



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB III

### METODOLOGI DAN PERANCANGAN

#### 3.1. Metode Penelitian

Metode penelitian yang digunakan untuk merancang dan membangun sistem rekomendasi ini adalah sebagai berikut:

1. Studi Literatur

Tahap pertama penelitian ini adalah dengan melakukan studi literatur menggunakan referensi buku, artikel, jurnal mengenai metode WASPAS dan berbagai sumber untuk menunjang pembangunan dan pengembangan sistem. Literatur-literatur tersebut dijadikan pedoman dalam melakukan penelitian.

2. Perancangan Aplikasi dan *Database*

Dalam tahap ini dilakukan perancangan sistem dan pembuatan *database* untuk menampung semua *data* yang telah didapatkan.

3. Pemrograman dan Implementasi Metode

Pada tahap ini dilakukan perancangan desain antar muka dengan bantuan *framework* Bootstrap, penulisan kode program dengan bahasa pemrograman PHP dan implementasi metode WASPAS pada sistem.

4. Pengujian Aplikasi

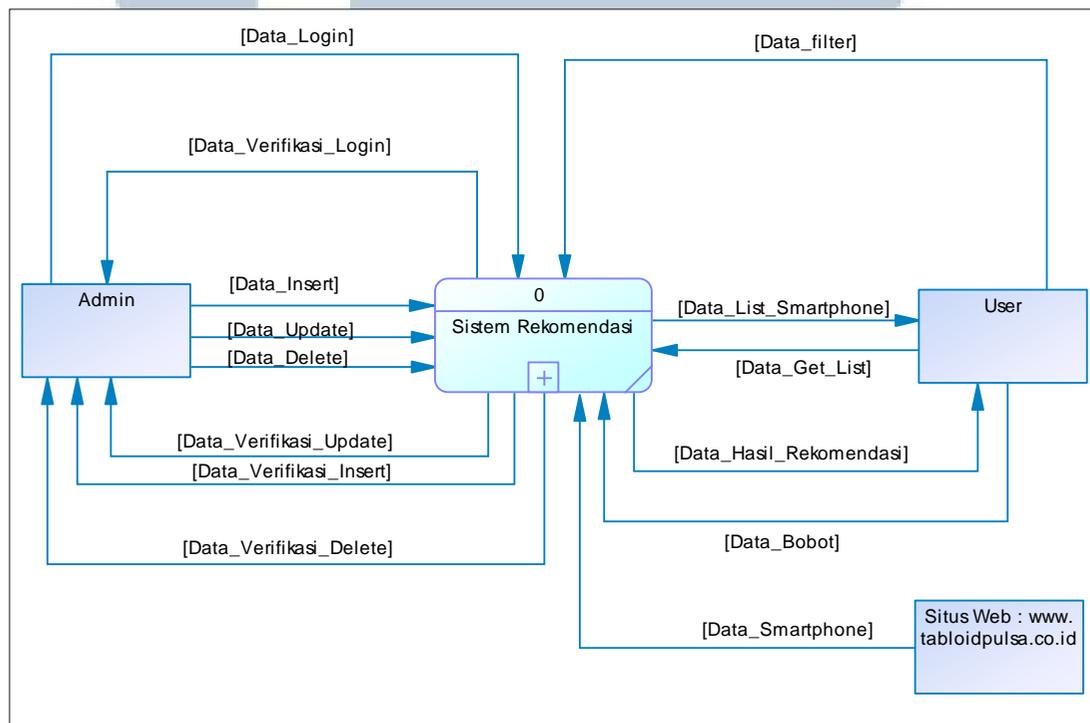
Pada tahap ini aplikasi diuji coba untuk mendapatkan hasil akurasi dari *data* yang ada dan meminimalisasikan kesalahan-kesalahan yang terdapat pada aplikasi tersebut.

### 3.2. Perancangan Sistem

Perancangan sistem yang dibuat dalam penelitian ini yaitu *data flow diagram* yang menggambarkan aliran *data* dari suatu proses ke proses lainnya di dalam sistem, *flowchart* yang menggambarkan alur proses pada sistem dan *Entity Relationship Diagram* yang menggambarkan hubungan antar tabel-tabel yang dibutuhkan dalam pembangunan sistem.

#### 3.2.1. Data Flow Diagram

Pada *data flow diagram* terdapat dua *level*, yaitu diagram *context*, dan *level 1*. Dimana pada *level 1* akan menjelaskan lebih rinci dari diagram *context*.



Gambar 3.1 Diagram *Context*

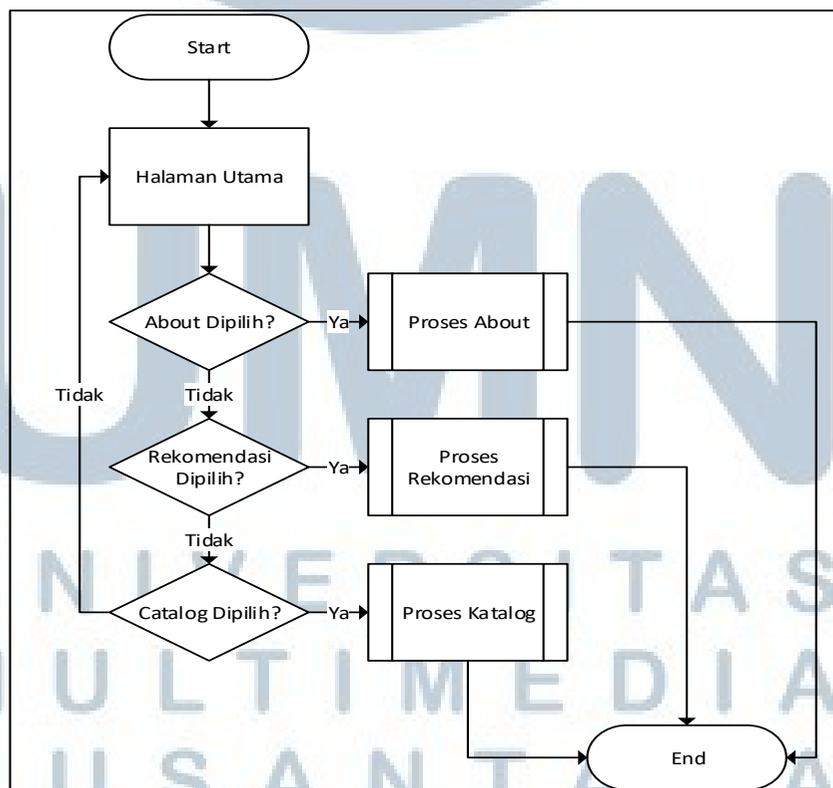
Pada sistem ini memiliki diagram *context* seperti pada gambar 3.1 dimana terdapat satu proses utama dan tiga entitas yang menggambarkan situs *web*, *admin*, dan *user* (pengguna) sistem. Terdapat delapan jenis *data* yang masuk ke dalam sistem, dan enam jenis *data* yang keluar dalam sistem. Secara detail, terdapat empat jenis *data*



proses-proses yang ada, tabel *smartphone* untuk memberikan *data* dari *list smartphone*, menambah *data* dan menyimpannya, memperbaharui *data*, atau menghapus *data*, sedangkan tabel *tab\_user* untuk menyimpan *data login admin*, dan tabel *brand* untuk memberikan *list* dari merk-merk *smartphone*.

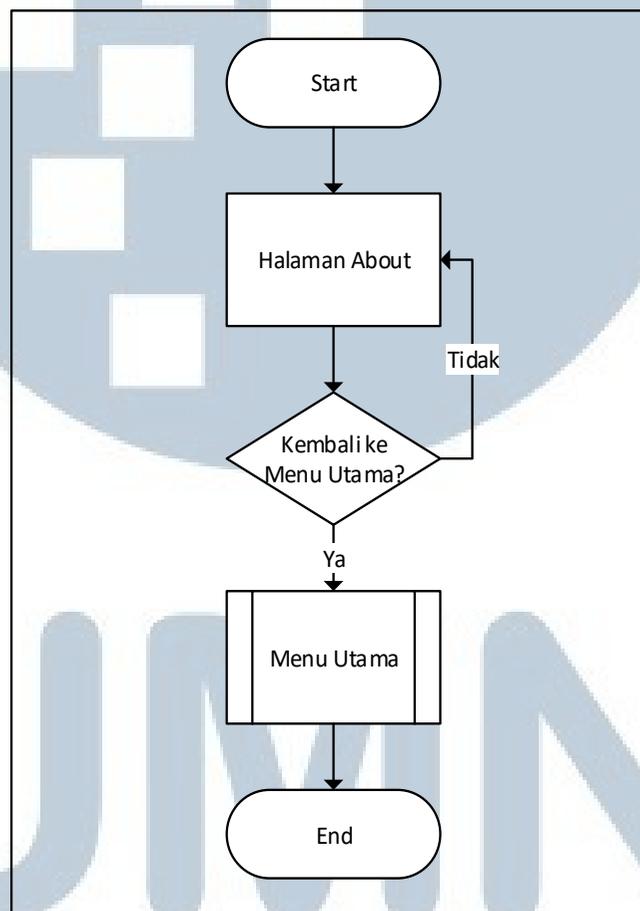
### 3.2.2. Flowchart

*Flowchart* atau bisa disebut diagram alir merupakan bagan-bagan yang memiliki arus yang menggambarkan langkah-langkah dan proses dari suatu sistem. Dalam sistem pemilihan *smartphone* ini dibagi menjadi 2 (dua), yaitu *frontend* dan *backend*. Pada bagian *frontend*, *user* dapat melihat rekomendasi *smartphone* berdasarkan kriteria dan bobot yang telah ditentukan oleh *user* sendiri. Sedangkan *backend* merupakan bagian administrasi yang melakukan pengolahan basis *data* yang digunakan oleh sistem.



Gambar 3.3 Flowchart Menu Utama

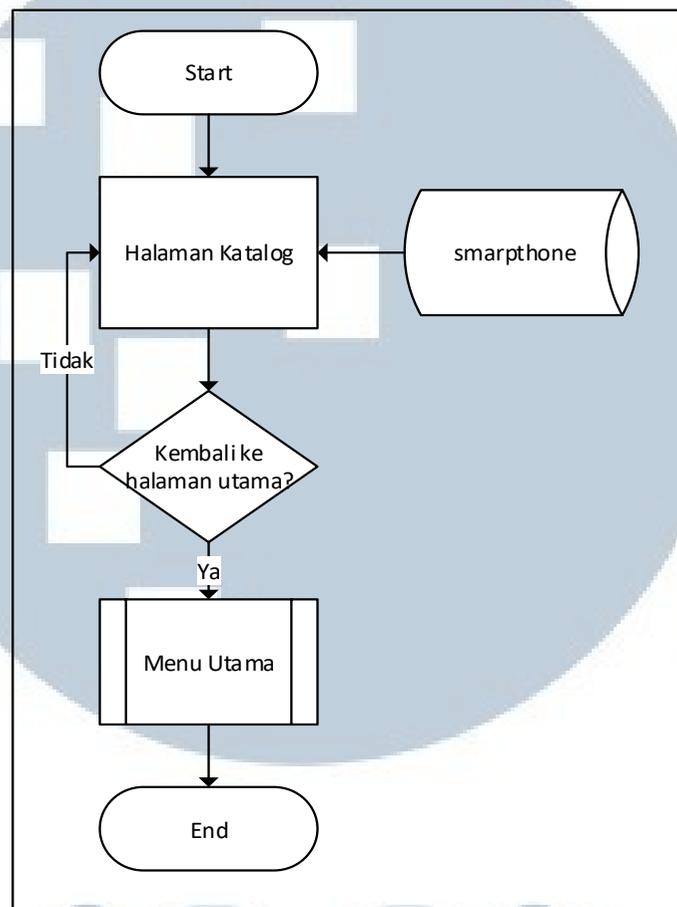
Halaman utama adalah halaman yang pertama kali muncul ketika membuka situs ini. Pada Halaman utama terdapat pilihan *menu* yang dapat dilihat oleh *user*, yaitu *about*, rekomendasi, dan katalog. Apabila *user* memilih *menu about*, maka proses *about* akan dilaksanakan, apabila *user* memilih *menu* rekomendasi, maka proses rekomendasi akan dilaksanakan, dan apabila *user* memilih katalog, maka proses katalog akan dilaksanakan.



Gambar 3.4 *Flowchart* Proses *About*

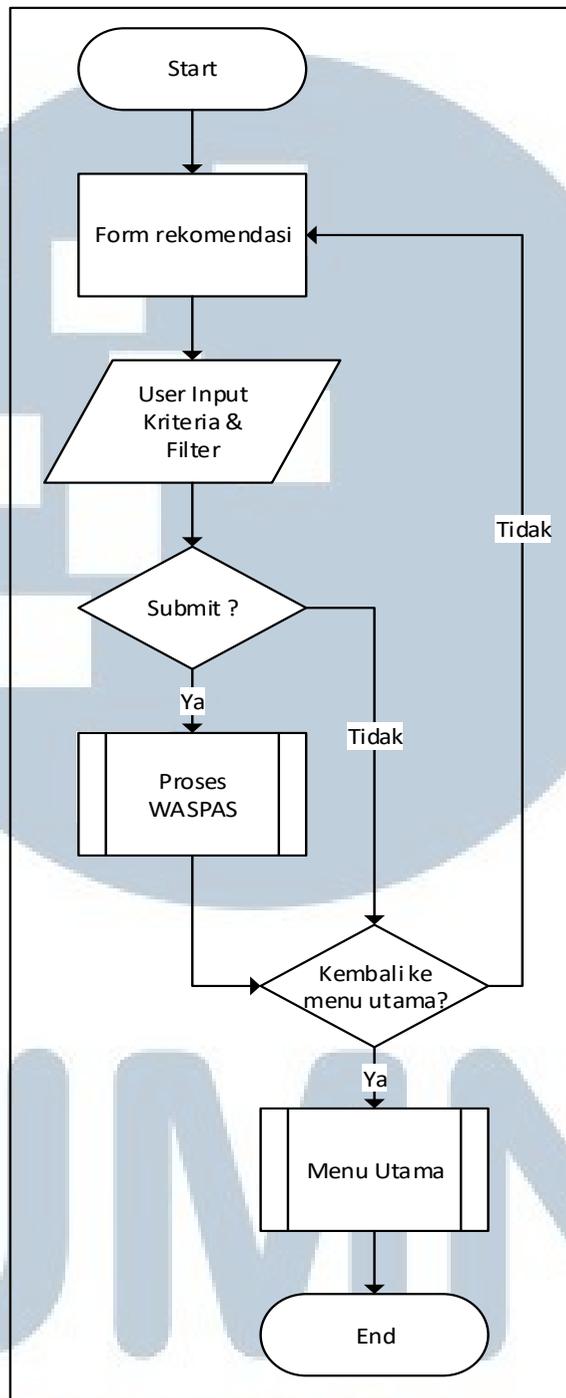
Pada gambar 3.4 menunjukkan alur proses yang dilakukan sistem ketika *user* memilih proses *about*. Halaman *about* berisi tentang penjelasan singkat tentang mengenai sistem rekomendasi pemilihan *smartphone*. *User* diberikan pilihan untuk

kembali ke *menu* utama atau tidak, jika *user* memilih untuk kembali ke *menu* utama, maka proses *menu* utama akan dilakukan.



Gambar 3.5 *Flowchart* Katalag

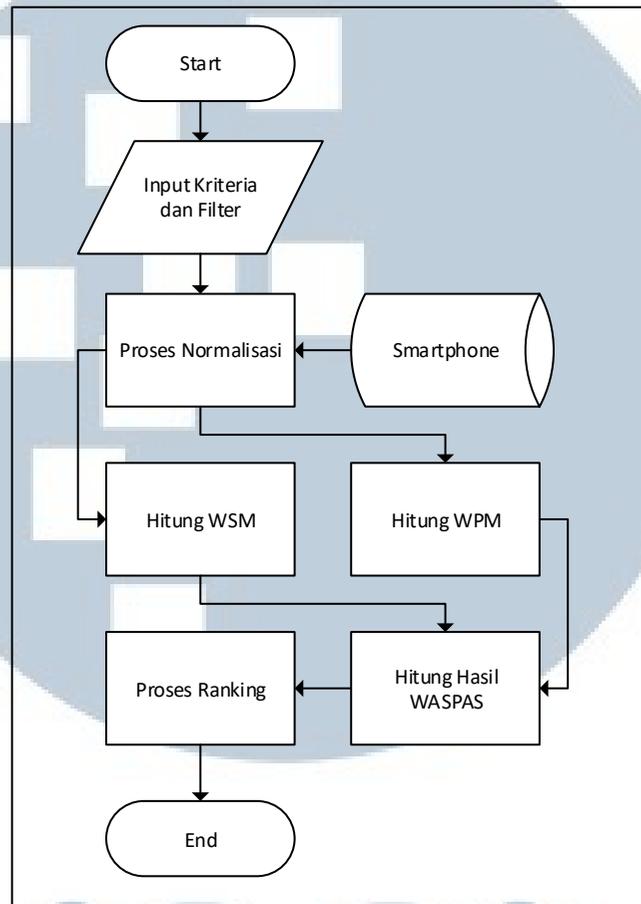
Gambar 3.5 menunjukkan alur proses yang dilakukan sistem ketika *user* memilih proses Katalog. Pada awal akan ditampilkan halaman katalog yang berisi *list data smartphone* yang diambil dari *database* dengan tabel *smartphone*. Setelah katalog ditampilkan, *user* dapat memilih apakah kembali ke *menu* utama atau tidak. Katalog sendiri berisi *list* dari *smartphone* yang akan ditampilkan secara keseluruhan dari semua merk tanpa adanya *filter*. Halaman katalog tidak diberikan paginasi sehingga *user* akan melihat semua *list* dari *smartphone* langsung pada satu halaman yang dapat digulir ke bawah.



Gambar 3.6 *Flowchart* Rekomendasi

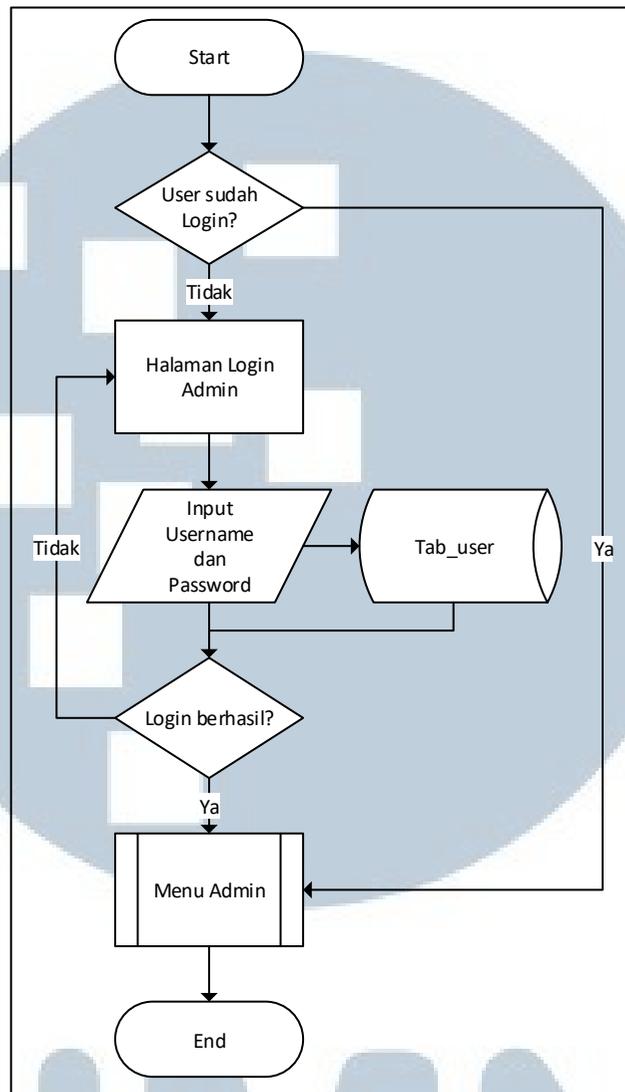
Pada gambar 3.6 digambarkan aliran proses yang dilakukan sistem ketika *user* memilih untuk melakukan proses sistem rekomendasi. Awalnya *user* akan diberikan halaman rekomendasi untuk diisi *input* bobot dan *filter* dari kriteria-kriteria yang ada, lalu kemudian dilakukan kalkulasi menggunakan metode WASPAS pada

proses WASPAS yang menghasilkan hasil rekomendasi berupa daftar *ranking* dari *smartphone* yang direkomendasikan oleh sistem.



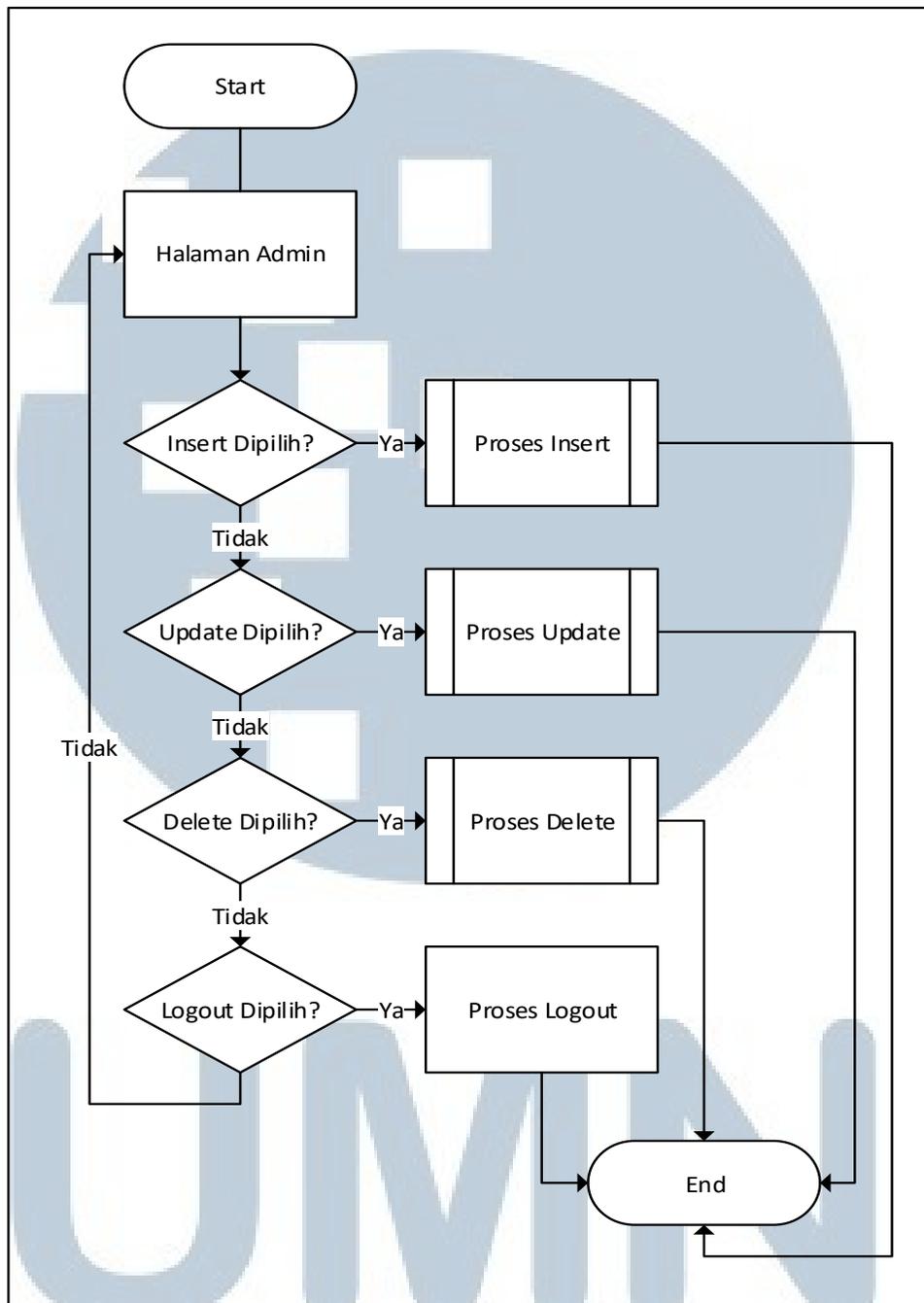
Gambar 3.7 Flowchart WASPAS

Gambar 3.7 menunjukkan alur proses pada proses WASPAS. Proses ini dimulai dengan menerima *input user* berupa nilai bobot dan *filter* dari kriteria-kriteria *smartphone*. Dari *list data smartphone* yang terdaftar pada *database*, dilakukan proses normalisasi, lalu hasil dari proses tersebut diolah lagi dengan menggunakan proses WSM dan proses WPM, lalu hasil dari masing-masing proses dimasukkan ke dalam proses hasil WASPAS, lalu hasil proses tersebut diurutkan berdasarkan nilai tertinggi di dalam proses *ranking*.



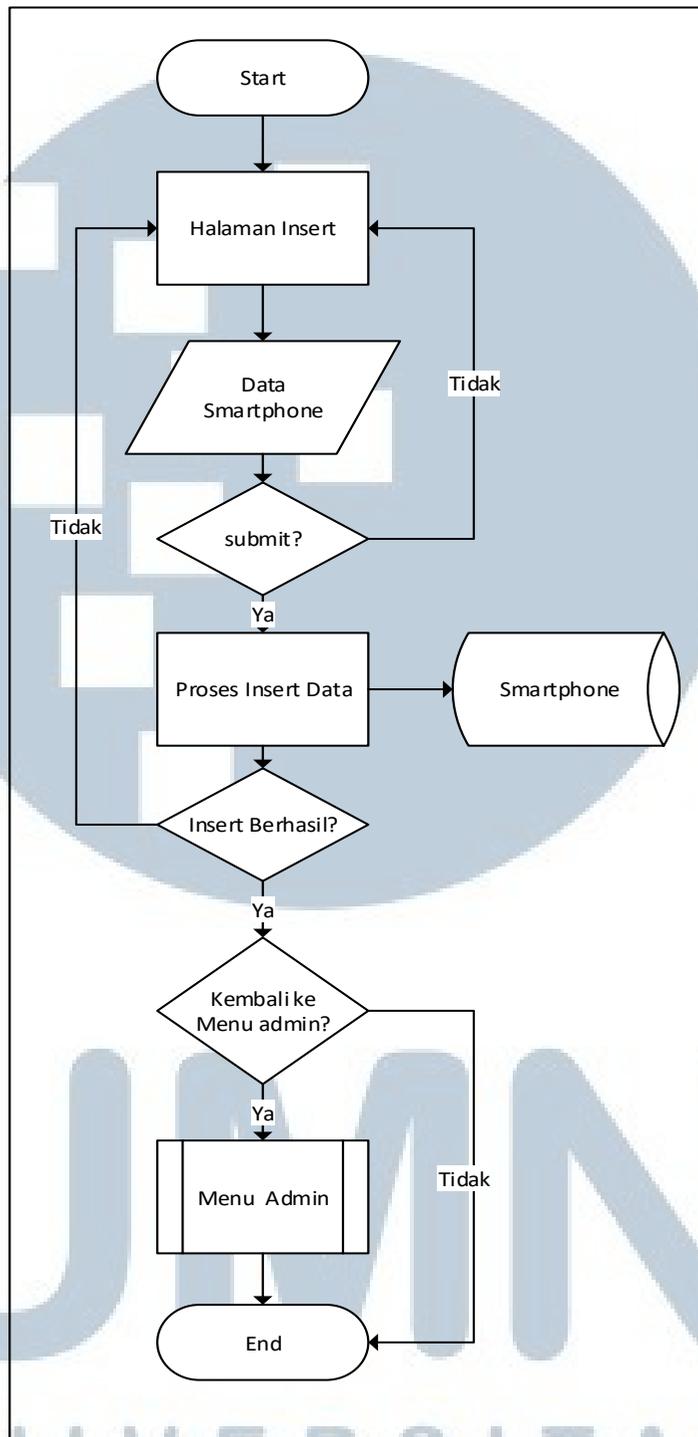
Gambar 3.8 Flowchart Login Admin

Pada gambar 3.8 menunjukkan alur proses pada *login admin*. Pada alur ini digambarkan alur *backend* saat *login* sebagai *admin*. Saat *user* memasuki halaman *login admin* atau halaman *admin*, *user* akan dicek terlebih dahulu apakah sudah *login* atau belum. Apabila sudah *login*, maka otomatis yang terbuka adalah tampilan halaman *admin*. Namun apabila belum *login*, maka otomatis yang terbuka adalah tampilan *login* di mana *user* dapat mengisi *username* dan *password*.



Gambar 3.9 Flowchart Menu Admin

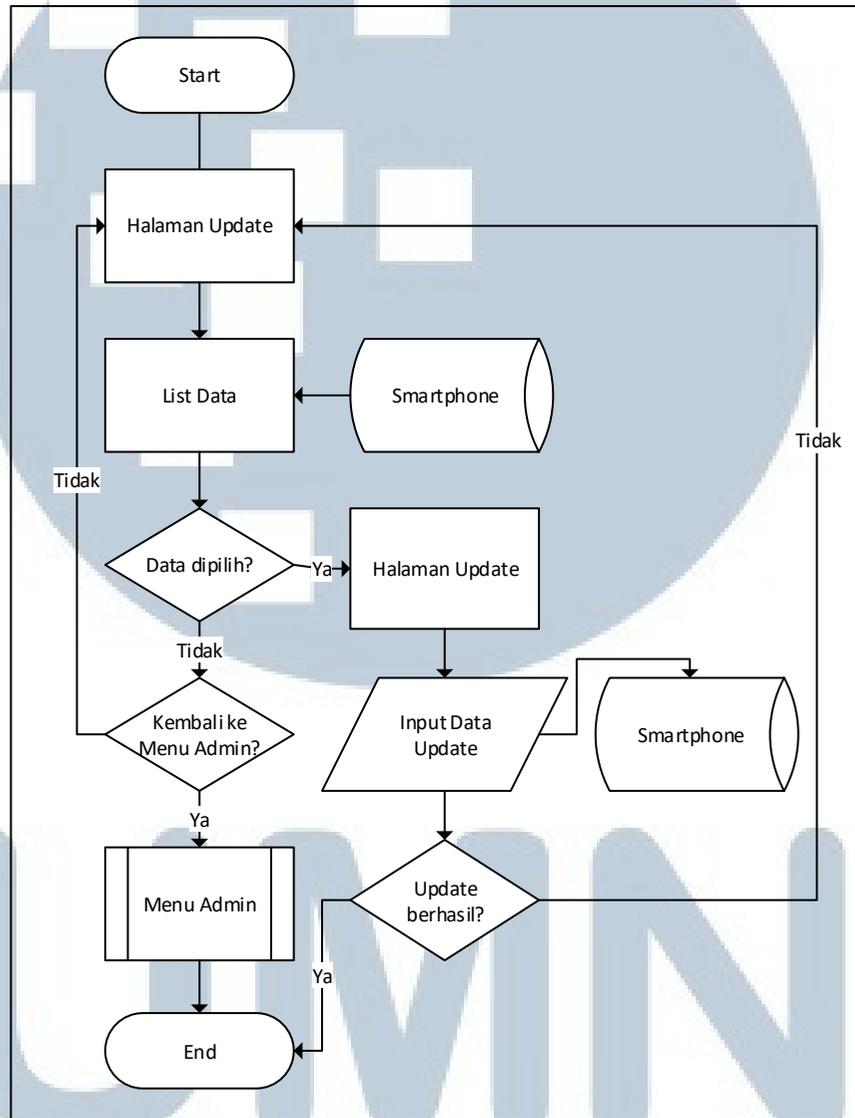
Gambar 3.9 menunjukkan alur proses pada proses *menu admin* setelah berhasil melakukan *login*. Sistem akan menampilkan halaman *admin* dimana terdapat empat proses yang dapat dipilih yaitu *insert*, *update*, *delete*, dan *logout*.



Gambar 3.10 Flowchart proses *Insert*

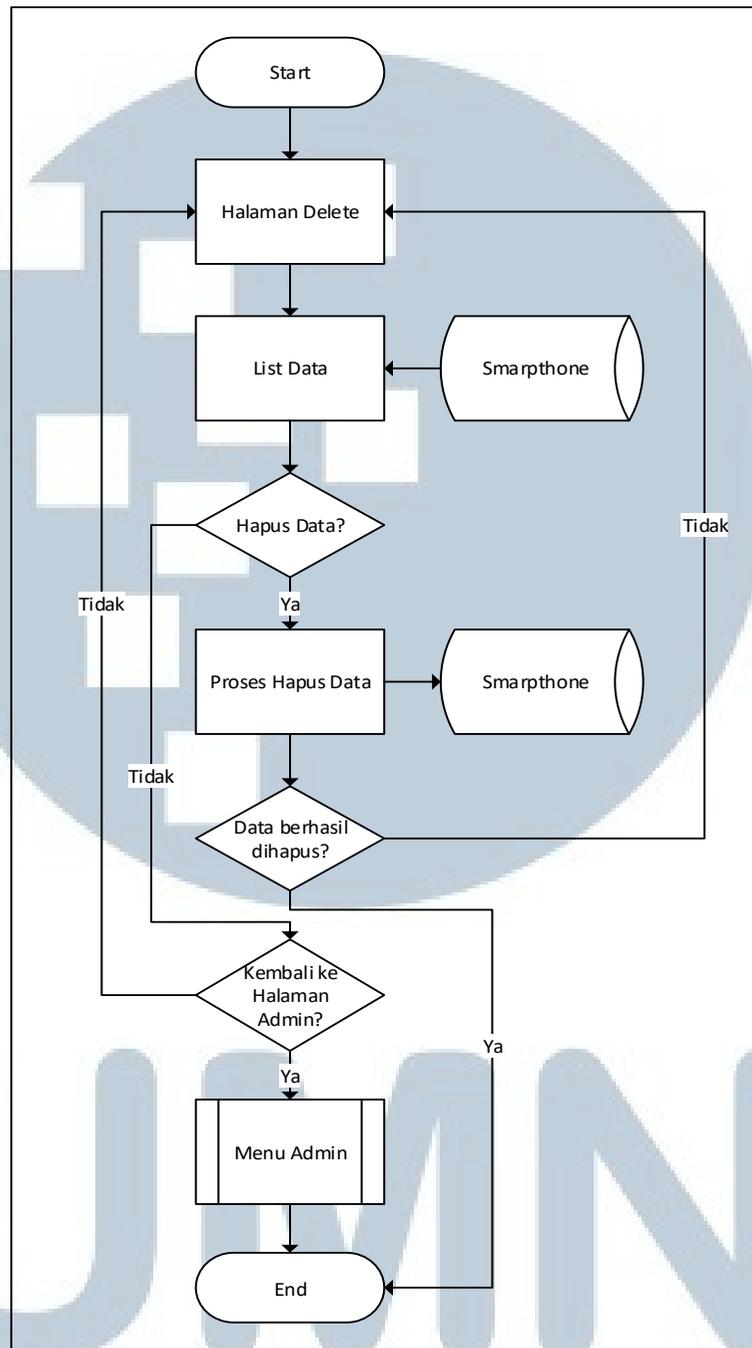
Pada gambar 3.10 ditunjukkan alur proses pada saat *admin* memilih proses *insert*. Sistem akan menampilkan halaman *insert* di mana *admin* dapat memberi *input data* yang akan dimasukkan ke dalam *database*. Setelah data dimasukkan, akan

dilakukan pengecekan apakah *data* tersebut berhasil masuk ke *database* atau gagal. Apabila gagal maka otomatis akan kembali pada halaman *insert* di mana akan diberikan notifikasi bahwa *data* gagal dimasukkan.



Gambar 3.11 *Flowchart* Proses Update

Gambar 3.11 menunjukkan alur proses pada proses *update* oleh *admin*. Pada alur pertama, halaman *update* akan diisi dengan *list data* dari *database*. Lalu setelah *data* yang diubah dipilih, masukkan *input data* dan apabila *update* berhasil, maka *data* berhasil diubah.

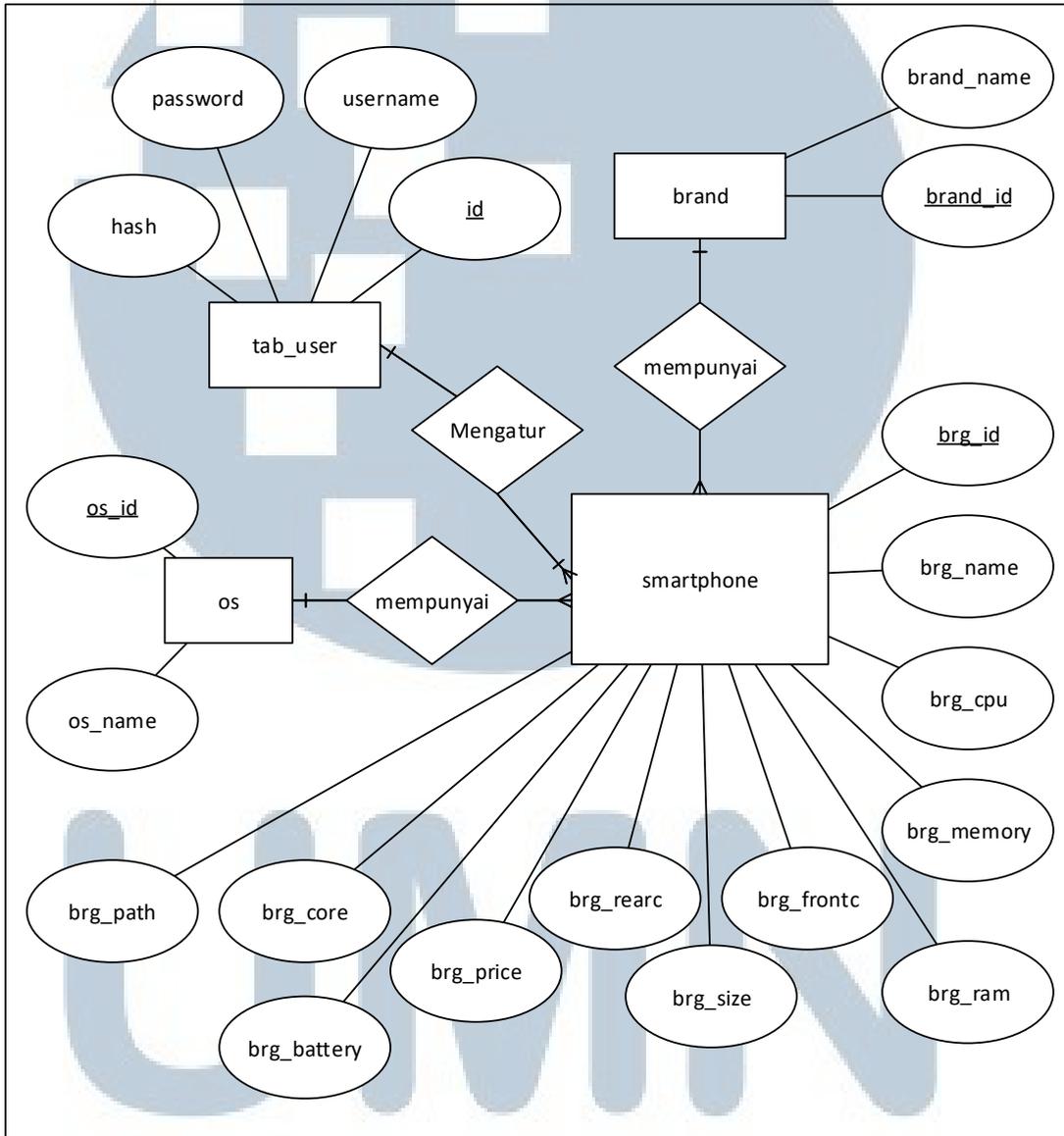


Gambar 3.12 Flowchart Proses Delete

Gambar 3.12 menunjukkan alur proses pada proses *delete* oleh *admin*. Halaman *delete* akan diisi dengan *list data* dari *database*. Lalu setelah *data* yang dihapus dipilih, maka *data* akan dihapus pada proses hapus *data* dan apabila berhasil, maka *data* berhasil dihapus.

### 3.2.3. Entity Relationship Diagram

Entity Relationship Diagram digunakan untuk menampilkan relasi antar objek pada suatu sistem.



Gambar 3.13 Entity Relationship Diagram

Pada gambar 3.13 terlihat hubungan antar entitas pada database yang tersedia, tabel tab\_user dapat mengatur data dari tabel smartphone dimana tabel smartphone adalah tabel yang berisi data smartphone, tabel os dan tabel brand adalah tabel yang memuat data yang akan dipakai oleh tabel smartphone.

### 3.2.4. Struktur Tabel

Dalam pembangunan sistem ini, terdapat empat tabel yang digunakan dalam *database*. Berikut penjabaran dari setiap tabel yang digunakan.

Nama Tabel : *brand*  
 Deskripsi : Tabel untuk menyimpan *data merk smartphone*  
 Primary Key : *brand\_id*  
 Foreign Key : -

Tabel 3.1 *Brand*

Nama Kolom	Tipe	Panjang	Keterangan
<i>brand_id</i>	Int	11	ID merk
<i>Brand_name</i>	Varchar	30	Nama merk

Nama Tabel : *smartphone*  
 Deskripsi : Tabel untuk menyimpan *data smartphone*  
 Primary Key : *brg\_id*  
 Foreign Key : *brand\_id, os\_id*

Tabel 3.2 *Smartphone*

Nama Kolom	Tipe	Panjang	Keterangan
<i>brg_id</i>	Int	11	ID Item
<i>brg_name</i>	Longtext		Nama <i>Smartphone</i>
<i>brg_cpu</i>	Float		Prosesor
<i>brg_memory</i>	Float		Memory
<i>brg_ram</i>	Float		Besar RAM item
<i>brg_size</i>	Float		Besar layar item
<i>brg_frontc</i>	Float		Besar MP kamera depan
<i>brg_rearc</i>	Float		Besar MP kamera belakang
<i>os_id</i>	Int	11	ID Sistem Operasi
<i>Brand_id</i>	Int	11	ID merk
<i>brg_price</i>	Decimal	10	Jenis harga item
<i>brg_battery</i>	Int	11	Besar daya simpan item
<i>brg_core</i>	Int	11	Harga item
<i>brg_path</i>	Longtext		Lokasi gambar item

Nama Tabel : tab\_user  
 Deskripsi : Tabel untuk menyimpan *data* admin  
 Primary Key : id  
 Foreign Key : -

Tabel 3.3 User

Nama Kolom	Tipe	Panjang	Keterangan
Id	int	11	ID admin
Username	varchar	20	Username admin
Password	varchar	30	Password admin
Hash	varchar	50	Hash dari md5 Password admin

Nama Tabel : os  
 Deskripsi : Tabel untuk menyimpan *data* sistem operasi  
 Primary Key : os\_id  
 Foreign Key : -

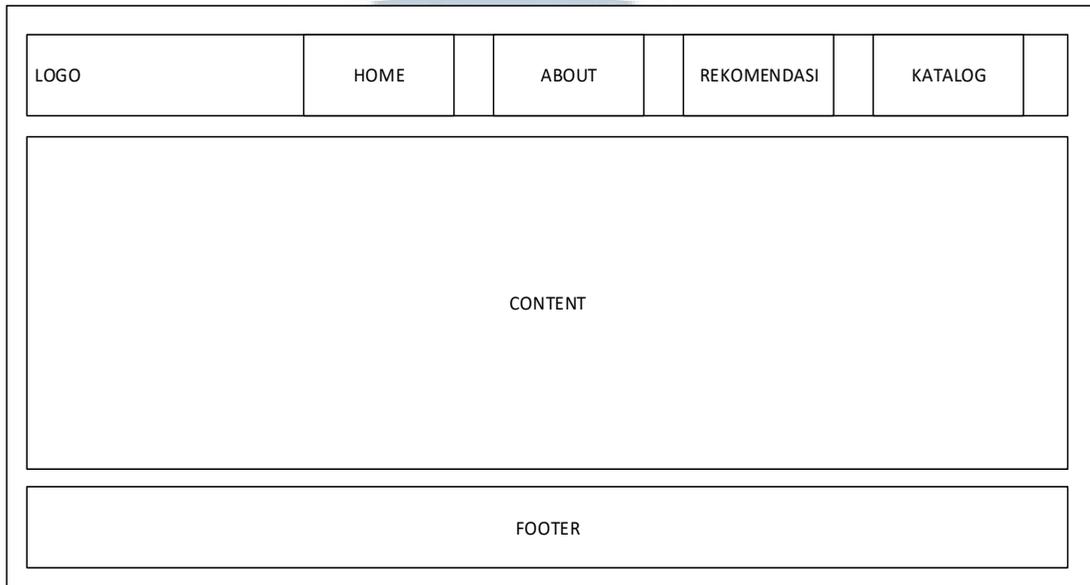
Tabel 3.4 Sistem Operasi

Nama Kolom	Tipe	Panjang	Keterangan
os_id	int	11	ID OS
os_name	varchar	25	Nama OS

### 3.2.5. Rancangan Antar Muka

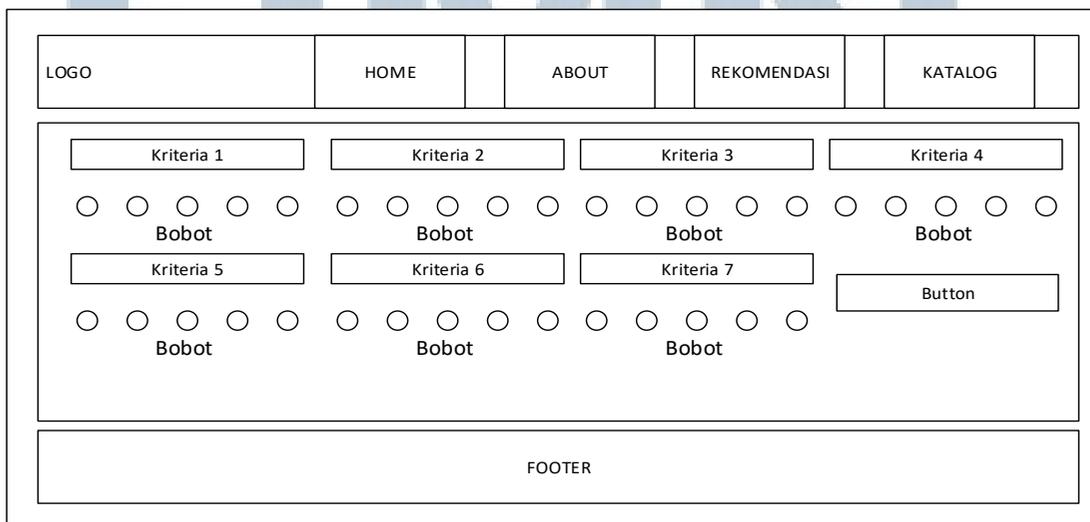
Sistem ini terbagi menjadi dua bagian, yaitu sistem *frontend* dan *backend*, dimana setiap sistem memiliki antarmuka yang berbeda. Sistem *frontend* digunakan sebagai tempat untuk melihat katalog *smartphone*, melihat halaman tentang, dan memberikan nilai bobot dan *filter* untuk mendapatkan hasil rekomendasi *smartphone*, sedangkan sistem *backend* digunakan sebagai tempat administrasi *data* dari *smartphone*, mulai dari *login* sebagai *admin*, memasukkan *data smartphone*, memperbaharui *data smartphone*, dan juga menghapus *data smartphone*.

## A. Sistem Frontend



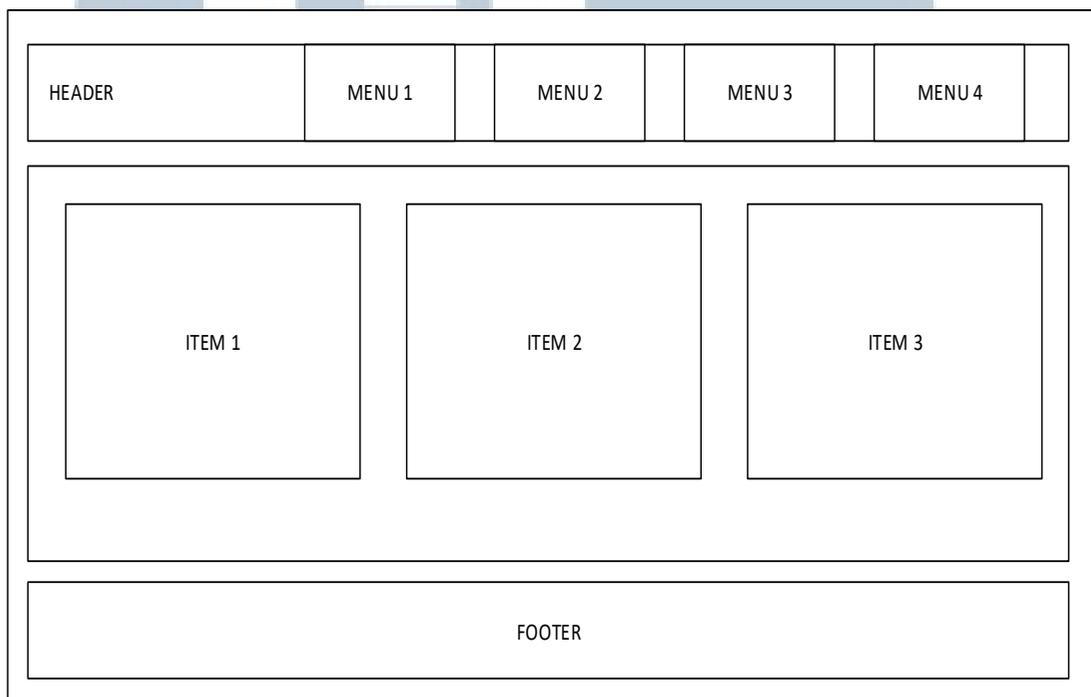
Gambar 3.14 Sketsa Tampilan Awal *Frontend*

Sketsa tampilan awal pada sistem *frontend* dapat dilihat pada gambar 3.14, dimana secara keseluruhan tampilan dibagi menjadi tiga bagian, yaitu bagian *header*, *footer* dan *content*. *Header* sendiri memuat empat *menu* utama, yaitu *Home* (halaman indeks), *About* (halaman tentang), *Recommend* (halaman rekomendasi), dan *Catalog* (halaman katalog). Pada bagian *content* akan diberikan yang menampilkan beberapa *smarphone* yang menjadi fitur. Pada bagian *footer* menampilkan logo.



Gambar 3.15 Sketsa Tampilan *Input*

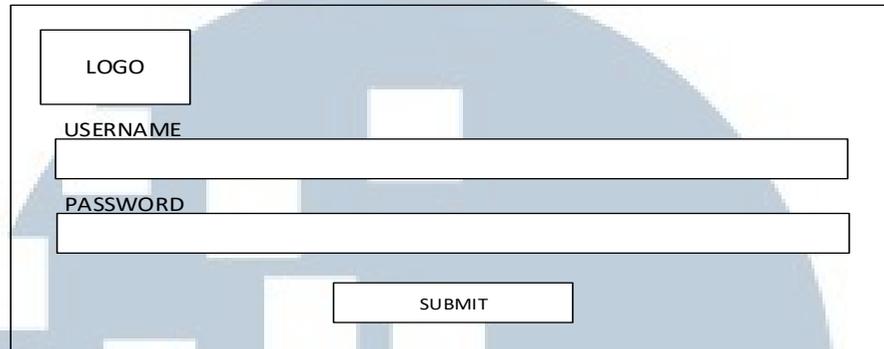
Pada halaman *input* kriteria, bagian *content* terbagi menjadi *combo box* dan *radio button* untuk memilih kriteria dan bobot. Akan terdapat tiga belas macam *input* yaitu tujuh *filter* kriteria dan enam pembobotan. *Data input* dalam pembobotan masih kosong namun harus diisi. *User* bisa mengubah *value* tersebut menjadi 1 sampai 5 tergantung keinginan *user*. Semua bobot harus diisi karena akan berpengaruh terhadap hasil rekomendasi yang akan ditampilkan.



Gambar 3.16 Sketsa Tampilan Hasil Rekomendasi

Setelah *user* memasukkan *data filter* kriteria dan bobot kriteria dari *smartphone* yang diinginkan, sistem akan menampilkan rekomendasi *smartphone* ke dalam tampilan *item*. *Item* akan menampilkan nama sekaligus gambar item dari *smartphone* rekomendasi. Pada setiap *item* terdapat kolom-kolom untuk menampilkan detail dari *smartphone*.

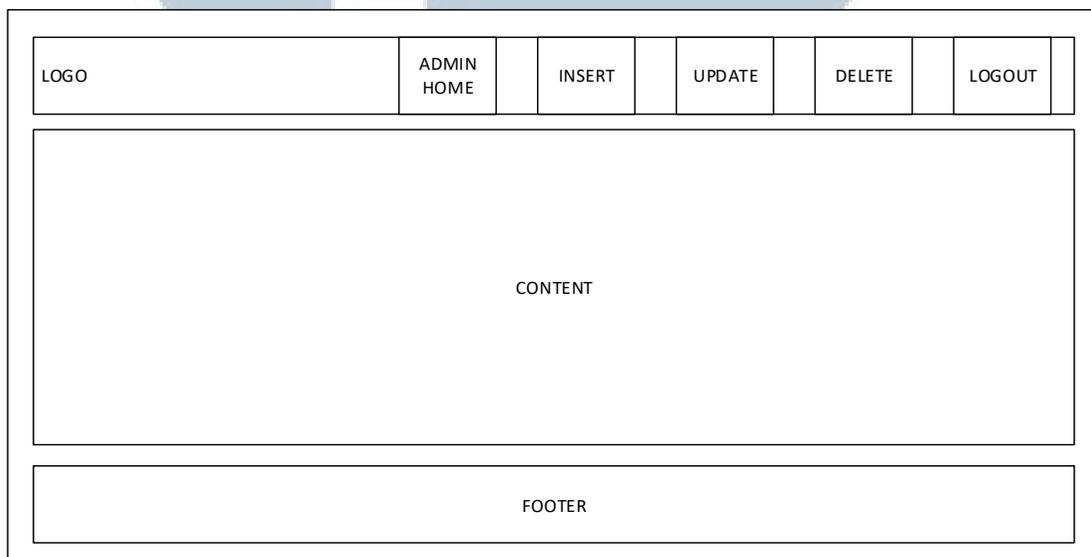
## B. Sistem Backend



A wireframe sketch of a login form. It consists of a rectangular container with a border. Inside, there is a small box labeled 'LOGO' in the top left corner. Below it are two horizontal input fields: the first is labeled 'USERNAME' and the second is labeled 'PASSWORD'. At the bottom center of the container is a rectangular button labeled 'SUBMIT'.

Gambar 3.17 Sketsa *Form Login*

Sistem *backend* dapat diakses apabila *user* terlebih dahulu melakukan login. Gambar 3.17 merupakan sketsa tampilan *form login*, dimana *user* diminta memasukkan *username* dan *password* terlebih dahulu.



A wireframe sketch of an admin home page layout. It is a large rectangular container with a border, divided into three main sections. The top section is a horizontal header bar containing several buttons: 'LOGO', 'ADMIN HOME', 'INSERT', 'UPDATE', 'DELETE', and 'LOGOUT'. The middle section is a large, empty rectangular area labeled 'CONTENT'. The bottom section is a horizontal bar labeled 'FOOTER'.

Gambar 3.18 Sketsa Tampilan *Home Admin*

Pada *backend*, tampilan terbagi menjadi tiga bagian. *Header* memuat logo, tombol untuk halaman *admin home*, tombol untuk halaman *insert*, tombol untuk halaman *update*, tombol untuk halaman *delete*, dan tombol *logout*, bagian tengah untuk *content*, bagian bawah untuk *footer*. Apabila *user* belum *login*, maka tampilan awal

berubah menjadi tampilan *login* seperti pada gambar 3.17, namun apabila sudah melakukan *login*, maka akan otomatis dipindahkan ke halaman *home admin*.

LOGO	ADMIN HOME	INSERT	UPDATE	DELETE	LOGOUT
------	------------	--------	--------	--------	--------

NAMA	<input type="text"/>
MERK	<input type="text"/>
PROSESOR	<input type="text"/>
CPU	<input type="text"/>
RAM	<input type="text"/>
MEMORY	<input type="text"/>
KAMERA	<input type="text"/>
UKURAN	<input type="text"/>
HARGA	<input type="text"/>
BATERAI	<input type="text"/>
OS	<input type="text"/>
GAMBAR	<input type="text"/>
	PILIH FILE
INSERT DATA	

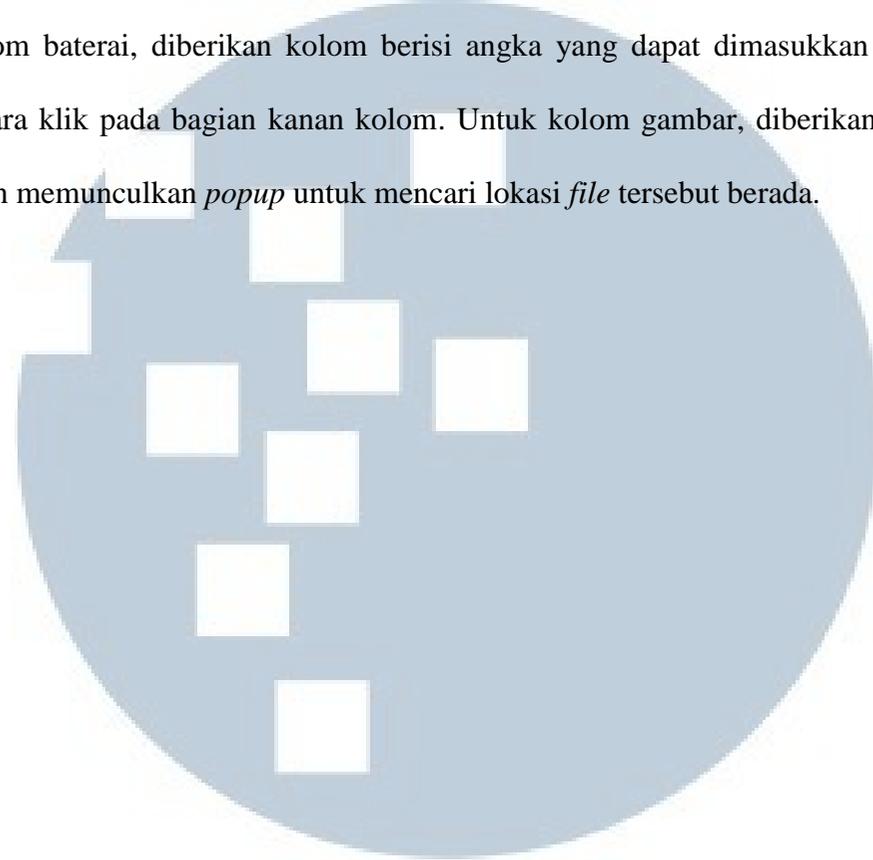
  

FOOTER
--------

Gambar 3.19 Sketsa Tampilan *Insert*

Pada gambar 3.19 terlihat sketsa untuk tampilan halaman *insert*, di mana *user* memasukkan *data* sesuai kolom yang tersedia, yaitu kolom nama, kolom merk, kolom prosesor, kolom CPU, kolom RAM, kolom memory, kolom kamera, kolom ukuran, kolom harga, kolom *baterai*, kolom OS, dan kolom gambar. Untuk kolom nama, diberikan kolom text yang bisa diisi sesuai nama text pada *smartphone* yang ingin dimasukkan. Untuk kolom merk dan OS, diberikan pilihan berdasarkan jenis-jenis

yang ada. Untuk kolom prosesor, CPU, RAM, *memory*, kamera, ukuran, harga, dan kolom baterai, diberikan kolom berisi angka yang dapat dimasukkan manual atau secara klik pada bagian kanan kolom. Untuk kolom gambar, diberikan kolom yang akan memunculkan *popup* untuk mencari lokasi *file* tersebut berada.



# UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA