

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Metode Penelitian yang digunakan adalah sebagai berikut:

1. Studi Literatur

Tahap ini dilakukan pembelajaran tentang metode MOORA. Selain itu dilakukan pembelajaran tentang hal-hal yang berhubungan dengan sistem rekomendasi. Sumber penelitian dapat diperoleh dari penelitian yang sudah dilakukan sebelumnya, jurnal, *thesis paper*, buku, maupun artikel yang relevan.

2. Analisis kebutuhan

Tahap ini melakukan analisis terhadap kebutuhan yang diperlukan dalam sistem seperti fitur dari aplikasi yang meliputi kriteria yang dibutuhkan oleh sistem untuk melakukan rekomendasi seperti harga yang akan ditampilkan, gambar yang disesuaikan dengan deskripsi, algoritma yang digunakan untuk menjalankan rekomendasi, dan seterusnya.

3. Perancangan Sistem

Tahap ini dirancang *Flowchart*, *Data Flow Diagram (DFD)* dan *Entity Relationship Diagram (ERD)* yang bertujuan untuk membantu proses perancangan sistem. Kemudian dilakukan perancangan *UI (user integererface)*.

4. Implementasi

Setelah dilakukan perancangan sistem dan *user integererface*, kemudian dilakukan proses penulisan kode menggunakan bahasa pemrograman *PHP* dan mengimplementasikan metode MOORA pada sistem aplikasi untuk pengambilan keputusan.

5. Uji Coba (*Testing*)

Tahap ini menguji aplikasi yang dibangun menggunakan skenario yang telah dibuat. Pengujian dilakukan untuk meminimalkan kesalahan dan mendapatkan hasil yang akurat..

6. Evaluasi dan Kesimpulan

Tahap ini mengevaluasi aplikasi yang dirancang berdasarkan umpan balik yang diberikan pengguna melalui kuesioner. Hasil *input* pengguna menjadi kriteria teknis keberhasilan sistem. Kesimpulan penelitian akan diambil berdasarkan hasil evaluasi.

3.2 Perancangan Sistem

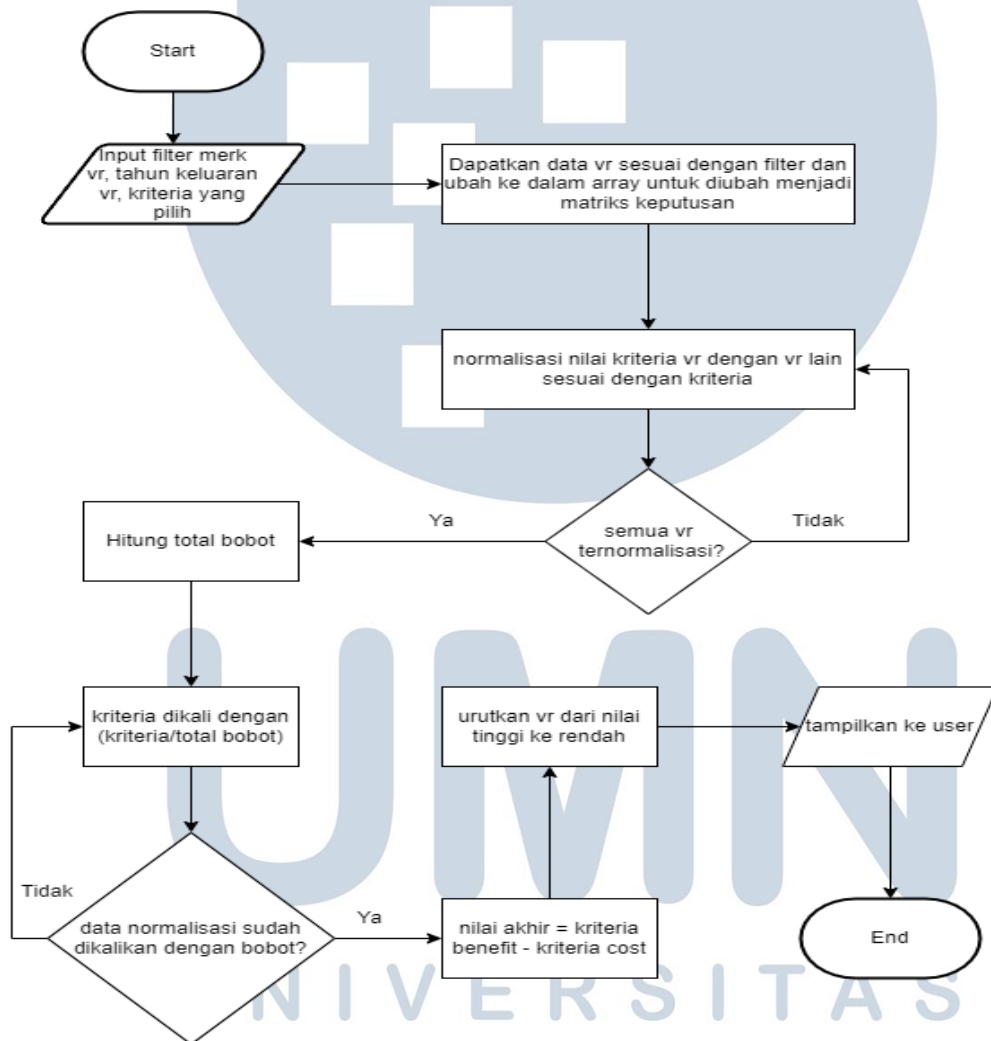
Pada Tahap ini dibuat flowchart sebagai gambaran fungsionalitas dan proses aplikasi yang dibangun. Diagram aliran data (DFD) kemudian dibuat, yang memberikan gambaran pertukaran data dalam sistem. Diagram aliran data dibagi menjadi 2 level, yaitu DFD level 1 dan DFD level 0. Selanjutnya, buatlah diagram hubungan entitas (ERD) untuk memberikan gambaran hubungan antar entitas dengan entitas lainnya.

3.2.1 Flowchart

Flowchart adalah sebuah representasi grafis dari suatu proses yang menunjukkan langkah-langkah secara berurutan menggunakan berbagai simbol. Simbol tersebut dihubungkan dengan garis yang menunjukkan arah aliran proses. *Flowchart* terdiri dari dua bagian yaitu untuk pengguna (*front-end*) dan untuk administrator (*back-end*). *Flowchart front-end* menjelaskan apa yang dilakukan oleh pengguna, seperti menjalankan sistem rekomendasi, melihat katalog,

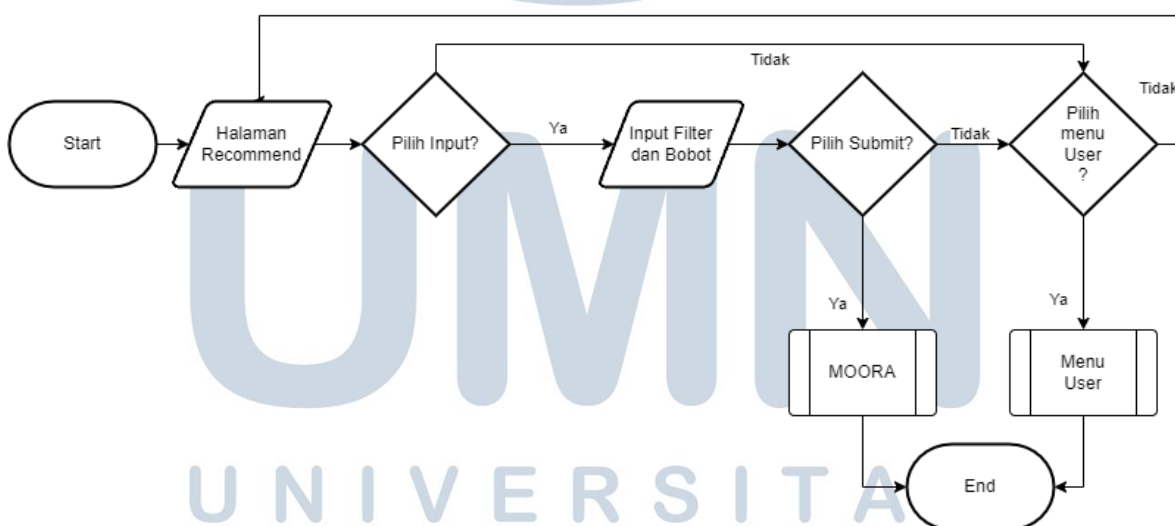
membaca halaman *About Us*. Sedangkan *Flowchart* untuk *back-end* menjelaskan pekerjaan administrator, seperti melakukan penambahan, pembaharuan dan pengurangan data yang digunakan oleh aplikasi.

Pengolahan *database* terdiri dari 3 fungsi yaitu *insert* digunakan untuk menambah data pada database, *update* digunakan untuk memperbaharui data pada database dan *delete* digunakan untuk menghapus data pada database.



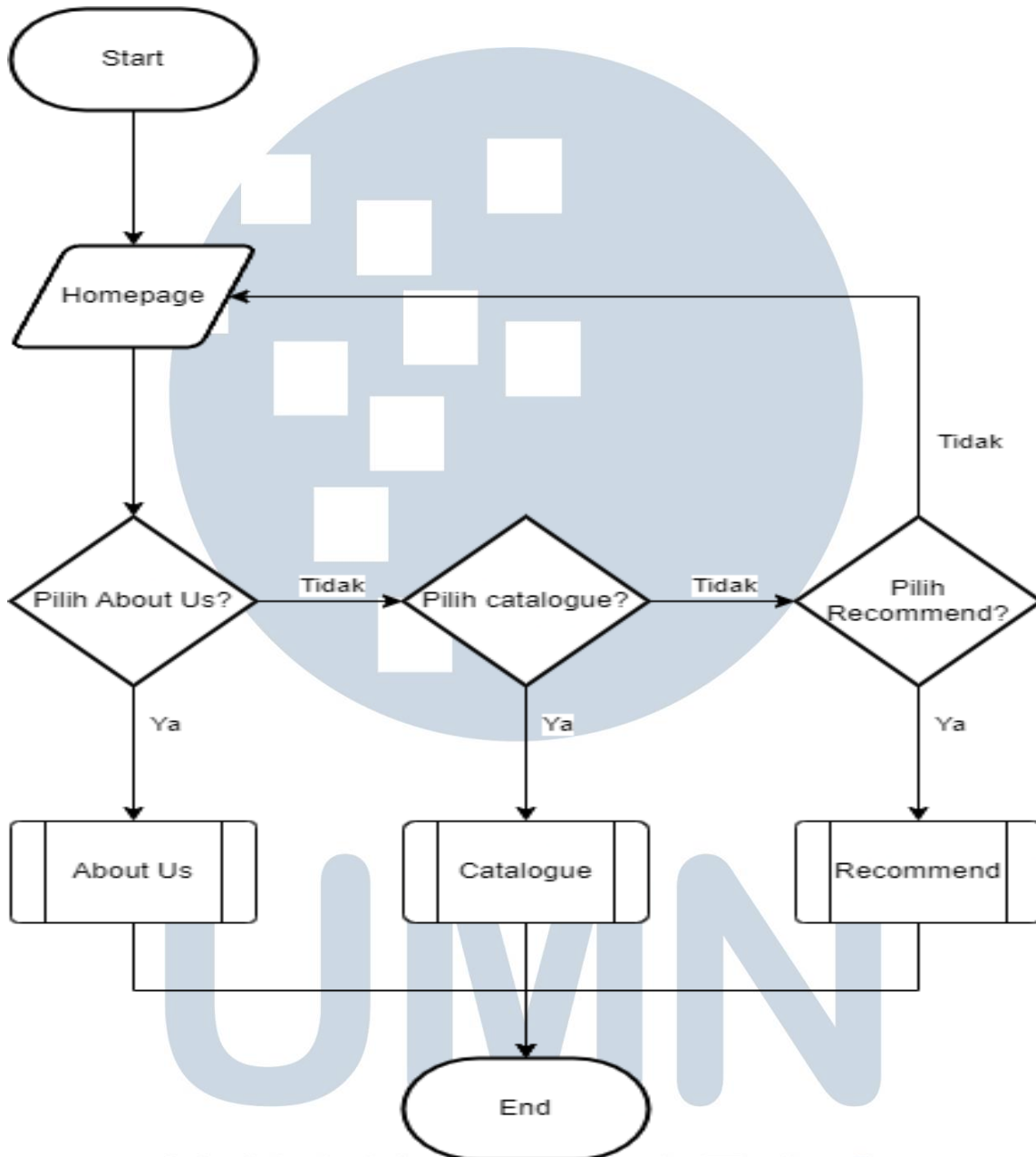
Gambar 3.1 *Flowchart* Metode MOORA

Gambar 3.1 menunjukkan alur proses metode MOORA. Prosesnya diawali dengan memasukkan input berupa preferensi merek *vr headset*, tahun rilis *vr Headset*, dan bobot kriteria. Matriks keputusan dibuat dari data *vr headset* yang terdaftar di dalam database. Setelah matriks keputusan dihasilkan, nilai referensi dari *vr headset* ditentukan terhadap nilai referensi dari *vr headset* lainnya sesuai dengan jenis referensinya. Kemudian menghitung total nilai tertimbang dengan mengalikan setiap kriteria dengan setiap bobot dan membaginya dengan total untuk mendapatkan produk tertimbang. Kemudian nilai harga dikurangkan dari nilai keuntungan, dan keuntungan dihitung sebagai jumlah RAM, berat, Refresh Rate dan resolusi gambar *vr headset*, dan harga tersebut dihitung ke dalam harga. Langkah selanjutnya adalah menghitung nilai akhir dengan mengurangkan basis biaya dari basis keuntungan. Langkah selanjutnya adalah mengurutkan *vr headset* dari nilai akhir tertinggi hingga terendah.



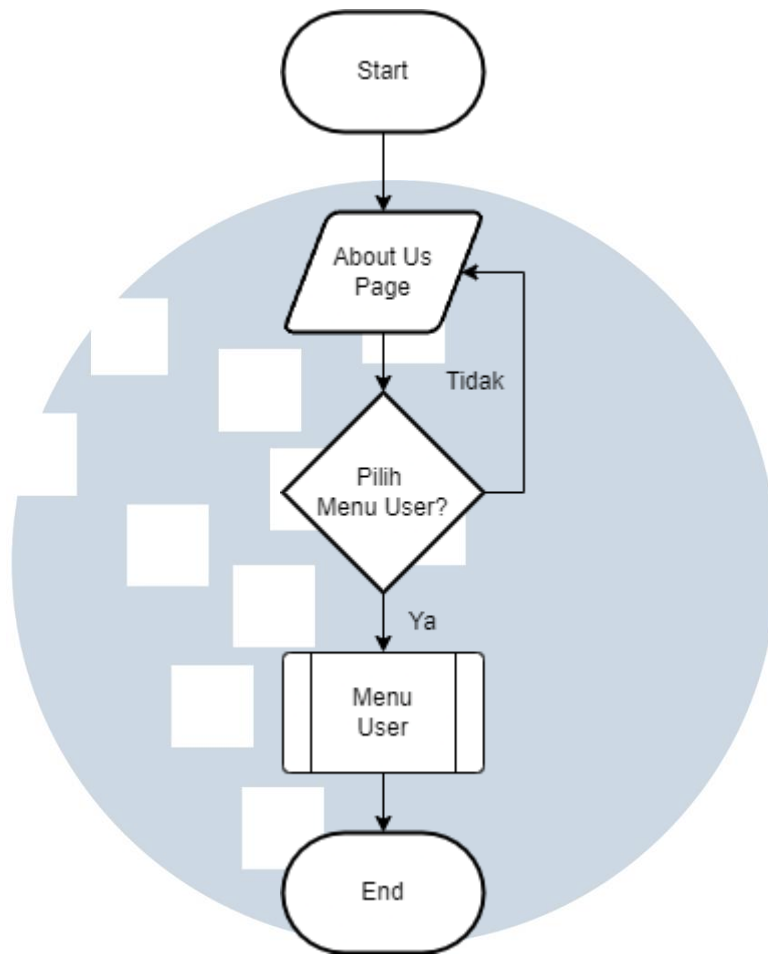
Gambar 3.2 Flowchart Recommendation

Gambar 3.2 menunjukkan alur kerja pengguna dalam proses sistem rekomendasi. Pertama, pengguna memasukkan bobot nilai referensi, kemudian sistem melakukan perhitungan menggunakan metode MOORA, yang menghasilkan hasil pengujian berupa daftar *vr headset* yang direkomendasikan.



Gambar 3.3 Flowchart User Menu

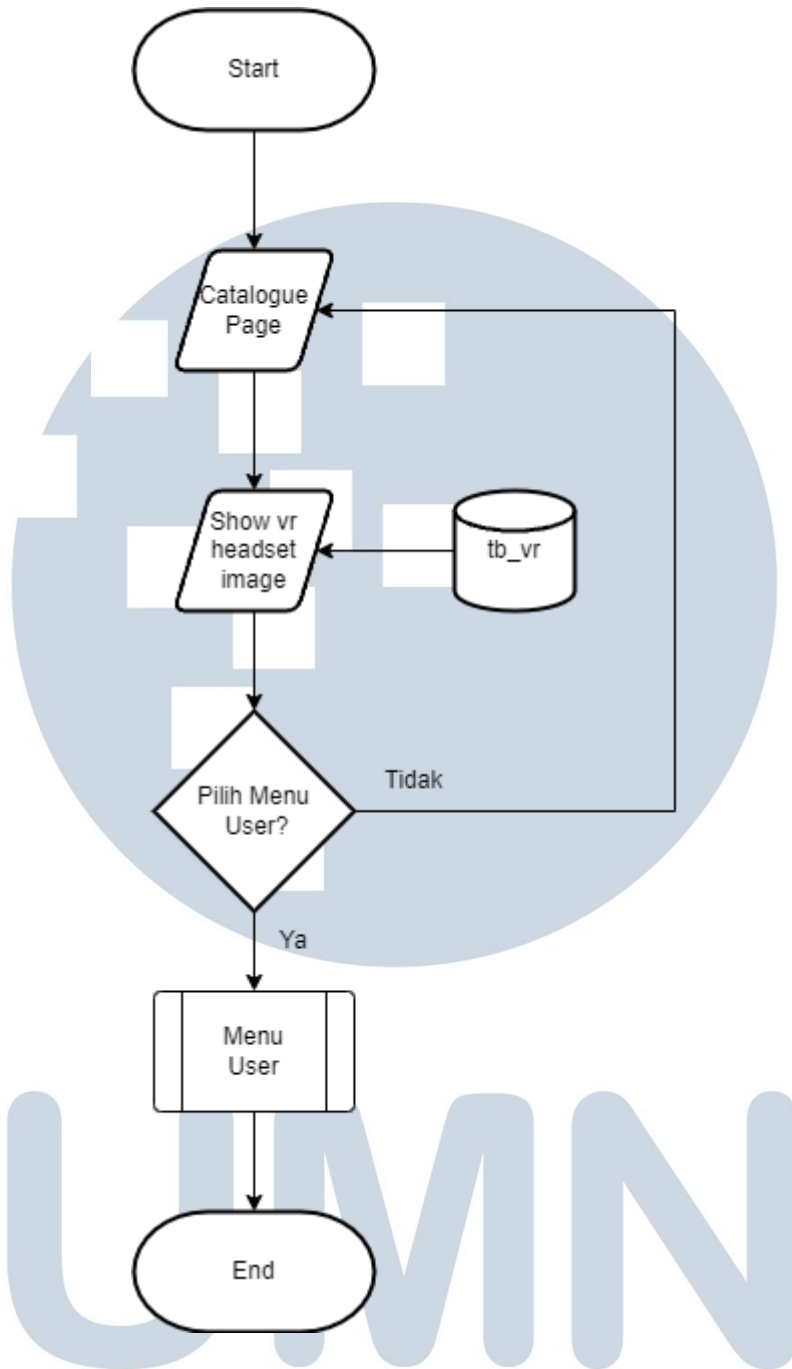
Gambar 3.3 menjelaskan aliran pada *frontend* yang diawali dengan menampilkan *homepage*. *Homepage* memiliki 3 pilihan yaitu membuka *About Us* membuka *Catalogue*, atau membuka *Recommend*.



Gambar 3.4 *Flowchart About Us*

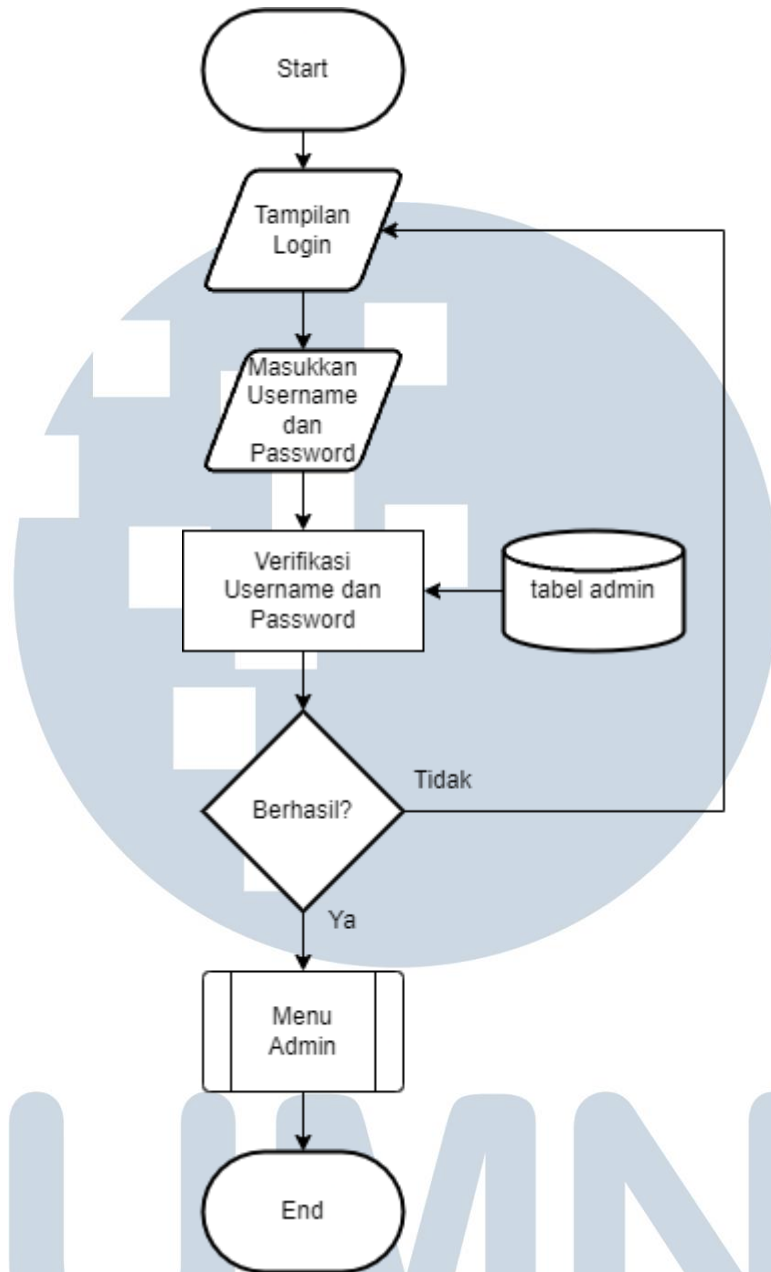
Gambar 3.4 menjelaskan aliran pengguna ketika memilih *About Us*. *About Us* berisi penjelasan singkat mengenai *Website* rekomendasi *vr headset*.

U M M N
UNIVERSITAS
MULTIMEDIA
NUSANTARA



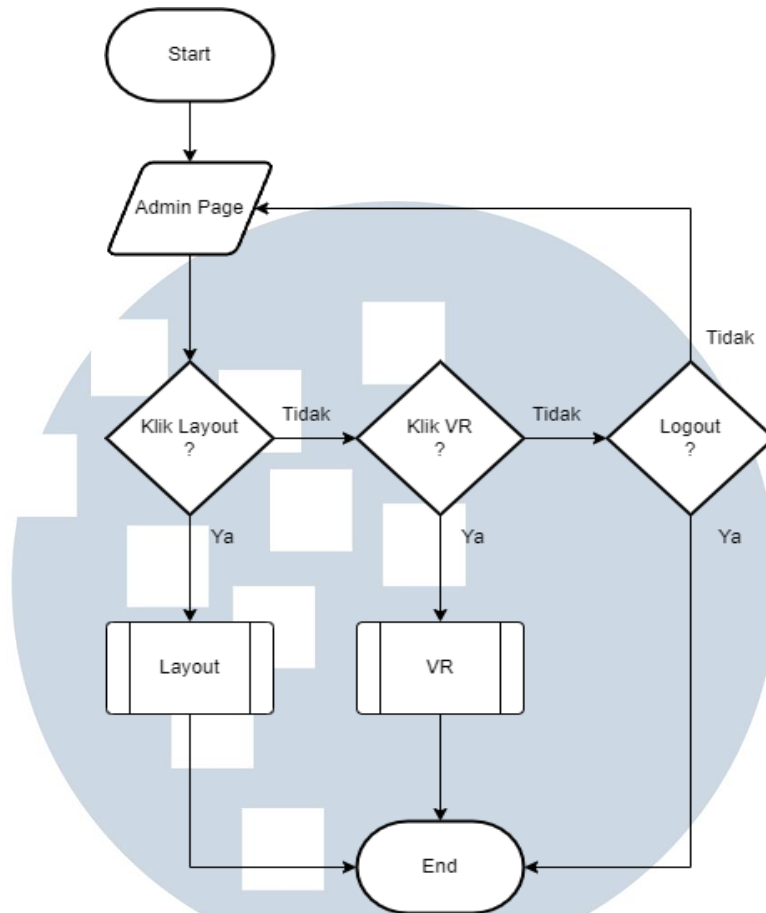
Gambar 3.5 *Flowchart* Halaman Catalogue

Gambar 3.5 Menjelaskan aliran pengguna dalam memilih *catalogue*. Awalnya ditampilkan *catalogue* yang berisi item dari *vr headset*. Lalu pengguna dapat melakukan *filtering* item *vr headset* apa saja yang ingin ditampilkan berdasarkan *Brand* dan model. *User* juga dapat memilih menu sebelumnya seperti *Homepage* dan *About Us*.



Gambar 3.6 *Flowchart Login Administrator*

Gambar 3.6 menjelaskan aliran administrator ketika *login*. Sistem menampilkan *login*, kemudian administrator memasukkan *username* dan *password*, jika berhasil administrator akan masuk ke dalam *backend* dari aplikasi.



Gambar 3.7 *Flowchart* Menu Administrator

Gambar 3.7 menjelaskan aliran setelah administrator *login*. Di awal, sistem akan menunjukkan halaman administrator. Kemudian administrator dapat memilih 2 halaman lainnya untuk mengolah *database vr headset*, yaitu:

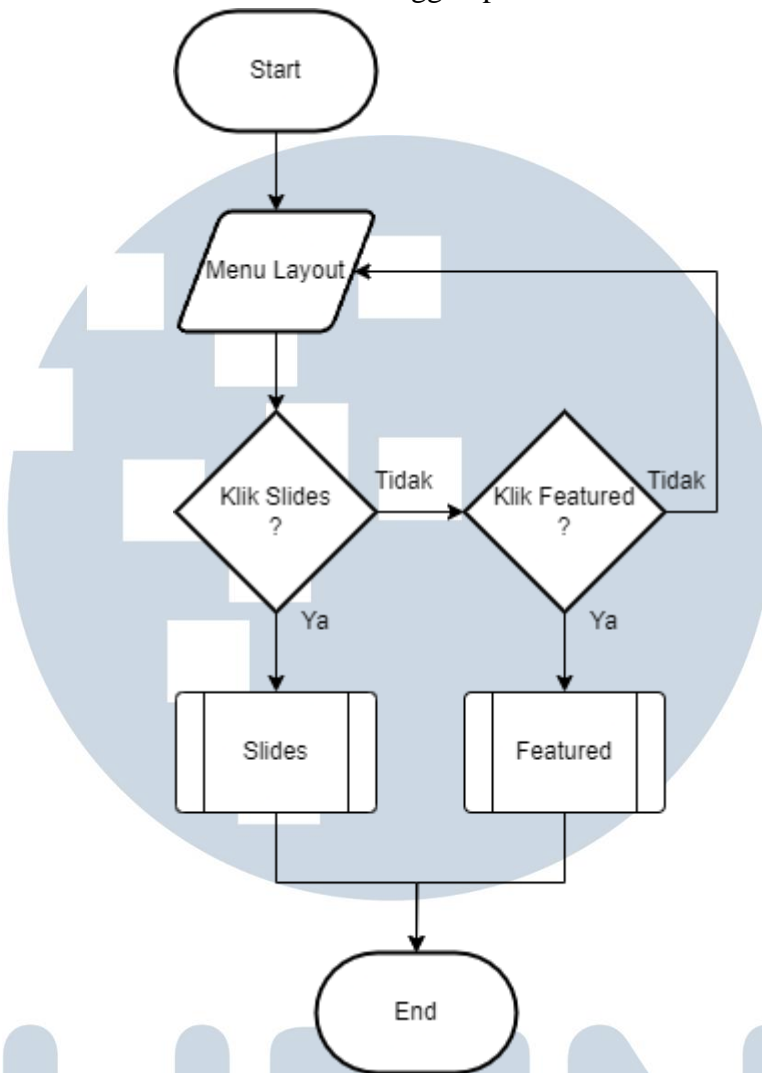
1. *Layout*

Halaman untuk menggarap gambar yang ada pada halaman *Homepage*.

2. Halaman *Vr headset*

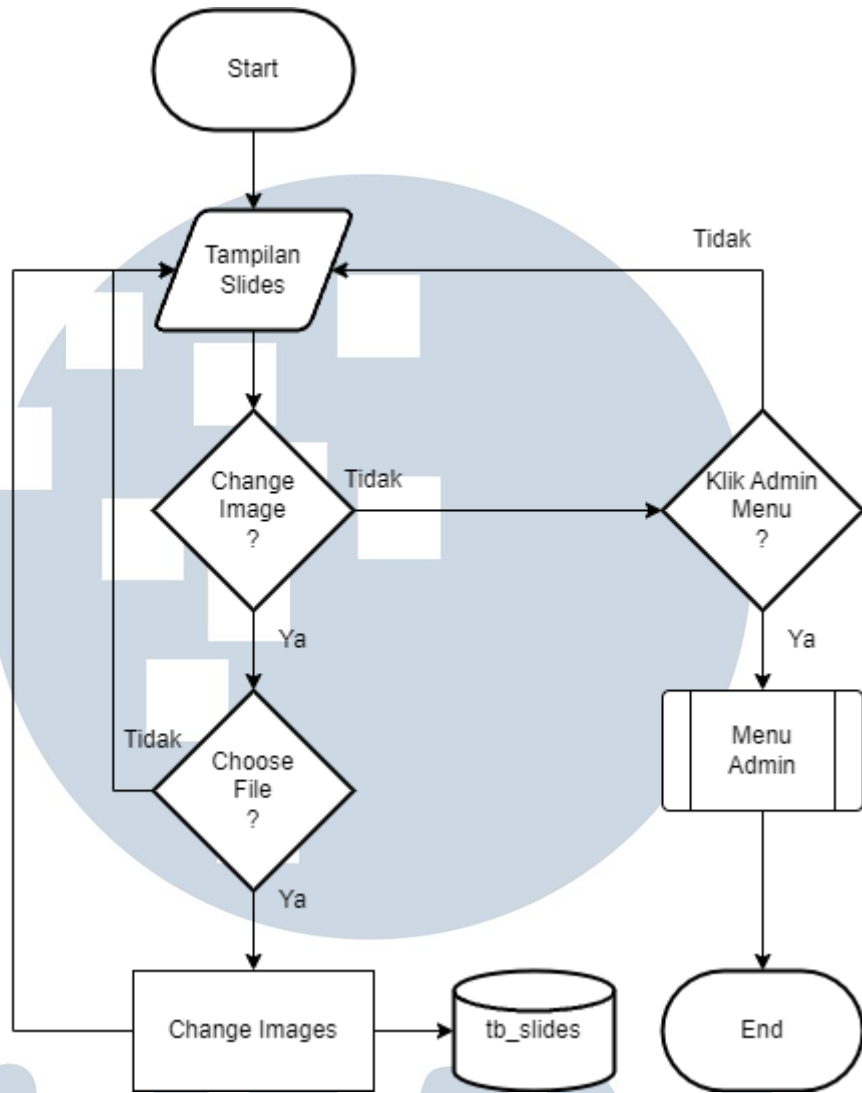
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Halaman *vr headset* untuk menggarap data *vr headset*.



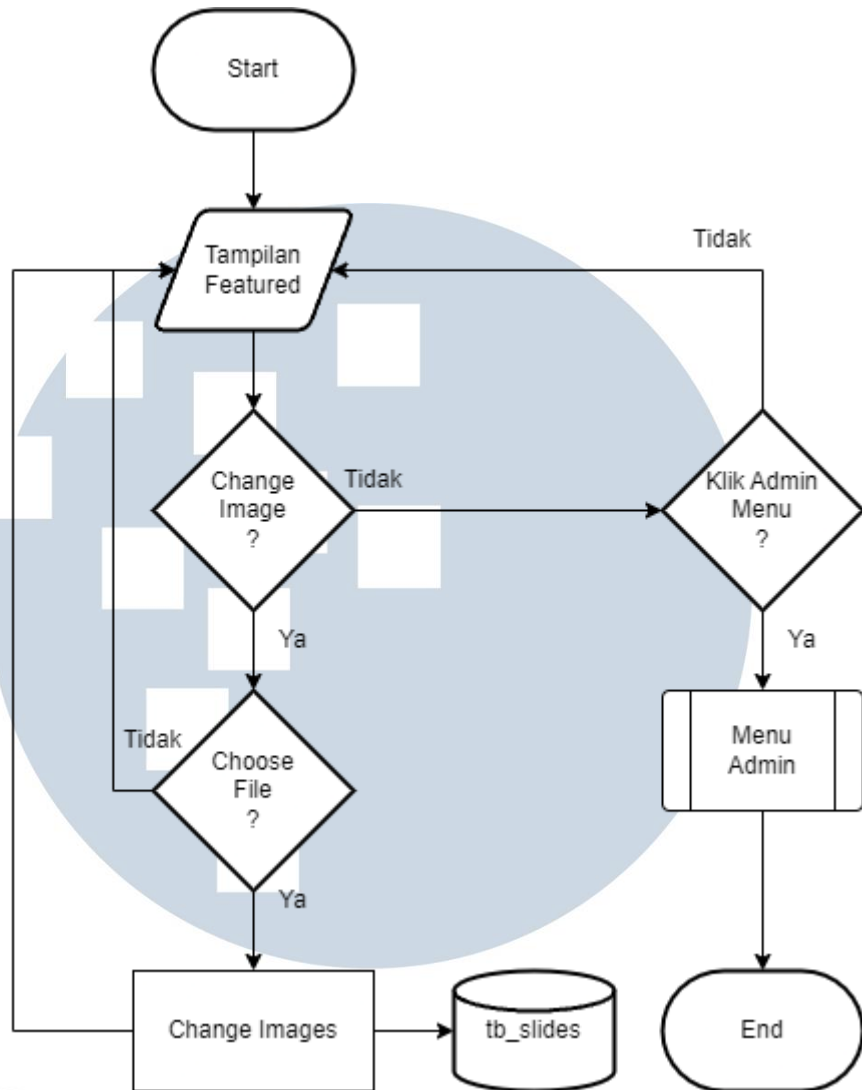
Gambar 3.8 *Flowchart Menu Layout*

Gambar 3.8 menjelaskan aliran saat memilih *layout*. Halaman *slides* yang digunakan untuk mengolah *image slides* pada *frontend* di halaman *Homepage* dan *Featured vr headset* yang digunakan untuk mengolah *bottom image* pada *frontend* di halaman *home*.



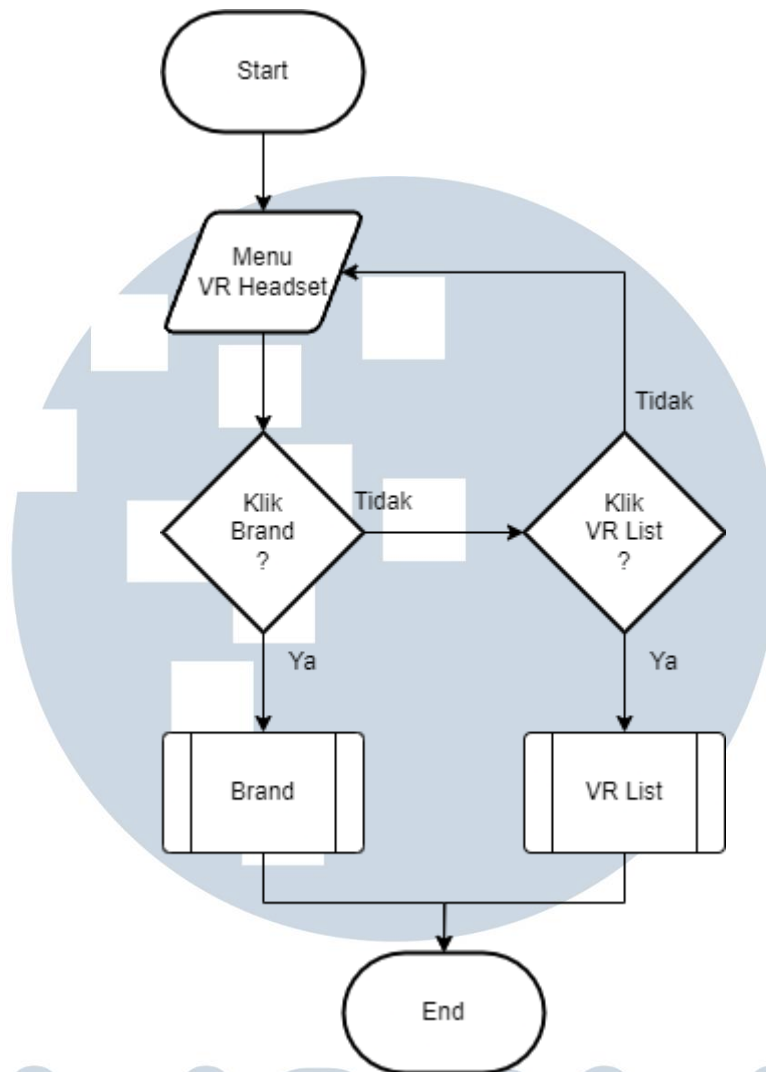
Gambar 3.9 Flowchart Menu Slides

Gambar 3.9 menjelaskan aliran saat administrator memilih menu *slides*. Pertama ditampilkan halaman *slideshow* yang dapat merubah *image slides* pada Bentuk Halaman *home* di *frontend*, kemudian administrator dapat melakukan perubahan gambar dengan melakukan *input* gambar yang diperlukan. Setelah *input* gambar, gambar tersebut disimpan ke *database* *vr headset*.



Gambar 3.10 *Flowchart Featured*

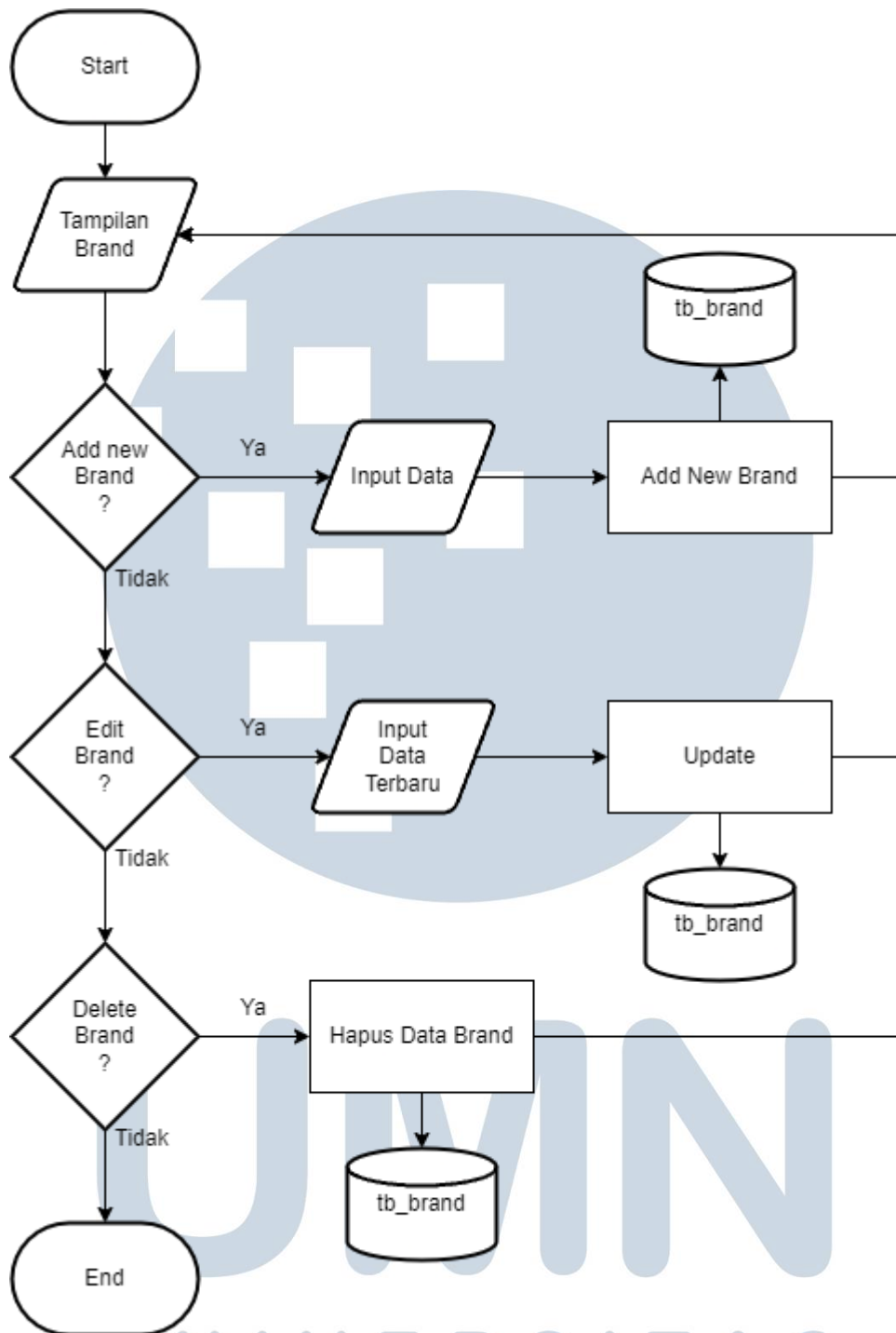
Gambar 3.10 menjelaskan aliran saat administrator memilih menu *slides*. Pertama ditampilkan halaman *featured vr headset* yang dapat merubah *bottom image* pada Bentuk Halaman *home* di *frontend*, kemudian administrator dapat melakukan perubahan gambar dengan melakukan *input* gambar yang diperlukan. Setelah *input* gambar, gambar tersebut disimpan ke *database vr headset*.



Gambar 3.11 *Flowchart Menu Vr headset*

Gambar 3.11 menjelaskan aliran saat administrator memilih menu *vr headset*. Menampilkan halaman *brand* yang digunakan untuk mengolah jenis data merek *vr headset* dan halaman *vr headset list* yang digunakan untuk mengolah data kriteria *vr headset*.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

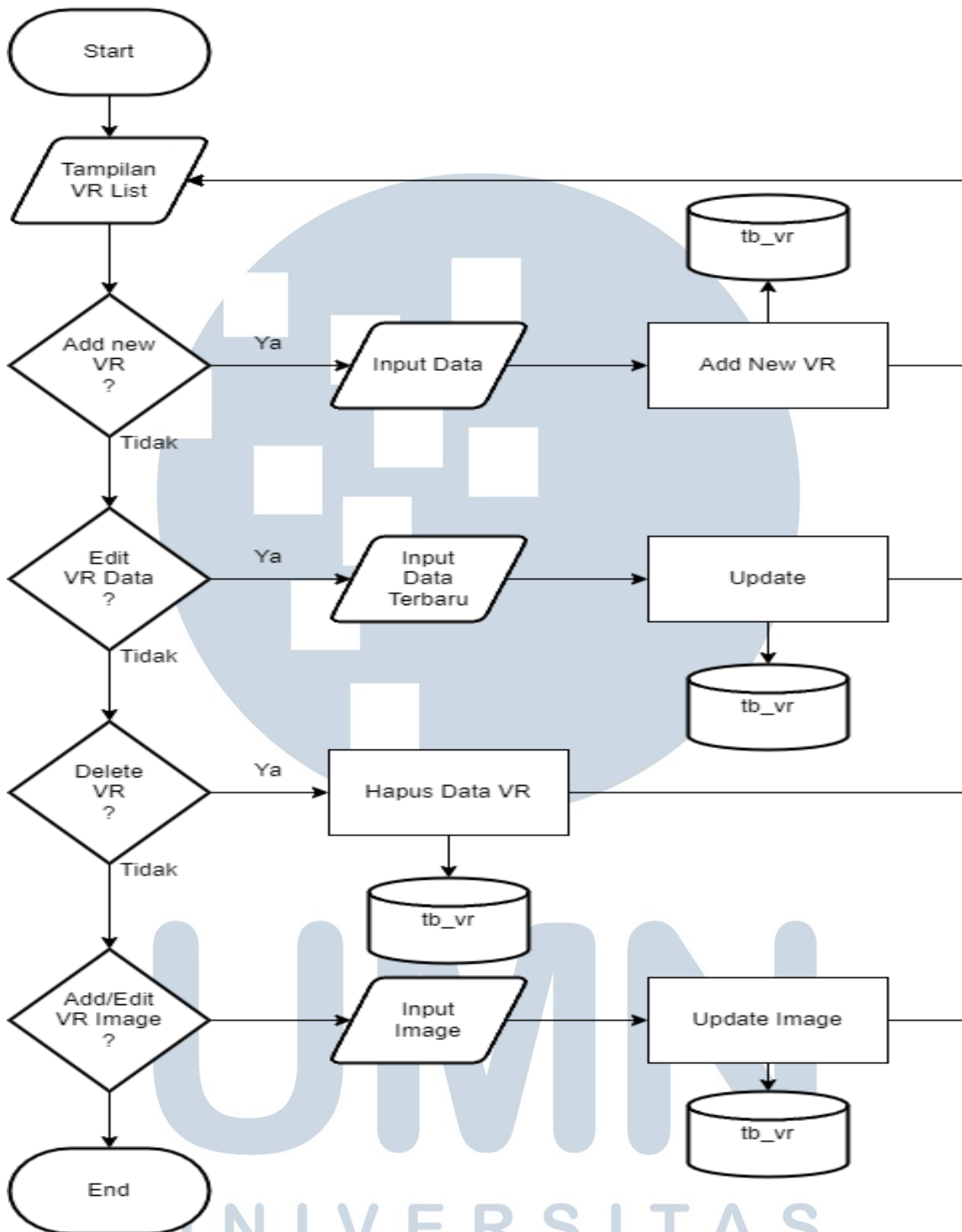


Gambar 3.12 Flowchart Menu Brand

Gambar 3.12 menjelaskan aliran saat administrator memilih menu *brand*. Diawali dengan halaman *brand* yang berisi data *brand vr headset*, kemudian administrator dapat menambahkan data baru dengan melakukan input

yang diperlukan. Setelah *input* data, data tersebut disimpan ke *database*. Kemudian administrator dapat melakukan *edit* data yang sudah ada pada *database* dengan cara *input* data baru yang akan diperbaharui. Setelah itu administrator memperbaharui data baru tersebut, kemudian data baru tersebut disimpan kembali ke dalam *database*. Kemudian admin dapat menghapus data yang sudah ada pada *database* dengan cara memilih data yang akan dihapus. Setelah data dihapus, daftar *vr headset* tersebut disimpan kembali ke dalam *database*.



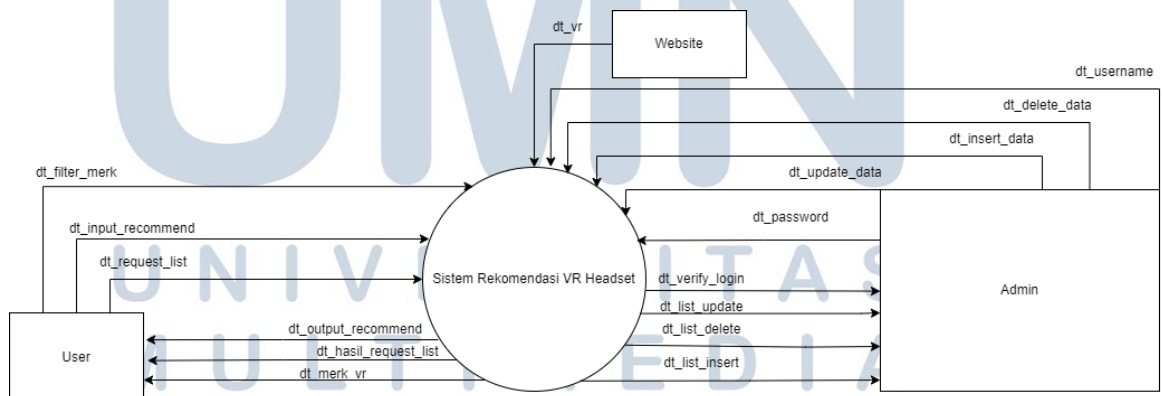


Gambar 3.13 Flowchart Menu Vr headset List

Gambar 3.13 menjelaskan aliran ketika administrator memilih *brand*. Di awali dengan halaman *brand* yang berisi data *brand vr headset*, kemudian administrator bisa melakukan tambah data dan gambar baru dengan melakukan *input* data. Setelah *input* data, data tersebut disimpan ke *database*. Kemudian administrator dapat memperbaharui data dan gambar yang sudah ada dengan cara memasukkan data dan gambar baru yang akan diubah. Setelah itu administrator memperbaharui data tersebut, kemudian data dan gambar tersebut disimpan kembali ke *database*. Kemudian administrator dapat melakukan penghapusan data dan gambar yang sudah ada dengan cara memilih data dan gambar yang akan dihapus. Setelah data dan gambar dihapus, daftar *vr headset* tersebut disimpan kembali ke dalam *database*.

3.2.2 Data Flow Diagram (DFD)

Data Flow Diagram digunakan untuk memperjelas dan mempermudah setiap alur data yang terjadi didalam sistem. Berikut adalah *Data Flow Diagram* (DFD) yang digunakan.

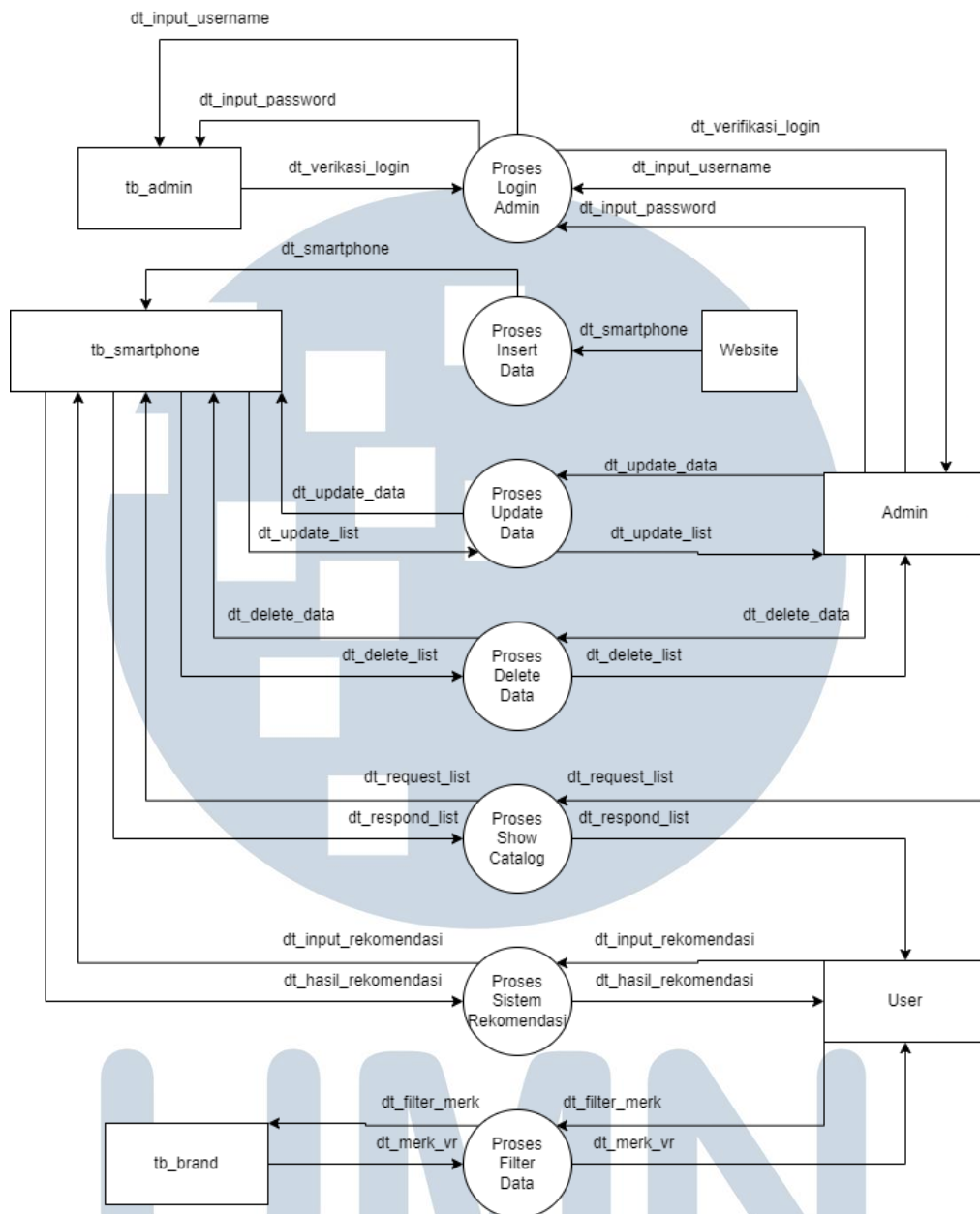


Gambar 3.14 *Data Flow Diagram*

Pada Gambar 3.14 menunjukkan *context diagram* yang menggambarkan bahwa sistem ini memiliki 1 proses utama dan terdapat tiga buah entitas yaitu *user*, administrator dan *website*. Pada entitas administrator terdapat aliran data yang berfungsi untuk melakukan proses *login* yang ditunjukkan oleh *dt_username*, *dt_password* dan *dt_verifikasi_login*.

Entitas administrator juga dapat mendapatkan *privilege* untuk mengolah data yang ditunjukkan oleh *dt_insert_data* untuk menambahkan data *vr headset* pada *database*, *dt_update_data* untuk merubah data *vr headset* yang sudah ada pada *database*, *dt_delete_data* untuk menghapus data *vr headset* yang sudah ada pada *database*. Sedangkan entitas *user* dapat melakukan proses rekomendasi *vr headset* yang ditunjukkan oleh *dt_input_rekomendasi*, *dt_filter_merk*, *dt_request_list* dan mendapatkan hasil rekomendasi yang ditunjukkan oleh *dt_output_rekomendasi*, *dt_hasil_request_list*, *dt_merk_vr headset*. Entitas *user* juga dapat melakukan *filter vr headset* pada halaman *catalogue* yang ditunjukkan oleh *dt_merk_vr headset*. Entitas *website* adalah sumber data *vr headset* yang berasal dari masing-masing situs resmi merk *vr headset* tersebut.





Gambar 3.15 Data Flow Diagram Level 1

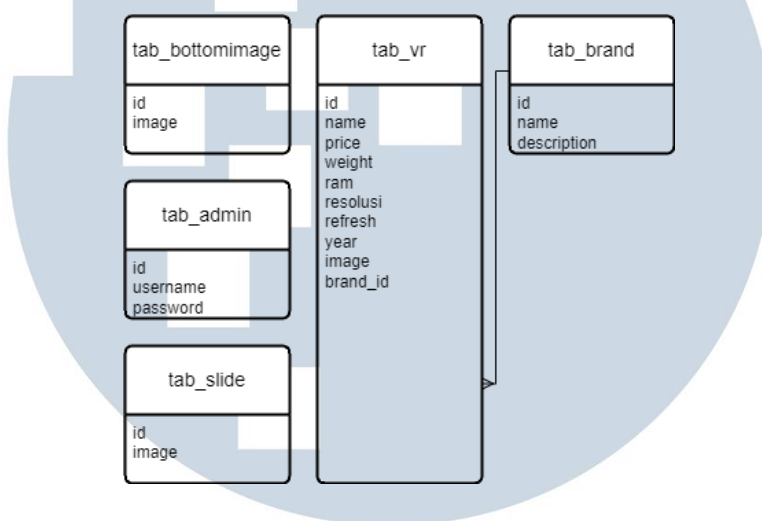
Gambar 3.15 Menjelaskan DFD level 1 memiliki subsistem yang terdapat pada sistem rekomendasi *vr headset* seperti proses sistem rekomendasi, proses *filter* data, proses *login* administrator, proses *insert* data, proses *update* data dan proses *delete* data. DFD juga menjelaskan data *storage* yang terdapat pada *user*, antara lain *tab_vr* untuk menyimpan *database vr headset*, *tab_administrator* untuk menyimpan *username* dan *password*

yang berfungsi untuk proses *login* dan *tab_brand* untuk menyimpan *brand*

vr headset.

3.2.3 Entity Relationship Diagram

Dalam suatu sistem yang dibangun dan dirancang, terdapat hubungan antara tabel yang satu dengan tabel yang lain dalam database sehingga dapat diberikan rekomendasi kepada sistem. Berikut gambaran ERD yang digunakan pada sistem..



Gambar 3.16 Entity Relationship Diagram

Gambar 3.16 menjelaskan bahwa *primary key* pada *tab_brand* menjadi *foreign key* pada *tab_vr*. Pada gambar diatas terdapat lima tabel yaitu *tab_administrator*, *tab_vr headset*, *tab_brand*, *tab_bottomimage*, *tab_slides*. Berikut adalah penjelasan dari ERD tersebut.

1. *tab_administrator* adalah tabel yang menyimpan data administrator dan digunakan saat proses *login* administrator.
2. *tab_vr* adalah tabel yang menyimpan data *vr headset* beserta spesifikasinya dan digunakan dalam proses rekomendasi *vr headset* berdasarkan *input* dari *user* menggunakan metode MOORA.

3. *tab_brand* adalah tabel yang menyimpan data merek *vr headset*.
4. *tab_bottomimage* adalah tabel yang menyimpan data gambar yang akan ditampilkan pada bagian *featured vr headset*.
5. *tab_slides* adalah tabel yang menyimpan data gambar yang akan ditampilkan pada bagian *slide show*.

Struktur Table yang dimiliki sistem rekomendasi *vr headset* adalah sebagai berikut.

a. Tabel Admin

Nama Tabel : *tab_administrator*

Fungsi : Menyimpan data administrator untuk melakukan login
Primary Key : *id*

Foreign Key : -

Tabel 3.1 Tabel *tab_administrator*

| No | Nama Kolom | Tipe Data | Panjang | Deskripsi |
|----|-----------------|----------------|---------|----------------|
| 1 | <i>id</i> | <i>integer</i> | 10 | Primary key |
| 2 | <i>username</i> | <i>varchar</i> | 30 | Username admin |
| 3 | <i>password</i> | <i>text</i> | - | Password admin |

b. Tabel Vr headset

Nama Tabel : *tab_vr*

Fungsi : Menyimpan data *vr headset* yang ada dalam sistem

Primary Key : *id*

Foreign Key : *brand_id*

Tabel 3.2 Tabel *tab_vr*

| No | Tipe Data | Panjang | Deskripsi |
|----|----------------|---------|-----------------------|
| 1 | <i>Integer</i> | 10 | Primary key |
| 2 | <i>Varchar</i> | 100 | Nama dari vr headset |
| 3 | <i>Integer</i> | 100 | Harga dari vr headset |
| 4 | <i>Integer</i> | 100 | Berat vr headset |

Tabel 3.2 Tabel tab_vr(lanjutan)

| | | | | |
|----|----------|----------------|-----|------------------------|
| 5 | ram | <i>Integer</i> | 50 | Kapasitas dari memory |
| 6 | resolusi | <i>Integer</i> | 50 | Ukuran resolusi layar |
| 7 | refresh | <i>Integer</i> | 100 | Refresh Rate |
| 8 | year | <i>Integer</i> | 100 | Tahun rilis vr headset |
| 9 | image | Varchar | 100 | Gambar dari vr headset |
| 10 | brand_id | <i>integer</i> | 100 | Merk dari vr headset |

c. Tabel Brand

Nama Tabel : *tab_brand*

Fungsi : Menyimpan brand_id dari *vr headset*

Primary Key : id

Foreign Key : -

Tabel 3.3 Tabel tab_brand

| No | Nama Kolom | Tipe Data | Panjang | Deskripsi |
|----|-------------|----------------|---------|----------------------------|
| 1 | id | <i>integer</i> | 10 | Primary key |
| 2 | name | varchar | 50 | Merk vr headset |
| 3 | description | text | - | Keterangan asal perusahaan |

d. Tabel Bottom Image

Nama Tabel : *tab_bottomimage*

Fungsi : Menyimpan gambar di bawah *Homepage*

Primary Key : id

Foreign Key : -

Tabel 3.4 Tabel tab_bottomimage

| No | Nama Kolom | Tipe Data | Panjang | Deskripsi |
|----|------------|----------------|---------|---|
| 1 | id | <i>integer</i> | 11 | Primary key |
| 2 | image | text | - | Menyimpan gambar pada featured vr headset |

e. Tabel Slides Image

Nama Tabel : *tab_slides*

Fungsi : Menyimpan gambar slides

Primary Key : *id*

Foreign Key : -

Tabel 3.5 Tabel *tab_slides*

| No | Nama Kolom | Tipe Data | Panjang | Deskripsi |
|----|--------------|----------------|---------|----------------------------------|
| 1 | <i>id</i> | <i>integer</i> | 11 | Primary key |
| 2 | <i>image</i> | <i>text</i> | - | Menyimpan gambar pada slide show |

3.2.3 Mockup Antarmuka

Mockup antarmuka terdiri dari dua bagian yaitu *front-end* dan *back-end*.

a. **Frontend**

Mockup antarmuka *front-end* dibagi menjadi 4 bagian yaitu *Homepage*, *About Us*, *Catalogue* dan *Recommend*. Berikut ini merupakan gambar dari Bentuk Halaman tersebut.



Gambar 3.17 Mockup Halaman *Homepage*

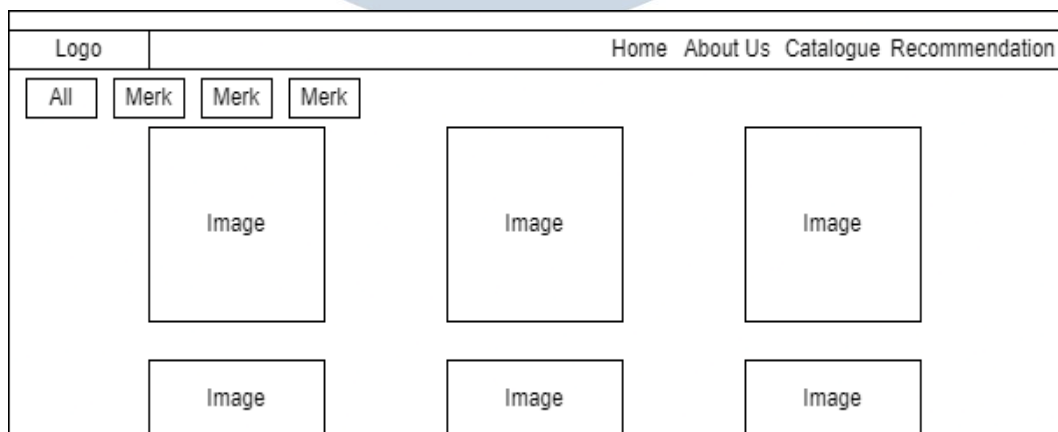
Gambar 3.17, menjelaskan *Homepage* dibagi menjadi 2 yaitu *header* dan *content*. Bagian *header* berisi logo dan menu yang terdiri dari halaman *home*, *about us*, *catalogue* dan *Recommendation*.

Content berisi *image slides* yang menampilkan gambar *vr headset*. Di bawah *image slides* terdapat 3 *image* kecil yang berisi gambar *vr headset*.

Gambar 3.18 *Mockup Halaman About Us*



Gambar 3.18 menjelaskan *About Us* yang dapat diakses oleh pengguna. *About Us* terdiri dari sebuah gambar aplikasi dan deskripsi singkat sistem rekomendasi *vr headset*.



Gambar 3.19 *Mockup Halaman Catalogue*

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Gambar 3.19 menjelaskan *Mockup Catalogue* yang dapat diakses oleh pengguna. Halaman *catalogue* menampilkan data dari *vr headset* yang terdapat pada *database* meliputi gambar, nama dan *brand*.

The mockup shows a web page with a header containing a logo and navigation links: Home, About Us, Catalogue, and Recommendation. Below the header, there are two 'Filter' buttons on either side of a large rectangular area. Inside this area, the text 'Daftar Pertanyaan Kriteria Pembobotan' is centered. At the bottom of this area is a 'Submit' button.

Gambar 3.20 *Mockup* Halaman *Recommendation*

Gambar 3.20, menjelaskan halaman *Recommendation*. Terdapat *filter* berisi *dropdown* yang berfungsi untuk menyaring data *vr headset* berdasarkan *brand vr headset*. Bagian pertanyaan pembobotan berisi 5 pertanyaan dan 5 *radio-button*. Di bawah pertanyaan terdapat tombol *submit* untuk memproses *input* dari *user*.

The mockup shows a web page with a header containing a logo and navigation links: Home, About Us, Catalogue, and Recommendation. Below the header, there are three rectangular boxes arranged horizontally. Each box has a smaller box below it containing the word 'Rank'. Below these three boxes is a single 'Back' button.

Gambar 3.21 *Mockup* Antarmuka Hasil *Recommendation*

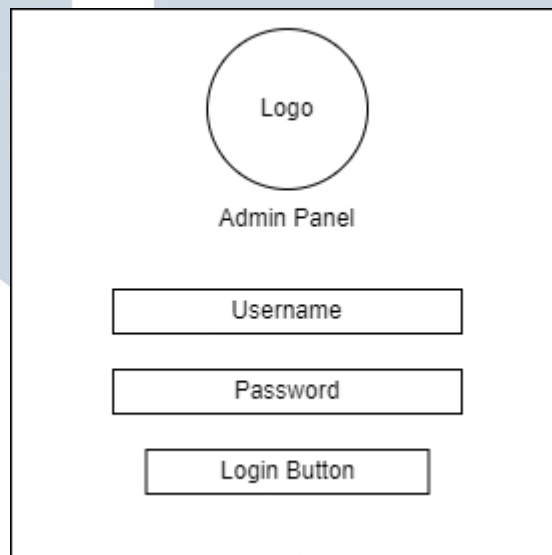
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Gambar 3.21 menjelaskan *Mockup* halaman untuk hasil rekomendasi bagi pengguna. Pada bagian tengah menampilkan 3 gambar *vr headset* beserta *ranking* dan data lengkap *database*.

b. Backend

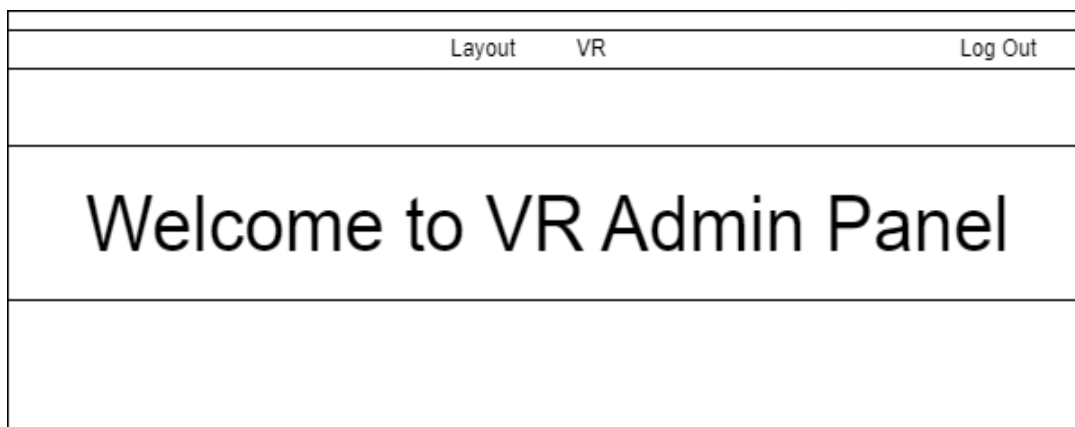
Mockup backend dibagi menjadi 6 yaitu *login administrator*, menu administrator, menu *slideshow*, menu *featured vr headset*, menu *brand* dan menu *vr headset list*. Berikut ini merupakan gambar dari Bentuk Halaman tersebut.

Gambar 3.22 *Mockup* Halaman *Login Administrator*



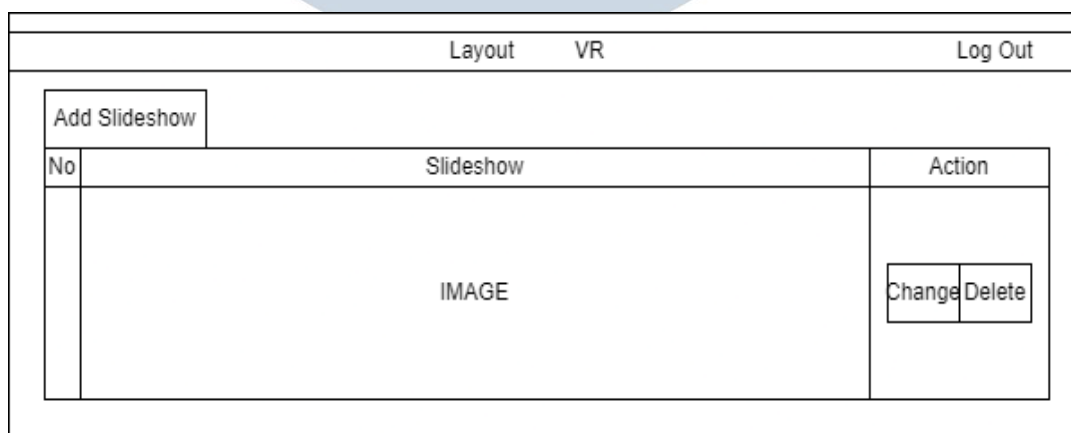
The image shows a wireframe for an administrator login page. At the top center is a circle labeled 'Logo'. Below it is the text 'Admin Panel'. There are three rectangular input fields stacked vertically, labeled 'Username', 'Password', and 'Login Button'.

Gambar 3.22 menjelaskan antarmuka *login administrator*. Halaman *login* adalah halaman dimana administratoristrator dimintegera untuk memasukkan nama pengguna dan kata sandi Pada bagian tengah halaman *login administrator* terdapat 2 kotak teks untuk dimasuki. Di bagian bawah terdapat 1 tombol *login* di bagian bawah untuk lanjut ke bagian *Dashboard*.



Gambar 3.23 *Mockup* Halaman Menu Administrator

Gambar 3.23 menjelaskan halaman menu administrator. Pada bagian tengah konten terdapat *text* yang berisi *welcome to vr headset dashboard*. Halaman ini berisi 4 fungsi yaitu *insert*, *update*, *delete* dan *logout*. Pada bagian *header* terdapat 2 menu yang di dalamnya terdapat fungsi untuk menampilkan konten dari fungsi yang dipilih administrator.



Gambar 3.24 *Mockup* Halaman Slideshow

Gambar 3.24 menjelaskan halaman *slideshow* yang dilakukan untuk memasukkan gambar ke dalam *database*. Pada bagian kanan konten terdapat *button change* dan *delete*, administrator dapat memilih gambar dengan klik

tombol *change image*. Kemudian menyimpannya ke dalam *database* dengan meng-klik *button save*.

| Layout | | VR | Log Out |
|----------|----------|----|---------|
| Featured | | | |
| No | Featured | | Action |
| | IMAGE | | Change |

Gambar 3.25 *Mockup* Antarmuka Halaman *Featured Vr headset*

Gambar 3.25 Menjelaskan halaman *featured vr headset* yang dilakukan untuk memasukkan gambar ke dalam *database*. Pada bagian kanan *content* terdapat *button change image*, administrator dapat memilih gambar menggunakan tombol *change image*. Kemudian menyimpannya ke dalam *database* dengan *button save*.

| Layout | | VR | Log Out |
|-----------|------|-------------|---------------|
| Add Brand | | | |
| No | Name | Description | Action |
| | | | Change Delete |

Gambar 3.26 *Mockup VR Headset Brand*

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Gambar 3.26 menjelaskan antarmuka dari halaman *brand* pada menu administrator. Pada bagian *content* terdapat 3 *button* yaitu *button add new brand*, *button edit* dan *button delete*. Pada *button add new brand* berfungsi untuk memasukkan data ke dalam *database*. Di bagian kanan *content* terdapat *button edit*

| Layout VR | | | | | | | | | | Log Out |
|-----------|-------|-------|------|-------|--------|-----|------------|--------------|------|---------------|
| VR List | | | | | | | | | | |
| Add VR | | | | | | | | | | |
| No | Image | Brand | Name | Price | Weight | RAM | Resolution | Refresh Rate | Year | Action |
| | | | | | | | | | | Change Delete |

dan *button delete*.

Gambar 3.27 Mockup Halaman Vr headset List

Gambar 3.27 menjelaskan halaman *vr headset list* pada menu administrator. Pada bagian *content* terdapat 4 *button* yaitu *button add new vr headset*, *button edit*, *button change image* dan *button delete*. Pada bagian *button add new vr headset* berfungsi untuk memasukkan data baru ke *database*.

Di bagian kanan *content* terdapat *button edit* yang berfungsi untuk merubah data yang sudah ada pada *database* menjadi data baru, *button change image* berfungsi untuk memasukkan gambar *vr headset* ke dalam *database* dan *button delete* yang berfungsi untuk menghapus data pada *database*.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A