JavaScript.

### Console Log

Metode `console.log()` adalah cara paling umum untuk menghasilkan output di JavaScript. Metode ini digunakan untuk mencetak informasi ke konsol browser atau terminal. Kita dapat menggunakan `console.log()` untuk mencetak nilai variabel atau pesan teks.

```
console.log(5+7);
```

Hasil

12

Untuk dapat membuka konsol atau terminal pada browser cukup tekan tombol F12 pada keyboard anda.

Console Browser

### Document Write

Metode `document.write()` digunakan untuk menulis HTML atau teks ke dalam dokumen web. Metode ini dapat digunakan untuk menampilkan pesan teks di halaman web.

```
<html>Document Write</html>
<p>Ini paragraf.</p>

<script>
document.write("Hello World!");
</script>
```

Hasil

Document Write

Ini paragraf.

Hello World!

`document.write` dijalankan setelah seluruh kode HTML dimuat.

### Inner HTML

Metode `innerHTML` digunakan untuk menetapkan isi dari elemen HTML. Metode ini dapat digunakan untuk mengganti isi elemen HTML yang sudah ada atau menambahkan elemen HTML baru ke halaman web.

```
<html>Inner HTML</html>
<p id="test"></p>

<script>
document.getElementById("test").innerHTML = "Paragraf ini menggunakan inner HTI";
</script>
```

Hasil

Inner HTML

Paragraf ini menggunakan inner HTML

### Pop Up (Alert)

Metode `alert()` digunakan untuk menampilkan pesan pop-up di jendela browser. Metode ini sering digunakan untuk memberikan feedback kepada pengguna atau memberikan peringatan.

```
<html>Alert Output</html>
<p>Tekan tombol di bawah!</p>
<button onclick="popup()">Klik disini!</button>

<script>
function popup(){
alert("Username atau Password anda salah!");
}
</script>
```

Hasil

Alert Output

Tekan tombol di bawah!

Klik disini!

Referensi

W3schools

Copyright © Code 2023

Privacy Policy · Terms & Conditions

## 4. Halaman Tutorial PHP

> Code

☰

Coba PHP

Mode Materi

👤

PHP Tutorial

PHP

PHP Intro

PHP Sintaks

PHP Variabel

PHP Echo/Print

PHP Tipe Data

PHP Operators

PHP Percabangan

PHP Loop

PHP Function

PHP Arrays

# PHP Variabel

Variabel merupakan salah satu konsep dasar dalam pemrograman. Variabel digunakan untuk menyimpan nilai yang dapat diakses dan diubah selama proses eksekusi program. Dalam PHP, Anda dapat membuat variabel untuk menyimpan berbagai jenis data, termasuk angka, teks, array, dan objek.

## Penulisan Variabel

- Variabel dalam PHP diawali dengan tanda dolar (\$), diikuti dengan nama variabel.
- Variabel harus diawali dengan huruf atau underscore (\_).
- Variabel tidak bisa diawali dengan angka.
- Nama variabel bersifat case-sensitive, yang berarti \$nama dan \$Nama dianggap sebagai dua variabel yang berbeda.
- Contoh penulisan variabel yang valid: \$nama, \$umur, \$jumlah\_barang.

## Variabel Scope

Variabel scope merujuk pada wilayah atau bagian program di mana sebuah variabel dapat diakses dan digunakan. Dalam PHP, terdapat beberapa jenis scope yang mempengaruhi ketersediaan variabel, yaitu scope global, scope lokal, dan scope statis.

```
// Scope global
$name = "John Doe"; // Variabel global

function tampilkanNama() {
    global $name; // Memanggil variabel global di dalam fungsi
    echo $name;
}

tampilkanNama();
```

Hasil

John Doe

**Scope Lokal**

Variabel lokal adalah variabel yang hanya dapat diakses di dalam blok kode tertentu di mana variabel tersebut dideklarasikan. Variabel lokal dideklarasikan di dalam fungsi atau metode, dan tidak dapat diakses dari luar blok tersebut.

```
// Scope lokal
// Variabel lokal
echo $umur;
}

tampilkanUmur();
// Jika $umur dipanggil di luar scope function tidak akan menghasilkan output
?>
```

Hasil

25

**Scope Statis**

Variabel statis adalah variabel yang mempertahankan nilainya di antara pemanggilan fungsi. Variabel statis dideklarasikan di dalam fungsi dan nilainya akan diingat setelah fungsi selesai dieksekusi. Ketika fungsi dipanggil kembali, variabel statis akan menyimpan nilainya dari pemanggilan sebelumnya.

```
// Scope static
// Variabel static
$jumlah++;
echo $jumlah;
}

hitungJumlah(); // Output : 1
echo "
"; // Membuat baris ke bawah
hitungJumlah(); // Output : 2
echo "
";
hitungJumlah(); // Output : 3
?>
```

Hasil

1
2
3

Referensi

[W3schools](#)

[Scope Variabel](#)

[PHP](#)

Copyright © Code 2023

[Privacy Policy](#) · [Terms & Conditions](#)

L-5

## 5. Halaman Kuis HTML

[Code](#)

HTML Kuis

PHP Kuis

Javascript Kuis

### HTML Kuis

1.Tag HTML yang digunakan untuk membuat heading adalah <...>

☐ html  
☐ head  
☐ title  
☐ h1

2.Tag yang menyimpan informasi halaman web adalah <...>

☐ script  
☐ body  
☐ html  
☐ head

3.Tag <title> berfungsi untuk

☐ Tag untuk memberi judul pada dokumen HTML  
☐ Tag untuk membuat link  
☐ Tag untuk membuat heading  
☐ Tag untuk membuat paragraf

4.Tag yang berfungsi untuk pembuatan list huruf atau angka adalah <...>

☐ li  
☐ ul  
☐ ol  
☐ select

5.Tag yang berfungsi untuk pembuatan list berupa simbol adalah <...>

☐ ol  
☐ ul  
☐ li  
☐ select

6.<a href="">. Yang merupakan atribut adalah...

☐ <a href="">  
☐ a  
☐ href  
☐ ""

7.Salah satu jenis type input pada HTML 5, kecuali...

☐ checkbox  
☐ text  
☐ password  
☐ submit

8.Pembuatan design halaman web dilakukan pada tag <...>

☐ script  
☐ img  
☐ style  
☐ iframe

9.Selector class pada CSS dan tag style berupa...

☐ #  
☐ \$  
☐ .  
☐ !

10.Pembuatan bahasa pemrograman Javascript dibuat pada tag <...>

☐ link  
☐ iframe  
☐ script  
☐ head

Submit

Copyright © Code 2023 [Privacy Policy](#) • [Terms & Conditions](#)

## 6. Halaman Kuis Javascript

[Code](#)

HTML Kuis

PHP Kuis

Javascript Kuis

### Javascript Kuis

1. Penulisan code Javascript pada HTML ditempatkan pada tag <...>

☐ script

☐ noscript

☐ code

☐ button

2. Deklarasi variabel pada Javascript, kecuali...

☐ var

☐ let

☐ const

☐ string

3. Output pada Javascript adalah...

☐ console.log();

☐ print;

☐ print\_r

☐ echo;

4. `var x = 5;`  
`var y = 10;`  
`console.log(x + y);`

Output yang ditampilkan adalah...

☐ 510

☐ Error

☐ 15

☐ 5

5. Fungsi dari `typeof()` adalah...

☐ Menampilkan Output

☐ Membuat alert

☐ Pengecekan tipe data pada variabel

☐ Sebuah variabel

6. `var x = 10;`  
`if (x > 5) {`  
`console.log("x is greater than 5");`  
`}`  
`else{`  
`console.log("x is less than or equal to 5");`  
`}`

Output yang ditampilkan adalah...

☐ Error

☐ x is greater than 5

☐ x is less than or equal to 5

☐ undefined

7. `function Hello(){`  
`console.log('Hello World');`  
`}`

Pemanggilan function yang benar adalah...

☐ Hello;

☐ hello();

☐ Hello();

☐ function Hello()

8. `arr = ["Budi"];`  
  
`console.log(arr.length);`

Output yang ditampilkan adalah...

☐ 1

☐ undefined

☐ 0

☐ null

9. `for (let i = 0; i < 5; i++) {`  
`console.log(1);`  
`}`

Output yang ditampilkan adalah...

☐ 1, 2, 3, 4, 5

☐ 0, 1, 2, 3, 4

☐ 1, 3, 5, 7, 9

☐ 1, 2, 3, 4, 5

10. Sintaks yang digunakan untuk mendapatkan elemen HTML di JavaScript adalah...

☐ `document.getElementsByName()`

☐ `document.querySelector()`

☐ `document.getElementById()`

☐ `document.getElementsByTagName()`

Submit

Copyright © Code 2023

[Privacy Policy](#) · [Terms & Conditions](#)

## 7. Halaman Kuis PHP

[Code](#)

HTML Kuis

PHP Kuis

Javascript Kuis

### PHP Kuis

1. Kepanjangan dari PHP pada awalnya adalah...

☐ Hypertext Preprocessor  
☐ Person Home Page  
☐ Personal Home Page  
☐ Hyperlink Preprocessor

2. Sintaks `echo` digunakan untuk...

☐ Membaca File  
☐ Menampilkan Output  
☐ Menampilkan tanggal  
☐ Mengambil Output

3. `= 'Budi'`; Penulisan variabel nama pada PHP yang benar adalah...

☐ nama  
☐ \$nama  
☐ \_nama  
☐ ?nama

4. `$a = 1;  
$b = 2;  
$c = $a*$b;  
echo $c;`

Output yang dihasilkan...

☐ 12  
☐ 1  
☐ 2  
☐ Error

5. `$nilai = 75;  
if ($nilai == 100){  
echo 'A';  
}  
elseif ($nilai == 75){  
echo 'B';  
}elseif ($nilai == 50){  
echo 'C';  
}elseif ($nilai == 25){  
echo 'D';  
}`

Output yang ditampilkan adalah...

☐ A  
☐ B  
☐ C  
☐ D

6. `switch ($pilihan){  
case "A":  
echo 80;  
break;  
case "B":  
echo 70;  
break;  
default:  
echo 'Pilihan tidak valid';  
break;  
}`

Jika \$pilihan bernilai 85, outputnya adalah...

☐ 80  
☐ undefined  
☐ 70  
☐ Pilihan tidak valid

7. Berikut adalah contoh syntax untuk melakukan looping pada PHP, kecuali:

☐ while  
☐ switch  
☐ for  
☐ foreach

8. `$i=0;  
while($i<5){  
echo $i;  
$i++;  
}`

Output yang ditampilkan adalah...

☐ 1234  
☐ 12345  
☐ 01234  
☐ 012345

9. `$colors = array("red", "green", "blue");  
echo count($color);`

Output yang dihasilkan...

☐ 1  
☐ 2  
☐ 3  
☐ 4

10. `$numbers = array(1, 3, 5, 7, 9);  
echo $numbers[2];`

Output yang dihasilkan...

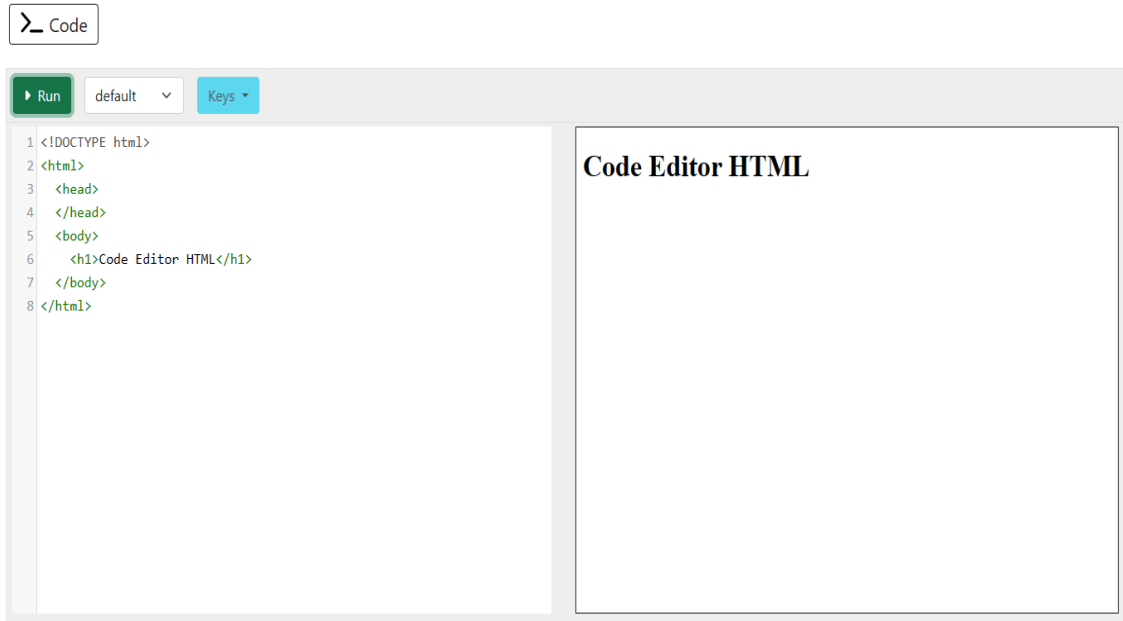
☐ 1  
☐ 3  
☐ 5  
☐ 7

Submit

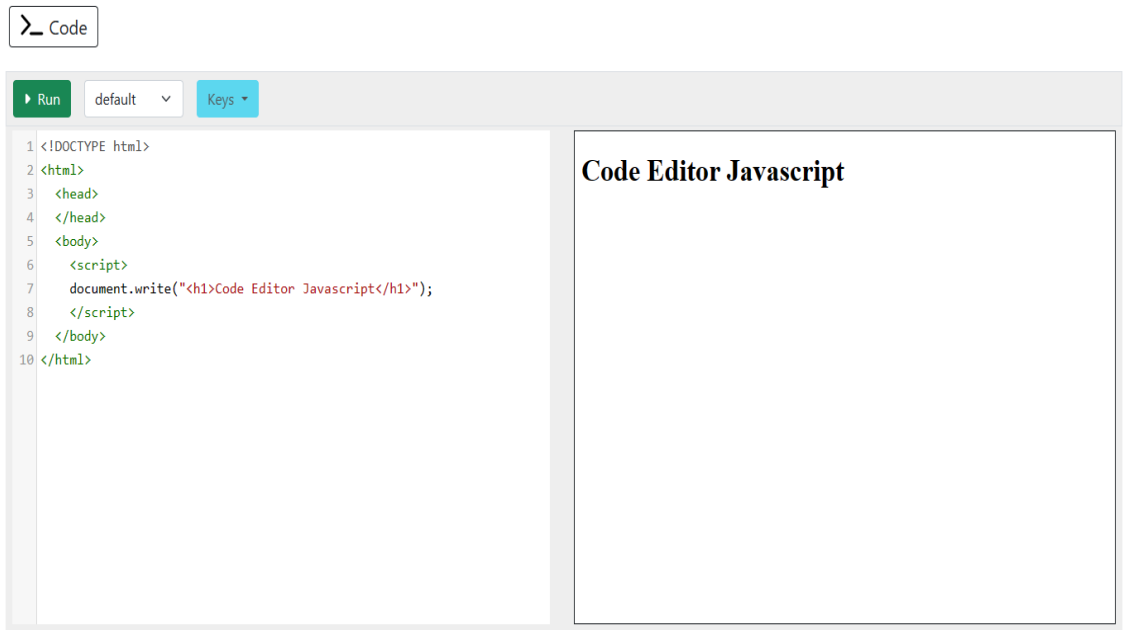
Copyright © Code 2023

[Privacy Policy](#) · [Terms & Conditions](#)

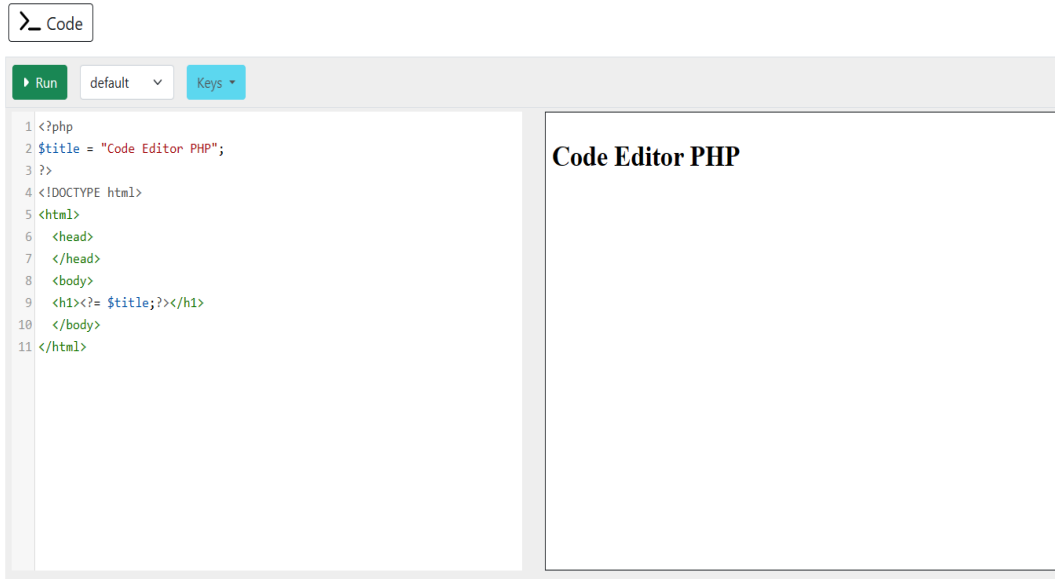
## 8. Halaman Code Editor HTML




## 9. Halaman Code Editor Javascript



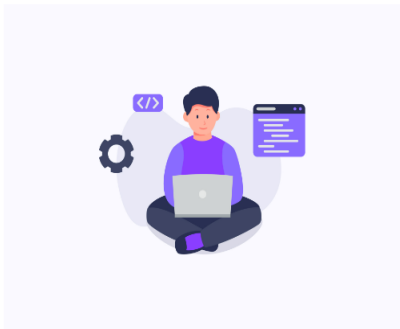
## 10. Halaman Code Editor Editor PHP



## 11. Halaman Register

 Code

# DAFTAR AKUN!




[Coding Vectors by Vecteezy](#)

[Login](#) jika sudah punya akun!

## 12. Halaman Login

> Code



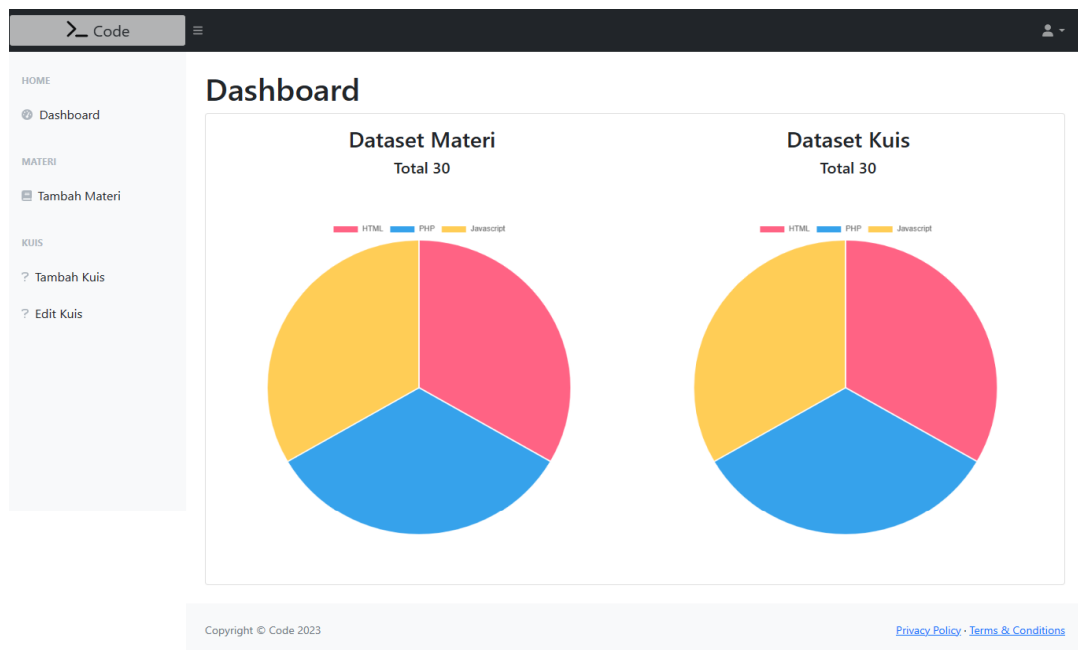
Programming Vectors by Vecteezy

### Login

Login

Ingin daftar [Register](#)

## 13. Halaman Admin Dashboard





## 15. Halaman Admin Tambah Materi

Code

HOME

DASHBOARD

MATERI

TAMBAH MATERI

KUIS

TAMBAH KUIS

EDIT KUIS

Tambah Materi

Mode BahasaBahasa...

Judul Materi

Referensi+

Isi Materi

XUndoRedoBoldItalicText ColorBackground ColorLinkListOrdered ListDecrease IncreaseTableMediaCodeFullscreenHorizontal LineVerticalLineMoreSource

BBI $\pi$ :|<=>>StylesFormat?

Preview Materi

Tambah Materi

Copyright © Code 2023

[Privacy Policy](#) · [Terms & Conditions](#)

## 16. Halaman Admin Tambah Kuis

> Code

[HOME](#)

[Dashboard](#)

---

MATERI

- Tambah Materi

---

KUIS

- Tambah Kuis
- Edit Kuis

## Tambah Kuis

**Soal Mode**

 Bahasa...

**Soal Kuis**

$\times$  | | | | | | | | | | | | | | | | Styles | Format | ?

**Jawaban**

 Jawaban...

**Pilihan Jawaban**

<b>A</b>	<input type="text"/>
<b>B</b>	<input type="text"/>
<b>C</b>	<input type="text"/>
<b>D</b>	<input type="text"/>

Buat Soal

Copyright © Code 2023
[Privacy Policy](#) • [Terms & Conditions](#)

## 17. Halaman Admin Edit Kuis

> Code

HOME

Dashboard

MATERI

Tambah Materi

KUIS

Tambah Kuis

Edit Kuis

### List Edit Kuis

Filter Kuis

HTML PHP Javascript Reset

Kuis aktif : HTML : 10 PHP : 10 Javascript : 10

#	Mode	Soal	Jawaban	Aksi
1	PHP	Kepanjangan dari PHP pada awalnya adalah...	Personal Home Page	ON Edit Delete
2	PHP	Sintaks echo digunakan untuk...	Menampilkan Output	ON Edit Delete
3	HTML	Tag HTML yang digunakan untuk membuat heading adalah <...>	h1	ON Edit Delete