

**LAPORAN FINAL PROJECT**  
**PEMRORGAMAN WEB**



Disusun oleh:

**Achmad Fitto Rizky** (L0123005)

**Afilia Rahmadani** (L0123009)

**Bintang Cahaya Putra** (L0123037)

**PROGRAM STUDI INFORMATIKA**  
**FAKULTAS TEKNOLOGI INFORMASI DAN SAINS DATA**  
**UNIVERSITAS SEBELAS MARET**

**2025**

## **BAB I**

### **PENDAHULUAN**

#### **A. LATAR BELAKANG**

Perkembangan teknologi informasi yang pesat telah mendorong berbagai bidang usaha untuk melakukan digitalisasi, termasuk dalam bidang usaha konveksi. Salah satu bentuk digitalisasi yang paling umum dan efektif adalah melalui media website. Website dapat menjadi sarana promosi, informasi, sekaligus komunikasi antara penyedia jasa dengan pelanggan secara lebih luas, cepat, dan efisien.

CLS (Cahaya Langgeng Sentosa) GARMENT merupakan usaha konveksi yang berdiri sejak tahun 2005 dan melayani pembuatan berbagai jenis pakaian seperti kaos, kemeja, dan jaket. CLS GARMENT juga menyediakan jasa sablon serta menerima pesanan dari jumlah kecil hingga ribuan, dengan kualitas produk yang disesuaikan dengan permintaan pelanggan. Namun, selama ini penyampaian informasi produk dan proses pemesanan masih dilakukan secara konvensional, yang berpotensi menimbulkan keterlambatan informasi atau miskomunikasi.

Untuk menjawab tantangan tersebut, dibutuhkan sebuah website yang dapat membantu mempublikasikan informasi mengenai produk, layanan, dan kontak pemesanan secara terstruktur dan mudah diakses kapan saja. Melalui pengembangan website ini, diharapkan proses interaksi antara pelanggan dan pihak CLS GARMENT menjadi lebih cepat, efisien, dan profesional.

Oleh karena itu, dalam proyek akhir ini akan dilakukan perancangan dan pengembangan website CLS GARMENT menggunakan teknologi web seperti HTML, CSS, dan JavaScript dengan tambahan framework Vite.js. Website ini akan memuat halaman-halaman penting seperti galeri produk, formulir pemesanan, hingga kontak. Diharapkan, hasil dari proyek ini mampu mendukung digitalisasi usaha CLS GARMENT sekaligus memberikan pengalaman yang baik bagi pengguna.

#### **B. RUMUSAN MASALAH**

Rumusan masalah dalam proyek ini dirumuskan sebagai berikut:

1. Bagaimana membangun sebuah website statis/dinamis yang dapat menampilkan informasi layanan dan produk dari CLS GARMENT?

2. Bagaimana merancang tampilan website yang menarik, interaktif, dan responsif agar mudah digunakan oleh pengunjung?
3. Bagaimana mengimplementasikan fitur form pemesanan produk agar pengguna dapat memesan langsung melalui website?

## C. TUJUAN

Adapun tujuan dari proyek pengembangan website ini adalah:

1. Mengembangkan website yang berfungsi sebagai media informasi dan promosi bagi usaha konveksi CLS GARMENT.
2. Menyediakan fitur pemesanan dan kontak.
3. Merancang tampilan user interface yang menarik, mudah dipahami, dan dapat diakses melalui berbagai perangkat (responsif).
4. Mengimplementasikan fitur form input yang memungkinkan pelanggan mengisi data pemesanan secara online.

## D. MANFAAT

1. Bagi Konsumen:
  - Mempermudah dalam mengakses informasi produk dan layanan konveksi secara cepat dan praktis melalui website
  - Memberikan kenyamanan dalam proses pemesanan, termasuk pemilihan jenis pakaian, jumlah pesanan, bahan, dan desain yang diinginkan.
  - Meningkatkan kepuasan pelanggan melalui layanan yang responsif dan transparan.
2. Bagi Produsen:
  - Membantu memperluas jangkauan pasar dengan memanfaatkan teknologi digital.
  - Meningkatkan efisiensi pengelolaan pesanan dan komunikasi dengan pelanggan.
  - Meningkatkan citra profesional dan daya saing usaha di era industri 4.0.
3. Bagi Developer:

- Memberikan pengalaman dalam merancang dan mengembangkan website.
- Menjadi media pembelajaran dalam mengimplementasikan konsep navigasi, dan pengolahan data dalam pengembangan aplikasi nyata.

## **BAB II**

### **ANALISIS KEBUTUHAN SISTEM**

Analisis kebutuhan sistem merupakan tahapan penting dalam perancangan dan pengembangan aplikasi web. Tujuannya adalah untuk menentukan apa saja yang harus disediakan oleh sistem agar dapat berjalan sesuai kebutuhan pengguna.

#### 1. Kebutuhan Fungsional

Kebutuhan fungsional adalah fitur-fitur yang harus tersedia dan dapat dijalankan oleh sistem. Berdasarkan tampilan halaman utama, berikut kebutuhan fungsional yang teridentifikasi:

##### a. Navigasi Halaman

- Terdapat menu navigasi utama seperti: Beranda, Tentang Kami, Layanan, Proses, Galeri, Pemesanan, dan Kontak.
- Navigasi tersebut bersifat klik-scroll atau redirect ke bagian yang sesuai.

##### b. Tampilan Halaman Utama (Landing Page)

- Menampilkan headline "Jasa Konveksi Pakaian Premium".
- Menampilkan deskripsi singkat tentang layanan CLS Garment.
- Menampilkan 2 tombol aksi utama:
  1. "Pesan Sekarang" → diarahkan ke form pemesanan.
  2. "Hubungi Kami" → diarahkan ke kontak (mungkin WhatsApp, Email, dll.).

##### c. Responsif di Berbagai Perangkat

- Website harus tampil optimal di perangkat desktop, tablet, dan smartphone.

##### d. Form Pemesanan (Fitur Lanjutan)

- Form input nama, jenis produk, jumlah, kontak pelanggan.
- Tombol submit untuk mengirim data.

##### e. Halaman Galeri Produk (Fitur Lanjutan)

- Menampilkan gambar-gambar produk/portofolio hasil konveksi.

##### f. Halaman Kontak

- Menampilkan informasi kontak: alamat, email, WhatsApp, atau peta lokasi.
2. Kebutuhan non-fungsional
- Kebutuhan non-fungsional berkaitan dengan kualitas sistem dan aspek teknis non-fitur:
- a. Kinerja Sistem (Performance)
    - Website dapat diakses dengan cepat (< 3 detik loading time).
    - Akses ringan meskipun koneksi lambat (optimasi gambar dan layout).
  - b. Reliabilitas
    - Website selalu tersedia 24/7.
    - Tidak menampilkan error saat navigasi antar halaman.
  - c. Keamanan (Security)
    - Menggunakan protokol HTTPS (terbukti pada URL).
    - Jika menggunakan form, data harus dikirim secara aman (misal validasi input, proteksi terhadap spam).
  - d. Kesesuaian Desain
    - Tampilan profesional dan sesuai identitas usaha CLS Garment (warna, font, dan logo konsisten).
  - e. Kemudahan Penggunaan (Usability)
    - Navigasi mudah dimengerti pengguna awam. Tombol dan menu terlihat jelas dan mudah dijangkau.

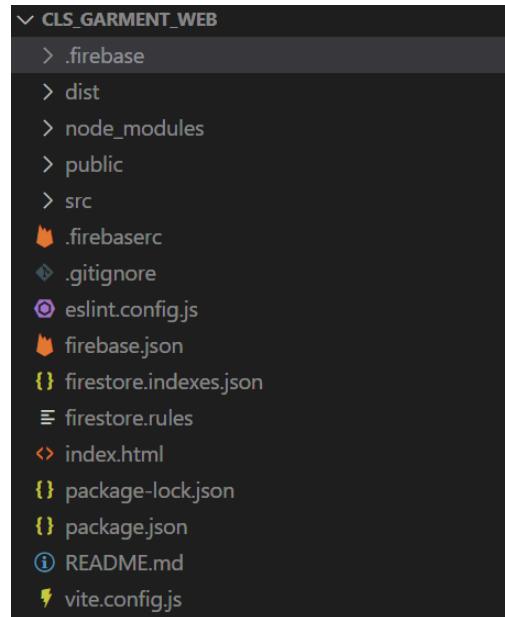
## **BAB III**

### **IMPLEMENTASI DAN PENGUJIAN**

#### **A. ENVIRONMENT**

Environment yang digunakan dalam pengembangan website CLS GARMENT adalah Visual Code Studio dengan HTML, CSS, JavaScript, dan [Vite.js](#) dengan Firebase sebagai hostinger,

#### **B. STRUKTUR FOLDER PROJECT**



Proyek website CLS Garment dikembangkan menggunakan Vite sebagai build tool modern dan Firebase sebagai platform hosting. Struktur direktori pada proyek ini dirancang untuk mendukung pengembangan dan deployment web secara efisien. Folder utama seperti src berisi seluruh source code aplikasi, termasuk file HTML, CSS, dan JavaScript.

Di dalam folder terdapat komponen user interface dan logika interaktif situs web. Sedangkan folder public digunakan untuk menyimpan asset statis seperti gambar dan file. Setelah proses build dijalankan, hasil akhir dari aplikasi akan dikompilasi ke dalam folder dist. Folder ini merupakan output dari proses bundling dan merupakan direktori yang akan di deploy ke Firebase Hosting sesuai pengaturan dalam file firebase.json. Selain itu, file .firebaserc dan firebase.json mengatur koneksi serta konfigurasi antara proyek lokal dengan proyek Firebase yang dituju. File firestore.rules dan firestore.indexes.json disiapkan jika aplikasi menggunakan Firestore sebagai basis data, untuk mengatur keamanan dan performa query database.

Pada sisi konfigurasi proyek, terdapat vite.config.js yang mengatur berbagai parameter build dan pengembangan menggunakan Vite. Kemudian package.json dan package-lock.json menyimpan informasi dependensi proyek dan skrip-skrip penting untuk pengelolaan proyek seperti menjalankan server lokal, membangun proyek, atau melakukan testing.

## C. IMPLEMENTASI FITUR

Berikut ini adalah implementasi fitur dari website CLS Garment yaitu source code dan output dari code.

### 1. Source Code

Berikut ini adalah source code yang digunakan:

#### 1.1. Home.jsx

Website ini dirancang untuk memberikan pengalaman pemesanan yang lengkap dan user-friendly untuk jasa konveksi, dengan fokus pada kemudahan penggunaan dan proses bisnis yang jelas. Kode berikut ini adalah aplikasi React untuk website CLS Garment yang cukup kompleks dengan banyak fitur. Berikut penjelasan detail per fiturnya:

##### 1.1.1. Data Management

###### - BAHAN PER-PRODUK



```
1 const bahanPerProduk = {
2     kaos: [
3         "Cotton Combed 20s",
4         "Cotton Combed 24s",
5         "Cotton Combed 30s",
6         "Cotton Carded",
7         "Polyester",
8         "TC",
9         "CVC",
10        "Bamboo Cotton"
11    ],
12    jersey: [
13        "Dry Fit",
14        "Polyester",
15        "Hyget",
16        "Spandek",
17        "Serena",
18        "Lotto",
19        "Paragon"
20    ],
21    polo: [
22        "Lacoste CVC",
23        "Lacoste PE",
24        "Cotton Combed Piqué",
25        "Lacoste Cotton",
26        "TC Piqué"
27    ],
28    hoodie: [
29        "Fleece",
30        "Baby Terry",
31        "CVC Fleece",
32        "PE Fleece"
33    ],
34    sweater: [
35        "Baby Terry",
36        "Cotton Fleece",
37        "Knitting Wool",
38        "CVC Fleece"
39    ],

```

```
40     kemeja: [
41         "American Drill",
42         "Japan Drill",
43         "Oxford",
44         "Katun Twill",
45         "Tropical",
46         "Ripstop",
47         "Canvas"
48     ],
49     workshirt: [
50         "Japan Drill",
51         "American Drill",
52         "Ripstop",
53         "Kanvas",
54         "Tropical",
55         "Twill"
56     ]
57 };
```

**Fungsi:** Menyimpan daftar bahan yang tersedia untuk setiap jenis produk

**Implementasi:** Object dengan key = jenis produk, value = array bahan

- **HARGA PRODUK**

```
1 const hargaProduk = {
2   kaos: {
3     "cotton-combed-20s": 62000,
4     "cotton-combed-24s": 60000,
5     "cotton-combed-30s": 58000,
6     "cotton-carded": 55000,
7     polyester: 52000,
8     tc: 53000,
9     cvc: 54000,
10    "bamboo-cotton": 68000
11  },
12  jersey: {
13    "dry-fit": 55000,
14    polyester: 52000,
15    hyget: 45000,
16    spandek: 58000,
17    serena: 60000,
18    lotto: 62000,
19    paragon: 64000
20  },
21  "polo": {
22    "lacoste-cvc": 66000,
23    "lacoste-pe": 64000,
24    "cotton-combed-pique": 68000,
25    "lacoste-cotton": 70000,
26    "tc-pique": 65000
27  },
28  hoodie: {
29    fleece: 85000,
30    "baby-terry": 82000,
31    "cvc-fleece": 88000,
32    "pe-fleece": 80000
33  },
34  sweater: {
35    "baby-terry": 75000,
36    "cotton-fleece": 78000,
37    "knitting-wool": 85000,
38    "cvc-fleece": 77000
39  },
40  kemeja: {
41    "american-drill": 115000,
42    "japan-drill": 120000,
43    oxford: 110000,
44    "katun-twill": 125000,
45    tropical: 108000,
46    ripstop: 118000,
47    canvas: 122000
}
```

```
48     },
49     workshirt: {
50       "japan-drill": 120000,
51       "american-drill": 115000,
52       ripstop: 118000,
53       kanvas: 122000,
54       tropical: 108000,
55       twill: 112000
56     }
57   };

```

**Fungsi:** Menyimpan harga satuan untuk setiap kombinasi produk-bahan

**Format:** Key menggunakan format kebab-case untuk konsistensi

#### - UTILITY FUNCTION

```
● ● ●
1  function getHargaSatuan(jenisProduk, jenisKain) {
2    const produkKey = jenisProduk.toLowerCase().replace(/\s+/g, "-");
3    const kainKey = jenisKain.toLowerCase().replace(/\s+/g, "-");
4    return hargaProduk[produkKey]?.[kainKey] || 0;
5  }
```

**Fungsi:** Mengkonversi input user ke format key dan mengambil harga

**Keamanan:** Menggunakan optional chaining untuk menghindari error

#### 1.1.2. Firebase Integration

##### - SETUP

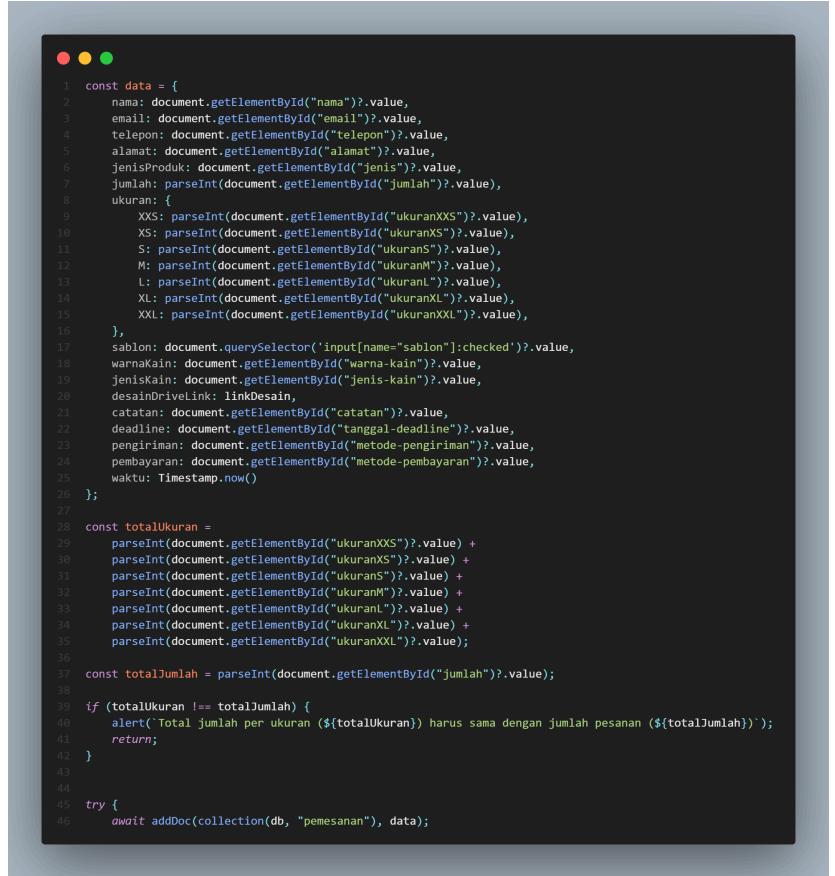


```
1 import { db } from '../firebase';
2 import { useEffect } from 'react';
3 import { collection, addDoc, Timestamp } from "firebase/firestore";
```

**Database:** Menggunakan Firestore untuk menyimpan data pesanan

**Timestamp:** Mencatat waktu pemesanan secara otomatis

#### - DATA SUBMISSION



```
1 const data = {
2   nama: document.getElementById("nama")?.value,
3   email: document.getElementById("email")?.value,
4   telepon: document.getElementById("telepon")?.value,
5   alamat: document.getElementById("alamat")?.value,
6   jenisProduk: document.getElementById("jenis")?.value,
7   jumlah: parseInt(document.getElementById("jumlah")?.value),
8   ukuran: {
9     XXS: parseInt(document.getElementById("ukuranXXS")?.value),
10    XS: parseInt(document.getElementById("ukuranXS")?.value),
11    S: parseInt(document.getElementById("ukuranS")?.value),
12    M: parseInt(document.getElementById("ukuranM")?.value),
13    L: parseInt(document.getElementById("ukuranL")?.value),
14    XL: parseInt(document.getElementById("ukuranXL")?.value),
15    XXL: parseInt(document.getElementById("ukuranXXL")?.value),
16  },
17  sablon: document.querySelector('input[name="sablon"]:checked')?.value,
18  warnaKain: document.getElementById("warna-kain")?.value,
19  jenisKain: document.getElementById("jenis-kain")?.value,
20  desainDriveLink: linkDesain,
21  catatan: document.getElementById("catatan")?.value,
22  deadline: document.getElementById("tanggal-deadline")?.value,
23  pengiriman: document.getElementById("metode-pengiriman")?.value,
24  pembayaran: document.getElementById("metode-pembayaran")?.value,
25  waktu: Timestamp.now()
26 };
27
28 const totalUkuran =
29   parseInt(document.getElementById("ukuranXXS")?.value) +
30   parseInt(document.getElementById("ukuranXS")?.value) +
31   parseInt(document.getElementById("ukuranS")?.value) +
32   parseInt(document.getElementById("ukuranM")?.value) +
33   parseInt(document.getElementById("ukuranL")?.value) +
34   parseInt(document.getElementById("ukuranXL")?.value) +
35   parseInt(document.getElementById("ukuranXXL")?.value);
36
37 const totalJumlah = parseInt(document.getElementById("jumlah")?.value);
38
39 if (totalUkuran !== totalJumlah) {
40   alert(`Total jumlah per ukuran (${totalUkuran}) harus sama dengan jumlah pesanan (${totalJumlah})`);
41   return;
42 }
43
44 try {
45   await addDoc(collection(db, "pemesanan"), data);
46 }
```

**Struktur Data:** Lengkap dengan semua informasi pesanan

**Validasi:** Data ukuran disimpan sebagai object terpisah

#### 1.1.3. Navigation & UI Features

#### - RESPONSIVE NAVIGATION

```
● ● ●  
1 const burgerBtn = document.getElementById('burgerBtn');  
2 const navLinks = document.getElementById('navLinks');  
3 const toggleBurger = () => navLinks?.classList.toggle("show");
```

**Mobile Menu:** Toggle burger menu untuk tampilan mobile

**Event Handler:** Click listener untuk buka/tutup menu

#### - AUTO-HIDE HEADER ON SCROLL

```
● ● ●  
1 const handleScroll = () => {  
2     const currentScroll = window.scrollY;  
3  
4     if (window.innerWidth <= 768) { // hanya untuk mobile  
5         if (currentScroll > lastScrollTop && currentScroll > 100) {  
6             header.classList.add('hide-navbar');  
7         } else {  
8             header.classList.remove('hide-navbar');  
9         }  
10    }  
11  
12    lastScrollTop = currentScroll <= 0 ? 0 : currentScroll;  
13};
```

**Fungsi:** Header otomatis sembunyi saat scroll down di mobile

**Threshold:** Mulai bekerja setelah scroll > 100px

#### - SMOOTH SCROLLING

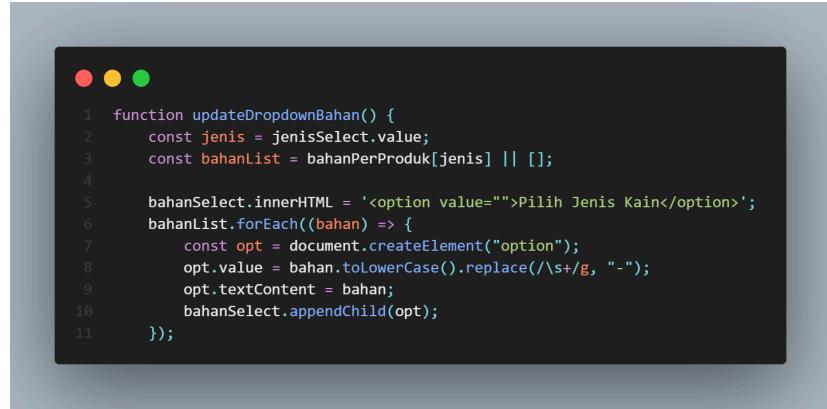
```
● ● ●  
1 anchors.forEach(anchor => {  
2     anchor.addEventListener('click', function (e) {  
3         e.preventDefault();  
4         const target = document.querySelector(this.getAttribute('href'));  
5         if (target) {  
6             target.scrollIntoView({  
7                 behavior: 'smooth',  
8             });  
9         }  
10    });  
11});
```

**Fungsi:** Smooth scroll untuk navigasi internal (anchor links)

**UX:** Mencegah default jump behavior

#### 1.1.4. Dynamic Form Features

##### - DYNAMIC MATERIAL SELECTION



```
● ● ●
1 function updateDropdownBahan() {
2     const jenis = jenisSelect.value;
3     const bahanList = bahanPerProduk[jenis] || [];
4
5     bahanSelect.innerHTML = '<option value="">Pilih Jenis Kain</option>';
6     bahanList.forEach((bahan) => {
7         const opt = document.createElement("option");
8         opt.value = bahan.toLowerCase().replace(/\s+/g, "-");
9         opt.textContent = bahan;
10        bahanSelect.appendChild(opt);
11    });
12}
```

**Fungsi:** Dropdown bahan berubah otomatis sesuai jenis produk yang dipilih

**Dynamic:** Event listener pada perubahan jenis produk

##### - PRICE CALCULATOR



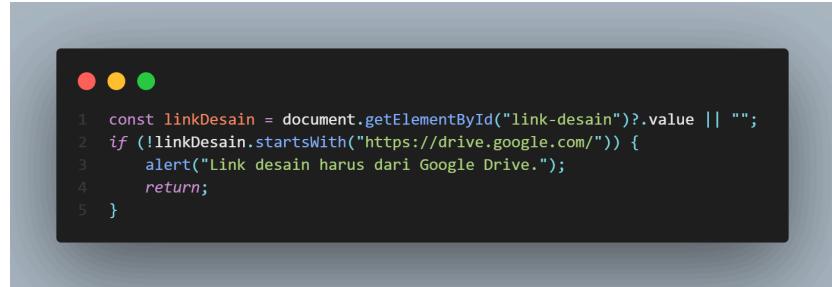
```
● ● ●
1 if (hitungBtn && hasilHarga) {
2     hitungBtn.addEventListener("click", () => {
3         const jenis = document.getElementById("jenis")?.value;
4         const kain = document.getElementById("jenis-kain")?.value;
5         const jumlah = parseInt(document.getElementById("jumlah")?.value) || 0;
6
7         const hargaSatuan = getHargaSatuan(jenis, kain);
8         const total = hargaSatuan * jumlah;
9
10        if (!jenis || !kain || !jumlah) {
11            hasilHarga.textContent = "Lengkapi pilihan produk, kain, dan jumlah.";
12            return;
13        }
14
15        if (hargaSatuan === 0) {
16            hasilHarga.textContent = "Kombinasi produk dan kain tidak tersedia.";
17            return;
18        }
19
20        hasilHarga.innerHTML = `
21         Harga satuan: Rp${hargaSatuan.toLocaleString("id-ID")}<br>
22         Total harga: Rp${total.toLocaleString("id-ID")}
23     `;
24    });
25 }
```

**Fungsi:** Menghitung total harga berdasarkan produk, bahan, dan jumlah

**Format:** Menggunakan toLocaleString untuk format rupiah Indonesia

### 1.1.5. Form Validation

#### - GOOGLE DRIVE LINK VALIDATION

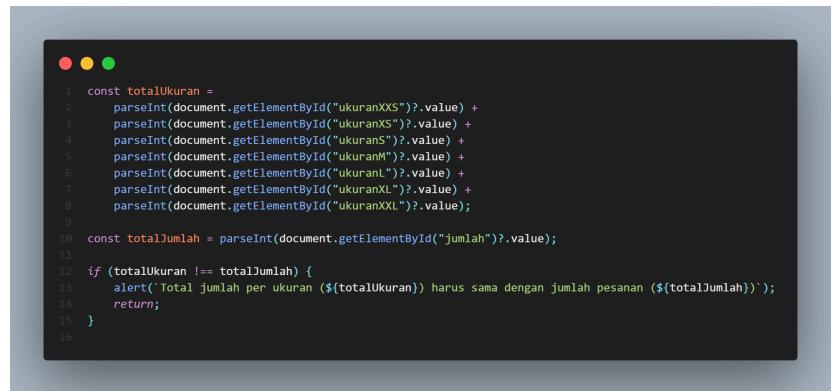


```
1 const linkDesain = document.getElementById("link-desain")?.value || "";
2 if (!linkDesain.startsWith("https://drive.google.com/")) {
3     alert("Link desain harus dari Google Drive.");
4     return;
5 }
```

**Fungsi:** Memastikan link desain harus dari Google Drive

**Keamanan:** Mencegah link yang tidak valid

#### - SIZE QUANTITY VALIDATION



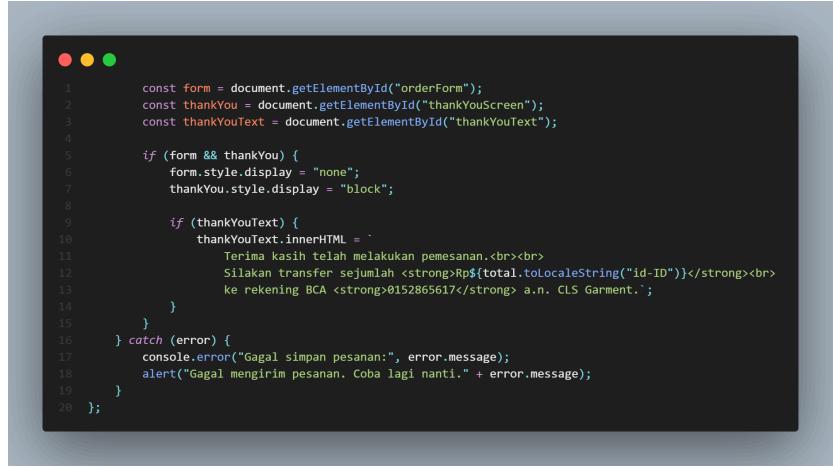
```
1 const totalUkuran =
2     parseInt(document.getElementById("ukuranXXS")?.value) +
3     parseInt(document.getElementById("ukuranXS")?.value) +
4     parseInt(document.getElementById("ukurans")?.value) +
5     parseInt(document.getElementById("ukuranM")?.value) +
6     parseInt(document.getElementById("ukuranL")?.value) +
7     parseInt(document.getElementById("ukuranXL")?.value) +
8     parseInt(document.getElementById("ukuranXXL")?.value);
9
10 const totalJumlah = parseInt(document.getElementById("jumlah")?.value);
11
12 if (totalUkuran !== totalJumlah) {
13     alert(`Total jumlah per ukuran (${totalUkuran}) harus sama dengan jumlah pesanan (${totalJumlah})`);
14     return;
15 }
16
```

**Fungsi:** Memastikan total ukuran sama dengan jumlah pesanan

**Logic:** Penjumlahan semua input ukuran harus match dengan total

### 1.1.6. User Experience Features

#### - THANK YOU SCREEN

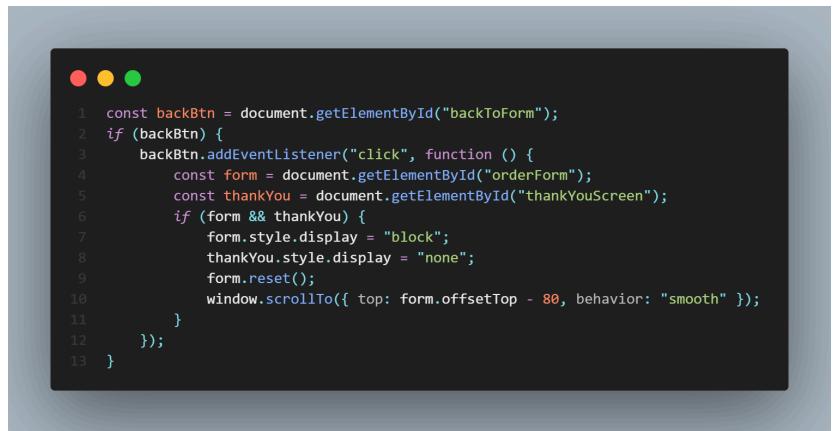


```
1 const form = document.getElementById("orderForm");
2 const thankYou = document.getElementById("thankYouScreen");
3 const thankYouText = document.getElementById("thankYouText");
4
5 if (form && thankYou) {
6     form.style.display = "none";
7     thankYou.style.display = "block";
8
9     if (thankYouText) {
10         thankYouText.innerHTML = `
11             Terima kasih telah melakukan pemesanan.<br><br>
12             Silakan transfer sejumlah <strong>Rp${total.toLocaleString("id-ID")}</strong><br>
13             ke rekening BCA <strong>0152865617</strong> a.n. CLS Garment.`;
14     }
15 } catch (error) {
16     console.error("Gagal simpan pesanan:", error.message);
17     alert("Gagal mengirim pesanan. Coba lagi nanti." + error.message);
18 }
19
20 };
```

**Fungsi:** Menampilkan screen konfirmasi setelah submit berhasil

**Info:** Menampilkan total pembayaran dan detail rekening

## - BACK TO FORM FEATURES



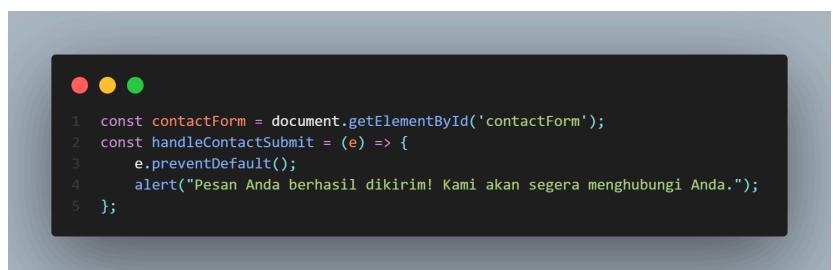
```
1 const backBtn = document.getElementById("backToForm");
2 if (backBtn) {
3     backBtn.addEventListener("click", function () {
4         const form = document.getElementById("orderForm");
5         const thankYou = document.getElementById("thankYouScreen");
6         if (form && thankYou) {
7             form.style.display = "block";
8             thankYou.style.display = "none";
9             form.reset();
10            window.scrollTo({ top: form.offsetTop - 80, behavior: "smooth" });
11        }
12    });
13 }
```

**Fungsi:** Kembali ke form dan reset semua input

**UX:** Auto scroll ke posisi form

### 1.1.7. Contact Form

## - BASIC CONTACT HANDLER



```
1 const contactForm = document.getElementById('contactForm');
2 const handleContactSubmit = (e) => {
3     e.preventDefault();
4     alert("Pesan Anda berhasil dikirim! Kami akan segera menghubungi Anda.");
5 };
```

**Fungsi:** Handler sederhana untuk form kontak

**Note:** Hanya menampilkan alert, tidak menyimpan data

### 1.1.8. Memory Management

#### - EVENT CLEANUP



```
1 return () => {
2   if (orderForm) orderForm.removeEventListener("submit", handleOrderSubmit);
3   if (contactForm) contactForm.removeEventListener("submit", handleContactSubmit);
4   if (burgerBtn) burgerBtn.removeEventListener("click", toggleBurger);
5   if (hitungBtn) hitungBtn.removeEventListener("click", () => { });
6   window.removeEventListener("scroll", handleScroll);
7 }
8 }, []);
```

**Fungsi:** Cleanup semua event listener saat component unmount

**Performance:** Mencegah memory leak

## 1.2. Style.css

### 1.2.1. CSS Variables (Custom Properties)



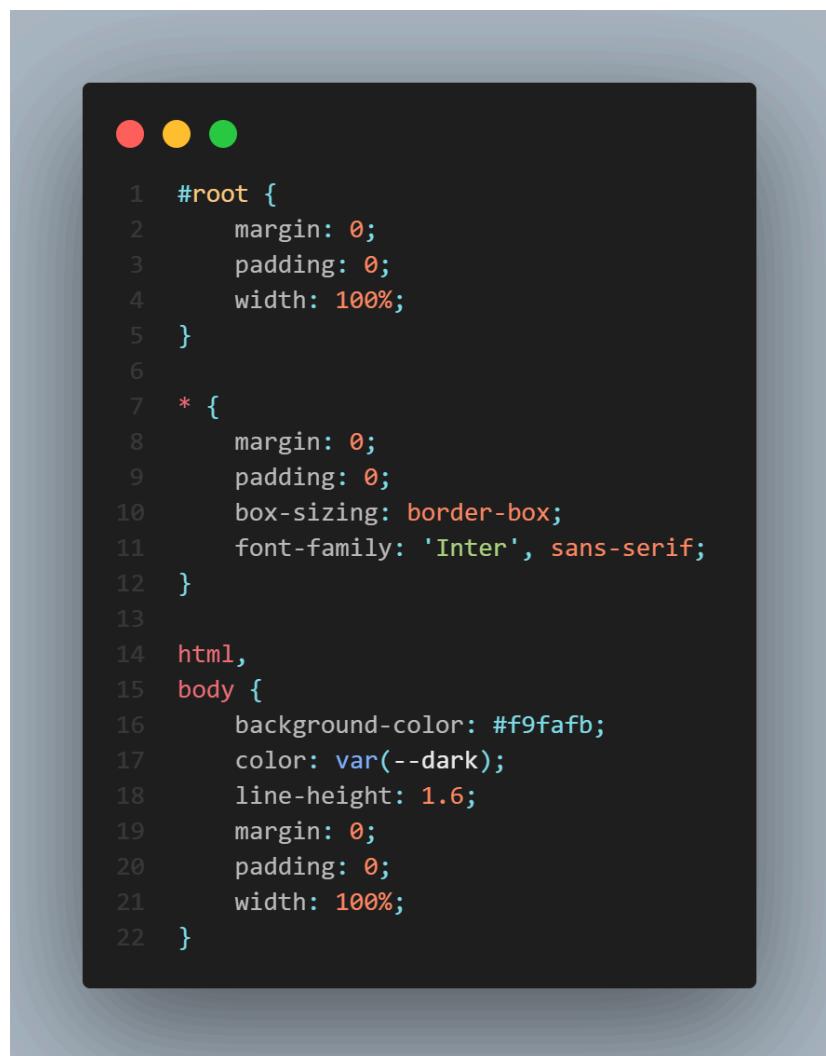
```
1 :root {
2   --primary: #3b82f6;
3   --primary-dark: #1d4ed8;
4   --secondary: #f59e0b;
5   --dark: #1f2937;
6   --light: #f3f4f6;
7   --gray: #9ca3af;
8 }
```

**Penjelasan:** Menyimpan warna-warna yang akan digunakan berulang kali di seluruh website

- primary: #3b82f6 → Warna biru utama untuk button, header

- primary-dark: #1d4ed8 → Warna biru gelap untuk hover effect
- secondary: #f59e0b → Warna kuning/orange untuk aksen
- dark: #1f2937 → Warna gelap untuk teks dan footer
- light: #f3f4f6 → Warna terang untuk background section
- gray: #9ca3af → Warna abu untuk teks sekunder

### 1.2.2. Global Reset & Root Styling



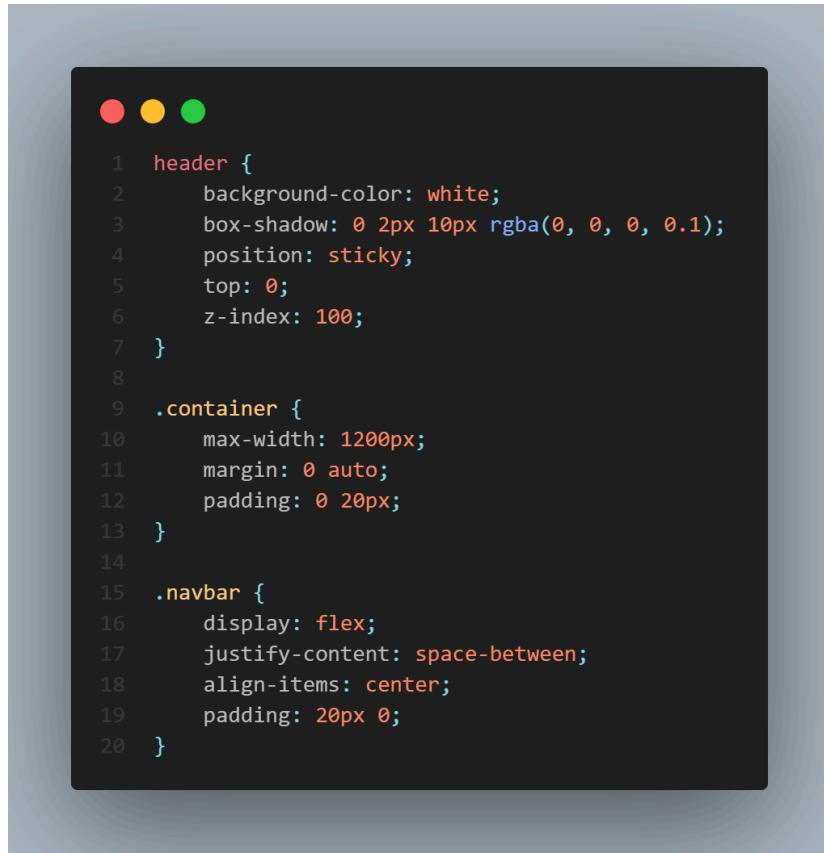
```

1  #root {
2      margin: 0;
3      padding: 0;
4      width: 100%;
5  }
6
7  * {
8      margin: 0;
9      padding: 0;
10     box-sizing: border-box;
11     font-family: 'Inter', sans-serif;
12 }
13
14 html,
15 body {
16     background-color: #f9fafb;
17     color: var(--dark);
18     line-height: 1.6;
19     margin: 0;
20     padding: 0;
21     width: 100%;
22 }

```

**Penjelasan:** Menghapus style default browser dan mengatur pengaturan dasar untuk seluruh website

### 1.2.3. Header & Navigation



```
1 header {  
2     background-color: white;  
3     box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);  
4     position: sticky;  
5     top: 0;  
6     z-index: 100;  
7 }  
8  
9 .container {  
10    max-width: 1200px;  
11    margin: 0 auto;  
12    padding: 0 20px;  
13 }  
14  
15 .navbar {  
16    display: flex;  
17    justify-content: space-between;  
18    align-items: center;  
19    padding: 20px 0;  
20 }
```

**Penjelasan:** Membuat header yang tetap terlihat saat user scroll (sticky header) dengan navigation horizontal.

### 1.2.4. Container System



```
1 .container {  
2     max-width: 1200px;  
3     margin: 0 auto;  
4     padding: 0 20px;  
5 }
```

**Penjelasan:** Membatasi lebar maksimal konten dan memusatkannya di tengah layar

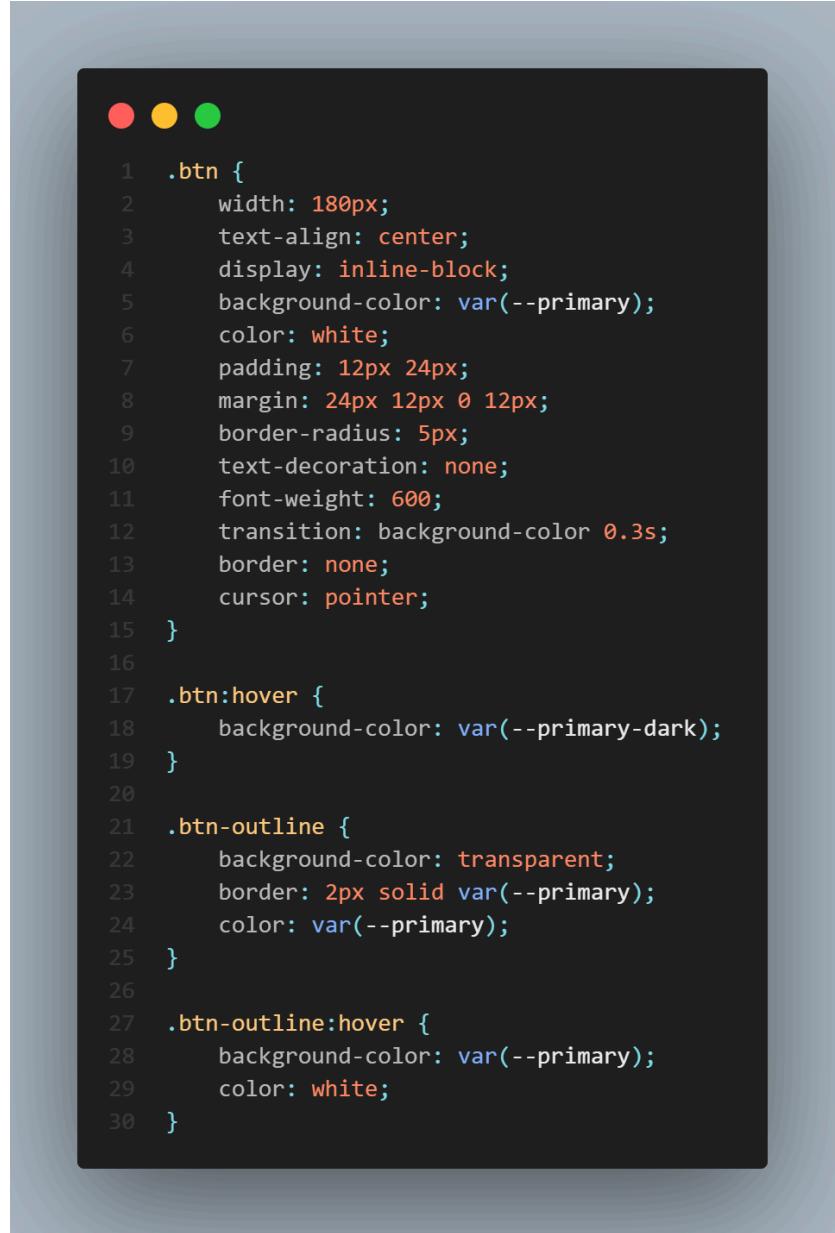
### 1.2.5. Logo & Navigation Links



```
1 .logo {
2     display: flex;
3     align-items: center;
4     font-weight: 700;
5     font-size: 24px;
6     color: var(--primary);
7 }
8
9 .logo span {
10     color: var(--secondary);
11 }
12
13 .nav-links {
14     display: flex;
15     list-style: none;
16 }
17
18 .nav-links li {
19     margin-left: 30px;
20 }
21
22 .nav-links a {
23     text-decoration: none;
24     color: var(--dark);
25     font-weight: 500;
26     transition: color 0.3s;
27 }
28
29 .nav-links a:hover {
30     color: var(--primary);
31 }
```

**Penjelasan:** Styling untuk logo dan menu navigation dengan hover effect.

### 1.2.6. Button System



```
1 .btn {
2   width: 180px;
3   text-align: center;
4   display: inline-block;
5   background-color: var(--primary);
6   color: white;
7   padding: 12px 24px;
8   margin: 24px 12px 0 12px;
9   border-radius: 5px;
10  text-decoration: none;
11  font-weight: 600;
12  transition: background-color 0.3s;
13  border: none;
14  cursor: pointer;
15 }
16
17 .btn:hover {
18   background-color: var(--primary-dark);
19 }
20
21 .btn-outline {
22   background-color: transparent;
23   border: 2px solid var(--primary);
24   color: var(--primary);
25 }
26
27 .btn-outline:hover {
28   background-color: var(--primary);
29   color: white;
30 }
```

**Penjelasan:** Sistem button dengan 2 variasi: solid dan outline, keduanya memiliki hover effect

- a. Button Primary (.btn):

- width: 180px → Lebar tetap 180px

- background-color: var(--primary) → Background biru
- border-radius: 5px → Sudut melengkung
- transition: background-color 0.3s → Animasi hover

b. Button Outline (.btn-outline):

- background-color: transparent → Background transparan
- border: 2px solid var(--primary) → Border biru 2px
- color: var(--primary) → Teks biru
- :hover → Background jadi biru, teks putih

### 1.2.7. Hero Section



```

1 .hero {
2   background: linear-gradient(rgba(0, 0, 0, 0.7), rgba(0, 0, 0, 0.7)), url('/assets/image/cls_workshop.jpg') center/cover no-repeat;
3   color: white;
4   padding: 180px 0;
5   text-align: center;
6 }
7
8 .hero h1 {
9   font-size: 48px;
10  margin-bottom: 20px;
11 }
12
13 .hero p {
14   font-size: 18px;
15   max-width: 700px;
16   margin: 0 auto 30px;
17 }

```

**Penjelasan:** Bagian pembuka website dengan background gambar dan overlay gelap untuk kontras teks.

### 1.2.8. Section Layout



```

1 .section {
2   padding: 80px 0;
3 }
4
5 .section-title {
6   font-size: 24px;
7   font-weight: bold;
8   margin-bottom: 10px;
9 }
10
11 .section-subtitle {
12   font-size: 18px;
13   font-weight: normal;
14   margin-bottom: 30px;
15 }
16
17 .section-content {
18   font-size: 16px;
19   line-height: 1.5;
20   margin-bottom: 20px;
21 }
22
23 .section-link {
24   color: inherit;
25   text-decoration: none;
26 }
27
28 .section-link::before {
29   content: '';
30   display: inline-block;
31   width: 1em;
32   height: 1em;
33   border-radius: 50%;
34   background-color: inherit;
35   vertical-align: middle;
36   margin-right: 0.5em;
37 }
38
39 .section-link::after {
40   content: 'Read More';
41   font-size: 14px;
42   font-weight: bold;
43   position: absolute;
44   right: -10px;
45   top: 50px;
46   transform: rotate(-45deg);
47 }

```

```
6      text-align: center;
7      margin-bottom: 50px;
8  }
9
10 .section-title h2 {
11   font-size: 36px;
12   margin-bottom: 15px;
13   color: var(--dark);
14 }
15
16 .section-title p {
17   color: var(--gray);
18   max-width: 700px;
19   margin: 0 auto;
20 }
```

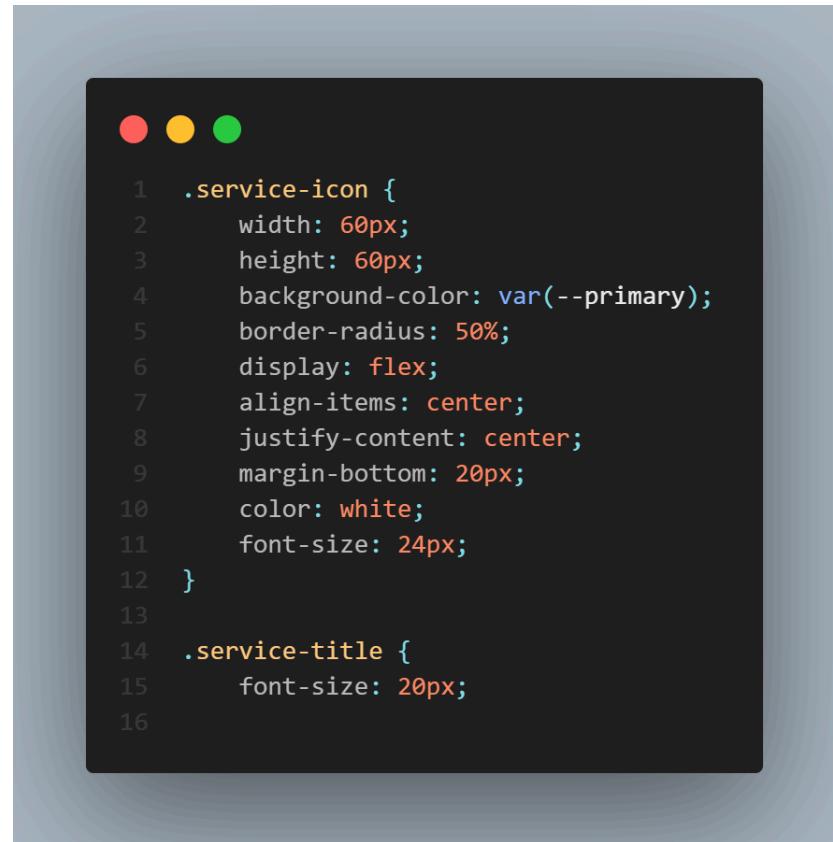
**Penjelasan:** Template layout untuk semua section di website dengan judul yang terpusat

### 1.2.9. Services & Grid Layout

```
● ● ●
1 .services {
2   display: grid;
3   grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
4   gap: 30px;
5 }
6
7 .service-card {
8   background-color: white;
9   border-radius: 10px;
10  padding: 30px;
11  box-shadow: 0 5px 15px rgba(0, 0, 0, 0.05);
12  transition: transform 0.3s;
13 }
14
15 .service-card:hover {
16   transform: translateY(-10px);
17 }
```

**Penjelasan:** Grid responsive untuk menampilkan card layanan yang otomatis menyesuaikan jumlah kolom dan memiliki efek hoover.

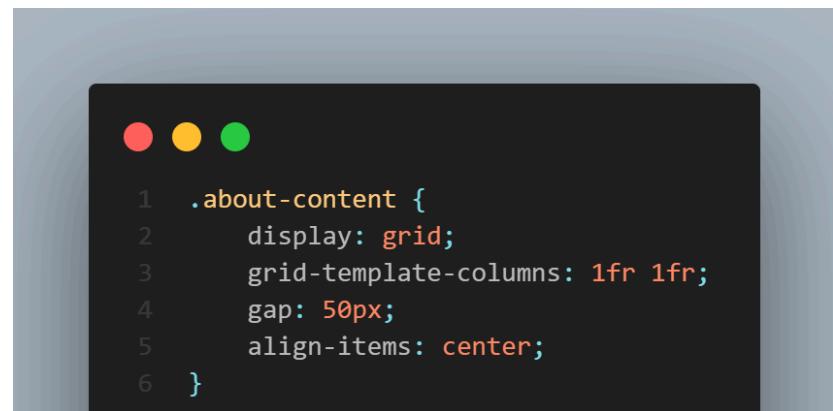
#### 1.2.10. Services Icon



```
1 .service-icon {
2     width: 60px;
3     height: 60px;
4     background-color: var(--primary);
5     border-radius: 50%;
6     display: flex;
7     align-items: center;
8     justify-content: center;
9     margin-bottom: 20px;
10    color: white;
11    font-size: 24px;
12 }
13
14 .service-title {
15     font-size: 20px;
16 }
```

**Penjelasan:** Ikon berbentuk lingkaran untuk setiap layanan dengan background biru

#### 1.2.11. About Section

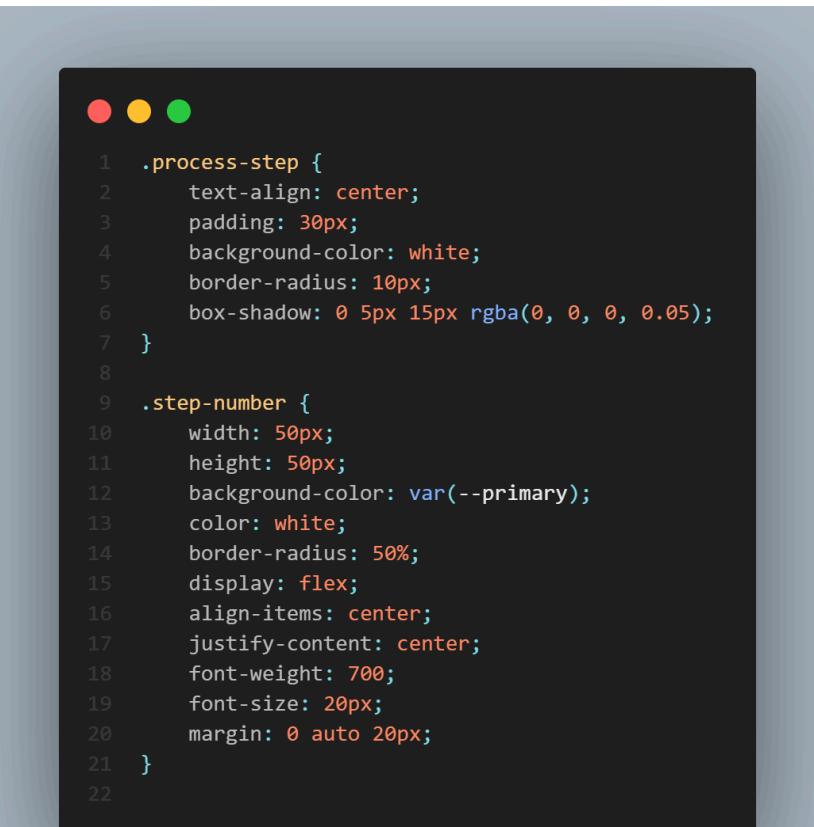


```
1 .about-content {
2     display: grid;
3     grid-template-columns: 1fr 1fr;
4     gap: 50px;
5     align-items: center;
6 }
```

```
7
8 .about-img {
9     border-radius: 10px;
10    overflow: hidden;
11 }
12
13 .about-img img {
14     width: 100%;
15     height: auto;
16     display: block;
17 }
18
19 .about-text h3 {
20     font-size: 28px;
21     margin-bottom: 20px;
22 }
```

**Penjelasan:** Layout 2 kolom dengan gambar dan teks, responsive dengan equal width.

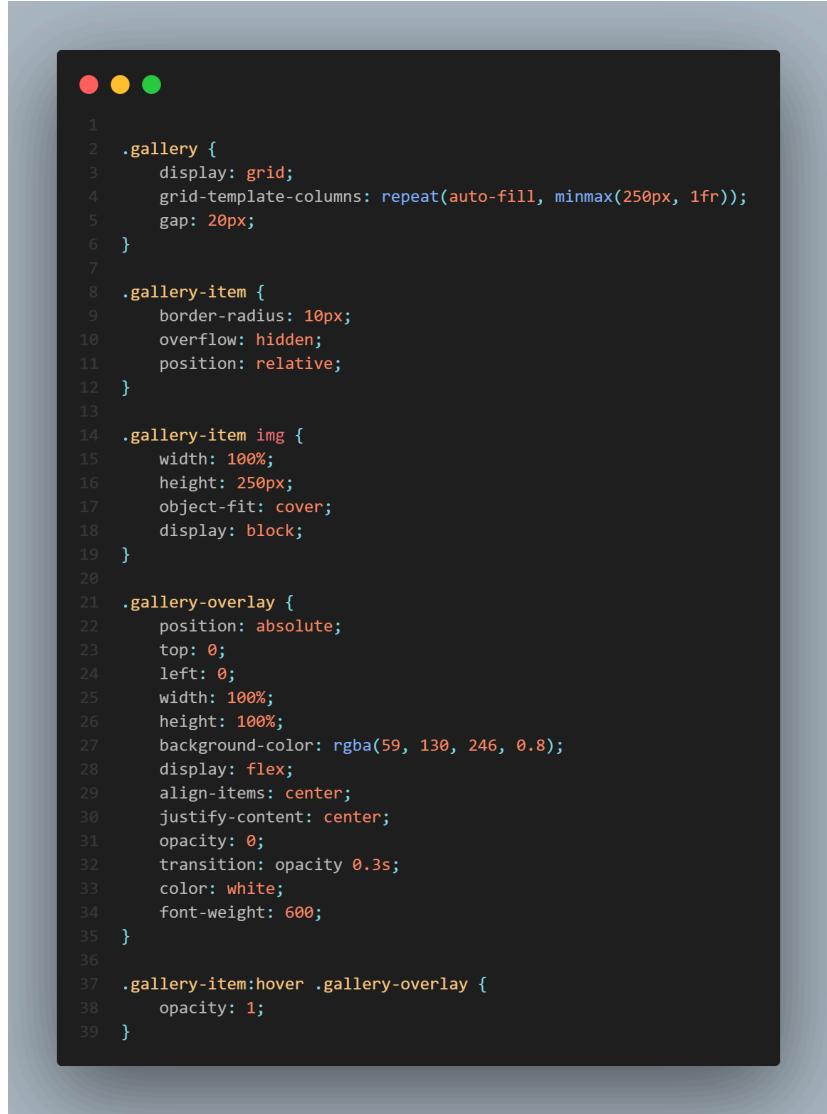
### 1.2.12. Process Step



```
1 .process-step {
2     text-align: center;
3     padding: 30px;
4     background-color: white;
5     border-radius: 10px;
6     box-shadow: 0 5px 15px rgba(0, 0, 0, 0.05);
7 }
8
9 .step-number {
10    width: 50px;
11    height: 50px;
12    background-color: var(--primary);
13    color: white;
14    border-radius: 50%;
15    display: flex;
16    align-items: center;
17    justify-content: center;
18    font-weight: 700;
19    font-size: 20px;
20    margin: 0 auto 20px;
21 }
22
```

**Penjelasan:** Steps dengan numbered circles, menggunakan flexbox untuk centering angka.

### 1.2.13. Gallery

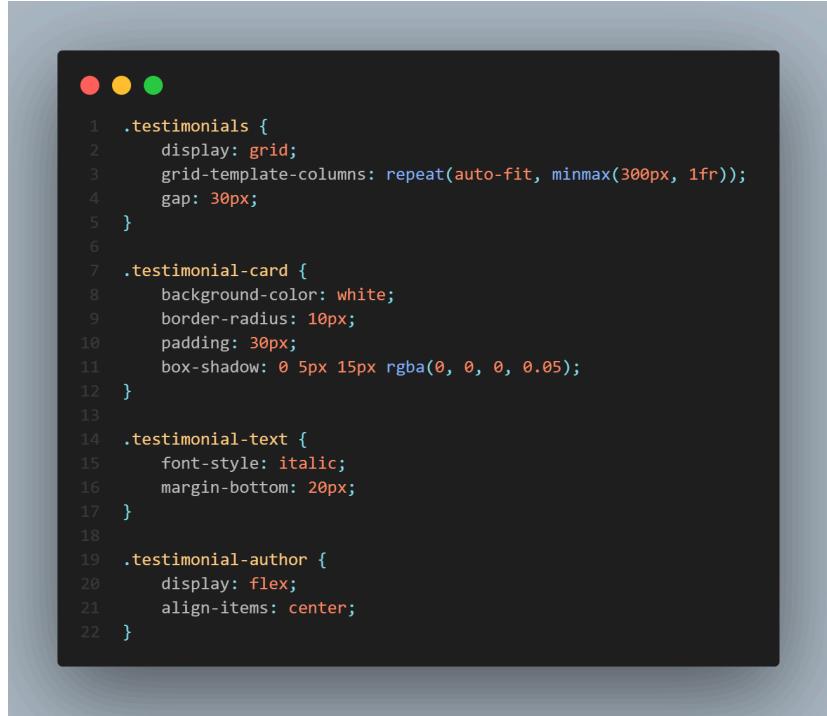


```
1 .gallery {
2   display: grid;
3   grid-template-columns: repeat(auto-fill, minmax(250px, 1fr));
4   gap: 20px;
5 }
6
7 .gallery-item {
8   border-radius: 10px;
9   overflow: hidden;
10  position: relative;
11 }
12
13 .gallery-item img {
14   width: 100%;
15   height: 250px;
16   object-fit: cover;
17   display: block;
18 }
19
20 .gallery-overlay {
21   position: absolute;
22   top: 0;
23   left: 0;
24   width: 100%;
25   height: 100%;
26   background-color: rgba(59, 130, 246, 0.8);
27   display: flex;
28   align-items: center;
29   justify-content: center;
30   opacity: 0;
31   transition: opacity 0.3s;
32   color: white;
33   font-weight: 600;
34 }
35
36
37 .gallery-item:hover .gallery-overlay {
38   opacity: 1;
39 }
```

### Penjelasan:

- Grid gallery dengan auto-fill untuk responsivitas
- Overlay yang muncul saat hover dengan opacity transition
- object-fit: cover untuk menjaga aspect ratio gambar

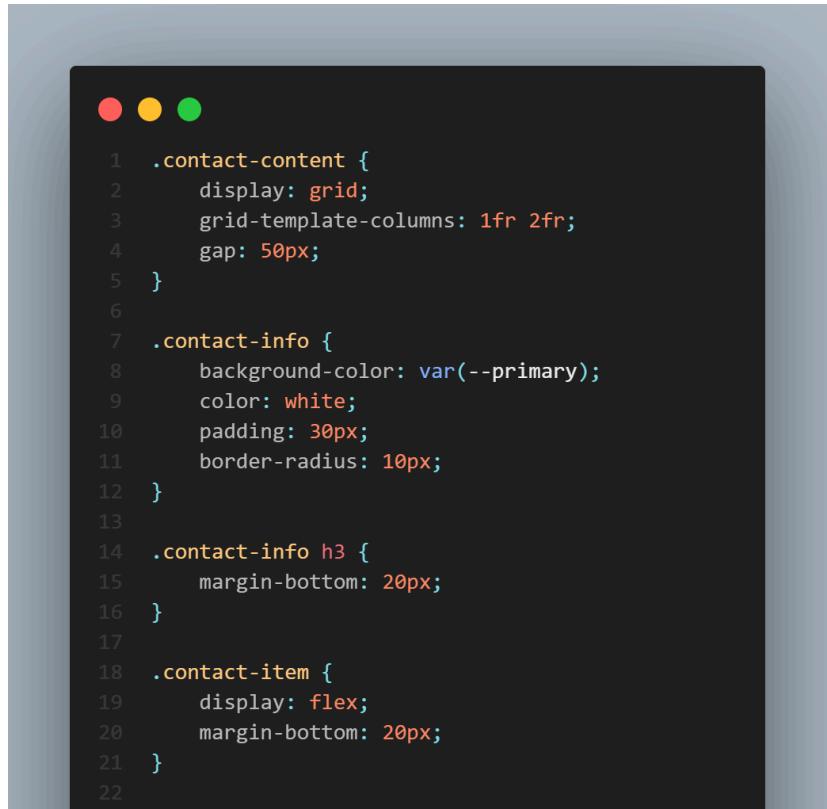
### 1.2.14. Testimonials



```
1 .testimonials {
2   display: grid;
3   grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
4   gap: 30px;
5 }
6
7 .testimonial-card {
8   background-color: white;
9   border-radius: 10px;
10  padding: 30px;
11  box-shadow: 0 5px 15px rgba(0, 0, 0, 0.05);
12 }
13
14 .testimonial-text {
15   font-style: italic;
16   margin-bottom: 20px;
17 }
18
19 .testimonial-author {
20   display: flex;
21   align-items: center;
22 }
```

**Penjelasan:** Card testimonial dengan avatar dan informasi author menggunakan flexbox.

### 1.2.15. Contact Station



```
1 .contact-content {
2   display: grid;
3   grid-template-columns: 1fr 2fr;
4   gap: 50px;
5 }
6
7 .contact-info {
8   background-color: var(--primary);
9   color: white;
10  padding: 30px;
11  border-radius: 10px;
12 }
13
14 .contact-info h3 {
15   margin-bottom: 20px;
16 }
17
18 .contact-item {
19   display: flex;
20   margin-bottom: 20px;
21 }
22 }
```

```
23 .contact-icon {  
24     margin-right: 15px;  
25     font-size: 20px;  
26 }  
27  
28 .contact-form {  
29     background-color: white;  
30     padding: 30px;  
31     border-radius: 10px;  
32     box-shadow: 0 5px 15px rgba(0, 0, 0, 0.05);  
33 }
```

**Penjelasan:** Layout 1:2 ratio untuk info kontak dan form, kontras warna untuk visual hierarchy.

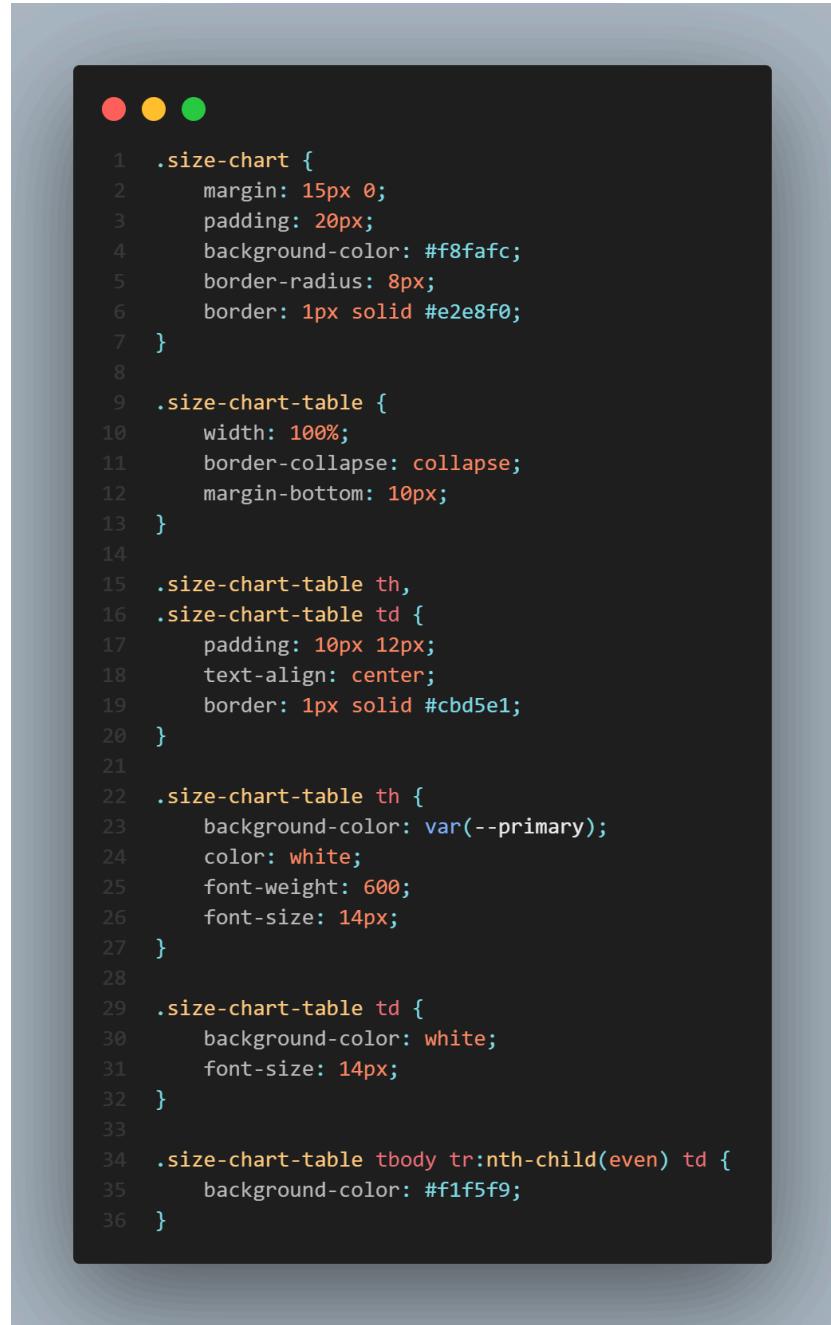
#### 1.2.16. Form Styles

```
1 .form-group {  
2     margin-bottom: 20px;  
3 }  
4  
5 .form-group label {  
6     display: block;  
7     margin-bottom: 8px;  
8     font-weight: 500;  
9 }  
10  
11 .form-control {  
12     width: 100%;  
13     padding: 12px 15px;  
14     border: 1px solid #e5e7eb;  
15     border-radius: 5px;  
16     font-size: 16px;  
17 }  
18  
19 textarea.form-control {  
20     height: 150px;  
21     resize: none;  
22 }
```

**Penjelasan:** Consistent form styling dengan proper spacing dan sizing.

### 1.2.17. Order Form Special Features

#### - SIZE CHART

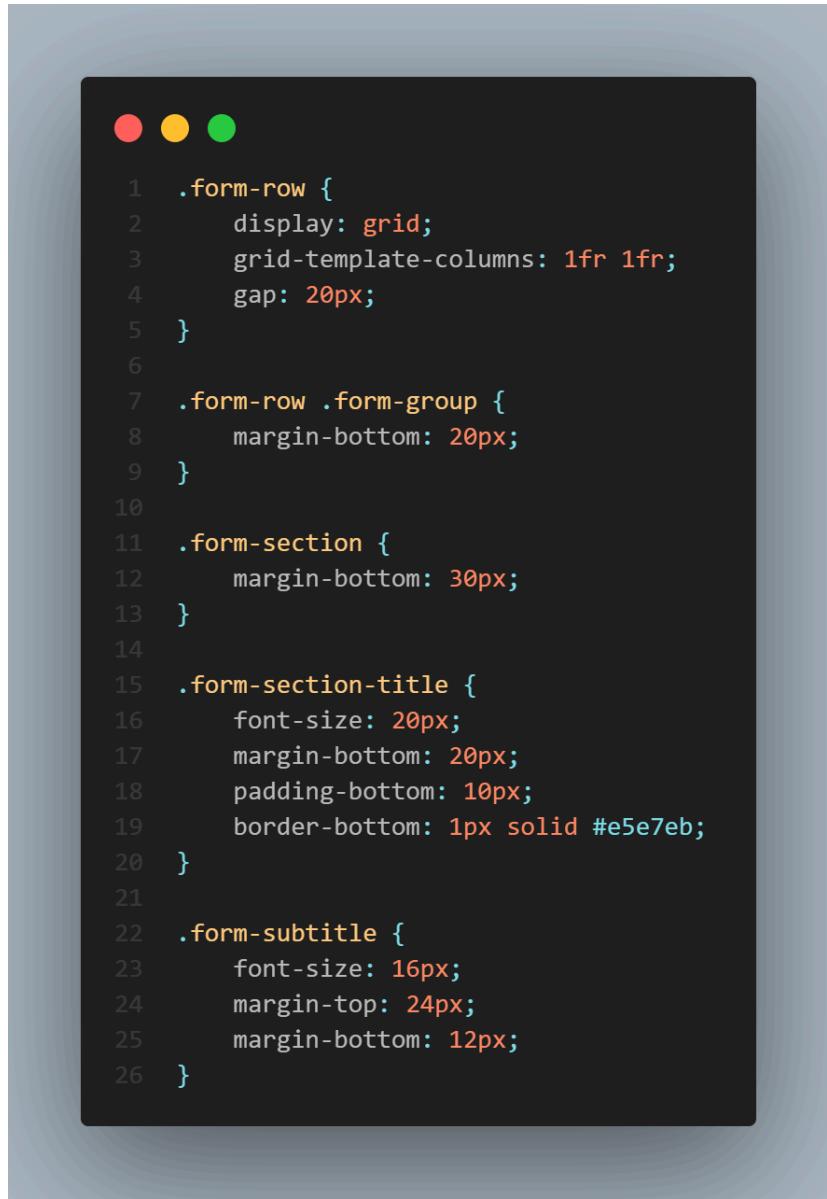


A screenshot of a code editor displaying CSS code for a size chart table. The code is organized into several classes and their properties. The classes include .size-chart, .size-chart-table, .size-chart-table th, .size-chart-table td, and .size-chart-table tbody tr:nth-child(even) td. The properties define styles such as margin, padding, background-color, border-radius, border, width, border-collapse, margin-bottom, padding, text-align, border, background-color, color, font-weight, font-size, and background-color for even rows.

```
1 .size-chart {  
2     margin: 15px 0;  
3     padding: 20px;  
4     background-color: #f8fafc;  
5     border-radius: 8px;  
6     border: 1px solid #e2e8f0;  
7 }  
8  
9 .size-chart-table {  
10    width: 100%;  
11    border-collapse: collapse;  
12    margin-bottom: 10px;  
13 }  
14  
15 .size-chart-table th,  
16 .size-chart-table td {  
17    padding: 10px 12px;  
18    text-align: center;  
19    border: 1px solid #cbd5e1;  
20 }  
21  
22 .size-chart-table th {  
23    background-color: var(--primary);  
24    color: white;  
25    font-weight: 600;  
26    font-size: 14px;  
27 }  
28  
29 .size-chart-table td {  
30    background-color: white;  
31    font-size: 14px;  
32 }  
33  
34 .size-chart-table tbody tr:nth-child(even) td {  
35    background-color: #f1f5f9;  
36 }
```

**Penjelasan:** Tabel size chart dengan striped rows dan header berwarna.

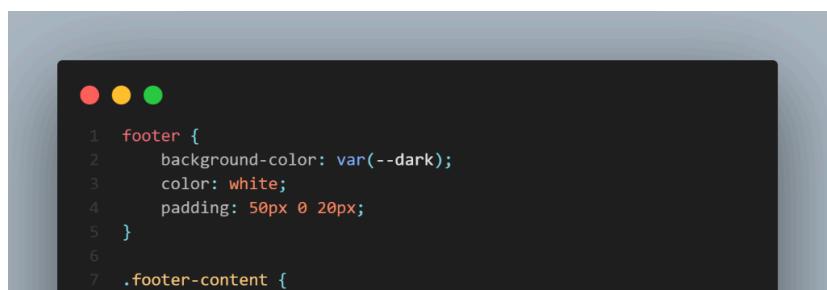
## - FORM LAYOUT



```
1 .form-row {  
2   display: grid;  
3   grid-template-columns: 1fr 1fr;  
4   gap: 20px;  
5 }  
6  
7 .form-row .form-group {  
8   margin-bottom: 20px;  
9 }  
10  
11 .form-section {  
12   margin-bottom: 30px;  
13 }  
14  
15 .form-section-title {  
16   font-size: 20px;  
17   margin-bottom: 20px;  
18   padding-bottom: 10px;  
19   border-bottom: 1px solid #e5e7eb;  
20 }  
21  
22 .form-subtitle {  
23   font-size: 16px;  
24   margin-top: 24px;  
25   margin-bottom: 12px;  
26 }
```

**Penjelasan:** Organisasi form dengan sections dan 2-column layout untuk fields.

### 1.2.18. Footer



```
1 footer {  
2   background-color: var(--dark);  
3   color: white;  
4   padding: 50px 0 20px;  
5 }  
6  
7 .footer-content {
```

```
8      display: grid;
9      grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
10     gap: 30px;
11     margin-bottom: 40px;
12   }
13
14 .footer-column h3 {
15   margin-bottom: 20px;
16   font-size: 18px;
17 }
18
19 .footer-links {
20   list-style: none;
21 }
22
23 .footer-links li {
24   margin-bottom: 10px;
25 }
26
27 .footer-links a {
28   color: var(--gray);
29   text-decoration: none;
30   transition: color 0.3s;
31 }
32
33 .footer-links a:hover {
34   color: white;
35 }
36
37 .footer-bottom {
38   text-align: center;
39   padding-top: 20px;
40   border-top: 1px solid rgba(255, 255, 255, 0.1);
41   color: var(--gray);
42   font-size: 14px;
43 }
44
45 .social-links {
46   display: flex;
47   gap: 15px;
48 }
49
50 .social-link {
51   width: 40px;
52   height: 40px;
53   background-color: rgba(255, 255, 255, 0.1);
54   border-radius: 50%;
55   display: flex;
56   align-items: center;
57   justify-content: center;
58   color: white;
59   text-decoration: none;
60   transition: background-color 0.3s;
61 }
```

**Penjelasan:** Footer dengan grid layout, social media icons dengan hover effects.

### 1.2.19. Mobile Navigation



**Penjelasan:** Hamburger menu button, hidden by default, akan ditampilkan di mobile.

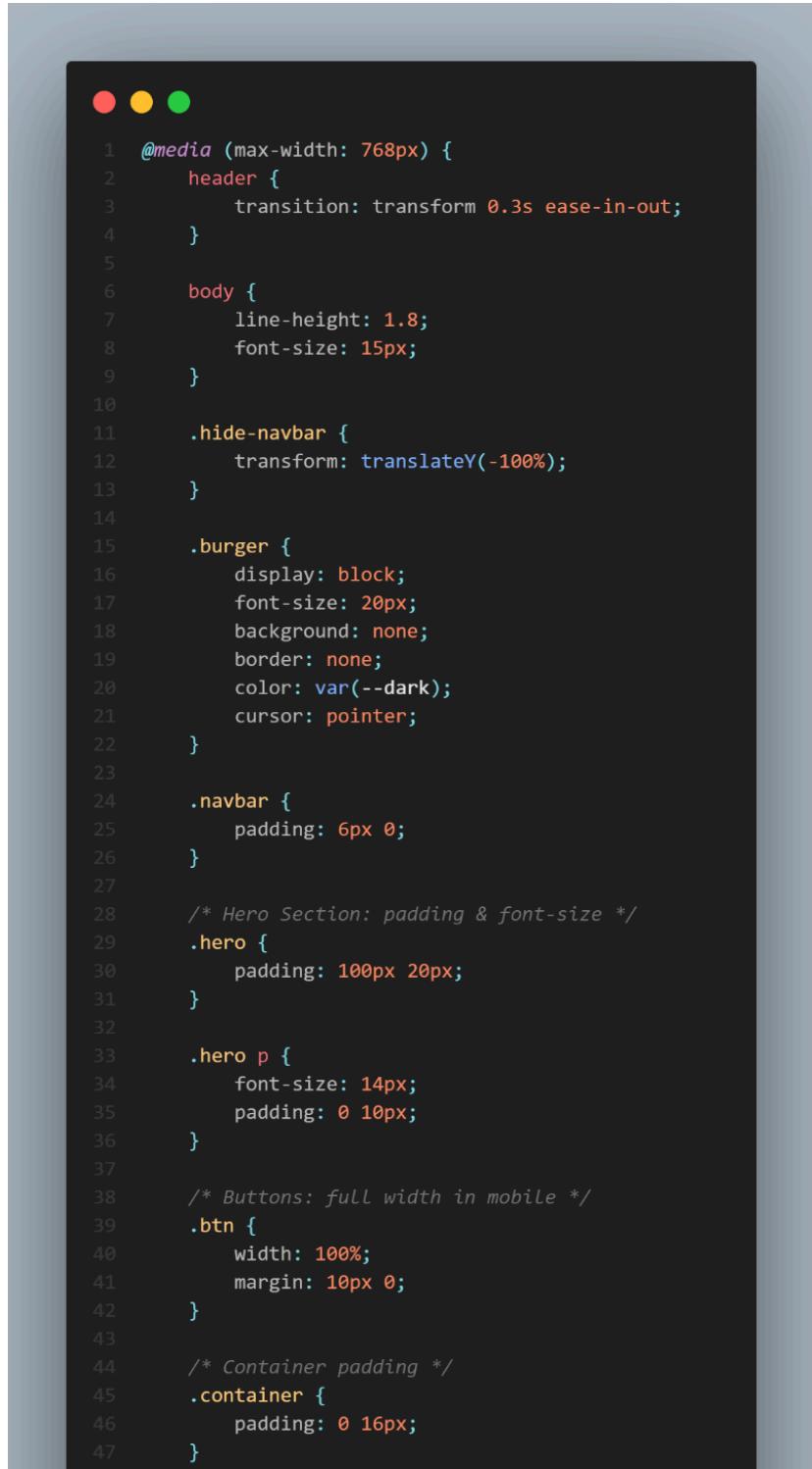
### 1.2.20. Animation



**Penjelasan:** Animasi fade-in dengan slide up effect untuk thank you screen.

### 1.2.21. Responsive Design (Media Queries)

#### - TABLET/MOBILE BREAKPOINT (768px)



```
1  @media (max-width: 768px) {
2      header {
3          transition: transform 0.3s ease-in-out;
4      }
5
6      body {
7          line-height: 1.8;
8          font-size: 15px;
9      }
10
11     .hide-navbar {
12         transform: translateY(-100%);
13     }
14
15     .burger {
16         display: block;
17         font-size: 20px;
18         background: none;
19         border: none;
20         color: var(--dark);
21         cursor: pointer;
22     }
23
24     .navbar {
25         padding: 6px 0;
26     }
27
28     /* Hero Section: padding & font-size */
29     .hero {
30         padding: 100px 20px;
31     }
32
33     .hero p {
34         font-size: 14px;
35         padding: 0 10px;
36     }
37
38     /* Buttons: full width in mobile */
39     .btn {
40         width: 100%;
41         margin: 10px 0;
42     }
43
44     /* Container padding */
45     .container {
46         padding: 0 16px;
47     }
```

```
48      .nav-links {
49        display: none;
50        flex-direction: column;
51        align-items: center;
52        width: 100%;
53        background-color: white;
54        position: absolute;
55        top: 100%;
56        left: 0;
57        box-shadow: 0 8px 16px rgba(0, 0, 0, 0.1);
58        padding: 10px 0;
59        z-index: 99;
60      }
61
62      .nav-links.show {
63        display: flex;
64      }
```

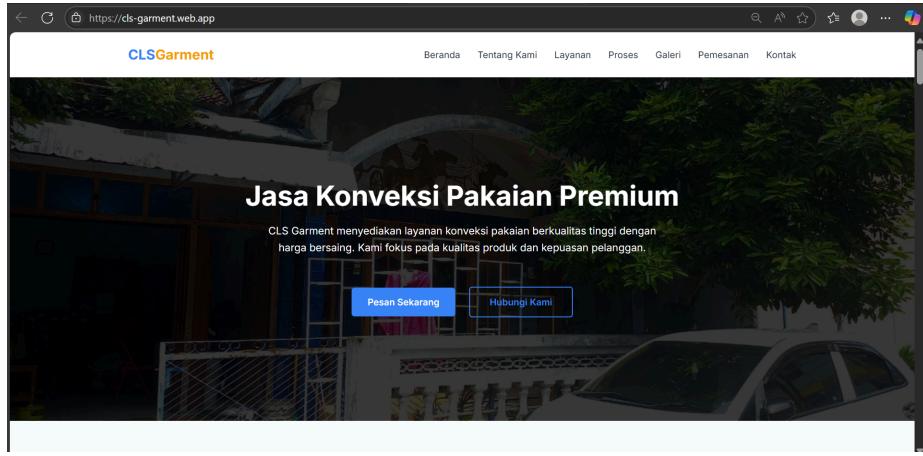
### Penjelasan Responsive Features:

- Navigation berubah menjadi mobile menu (hamburger)
- Grid layouts menjadi single column
- Font sizes diperkecil
- Padding dan margins disesuaikan
- Button menjadi full width
- Image heights disesuaikan
- Footer menjadi single column

## 2. Output

Berikut adalah output dari source code program yang telah selesai dengan skala website 80%. Web CLS Garment telah mengimplementasikan responsive layout dengan baik.

### 2.1. Beranda



Halaman pertama adalah beranda dimana menampilkan judul utama dari website dan tulisan sekunder. Lalu juga ada navigasi shortcut untuk menuju form pemesanan atau contact person.

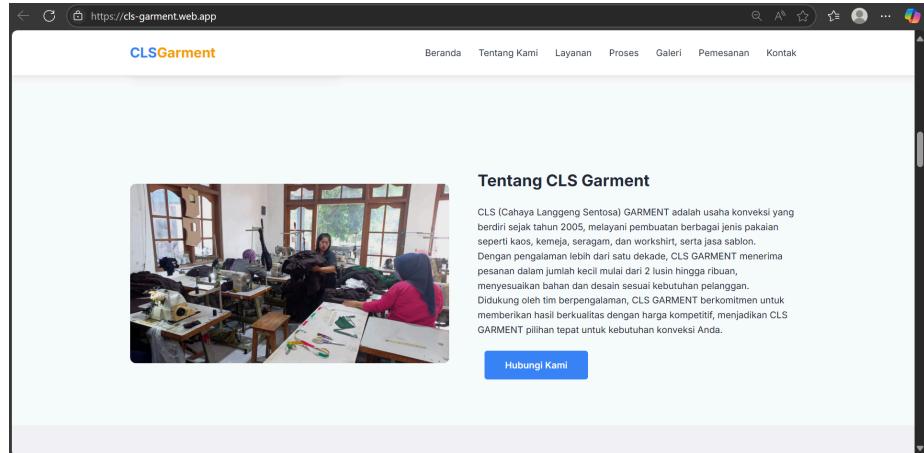
## 2.2. Proses Layanan

A screenshot of the 'Layanan Kami' (Our Services) section of the website. The title 'Layanan Kami' is at the top, followed by a subtext about providing various services to meet personal and business needs. Three service categories are shown in boxes: 'T-shirt', 'Shirt', and 'Uniform'. Each box contains a checkmark icon and a brief description of the service.

A screenshot of the 'Work Shirt' service details section. It shows a box with a checkmark icon and a detailed description of the service, mentioning various models like crew neck, raglan, and polo shirts, and how they can be customized.

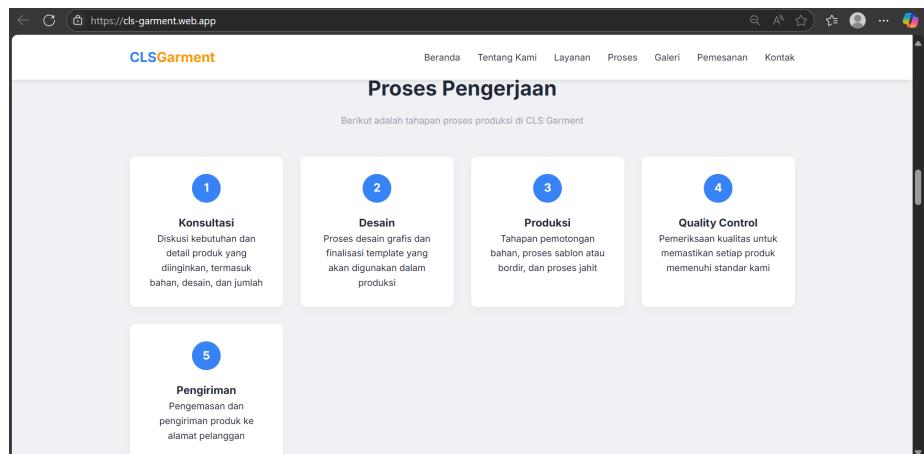
Jika di scroll kebawah akan muncul layanan yang dapat di handle oleh CLS Garment yaitu pembuatan kaos, kemeja, seragam dan work shirt.

### 2.3. Tentang Kami



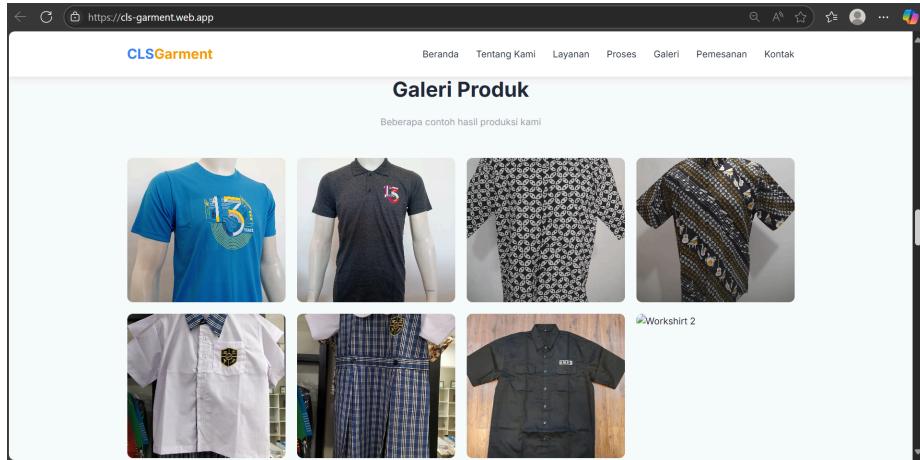
Pada halaman tentang kami akan menampilkan profil singkat dari CLS Garment dan ada navigasi untuk menuju contact person.

### 2.4. Proses Pengerjaan



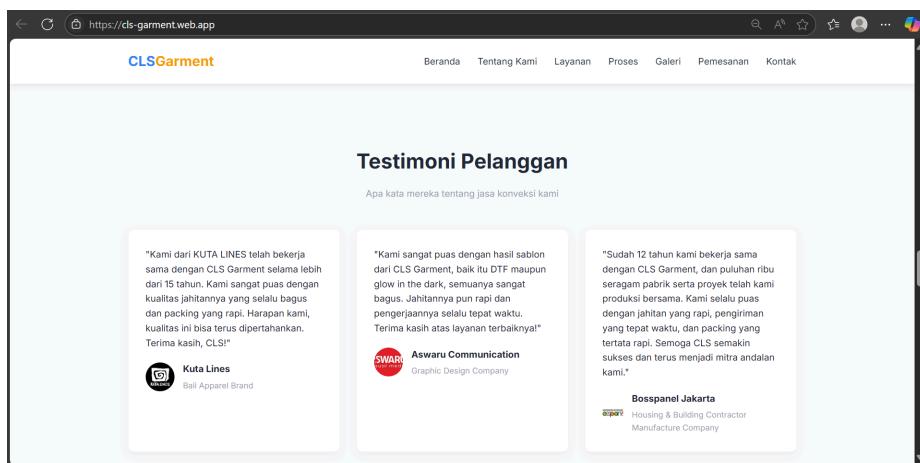
Selanjutnya, adalah proses pengerjaan pakaian yang dipesan. Yaitu dimulai dari konfirmasi pesanan, desain, produksi pakaian, quality control, dan yang terakhir adalah pengiriman ke konsumen.

### 2.5. Galeri Produk



Website juga menampilkan galeri produk yang isinya adalah pakaian yang pernah dibuat oleh CLS Garment. Diantaranya ada kaos custom, seragam, kemeja, dan work shirt.

## 2.6. Testimoni Pelanggan



Website juga menampilkan testimoni dari pelanggan terdahulu. Website kami menampilkan top 3 testimoni dari beberapa perusahaan di Indonesia yang membuktikan bahwa CLS Garment telah dipercaya oleh mereka.

## 2.7. Formulir Pemesanan

<https://cls-garment.web.app>

**CLSGarment**

Beranda Tentang Kami Layanan Proses Galeri Pemesanan Kontak

## Formulir Pemesanan

Isi formulir di bawah ini untuk melakukan pemesanan

**Informasi Pribadi**

Nama Lengkap	Email
[REDACTED]	[REDACTED]
Nomor Telepon	Alamat
[REDACTED]	[REDACTED]

**Detail Pesanan**

Jenis Produk	Jumlah
Pilih Jenis Produk	[REDACTED]
Minimal order 24 pcs	

<https://cls-garment.web.app>

**CLSGarment**

Beranda Tentang Kami Layanan Proses Galeri Pemesanan Kontak

Size Chart

Size	Lebar Dada (cm)	Panjang Badan (cm)
XXS	42	60
XS	45	63
S	48	66
M	51	69
L	54	72
XL	57	75
XXL	59	78

\*Toleransi ukuran 1-1,5 cm

Ukuran (Jumlah per ukuran)

XXS	XS
[REDACTED]	[REDACTED]

<https://cls-garment.web.app>

**CLSGarment**

Beranda Tentang Kami Layanan Proses Galeri Pemesanan Kontak

[REDACTED]	[REDACTED]
L	XL
[REDACTED]	[REDACTED]
XXL	
[REDACTED]	

**Metode Produksi**

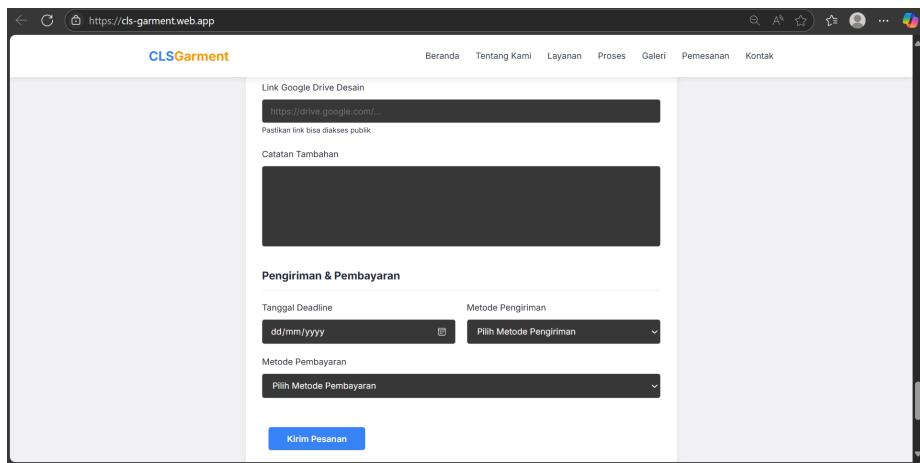
Jenis Sablon / Bordir

- Sablon Plastisol    Sablon Rubber    Sablon Discharge    Bordir    DTF (Direct to Film)

Warna Kain

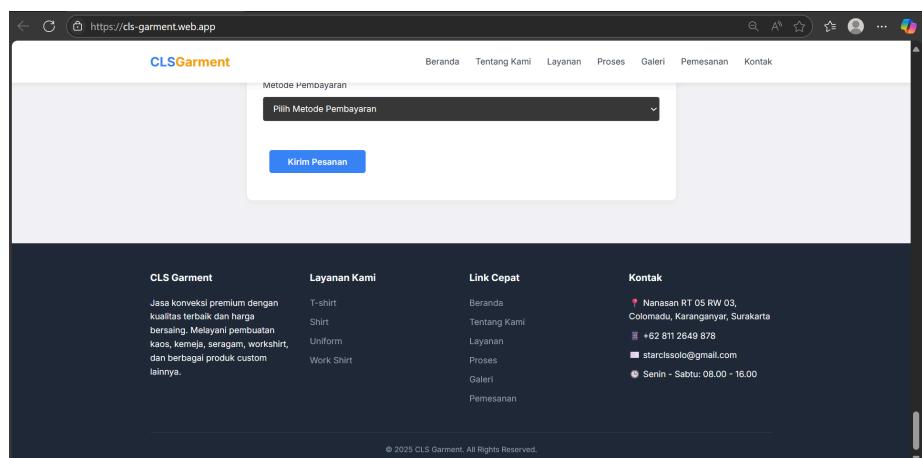
Jenis Kain

Pilih Jenis Kain



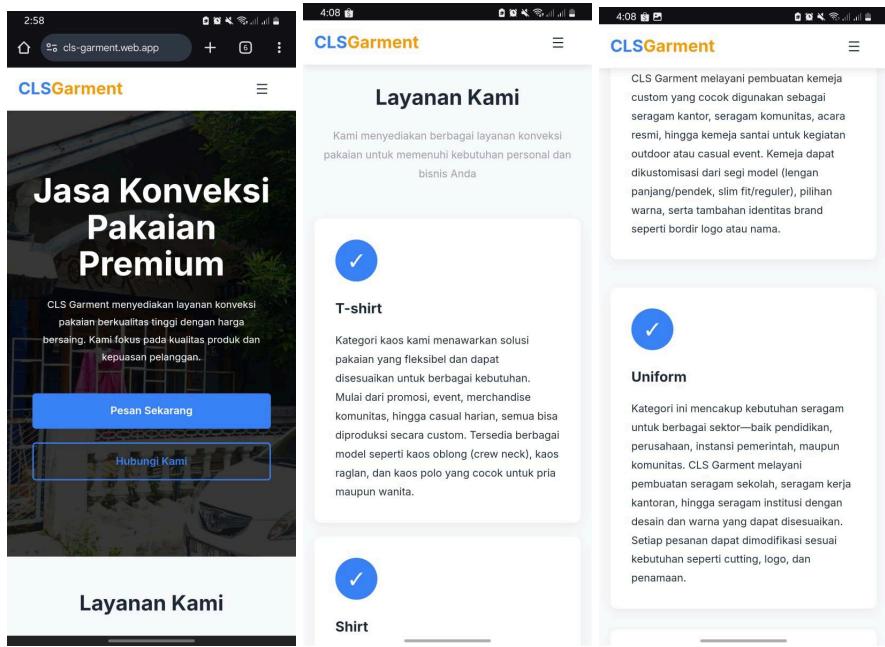
Fitur utama dari website ini adalah fitur pemesanan, yaitu terdiri dari pengisian nama, alamat, hingga detail pesanan. Semua fill field mencakup informasi dari pesanan pelanggan sehingga mudah dibaca oleh admin dan pelanggan.

## 2.8. Footer Contact Person



Bagian footer menampilkan contact person lengkap dari CLS Garment.

## 2.9. Tampilan Mobile



Website juga telah responsive terhadap device yang dipakai pengguna sehingga tampilannya tetap nyaman untuk dilihat dan digunakan.

#### D. PENGUJIAN

No.	Nama Fitur	Skenario Uji	Input	Hasil yang Diharapkan	Status
1.	Klik link website	Membuka website dan menampilkan beranda	-	Launching web page di layar customer sesuai device yang digunakan	Sukses
2.	Scroll	Menggeser aplikasi secara vertikal	-	Web dapat digeser secara seamless	Sukses
3.	Navigasi halaman	Pengguna klik semua navigasi	Klik tiap navigasi	Navigasi berfungsi dan dapat berpindah halaman	Sukses

4.	Galeri produk	Pengguna klik navigasi galeri	Klik galeri	Menuju halaman galeri dan saat kursor diarahkan ke foto akan muncul tulisan tipe atau model baju	Sukses
5.	Formulir Pemesanan	Pengguna mengisi formulir pemesanan	Mengisi semua fill field yang ada pada formulir. Terdapat juga dropdown yang berisi jenis produk, jumlah, jenis kain, metode pengiriman, dan metode pembayaran.	Data tersimpan dan muncul pesan konfirmasi saat di klik kirim pesanan	Sukses
6.	Notifikasi wajib isi sebelum submit pesanan	Pengguna langsung klik kirim pesanan tanpa mengisi formulir pesanan	Klik kirim pesanan	Muncul notifikasi “please fill out this field”	Sukses

7.	Notifikasi link harus dari gdrive	Pengguna mengisi link design dengan link selain google drive	Klik kirim pesanan	Muncul notifikasi “Link desain harus dari Google Drive.”	Sukses
8.	Kirim pesanan	Pengguna mengisi semua formulir dengan benar sesuai ketentuan	Klik kirim pesanan	Berpindah ke halaman konfirmasi	Sukses
9.	Kirim bukti pembayaran melalui whatsapp	Pengguna memilih opsi kirim bukti via whatsapp pada halaman konfirmasi	Klik “kirim bukti pembayaran”	Direct ke whatsapp admin	Sukses

## **BAB IV**

### **KESIMPULAN**

Berdasarkan proses perancangan dan pengembangan aplikasi *CLS GARMEN*T yang telah dilakukan, maka dapat disimpulkan hal-hal berikut:

1. Aplikasi *CLS GARMEN*T berhasil dikembangkan menggunakan bahasa pemrograman Kotlin dengan platform Android Studio, sesuai dengan kebutuhan konveksi Cahaya Langgeng Sentosa dalam menyajikan informasi produk dan layanan secara digital.
2. Aplikasi ini menyediakan fitur-fitur utama yang mendukung proses bisnis CLS GARMEN, seperti tampilan katalog produk, informasi detail produk, formulir pemesanan, navigasi antar halaman, profil pengguna, hingga notifikasi pemesanan.
3. Hasil pengujian menunjukkan bahwa seluruh fitur utama dapat berjalan dengan baik sesuai dengan skenario yang telah dirancang. Aplikasi memiliki antarmuka yang user-friendly dan dapat diakses secara offline untuk data statis (katalog).
4. Pengembangan aplikasi ini tidak hanya mendukung proses digitalisasi usaha konveksi, tetapi juga memberikan nilai tambah dari sisi kemudahan pelanggan dalam mengakses informasi serta melakukan pemesanan secara praktis melalui perangkat mobile.