

LAPORAN TUGAS 6

“WEB PRIBADI”



PEMROGRAMAN WEB 1

Dosen Pengampu :

Muhammad Miftakhul Syaikhudin, M.Kom

Disusun Oleh:

Ayu Indrawati (4123017)

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS PESANTREN TINGGI DARUL 'ULUM JOMBANG

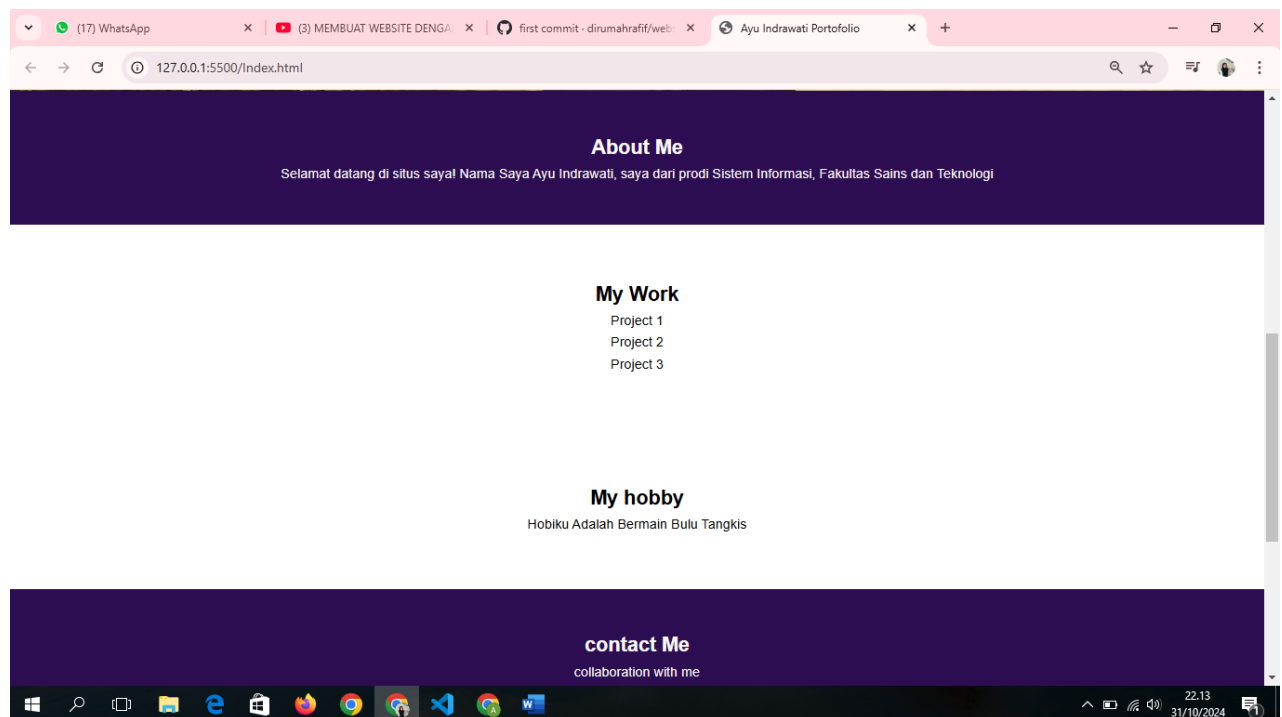
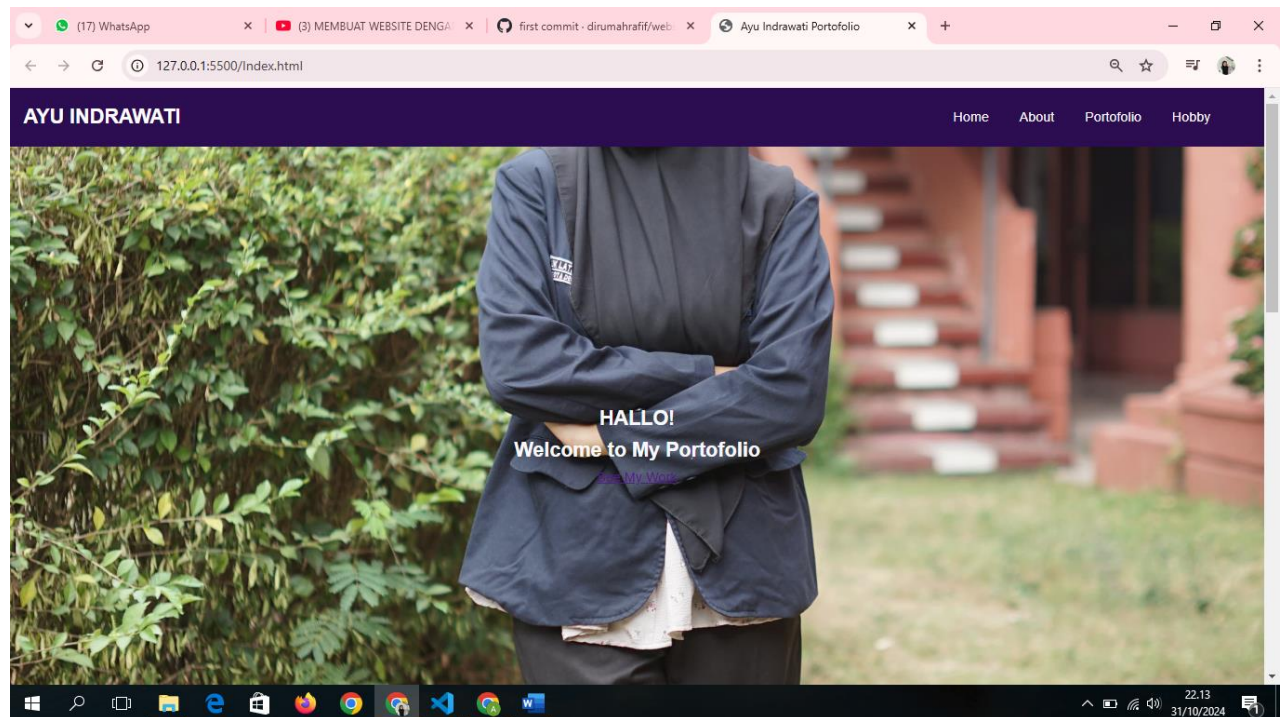
2024

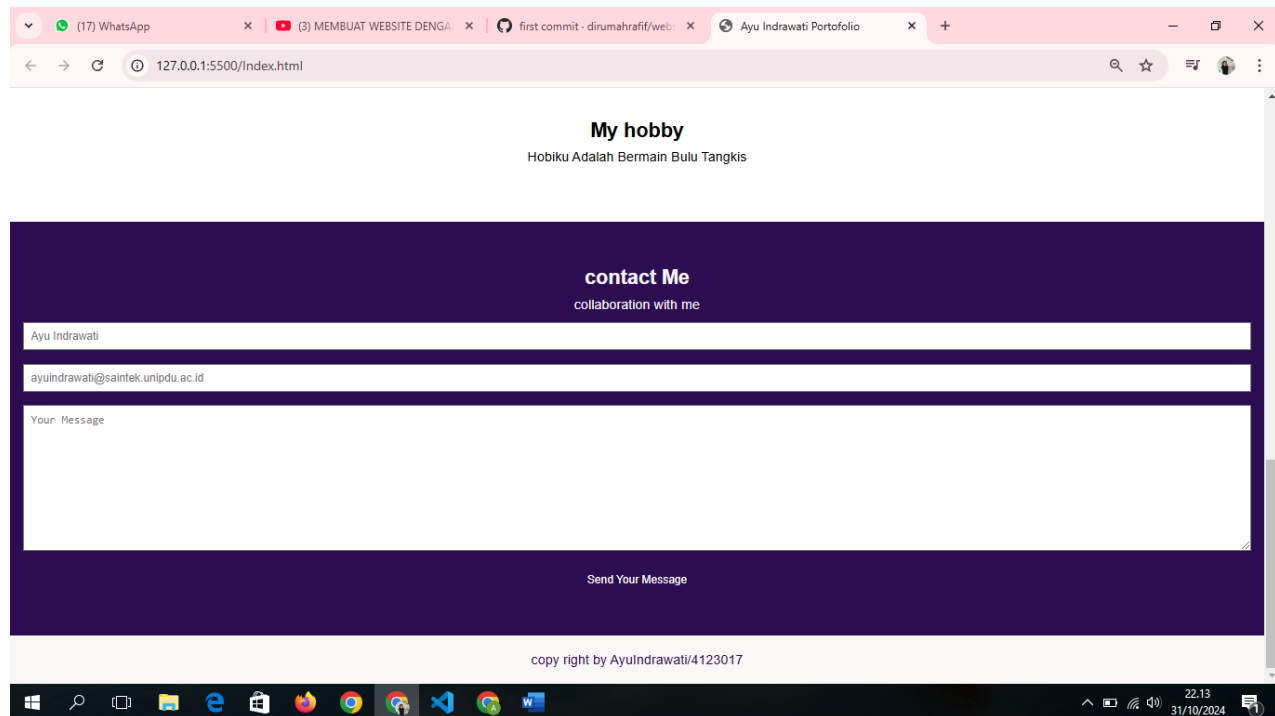
NAMA: Ayu Indrawati

MIM:4123017

KELAS: A

LINK: <https://github.com/AyuIndrawati1/PEMWEB1.git>





HTML:

1. Struktur Dasar:

- Dokumen dimulai dengan deklarasi `<!DOCTYPE html>`, yang menunjukkan bahwa ini adalah dokumen HTML5.
- Elemen `<html>` memiliki atribut `lang="en"`, menandakan bahwa konten utama dalam bahasa Inggris.

2. Bagian `<head>`:

- Mengandung informasi meta, seperti karakter set (UTF-8) dan pengaturan viewport untuk responsivitas di perangkat mobile.
- Judul halaman ditetapkan sebagai "Ayu Indrawati Portfolio".
- Tautan ke file CSS eksternal (`style.css`) untuk gaya halaman.

3. Bagian `<body>`:

- Navigasi (`<nav>`): Berisi logo dan tautan navigasi ke berbagai bagian halaman (Home, About, Portfolio, Hobbies, Contact).
- Bagian Home (`<section id="home">`): Menyambut pengunjung dengan pesan "HALLO!" dan "Welcome to My Portfolio", serta tombol untuk melihat karya.
- Bagian About (`<section id="about">`): Menyediakan informasi tentang Ayu Indrawati, termasuk latar belakang pendidikan.
- Bagian Portfolio (`<section id="portfolio">`): Menampilkan tiga proyek yang dikerjakan, meskipun detail proyek belum ditentukan.
- Bagian Hobbies (`<section id="hobbies">`): Mengungkapkan hobi Ayu, yaitu bermain bulu tangkis.
- Bagian Contact (`<section id="contact">`): Menyediakan formulir untuk pengunjung menghubungi Ayu, dengan input untuk nama, email, dan pesan.
- Footer (`<footer>`): Menyertakan informasi hak cipta.

4. Interaktivitas dan Gaya:

- Tautan ke file JavaScript eksternal (script.js) di bagian bawah sebelum penutup body, memungkinkan penambahan interaktivitas di halaman.
- File CSS (style.css) diharapkan berisi gaya untuk tata letak, warna, font, dan elemen visual lainnya agar halaman terlihat menarik.

5. Aksesibilitas:

- Formulir memiliki label untuk setiap input, yang meningkatkan aksesibilitas bagi pengguna dengan kebutuhan khusus.
- Struktur yang jelas dan penggunaan elemen semantik membantu dalam SEO dan pengalaman pengguna.

CSS:

* {

Mengatur margin dan padding semua elemen ke 0 untuk menghapus gaya default browser.
Menggunakan box-sizing: border-box agar padding dan border dihitung dalam lebar dan tinggi elemen.

body {

Mengatur font dasar untuk seluruh halaman menjadi Arial atau font sans-serif lainnya.
Mengatur line-height untuk meningkatkan keterbacaan teks.

nav {

Menggunakan flexbox untuk mengatur tata letak navigasi, dengan logo di satu sisi dan tautan di sisi lain.
Mengatur warna latar belakang navigasi dan menyesuaikan warna teks untuk logo dan tautan.

section {

Mengatur padding untuk setiap bagian (section) dan menyelaraskan teks ke tengah.

#home {

Menambahkan gambar latar belakang untuk bagian home, dengan pengaturan agar gambar tidak terulang dan mengisi seluruh area.
Mengatur tinggi bagian home menjadi 100% dari tinggi viewport (100vh).
Menggunakan flexbox untuk menyelaraskan konten di tengah secara vertikal dan horizontal.
Bagian Lain:

#about, #portfolio, #skills, #contact {

Mengatur warna latar belakang dan padding untuk bagian-bagian lain, memberikan konsistensi visual.

.portfolio-items {

Menggunakan flexbox untuk mengatur item portfolio agar dapat dibagi secara merata dan responsif.
Mengatur latar belakang, warna teks, padding, dan margin untuk setiap item portfolio.
Formulir:

form input, form textarea {

Mengatur input dan textarea dalam formulir agar memenuhi lebar penuh dan memiliki padding yang nyaman.

Mengatur tombol formulir dengan padding, warna latar belakang, dan warna teks.

footer {

Mengatur footer agar teks terpusat dengan padding yang nyaman dan warna latar belakang yang kontras.

JS:

```
document.querySelectorAll('nav a')
```

Kode ini menggunakan `document.querySelectorAll` untuk memilih semua elemen `<a>` yang terdapat dalam elemen `<nav>`. Ini menghasilkan `NodeList` yang berisi semua tautan navigasi di halaman.

```
.forEach(anchor => {
```

Metode `forEach` digunakan untuk melakukan iterasi pada setiap tautan yang telah dipilih. Setiap tautan akan diwakili oleh parameter `anchor` dalam fungsi `callback`.

```
anchor.addEventListener('click', function(e){
```

Di sini, kita menambahkan event listener untuk peristiwa `'click'` pada setiap tautan. Ketika tautan diklik, fungsi `callback` yang didefinisikan akan dijalankan.

Parameter `e`: Merupakan objek event yang berisi informasi tentang peristiwa klik yang terjadi.

```
e.preventDefault();
```

Metode ini digunakan untuk mencegah perilaku default dari tautan, yang biasanya akan mengarahkan pengguna ke lokasi yang ditunjuk oleh atribut `href` secara langsung.

```
document.querySelector(this.getAttribute('href'))
```

Kode ini mengambil nilai dari atribut `href` tautan yang diklik menggunakan `this.getAttribute('href')`. Misalnya, jika tautan memiliki `href="#about"`, maka kita akan mencari elemen dengan ID `about`.

```
.scrollIntoView({  
  behavior: 'smooth'  
});
```

Metode `scrollIntoView` digunakan untuk menggulir halaman sehingga elemen target (yang telah ditemukan sebelumnya) menjadi terlihat di dalam viewport.

Opsi { `behavior: 'smooth'` }: Mengatur efek scroll menjadi halus, sehingga transisi ke elemen target terasa lebih alami dan tidak mendadak.