

BAB III

METODELOGI PENELITIAN

3.1 Alat dan Bahan Penelitian yang Digunakan

Alat dan bahan yang digunakan dalam penelitian ini adalah sebagai berikut:

3.1.1 Alat Penelitian

Pada penelitian ini penulis menggunakan alat dan bahan sebagai pendukung perancangan aplikasi pengenalan wajah untuk mengetahui keaslian mahasiswa dalam mengikuti ujian. Adapun kebutuhan spesifikasi perangkat keras untuk perancangan pada penelitian ini adalah:

3.1.1.1 Spesifikasi Kebutuhan *Hardware*

Untuk dapat menjalankan aplikasi dengan baik, tentunya struktur dari perangkat keras (*hardware*) haruslah memenuhi spesifikasi kebutuhan aplikasi yang dibutuhkan, adapun kebutuhan aplikasi terhadap struktur komputer adalah:

1. *Processor* : *Intel Core i3-4030U*
2. *Ram* : *8,00 GB*
3. *Hardisk* : *500 GB*
4. *Sysitem Type* : *64-bit Operating Syatem*

3.1.1.2 Spesifikasi Kebutuhan *Software*

Perangkat lunak (*software*) yang digunakan dalam pembuatan sistem pengenalan wajah mahasiswa adalah:

1. Sistem Operasi : *Microsoft Windows 8.1 Pro*
2. Bahasa Pemrograman : *Android, PHP*
3. *Database ManagementSystem (DBMS)* : *MySQL*
4. *Web Browser* : *Google Chrome 61.0*
5. Desain Logika Program : *Microsoft Office Visio 2007*

3.1.2 Bahan Penelitian

3.1.2.1 Jenis Data Penelitian

Adapun jenis data yang digunakan dalam penelitian ini adalah data primeryang dikumpulkan melalui observasi yang melibatkan mahasiswa kampus UIR yang berada di daerah Pekanbaru untuk mencari data mengenai sample foto mahasiswa jurusan teknik informatika agar nantinya dapat dijadikan data training, sehingga didapat data-data sebagai berikut :

1. Data mahasiswa berserta foto terbarunya.
2. Data lengkap mahasiswa.

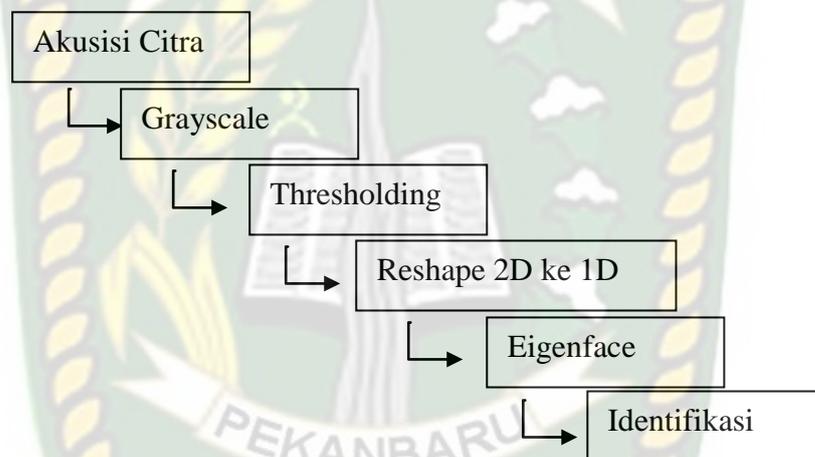
3.1.2.2 Teknik Pengumpulan Data

Adapun teknik pengumpulan data yang diperlukan dalam pengenalan mahasiswa diperoleh dari wawancara dan studi pustaka.

1. Wawancara dilakukan untuk mengumpulkan informasi yang akan berguna dalam memperoleh data mahasiswa. Wawancara dilakukan pada bagian akademik prodi Teknik Informatika.
2. Studi pustaka, mencari referensi-referensi ke pustaka sebagai pedoman penelitian yang penulis lakukan baik berupa buku maupun literatur yang berhubungan dengan penelitian.

3.2 Metodologi Penelitian

Pada saat ujian berlangsung pengawas tidak akan mengenali apakah mahasiswa tersebut benar atau joki ujian. Saat ujian berlangsung pengawas hanya membagikan kertas ujian kemudian mahasiswa melakukan pengisian terhadap lembar jawaban. Tidak akan diketahui bahwa mahasiswa yang melakukan ujian benar adanya atau hanya joki mahasiswa. Adapun uraian langkah-langkah proses pengolahan citra sebagai berikut :



Gambar 3.1 Langkah proses pengolahan citra

3.2.1 Akuisisi Citra

Pada tahap ini sistem menerima inputan data berupa file gambar (gambar digital yang diambil adalah gambar wajah). Data gambar tersebut nantinya akan diproses lebih lanjut pada tahap merubah citra (gambar) warna ke citra *grayscale*.

Ilustrasi gambar yang akan digunakan misalnya sebagai berikut ini :



Gambar 3.2 Ilustrasi foto wajah mahasiswa

Hasil dari citra warna diatas akan disajikan dalam contoh pembacaan RGB di bawah ini. Citra RGB dibawah merupakan contoh pembacaan dari citra warna yang ditunjukkan pada gambar 3.2. berikut ini hasil citra warna :

R=10	R=10	R=20
G=12	G=43	G=12
B=120	B=120	B=120
R=30	R=10	R=30
G=14	G=12	G=12
B=17	B=120	B=70
R=30	R=10	R=30
G=12	G=12	G=42
B=80	B=120	B=30

Gambar 3.3 Matriks Nilai Citra Warna

3.2.2 Citra Grayscale

Pada tahap ini sistem melakukan proses merubah citra warna menjadi citra *grayscale* (gradasi abu-abu). Ukuran pixel yang digunakan dalam contoh berikut adalah ukuran 3x3. Maka untuk akan didapatkan ukuran matrik citra N x (H x W) pixel dengan N adalah jumlah citra referensi. Maka hasilnya 3 x (3x3) adalah 3x9. Berikut contoh matriks nilai 3 x 9 pada citra referensi dan citra baru.

R=10	R=10	R=20
G=12	G=43	G=12
B=120	B=120	B=120
R=30	R=10	R=30
G=14	G=12	G=12
B=17	B=120	B=70
R=30	R=10	R=30
G=12	G=12	G=42
B=80	B=120	B=30
R=90	R=43	R=45
G=25	G=56	G=42
B=44	B=36	B=30

Gambar 3.4 Matriks Nilai Citra Warna

Setelah matriks nilai citra warna diketahui, maka langkah selanjutnya merubah matriks nilai citra warna menjadi citra grayscale. Berikut contoh penyelesaian perhitungan pada matriks nilai citra referensi (x1,y1) dan citra baru (x1,y2).

$$\frac{10+12+120}{3} = 48 \quad \text{dan} \quad \frac{60+12+60}{3} = 66$$

Berikut matriks nilai yang diperoleh dari perhitungan nilai kanal pada citra referensi dan citra baru.

48	58	51
21	48	54
41	48	34

Gambar 3.5 Matriks Nilai Hasil Perhitungan



Gambar 3.6 Matriks Nilai Hasil Perhitungan

3.2.3 Threshold

Pada tahap ini sistem melakukan proses perubahan citra *grayscale* yang sudah ditingkatkan kualitasnya (perbaikan) menjadi citra hitam putih (citra biner). Berikut contoh penyelesaian perhitungan matriks nilai citra referensi dan citra baru.

Pada Matriks Nilai Citra Referensi :

Nilai terbesar dari citra *grayscale* ditambah dengan nilai terkecil dibagi 2 maka akan mendapatkan batas ambang atau *threshold*.

Citra 1:

$$\frac{58+21}{2} = 40$$

Citra 2:

$$\frac{69+27}{2} = 48$$

Citra 3:

$$\frac{58+21}{2} = 39$$

Pada Matriks Nilai Citra Baru :

$$\frac{54+31}{2} = 43$$

Bila nilai threshold pada kedua citra sudah diketahui, maka dilakukan tahap selanjutnya untuk mendapatkan matriks nilai citra biner. Berikut contoh penyelesaian matriks nilai pada citra referensi dan citra.

Penyelesaian pada matriks nilai citra referensi citra 1 :

$$f(x,y) = \begin{cases} 255, & \text{jika } f(x,y) \geq 40 \\ 0, & \text{jika } f(x,y) < 40 \end{cases}$$

Penyelesaian pada matriks nilai citra referensi citra 2 :

$$f(x,y) = \begin{cases} 255, & \text{jika } f(x,y) \geq 48 \\ 0, & \text{jika } f(x,y) < 48 \end{cases}$$

Penyelesaian pada matriks nilai citra referensi citra 3 :

$$f(x,y) = \begin{cases} 255, & \text{jika } f(x,y) \geq 39 \\ 0, & \text{jika } f(x,y) < 39 \end{cases}$$

Penyelesaian pada matriks nilai citra baru :

$$f(x,y) = \begin{cases} 255, & \text{jika } f(x,y) \geq 43 \\ 0, & \text{jika } f(x,y) < 43 \end{cases}$$

Maka diperoleh matriks nilai citra biner berikut :

Citra Referensi:

255	255	255
0	255	255
255	255	255

Gambar 3.7 Matriks Nilai Hasil Perhitungan Citra Biner



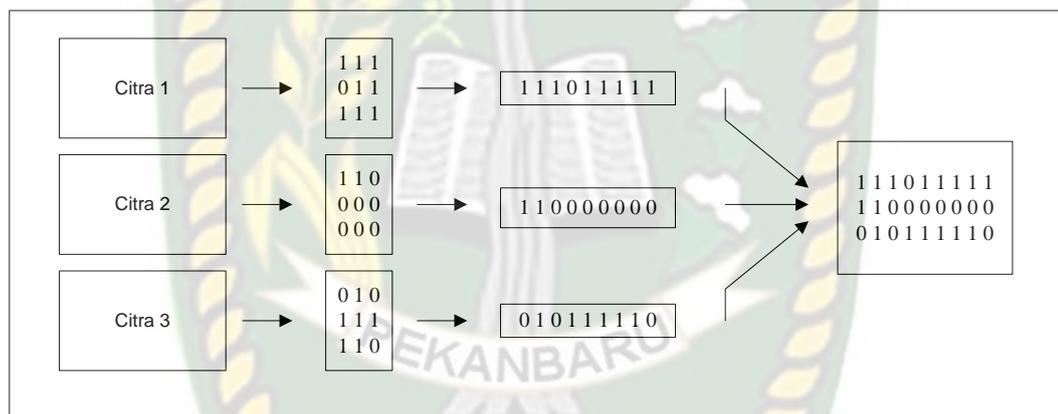
Gambar 3.8 Matriks Nilai Citra Biner

3.2.4 Eigenface

Pada tahap ini sistem melakukan pendeteksian wajah. Aplikasi akan membaca wajah dan mencocokkan dengan citra wajah sample agar diketahui bahwa benar mahasiswa tersebut yang mengikuti ujian. Berikut tahapan proses yang akan dilakukan dalam *eigenface* :

1. Penyusunan *flatvector*

Selanjutnya adalah menyusun seluruh training image menjadi suatu matrikstunggal. Misalnya image yang disimpan berukuran $H \times W$ piksel dan jumlahnya N buah, maka akan dimiliki vector ciri dengan dimensi $N \times (W \times H)$. Misalnya dalam training image terdapat 3 image dengan ukuran 3×4 piksel maka kita akan mempunyai eigenvector ukuran 3×9 . Pada proses flatvektor agar lebih mudah dalam perhitungan nilai biner threshold yang bernilai 255 diganti menjadi 1. Sehingga Ilustrasinya sebagai berikut.



Gambar 3.9 Penyusunan *Flatvector*

2. Perhitungan Rataan Fletvektor

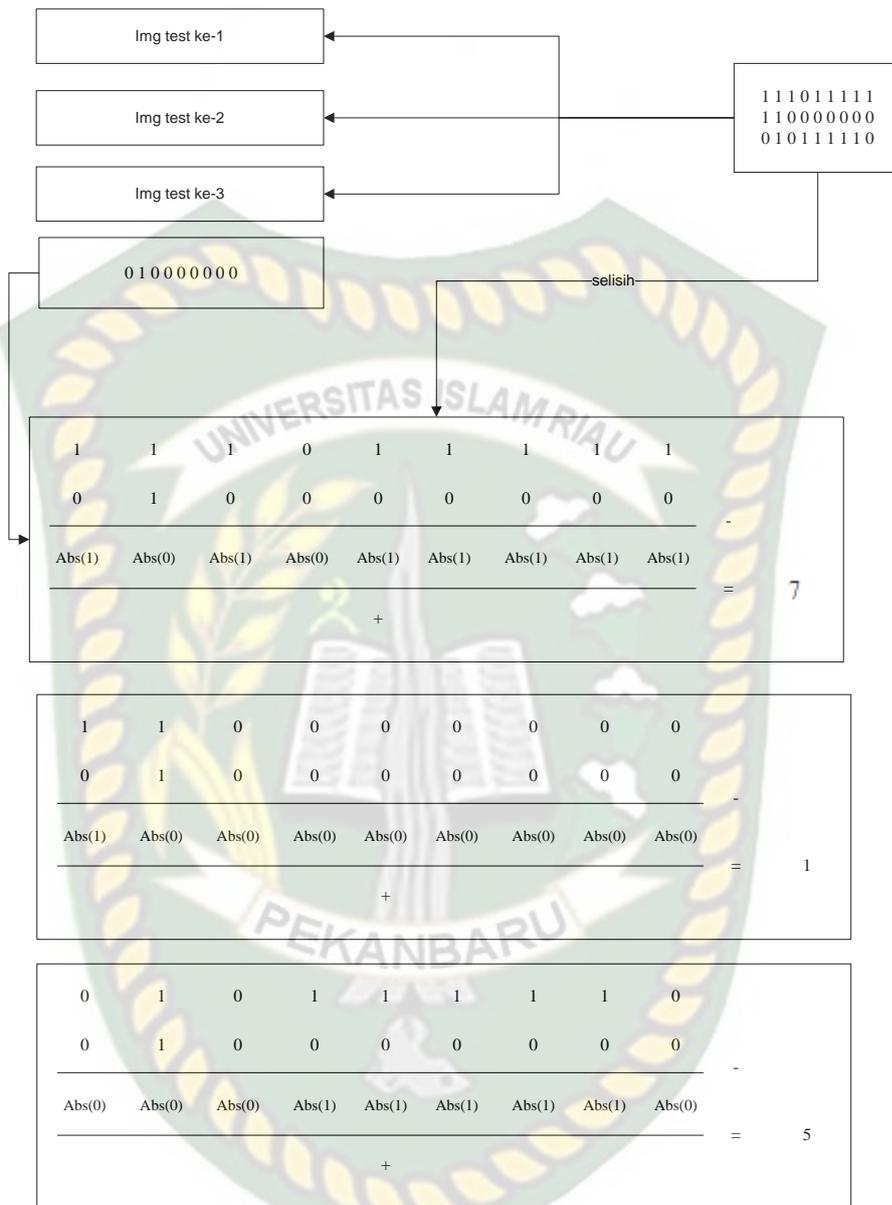
Dari vector ciri yang telah diperoleh, jumlahkan seluruh barisnya sehingga diperoleh matriks berukuran $1 \times (W \times H)$. Setelah itu bagi matriks tersebut dengan jumlah image N untuk mendapatkan nilai rata-rata vector ciri :



Gambar 3.10 Rata-rata Nilai Vektor

3. Mengitung eigenface mencari nilai terkecil dari citra baru

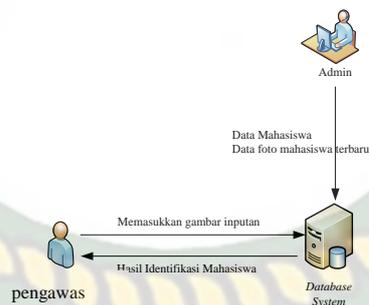
Cari distance minimum antara image test dan image hasil training. Bandingkan nilai euclidean distance minimum dari image yang di-capture dengan image yang sudah ada di database. euclidean distance itu angka pixel dari hasil flatvektor, angka tersebut digunakan untuk mencari jarak terdekat antara gambar uji dan training. Setelah nilai eigenface untuk image test diperoleh maka kita bisa melakukan identifikasi dengan menentukan jarak (distance) terpendek dengan eigenface dari eigen vector training image. Caranya dengan menentukan nilai absolute dari pengurangan baris i pada matriks eigenface training image dengan eigenface dari testface, kemudian jumlahkan elemen-elemen penyusun vektor yang dihasilkan dari pengurangan tadi dan ditemukan jarak d indeks i. Lakukan untuk semua baris. Cari nilai d yang paling kecil.



Gambar 3.11 Perhitungan Eigenface

3.3 Pengembangan Sistem

Dalam penelitian ini akan dirancang sebuah aplikasi akan membantu pengawas dalam mengidentifikasi peserta ujian agar tidak adanya joki mahasiswa, bisa dilihat pada gambar 3.5.



Gambar 3.12 Pengembangan Sistem

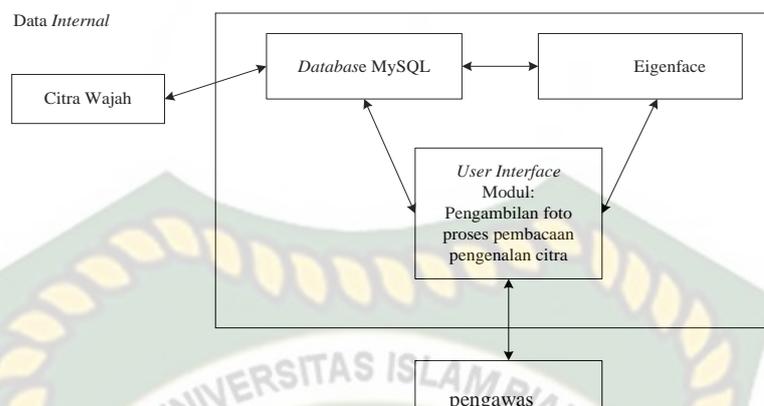
Dari gambar 3.11 dijelaskan bahwa Sebelum aplikasi mengenai pengenalan wajah mahasiswa melalui foto yang diambil dibangun, maka harus mengetahui proses yang akan berjalan. Pada proses yang berjalan di aplikasi. Pemakai aplikasi akan melakukan pengambilan foto pada wajah mahasiswa. Kemudian proses kedua aplikasi akan memproses menggunakan pengolahan citra digital dan aplikasi akan menunjukkan data foto apakah mahasiswa terdaftar atau tidak. Pada proses ini sebelumnya admin atau pembuatan aplikasi terlebih dahulu memasukan data foto terbaru dari mahasiswa dan informasi lengkapnya kedalam aplikasi agar dapat diketahui.

3.4 Perancangan Sistem

Pada tahap ini akan dijelaskan hal yang berhubungan dengan perancangan sistem yang akan dibuat :

3.4.1 Arsitektur Sistem

Tahapan ini menentukan apa saja yang dibutuhkan oleh sistem dan metode yang akan digunakan dalam pengolahan citra. Seperti menghubungkan *database* dengan *source* sistem dan membuat desain antarmuka.

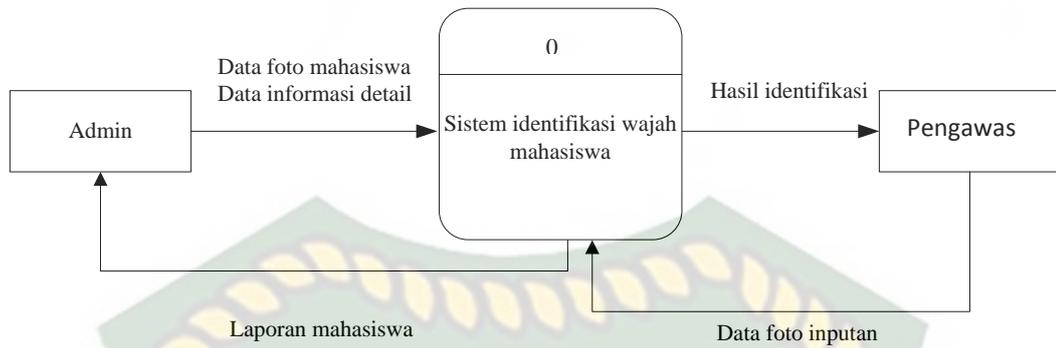


Gambar 3.13 Arsitektur Sistem

Dalam mendesain atau merancang perangkat lunak termasuk didalamnya adalah model *management* dengan menggunakan metode *Eigenface*. Data internal yang digunakan yaitu data citra foto mahasiswa yang disimpan dalam *database* MySQL. *User interface* berupa proses memasukan data foto terbaru mahasiswa, proses memasukan informasi lengkap mahasiswa, dan proses menampilkan hasil.

3.4.2 Diagram Konteks

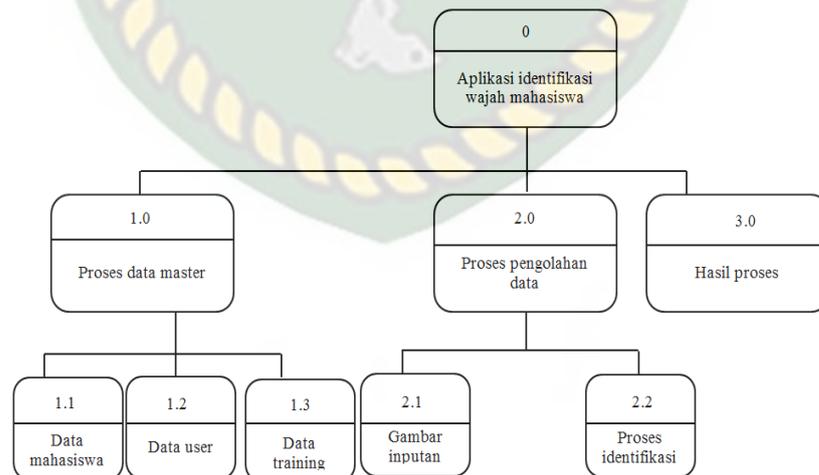
Diagram konteks (*Context Diagram*) digunakan untuk menggambarkan hubungan input dan *output* antara sistem dengan entitas luar, suatu diagram konteks selalu memiliki satu proses yang mewakili seluruh sistem. Sistem ini memiliki dua buah eksternal *entity* yaitu admin dan pengawas.



Gambar 3.14 Diagram Konteks

3.4.3 Hierarchy Chart

Hierarchy chart merupakan gambaran subsistem yang menjelaskan proses-proses yang terdapat dalam sistem utama dimana semua subsistem yang berada dalam ruang lingkup sistem utama saling berhubungan satu dan lainnya yang membedakan adalah pada level prosesnya. *Hierarchy chart* sistem yang akan dibangun bisa dilihat pada gambar 3.14.



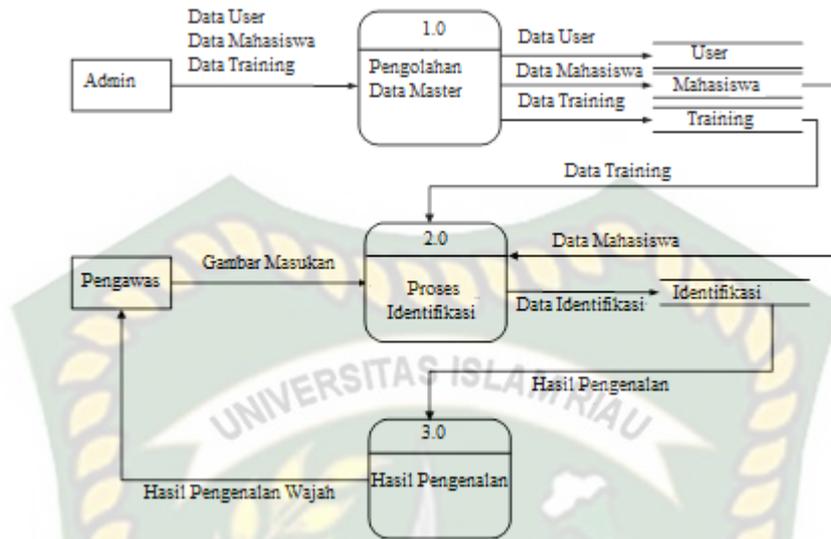
Gambar 3.15 Hierarchy Chart

3.4.4 Data Flow Diagram (DFD)

Data flow diagram (DFD) akan menjelaskan alur sistem, DFD ini juga akan menggambarkan secara visual bagaimana data tersebut mengalir, pada aplikasi pengenalan identifikasi mahasiswa dalam mengikuti ujian ini terdapat beberapa level proses yaitu :

3.4.4.1 DFD Level 0

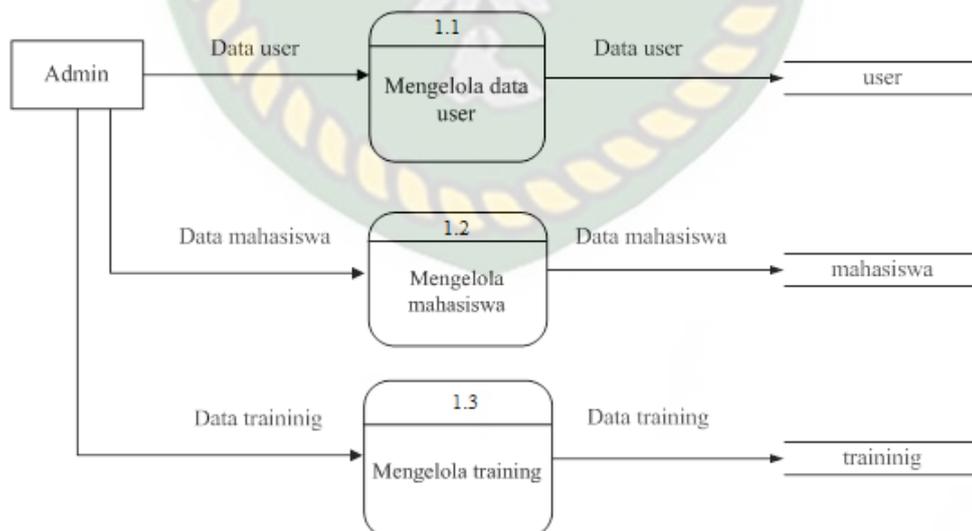
Bisa dilihat pada gambar 3.15 proses pengolahan data master bertugas mengelola data mahasiswa, data training. Data data mahasiswa, data training yang diinputkan oleh admin kemudian disimpan pada *data store*. Sedangkan data inputan masukan pengambilan foto mahasiswa akan dilakukan oleh pengawas untuk mencocokkan dengan data training. Selanjutnya dari *data store* data training dan data inputan foto mahasiswa akan diproses dengan metode *Eigenface*. Hasil proses tersebut merupakan hasil yang menunjukkan mahasiswa benar atau tidak.



Gambar 3.16 DFD Level 0

3.4.4.2 DFD Level 1 Proses 1

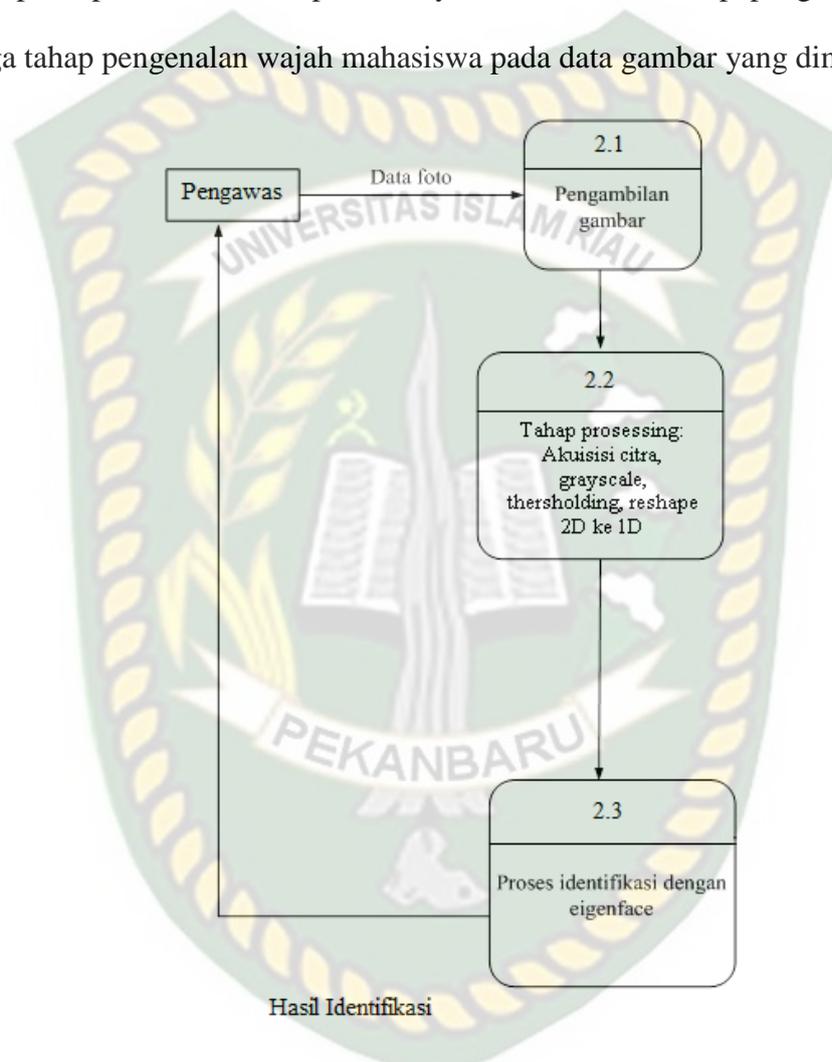
Pada proses pengelolaan data dibagi dalam 3 proses yaitu mengelola mengelola data user, data mahasiswa, dan data training yang akan dikelola oleh admin, bisa dilihat pada gambar 3.17.



Gambar 3.17 DFD Level 1 Proses 1

3.4.4.3 DFD Level 1 Proses 2

Pada gambar 3.18 dijelaskan pemakai menginputkan foto. Data foto akan di proses pada pemrosesan tahap kedua yaitu mulai dari tahap pengambilan foto hingga tahap pengenalan wajah mahasiswa pada data gambar yang dimasukan.



Gambar 3.18 DFD Level 1 Proses 2

3.4.5 Desain Input

Desain input pada sistem pendeteksiian wajah ini terdiri dari:

1. Rekam Mahasiswa

Fungsi : Mengelola data mahasiswa

Nama tabel : mahasiswa

Tombol Simpan : Untuk menyimpan data yang diinput

Tombol Reset : Untuk mengembalikan form yang sudah diisi seperti semula

Tombol Edit : Untuk mengedit data yang sudah ada pada database

Tombol Hapus : Untuk menghapus data mahasiswa

DATA MAHASISWA

FORM MAHASISWA

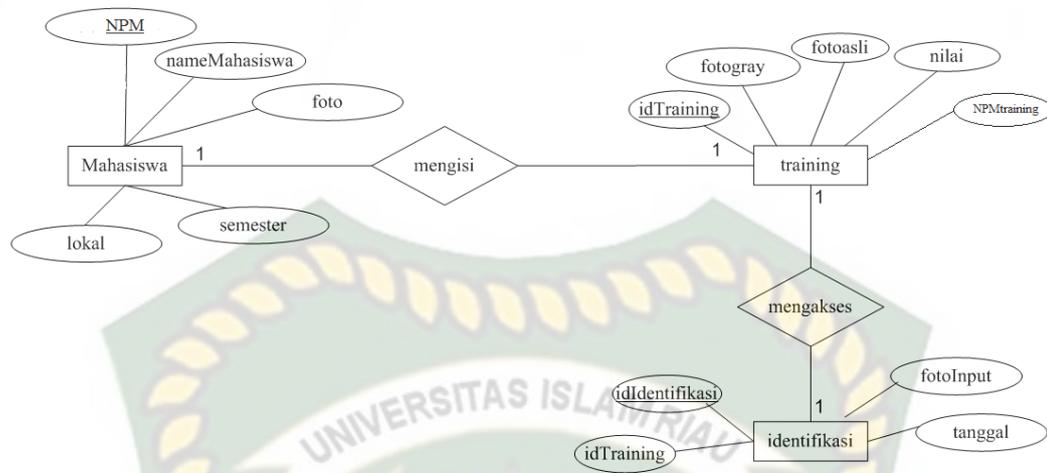
NPM	X (6)
Nama	X (30)
Foto	X (100) ▼
Lokal	X (3)
Semester	X (3)

Gambar 3.19 Rekam Mahasiswa

3.4.6 Perancangan Database

3.4.6.1 Entity Relationship Diagram (ERD)

ERD merupakan tampilan dari hubungan antar entitas yang ada pada database.



Gambar 3.20 Entity Relationship Diagram (ERD)

3.4.6.2 Desain Database

1. Tabel Kriteria

Nama Database : db_mahasiswa

Nama Tabel : mahasiswa

Tabel 3.1 Tabel mahasiswa

No	Field	Data Type	Size	Ket
1	NPM	Varchar	15	Primary Key
2	nameMahasiswa	Varchar	30	
3	Foto	Varchar	100	
4	Lokal	Varchar	3	
5	Semester	Int	3	

2. Tabel Nilai Training

Nama *Database* : db_mahasiswa

Nama Tabel : training

Tabel 3.2 Tabel training

No	<i>Field</i>	<i>Data Type</i>	<i>Size</i>	Ket
1.	IdTraining	Int	10	<i>Primary Key</i>
1	NPM Training	Varchar	6	
2	Fotogray	Varchar	50	
3	Fotoasli	Varchar	50	
4	Nilai	Varchar	10	

3. Tabel training

Nama *Database* : db_mahasiswa

Nama Tabel : identifikasi

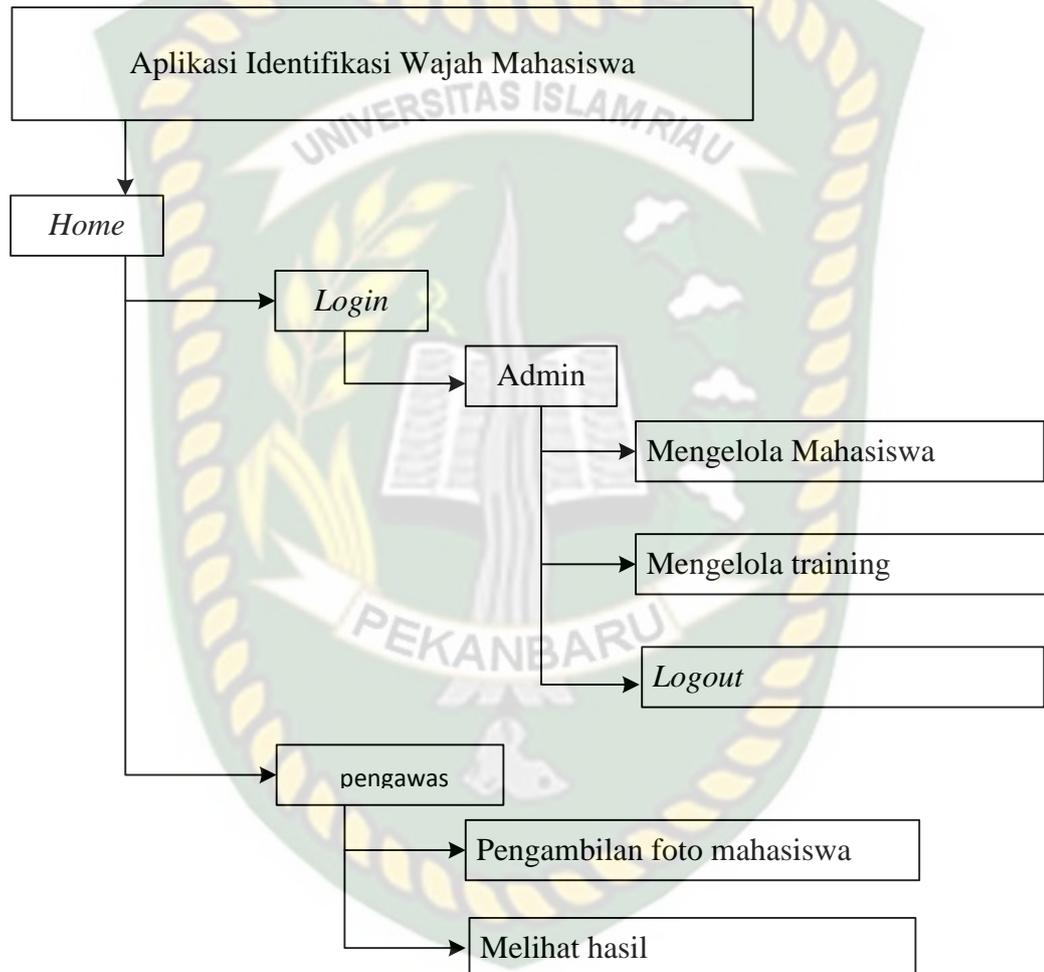
Tabel 3.3 Tabel identifikasi

No	<i>Field</i>	<i>Data Type</i>	<i>Size</i>	Ket
1	idIdentifikasi	Int	10	<i>Primary Key</i>
2	IdTraining	Int	10	<i>Foringn Key</i>
3	Fotoinput	Varchar	50	
4	Tanggal	Date		

3.4.7 Perancangan Antar muka

3.4.7.1 Struktur Menu Antar Muka Program

Perancangan Antar muka sistem pengenalan wajah mahasiswa dapat digambarkan dengan struktur menu seperti gambar 3.20.



Gambar 3.21 Struktur Menu Program

Pada gambar 3.21 menampilkan menu-menu yang terdapat pada aplikasi identifikasi wajah mahasiswa dengan rincian penjelasan sebagai berikut:

1. Bagi Admin

Admin melakukan *login* terlebih dahulu untuk dapat mengelola dan data training.

a Mengelola Mahasiswa

Menu ini berfungsi untuk admin melakukan penambahan, edit, dan hapus mahasiswa.

b Mengelola Data Training

Menu ini berfungsi untuk admin melakukan penambahan, edit, dan hapus data training.

2. Bagi Pengawas

Pengawas tidak perlu melakukan *login* seperti admin. Pengawas dapat langsung memilih menu yang ada di *home* yaitu menu pengenalan wajah dan melihat hasil pengenalan wajah mahasiswa.

a. Pengenalan Mahasiswa

Menu ini berfungsi untuk melakukan pengenalan wajah mahasiswa dengan menginputkan data foto mahasiswa.

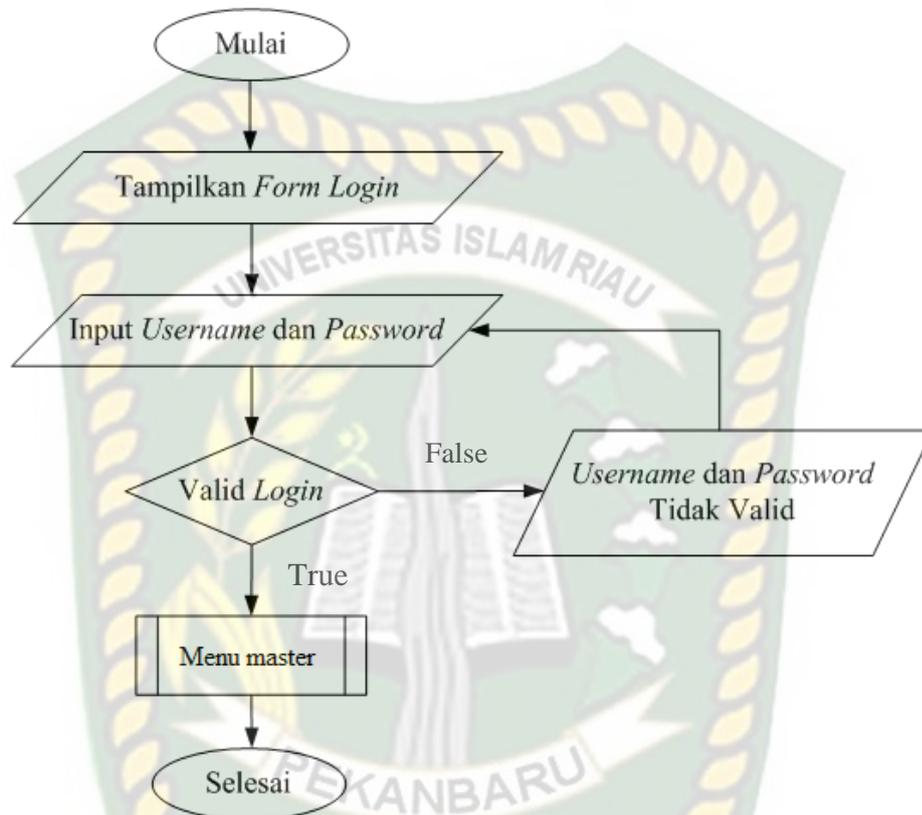
b. Hasil Pengenalan

Menu ini adalah untuk melihat hasil pengenalan mahasiswa.

3.4.8 Desain Logika Program

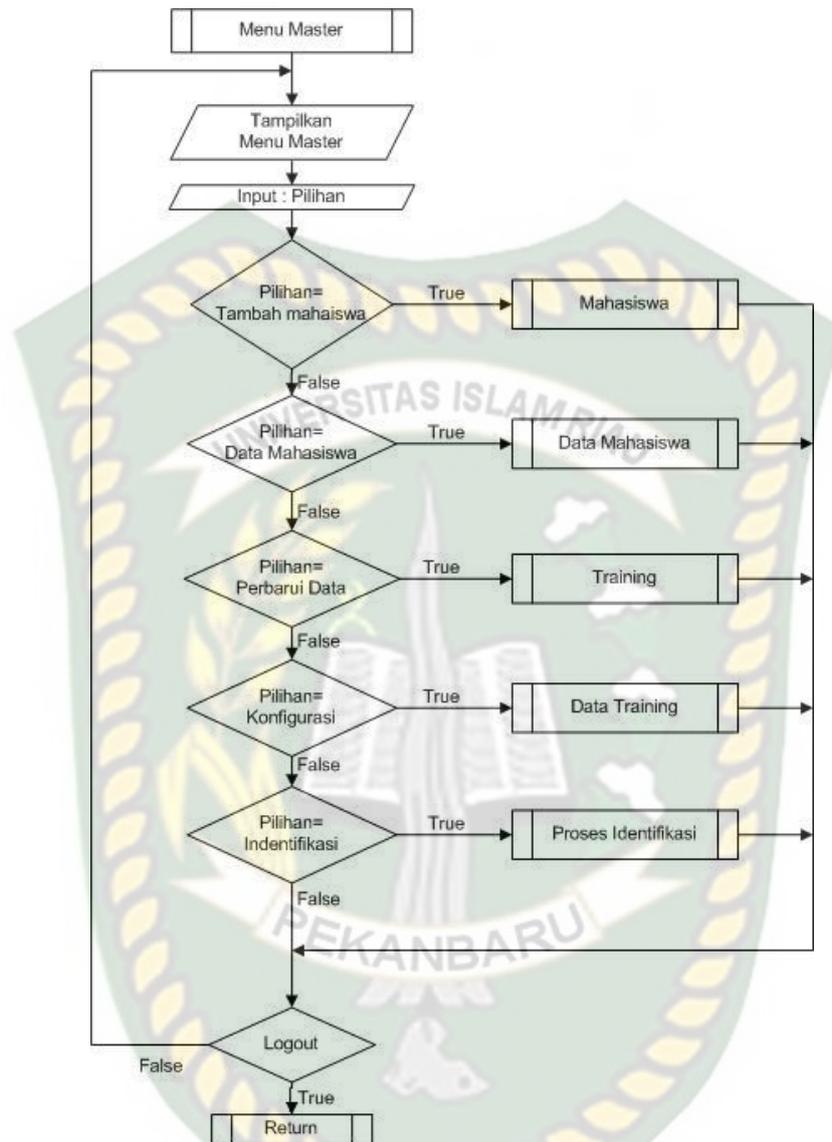
Dalam merancang sebuah sistem, pengembangan alur adalah hal yang sangat penting dalam memahami proses dari sebuah sistem. Pada tahap ini akan

digambarkan alur proses dalam pengenalan wajah mahasiswa dalam bentuk *Flowchart*.



Gambar 3.22 *Flowchart Login*

Ketika sistem pertama kali dijalankan akan menampilkan halaman utama dan pada halaman utama akan muncul menu *login*, *home*, mahasiswa, training, dan *about*. Ketika admin ingin menjalankan sistem admin harus melakukan *login* terlebih dahulu dan akan menampilkan form *login* seperti gambar 3.20. Ketika *login* berhasil maka akan diarahkan ke menu master untuk mengelola mahasiswa, mengelola training bisa dilihat pada gambar 3.22.



Gambar 3.23 Flowchart Menu Master

Pada rancangan gambar 3.23 setelah admin login selanjutnya dilakukan proses pemilihan menu yang terdiri dari menu mahasiswa dan menu training. Apabila memilih menu mahasiswa maka sistem mengeluarkan pengolahan data mahasiswa dan apabila memilih menu training maka akan keluar menu kelola data training.