

## BAB 3 METODOLOGI PENELITIAN

### 3.1 Metodologi Penelitian

Pada penelitian “Rancang Bangun Sistem Pendukung Keputusan Pemilihan Komputer Menggunakan Metode AHP dan TOPSIS Berbasis Website” terdapat tahapan – tahapan penelitian agar sistem dapat dirancang dan dibangun. Adapun penjelasan dari tiap tahapan, yaitu:

1. Penentuan Topik

Mengetahui masalah yang terjadi dalam di lingkungan sekitar yang kesulitan dalam memilih prebuilt PC. Dengan permasalahan ini dibutuhkan sebuah solusi untuk memecahkan masalah ini.

2. Studi Literatur

Pada tahap ini dilakukan pengumpulan, membaca dan memahami buku, jurnal dan karya ilmiah sebagai referensi untuk memperkuat teori. Tahap ini juga dapat membantu untuk lebih memahami topik seputar AHP dan TOPSIS sehingga teori tersebut dapat dikembangkan sebagai penelitian.

3. Pengumpulan Data

Pada tahap ini dilakukan pengumpulan data dari kriteria yang digunakan beserta bobotnya dan data dari prebuilt PC. Wawancara dilakukan untuk mendapatkan informasi dan memperoleh data yang dapat menjelaskan ataupun menjawab suatu permasalahan penelitian. Berdasarkan wawancara kriteria yang digunakan adalah storage, RAM, VGA, Motherboard, Processor dan Harga, Selanjutnya untuk mendapatkan bobot dari produk akan digunakan *website Passmark benchmark*. Ketika data untuk kriteria berhasil didapatkan akan dilanjutkan pengumpulan data terkait dari prebuilt PC yang akan dimasukkan ke dalam sistem pendukung keputusan. Data prebuilt PC didapatkan dari *marketplace*.

4. Perancangan Sistem

Di tahap ini dilakukan pembuatan *flowchart*, struktur *database* dan *mockup* dari sistem yang akan dibangun.

## 5. Pembangunan Sistem

Pada tahap ini sistem dibuat berdasarkan perancangan sistem dan data yang sudah dikumpulkan meliputi hasil dari penerapan metode dan hasil perhitungan dari AHP dan TOPSIS. Sistem dibuat menggunakan Bahasa pemrograman PHP dengan *framework* Laravel.

## 6. Uji Coba Sistem

Pada tahapan ini dilakukan pengujian sistem yang sudah dibuat untuk menentukan apakah hasil sesuai dengan perhitungan secara manual dan mengetahui adanya *error* dalam sistem. Pada tahapan ini juga dilakukan uji coba *user* untuk mengetahui penilaian berdasarkan USE Questionnaire.

## 7. Penulisan Laporan

Laporan dibuat sebagai penulisan dan dokumentasi hasil dari penelitian “Rancang Bangun Sistem Pendukung Keputusan Pemilihan Komputer Menggunakan Metode AHP dan TOPSIS Berbasis Website”

### 3.2 Perancangan Sistem

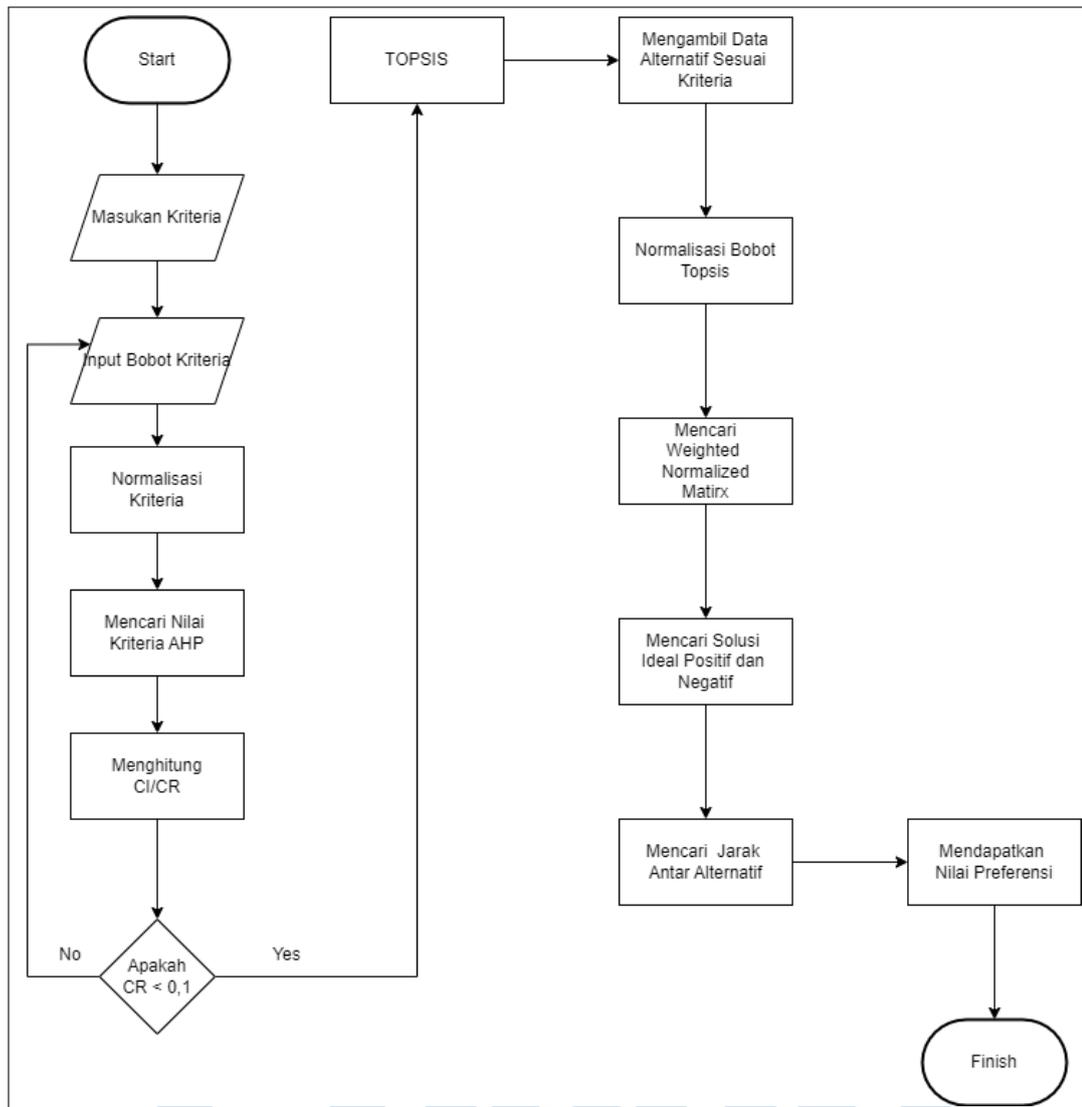
Setelah melakukan penelitian tentang AHP-TOPSIS dilakukan perancangan sistem yang dimulai dengan *flowchart* yang berisi alur dari sistem. Selanjutnya ada struktur dari *database* yang digunakan dalam sistem dan rancangan desain awal website dalam bentuk *wireframe*

#### 3.2.1 Flowchart

*Flowchart* adalah diagram alur yang menggambarkan alur suatu proses dari sistem atau *algorithm*. Bagan ini menjelaskan urutan dari prosedur apa saja yang dikerjakan oleh sistem.

##### A Flowchart AHP-TOPSIS

Flowchart ini merupakan perhitungan dari AHP-TOPSIS. Untuk perhitungan AHP jika saat perhitungan matriks yang sudah dibuat tidak konsisten akan dibuat kembali matriks sampai hasil dari perhitungan tersebut konsisten agar dapat melanjutkan ke dalam perhitungan TOPSIS. Setelah itu perhitungan TOPSIS akan digunakan untuk mendapatkan *ranking*.

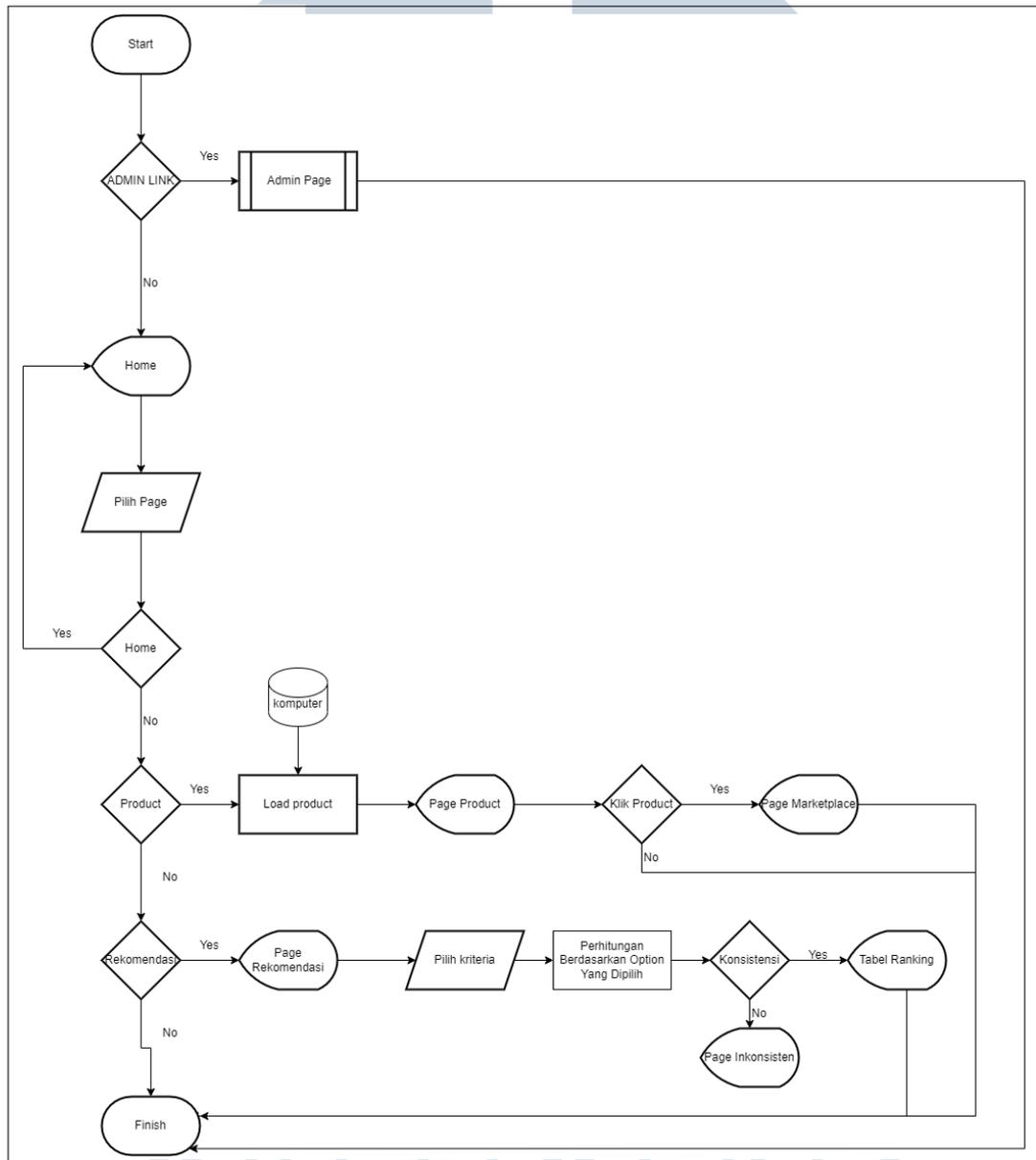


Gambar 3.1. Flowchart AHP-TOPSIS

Pada Gambar 3.1 Merupakan flowchart dari AHP-TOPSIS. Fungsi AHP-TOPSIS akan mengambil kriteria beserta bobot yang di-input pengguna. Kemudian metode AHP akan menghitung nilai bobot kriteria yang telah di-input. Jika CR dari metode AHP lebih dari 0,1 maka matriks yang sudah di-input tidak konsisten. Jika nilai CR kurang dari 0,1 akan diteruskan ke dalam perhitungan TOPSIS. Nilai kriteria yang sebelumnya sudah dihitung dengan metode AHP akan menjadi bobot kriteria dan akan dihitung kembali ke dalam metode TOPSIS. Dalam metode TOPSIS akan mengambil data dari produk beserta bobot per kriteria nya sebagai perhitungan. Ketika sudah mendapatkan nilai preferensi, nilai ini digunakan sebagai acuan untuk *ranking* dari produk,

## B Flowchart Main Page

Main Page flowchart dibuat untuk merancang halaman utama dari sistem ini. Jika berhasil login akan dialihkan ke dalam halaman admin.



Gambar 3.2. Flowchart Main Page

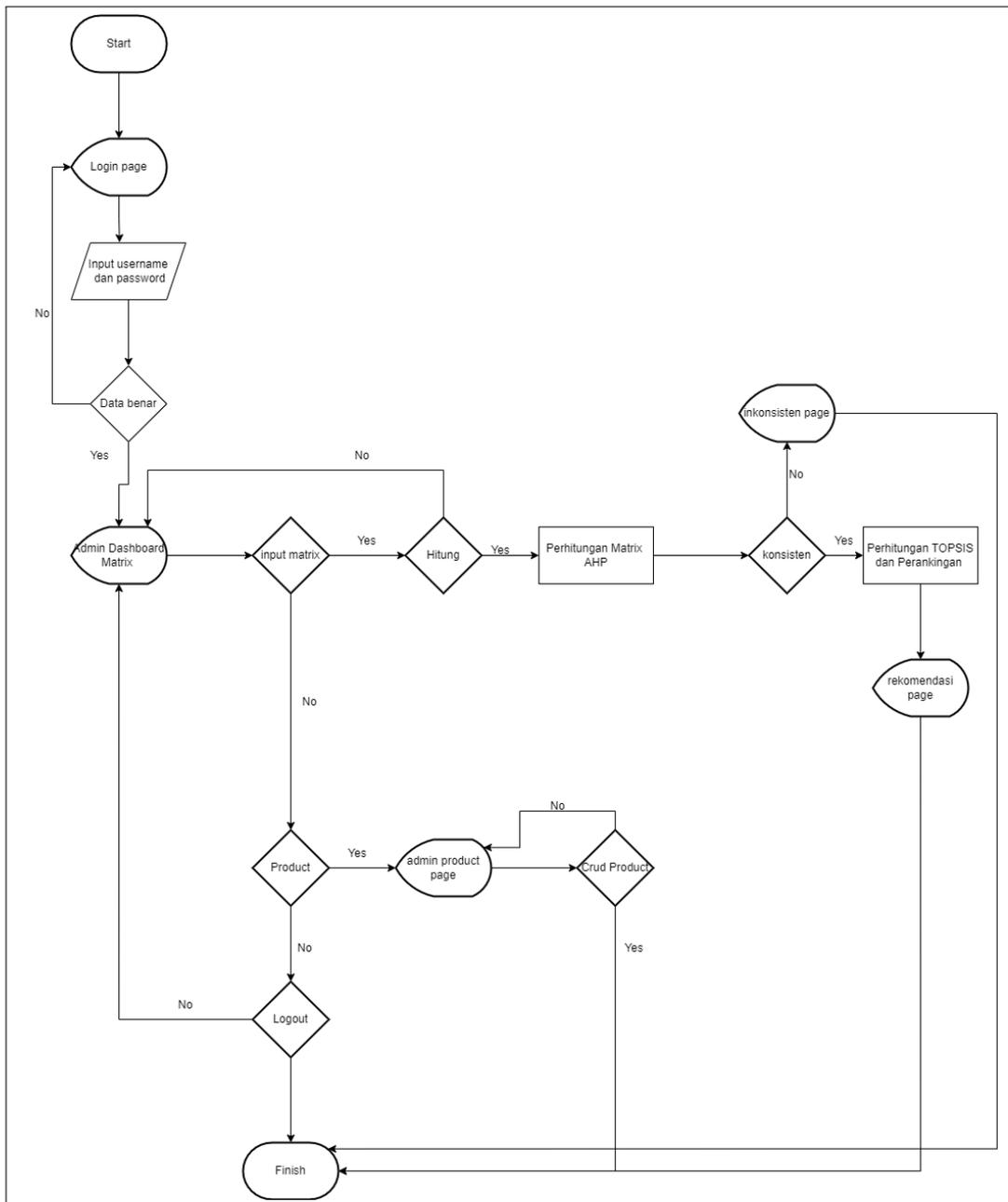
Pada gambar 3.2 merupakan *flowchart* dari halaman *website* ini. Terdapat 2 page awal yang dapat diakses yaitu *admin page* dan *home page*. Pada *home page* terdapat *home*, *product* dan rekomendasi. Jika *home* di klik akan kembali pada halaman *home page*. Jika *product* di klik, sistem akan memuat *database komputer*

dan menampilkannya pada *product page*. Ketika gambar atau nama di klik, *user* akan dialihkan ke dalam halaman *marketplace* dimana *product* tersebut dijual. Dan terakhir ada rekomendasi dimana *user* dapat memasukan data pada matriks. Jika matriks yang di *input user* tidak konsisten akan dialihkan ke halaman inkonsisten. Data dari perhitungan komputer akan keluar jika matriks yang di *input* konsisten.

### C Flowchart Admin Page

Flowchart Admin Page dibuat untuk admin ketika berhasil login ke dalam Admin *page*. Terdapat fungsi untuk mengubah matriks AHP dan data dari *product*.





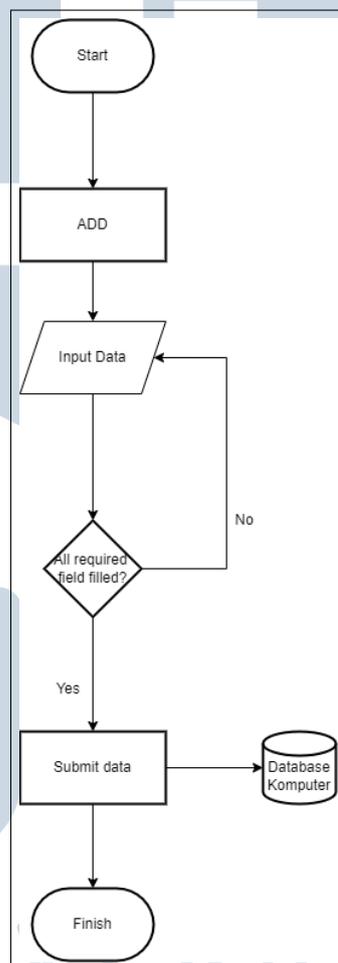
Gambar 3.3. Flowchart Admin Page

Pada gambar 3.3, *user* menambahkan /admin pada *link* dari *website* dan memasuki *admin page*. Di dalam admin ada *product* dan matriks bobot kriteria AHP. Sesudah *login*, *user* dialihkan ke halaman matriks. Disini *user* dapat memasukkan matriks yang nantinya dapat dihitung untuk mendapatkan kriteria komputer. Saat matriks tersebut konsisten akan diarahkan ke dalam rekomendasi *page* dan jika tidak konsisten, akan diarahkan ke dalam inkonsisten *page*. Saat *user* menekan

tombol *product page* akan menampilkan data *product*.

#### D Flowchart Add

Flowchart Add merupakan fitur yang dapat diakses ketika mengakses *product*. Admin dapat membuat data *product* baru. Data dari *product* ini dapat digunakan untuk rekomendasi komputer pada sistem ini. Data yang dibuat dapat di *submit* jika sudah mengisi seluruh *form* yang disediakan

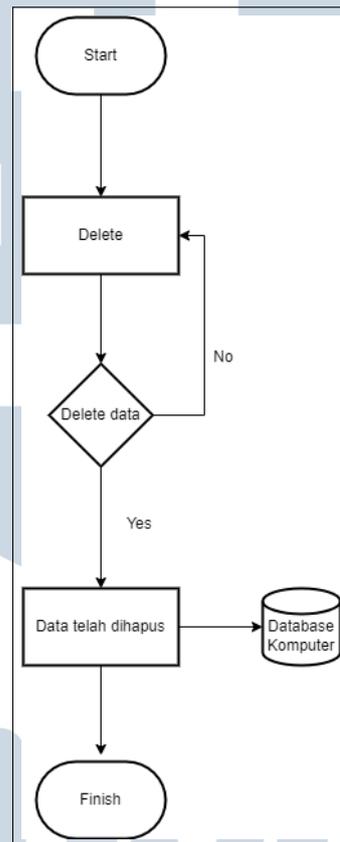


Gambar 3.4. Flowchart Add

Gambar 3.4 merupakan flowchart untuk *add* data. Data dalam *form* tidak boleh ada yang kosong. Jika kosong tidak dapat submit data. Ketika sudah *submit*, data akan otomatis ke dalam *database*.

## E Flowchart Delete

Flowchart Delete digunakan untuk fungsi menghapus data dari product yang sudah di-*input* sebelumnya. Data yang sudah masuk tidak akan masuk ke dalam rekomendasi komputer.

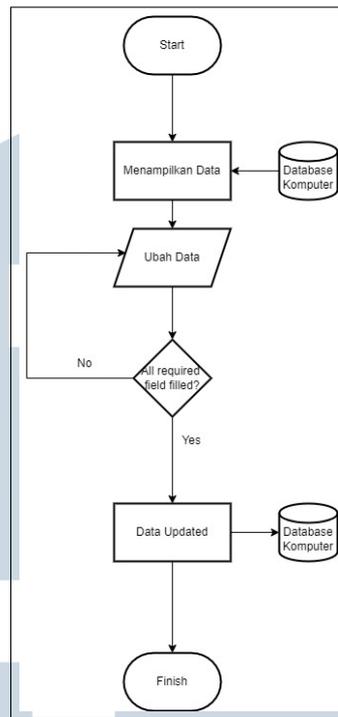


Gambar 3.5. Flowchart Delete

Di gambar 3.5 adalah *flowchart* untuk fungsi *delete*. *Delete* akan menghapus data yang sudah dimasukkan ke dalam *website*. Data yang sudah dihapus tidak dapat dipulihkan. Saat melakukan proses *delete* akan muncul *pop up* untuk memastikan apakah data akan dihapus atau tidak.

## F Flowchart Update

Flowchart Update untuk fungsi mengubah data yang sudah di-*input* sebelumnya. Data tersebut dapat diubah jika seluruh *form* sudah terisi.



Gambar 3.6. Flowchart Update

Gambar 3.6 adalah fungsi *update* yang digunakan untuk mengubah data yang sudah ada. *Form* tidak bisa kosong dan jika kosong tidak dapat *submit* data. Setelah selesai melakukan *update* data dalam database akan berubah secara otomatis.

### 3.2.2 Database

Berikut merupakan struktur database yang digunakan dalam program ini.

#### A Tabel Komputer

Tabel 3.1 digunakan untuk menyimpan data dari produk. Di dalam tabel ini juga terdapat bobot dari setiap komponen yang ada dalam produk tersebut.

Tabel 3.1. Tabel Komputer

Name	Type	Keterangan
id	int(11)	Primary key, AUTO_INCREMENT
nama	varchar(255)	Berisi nama dari prebuilt PC
motherboard	float	Bobot untuk Motherboard
processor	float	Bobot untuk Processor
vga	float	bobot untuk VGA
ram	float	Bobot untuk RAM
storage	float	Bobot untuk Storage
harga	float	Bobot untuk harga
img	longtext	Berisi <i>link image resource</i> dari produk
link	longtext	Link <i>marketplace</i> dari produk tersebut
brand_processor	longtext	Brand dari Processor
brand_vga	longtext	Brand dari VGA

## B Tabel temp bobot

Tabel 3.2 digunakan untuk memasukan bobot di matriks untuk perhitungan dengan metode AHP-TOPSIS

Tabel 3.2. Tabel temp bobot

Name	Type	Keterangan
id	int(11)	Primary key, AUTO_INCREMENT
nama	varchar(255)	Berisi nama dari prebuilt pc
motherboard	float	Bobot matriks yang di- <i>input</i> untuk motherboard
processor	float	Bobot matriks yang di- <i>input</i> untuk processor
vga	float	Bobot matriks yang di- <i>input</i> untuk Vga
ram	float	Bobot matriks yang di- <i>input</i> untuk ram
storage	float	Bobot matriks yang di- <i>input</i> untuk storage
harga	float	Bobot matriks yang di- <i>input</i> untuk harga

## C Tabel User

Tabel 3.3 digunakan untuk membuat dan menyimpan data *user* yang nantinya bisa digunakan untuk login.

Tabel 3.3. Tabel User

Nama	Type	Keterangan
id	int(10)	Primary key untuk menyimpan ID <i>user</i>
name	varchar(255)	Untuk menyimpan nama dari <i>user</i>
email	varchar(255)	Untuk menyimpan email untuk login <i>user</i>
password	varchar(255)	Untuk menyimpan password login <i>user</i>
created_at	timestamp	Untuk menyimpan tanggal dibuatnya <i>user</i>
updated_at	timestamp	Untuk menyimpan tanggal dibuatnya <i>user</i>
deleted_at	timestamp	Untuk menyimpan tanggal dibuatnya <i>user</i>

#### D Tabel temp\_d\_neg

Tabel 3.4 merupakan tabel untuk menyimpan data solusi ideal negatif dalam perhitungan metode *Technique for Order Preference by Similarity to Ideal Solution*.

Tabel 3.4. Tabel temp\_d\_neg

Nama	Type	Keterangan
id	int(10)	Menyimpan id dari solusi ideal negatif
nama	varchar(191)	Nama dari produk
dNegatif	double	Hasil dari solusi ideal negatif

#### E Tabel temp\_d\_pos

Tabel 3.5 merupakan tabel untuk menyimpan data solusi ideal positif dalam perhitungan metode *Technique for Order Preference by Similarity to Ideal Solution*.

Tabel 3.5. Tabel temp\_d\_pos

Nama	Type	Keterangan
id	int(10)	Menyimpan id dari solusi ideal positif
nama	varchar(191)	Nama dari produk
dPositif	double	Hasil dari solusi ideal Positif

## F Tabel temp\_normalisasi

Tabel 3.6 merupakan hasil dari normalisasi perhitungan metode *Technique for Order Preference by Similarity to Ideal Solution*.

Tabel 3.6. Tabel temp\_normalisasi

Name	Type	Keterangan
id	int(11)	Primary key, AUTO_INCREMENT
nama	varchar(255)	Berisi nama dari prebuilt pc
motherboard	double	Bobot matriks yang di-input untuk motherboard
processor	double	Bobot matriks yang di-input untuk processor
vga	double	Bobot matriks yang di-input untuk Vga
ram	double	Bobot matriks yang di-input untuk ram
storage	double	Bobot matriks yang di-input untuk storage
harga	double	Bobot matriks yang di-input untuk harga

## G Tabel temp\_normalisasi kriteria

Tabel 3.7 merupakan tabel yang berisi hasil dari normalisasi matriks ahp pada Tabel 3.2.

Tabel 3.7. Tabel temp\_normalisasi kriteria

Name	Type	Keterangan
id	int(11)	Primary key, AUTO_INCREMENT
nama	varchar(255)	Berisi nama dari prebuilt pc
motherboard	double	Bobot matriks yang di-input untuk motherboard
processor	double	Bobot matriks yang di-input untuk processor
vga	double	Bobot matriks yang di-input untuk Vga
ram	double	Bobot matriks yang di-input untuk ram
storage	double	Bobot matriks yang di-input untuk storage
harga	double	Bobot matriks yang di-input untuk harga
matriks_aw	double	Mendapatkan data yang sudah ternormalisasi matriks_aw

## H Tabel temp\_nilai\_pref

Pada tabel 3.8 terdapat data yang merupakan hasil akhir dari perhitungan sistem. Nilai preferensi yang sudah didapat akan di *store* dalam tabel ini.

Tabel 3.8. temp\_nilai\_pref

Name	Type	Keterangan
id	int(10)	ID dari hasil akhir
nama	varchar(191)	Nama produk
val	double	Value dari perhitungan metode AHP-TOPSIS

### 3.2.3 Wireframe

*Wireframe* adalah kerangka awal dari sebuah website yang nantinya akan diimplementasikan. Pada *wireframe* ini terdapat *wireframe home*, *product*, rekomendasi dan tampilan admin

#### A Wireframe Home

Halaman Home adalah halaman yang pertama kali diakses saat masuk ke dalam website. Gambar 3.7 merupakan wireframe dari halaman Home.

UMMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

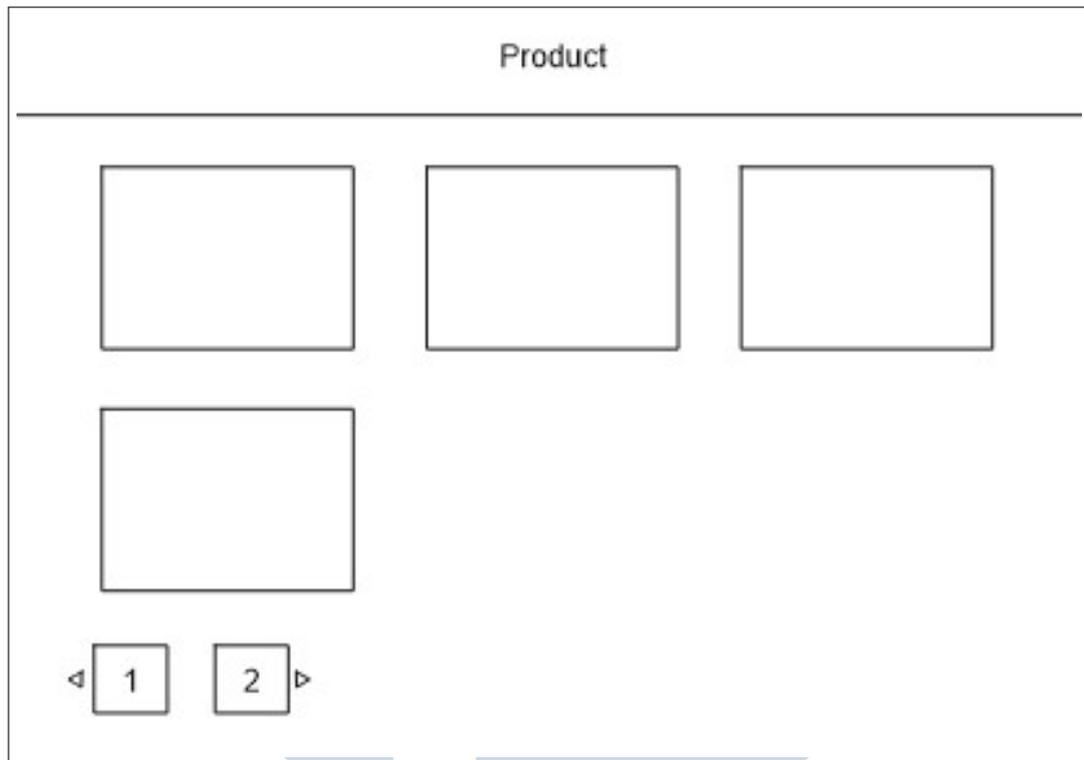


Gambar 3.7. Wireframe Home

Rancangan awal untuk *home page* terdapat pada gambar 3.7. Pada gambar ini Home akan memuat *head*, gambar, *feature*, *product* dan *footer*. Untuk *feature* akan dijelaskan pengertian dari sistem pendukung keputusan dan disertai button try dss untuk agar *user* mencoba sistem rekomendasi. Sebelum *footer* akan terdapat beberapa produk yang sudah dimasukkan ke dalam *website*.

## **B Wireframe Product**

Halaman Product digunakan untuk melihat *product* yang sudah dimasukkan ke dalam *website* ini. *User* dapat melihat *product* yang sudah dimasukkan ke dalam *website* pada halaman ini.

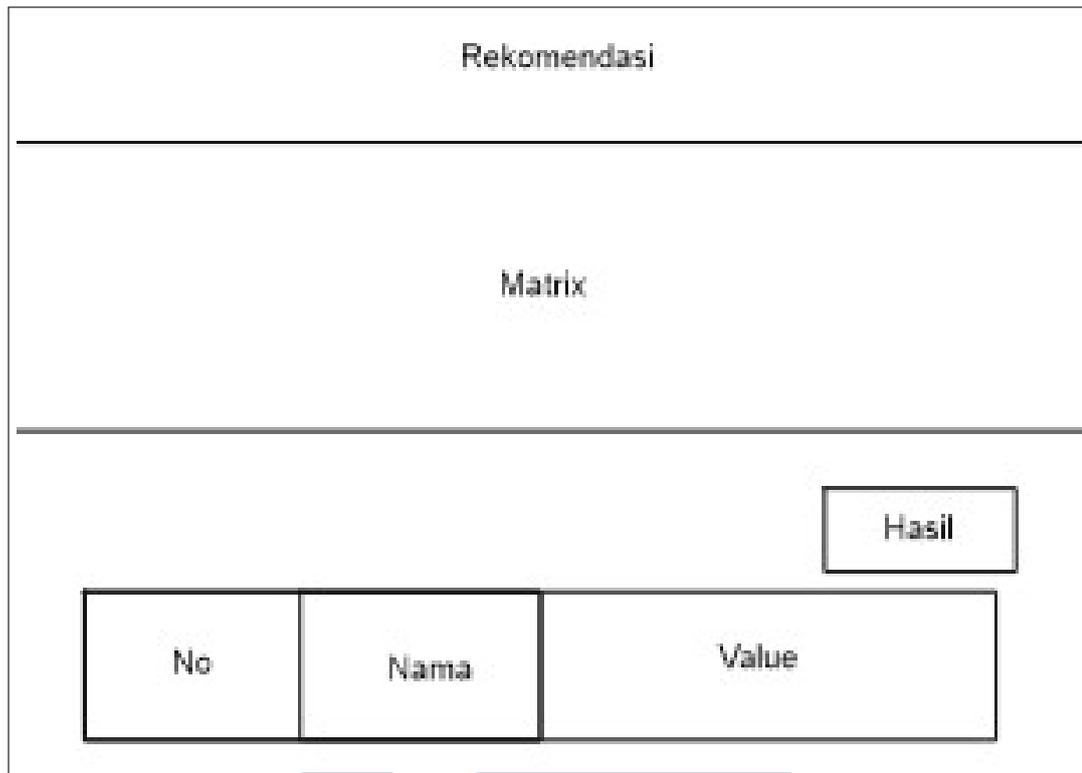


Gambar 3.8. Wireframe Product

Pada gambar 3.8 ini adalah kerangka awal untuk tampilan *product page*. Pada halaman ini akan memuat 4 data dan jika *product* lebih dari 4 maka akan ada panah untuk *next*. Disini juga akan ada panah untuk *previous* untuk melihat *product slide* sebelumnya.

### C Wireframe Rekomendasi

Halaman dimana user dapat melihat hasil *ranking* yang diurutkan berdasarkan bobot komputer terbesar pertama. *User* dapat memilih opsi untuk *brand* dari Processor dan VGA beserta harga yang diinginkan.

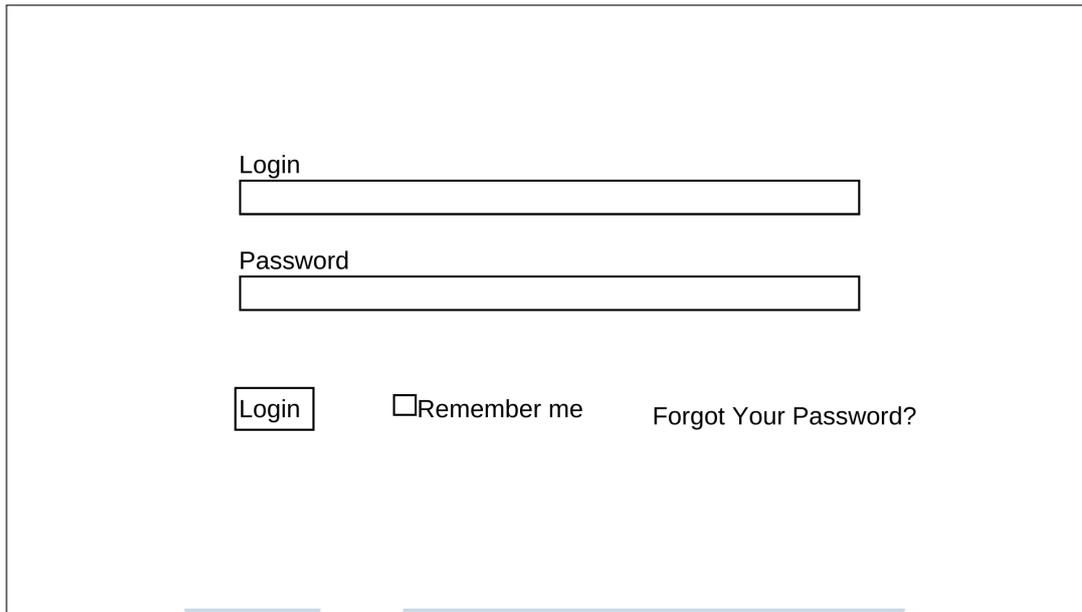


Gambar 3.9. Wireframe Rekomendasi

Di dalam gambar 3.9 adalah *page* awal untuk rekomendasi. Di dalam *page* ini akan memuat sebuah matriks dan hasilnya dapat dihitung dengan tombol hitung. Jika sudah hasil dari perhitungan rekomendasi akan muncul dalam bentuk tabel dibawah matriks tersebut.

#### D Wireframe Login

Wireframe Login adalah tampilan yang akan muncul ketika menggunakan admin *link*. Admin dapat mengakses Dashboard dan Product serta mengubah isi yang disediakan pada halaman tersebut.



A wireframe of a login form. It consists of two input fields: the first is labeled "Login" and the second is labeled "Password". Below the input fields, there is a "Login" button, a checkbox labeled "Remember me", and a link labeled "Forgot Your Password?".

Gambar 3.10. Wireframe Login

Gambar 3.10 merupakan tampilan awal dari halaman login untuk memasuki admin page.

### **E Wireframe Product**

Wireframe Product untuk halaman dari Admin Product untuk admin membuat, menghapus dan mengubah data komputer yang akan ditampilkan dan dihitung.



Dashboard Products Logout	Header									
		Nama Komputer	Motherboard	Processor	VGA	RAM	Storage	Harga	Action	
									<input type="button" value="Edit"/>	<input type="button" value="Delete"/>

Gambar 3.11. Wireframe Product

Gambar 3.11 merupakan tampilan admin untuk *product*. Di dalam halaman ini *user* dapat melihat, meng-*edit*, menghapus dan menambahkan produk

#### F Wireframe Add Product

Pada Add Product akan diberikan *form* yang wajib diisi. Data tersebut digunakan untuk menambah *product* komputer.

U M M N  
 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

Dashboard  Products  Logout	Header	
	Create Product	
	Nama	<input type="text"/>
	Merek	<input type="text"/>
	Processor	<input type="text"/>
	VGA	<input type="text"/>
	RAM	<input type="text"/>
	Storage	<input type="text"/>
	Harga	<input type="text"/>
	img	<input type="text"/>
Link	<input type="text"/>	
Brand Processor	<input type="text"/>	
Brand VGA	<input type="text"/>	
<input type="button" value="Save"/>		

Gambar 3.12. Wireframe Add Product

Ketika *user* ingin menambahkan produk *user* akan dipindahkan ke halaman *add product*. Tampilan awal untuk *add product* seperti yang ditampilkan pada Gambar 3.12. *User* harus mengisi *form* tersebut dan dapat menyimpan datanya.

### G Wireframe Edit Product

Wireframe ini merupakan halaman untuk fungsi edit *product*. terdapat *form* beserta isi dari data komputer yang sebelumnya sudah ditambahkan ke dalam sistem ini.

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

Dashboard	Header	
	Products	Create Product
		Nama <input type="text"/>
		Merek <input type="text"/>
		Processor <input type="text"/>
		VGA <input type="text"/>
		RAM <input type="text"/>
		Storage <input type="text"/>
		Harga <input type="text"/>
		img <input type="text"/>
Link <input type="text"/>		
Logout	Brand Processor <input type="text"/>	
	Brand VGA <input type="text"/>	
<input type="button" value="Save"/>		

Gambar 3.13. Wireframe Edit Product

Ketika *user* ingin mengubah produk *user* akan dipindahkan ke halaman *edit product*. Tampilan awal untuk *edit product* seperti yang ditampilkan pada Gambar 3.13. *User* harus mengisi *form* tersebut untuk menyimpan datanya.

## H Wireframe View Product

View Product akan menampilkan data yang sudah dibuat sebelumnya. Gambar dari *product* yang ditampilkan berdasarkan link yang di-*input* pada Add Product. Jika dilakukan perubahan pada suatu product maka isi dari View Product juga akan berubah.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

<b>Dashboard</b>  <b>Products</b>  <b>Logout</b>	<b>Header</b>	
	<b>View Product</b>	
	<b>Nama</b>	<input type="text"/>
	<b>Merek</b>	<input type="text"/>
	<b>Processor</b>	<input type="text"/>
	<b>VGA</b>	<input type="text"/>
	<b>RAM</b>	<input type="text"/>
	<b>Storage</b>	<input type="text"/>
	<b>Harga</b>	<input type="text"/>
	<b>img</b>	<input type="text"/>
	<b>Link</b>	<input type="text"/>
	<b>Brand Processor</b>	<input type="text"/>
	<b>Brand VGA</b>	<input type="text"/>

Gambar 3.14. Wireframe View Product

User dapat melihat data dari produk yang sudah ada pada *view product*. Tampilan awal untuk *view product* seperti yang ditampilkan pada Gambar 3.14.

