**LAPORAN**

**PRAKTIKUM PEMROGRAMAN WEB CLIENT**

**Pertemuan-12**



**Disusun oleh :**

Nama : Catur Setyono

NIM : 233210009

**PROGRAM STUDI SIA**

**PROGRAM DIPLOMA TIGA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA**

**YOGYAKARTA**

**2024**

**MODUL – 11**

**DOM dan Validasi Form**

1. **TUJUAN PRAKTIKUM**
2. Menuliskan script Javascript untuk membuat web dinamis menggunakan DOM.
3. Menuliskan Script Untul validasi Form.
4. **DASAR TEORI**

Pada modul ini merupakan studi kasus pembuatan desain website dengan tema Kuliner Yogyakarta



selain mengatur layout dengan css, diperlukan pembuatan gambar header menggunakan MS. Paint. Program pada modul 12 dan 13 saling terkait untuk membuat website seperti gambar di atas. Sehingga mahasiswa apabila modul 12 belum selesai maka harus dilanjutkan pada pertemuan modul 13. Atau kondisi sebaliknya, bila mahasiswa sudah selesai modul 12 namun masih ada waktu maka langsung dapat melanjutkan ke modul 13.

1. **PRAKTIK**
2. Praktik 1

A screenshot of a computer

Description automatically generated

<!DOCTYPE html>

<html lang="en">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <title>Kuliner Nikmat Jogja</title>

    <link rel="stylesheet" href="style.css" />

  </head>

  <body>

    <header class="header">

      <div class="logo"></div>

      <nav class="navbar">

        <div class="nav-link"></div>

        <div class="nav-link"></div>

        <div class="nav-link"></div>

      </nav>

    </header>

    <section class="hero">

      <div class="hero-text"></div>

      <div class="hero-image"></div>

    </section>

    <section class="content">

      <div class="content-left">

        <div class="content-box"></div>

        <div class="content-box"></div>

      </div>

      <div class="content-right">

        <div class="list-item"></div>

        <div class="list-item"></div>

        <div class="list-item"></div>

      </div>

    </section>

    <section class="services">

      <div class="service-item"></div>

      <div class="service-item"></div>

      <div class="service-item"></div>

    </section>

    <footer class="footer">

      <div class="footer-item"></div>

      <div class="footer-item"></div>

      <div class="footer-item"></div>

    </footer>

  </body>

</html>

\* {

  margin: 0;

  padding: 0;

  box-sizing: *border-box*;

}

body {

  font-family: *Arial*, *sans-serif*;

}

.header {

  display: *flex*;

  justify-content: *space-between*;

  align-items: *center*;

  padding: 1rem;

  background-color: #1f2937;

}

.logo {

  width: 100px;

  height: 40px;

  background-color: #4ade80;

}

.navbar {

  display: *flex*;

  gap: 1rem;

}

.nav-link {

  width: 80px;

  height: 20px;

  background-color: #4ade80;

}

.hero {

  display: *flex*;

  flex-wrap: *wrap*;

  padding: 2rem;

  gap: 1rem;

}

.hero-text {

  flex: 1;

  height: 150px;

  background-color: #1f2937;

}

.hero-image {

  flex: 1;

  height: 150px;

  background-color: #4ade80;

}

.content {

  display: *flex*;

  flex-wrap: *wrap*;

  gap: 1rem;

  padding: 2rem;

}

.content-left,

.content-right {

  flex: 1;

}

.content-box {

  height: 100px;

  background-color: #4ade80;

  margin-bottom: 1rem;

}

.list-item {

  height: 50px;

  background-color: #1f2937;

  margin-bottom: 1rem;

}

.services {

  display: *flex*;

  justify-content: *space-around*;

  padding: 2rem;

}

.service-item {

  width: 100px;

  height: 100px;

  background-color: #4ade80;

  border-radius: 50%;

}

.footer {

  display: *flex*;

  justify-content: *space-around*;

  align-items: *center*;

  padding: 1rem;

  background-color: #1f2937;

}

.footer-item {

  width: 50px;

  height: 50px;

  background-color: #1f2937;

  border-radius: 50%;

}

@media (max-width: 768px) {

  .content {

    flex-direction: *column*;

  }

  .services {

    flex-direction: *column*;

  }

  .footer {

    flex-direction: *column*;

  }

  .footer-item {

    margin-bottom: 1rem;

  }

  .service-item {

    margin-bottom: 1rem;

  }

  .content-left,

  .content-right {

    flex: 1;

  }

  .hero {

    flex-direction: *column*;

  }

  .hero-text,

  .hero-image {

    flex: 1;

  }

  .nav-link {

    width: 50px;

  }

  .logo {

    width: 50px;

  }

  .list-item {

    height: 30px;

  }

  .service-item {

    width: 50px;

    height: 50px;

  }

  .footer-item {

    width: 30px;

    height: 30px;

  }

  .content-box {

    height: 50px;

  }

}

Penjelasan :

Kode tersebut merupakan contoh program yang Menggunakan HTML untuk penataan struktur dan CSS untuk styling, halaman ini dirancang dengan layout fleksibel menggunakan flexbox yang memudahkan penataan elemen secara horizontal dan vertikal. Warna latar belakang yang kontras (gelap untuk header dan footer, hijau terang untuk elemen lainnya) memberi tampilan yang menarik. Kode ini juga dilengkapi dengan media query untuk menyesuaikan tampilan pada perangkat dengan layar kecil, seperti ponsel dan tablet, dengan mengubah tata letak menjadi vertikal dan menyesuaikan ukuran elemen. Meskipun elemen-elemen halaman seperti logo, navigasi, layanan, dan konten masih berupa placeholder, kode ini menyediakan fondasi yang solid untuk mengembangkan halaman website yang responsif dan modern.

1. **KESIMPULAN**

Dalam praktik ini kita diminta untuk menyelesaikan sebuah kasus jika diminta untuk membuat sebuah website kuliner jogja dengan menggunakan pemahaman html dan css untuk membuat layoutnya terlebih dahulu sehingga nantinya memudahkan kita untuk mengisikan konten apa yang hendak kita buat didalamnya. Ini sangat berguna untuk kerangka dasar yang menentukan struktur dan alur visual halaman. Dengan merancang layout terlebih dahulu, pengembang dapat memastikan setiap elemen, seperti header, navigasi, konten utama, dan footer, memiliki penempatan yang strategis untuk menciptakan pengalaman pengguna yang baik. Layout yang terencana juga membantu menjaga konsistensi desain, mempermudah proses pengisian konten, dan memastikan responsivitas di berbagai perangkat. Selain itu, layout yang rapi dan fungsional memudahkan tim untuk berkolaborasi, mengidentifikasi kebutuhan desain, dan melakukan penyesuaian sebelum konten final dimasukkan. Dengan begitu, waktu pengembangan menjadi lebih efisien dan hasil akhir lebih optimal.