

**JURNAL MODUL 4**  
**INTEGRASI HTML, CSS, DAN JAVASCRIPT**

HAFIZ RAFIE ADITYA

607022400023

D3TK 48-03

**A. TUJUAN PRAKTIKUM**

1. Mahasiswa memahami cara membangun struktur dasar halaman web menggunakan HTML.
2. Mahasiswa mampu menerapkan CSS untuk membuat tampilan web yang menarik dan responsif.
3. Mahasiswa dapat menggunakan JavaScript untuk menambahkan interaktivitas dan validasi form pada halaman web.
4. Mahasiswa mampu mengintegrasikan HTML, CSS, dan JavaScript menjadi satu proyek web yang utuh dan berfungsi dengan baik.

**B. KEGIATAN PRAKTIKUM**

1. Membuat struktur dasar halaman web dengan elemen *header*, *content*, dan *footer* menggunakan HTML.
2. Menambahkan *file styles.css* untuk mendesain tata letak dan gaya visual halaman agar responsif di berbagai perangkat.
3. Menulis skrip JavaScript untuk validasi form dan menambahkan interaksi pada tombol serta animasi pada menu navigasi.
4. Membuat halaman web lengkap dengan menu navigasi, artikel, form kontak, dan footer sesuai ketentuan.
5. Menguji fitur JavaScript seperti validasi form, animasi hover pada menu, dan pesan selamat datang ketika halaman pertama kali diakses.
6. Mendokumentasikan hasil proyek serta menjelaskan bagaimana HTML, CSS, dan JavaScript berkolaborasi membentuk sistem web yang dinamis.

## C. Praktikum

Modul 4

Beranda Artikel Kontak

Selamat datang, Hafiz Rafie, halaman ini sudah interaktif.

**Artikel Utama**

Ini bagian Artikel  
Bisa di resize dong

**Tips Singkat**

- Biar keren aja
- Keren kan aowkaokwoakwo
- Validasi di client dan server.

**Form Kontak**

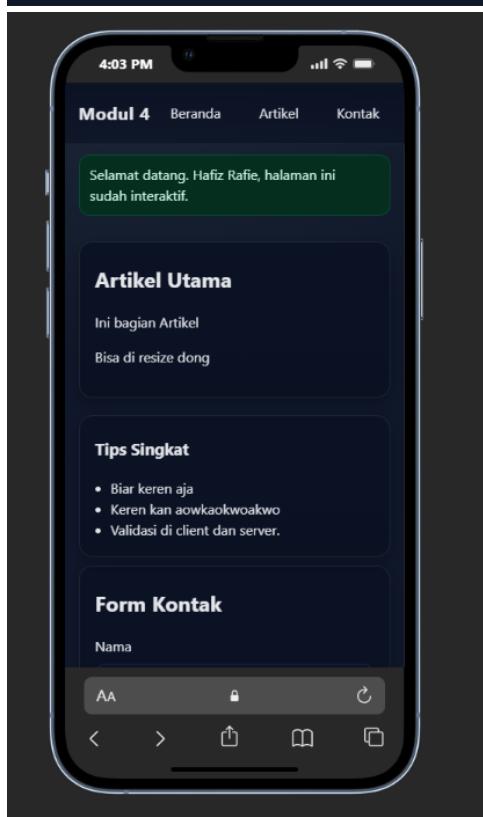
Nama  
Nama lengkap

Email  
email@domain.com

Pesan  
Tulis pesan...

Kirim

© 2025



## HTML

```
<!DOCTYPE html>
<html lang="id">
<head>
<meta charset="utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<title>Proyek Web Sederhana</title>
<link rel="stylesheet" href="style.css" />
</head>
<body>
<header class="site-header">
<div class="container header-inner">
<h1 class="logo">Modul 4</h1>
<nav aria-label="Navigasi utama">
<ul class="nav" id="mainNav">
<li><a href="#beranda">Beranda</a></li>
<li><a href="#artikel">Artikel</a></li>
<li><a href="#kontak">Kontak</a></li>
</ul>
</nav>
</div>
</header>

<div
id="welcomeBanner"
class="welcome"
role="status"
aria-live="polite"
hidden
>
Selamat datang. Hafiz Rafie, halaman ini sudah interaktif.
<button type="button" id="closeWelcome" aria-label="Tutup"></button>
</div>

<main class="container grid" id="beranda">
<article id="artikel" class="card">
<h2>Artikel Utama</h2>
<p>
    Ini bagian Artikel
</p>
<p>
    Bisa di resize dong
</p>
</article>

<aside class="card tips">
<h3>Tips Singkat</h3>
<ul>
<li>Biar keren aja</li>
<li>Keren kan awkaokwoakwo</li>
<li>Validasi di client dan server.</li>
</ul>
</aside>

```

The screenshot shows a code editor interface with two tabs: `index.html` and `script.js`. The `index.html` tab is active, displaying the following HTML code:

```
M4 > index.html X
M4 > index.html > html > body > main#beranda.container.grid > aside.card.tips > ul > li
2   <html lang="id">
3     <body>
4       <main class="container grid" id="beranda">
5         <aside class="card tips">
6           </aside>
7         </main>
8
9         <section id="kontak" class="container card">
10          <h2>Form Kontak</h2>
11          <form id="contactForm" novalidate>
12            <div class="field">
13              <label for="nama">Nama</label>
14              <input
15                id="nama"
16                name="nama"
17                type="text"
18                placeholder="Nama lengkap"
19                required
20              />
21              <small class="error" id="err-nama"></small>
22            </div>
23
24            <div class="field">
25              <label for="email">Email</label>
26              <input
27                id="email"
28                name="email"
29                type="email"
30                placeholder="email@domain.com"
31                required
32              />
33              <small class="error" id="err-email"></small>
34            </div>
35
36            <div class="field">
37              <label for="pesan">Pesan</label>
38              <textarea
39                id="pesan"
40                name="pesan"
41                rows="4"
42                placeholder="Tulis pesan..."
43                required
44              ></textarea>
45              <small class="error" id="err-pesan"></small>
46            </div>
47
48            <button type="submit" class="btn">Kirim</button>
49
50          <p
51            class="form-status"
52            id="formStatus"
53            role="status">
```

A modal dialog is overlaid on the code editor, containing the following data:

+	...	×
HAFIZ RAFIE ADITYA 607022400023 D3TK 48-03		

## JAVASCRIPT

The screenshot shows a code editor interface with a dark theme. On the left, a file named 'script.js' is open, displaying 51 lines of JavaScript code. The code includes functions for handling a welcome banner, animating menu items on hover, and validating a contact form. On the right, a floating modal window displays the following user information:

HAFIZ RAFIE ADITYA  
60702240023  
D3TK 48-03

```
script.js  X

M4 > script.js > ...
1 // Pesan selamat datang
2 document.addEventListener("DOMContentLoaded", () => {
3     const banner = document.getElementById("welcomeBanner");
4     const closer = document.getElementById("closeWelcome");
5     setTimeout(() => {
6         if (banner) banner.hidden = false;
7     }, 350);
8     closer?.addEventListener("click", () => (banner.hidden = true));
9 });
10
11 // Animasi hover menu via JS (tambah/hapus class)
12 function navHoverAnimation() {
13     const links = document.querySelectorAll("#mainNav a");
14     links.forEach((a) => {
15         a.addEventListener("mouseenter", () => a.classList.add("hover-bounce"));
16         a.addEventListener("mouseleave", () => a.classList.remove("hover-bounce"));
17     });
18 })();
19
20 // Validasi form sederhana
21 function formValidation() {
22     const form = document.getElementById("contactForm");
23     const status = document.getElementById("formStatus");
24     const err = {
25         nama: document.getElementById("err-nama"),
26         email: document.getElementById("err-email"),
27         pesan: document.getElementById("err-pesan"),
28     };
29
30     const emailOK = (v) => /^[^@\s]+@[^\s@]+\.\[^@\s]+\$/ .test(v);
31
32     form?.addEventListener("submit", (e) => {
33         e.preventDefault();
34         // ambil nilai
35         const nama = document.getElementById("nama").value.trim();
36         const email = document.getElementById("email").value.trim();
37         const pesan = document.getElementById("pesan").value.trim();
38
39         // reset error
40         Object.values(err).forEach((el) => (el.textContent = ""));
41
42         let valid = true;
43         if (!nama) {
44             err.nama.textContent = "Nama wajib diisi.";
45             valid = false;
46         }
47         if (!email) {
48             err.email.textContent = "Email wajib diisi.";
49             valid = false;
50         } else if (!emailOK(email)) {
51             err.email.textContent = "Format email tidak valid.";
```

Ln 19, Col 1 Spaces: 2 UTF-8 CRLF {} JavaScript ⚡ Go Live ✨ Prettier

```

41  let valid = true;
42  if (!nama) {
43    err.nama.textContent = "Nama wajib diisi.";
44    valid = false;
45  } const email: any
46  if (!email) {
47    err.email.textContent = "Email wajib diisi.";
48    valid = false;
49  } else if (!emailOK(email)) {
50    err.email.textContent = "Format email tidak valid.";
51    valid = false;
52  }
53  if (!pesan) {
54    err.pesan.textContent = "Pesan wajib diisi.";
55    valid = false;
56  }

57  if (!valid) {
58    status.textContent = "";
59    return;
60  }

61  status.textContent = "Form valid. Data siap dikirim.";
62  form.reset();
63  setTimeout(() => (status.textContent = ""), 2500);
64  });
65 })();
66

```

HAFIZ RAFIE ADITYA  
607022400023  
D3TK 48-03

Penjelasan:

HTML: Menyusun kerangka halaman: header dengan logo dan <nav> berisi tautan #beranda, #artikel, #kontak; banner sambutan <div id="welcomeBanner"> yang bisa ditutup; <main> berisi <article> untuk konten utama dan <aside> untuk tips; section kontak dengan <form id="contactForm"> berisi input nama, email, pesan, masing-masing punya <small id="err-...>> untuk pesan error; footer di akhir; file script.js diload dengan defer agar DOM siap dulu.

CSS: Mengatur tema gelap responsif: variabel warna di :root, tipografi dasar, header sticky dengan blur, menu .nav fleksibel dengan efek hover, layout .grid dua kolom yang turun jadi satu kolom pada lebar kecil via @media, komponen .card untuk panel konten, form dengan fokus bercincin (box-shadow) dan tombol .btn dengan transisi; kelas .hover-bounce dipicu JS untuk animasi halus pada item menu; banner .welcome ditata agar kontras dan mudah ditutup.

JavaScript: Tiga modul ringan: saat DOMContentLoaded, banner sambutan dimunculkan setelah 350 ms dan bisa disembunyikan lewat tombol tutup; modul animasi nav menambahkan/menghapus kelas hover-bounce pada mouseenter/mouseleave untuk transisi visual; modul validasi form menyetop submit default, membaca nilai, mereset pesan error, mengecek wajib isi dan format email via regex, menampilkan error spesifik di <small> terkait, lalu jika valid menampilkan status "Form valid..." dan mereset form.

Link Github:

<https://github.com/hfzrfa/WEBPRO-2025/tree/main/M4>

Pertanyaan Praktikum:

1. Apa tantangan yang dihadapi saat mengintegrasikan HTML, CSS, dan JavaScript dalam satu proyek?  
Tantangan utamanya adalah menjaga struktur kode agar tetap rapi dan mudah dipahami ketika ketiga bahasa tersebut saling berinteraksi. HTML berfungsi sebagai struktur, CSS untuk tampilan, dan JavaScript untuk logika serta interaktivitas. Kesalahan kecil seperti penempatan file, selector yang tidak sesuai, atau konflik antar-style dan script sering menyebabkan tampilan atau fungsi tidak berjalan sebagaimana mestinya.
2. Bagaimana cara membuat halaman web Anda lebih interaktif menggunakan JavaScript?  
Halaman bisa dibuat lebih interaktif dengan menambahkan event seperti *onClick*, *onHover*, atau *onSubmit* untuk merespons tindakan pengguna. JavaScript juga bisa digunakan untuk membuat animasi sederhana, validasi form secara real-time, menampilkan pesan dinamis, serta mengubah konten halaman tanpa perlu memuat ulang (*DOM manipulation*).

Kesimpulan:

Dari praktikum ini, mahasiswa memahami bagaimana HTML berfungsi sebagai kerangka utama halaman web, CSS digunakan untuk mempercantik tampilan serta membuatnya responsif, dan JavaScript berperan menambah interaktivitas serta logika pada halaman. Integrasi ketiga komponen tersebut menghasilkan proyek web sederhana yang tidak hanya statis, tetapi juga menarik dan interaktif bagi pengguna.