

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Hasil Pengujian Metode

Setelah dijabarkan mengenai contoh perhitungan menggunakan sampel data *smartphone* pada Bab III, maka dalam implementasi ini akan menggunakan keseluruhan data *smartphone* yang rilis di Indonesia dari Januari 2023 hingga Januari 2024 dengan data yang diperoleh dari GSMArena serta *Official Online Store*.

##### 4.1.1 Mempersiapkan Data *Smartphone*

Sumber data *smartphone* diperoleh dari GSMArena serta dari *website* resmi produk tersebut. Dalam hal ini karena rentang waktu yang ditetapkan dalam data yang digunakan dari Januari 2023 hingga Januari 2024, maka data yang digunakan merupakan data *smartphone* yang masih dijual oleh *official store* merek tersebut. Total data yang diperoleh setelah melakukan pendataan berjumlah 103 data *smartphone* dari *brand* Asus, Samsung, Apple, Xiaomi, Oppo, Vivo, Realme, Infinix, dan Redmi.

Tabel 4.1 Data *Smartphone*

<i>Smartphone</i>				<i>Body</i>		...	Price
No	Brand	Merk	Dimension	Weight	build	...	
1	ASUS	Zenfone 10	146.5 x 68.1 x 9.4 mm	172 g	aluminum frame, plastic back	...	Rp 8.999.000,00
2	ASUS	Zenfone 10	146.5 x 68.1 x 9.4 mm	172 g	aluminum frame, plastic back	...	Rp 11.999.000,00
3	ASUS	ROG Phone 7 Ultimate	173 x 77 x 10.3 mm	239 g	aluminum frame, Glass back	...	Rp 23.499.000,00
4	ASUS	ROG Phone 7	173 x 77 x 10.3 mm	239 g	aluminum frame, Glass back	...	Rp 13.499.000,00
5	ASUS	ROG Phone 7	173 x 77 x 10.3 mm	239 g	aluminum frame, Glass back	...	Rp 26.990.000,00
6	APPL E	Iphone 15 Pro Max	159.9 x 76.7 x 8.3 mm	221 g	titanium frame, Glass back	...	Rp 22.999.000,00

<i>Smartphone</i>			<i>Body</i>		...		
No	Brand	Merk	Dimension	Weight	build	...	Price
7	APPL E	Iphone 15 Pro Max	159.9 x 76.7 x 8.3 mm	221 g	titanium frame, Glass back	...	Rp 27.999.000,00
8	APPL E	Iphone 15 Pro Max	159.9 x 76.7 x 8.3 mm	221 g	titanium frame, Glass back	...	Rp 31.999.000,00
9	APPL E	Iphone 15 Pro	146.6 x 70.6 x 8.3 mm	187 g	titanium frame, Glass back	...	Rp 18.999.000,00
10	APPL E	Iphone 15 Pro	146.6 x 70.6 x 8.3 mm	187 g	titanium frame, Glass back	...	Rp 21.999.000,00
11	APPL E	Iphone 15 Pro	146.6 x 70.6 x 8.3 mm	187 g	titanium frame, Glass back	...	Rp 25.999.000,00
12	APPL E	Iphone 15 Pro	146.6 x 70.6 x 8.3 mm	187 g	titanium frame, Glass back	...	Rp 29.999.000,00
13	APPL E	Iphone 15 Plus	160.9 x 77.8 x 7.8 mm	201 g	aluminum frame, Glass back	...	Rp 15.999.000,00
...	...	...	...	...	...	...	...
10 3	INFINI X	INFINIX HOT 30i	164 x 75.8 x 8.4 mm	191 g	plastic frame, plastic back	...	Rp 1.660.000,00

#### 4.1.2 Proses Hitung Algoritma SMART

##### 1. Perhitungan Bobot

Tabel 4.2 Penentuan Bobot Kriteria

No	Kriteria	Bobot	Normalisasi	Bobot Akhir
1	Body	50	0,119047619	12%
2	Display	10	0,023809524	2%
3	System	95	0,226190476	23%
4	Memory	90	0,214285714	21%
5	Front Camera	0	0	0%
6	Main Camera	0	0	0%
7	Battery	80	0,19047619	19%
8	Price	95	0,226190476	23%
Total		420	1	100%

Dalam perhitungan bobot untuk menentukan nilai normalisasi dengan cara nilai bobot dibagi dengan total nilai bobot. Untuk menemukan bobot akhir dihitung dengan cara nilai normalisasi dikali 100%. Bobot

ditentukan oleh pengguna umum, sesuai preferensi mereka. Tabel diatas merupakan contoh penentuan bobot dari pengguna umum.

## 2. Konversi Data

Tabel 4.3 Konversi Data

<i>Smartphone</i>	<i>Dimensi</i>	<i>Berat</i>	<i>Build</i>	<i>Tipe LCD</i>	<i>Ukuran LCD</i>	<i>...</i>	<i>Harga</i>
ASUS Zenfone 10(8/128 gb)	40	20	90	70	10	...	26
ASUS Zenfone 10(16/512 gb)	40	20	90	70	10	...	36
ASUS ROG Phone 7 Ultimate(16/512 gb)	100	100	150	65	100	...	73
ASUS ROG Phone 7(12/256 gb)	100	100	150	65	100	...	41
ASUS ROG Phone 7(16/512 gb)	100	100	150	65	100	...	84
APPLE Iphone 15 Pro Max(8/256 gb)	40	80	170	100	100	...	71
APPLE Iphone 15 Pro Max(8/512 gb)	40	80	170	100	100	...	88
APPLE Iphone 15 Pro Max(8/1000 gb)	40	80	170	100	100	...	100
APPLE Iphone 15 Pro(8/128 gb)	20	40	170	100	10	...	58
APPLE Iphone 15 Pro(8/256 gb)	20	40	170	100	10	...	68
APPLE Iphone 15 Pro(8/512 gb)	20	40	170	100	10	...	81

<i>Smartphone</i>	<i>Dimensi</i>	<i>Berat</i>	<i>Build</i>	<i>Tipe LCD</i>	<i>Ukuran LCD</i>	<i>...</i>	<i>Harga</i>
APPLE Iphone 15 Pro(8/1000 gb)	20	40	170	100	10	...	94
APPLE Iphone 15 Plus(8/128 gb)	40	40	150	100	100	...	49
APPLE Iphone 15 Plus(8/256 gb)	40	40	150	100	100	...	58
APPLE Iphone 15 Plus(8/512 gb)	40	40	170	100	100	...	71
APPLE Iphone 15(8/128 gb)	20	20	150	100	10	...	49
APPLE Iphone 15(8/256 gb)	20	20	150	100	10	...	58
...	...	...	...	...	...	...	...
INFINIX INFINIX HOT 30i(8/128 gb)	60	40	20	30	100	...	2

Konversi data seperti yang telah dijelaskan di BAB III, akan di implementasikan pada konversi ini dengan nilai konversi sesuai dengan yang telah ditentukan.

### 3. Normalisasi Data

Tabel 4.4 Normalisasi Data

<i>Smartphone</i>	<i>Dimensi</i>	<i>Berat</i>	<i>Build</i>	<i>...</i>	<i>Harga</i>
ASUS Zenfone 10(8/128 gb)	0,25	0	0,466666666	...	0,7474747
ASUS Zenfone 10(16/512 gb)	0,25	0	0,466666666	...	0,6464646
ASUS ROG Phone 7 Ultimate(16/512 gb)	1	1	0,866666666	...	0,2727272
ASUS ROG Phone 7(12/256 gb)	1	1	0,866666666	...	0,5959596
ASUS ROG Phone 7(16/512 gb)	1	1	0,866666666	...	0,16161616
APPLE Iphone 15 Pro Max(8/256 gb)	0,25	0,75	1	...	0,2929292
APPLE Iphone 15 Pro Max(8/512 gb)	0,25	0,75	1	...	0,1212121

<i>Smartphone</i>	<i>Dimensi</i>	<i>Berat</i>	<i>Build</i>	<i>...</i>	<i>Harga</i>
APPLE Iphone 15 Pro Max(8/1000 gb)	0,25	0,75	1	...	0
APPLE Iphone 15 Pro(8/128 gb)	0	0,25	1	...	0,42424242
APPLE Iphone 15 Pro(8/256 gb)	0	0,25	1	...	0,32323232
APPLE Iphone 15 Pro(8/512 gb)	0	0,25	1	...	0,19191919
APPLE Iphone 15 Pro(8/1000 gb)	0	0,25	1	...	0,0606061
APPLE Iphone 15 Plus(8/128 gb)	0,25	0,25	0,86666666	...	0,5151515
APPLE Iphone 15 Plus(8/256 gb)	0,25	0,25	0,86666666	...	0,4242424
APPLE Iphone 15 Plus(8/512 gb)	0,25	0,25	1	...	0,2929292
APPLE Iphone 15(8/128 gb)	0	0	0,86666666	...	0,5151515
APPLE Iphone 15(8/256 gb)	0	0	0,86666666	...	0,4242424
APPLE Iphone 15(8/512 gb)	0	0	0,86666666	...	0,2929292
SAMSUNG Galaxy S24(8/512 gb)	0	0	1	...	0,515151
SAMSUNG Galaxy S24 Ultra(12/512 gb)	0,5	0,75	1	...	0,2525252
...	...	...	...	...	...
INFINIX INFINIX NOTE 30 PRO(8/256 gb)	0,25	0,5	0,4	...	0,9393939

Setelah memperoleh hasil konversi data, maka akan dilakukan normalisasi data, normalisasi data ini keseluruhan menggunakan *utility benefit* kecuali untuk harga menggunakan *utility const.*

#### 4. Hasil Akhir

Tabel 4.5 Hasil Akhir

<i>Smartphone</i>	<i>Dimensi</i>	<i>Berat</i>	<i>Build</i>	<i>...</i>	<i>Harga</i>
ASUS Zenfone 10(8/128 gb)	0,029761 905	0	0,055555 556	...	0,169071 669
ASUS Zenfone 10(16/512 gb)	0,029761 905	0	0,055555 556	...	0,146224 146
ASUS ROG Phone 7 Ultimate(16/512 gb)	0,119047 619	0,119047 619	0,103174 603	...	0,061688 312
ASUS ROG Phone 7(12/256 gb)	0,119047 619	0,119047 619	0,103174 603	...	0,134800 385
ASUS ROG Phone 7(16/512 gb)	0,119047 619	0,119047 619	0,103174 603	...	0,036556 037
APPLE Iphone 15 Pro Max(8/256 gb)	0,029761 905	0,089285 714	0,119047 619	...	0,066257 816
APPLE Iphone 15 Pro Max(8/512 gb)	0,029761 905	0,089285 714	0,119047 619	...	0,027417 027

<i>Smartphone</i>	<i>Dimensi</i>	<i>Berat</i>	<i>Build</i>	<i>...</i>	<i>Harga</i>
APPLE Iphone 15 Pro Max(8/1000 gb)	0,029761 905	0,089285 714	0,119047 619	...	0
APPLE Iphone 15 Pro(8/128 gb)	0	0,029761 905	0,119047 619	...	0,095959 596
APPLE Iphone 15 Pro(8/256 gb)	0	0,029761 905	0,119047 619	...	0,073112 073
APPLE Iphone 15 Pro(8/512 gb)	0	0,029761 905	0,119047 619	...	0,043410 293
APPLE Iphone 15 Pro(8/1000 gb)	0	0,029761 905	0,119047 619	...	0,013708 514
APPLE Iphone 15 Plus(8/128 gb)	0,029761 905	0,029761 905	0,103174 603	...	0,116522 367
APPLE Iphone 15 Plus(8/256 gb)	0,029761 905	0,029761 905	0,103174 603	...	0,095959 596
APPLE Iphone 15 Plus(8/512 gb)	0,029761 905	0,029761 905	0,119047 619	...	0,066257 816
...	...	...	...	...	...
INFINIX HOT 30i(8/128 gb)	0,059523 81	0,029761 905	0 524	0,023809 524	0,223905 724

Setelah data ternormalisasi dan ditentukan *utility*-nya maka akan dilakukan perhitungan hasil akhir dengan cara nilai bobot yang diberikan dikali dengan data normalisasi.

## 5. Perangkingan

Tabel 4 6 Perangkingan

<i>Smartphone</i>	<i>Body</i>	<i>LCD</i>	<i>System</i>	<i>Memory</i>	<i>Main Camera</i>	<i>...</i>	<i>Total</i>	<i>Rank</i>
ASUS Zenfone 10(8/128 gb)	0,0284	0,0065	0,1194	0,0357	0	...	0,4147	99
ASUS Zenfone 10(16/512 gb)	0,0284	0,0065	0,1194	0,1786	0	...	0,5347	36
ASUS ROG Phone 7 Ultimate(16/512 gb)	0,1138	0,0139	0,1194	0,1786	0	...	0,6248	10
ASUS ROG Phone 7(12/256 gb)	0,1138	0,0139	0,1194	0,1071	0	...	0,6265	9
ASUS ROG Phone 7(16/512 gb)	0,1138	0,0139	0,1194	0,1786	0	...	0,5997	12
APPLE Iphone 15 Pro Max(8/256 gb)	0,0794	0,0198	0,1232	0,0714	0	...	0,5030	56
APPLE Iphone 15 Pro Max(8/512 gb)	0,0794	0,0198	0,1232	0,1071	0	...	0,4999	62
APPLE Iphone 15 Pro Max(8/1000 gb)	0,0794	0,0198	0,1232	0,1429	0	...	0,5082	53
APPLE Iphone 15 Pro(8/128 gb)	0,0496	0,0119	0,1232	0,0357	0	...	0,4117	100
APPLE Iphone 15 Pro(8/256 gb)	0,0496	0,0119	0,1232	0,0714	0	...	0,4245	96
APPLE Iphone 15 Pro(8/512 gb)	0,0496	0,0119	0,1232	0,1071	0	...	0,4305	92
APPLE Iphone 15 Pro(8/1000 gb)	0,0496	0,0119	0,1232	0,1429	0	...	0,4366	90

<i>Smartphone</i>	<i>Body</i>	<i>LCD</i>	<i>System</i>	<i>Memory</i>	<i>Main Camera</i>	<i>...</i>	<i>Total</i>	<i>Rank</i>
APPLE Iphone 15 Plus(8/128 gb)	0,0542	0,0198	0,1232	0,0357	0	...	0,4289	93
APPLE Iphone 15 Plus(8/256 gb)	0,0542	0,0198	0,1232	0,0714	0	...	0,4441	88
APPLE Iphone 15 Plus(8/512 gb)	0,0595	0,0198	0,1232	0,1071	0	...	0,4554	86
APPLE Iphone 15(8/128 gb)	0,0344	0,0119	0,1232	0,0357	0	...	0,3535	103
APPLE Iphone 15(8/256 gb)	0,0344	0,0119	0,1232	0,0714	0	...	0,3687	102
APPLE Iphone 15(8/512 gb)	0,0344	0,0119	0,1232	0,1071	0	...	0,3747	101
SAMSUNG Galaxy S24(8/512 gb)	0,0397	0,0082	0,1629	0,1071	0	...	0,5535	26
SAMSUNG Galaxy S24 Ultra(12/512 gb)	0,0893	0,0201	0,1629	0,1429	0	...	0,6389	6
SAMSUNG Galaxy S24 Ultra(12/1000 gb)	0,0893	0,0201	0,1629	0,1786	0	...	0,6449	5
SAMSUNG Galaxy S24+(Online Exclusive)(12/512 gb)	0,0542	0,0201	0,2006	0,1429	0	...	0,6566	4
SAMSUNG Galaxy S24 Ultra(12/256 gb)	0,0893	0,0201	0,1629	0,1071	0	...	0,6238	11
SAMSUNG Galaxy S23 FE(8/256 gb)	0,0642	0,0156	0,1281	0,0714	0	...	0,5526	27
SAMSUNG Galaxy S23 FE(8/128 gb)	0,0642	0,0156	0,1281	0,0357	0	...	0,5237	39
SAMSUNG Galaxy S23 Ultra(12/512 gb)	0,0840	0,0196	0,1194	0,1429	0	...	0,5875	13
SAMSUNG Galaxy S23+(8/256 gb)	0,0542	0,0156	0,1194	0,0714	0	...	0,4883	70
...	...	...	...	...	...	...	...	...
INFINIX INFINIX HOT 30i(8/128 gb)	0,0298	0,0079	0,0643	0,0357	0	...	0,4648	79

Perhitungan pada perangkingan ini dijumlahkan setiap subkriteria yang telah di tentukan dan dibagi dengan jumlah subkriteria tiap-tiap kriteria yang ada. Setelah mendapatkan nilai setiap kriteria maka akan ditotal untuk setiap data *smartphone*.

## 4.2 Implementasi *Code*

Pada implementasi ini akan menjelaskan tahapan perhitungan algoritma SMART yang telah diimplementasikan pada sistem yang telah dibuat. Dimulai dari perhitungan bobot kriteria, konversi data, normalisasi data, hingga hasil akhir dan perangkingan.

#### 4.2.1 Perhitungan Bobot Kriteria

Tabel 4. 7 Pseudocode Bobot Kriteria

```

fungsi simpanBobot() {
    // Ambil data dari pengguna
    bobotBadan = ambilPost('body')      // Ambil bobot untuk atribut body
    bobotLayar = ambilPost('display')    // Ambil bobot untuk atribut display
    bobotSistem = ambilPost('system')    // Ambil bobot untuk atribut system
    bobotMemori = ambilPost('memory')    // Ambil bobot untuk atribut memory
    bobotKameraUtama = ambilPost('mainCamera') // Ambil bobot untuk atribut main camera
    bobotKameraDepan = ambilPost('frontCamera') // Ambil bobot untuk atribut front camera
    bobotBaterai = ambilPost('battery')  // Ambil bobot untuk atribut battery
    bobotHarga = ambilPost('price')      // Ambil bobot untuk atribut price
    hargaMin = ambilPost('min')          // Ambil filter harga minimum
    hargaMax = ambilPost('max')          // Ambil filter harga maksimum

    // Periksa jika total bobot setidaknya 1
    totalBobot = bobotBadan + bobotLayar + bobotSistem + bobotMemori +
    bobotKameraUtama + bobotKameraDepan + bobotBaterai + bobotHarga
    jika totalBobot < 1 {
        tampilkanError('Total bobot harus setidaknya 1') // Tampilkan error jika total bobot kurang
        dari 1
        kembali
    }

    // Tentukan rentang harga
    jika hargaMin kosong dan hargaMax tidak kosong {
        nilaiHargaMin = 0 // Setel harga minimum ke 0
        nilaiHargaMax = hapusKarakterNonAngka(hargaMax) // Bersihkan harga maksimum
        hanya dengan angka
        aktifkanFilter() // Aktifkan filter
    } jika tidak, jika hargaMin tidak kosong dan hargaMax tidak kosong {
        nilaiHargaMin = hapusKarakterNonAngka(hargaMin) // Bersihkan harga minimum hanya
        dengan angka
        nilaiHargaMax = hapusKarakterNonAngka(hargaMax) // Bersihkan harga maksimum
        hanya dengan angka
        aktifkanFilter() // Aktifkan filter
    } jika tidak, jika hargaMin tidak kosong dan hargaMax kosong {
        nilaiHargaMax = ambilHargaMaxDariDB() // Ambil harga maksimum dari database
        nilaiHargaMin = hapusKarakterNonAngka(hargaMin) // Bersihkan harga minimum hanya
        dengan angka
        aktifkanFilter() // Aktifkan filter
    } jika tidak, jika hargaMin kosong dan hargaMax tidak kosong {
        nilaiHargaMin = ambilHargaMinDariDB() // Ambil harga minimum dari database
        nilaiHargaMax = hapusKarakterNonAngka(hargaMax) // Bersihkan harga maksimum
        hanya dengan angka
        aktifkanFilter() // Aktifkan filter
    } jika tidak, jika hargaMin >= 0 dan hargaMax >= 1 dan hargaMin >= hargaMax {
        tampilkanError('Harga minimum tidak boleh lebih besar dari harga maksimum') //
        Tampilkan error jika harga minimum lebih besar atau sama dengan harga maksimum
        kembali
    } jika tidak {
        nilaiHargaMin = ambilHargaMinDariDB() // Ambil harga minimum dari database
        nilaiHargaMax = ambilHargaMaxDariDB() // Ambil harga maksimum dari database
    }

    // Periksa jika ada cukup smartphone dalam rentang harga
    jumlahSmartphone = hitungSmartphone(nilaiHargaMin, nilaiHargaMax)
}

```



```

    jika jumlahSmartphone < 1 {
        tampilkanError('Tidak cukup smartphone dalam rentang harga') // Tampilkan error jika
        tidak ada cukup smartphone dalam rentang harga
        kembali
    }

    // Normalisasi bobot
    bobotNormal = normalisasiBobot({
        body: bobotBadan,
        display: bobotLayar,
        system: bobotSistem,
        memory: bobotMemori,
        mainCamera: bobotKameraUtama,
        frontCamera: bobotKameraDepan,
        battery: bobotBaterai,
        price: bobotHarga
    })
    // Simpan bobot dalam sesi
    simpanBobotDalamSesi(bobotNormal) // Simpan bobot yang dinormalisasi ke dalam sesi
    simpanRentangHargaDalamSesi(nilaiHargaMin, nilaiHargaMax) // Simpan rentang harga ke
    dalam sesi
    aktifkanRekomendasi() // Aktifkan rekomendasi

    // Simpan bobot ke dalam database
    simpanBobotKeDB({
        body: bobotBadan,
        display: bobotLayar,
        system: bobotSistem,
        memory: bobotMemori,
        mainCamera: bobotKameraUtama,
        frontCamera: bobotKameraDepan,
        battery: bobotBaterai,
        price: bobotHarga,
        minPrice: nilaiHargaMin,
        maxPrice: nilaiHargaMax,
        createdAt: ambilWaktuSekarang(),
        updatedAt: ambilWaktuSekarang()
    })

    // Alihkan ke halaman algoritma rekomendasi
    alihkanKeHalamanAlgoritmaRekomendasi() // Alihkan ke halaman algoritma
    rekomendasi
}

// Fungsi bantu
fungsi ambilPost(field) { /* Ambil data POST */ }
fungsi tampilkanError(pesan) { /* Tampilkan pesan error */ }
fungsi hapusKarakterNonAngka(nilai) { /* Hapus karakter non-angka dari nilai */ }
fungsi aktifkanFilter() { /* Aktifkan filter */ }
fungsi ambilHargaMaxDariDB() { /* Ambil harga maksimum dari database */ }
fungsi ambilHargaMinDariDB() { /* Ambil harga minimum dari database */ }
fungsi hitungSmartphone(min, max) { /* Hitung smartphone dalam rentang harga */ }
fungsi normalisasiBobot(bobot) { /* Normalisasi bobot */ }
fungsi simpanBobotDalamSesi(bobot) { /* Simpan bobot dalam sesi */ }
fungsi simpanRentangHargaDalamSesi(min, max) { /* Simpan rentang harga dalam sesi */ }
fungsi aktifkanRekomendasi() { /* Aktifkan rekomendasi */ }
fungsi simpanBobotKeDB(bobot) { /* Simpan bobot ke dalam database */ }
fungsi ambilWaktuSekarang() { /* Ambil waktu sekarang */ }

```

```
fungsi alihkanKeHalamanAlgoritmaRekomendasi() { /* Alihkan ke halaman algoritma
rekomendasi */ }
```

Dalam *function* ini akan menerima input dari *form* penentuan halaman rekomendasi yang dimasukan oleh pengguna publik, data bobot yang telah ditentukan pengguna publik akan disimpan pada *database* dan di simpan pada *session* seperti yang telah dituliskan pada baris 52-79.

#### 4.2.2 Konversi Data

Tabel 4. 8 Pseudocode Kriteria Data

```
Fungsi konversi(kriteria, sub_kriteria)
// Ambil data smartphone berdasarkan sub_kriteria
ambil = smartphone.getBy(sub_kriteria)

// Buat array kosong untuk menyimpan hasil konversi dari data yang diambil
konversi_ambil = []
Untuk setiap data dalam ambil
    tambahkan strtolower(data[sub_kriteria]) ke dalam konversi_ambil

// Ambil data bobot yang sesuai dengan sub_kriteria
bobot_data = bobot.select('konversi').where('sub_kriteria', sub_kriteria).findAll()

// Buat array kosong untuk menyimpan hasil konversi dari bobot
konversi_bobot = []
Untuk setiap bobot dalam bobot_data
    tambahkan bobot['konversi'] ke dalam konversi_bobot

// Tambahkan data yang tidak ada dalam konversi_bobot
Untuk setiap konversi dalam konversi_ambil
    Jika konversi tidak ada dalam konversi_bobot
        input = [      'kriteria' => kriteria,      'sub_kriteria' => sub_kriteria,
'konversi' => konversi,      'created_at' => tanggal_dan_waktu_sekarang(),
'updated_at' => tanggal_dan_waktu_sekarang(),      ]
        bobot.insert(input)

// Hapus data yang ada dalam konversi_bobot tetapi tidak ada dalam konversi_ambil
Untuk setiap konversi dalam konversi_bobot
    Jika konversi tidak ada dalam konversi_ambil
        bobot.where('konversi', konversi).where('sub_kriteria', sub_kriteria).delete()
```

Bobot konversi data mengambil dari data laptop berdasarkan sub kriteria yang telah di tentukan, data laptop sebelum di ambil akan di lakukan penyortiran agar tidak ada kesamaan dalam pengambilan bobot data konversi, dan jika bobot data konversi sudah ada berdasarkan data laptopnya maka bobot data konversi tidak ditambahkan, selanjutnya akan disimpan pada tabel

bobot konversi pada *database* berdasarkan kriteria dan sub kriteria yang telah ditentukan sebelumnya.

Tabel 4.9 Pseudocode Filter Harga

```

Fungsi filter(harga)
// Mengambil nilai harga minimum dan maksimum dari sesi
hargaMin = sesi()->dapatkan('hMin')
hargaMax = sesi()->dapatkan('hMax')

// Mengambil data harga minimum dan maksimum dari database smartphone
hargaMinResult = smartphone.pilih('harga').dimana('harga >=', hargaMin)
                .dimana('harga <=', hargaMax).urutkan('harga', 'asc').pertama()
hargaMaxResult = smartphone.pilih('harga').dimana('harga >=', hargaMin)
                .dimana('harga <=', hargaMax).urutkan('harga', 'desc').pertama()

// Menghitung jarak harga
jarak = hargaMaxResult['harga'] - hargaMinResult['harga']
step = jarak / 100
hargaPertama = hargaMinResult['harga'] + step

// Inisialisasi nilai konversi
nilai = 0
Untuk i = 0 hingga 99
    Jika harga <= hargaPertama + (step * i)
        nilai = i
        Berhenti
    Selain itu
        nilai = 100
    Akhir Jika
    Akhir Untuk
// Mengembalikan nilai konversi
Kembalikan nilai
Akhir Fungsi
  
```

Tabel 4.10 Pseudocode Normalisasi Harga

```

Fungsi normalisasiHarga(data)
    normalisasi = (100 - data) / (100 - 0)
    Kembalikan normalisasi
Akhir Fungsi
  
```

Tabel 4.11 Pseudocode Filter Bobot Harga

```

Fungsi filterBobotHarga(data)
    hasil = data = sesi()->dapatkan('price')
    Kembalikan hasil
Akhir Fungsi
  
```

Jika pada saat pengguna publik ingin menentukan filter harga pada halaman rekomendasi, maka untuk data konversi, normalisasi, dan hasil

akhir akan dibuatkan struktur konversi, normalisasi, hasil akhir otomatis yang baru secara sistematis.

Tabel 4.12 Pseudocode Konversi Data Umum

```

Fungsi umum(sub)
// Ambil data smartphone berdasarkan sub_kriteria
dSm = smartphone.pilih('id, ' + sub).temukanSemua()

// Untuk setiap data smartphone
Untuk setiap sm dalam dSm
// Ambil nilai bobot berdasarkan sub_kriteria dan konversi
bobot = bobot.dimana('sub_kriteria', sub).dimana('konversi', sm[sub]).pertama()

// Update nilai kuantitatif pada tabel kuantitatif berdasarkan id smartphone
kuantitatif.dimana('id_smartphone', sm['id']).atur(sub, bobot['nilai']).atur('updated_at',
tanggal_dan_waktu_sekarang()).perbarui()
Akhir Fungsi

```

Tabel 4. 13 Pseudocode Konversi Data Khusus

```

Fungsi angka(sub)
// Ambil data smartphone berdasarkan id, merek, dan sub_kriteria
smartphone = smartphone.pilih('id, merek, ' + sub).temukanSemua()

// Untuk setiap data smartphone
Untuk setiap sm dalam smartphone
// Ambil bobot konversi berdasarkan sub_kriteria dan urutkan berdasarkan nilai
getBobot = bobot.dimana('sub_kriteria', sub).urutkan('CAST(nilai SEBAGAI UNSIGNED)',
'asc').temukanSemua()

// Ambil nilai maksimum berdasarkan sub_kriteria dan urutkan berdasarkan konversi
max = bobot.dimana('sub_kriteria', sub).urutkan('CAST(konversi SEBAGAI UNSIGNED)',
'desc').pertama()

// Untuk setiap bobot dalam getBobot
Untuk setiap bbt dalam getBobot
// Jika nilai sub_kriteria smartphone lebih besar dari konversi maksimum
Jika (float)sm[sub] > (float)max['konversi']
    nilai = 100
// Update nilai kuantitatif pada tabel kuantitatif berdasarkan id smartphone
kuantitatif.dimana('id_smartphone', sm['id'])
    .atur(sub, nilai)
    .atur('updated_at', tanggal_dan_waktu_sekarang())
    .perbarui()
Berhenti
Akhir Jika

// Jika nilai sub_kriteria smartphone lebih kecil dari konversi bobot
Jika (float)sm[sub] < (float)bbt['konversi']
    nilai = bbt['nilai']
// Tampilkan nilai bobot
tampilkan bbt['nilai'] + ' <b>nilai-nya</b><br>'

```

```
// Update nilai kuantitatif pada tabel kuantitatif berdasarkan id smartphone
kuantitatif.dimana('id_smartphone', sm['id'])
    .atur(sub, nilai)
    .atur('updated_at', tanggal_dan_waktu_sekarang())
    .perbarui()
Berhenti
Akhir Jika
Akhir Untuk
Akhir Untuk
Akhir Fungsi
```

Setelah bobot konversi ditentukan dan telah diberi nilai oleh admin, maka sistem dapat melakukan konversi, ada beberapa *function* untuk melakukan konversi data *smartphone*. Gambar 4.4 diatas merupakan *function* untuk konversi data yang hanya memerlukan kesamaan, seperti Tipe Layar, Tipe Kamera, RAM, ROM, Chipset, *Build*, dll. Data yang telah dikonversi akan disimpan kedalam tabel *database* kuantitatif berdasarkan *id\_smartphone* dari data *smartphone*. Pada gambar diatas merupakan konversi dari data bobot konversi dengan subkriteria yang berupa angka.

#### 4.2.3 Normalisasi Data

Tabel 4. 14 Pseudocode Normalisasi Data

```
Fungsi konversi(sub)
// Ambil data kuantitatif berdasarkan id_smartphone dan sub_kriteria
dKuan = kuantitatif.pilih('id_smartphone, ' + sub).temukanSemua()

// Ambil nilai minimum dari sub_kriteria
min = kuantitatif.pilih(sub)
    .urutkan('CAST(' + sub + ' SEBAGAI UNSIGNED)', 'asc')
    .pertama()

// Ambil nilai maksimum dari sub_kriteria
max = kuantitatif.pilih(sub)
    .urutkan('CAST(' + sub + ' SEBAGAI UNSIGNED)', 'desc')
    .pertama()

// Hitung total konversi
totalKonversi = kuantitatif.hitungSemuaHasil()

// Jika total konversi kurang dari atau sama dengan 1
Jika totalKonversi <= 1
    // Set pesan error dalam sesi
    sesi().setFlashdata('error', 'Data belum dikonversi!')
    // Redirect ke halaman sebelumnya
    kembaliKeHalamanSebelumnya()
    kembali
```

```

// Jika nilai minimum dan maksimum dari sub_kriteria bukan 0
Jika min[sub] != 0 dan max[sub] != 0
  // Untuk setiap data kuantitatif
  Untuk setiap dk dalam dKuan
    // Jika sub_kriteria bukan 'harga'
    Jika sub != 'harga'
      // Hitung nilai normalisasi
      nilai = (dk[sub] - min[sub]) / (max[sub] - min[sub])
      // Update nilai konversi dalam tabel konversi
      convert.dimana('id_smartphone', dk['id_smartphone'])
        .atur(sub, nilai)
        .perbarui()
    // Jika sub_kriteria adalah 'harga'
    Lain
      // Hitung nilai normalisasi
      nilai = (max[sub] - dk[sub]) / (max[sub] - min[sub])
      // Set tanggal dan waktu sekarang
      tanggal = tanggal_dan_waktu_sekarang()
      // Update nilai konversi dalam tabel konversi
      convert.dimana('id_smartphone', dk['id_smartphone'])
        .atur(sub, nilai)
        .perbarui()

// Set pesan sukses dalam sesi
sesi().setFlashdata('successKonversi', 'Berhasil Melakukan Normalisasi Data.')
Akhir Fungsi

```

Setelah data diubah menjadi nilai kuantitatif maka akan dilakukan normalisasi setiap data *smartphone* dari tabel *database* kuantitatif. *Utility* kriteria harga menggunakan rumus *benefit*, selain itu menggunakan rumus *cost*. Setelah ternormalisasi maka data akan tersimpan pada tabel *database* normalisasi.

#### 4.2.4 Hasil Akhir

Tabel 4. 15 Pseudocode Hasil Akhir

```

Fungsi hitung(perhitungan)
  // Ambil data normalisasi dengan rentang harga dari sesi
  dNorm = norm.ambilDenganRentang(sesi().ambil('hMin'), sesi().ambil('hMax'))
  // Untuk setiap data dalam dNorm
  Untuk setiap data dalam dNorm
    // Mengalikan dimensi dengan bobot 'body' dari sesi
    data['dimensi'] = data['dimensi'] * sesi().ambil('body')

    // Mengalikan berat dengan bobot 'body' dari sesi
    data['berat'] = data['berat'] * sesi().ambil('body')

    // Mengalikan build dengan bobot 'body' dari sesi
    data['build'] = data['build'] * sesi().ambil('body')

    // Mengalikan lcd_type dengan bobot 'display' dari sesi
    data['lcd_type'] = data['lcd_type'] * sesi().ambil('display')

```

```

// Mengalikan lcd_size dengan bobot 'display' dari sesi
data['lcd_size'] = data['lcd_size'] * sesi().ambil('display')

// Mengalikan lcd_resolusi dengan bobot 'display' dari sesi
data['lcd_resolusi'] = data['lcd_resolusi'] * sesi().ambil('display')

// Mengalikan os dengan bobot 'system' dari sesi
data['os'] = data['os'] * sesi().ambil('system')

// Mengalikan chipset dengan bobot 'system' dari sesi
data['chipset'] = data['chipset'] * sesi().ambil('system')

// Mengalikan cpu dengan bobot 'system' dari sesi
data['cpu'] = data['cpu'] * sesi().ambil('system')

// Mengalikan ram dengan bobot 'memory' dari sesi
data['ram'] = data['ram'] * sesi().ambil('memory')

// Mengalikan rom dengan bobot 'memory' dari sesi
data['rom'] = data['rom'] * sesi().ambil('memory')

// Mengalikan main_camera dengan bobot 'mainCamera' dari sesi
data['main_camera'] = data['main_camera'] * sesi().ambil('mainCamera')

// Mengalikan main_type dengan bobot 'mainCamera' dari sesi
data['main_type'] = data['main_type'] * sesi().ambil('mainCamera')

// Mengalikan main_video dengan bobot 'mainVideo' dari sesi
data['main_video'] = data['main_video'] * sesi().ambil('mainVideo')

// Mengalikan front_camera dengan bobot 'frontCamera' dari sesi
data['front_camera'] = data['front_camera'] * sesi().ambil('frontCamera')

// Mengalikan front_video dengan bobot 'frontCamera' dari sesi
data['front_video'] = data['front_video'] * sesi().ambil('frontCamera')

// Mengalikan usb dengan bobot 'battery' dari sesi
data['usb'] = data['usb'] * sesi().ambil('battery')

// Mengalikan battery_capacity dengan bobot 'battery' dari sesi
data['battery_capacity'] = data['battery_capacity'] * sesi().ambil('battery')

// Jika filter diaktifkan dalam sesi
Jika sesi().ambil('filter')
  Jika perhitungan >= 1
    // Menggunakan fungsi filter untuk mengatur harga
    data['harga'] = filter(data['sHarga'])

  Jika perhitungan >= 2
    // Normalisasi harga
    data['harga'] = normalisasiHarga(data['harga'])

  Jika perhitungan >= 3
    // Menggunakan filter bobot harga
    data['harga'] = filterBobotHarga(data['harga'])
Lain
  // Mengalikan harga dengan bobot 'price' dari sesi

```

```

data['harga'] = data['harga'] * sesi().ambil('price')

// Menghitung total nilai
data['total'] = (
    ((data['dimensi'] + data['berat'] + data['build']) / 3) * sesi().ambil('body')
    + ((data['lcd_type'] + data['lcd_size'] + data['lcd_resolusi']) / 3) *
sesi().ambil('display')
    + ((data['os'] + data['chipset'] + data['cpu']) / 3) * sesi().ambil('system')
    + ((data['ram'] + data['rom']) / 2) * sesi().ambil('memory')
    + ((data['usb'] + data['battery_capacity']) / 2) * sesi().ambil('battery')
    + ((data['main_camera'] + data['main_type'] + data['main_video']) / 3) *
sesi().ambil('mainCamera')
    + ((data['front_camera'] + data['front_video']) / 2) * sesi().ambil('frontCamera')
    + (data['harga'] * sesi().ambil('price'))
)
Akhir Untuk

Kembalikan dNorm
Akhir Fungsi

```

Dalam function hitung, perhitungan hasil akhir membawa parameter ketentuan jika pengguna publik menggunakan *filter* harga atau tidak, perhitungan hasil akhir mengambil data dari tabel *database* normalisasi dan akan dikalikan dengan hasil normalisasi bobot dari pengguna publik. Setelah itu ditambahkan *array* baru dengan nama total yang berisi total dari semua hasil akhir *smartphone* tersebut.

#### 4.2.5 Perangkingan

Tabel 4. 16 Pseudocode Perangkingan

```

// Memanggil fungsi hitung dengan parameter 3 dan menyimpan hasilnya ke dalam variabel
dataHasil
dataHasil = hitung(3)
// Mengurutkan data dalam dataHasil
Urutkan dataHasil berdasarkan fungsi berikut:
Fungsi urutkan(a, b)
    // Jika total nilai b tidak sama dengan total nilai a
    Jika b['total'] tidak sama dengan a['total']
        // Bandingkan nilai total b dan a menggunakan bccomp dengan skala 10
        Kembalikan hasil dari bccomp(b['total'], a['total'], 10)
    Lain
        // Jika total nilai b sama dengan total nilai a, bandingkan sHarga a dan b menggunakan
strcmp
        Kembalikan hasil dari strcmp(a['sHarga'], b['sHarga'])
Akhir Fungsi
// Menerapkan fungsi urutkan untuk mengurutkan dataHasil
urutkan(dataHasil, urutkan)

```

Mendelarasikan variabel \$dataHasil berisi hasil dari *function* hitung, setelah itu menggunakan *function* bawaan PHP untuk melakukan pengurutan

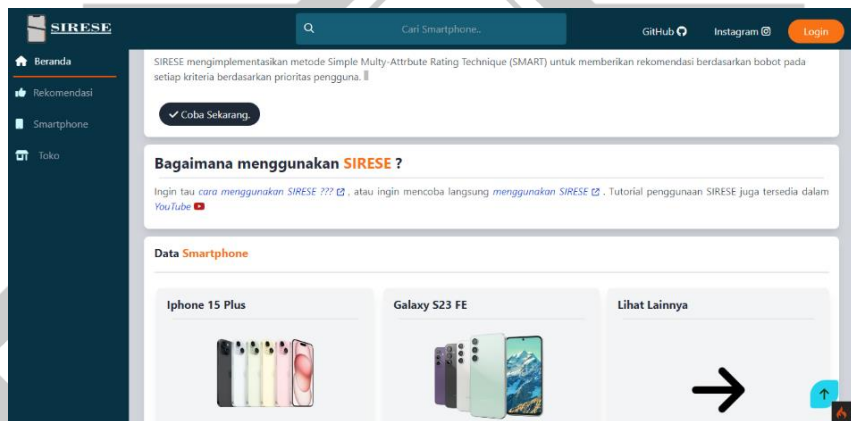


berdasarkan *array* total tertinggi, jika ada *array* total memiliki kesamaan, maka pada kesamaan tersebut diurutkan berdasarkan harga terendah.

### 4.3 Implementasi Sistem

Produk yang dihasilkan dari penelitian ini merupakan sebuah sistem yang membantu memberikan rekomendasi pembelian *smartphone* menggunakan metode SMART.

#### 4.3.1 Tampilan Halaman Beranda Publik



Gambar 4. 1 Halaman Beranda Umum

Pada halaman ini berisikan penjelasan mengenai sistem, tatacara penggunaan sistem, dua data *smartphone* acak, pencarian, dan tombol-tombol lainnya.

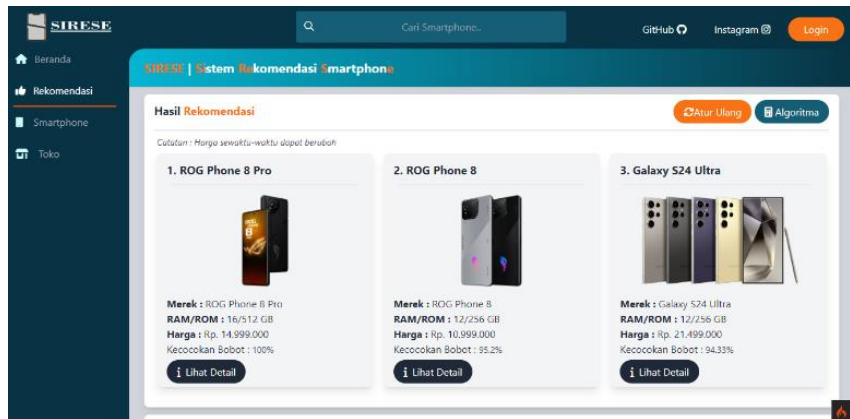
#### 4.3.2 Tampilan Halaman Rekomendasi

Gambar 4.2 Halaman Rekomendasi

Halaman rekomendasi akan berisikan *form* untuk menentukan bobot setiap kriteria berdasarkan preferensi pengguna. Terdapat juga *filter* harga

jika pengguna ingin mendapatkan rekomendasi berdasarkan harga yang sesuai.

#### 4.3.3 Tampilan Halaman Hasil Rekomendasi



Gambar 4.3 Halaman Hasil Rekomendasi

Setelah memasukan bobot kriteria, pengguna akan diarahkan ke halaman hasil rekomendasi yang akan memberikan 3 (tiga) hasil rekomendasi teratas dan memberikan tabel peringkat dari keseluruhan *smartphone*. Setelah tahapan ini maka tombol navigasi rekomendasi akan langsung mengarah pada halaman hasil rekomendasi, kecuali pengguna menekan tombol atur ulang.

#### 4.3.4 Tampilan Halaman Algoritma Perhitungan

← kembali

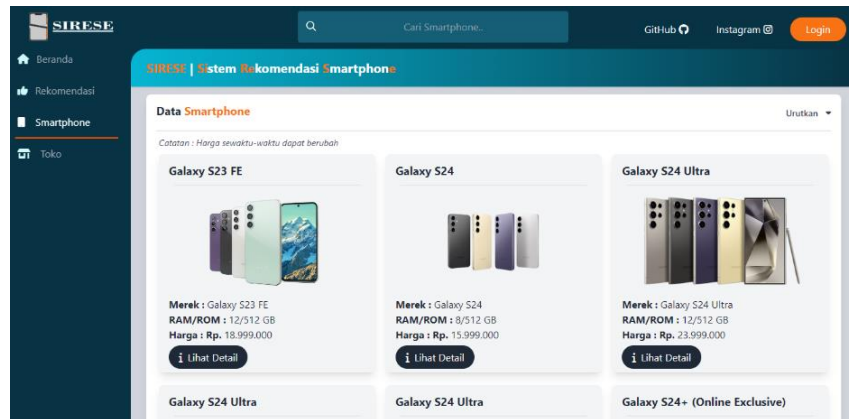
</

Gambar 4.4 Halaman Algoritma Perhitungan

Jika dalam halaman hasil rekomendasi menekan tombol algoritma, maka akan menampilkan halaman algoritma perhitungan. Disini akan

menampilkan keseluruhan perhitungan dari perhitungan bobot kriteria, konversi data, normalisasi data, hingga hasil akhir.

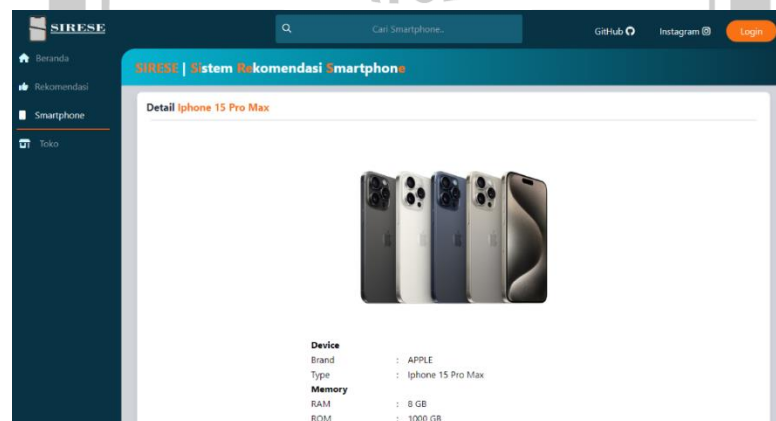
#### 4.3.5 Tampilan Halaman *Smartphone*



Gambar 4.5 Halaman *Smartphone*

Halaman *smartphone* menampilkan keseluruhan data *smartphone* yang ada dalam *database* dan dibuat pemisahan 6 data setiap slide halaman agar meningkatkan kenyamanan pengguna. Pada halaman ini disediakan pengurutan data berdasarkan merek *ascending* / *descending*, harga *ascending* / *descending*, dan *filter* berdasarkan harga.

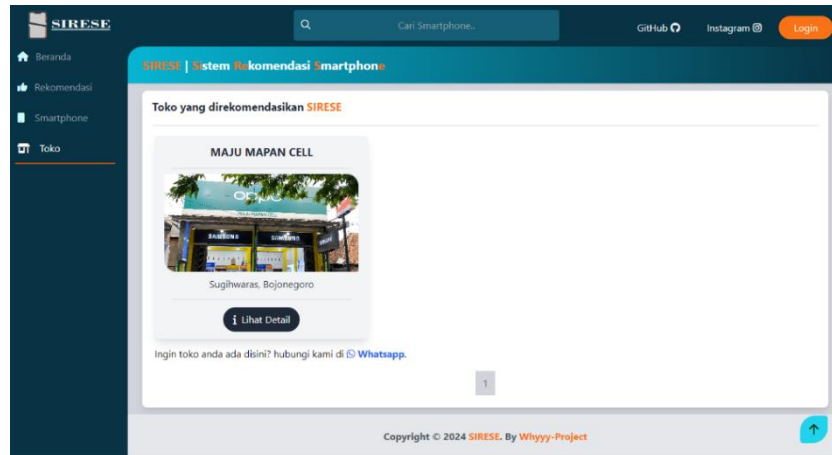
#### 4.3.6 Tampilan Halaman Detail *Smartphone*



Gambar 4.6 Halaman Detail *Smartphone*

Halaman detail *smartphone* akan menampilkan keseluruhan detail pada data *smartphone* yang digunakan. Detail-detail tersebut juga merupakan kriteria serta subkriteria yang diperhitungkan.

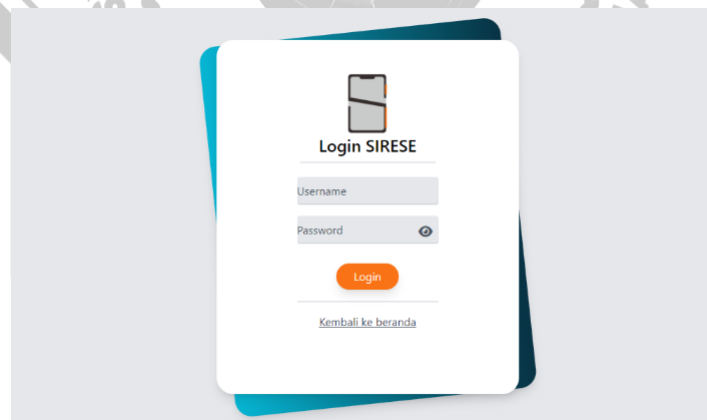
#### 4.3.7 Tampilan Halaman Toko



Gambar 4.7 Halaman Toko

Halaman toko akan menampilkan toko yang bekerjasama dengan sistem tersebut. Toko dapat ditambahkan melalui sistem admin, jika akan ada toko yang bekerjasama dengan system tersebut dikemudian hari maka dapat menghubungi admin.

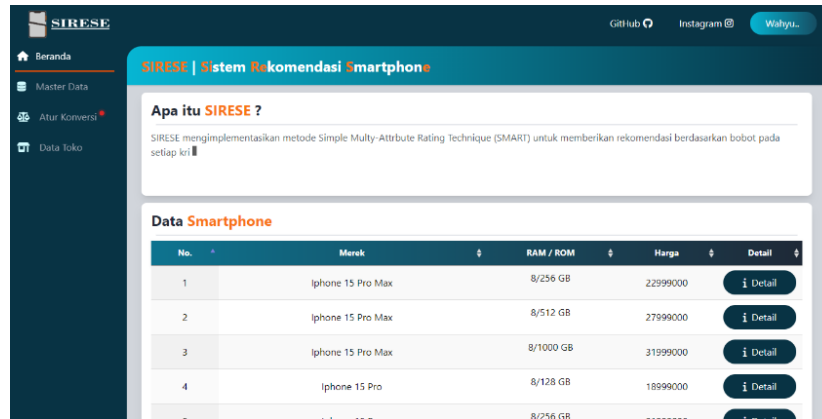
#### 4.3.8 Tampilan Halaman *Login*



Gambar 4.8 Halaman *Login* Admin

Halaman *login* ini berfungsi untuk mengatur keseluruhan data pada sistem rekomendasi tersebut. Halaman ini juga akan ditampilkan jika ada pengguna ingin mengakses halaman admin secara langsung tanpa login.

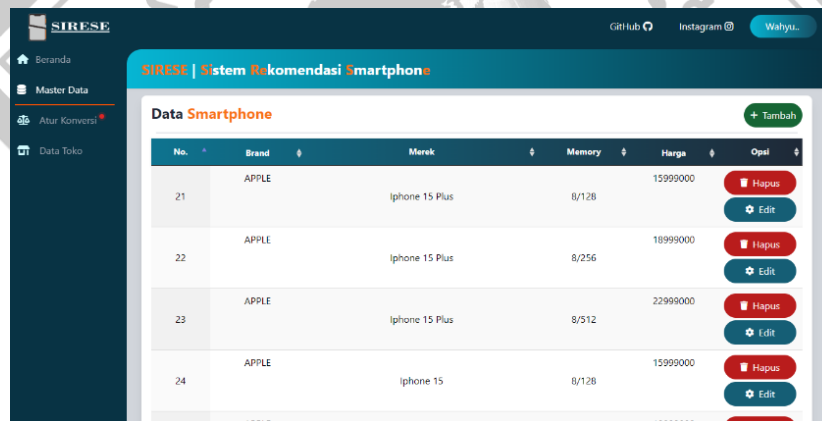
### 4.3.9 Tampilan Halaman Beranda Admin



Gambar 4.9 Halaman Beranda Admin

Dalam halaman beranda admin akan menampilkan keseluruhan data *smartphone* yang ada dalam sistem tersebut dan juga tombol-tombol navigasi.

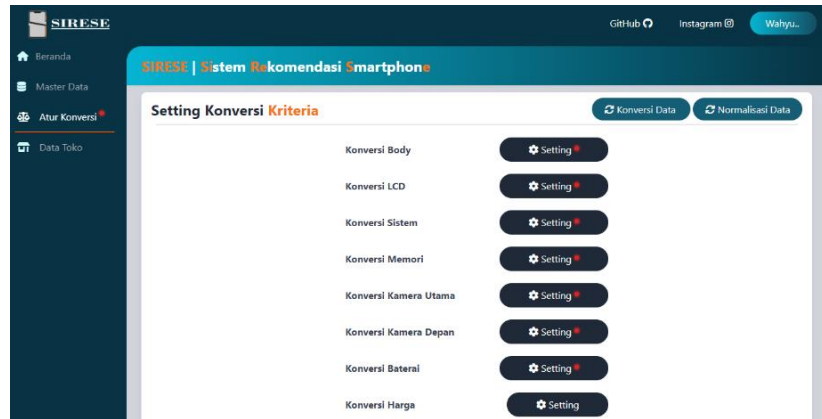
### 4.3.10 Tampilan Halaman Master Data Admin



Gambar 4.10 Halaman Master Data

Halaman master data berisi data *smartphone*, tambah data *smartphone*, edit data *smartphone*, hapus data *smartphone*, data konversi, dan data normalisasi. Untuk melakukan konversi maupun normalisasi data, admin perlu menekan tombol konversi ataupun tombol normalisasi.

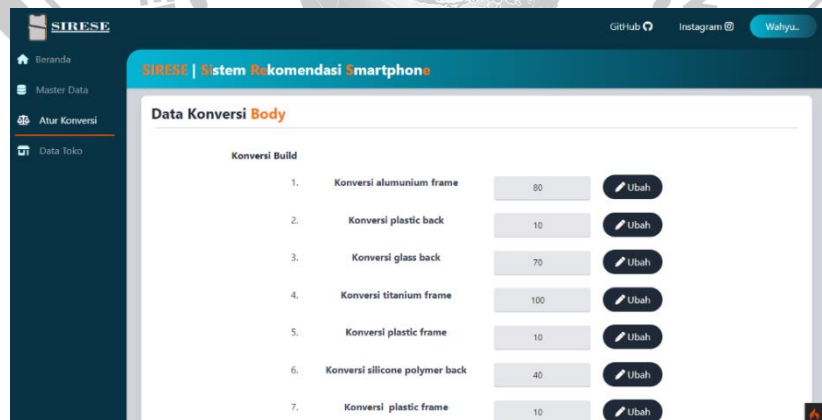
#### 4.3.11 Tampilan Halaman Atur Konversi Admin



Gambar 4.11 Halaman Atur Konversi

Halaman atur konversi ini akan menampilkan keseluruhan kriteria yang akan dikonversi, dalam setiap kriteria akan memiliki subkriteria masing-masing. Pada halaman ini juga akan menampilkan tombol untuk mengkonversi data dan juga tombol untuk normalisasi data.

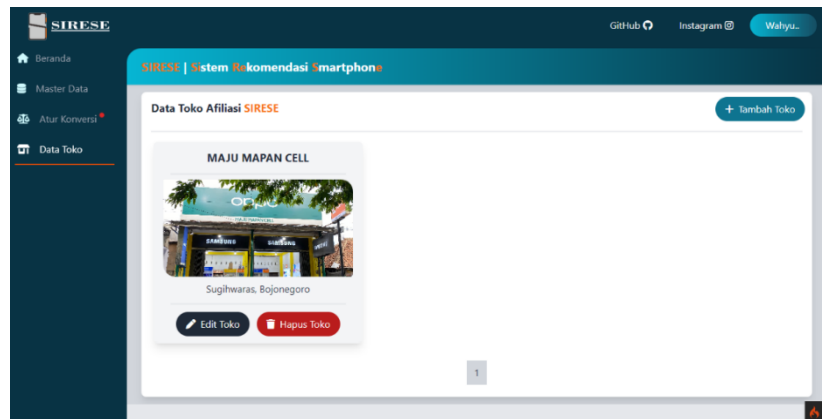
#### 4.3.12 Tampilan Halaman Data Konversi *Body* Admin



Gambar 4.12 Halaman Data Konversi *Body*

Gambar diatas merupakan contoh salah satu halaman kriteria *body* yang memiliki sub kriteri *build*, dimensi, dan berat. Setiap kriteria memiliki sub kriteria masing-masing, dalam hal ini bobot konversi akan diberikan oleh admin dengan rentang nilai 0-100.

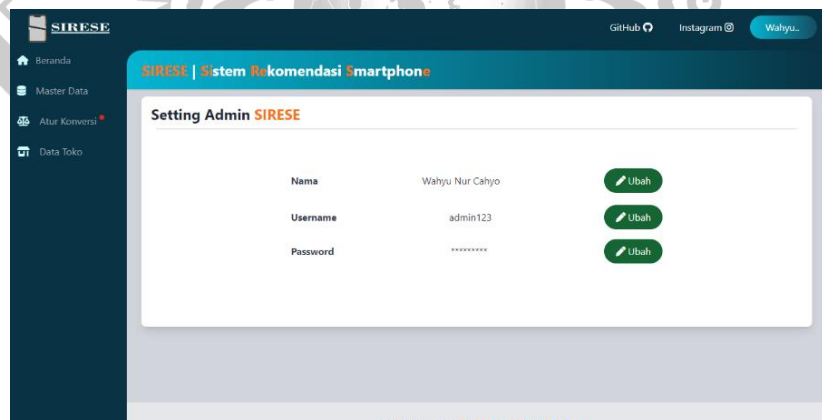
### 4.3.13 Tampilan Halaman Data Toko Admin



Gambar 4.13 Halaman Data Toko

Halaman data toko menampilkan data toko yang akan ditampilkan ditampilkan pengguna publik. Admin dapat menambah, mengubah, maupun menghapus data toko yang ada.

### 4.3.14 Tampilan Profil Admin



Gambar 4.14 Halaman Profil Admin

Halaman profil admin akan menampilkan data akun yang digunakan untuk login pada sistem. Admin dapat mengganti Nama, *Username*, dan *Password* pada akun admin tersebut.

## 4.4 Hasil Pengujian

Hasil pengujian sistem aplikasi ini akan berupa hasil pengujian *black-box* serta hasil pengujian uji angket, berikut merupakan hasil dari pengujian *black-box* dan hasil pengujian uji angket :

#### 4.4.1 Hasil Pengujian *Black-Box*

Pengujian sistem yang menggunakan metode uji *black-box* telah dijelaskan dalam BAB III. Sesuai dengan rencana pengujian *black-box* maka akan diperoleh hasil pengujian sebagai berikut :

Tabel 4.17 Hasil Pengujian *Black-Box*

No	Halaman yang diuji	Kasus Uji	Skenario Uji	Hasil yang diinginkan	Hasil	
					Valid	Tidak
1.	Beranda Umum	Halaman beranda umum	Masuk kedalam halaman beranda umum	Menampilkan halaman beranda umum	✓	
		Tombol pada beranda umum	Menekan tombol - tombol yang berada di <i>dashboard</i> umum	Menampilkan halaman/hasil sesuai penjelasan sebelumnya	✓	
		Data yang ditampilkan	Melihat keseluruhan data yang ditampilkan pada halaman tersebut	data yang ditampilkan sesuai dengan keterangan yang ada	✓	
2.	Rekomendasi	Halaman rekomendasi	Masuk kedalam halaman rekomendasi	Melihat halaman rekomendasi <i>smartphone</i>	✓	
		<i>Form</i> penentuan	Melihat, mengisi, mengirimkan	Sistem dapat memproses data yang	✓	



No	Halaman yang diuji	Kasus Uji	Skenario Uji	Hasil yang diinginkan	Hasil	
					Valid	Tidak
		bobot kriteria	data <i>form</i> pada halaman rekomendasi <i>smartphone</i>	dimasukan, serta menampilkan halaman hasil perhitungan		
		Halaman hasil perhitungan	Setelah mengisi <i>form</i> bobot kriteria maka akan diarahkan ke halaman hasil perhitungan	Sistem menampilkan hasil perhitungan rekomendasi <i>smartphone</i>	✓	
		Tombol-tombol fitur pada halaman hasil perhitungan	Menekan tombol-tombol pada halaman hasil perhitungan	Tombol-tombol pada halaman hasil perhitungan berfungsi sepenuhnya	✓	
3.	Smartphone	Halaman <i>smartphone</i>	Membuka halaman data <i>smartphone</i>	Menampilkan data <i>smartphone</i>	✓	
		Data dan fitur pada halaman <i>smartphone</i>	Melihat dan menekan tombol/fitur pada halaman <i>smartphone</i>	Menampilkan data sesuai dengan keterangan yang ada	✓	
4.	Toko	Halaman Toko	Membuka halaman toko	Menampilkan halaman toko	✓	

No	Halaman yang diuji	Kasus Uji	Skenario Uji	Hasil yang diinginkan	Hasil	
					Valid	Tidak
		Detail Toko	Membuka halaman detail toko	Sistem menampilkan data detail toko	✓	
5.	Pencarian	<i>Form</i> pencarian	Mengisi form pencarian yang ada pada sistem	Menampilkan hasil pencarian sesuai data yang dimasukan	✓	
		Halaman hasil pencarian	Melihat halaman hasil pencarian	Menampilkan halaman hasil pencarian yang relevan	✓	
		Fitur dan data dalam halaman hasil pencarian	Melihat dan mencoba fitur yang ada pada halaman tersebut	Menampilkan sesuai dengan keterangan pada fitur dan data tersebut	✓	
6.	<i>Login</i>	Halaman <i>login</i>	Membuka halaman <i>login</i>	Menampilkan halaman <i>form login</i>	✓	
		<i>Form login</i>	Mengisi <i>username password</i> dan mengirimkan data <i>form login</i>	Sistem dapat melakukan validasi akun, jika benar akan membuat sesi login admin, jika tidak akan menampilkan notifikasi	✓	

No	Halaman yang diuji	Kasus Uji	Skenario Uji	Hasil yang diinginkan	Hasil	
					Valid	Tidak
				<i>username</i> atau <i>password</i> salah		
7.	Beranda admin	Halaman beranda admin	Menampilkan halaman beranda admin	Setelah <i>login</i> tervalidasi sistem dapat menampilkan halaman beranda admin	✓	
		Data dan fitur halaman beranda admin	Melihat dan mengecek fitur dan data beranda admin	Fitur dan data telah sesuai dengan keterangan yang ada	✓	
8.	Master data	Halaman master data smartphone	Menampilkan data yang sesuai dengan keterangan yang ada	Data yang ditampilkan pada halaman tersebut	✓	
		Data yang ditampilkan	Membuka halaman master data <i>smartphone</i>	Data pada halaman tersebut telah sesuai dengan keterangan yang ada	✓	
		Fitur tambah data	Melihat dan mengisi data <i>smartphone</i> yang baru pada halaman	Data yang telah di masukan tersimpan pada database sistem	✓	

No	Halaman yang diuji	Kasus Uji	Skenario Uji	Hasil yang diinginkan	Hasil	
					Valid	Tidak
			tambah data <i>smartphone</i>			
		Fitur edit data	Merubah data yang ditampilkan pada halaman edit data <i>smartphone</i>	Data yang telah diubah tersimpan pada <i>database</i>	✓	
		Fitur hapus data	Menekan hapus data salah satu <i>smartphone</i>	Data yang di hapus akan hilang dari <i>database</i>	✓	
9.	Bobot kriteria	Halaman bobot kriteria	Membuka halaman bobot kriteria	Menampilkan halaman bobot kriteria	✓	
		Halaman opsi suatu bobot kriteria	Membuka halaman opsi suatu bobot kriteria	Menampilkan form bobot kriteria sesuai opsi sebelumnya	✓	
		Form opsi suatu bobot kriteria	Mengubah / mengisi <i>form</i> yang ada	Data yang dimasukan tersimpan pada <i>database</i>	✓	
10.	Data Toko	Halaman data toko	Membuka halaman data toko	Menampilkan data toko	✓	

No	Halaman yang diuji	Kasus Uji	Skenario Uji	Hasil yang diinginkan	Hasil	
					Valid	Tidak
		Tambah Toko	Mengisi <i>form</i> tambah toko	Data toko yang ditambahkan disimpan dalam <i>database</i>	✓	
11.	Profil admin	Halaman profil dan akun admin	Membuka halaman profil dan akun admin	Menampilkan halaman profil admin	✓	
		<i>Form</i> ubah profil/akun admin	Menekan tombol ubah pada profil/akun admin	Menampilkan <i>form</i> ubah profil/akun admin	✓	
12.	<i>Logout</i>	Tombol <i>logout</i>	Menekan tombol <i>logout</i>	Keluar dari sistem admin, serta sistem menghapus sesi <i>login</i> admin, lalu menampilkan halaman beranda umum	✓	

#### 4.4.2 Hasil Pengujian Uji Angket Kelayakan

Pengujian uji angket kelayakan memiliki tujuan untuk menilai apakah suatu rancangan atau proyek sistem aplikasi yang telah dikerjakan memiliki aspek-aspek yang dibutuhkan. Berdasarkan angket yang telah di nilai oleh beberapa responden berikut merupakan nilai rata-rata yang diperoleh :

Tabel 4. 18 Nilai Rata – Rata Angket Kelayakan

No	Pertanyaan	Penilaian			
		Sangat Buruk	Buruk	Cukup Baik	Sangat baik
1	Desain Antarmuka Pengguna (UI) menarik dan mudah dipahami				96
2	Pengalaman Pengguna (UX) yang nyaman, menyenangkan, dan efisien				95
3	Responsif terhadap berbagai perangkat dan ukuran layar.				90
4	Waktu memuat dan pemrosesan data yang cepat.				94
5	Dalam memproses rekomendasi <i>smartphone</i> menggunakan metode SMART sistem dapat menampilkan rekomendasi sesuai kriteria yang dimasukkan.				93

Jika dilakukan perhitungan persentase nilai angket uji kelayakan, maka akan diperoleh hasil persentase sebagai berikut :

Tabel 4. 19 Hasil Persentase Angket Kelayakan

No	Pertanyaan	Persentase
1	Desain Antarmuka Pengguna (UI) menarik dan mudah dipahami	96%
2	Pengalaman Pengguna (UX) yang nyaman, menyenangkan, dan efisien	95%
3	Responsif terhadap berbagai perangkat dan ukuran layar.	90%

No	Pertanyaan	Persentase
4	Waktu memuat dan pemrosesan data yang cepat.	94%
5	Dalam memproses rekomendasi <i>smartphone</i> menggunakan metode SMART sistem dapat menampilkan rekomendasi sesuai kriteria yang dimasukan.	93%

Nilai rata-rata dari keseluruhan hasil persentase angket uji kelayakan yaitu 93%, hal ini menunjukkan bahwa sistem sudah layak jika digunakan untuk umum.

