

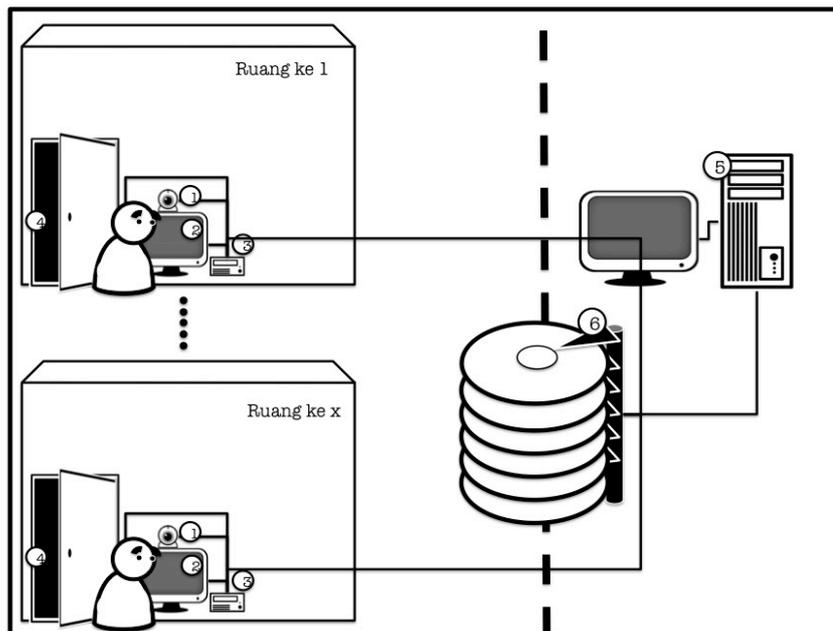
BAB 2

SISTEM REAL TIME AUTO DOOR-LOCK

Bab ini akan menjelaskan tentang arsitektur dari sistem, proses analisis kebutuhan dan desain dari perangkat lunak sistem, skema dari perangkat lunak sistem, serta proses konstruksi dari perangkat lunak sistem yang dikerjakan dalam tugas akhir ini.

2.1 Arsitektur Sistem *Real Time Auto Door-Lock*

Sistem *Real Time Auto Door-Lock* terdiri dari perangkat – perangkat yang terpisah, baik perangkat keras, maupun perangkat lunak yang saling berinteraksi satu dengan yang lain. Gambaran umum mengenai arsitektur sistem dapat dilihat pada gambar 2.1.



Gambar 2. 1 Gambaran Umum Arsitektur Sistem

Secara garis besar, arsitektur sistem *Real Time Auto Door-Lock* terbagi menjadi 6 bagian, yaitu:

1. Kamera. Perangkat ini berfungsi untuk mengambil data wajah pengguna yang akan masuk ke dalam ruangan. Kamera yang digunakan adalah kamera standar

khusus PC yang menggunakan koneksi USB. Kamera yang digunakan memiliki dimensi 320 * 240 piksel. Data wajah yang akan diambil adalah bagian wajah dari pengguna saja, yang akan memiliki dimensi 64 * 64 piksel.

2. *LCD touchscreen monitor*. *LCD touchscreen monitor* memiliki dua fungsi dalam sistem *Real Time Auto Door-Lock*. Fungsi pertama adalah sebagai tempat menampilkan antar muka dari perangkat lunak *Real Time Auto Door-Lock* dan keluaran dari kamera. Fungsi kedua adalah sebagai alat navigasi untuk sistem dengan kemampuan *touchscreen* yang terdapat dalam monitor ini.
3. *Mini PC*. Peran dari *Mini PC* adalah menjadi tempat untuk berjalannya perangkat lunak dari sistem *Real Time Auto Door-Lock*. Pada *mini PC* inilah dilakukan pengendalian kamera, pintu, serta pengujian data wajah dari pengguna.
4. *Pintu Otomatis*. Perangkat ini sebenarnya adalah sebuah pintu dengan mekanisme buka tutup yang menggunakan kekuatan magnet. Untuk mengendalikan keadaan buka atau tutup dari pintu, terdapat sebuah alat kontrol yang terhubung dengan *mini PC* lewat *port* paralel dari *mini PC* dan sumber listrik. Alat kontrol pintu akan membuka dan menutup pintu sesuai sinyal yang diberikan dari sistem. Jika data wajah pengguna yang ingin masuk terdaftar dalam data wajah yang memiliki akses ke ruang tersebut, maka sistem akan membukakan pintu bagi pengguna.
5. *Administrator PC*. Perangkat ini sebenarnya adalah sebuah PC yang berfungsi untuk mengatur semua *mini PC* yang ada di tiap ruang, serta dapat melakukan proses manajemen data pelatihan yang terdapat pada tempat penyimpanan. Pada administrator PC juga dapat dilakukan pendaftaran pengguna baru dari sistem.
6. *Penyimpanan data pelatihan*. Perangkat ini adalah tempat untuk mengumpulkan semua data penunjang dari sistem *Real Time Auto Door-Lock*. Data – data tersebut antara lain adalah, data wajah pengguna yang terdaftar,

data pengguna yang terdaftar pada setiap ruang, data pribadi dari tiap pengguna yang terdaftar.

2.2 Analisis Kebutuhan Perangkat Lunak Sistem *Real Time Auto Door-Lock*

Dalam pengembangan perangkat lunak, analisis kebutuhan dari sistem sangat diperlukan untuk memberikan dasar yang baik dalam proses perancangan atau desain dari sistem. Berikut adalah detil dari analisis kebutuhan dari perangkat lunak sistem *Real Time Auto Door-Lock*.

2.2.1 *Overview* perangkat lunak sistem

perangkat lunak dari sistem *Real Time Auto Door-Lock* dikembangkan menggunakan bahasa pemrograman Java. Alasan dari penggunaan bahasa pemrograman Java adalah kemudahan penggunaan yang ditawarkan dan dokumentasi yang tersedia lengkap untuk pencarian bantuan penyelesaian masalah.

Berikut adalah penjabaran arsitektur yang digunakan dalam pengembangan perangkat lunak sistem.

- *Java Development Kit* versi 1.6.0_11. Dasar dari pengembangan perangkat lunak sistem.
- *Java Media Framework API (JMF)* versi 2.1.1e. *framework* dari Java yang berguna untuk mengakses audio, video dan media *time-based* lain untuk aplikasi yang dibuat dengan Java. JMF dalam pengembangan perangkat lunak sistem *Real Time Auto Door-Lock* berguna untuk menyediakan akses ke kamera dari PC.
- *Java Parallel Port Library (Java parport)*. Merupakan sebuah *library* tambahan bagi Java yang menyediakan akses langsung kepada aplikasi ke *port* paralel dari komputer. Penggunaan *library* ini bertujuan untuk menyediakan akses kepada sistem untuk mengirim sinyal kepada pintu otomatis lewat *port* paralel dari komputer.

- *Userport Windows driver*. *Userport* adalah *kernel mode driver* bagi Windows NT/2000/XP yang memberikan program yang berjalan dalam mode *usermode*. *Driver* ini memungkinkan akses langsung ke perangkat keras layaknya akses perangkat keras pada Windows 95/98/ME.

2.2.2 Kebutuhan minimum dari sistem

Agar perangkat lunak sistem *Real Time Auto Door-Lock* dapat berjalan dengan baik, sistem secara keseluruhan sebaiknya memenuhi kebutuhan berikut:

Hardware:

- Prosesor 1.6 Ghz
- RAM 512 MB
- USB *Web Camera*
- *Port* Paralel untuk akses ke pintu otomatis

Software:

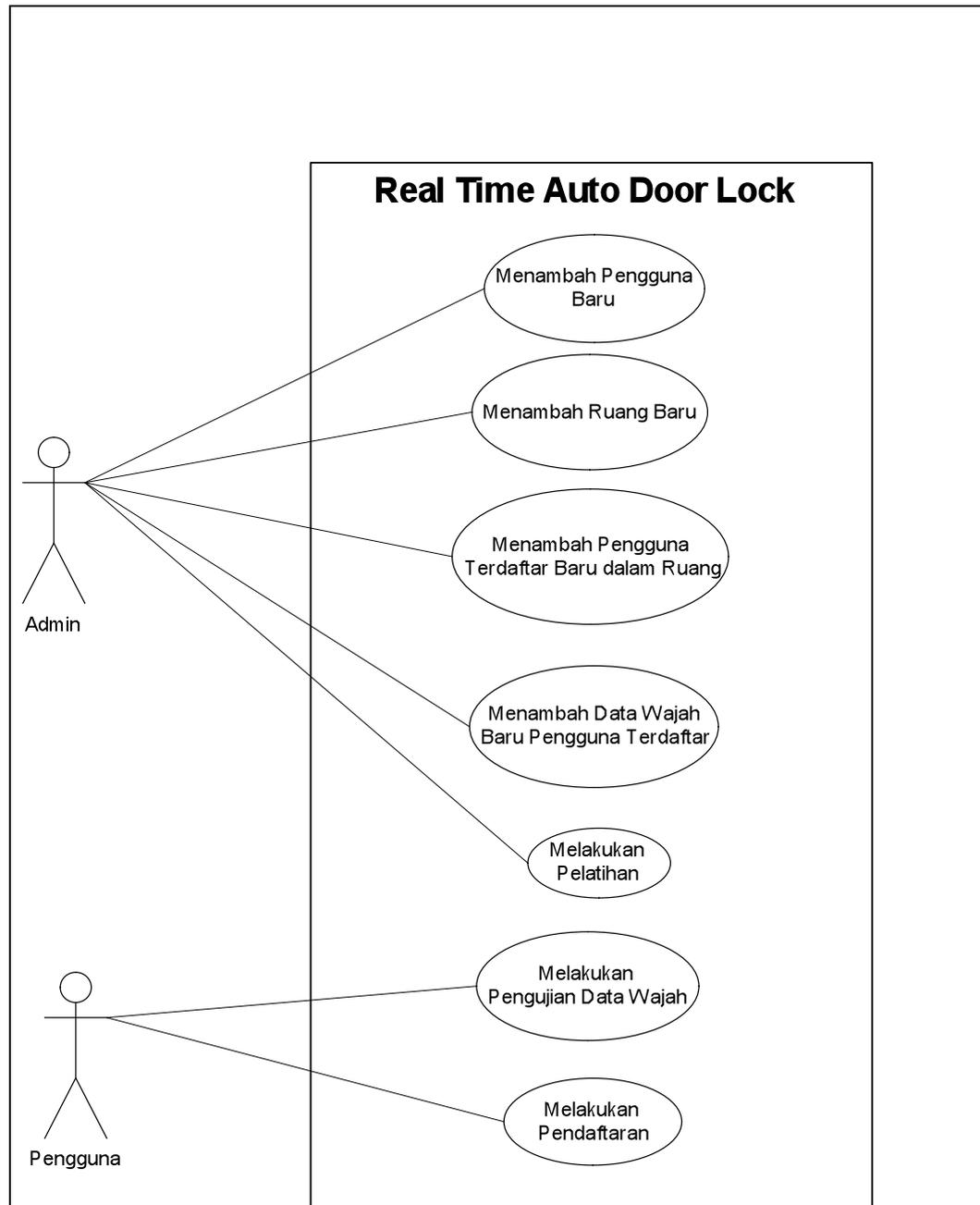
- Sistem operasi Windows XP Service Pack 2
- *Java Standard Development Kit* 1.6
- *Java Media Framework API* 2.1
- *Java Parallel Port library*
- *UserPort Windows Driver*

2.3 Desain perangkat lunak sistem *Real Time Auto Door-Lock*

Dalam pengembangan perangkat lunak yang baik, proses desain sangat menentukan keberhasilan dalam proses konstruksi dari perangkat lunak itu sendiri. Berikut adalah rangkaian desain dari perangkat lunak sistem *Real Time Auto Door-Lock*.

2.3.1 Pembuatan *Use Case Diagram*

Tujuan dari pembuatan *Use Case Diagram* adalah untuk mengetahui *functional requirement* yang terdapat dalam sistem. Dengan adanya *use case diagram*, diharapkan proses konstruksi sistem dapat lebih terarah dalam proses penyelesaian. Gambar 2.2 adalah *use case diagram* dari sistem.



Gambar 2. 2 *Use Case Diagram* dari Sistem

Berikut adalah penjelasan singkat mengenai setiap *use case*:

1. *Use case* Menambah Pengguna Baru.

Use case ini menjelaskan fungsi untuk menambah pengguna baru dalam data pengguna terdaftar. Data pengguna yang akan ditambahkan merupakan keluaran dari *use case* Melakukan Pendaftaran.

2. *Use case* Menambah Ruang Baru.

Use case ini menjelaskan fungsi untuk menambah ruangan baru dalam sistem.

3. *Use case* Menambah Pengguna Terdaftar Baru dalam Ruang.

Use case ini menjelaskan fungsi untuk menambah pengguna terdaftar dalam sebuah ruangan yang sudah ada dalam sistem.

4. *Use case* Menambah Data Wajah Baru Pengguna Terdaftar.

Use case ini menjelaskan fungsi untuk menambah data wajah pelatihan dari seorang pengguna terdaftar.

5. *Use case* Melakukan Pelatihan.

Use case ini berfungsi untuk melakukan pelatihan terhadap semua data wajah dari semua pengguna terdaftar pada suatu ruangan.

6. *Use case* Melakukan Pengujian Data Wajah.

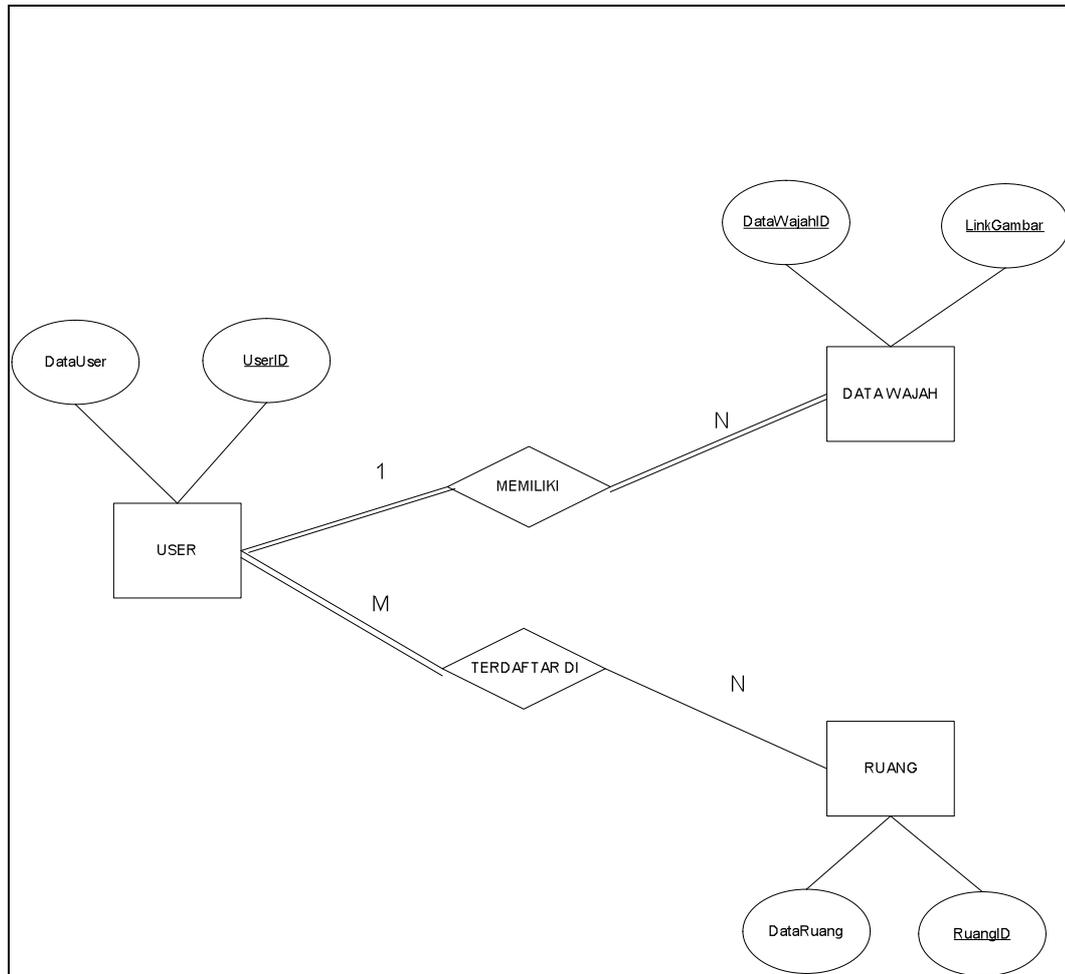
Use case ini berfungsi untuk melakukan pengujian terhadap data wajah dari pengguna yang ingin masuk ke dalam suatu ruangan.

7. *Use case* Melakukan Pendaftaran.

Use case ini berfungsi untuk memfasilitasi pengguna baru yang ingin mendaftarkan diri dalam sistem. Keluaran dari *use case* ini akan diproses oleh *use case* Menambah Pengguna Baru.

2.3.2 Entity Relationship Diagram (ERD)

Pembuatan ERD bertujuan untuk memperjelas entitas – entitas yang terdapat dalam keseluruhan sistem beserta atribut – atributnya. Gambar 2.3 berikut adalah ERD dari sistem.



Gambar 2.3 Entity Relationship Diagram dari Sistem

Perangkat lunak dari sistem *Real Time Auto Door-Lock* memiliki tiga entitas, yaitu:

- USER, merupakan entitas yang merepresentasikan pengguna dalam sistem. Entitas USER memiliki dua atribut, yaitu:

- UserID, *primary key* dari USER. Merupakan identitas unik dari tiap *user* dari sistem.
- DataUser, merupakan data detil tentang *user* yang bersangkutan. Karena penjabaran detil data yang dibutuhkan dalam sistem belum diketahui secara jelas, maka dibuat satu atribut yang mewakili keseluruhan data dari seorang pengguna, yaitu atribut DataUser.
- DATA WAJAH, merupakan entitas yang merepresentasikan data wajah dari tiap user/pengguna yang terdaftar. Atribut dari DATA WAJAH adalah:
 - DataWajahID, *primary key* dari DATA WAJAH. Identitas unik tiap entitas.
 - LinkGambar. Atribut ini menyimpan *link* atau *path* menuju data gambar yang tersimpan dalam basis data penyimpanan data.
- RUANG, merupakan entitas yang merepresentasikan satu ruangan dalam sistem. Atribut dari entitas RUANG ada dua, yaitu:
 - RuangID, *primary key* dari entitas RUANG.
 - DataRuang, detil dari ruang tersebut. Layaknya Entitas USER, detil dari data yang ada dalam sebuah entitas RUANG belum didapatkan secara jelas, sehingga dibuat sebuah atribut untuk merepresentasikan keberadaan data dalam ERD.

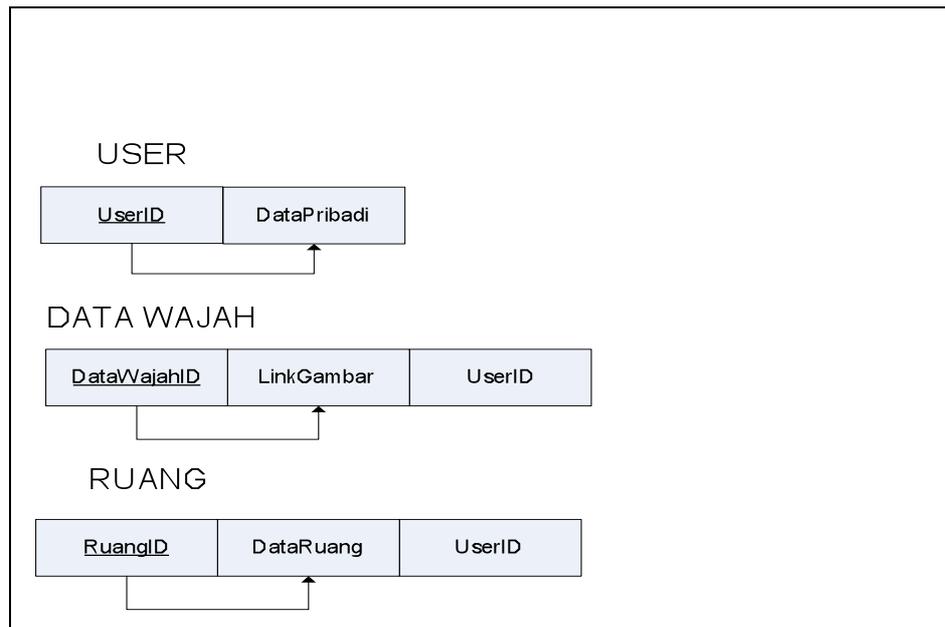
Relasi yang terdapat dalam sistem adalah:

- Relasi antara USER dengan RUANG yang bernama TERDAFTAR DI. Relasi ini bersifat total pada entitas USER dan parsial pada entitas RUANG. Dengan kata lain, setiap USER pasti terdaftar pada minimal satu buah RUANG, tetapi tidak semua RUANG terdapat satu buah USER tertentu.
- Relasi antara USER dengan DATA WAJAH yang bernama MEMILIKI. Relasi yang bersifat total pada kedua entitas. Hal ini berarti bahwa semua

USER pasti memiliki setidaknya satu buah DATA WAJAH, dan setiap DATA WAJAH pasti dimiliki oleh satu buah USER.

2.3.3 Pembuatan *Data Model*

Pembuatan data model adalah kelanjutan dari proses desain ERD sistem. *Data model* dari sistem dapat dilihat pada gambar 2.4 berikut.

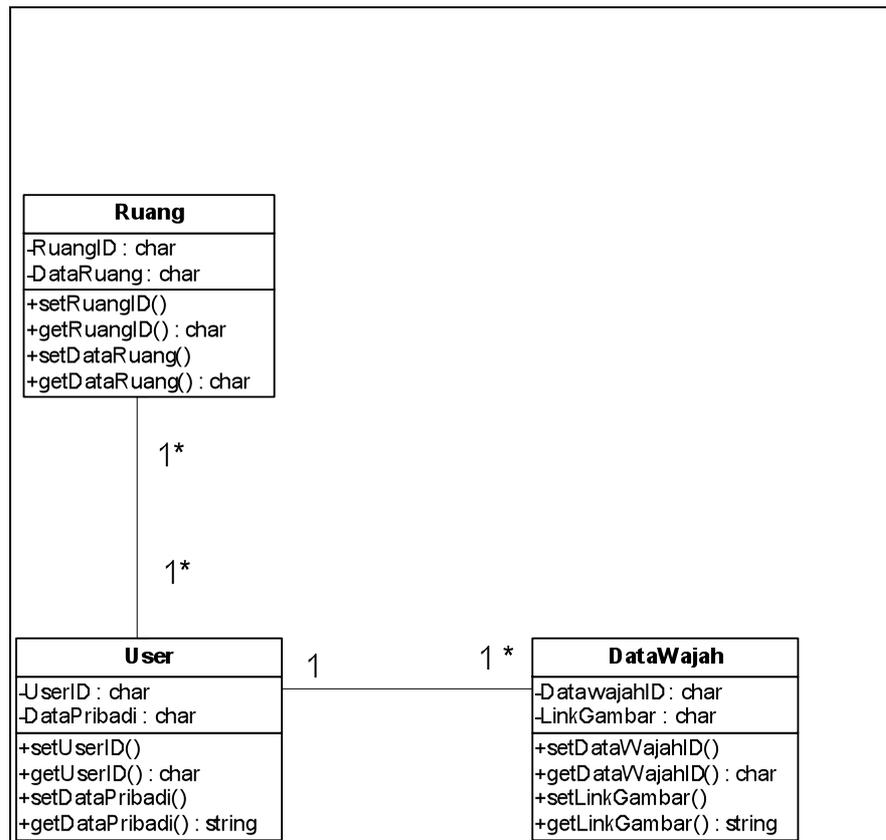


Gambar 2. 4 Data Model dari Sistem

Pada tabel RUANG dan DATA WAJAH terdapat satu buah *foreign key* UserID yang merujuk pada UserID pada tabel USER. Hal ini terjadi disebabkan adanya relasi antara USER dan DATA WAJAH serta RUANG pada ERD.

2.3.4 Pembuatan *Class Diagram*

Pemodelan data dari sistem juga dilakukan dengan pembuatan *class diagram*. Berikut adalah gambar dari *class diagram* sistem.



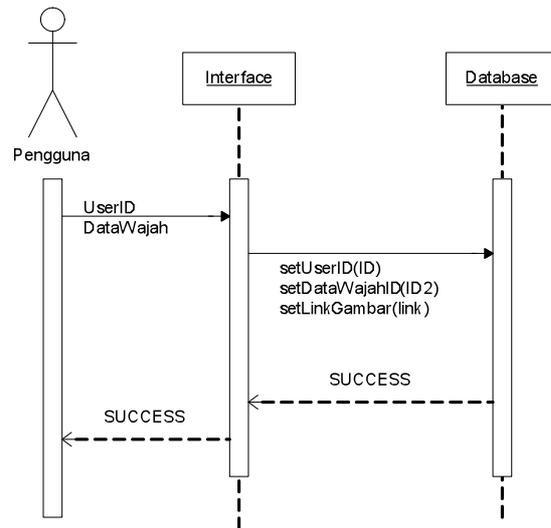
Gambar 2. 5 *Class Diagram* dari Sistem

Hubungan antara objek Ruang dengan User adalah setiap Ruang bisa memiliki satu atau lebih User yang terdaftar, dan setiap User dapat terdaftar pada satu atau lebih Ruang. Hubungan antara User dan DataWajah adalah, setiap User memiliki satu atau lebih DataWajah tetapi satu DataWajah hanya dimiliki tepat satu User.

2.3.5 Pembuatan *Sequence Diagram*

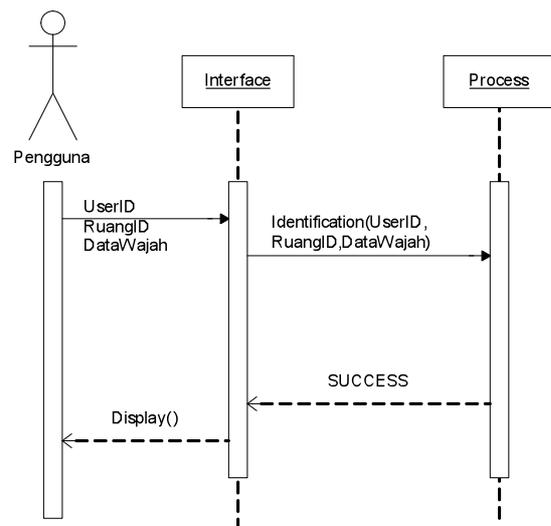
Cara pemodelan proses dari sistem yang akan dibuat adalah dengan membuat *sequence diagram*. Dari keseluruhan *use case* yang ada, hanya akan dibuat dua *sequence diagram*. Hal ini disebabkan oleh terbatasnya waktu pengembangan dan kurangnya sumber daya manusia dalam proses konstruksi.

Gambar 2.6 adalah *sequence diagram* dari proses melakukan penambahan data wajah baru pada pengguna terdaftar.



Gambar 2. 6 *Sequence Diagram* Penambahan Data Wajah Baru pada Pengguna Terdaftar

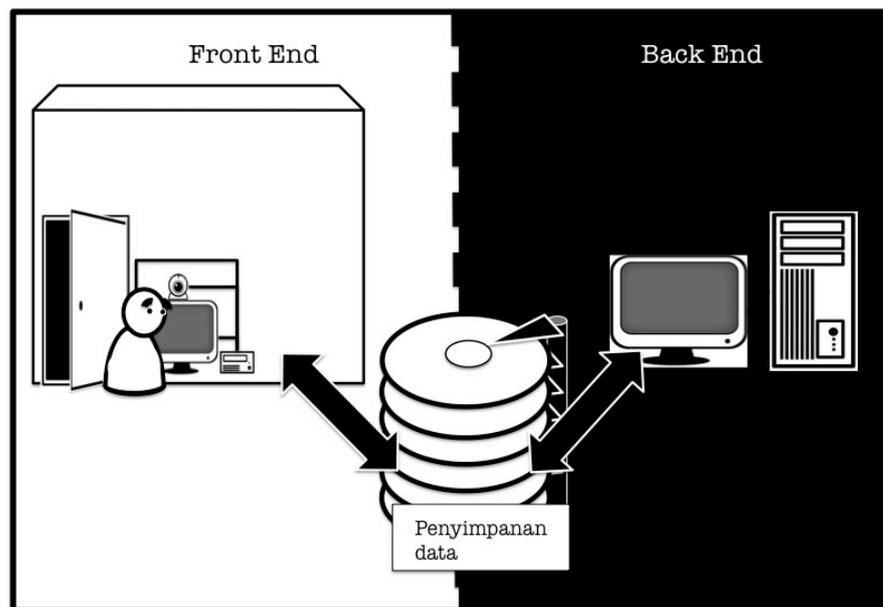
Gambar 2.7 adalah *sequence diagram* dari *use case* melakukan pengujian data wajah.



Gambar 2. 7 *Sequence Diagram* Pengujian Data

2.4 Skema Perangkat Lunak Sistem *Real Time Auto Door-Lock*

Perangkat lunak dari sistem *Real Time Auto Door-Lock* dapat terbagi menjadi dua bagian utama, yaitu bagian *front-end* dan bagian *back-end*. Bagian *front-end* adalah bagian sistem yang terinstalasi pada mini PC di setiap ruangan. Bagian *back-end* adalah bagian dari sistem yang terinstalasi pada Administrator PC. Gambar 2.8 adalah ilustrasi dari skema perangkat lunak sistem *Real Time Auto Door-Lock*.



Gambar 2. 8 Skema Perangkat Lunak Sistem

2.4.1 *Front-end*

Berikut adalah penjabaran dari modul –modul yang terdapat pada *front-end* sistem:

- Modul *webcam controller*. Modul ini berfungsi sebagai alat kontrol dari *webcam*. *Webcam controller* akan menangkap data masukan dari wajah pengguna yang ingin masuk dan meneruskan data masukan ke modul *Image Processing*.
- Modul *Image Processing*. Fungsi utama dari modul ini adalah melakukan tahap pemrosesan awal terhadap data masukan sehingga data masukan

menjadi data wajah siap olah. Modul *Image Processing* terdiri dari tiga fungsi utama, yaitu:

- *Cropping*. Data masukan dari *webcam controller* berukuran $320 * 240$ piksel dan juga menyimpan data gambar latar belakang dari pengguna yang tidak bermanfaat dalam proses indentifikasi wajah. Oleh karena itu, dilakukan proses *cropping* dari data masukan sehingga menghasilkan data masukan yang hanya mengandung gambar wajah dari pengguna yang ingin masuk. Dimensi data sesudah proses *cropping* adalah $150 * 150$ piksel.
- *Dimension reduction*. Demi menghemat sumber daya dari mini PC yang terbatas, maka dilakukan proses reduksi dari dimensi data masukan dari ukuran $150 * 150$ piksel menjadi data berukuran $64 * 64$ piksel.
- $RGB \rightarrow Grayscale$ *converter*. Proses identifikasi menggunakan data tingkat keabuan (*grayvalue*) dari data wajah, maka perlu dilakukan penggantian mode warna dari data yang awalnya RGB menjadi mode warna *grayscale*.

Setelah melewati tiga proses tersebut, maka data masukan dari *Webcam Controller* menjadi data wajah siap olah untuk proses identifikasi.

- Modul *Training*. Modul ini berfungsi untuk melakukan pelatihan terhadap data wajah pengguna yang terdaftar pada ruangan yang bersangkutan.
- Modul Aplikasi. Modul ini berfungsi untuk melakukan proses pengujian terhadap data wajah siap olah keluaran dari modul *Image Processing*. Pengujian dilakukan berdasarkan pelatihan yang telah dilakukan terlebih dahulu oleh modul *Training*. Hasil dari modul aplikasi adalah identifikasi dari pengguna yang ingin memasuki ruangan, apakah dikenali sebagai pengguna yang terdaftar atau tidak.

- Modul *Door-Lock Controller*. Fungsi utama dari modul ini adalah memberikan sinyal kepada pintu otomatis sesuai dengan hasil identifikasi yang dilakukan oleh modul Aplikasi. Jika hasil identifikasi menyatakan pengguna yang ingin masuk ternyata adalah pengguna terdaftar, maka *Door-Lock Controller* akan mengirimkan sinyal untuk membuka pintu, begitu pula sebaliknya.

2.4.2 *Back-end*

Modul yang terdapat pada bagian *back-end* adalah modul penambahan data baru. Modul ini berfungsi untuk menambahkan data wajah baru dari pengguna yang terdaftar pada sistem. Pada modul ini juga terdapat modul *Webcam Controller* dan modul *Image Processing* seperti pada bagian *front-end*

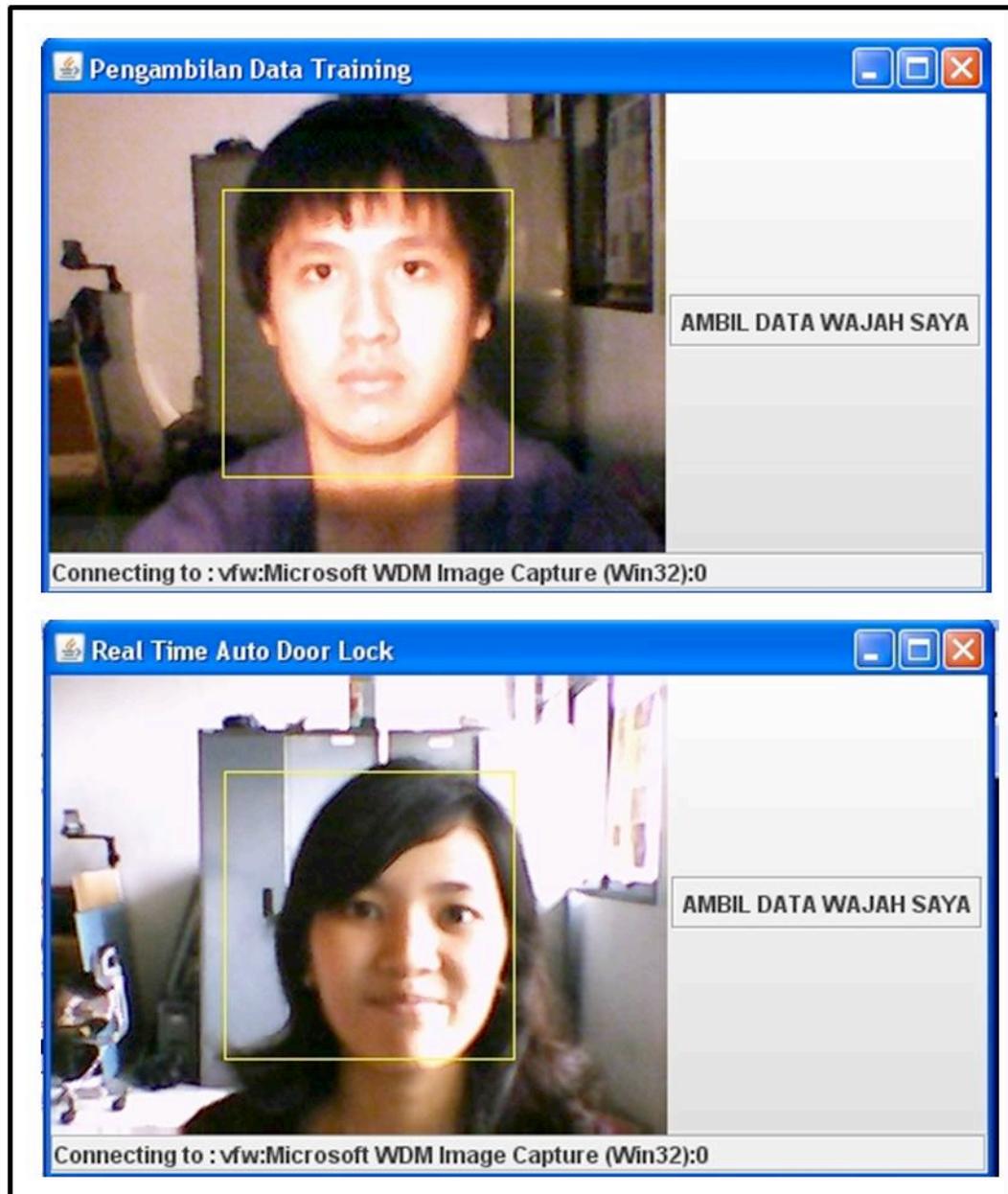
2.5 Konstruksi Perangkat Lunak Sistem

Pada tugas akhir ini, tidak semua fungsionalitas yang dijabarkan dalam proses desain perangkat lunak sistem diselesaikan. Hanya dua fungsi utama dari perangkat lunak sistem yang akan dikonstruksi. Fungsi – fungsi tersebut adalah (1) menambah data wajah baru pada pengguna yang sudah terdaftar dan (2) melakukan pengujian pada data wajah. Kedua fungsi utama ini adalah fungsi – fungsi yang dibutuhkan untuk menjalankan pengujian data wajah dalam sistem, sehingga bisa didapatkan kinerja dari algoritma pengenalan wajah yang diimplementasikan. Proses konstruksi dari perangkat lunak sistem terbagi sesuai dengan modul – modul yang telah dijelaskan dalam subbab 2.4.

2.5.1 Modul *Webcam Controller*

Fungsi utama modul ini adalah sebagai alat kontrol dari kamera. *Webcam controller* akan menangkap data masukan dari wajah pengguna yang ingin masuk dan meneruskan data masukan ke modul *Image Processing*. Pada proses konstruksi, digunakan Java Media Framework (JMF) untuk mengakses *webcam* dari PC. JMF memungkinkan pembuatan sebuah objek `player` yang berfungsi untuk mengakses media audio atau video. Pada keluaran dari kamera pada layar, diberikan sebuah bingkai yang bertujuan untuk memberikan arahan kepada

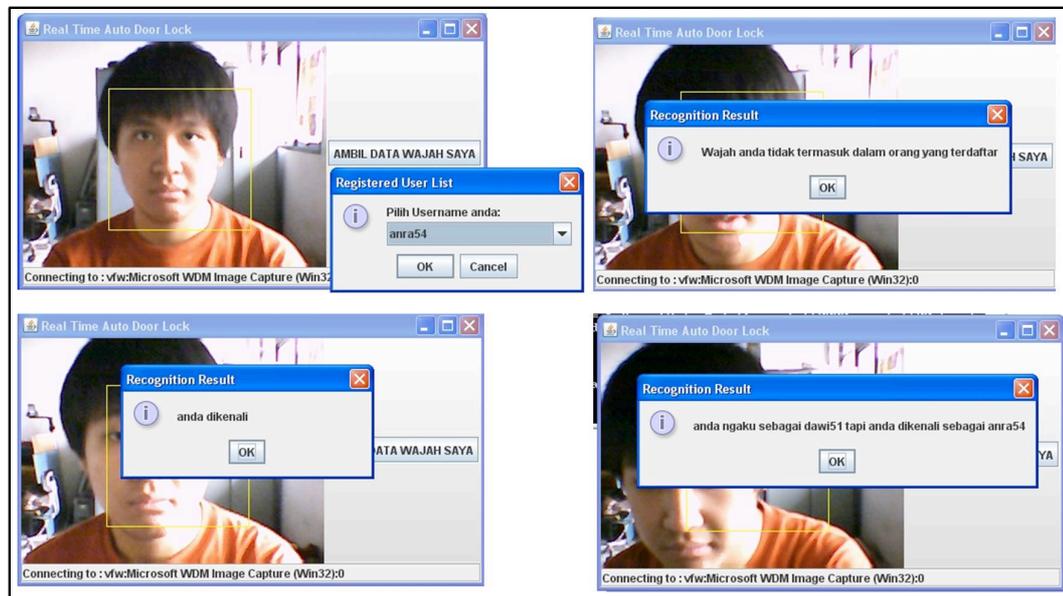
pengguna agar wajah pengguna tepat pada posisi yang dibutuhkan pada proses *cropping*.



Gambar 2. 9 Screenshot Antarmuka Perangkat Lunak

Untuk penerapan bingkai pada *Webcam Controller*, digunakan metode `paintComponent` dari kelas `Graphics`. Gambar 2.9 adalah *screenshot* dari *Webcam Controller* bagian pengambilan data wajah dari sistem pada bagian *back-end* (atas) dan bagian *front-end* (bawah), sedangkan gambar 2.10 adalah rangkaian

antarmuka *front-end* pada saat pengambilan data, dan pemberitahuan hasil pengenalan.



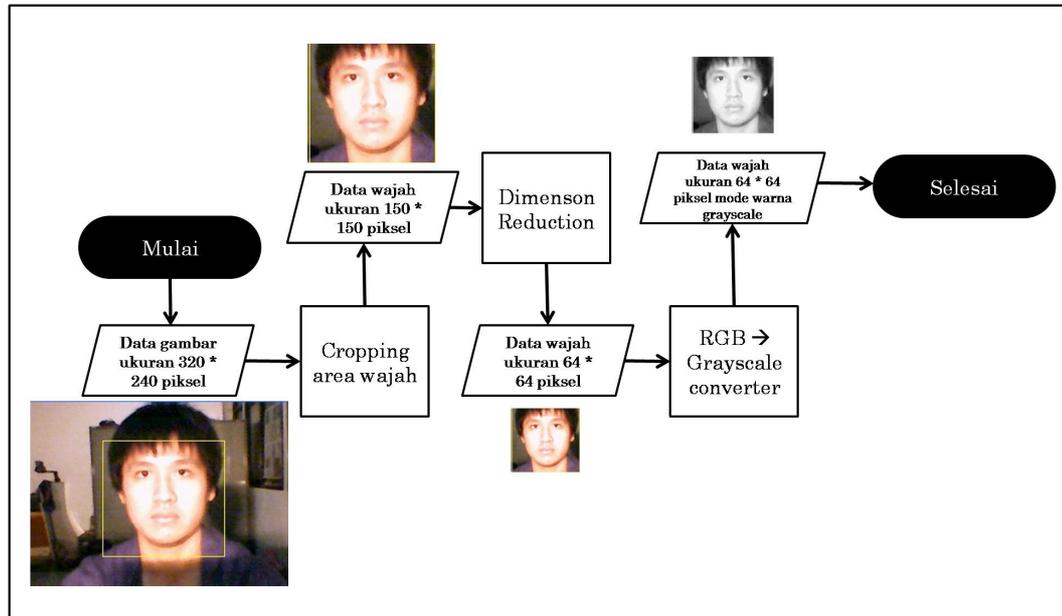
Gambar 2. 10 Rangkaian Screenshot Antarmuka Bagian Front-end

2.5.2 Modul *Image Processing*

Fungsi utama dari modul ini adalah melakukan proses pada data keluaran dari *Webcam Controller* sehingga menjadi data wajah siap olah. Data wajah siap olah adalah gambar yang hanya mengandung data wajah dari pengguna yang berukuran 64 * 64 piksel, bermode warna *grayscale* dan disimpan sebagai berkas JPEG. Gambar 2.10 merupakan diagram alur dari proses yang dilakukan dalam modul *Image Processing*.

Untuk fungsi *cropping* digunakan kelas `CropImageFilter` untuk menghasilkan data wajah yang berukuran 150 * 150 piksel. Fungsi *dimension reduction* ditangani oleh metode `getScaledInstance` pada kelas `Image`. Metode yang digunakan untuk pereduksian dimensi adalah metode standar bawaan dari Java, yang ditandai dengan parameter `SCALE_SMOOTH` pada pemanggilan metode `getScaledInstance`. Parameter `SCALE_SMOOTH` berarti penggunaan algoritma standar dari Java yang lebih mementingkan kehalusan dari hasil reduksi daripada kecepatan eksekusi.

Prose terakhir dari modul *Image Processing* adalah perubahan mode warna gambar dari mode warna RGB menjadi mode warna *grayscale*. Perubahan mode warna gambar menggunakan metode `getInstance` dari kelas `ColorSpace`.



Gambar 2. 11 Diagram Alur dari *Image Processing*

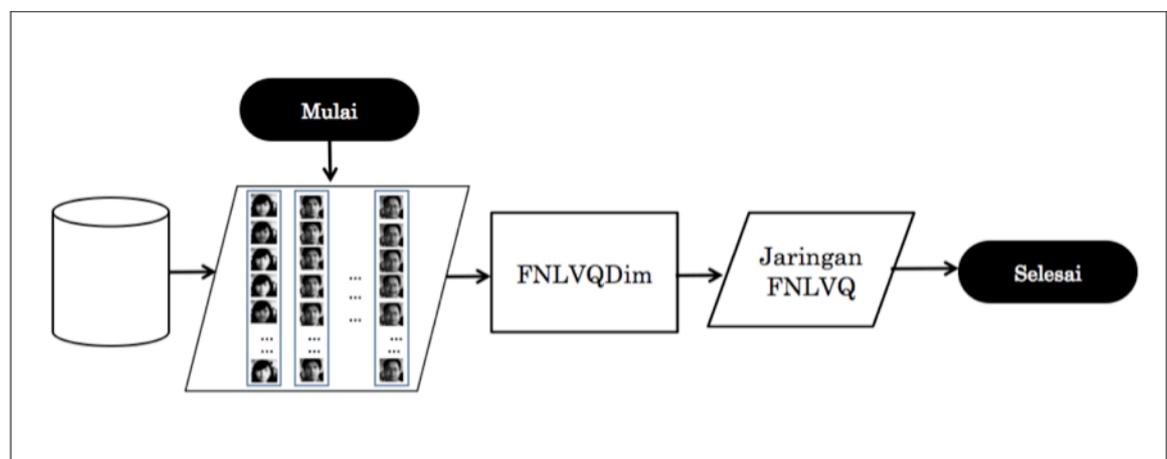
2.5.3 Modul *Training*

Modul training memiliki tiga fungsi utama yaitu, (1) pengumpulan data pelatihan, (2) ekstraksi data *gray value* dari tiap gambar menjadi data angka, dan (3) pelatihan jaringan dengan data pelatihan yang sesuai. Penjelasan dari fungsi – fungsi tersebut adalah sebagai berikut:

1. Pengumpulan data *link* menuju gambar dari data – data wajah dari pengguna yang akan dilakukan pelatihan. Pengumpulan data *link* menggunakan metode `getjpgFileList` pada kelas `GeneralFileHandler`.
2. Semua data gambar diolah untuk mendapatkan data *grayvalue* masing – masing piksel dari gambar lalu disimpan dalam sebuah variabel bertipe `double`. Pengambilan data *grayvalue* ini menerapkan metode `getImageInputs` dari kelas `ImageProcess`.

3. Data dari *grayvalue* dari semua data wajah pelatihan lalu akan dimasukkan dalam pelatihan dengan menggunakan algoritma *FNLVQ dimension based* yang akan dijelaskan lebih lanjut pada bab 3.

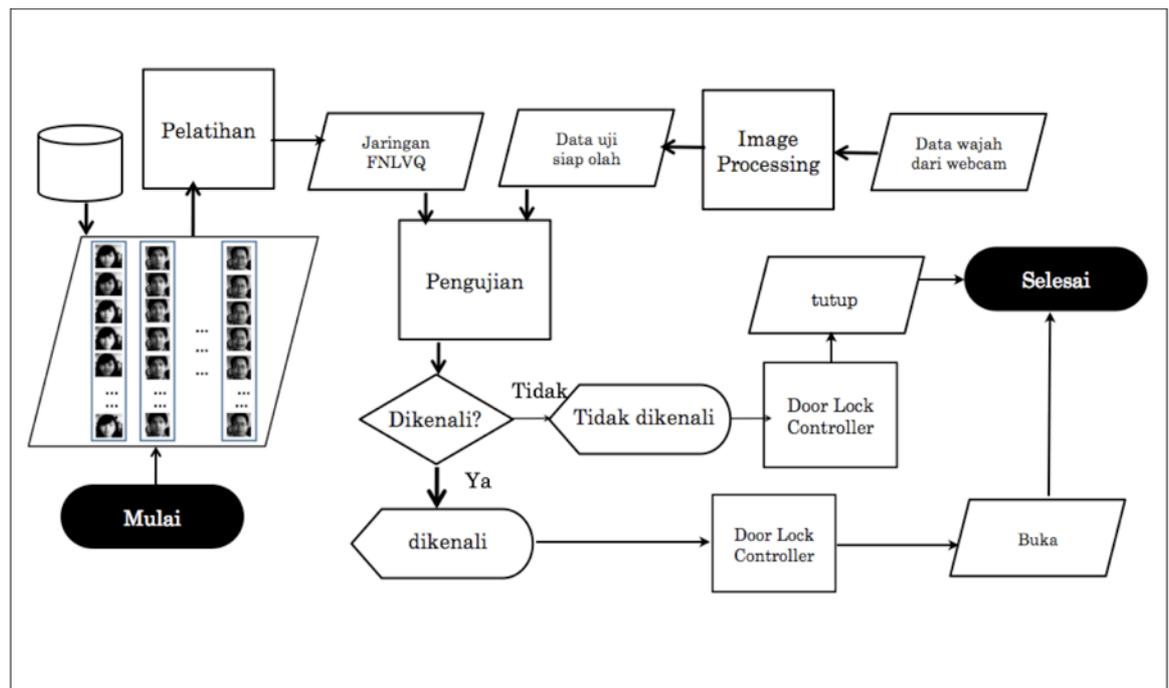
Setelah melalui tiga tahap tersebut, maka sudah terbentuk jaringan yang siap menerima masukan data wajah untuk dilakukan pengujian. Pada penerapannya, modul *Training* merupakan bagian dari modul aplikasi sehingga jika dilihat pada diagram alur dari modul aplikasi, modul *Training* merupakan salah satu tahapan didalam modul Aplikasi. Gambar 2.12 adalah diagram alur dari modul *Training*.



Gambar 2. 12 Diagram Alur Modul *Training*

2.5.4 Modul Aplikasi

Modul Aplikasi memiliki fungsi utama untuk melakukan pengujian data wajah dari pengguna yang ingin memasuki ruangan. Keluaran dari modul Aplikasi akan diolah oleh modul *Door-Lock Controller* yang mengendalikan buka – tutup pintu. Gambar 2.13 adalah diagram alur dari proses yang terjadi dalam modul Aplikasi.



Gambar 2. 13 Diagram Alur Modul Aplikasi

2.5.5 Modul *Door-Lock Controller*

Fungsi utama dari modul ini adalah memberikan sinyal kepada pintu otomatis sesuai dengan hasil identifikasi yang dilakukan oleh modul Aplikasi. Jika hasil identifikasi menyatakan pengguna yang ingin masuk ternyata adalah pengguna terdaftar, maka modul *Door-Lock Controller* akan mengirimkan sinyal untuk membuka pintu, begitu pula sebaliknya. Pada penerapan dalam perangkat lunak, modul *Door-Lock Controller* terintegrasi dalam modul Aplikasi.

Pintu otomatis terhubung ke PC melalui *port* paralel, atau dikenal juga sebagai *port printer*. Gambar 2.14 adalah gambar dari *port* paralel dari PC.



Gambar 2. 14 Port Paralel pada Komputer.

Untuk mendapatkan akses menuju port paralel, digunakan *library* Java tambahan bernama *Java Parallel Port Library* dan *driver* khusus untuk sistem operasi Windows XP yang memungkinkan program dari pengguna memiliki akses langsung menuju *port* paralel. *Driver* ini bernama Userport windows driver.

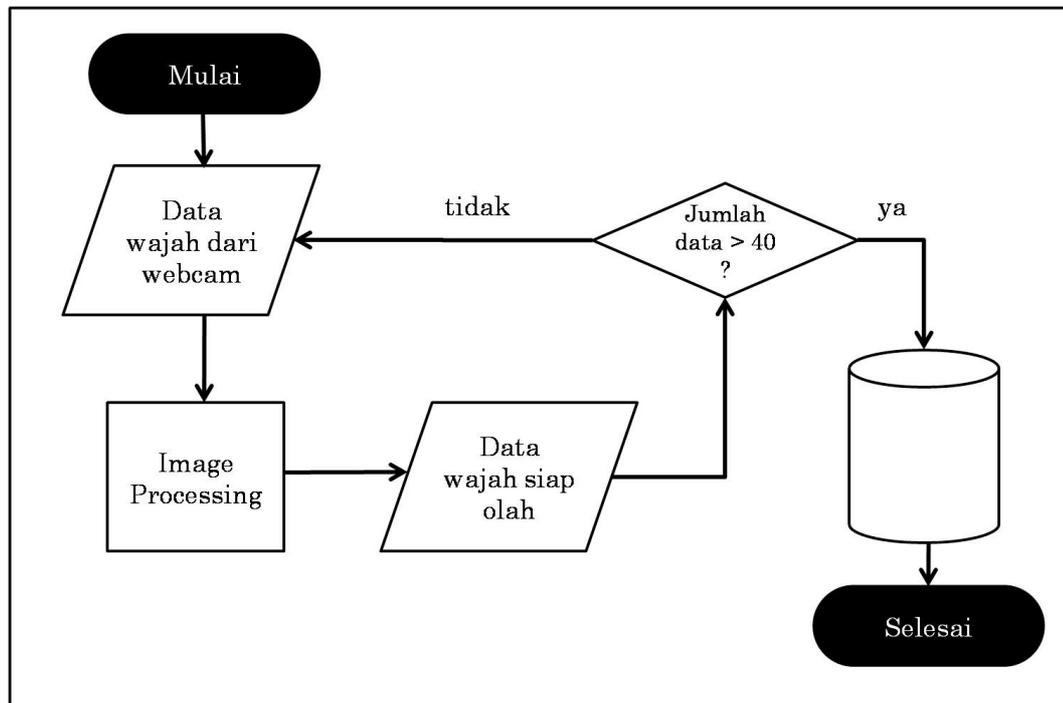
Penerapan *library Parallel Port* sangat sederhana, *library* ini hanya cukup diimpor kedalam kode program. Representasi port paralel dalam Java adalah objek `ParallelPort`. Untuk memberikan sinyal kepada pintu otomatis, cukup menggunakan metode `write` pada objek `ParallelPort`.

Pada saat pertama kali berjalan, modul ini akan memberikan sinyal untuk menutup pintu untuk memastikan pintu dalam keadaan tertutup. Selanjutnya, jika menerima masukan dari hasil pengenalan yang membolehkan pengguna masuk, maka akan dikirimkan sinyal untuk membuka pintu kepada pintu otomatis selama rentang waktu 5 detik. Setelah 5 detik, maka *Door-Lock Controller* akan mengirim sinyal untuk menutup pintu kembali.

2.5.6 Modul Penambahan Data Baru

Fungsi utama dari modul ini adalah menambahkan data wajah baru dari pengguna yang telah terdaftar. Dasar dari modul ini adalah modul *Webcam Controller* yang akan mengambil data wajah dari pengguna sebanyak 40 gambar sekaligus dengan rentang 0.7 detik untuk setiap pengambilan satu data wajah.

Data wajah yang telah diambil akan disimpan dalam penyimpanan data sesuai dengan userID dari pengguna yang melakukan penambahan data wajah baru. Gambar 2.15 adalah diagram alur dari modul penambahan data baru.



Gambar 2. 15 Diagram Alur Penambahan Data Baru