BAB II
LANDASAN TEORI

2.1. Tinjauan Perusahaan

Studi kasus dilaksanakan di X-Target Cabang Abdurahman Saleh yaitu suatu bimbingan belajar X-Target sebagai kantor cabang yang berlokasi pada jl Abdurahman saleh no 41 Bandung.

2.1.1. Sejarah Singkat Perusahaan

Berawal dari Bimbingan *Private* pada saat masih kuliah, dan juga latar belakang karir setelah lulus dari Teknik Sipil ITB, Mohammad Anwar (CEO X-Target) membangun Bimbingan Belajar untuk SD, SMP dan SMA yang menerapkan sistem pembelajaran yang diberi nama S2LM *Soft Skill Learning Method.*

S2LM *Soft Skill Learning Method* memiliki 4 kekuatan memberdayakan. Tujuan adalah arah, tujuan adalah niat dan tujuan adalah rencana. Rencana adalah awal penulisan takdir, realisasinya adalah doa dan ikhtiar serta ujungnya adalah kehendak Allah. Gagal dalam rencana adalah sama dengan merencanakan kegagalan. Awali hari dengan rencana akhiri hari dengan prestasi dan evaluasi.

Empat kekuatan memberdayakan tersebut adalah :

1. Kekuatan Tujuan

Pada Kekuatan 1 ini, setiap siswa akan menuliskan *Mission Statement* dan juga menuliskan *Daily* Agenda untuk mencapai tujuan tersebut.

2. Kekuatan Ikhtiar

Pada Kekuatan 2 ini, setiap siswa disadarkan bahwa tiap jiwa adalah unik, memiliki ciri, watak dan caranya sendiri dalam belajar. Belajar bukanlah merubahnya melainkan menyempurnakannya.

3. Kekuatan Evaluasi

Pada Kekuatan 3, setiap siswa dibangunkan untuk memahami bahwa setiap jiwa hendaklah menghisab diri sebelum hari penghisaban sesungguhnya tiba.

4. Kekuatan Perbaikan

Kekuatan 4 atau Kekuatan Perbaikan, adalah setiap siswa diharapkan bisa lebih baik, yaitu dari tidak bisa menjadi bisa, dari tidak tahu menjadi tahu, dan dari baik menjadi lebih baik. Tidak peduli berapa kali kita terjatuh yang terpenting adalah setiap kali kita terjatuh harus dipastikan kita selalu bangkit berdiri.

X-Target berdiri pada Juni 2013 dengan diawali di dua kota, yaitu Bandung dan Bekasi, adapun kenapa didirikan di dua kota ini, karena Mohammad Anwar (CEO) X-Target memang menyelesaikan pendidikan dasar sampai dengan lulus kuliah adalah di dua kota ini.

2.1.2. Visi dan Misi

Dalam hal pencapaian suatu tujuan di perlukan suatu perencanaan dan tindakan nyata untuk dapat mewujudkannya, secara umum bisa di katakan bahwa Visi dan Misi adalah suatu konsep perencanaan yang di sertai dengan tindakan sesuai dengan apa yang di rencanakan untuk mencapai suatu tujuan. Bagi mereka yang berkecimpung dalam kegiatan organisasi tentu tidak asing dengan kalimat Visi dan Misi di karenakan suatu organisasi, kelompok atau badan suatu instansi pasti memiliki Visi dan Misi untuk mewujudkan tujuannya.

Berdasarkan pendapat dari Wibisono, Visi merupakan rangkaian kalimat yang menyatakan cita-cita atau impian sebuah organisasi atau perusahaan yang ingin dicapai di masa depan (Wibisono 2006, p. 43). Atau dapat dikatakan bahwa visi merupakan pernyataan want to be dari organisasi atau perusahaan. Visi adalah cara pandang jauh ke depan kemana organisasi harus dibawa agar dapat eksis, antisipatif dan inovatif. Visi adalah suatu gambaran yang menantang tentang keadaan masa depan yang diinginkan oleh organisasi.

Dapat disimpulkan bahwa Visi adalah cita-cita dari sebuah organisasi atau perusahaan yang ingin dicapai di masa depan untuk menjamin kelestarian dan kesuksesan jangka panjang.

Berikut ini adalah Visi dari bimbingan belajar X-Target, yaitu :

“Dengan seizin Allah, X-Target mampu membangun insan yang senantiasa membekali diri dengan takwa, menggapai cita dengan rencana dan Ikhtiar,

menyandarkan soal pada tawakal. Sehingga terbangunlah setiap Insan yang cerdas."

Menurut Mulyadi Misi adalah jalan pilihan (*the chosen track*) suatu organisasi untuk menyediakan produk/jasa bagi customer-nya (Mulyadi 2001, p. 798). Sedangkan menurut Wibisono Misi merupakan rangkaian kalimat yang menyatakan tujuan atau alasan eksistensi organisasi, yang memuat apa yang disediakan oleh perusahaan kepada masyarakat, baik berupa produk ataupun jasa (Wibisono 2006, p. 46).

Dapat disimpulkan misi yaitu suatu pernyataan oleh organisasi atau perusahaan mengenai produk/jasa yang akan diberikan kepada masyarakat yang berkaitan dengan pencapaian tujuan perusahaan.

Berikut ini adalah Misi yang dimiliki oleh bimbingan belajar X-Target, yaitu "Menerapakan S2LM, baik S2LM Hafalan maupun S2LM Hitungan, hingga terbangun generasi hebat yang memiliki kekuatan ilmu dan amal."

2.1.3. Logo

Logo merupakan suatu unsur yang sangat berpengaruh dalam sebuah perusahaan. Sebuah logo bisa berupa nama, lambang atau elemen grafis lain yang ditampilkan. Secara visualisasi logo adalah suatu gambar.

Logo adalah desain yang mudah dikenali, grafis khusus, nama yang berbentuk, simbol unik, atau perangkat lain untuk mengidentifikasi sebuah organisasi. Logo biasanya ditempelkan, dimasukan, atau dicetak pada semua iklan, bangunan, komunikasi, dokumen literatur, produk, alat tulis, dan kendaraan. (BusinessDirectory 2016)

Menurut Perum Percetakan Uang Republik Indonesia (PERURI), Logo adalah identitas perusahaan sebagai tanda / alamat yang akan mengkomunikasikan arti dan memberikan makna yang dalam terhadap jati diri perusahaan. Logo juga sebagai jiwa yang menghidupkan dan memberikan semangat bagi perusahaan di dalam melaksanakan misi dan mencapai tujuan perusahaan. (PERURI 2014)

Selanjutnya, Sularko dalam buku "*How Do They Think*", mengemukakan bahwa logo atau *corporate identity* atau *brand identity* adalah sebuah tanda yang

secara langsung tidak menjual, tetapi memberi suatu identitas yang pada akhirnya sebagai alat pemasaran yang signifikan, bahwa logo mampu membantu membedakan suatu produk atau jasa dari kompetitornya. Suatu logo diperoleh maknanya dari suatu kualitas yang disimbolkan, melalui pendekatan budaya perusahaan (*corporate culture*), penempatan posisi (*positioning*) historis atau aspirasi perusahaan, apa yang diartikan atau dimaksudkan.

Sehingga dapat ditarik kesimpulan bahwa logo adalah suatu gambar atau sekadar sketsa dengan arti tertentu, dan mewakili suatu arti dari perusahaan, daerah, organisasi, produk, negara, lembaga, dan hal lainnya yang membutuhkan sesuatu yang singkat dan mudah diingat sebagai pengganti dari nama sebenarnya.

Berikut ini merupakan Logo ataupun lambang dari bimbingan belajar X-Target :



Gambar 2.1. Logo X-Target

Berdasarkan gambar 2.1 adapun arti dari logo perusahaan adalah sebagai berikut :

1. Bentuk Siswa : Menandakan bimbingan belajar yang mengedepankan kepada siswa.
2. Bentuk Buku : Menandakan tujuan utama kepada pendidikan.
3. X-Target : filosofi X-Target menandakan pengajaran yang berorientasi sasaran. Simbol atau huruf X menganalogikan suatu koordinat tujuan. Sedangkan target sendiri bermakna sasaran yang akan diperoleh.

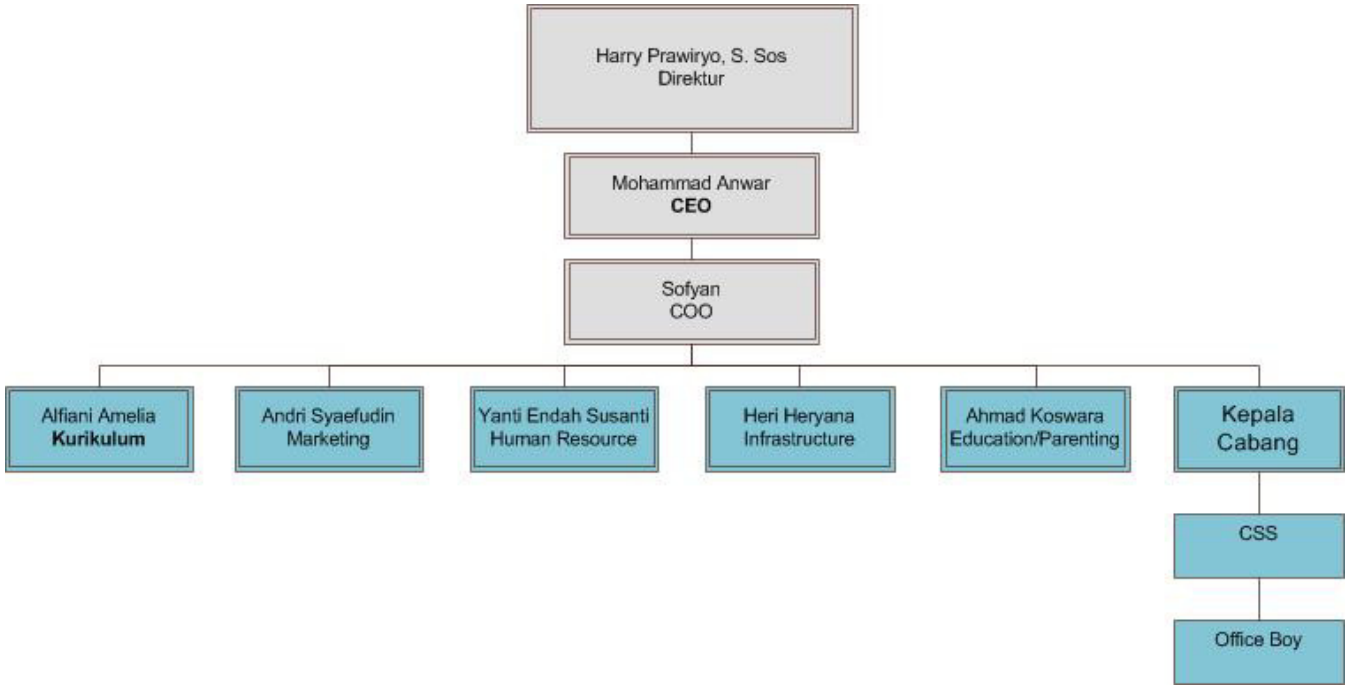
2.1.4. Struktur Organisasi

Bentuk dari struktur organisasi biasanya dibentuk dalam gambar atau bagan secara teratur yang memiliki hirarki, baik itu bersifat horizontal(kesetaraan) maupun secara vertikal (tanggung jawab).

Struktur organisasi adalah bagaimana pekerjaan dibagi, dikelompokkan, dan dikoordinasikan secara formal. (Robbins & Judge 2008, pp. 214-224). Struktur organisasi adalah salah satu sarana yang digunakan manajemen untuk mencapai sasarannya. Karena sasaran diturunkan dari strategi organisasi secara keseluruhan, sehingga strategi dan struktur harus terkait erat. Tepatnya, struktur harus mengikuti strategi. Jika manajemen melakukan perubahan signifikan dalam strategi organisasinya, struktur pun perlu dimodifikasi untuk menampung dan mendukung perubahan ini. (Pennings 1992, pp. 267-309)

Struktur organisasi di Bimbingan belajar X-Target dibagi kedalam 5 tingkatan, secara hirarki adalah Direktur, CEO (*Chief Executive Officer*), COO(*Chief Operating Officer*), dan Sub Divisi. Sedangkan pada Sub Divisi dibagi kedalam 4 Sub Divisi pada level yang sama, yaitu *Human Resource, Infrastructure, Marketing , Education/Parenting*, Kurikulum, dan Kepala Cabang. Dibawah kepala cabang terdapat CSS (*Custumer Service Specialist*) dan OB (*Office Boy*).

Berdasarkan gambaran diatas, berikut adalah bagan dari struktur organisasi bimbingan belajar X-Target :



Gambar 2.2. Struktur Organisasi X-Target

2.1.5. Deskripsi Tugas

Untuk kelancaran dan optimalisasi pelaksanaan tugas dan kinerja didalam perusahaan, dilakukan pembagian tugas dan tanggung jawab terhadap pekerjaan yang di ampunya. Deskripsi pembagian tugas berdasarkan pembagian tugasnya di dalam perusahaan adalah sebagai berikut :

2.1.5.1. *Direktur*

Jabatan direktur yang terdapat pada perusahaan X-Target dipimpin oleh Harry Prawiryo, S. Sos. Adapun deskripsi pekerjaannya adalah :

1. Memimpin perusahaan dengan menerbitkan kebijakan-kebijakan perusahaan
2. Memilih, menetapkan, mengawasi tugas dari karyawan dan kepala bagian (manajer)
3. Menyetujui anggaran tahunan perusahaan
4. Menyampaikan laporan kepada pemegang saham atas kinerja perusahaan.

2.1.5.2. *CEO (Chief Executive Officer)*

Jabatan CEO yang terdapat pada perusahaan X-Target dipimpin oleh Mohammad Anwar. Adapun deskripsi pekerjaannya adalah :

1. Menyusun kebijakan perusahaan, memimpin, mengkoordinir dan membuat program umum untuk meningkatkan produktifitas perusahaan, pendapatan perusahaan sesuai dengan target yang telah ditetapkan, juga mengontrol kerja team.
2. Merencanakan, mengkoordinasi, dan mengendalikan arus pekerjaan sesuai dengan jadwal dan prosedur pelaksanaan pekerjaan dengan mendasarkan pada budaya kerja profesional, tanggung jawab, sadar mutu, sadar waktu, sadar biaya, dan inisiatif.

2.1.5.3. *COO (Chief Operating Officer)*

Jabatan CEO yang terdapat pada perusahaan X-Target dipimpin oleh Sofyan. Adapun deskripsi pekerjaannya adalah :

1. Bertanggung jawab terhadap pelaksanaan harian untuk meningkatkan produktifitas perusahaan, juga mengontrol kerja team.

2. Mengawasi, mengkoordinasi, dan membuat pelaporan hasil arus pekerjaan sesuai dengan jadwal dan prosedur pelaksanaan pekerjaan

2.1.5.4. *Sub Divisi*

Sub divisi merupakan bagian organisasi yang berada tepat dibawah CEO. Sub divisi memiliki rekanan divisi yang sama derajatnya secara *horizontal.* Berikut deskripsi tugas dari masing-masing sub divisi :

2.1.5.4.1. *Human Resource*

Jabatan HR yang terdapat pada perusahaan X-Target dipimpin oleh Yanti Endah Susanti. Adapun deskripsi pekerjaannya adalah :

1. Bertanggung jawab di dalam pengelolaan dan pengembangan Sumber Daya Manusia, yaitu dalam hal perencanaan, pelaksanaan dan pengawasan kegiatan sumber daya manusia, termasuk pengembangan kualitasnya dengan berpedoman pada kebijaksanaan dan prosedur yang berlaku di perusahaan.
2. Bertanggung jawab terhadap hal-hal yang berkaitan dengan kegiatan-kegiatan pembinaan kualitas perusahaan serta mempunyai kewajiban memelihara dan menjaga citra perusahaan.

2.1.5.4.2. *Infrastructure*

Jabatan *infrastructure* yang terdapat pada perusahaan X-Target dipimpin oleh Heri Heryana. Adapun deskripsi pekerjaannya adalah :

1. Bertanggung jawab dalam pengelolaan rencana kebutuhan sarana dan prasarana, yaitu dalam hal menyusun, mengadministrasikan, mengelola biaya dan menyusun laporan.
2. Bertanggung jawab terhadap hal-hal yang berkaitan dengan sarana dan prasarana dalam kegiatan-kegiatan pembinaan kualitas perusahaan.
3. Inventarisasi barang

2.1.5.4.3. *Marketing*

Jabatan *marketing* yang terdapat pada perusahaan X-Target dipimpin oleh Andri Syaefudin. Adapun deskripsi pekerjaannya adalah :

1. Meningkatkan serta mempertahankan pelanggan perusahaan.
2. Bertanggung jawab dalam pengelolaan rencana dan strategi promosi, baik dalam hal menyusun, mengelola dan menyusun laporan sesuai dengan target pencapaian perusahaan.

2.1.5.4.4. *Education/Parenting*

Jabatan *Educating* yang terdapat pada perusahaan X-Target dipimpin oleh Achmad Koswara. Adapun deskripsi pekerjaannya adalah :

1. Bertanggung jawab dalam pengelolaan rencana dan strategi penguatan mental dan psikologi peserta didik
2. Bertanggung jawab dalam pengelolaan pelatihan tenaga pendidik yaitu pembekalan materi, pengarahan, dan penanganan peserta didik.

2.1.5.4.5. *Kurikulum*

Jabatan kurikulim yang terdapat pada perusahaan X-Target dipimpin oleh Alfiani Amelia. Adapun deskripsi pekerjaannya adalah :

1. Bertanggung jawab dalam pengelolaan rencana jadwal pengajaran, paket pembelajaran dan pengelolaan tenaga pengajar
2. Bertanggung jawab dalam materi pengajaran, yaitu menyusun kurikulum, mengevaluasi dan melakukan evaluasi dan pelaporan terhadap materi pengajaran.

2.1.5.5. *Kepala Cabang*

Kepala Cabang adalah orang yang bertanggung jawab sepenuhnya terhadap kantor cabang yang ditempati. Adapun deskripsi pekerjaannya adalah :

1. Bertanggung jawab dalam pengelolaan kegiatan pembelajaran baik organisir kelas dan ruangan
2. Bertanggung jawab dalam semua kegiatan administrasi yang terjadi di cabang tersebut, mulai dari *cash-in* dan *cash-out*, pengajuan dari kantor cabang, hingga kewajiban administrasi kepada perusahaan pihak ke-tiga (listrik, internet dll).

2.1.5.6. *CSS (Custumer Service Specialist)*

CSS(Customer Service Specialist) Operator adalah operator atau karyawan pada bimbingan belajar X-Target yang bertugas menyediakan layanan atau informasi produk atau informasi-informasi maupun solusi dan rincian yang relevan mengenai bimbingan belajar X-Target kepada pelanggan.

2.2. Landasan Teori

Untuk mendukung pembangunan sistem, maka perlu dikemukakan hal-hal atau teori-teori yang berkaitan dengan permasalahan dan ruang lingkup pembahasan sebagai landasan dalam pembuatan laporan ini.

2.2.1. SDLC (*Software Development Life Cycle*)

SDLC (*Software Development Life Cycle*) merupakan sebuah siklus hidup pengembangan perangkat lunak yang terdiri dari beberapa tahapan-tahapan penting dalam membangun perangkat lunak yang dilihat dari segi pengembangannya. Dengan siklus SDLC, proses membangun sistem dibagi menjadi beberapa langkah dan pada sistem yang besar, masing-masing langkah dikerjakan oleh tim yang berbeda. SDLC tidak hanya penting untuk proses produksi *software*, tetapi juga sangat penting untuk proses *maintenance software* itu sendiri.

Terdapat 4 metodologi penting dalam pengembangan *software* berbasis SDLC yaitu *Waterfall, Prototype, RAD (Rapid Application Development),* dan AGILE. (Rinjani 2013).

2.2.2. *Waterfall*

"*Classic Life Cycle*" atau model *Waterfall* merupakan model yang paling banyak dipakai didalam *Software Engineering* (SE). Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap analisis, desain, *coding, testing / verification, dan maintenance*. Disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Roger S. Pressman memecah model ini menjadi 6 tahapan, yaitu :

1. *System / Information Engineering and Modeling.*

Permodelan ini diawali dengan mencari kebutuhan dari keseluruhan sistem yang akan diaplikasikan ke dalam bentuk *software*. Hal ini sangat penting, mengingat *software* harus dapat berinteraksi dengan *elemen-elemen* yang lain seperti *hardware*, *database*, dsb. Tahap ini sering disebut dengan *Project Definition*.

2. *Software Requirements Analysis.*

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka para *software engineer* harus mengerti tentang domain informasi dari *software*, misalnya fungsi yang dibutuhkan, *user interface*, dsb. Dari 2 aktivitas tersebut (pencarian kebutuhan sistem dan *software*) harus didokumentasikan dan ditunjukkan kepada pelanggan.

3. *Design*

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk "*blueprint*" *software* sebelum *coding* dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya. Seperti 2 aktivitas sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari *software*.

4. *Coding*

Desain yang telah dibuat kemudian diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap design yang secara teknis nantinya dikerjakan oleh programmer.

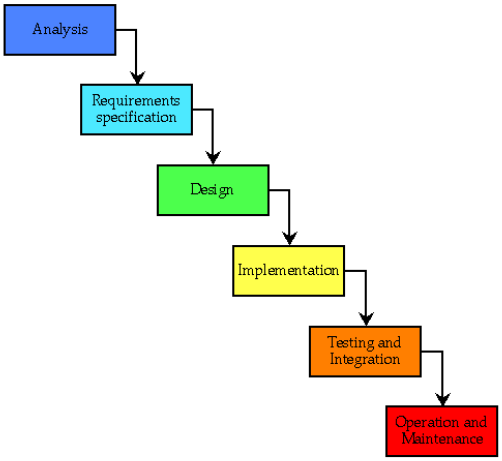
5. *Testing / Verification*

Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari *error*, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

6. *Maintenance*

Pemeliharaan suatu *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada *errors* kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengembangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergantian sistem operasi, atau perangkat lainnya.

Berikut adalah bagan dari *Waterfall* Model :



Gambar 2.3 Konsep SDLC – *Waterfall*

2.2.3. Penerapan SDLC

Penerapan SDLC dalam Pengembangan Sistem Presensi pada Bimbingan Belajar X-Target adalah menggunakan pendekatan *waterfall*.

2.2.4. UML (*Unified Modeling language*)

UML merupakan singkatan dari *Unified Modeling Language*. Sesuai dengan kata terakhir dari kepanjangannya, UML adalah salah satu bentuk language atau bahasa. UML di definisikan sebagai bahasa *visual* untuk menjelaskan, memberikan spesifikasi, merancang, membuat *model*, dan mendokumentasikan aspek-aspek dari sebuah *system*.

Karena tergolong bahasa *visual*, UML lebih mengedepankan penggunaan diagram untuk menggambarkan aspek dari system yang sedang dimodelkan. Pemahaman UML sebagai bahasa *visual* membedakan penekanannya kepada bahasa pemrograman yang lebih dekat ke mesin. Bahasa *visual* lebih dekat ke mental model pikiran kita, sehingga pemodelan menggunakan bahasa *visual* bisa lebih mudah dan lebih cepat dipahami dibandingkan apabila dituliskan dalam sebuah bahasa pemrograman.

UML adalah salah satu bentuk notasi atau bahasa yang digunakan oleh professional dibidang *software* untuk menggambarkan atau memodelkan sebuah system *software*. Sebelumnya ada banyak notasi atau bahasa lain untuk mencapai keperluan yang sama misalnya DFD (*Data Flow Diagram*) dan *Booch Diagram*. Tetapi sejak matang dan populernya teknologi pemrograman, perancangan, dan analisis berorientasi object, UML telah menjadi *de facto standard language*. UML mencoba untuk mendeskripsikan pemodelan sebuah sIstem dari segala aspek: pemodelan struktur (aspek statis), pemodelan perilaku (aspek dinamis), dan pemodelan arsitektur.

2.2.4.1. *Usecase Diagram*

Usecase diagram digunakan untuk memodelkan dan menyatakan unit fungsi/layanan yang disediakan oleh sistem (or bagian sistem: *subsistem* atau *class*) ke pemakai. *Usecase* dapat dilingkupi dengan batasan sistem yang diberi label nama sistem. *Usecase* adalah sesuatu yang menyediakan hasil yang dapat diukur ke pemakai atau sistem eksternal.

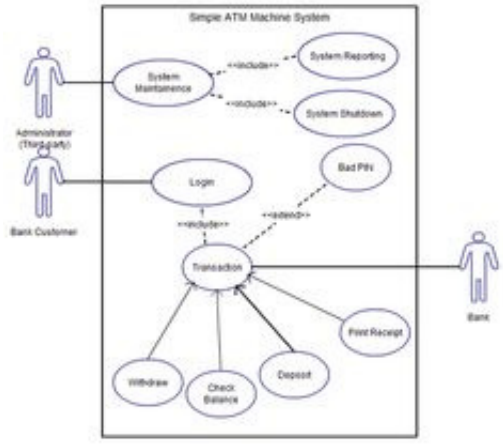
Usecase dibentuk untuk merepresentasikan bagaimana suatu sistem bekerja atau berjalan. Suatu sistem tidak lepas dari bagian-bagian tertentu untuk membentuk sistem tersebut dibentuk. Dalam hal ini komponen-komponen atau bagian yang membentuk *Usecase* adalah sebagai berikut :

1. *System*

Menyatakan batasan sistem dalam relasi dengan actor-aktor yang menggunakannya (diluar sistem) dan fitur-fitur yang harus disediakan (dalam sistem). Digambarkan dengan segi empat yang membatasi semua usecase dalam

sistem terhadap pihak mana sistem akan berinteraksi. Sistem disertai label yang menyebutkan nama dari sistem, tapi umumnya tidak digambarkan karena tidak terlalu memberi arti tambahan pada diagram.

Dalam memodelkan suatu sistem, bentuk yang digunakan atau menyatakan bahwa bagian tersebut adalah sistem, dengan memberikan ruang lingkup sistem atau *system boundary*.bentuk tersebut direpresentasikan dengan kotak atau persegi panjang.



Gambar 2.4. ***System Boundaries***

2. Usecase

Usecase adalaha sekelompok rangkaian/uraian yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh aktor. *Usecase* digunakan untuk membentuk tingkah laku benda dalam sebuah mode serta direalisasikan oleh sebuah *collaborator*, umumnya *usecase* digambarkan dengan sebuah elips dengan garis yang solid, biasanya mengandung nama. *Usecase* menggambarkan proses sistem(kebutuhan sistem dari sudut pandang user). Secara umum *usecase* adalah :

a. Pola Perilaku Sistem
b. Urutan transaksi yang berhubungan yang dilakukan oleh aktor.

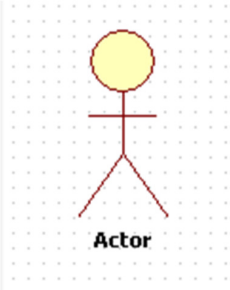
Usecase menggambarkan bagaimana seseorang akan menggunakan/memanfaatkan sistem. Pada diagram UML, *usecase* diwakili oleh simbol *Oval* atau lingkaran.



Gambar 2.5. Bentuk *Usecase*

3. Aktor

Pada dasarnya aktor bukanlah bagian dari *usecase* diagram, namun untuk dapat terciptanya suatu *usecase* diagram diperlukan beberapa aktor. *Aktor* tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah aktor mungkin hanya memberikan informasi inputan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. *Aktor* hanya berinteraksi dengan usecase, tetapi tidak memiliki kontrol atas usecase. *Aktor* digambarkan dengan stick man . *Aktor* dapat digambarkan secara secara umum atau spesifik, dimana untuk membedakannya kita dapat menggunakan relationship.



Gambar 2.6. Simbol Aktor

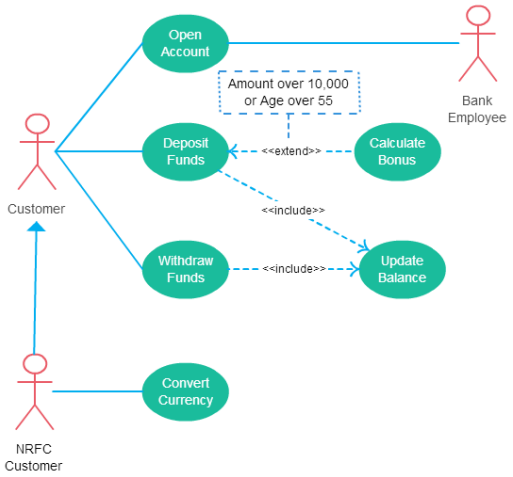
4. Relasi dalam Usecase

Pada umumnya, suatu *Usecase* terdiri dari minimal 2 atau lebih proses/*Usecase*. Keterhubungan anatara *Usecase* yang satu dengan yang lainnya dapat direpresentasikan kedalam diagram dengan sebuah garis. Setiap garis menyatakan bagaimana hubungan antar dua *Usecase* yang saling terhubung. Penggunaan tipe garis pada *Usecase* adalah sebagai berikut.

<<*include*>> , yaitu kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *usecase* adalah bagian dari *usecase* lainnya.

<<*extends*>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.

<<*communicates*>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah *communicates association*. Ini merupakan pilihan selama asosiasi hanya tipe *relationship* yang dibolehkan antara *actor* dan *usecase*.



Gambar 2.7. Simbol relasi

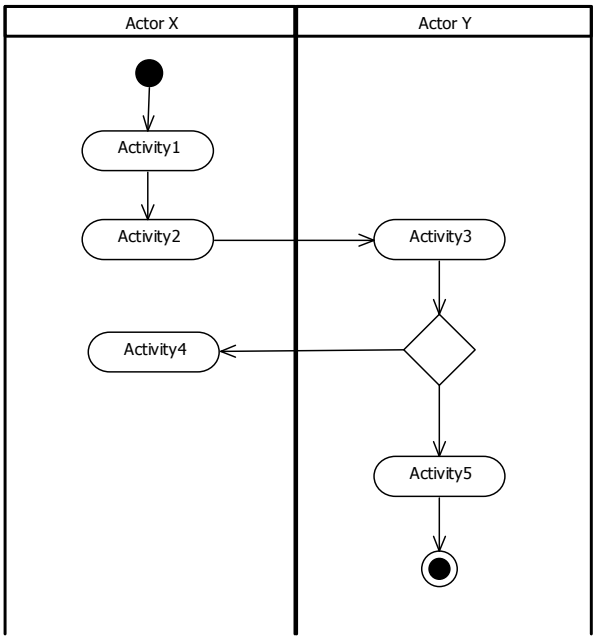
2.2.4.2. *Usecase Skenario*

Usecase skenario adalah penjelasan detail dari *Use-Case* yang kita buat sebelumnya. Setiap usecase harus dijelaskan alur prosesnya melalui sebuah deskripsi *usecase* (*usecase description)* atau *skenario usecase*. Usecase skenario menjelaskan mengenai tujuan *usecase* dan nilai yang akan didapatkan oleh aktor.

2.2.4.3. *Activity Diagram*

Activity Diagram adalah diagram yang menggambarkan *worlflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

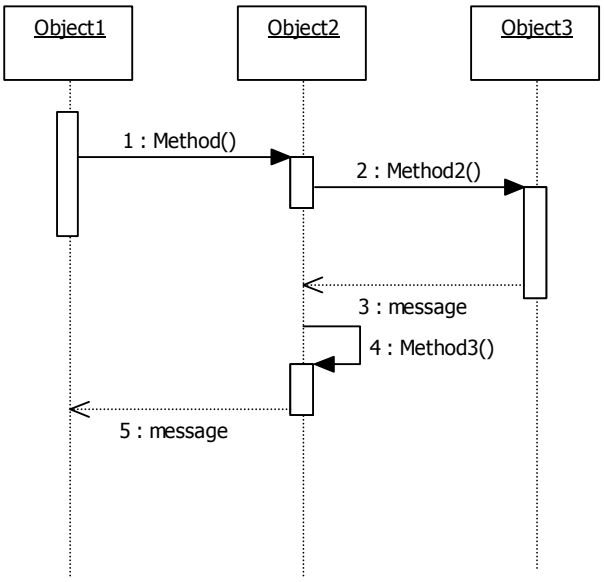
Perbedaan mendasar dari *use case diagram* dengan *activity diagram* adalah bahwa sebuah *activity* dapat direalisasikan oleh satu *use case* atau lebih. *Activity diagram* menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Untuk lebih jelasnya, berikut ini merupakan contoh bentuk *activity diagram* yang dapat dilihat pada Gambar 2.8.



Gambar 2.8. Contoh Bentuk *Activity Diagram*

2.2.4.4. *Sequence Diagram*

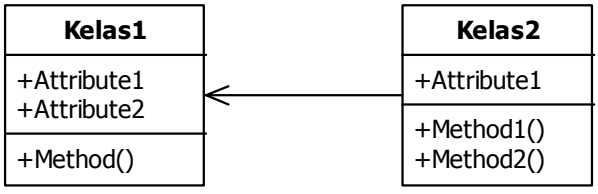
Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu (Rosa & Shalahudin 2011 , p. 165). Banyaknya diagram *sequence* yang harus dibuat adalah sebanyak pendefinisian *use case.* Untuk lebih jelasnya, berikut ini merupakan contoh bentuk *sequence diagram* yang dapat dilihat pada gambar berikut .



Gambar 2.9. Contoh Bentuk *Sequence Diagram*

2.2.4.5. *Class Diagram*

Class diagram adalah diagram yang menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. kelas memiliki 3 bagian utama yaitu *attribute*, *operation*, dan *name*. kelas-kelas yang ada pada struktur sistem harus dapat melakukan fungsi-fungsi sesuai dengan kebutuhan sistem. Untuk lebih jelasnya, berikut ini merupakan contoh bentuk *class diagram* yang dapat dilihat pada Gambar 2.10.



Gambar 2.10. Contoh Bentuk *Class Diagram*

2.2.5. Sistem

Kata “sistem” banyak sekali digunakan dalam percakapan sehari-hari, dalam forum diskusi maupun dokumen ilmiah. Kata ini digunakan untuk banyak hal, dan pada banyak bidang pula, sehingga maknanya menjadi beragam.

2.2.5.1. Pengertian Sistem

Menurut Kamus Besar Bahasa Indonesia (KBBI) Badan Pengembangan dan Pembinaan Bahasa, KEMDIKBUD (Pusat Bahasa) bahwa sistem adalah perangkat unsur yang secara teratur saling berkaitan sehingga membentuk suatu totalitas.

Berikut pengertian dan definisi sistem menurut beberapa ahli:

Jogiyanto mengatakan sistem adalah kumpulan dari *elemen-elemen* yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata, seperti tempat, benda dan orang-orang yang betul-betul ada dan terjadi (Jogiyanto 2005).

Agus Mulyanto mendefinisikan sistem secara umum sebagai kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu sebagai satu kesatuan. (Mulyanto 2009, p. 1)

Menurut Mustakini, “Sistem dapat didefinisikan dengan pendekatan prosedur dan pendekatan komponen, sistem dapat didefinisikan sebagai kumpulan dari prosedur-prosedur yang mempunyai tujuan tertentu”. (Muktasini & Jogiyanto 2009, p. 34)

Menurut Moekijat dalam , “Sistem adalah setiap sesuatu terdiri dari obyek-obyek, atau unsur-unsur, atau komponen-komponen yang bertata kaitan dan bertata hubungan satu sama lain, sedemikian rupa sehingga unsur-unsur tersebut merupakan satu kesatuan pemrosesan atau pengolahan yang tertentu”. (Moekijati 2011, p. 152)

Menurut Tata Sutabri, secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu. (Sutabri 2012, p. 10)

Dari pendapat yang dikemukakan di atas dapat disimpulkan bahwa sistem adalah suatu kumpulan atau kelompok dari elemen atau komponen yang saling berhubungan atau saling berinteraksi dan saling bergantung satu sama lain untuk mencapai tujuan tertentu.

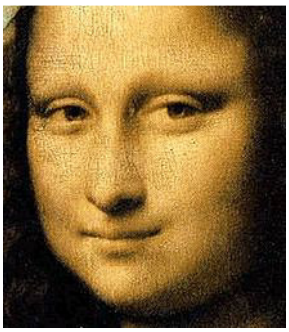
2.2.5.2. Sistem Komputer

Sistem Komputer adalah Suatu jaringan elektronik yang terdiri dari perangkat lunak dan perangkat keras yang melakukan tugas tertentu (menerima *input*, memproses *input*, menyimpan perintah-perintah, dan menyediakan *output*) dalam bentuk informasi.

2.2.6. Wajah

Identitas menurut Stella Ting Toomey merupakan refleksi diri atau cerminan diri yang berasal dari keluarga, gender, budaya, etnis dan proses sosialisasi. Identitas pada dasarnya merujuk pada refleksi dari diri kita sendiri dan persepsi orang lain terhadap diri kita. Sementara itu, Gardiner W. Harry dan Kosmitzki Corinne melihat identitas sebagai pendefinisian diri seseorang sebagai individu yang berbeda dalam perilaku, keyakinan dan sikap.

Wajah atau muka adalah bagian depan dari kepala, pada manusia meliputi wilayah dari dahi hingga dagu, termasuk rambut, dahi, alis, mata, hidung, pipi, mulut, bibir, gigi, kulit, dan dagu. Wajah terutama digunakan untuk ekspresi wajah, penampilan, serta identitas. Tidak ada satu wajahpun yang serupa mutlak, bahkan pada manusia kembar identik sekalipun. Untuk yang kembar identik, bentuk wajah secara kasat mata tidak dapat dibedakan.



Gambar 2.11. Wajah Monalisa

2.2.7. Presensi

Presensi dalam Kamus Besar Bahasa Indonesia berarti kehadiran. Presensi juga dapat diartikan sebagai daftar hadir. Daftar hadir menyatakan bahwa

seseorang telah menghadiri suatu kegiatan baik itu sebagai individu maupun sebagai perwakilan.

Presensi diterapkan pada kegiatan yang berulang dam memiliki jadwal untuk diikuti, sebagai contoh presensi kehadiran siswa, presensi kehadiran karyawan, presensi kehadiran rapat dan lain-lain. Dalam kehidupan sehari-hari pada banyak kegiatan yang dihadiri oleh banyak orang, sering dibuat daftar kehadiran. Dalam daftar tersebut yang telah disediakan, diminta kepada setiap orang yang hadir untuk mencantumkan tandatangannya sesuai dengan namanya yang sudah tertera.

Presensi merupakan kata serapan dari bahasa inggris yaitu *presence* yang berarti daftar kehadiran. Hal ini berbeda dengan absensi yang diserap dari bahasa inggris *abcent* yang berarti daftar tidak hadir/ ketidakhadiraan. Dalam kehidupan sehari-hari, presensi dan absensi sering tertukar makna dan penggunaan. Banyak di antara masyarakat sekarang yang mengartikan absensi sebagai daftar kehadiran. Absensi dan presensi adalah dua hal yang kontradiktif atau dapat disebut sebagai bertolak belakang.

Dengan memanfaatkan teknoligi, dapat membantu dalam pengerjaan atau tugas manusia. Salah satu penerapan yang dilakukan terdapat pada proses presensi. Beberapa teknologi yang digunakan dalam presensi adalah sebagai berikut :

1. Kertas

Penggunaan kertas dalam presensi adalah dengan membuat list nama peserta atau anggota dari suatu kelompok. Disetiap penggunaan kertas, biasanya ditampilkan dalam bentuk tabel dengan beberapa kolom yang menandakan identitas para anggota. Beberapa kertas presensi menyertakan tanda tangan dalam kolom diharapkan agar kehadiran masing-masing anggota dapat menunjukan orisinalitas atau keasliannya.

Format tampilan yang dicetak dalam kertas berbeda – beda dari satu sistem presensi dengan presensi yang lain. Penggunaan format tersebut disesuaikan berdasarkan kebutuhan masing – masing presensi. Sebagai contoh, presensi untuk

siswa menyertakan NIM, atau menerapkan data lain sebagai pembeda semisal alamat dan lain-lain.

Tabel 2.1 Contoh Format Presensi Siswa

No	NIM	Nama	Ttd.	

Tabel 2.2 Contoh Format Presensi Lain

No	Nama	Alamat	Ttd.	

2. Kartu Manual (Mesin ketok)

Kartu manual atau di indonesi sering disebut sebagai mesin ketok adalah bentuk lain dari kertas, namun jenis kertas yang digunakan lebih tebal atau lebih menyerupai kartu. Kartu manual ini sama halnya dalam presensi kertas, namun melibatkan perangkat lain untuk mencatat kehadiran masing masing anggota.

Kartu manual ini terdiri dari slot atau lobang untuk memasukkan kartu manual serta jam analog yang menunjukkan waktu saat ini. Pada saat anggota memasukkan kartunya ke dalam slot, waktu yang tertera pana mesin inilah yang akan dicetak pada kartu manual secara *real time*.



Gambar 2.12 Contoh Mesin Kartu Manual

3. Sidik jari

Presensi dengan *fingerprint*/sidik jari adalah salah satu teknologi presensi yang menggunakan metode pendeteksian melalui biometrik sidik jari dari

anggotanya. Sidik jari merupakan salah satu bagian pada tubuh manusia yang unik dan berbeda satu sama lain. Bahkan, kembar identik sekalipun memiliki jenis dan bentuk sidik jari yang berbeda. Hal inilah yang menjadi pemicu timbulnya gagasan mengintegrasikan mesin presensi fingerprint. Penggunaan presensi ini ditujukan agar tidak ada kecurangan, dikarenakan perbedaan antara sidik jari dari masing-masing anggota.



Gambar 2.13 *Visualiasi* Sidik Jari

4. Presensi Website

Presensi dengan bantuan website yaitu meminimalisir kekurangan penggunaan kertas dalam hal presesni. Presensi *website* merupakan pengembangan sistem presensi dimana setiap kehadiran setiap anggota dicatat langsung pada suatu portal atau *website* yang telah disediakan.

Dalam hal ini tentunya dibutuhkan moderator untuk mendata presensi dari setiap anggota. Moderator dapat melakukan presensi dengan memanggil nama satu persatu anggota kemudian mencatat kehadirannya pada halaman *website* yang telah disediakan. Pada *website* ini, dapat ditampilkan dengan *form* berupa *checklist* yang menunjukan kehadiran anggota tersebut.

5. Pengenalan Pola Wajah

Pengenalan pola wajah merupakan terobosan terbaru dalam memanfaatkan teknologi dalam hal presensi. Dengan adanya teknologi pengenalan pola wajah ini, mesin presensi wajah dapat mendeteksi dan mencatat kehadiran anggota tanpa bersentuhan dengan mesin presensi.

Pada mulanya setiap wajah dari masing masing anggota disimpan dalam database. Kemudian ketika anggota yang akan menggunakan atau melakukan presensi cukup mendekatkan wajah dirinya ke mesin presensi, mesin presensi

akan melakukan scanning terhadap wajah anggota tersebut, kemudian akan dilakukan pendeteksian wajah yang bersesuaian dengan wajah yang di baca.

2.2.8. Citra

Secara harafiah citra (*image*) adalah gambar pada bidang dwimatra (dua dimensi). Ditinjau dari sudut pandang matematis citra merupakan fungsi menerus (*continue*) dari intensitas cahaya pada bidang dwimatra. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. pantulan cahaya ini ditangkap oleh alat-alat optik misalnya mata pada manusia kamera pemindai (*scanner*) dan sebagainya sehingga bayangan objek yang disebut citra tersebut terekam (Munir 2004, p. 2).

Citra adalah representasi dari suatu obyek nyata baik dalam bentuk dua dimensi ataupun tiga dimensi menjadi bentuk gambar digital yang dimengerti oleh komputer. Definisi citra lainnya adalah sebuah fungsi intensitas warna dua dimensi f(x,y) dimana x dan y mewakili koordinat lokasi suatu titik dan nilai dari fungsi yang merupakan tingkat intensitas warna atau tingkat keabu-abuan dari titik tersebut.

Pengolahan citra "*image processing*" merupakan bidang yang berhubungan dengan proses transformasi citra(image) yang bertujuan untuk mendapatkan kualitas citra yang lebih baik (C.Fairhurst 1988, p. 5). Pengenalan pola "*pattern recognition*" merupakan bidang yang berhubungan dengan proses identifikasi obyek pada citra atau interpretasi citra. proses ini bertujuan untuk mengekstrak informasi atau pesan yang disampaikan oleh gambar atau citra.

2.2.9. Piksel

Piksel adalah unsur gambar atau representasi sebuah titik terkecil dalam sebuah gambar grafis yang dihitung per inci. Pada ujung tertinggi skala resolusi, mesin cetak gambar berwarna dapat menghasilkan hasil cetak yang memiliki lebih dari 2.500 titik per inci denga pilihan 16 juta warna lebih untuk setiap inci, dalam istilah komputer berarti gambar seluas satu inci persegi yang bisa ditampilkan pada tingkat resolusi tersebut sepadan dengan 150 juta bit informasi.

Monitor atau layar datar yang sering kita temui terdiri dari ribuan piksel yang terbagi dalam baris-baris dan kolom-kolom. Jumlah piksel yang terdapat dalam sebuah monitor dapat kita ketahui dari resolusinya. Resolusi maksimum yang disediakan oleh monitor adalah 1024x768, maka jumlah pixel yang ada dalam layar monitor tersebut adalah 786432 piksel. Semakin tinggi jumlah piksel yang tersedia dalam monitor, semakin tajam gambar yang mampu ditampilkan oleh monitor tersebut.

2.2.10. *Grayscalling*

Grayscalling citra merupakan tahapan pertama dari proses penyelarasan, pada tahap ini terjadi pengkonversian citra warna RGB menjadi citra berwarna abu. Citra warna RGB terdiri dari 3 parameter warna yaitu merah (red), hijau (green) dan biru (blue), jika citra warna RGB ini dimasukan ke dalam proses ekstraksi, maka proses tersebut akan sulit untuk dilakukan karena citra RGB terdiri dari 3 parameter, oleh karena itu diperlukan penyamaan parameter yaitu dengan melakukan tahap grayscaling ini. persamaan grayscaling citra :

$$X = (Red * 0.3) + (Green * 0.59) + (Blue * 0.11) \quad \ldots\ldots\ldots (2.1)$$

Setiap piksel dalam gambar dilakukan proses satu-persatu terhadap persamaan agar diperoleh nilai *grayscale*. Setiap piksel memiliki nilai parameter warna dimana nilai citra RGB adalah (red,ggreen,blue) dengan nilai integer diantara 0 sampai 255, dan x adalah nilai *grayscale*.



Gambar 2.14. *Graycale* Citra

2.2.11. Haar-Like Feature dan Haar Cascade

Haar like Feature merupakan metode yang lazim digunakan dalam pendeteksian obyek. Nama Haar sendiri mengacu pada *Haar Wavelet*, sebuah fungsi matematika yang berbentuk kotak dan memiliki prinsip seperti pada fungsi Fourier. Haar-like features merupakan rectangular *features* (fungsi persegi), yang memberikan indikasi secara spesifik pada sebuah gambar atau image. Prinsip Haar-like features adalah mengenali obyek berdasarkan nilai sederhana dari fitur tetapi bukan merupakan nilai piksel dari image obyek tersebut. Metode ini memiliki kelebihan yaitu komputasinya sangat cepat, karena hanya bergantung pada jumlah piksel dalam persegi bukan setiap nilai piksel dari sebuah *image* (Viola & Jones 2001) Deteksi obyek kendaraan pada penelitian ini merupakan pemodifikasian sistem Haar-like features dari deteksi wajah yang pertama kali dilakukan oleh Viola dan Jones kemudian dikembangkan oleh Lienhart (Viola & Jones 2001). Metode yang diusulkan Viola dan Jones menggabungkan empat kunci utama untuk mendeteksi sebuah obyek :

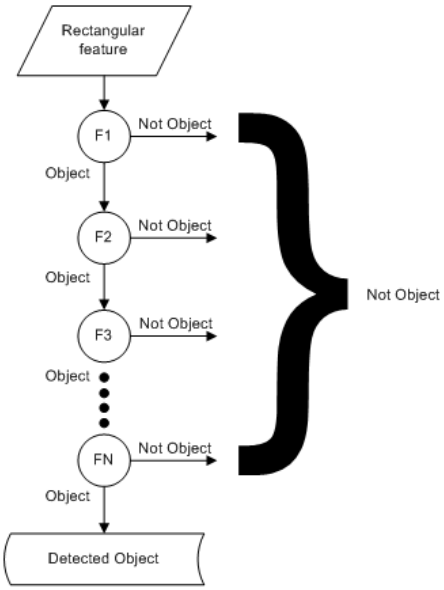
1. Fitur persegi sederhana, disebut fitur Haar
2. Integral image untuk pendeteksian fitur dengan cepat
3. Metoda AdaBoost *machine-learning*
4. Cascade classifier untuk mengkombinasikan banyak fitur

Haar like *feature* memproses citra dalam sebuah kotak persegi dengan ukuran tertentu misalnya 24 x 24 pixel seperti ditunjukkan pada gambar 1 sebagai berikut:



Gambar 2.15 *Rectangular* Feature Haar Cascade

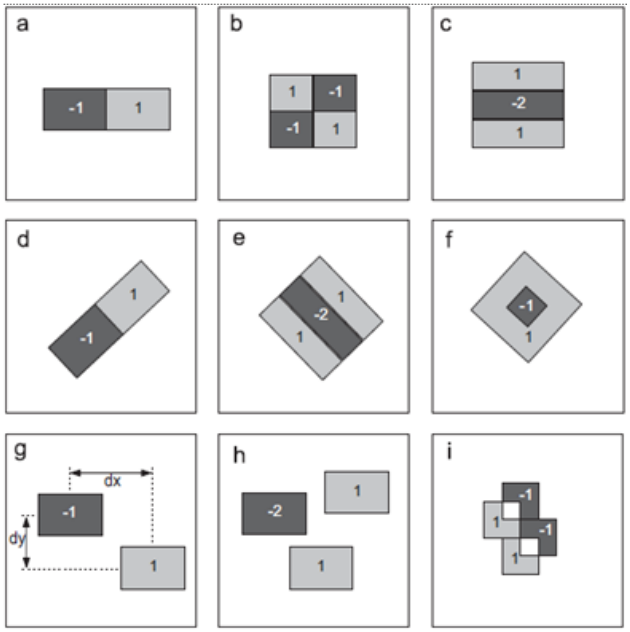
Di dalam kotak inilah proses filtering obyek dilakukan untuk diketahui apakah ada atau tidak obyek yang akan dideteksi. Proses filterisasi ini dilakukan secara bertingkat yang menyebabkan metode ini nantinya disebut sebagai Haar *Cascade Classifier* seperti ditunjukkan pada skema filter di gambar 2 sebagai berikut:



Gambar 2.16 Skema Pendeteksi Obyek

Hasil deteksi *Haar-like Feature* kurang akurat jika hanya menggunakan satu fungsi saja. Semakin tinggi tingkatan filter pendeteksian maka semakin tepat pula sebuah obyek dideteksi akan tetapi akan semakin lama proses pendeteksiannya. Pemrosesan Haar-like feature yang banyak tersebut diatur dalam *classifier cascade*.

Haar-Wavelet (*Wave* = Gelombang) merupakan gelombang persegi (interval gelap dan interval terang) yang kemudian dibandingkan nilai rata-rata *pixel* keduanya. Apabila perbandingan nilai rata-rata intensitas tersebut berada di atas *threshold* (ambang batas), maka dikatakan memenuhi syarat fitur Haar. Untuk gambar bergerak seperti video, proses ini dilakukan secara diskrit dengan mencuplik video pada *frame rate* tertentu. Macam-macam variasi Haar-like *feature* ditunjukkan pada gambar 3 sebagai berikut (Pavani, Frangi & Delgado 2010)



Gambar 2.17 Berbagai variasi persegi Haar

2.2.12. *Nearest-Neighboard Interpolation*

Proses *resize* digunakan untuk membuat setiap gambar yang diterima memiliki ukuran dimensi atau ordo matrik yang sama. Proses *resize* menggunakan metode interpolasi dalam melakukan pengolahan citra. Perhitungan mendasar dalam melakukan perubahan/*resize* sebuah citra, haruslah diketahui terlebih dahulu *rasio* yang digunakan pada gambar awal terhadap citra tujuan. Berikut proses menghitung nilai *rasio* yang digunakan dalam proses konversi.

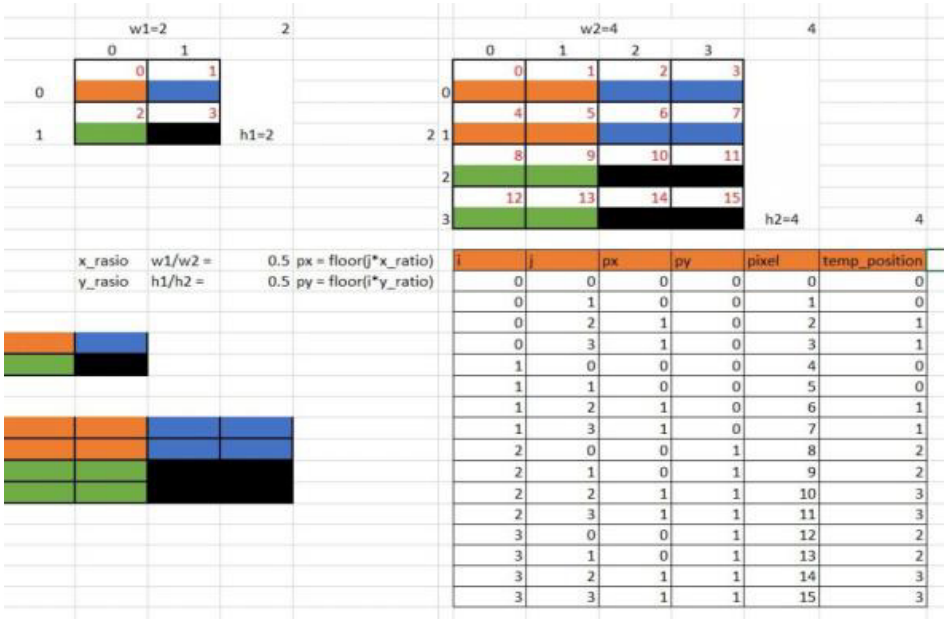
(2.2)

$$rasio_x = \frac{lebar_{awal}}{lebar_{tujuan}}$$

$$rasio_y = \frac{tinggi_{awal}}{tinggi_{tujuan}}$$

$$temp = (rasio_y * lebar_awal) + rasio_x$$

Untuk memudahkan perhitungan, pembuatan nilai pixel dapat disajikan dalam bentuk tabel. Adapun tabel yang dimaksud adalah sebagai berikut :



i	j	px	py	pixel	temp_position
0	0	0	0	0	0
0	1	0	0	1	0
0	2	1	0	2	1
0	3	1	0	3	1
1	0	0	0	4	0
1	1	0	0	5	0
1	2	1	0	6	1
1	3	1	0	7	1
2	0	0	1	8	2
2	1	0	1	9	2
2	2	1	1	10	3
2	3	1	1	11	3
3	0	0	1	12	2
3	1	0	1	13	2
3	2	1	1	14	3
3	3	1	1	15	3

Gambar 2.18 Tabel Perhitungan *Rasio*

2.2.13. PCA (*Principal Component Analysis*)

Wajah merupakan bagian dari tubuh manusia yang paling mudah untuk dikenali. Seiring dengan perkembangan teknologi muncul fenomena kebutuhan komputasi untuk pengenalan pola wajah. Pengenalan pola wajah tidak hanya bermanfaat untuk ilmu pengetahuan tetapi juga untuk beberapa aplikasi praktis seperti *autentifikasi* identitas, *surveillance* dan *security*, *human-computer intelligent interaction* serta film. Salah satu teknik pengenalan pola wajah adalah PCA (*Principal component analysis*) dan Eigenface adalah salah satu algoritma yang menggunakan konsep PCA.

Principal component analysis adalah teknik linear reduksi menggunakan teori sederhana dari statistik. Misalkan diketahui suatu citra dengan ukuran h x w, dengan h adalah tinggi citra dan w adalah lebar citra, maka dimensi citra adalah n dengan n = h x w. Jika suatu citra dengan ukuran n, maka jumlah kombinasi linier dari citra sebanyak n merupakan dimensi yang tinggi, dan ini merupakan masalah besar pada komputasi ketika proses pengukuran. Dengan menggunakan PCA sebagai ekstraksi fitur, maka dimensi yang tinggi tersebut dapat dikurangi menjadi dimensi yang rendah. Jumlah dimensi yang dihasilkan oleh PCA tergantung pada

jumlah data yang digunakan oleh pelatihan dan jumlah sampel pada masing-masing data pelatihan. Misalkan jumlah data pelatihan adalah k dan masing-masing data mempunyai s model, maka jumlah sampel keseluruhan adalah m dengan m = k x s. Metode PCA efisien digunakan pada suatu kondisi m << n (jumlah data jauh lebih kecil dibandingkan dengan dimensi citra yang digunakan penelitian). (Purnomo, Mauridhi & Muntasa 1991)

Principal Component Analysis (PCA) dibuat pertama kali oleh para ahli statistik dan ditemukan oleh Karl Pearson pada tahun l90l yang memakainya pada bidang biologi. Kemudian tidak ada perkembangan baru pada teknik ini, dan. perkembangannya baru mulai pesat pada akhir tahun l930 dan awal 1940. Setelah itu perkembangannya berkurang sebentar sampai komputer telah berhasil didesain sehingga dapat mengaplikasikan teknik ini pada masalah-masalah yang masuk akal. Pada tahun 1947 teori ini muncul lagi dan cukup independen sebagai teori probabilitas yang ditemukan oleh Karhunen, dan kemudian dikembangkan oleh Loeve pada tahun l963, sehingga teori ini juga dinamakan Karhunen-Loeve transform pada bidang ilmu telekomunikasi. (Informatika 2013).

2.2.14. EigenFace

Kata eigenface sebenarnya berasal dari bahasa Jerman “eigenwert” dimana “eigen” artinya karakteristik dan “wert” artinya nilai. Eigenface adalah salah satu algoritma pengenalan pola wajah yang berdasarkan pada *Principle Component Analysis* (PCA) yang dikembangkan di MIT.

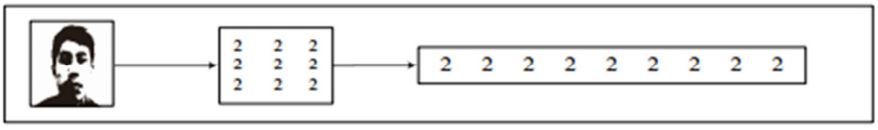
Teori ini dikembangkan dengan membagi sebuah citra wajah menjadi data set fitur karakteristik yang disebut eigenface. Fitur karakteristik ini merupakan komponen utama (*principal component*) dari training set awal dari citra wajah. Penelitian yang dilakukan oleh Carey dan Diamond menunjukkan bahwa fitur wajah sebuah individu dan hubungan langsung antar fitur tersebut tidak dapat menyamai kemampuan manusia dalam memperhatikan dan mengenal wajah.

Eigenface adalah sekumpulan *standardize face ingredient* yang diambil dari analisis statistik dari banyak gambar wajah (Hanif 2009). Eigenface merupakan kumpulan dari eigenvector yang digunakan untuk masalah computer vision pada

pengenalan wajah manusia. Untuk menghasilkan eigenface, sekumpulan citra digital dari wajah manusia diambil pada kondisi pencahayaan yang sama kemudian dinormalisasikan dan diproses pada resolusi yang sama (misal m x n), kemudian citra tadi diperlakukan sebagai vektor dimensi m x n dimana komponennya diambil dari nilai piksel citra. Alur kerja dari eigenface dilakukan beberapa tahap, diantaranya :

2.2.14.1.*Merubah vektor 2D ke 1D*

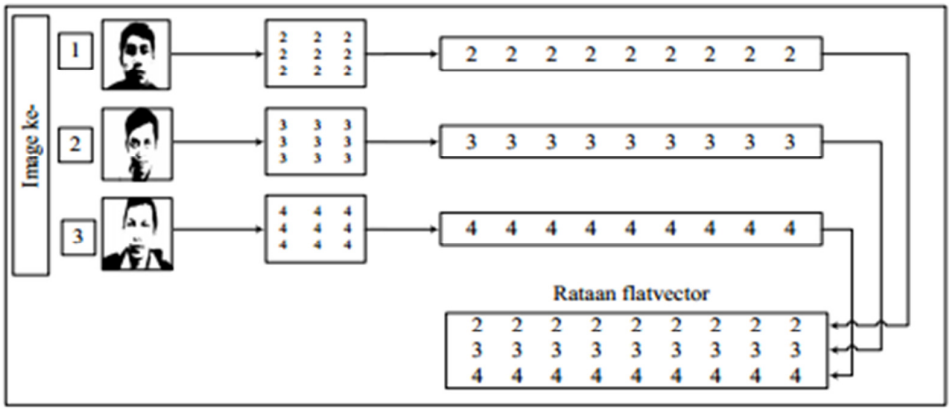
Saat melakukan pengenalan pola wajah, sebuah citra akan melakukan *preprocessing* sebelumnya hingga citra *ideal.* Salah satu kriteria agar gambar tersebut ideal adalah dengan mengubah nilai matrik piksel ke dalam bentuk vektor 1 dimensi.



Gambar 2.19. Contoh Perubahan Vektor 2D ke 1D

2.2.14.2.*Penyusunan flatvector*

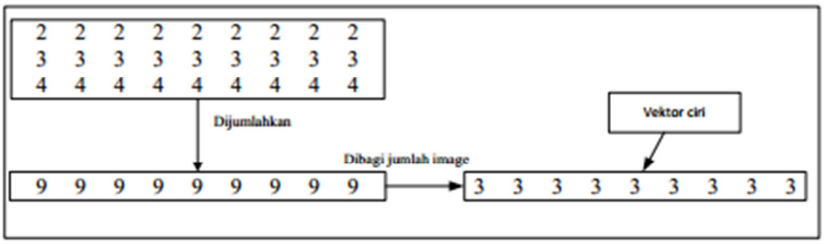
Penyusunan *flatvektor* adalah membuat data training dari data wajah yang disimpan di database. Untuk setiap wajah yang disimpan, dilakukan proses *preprocessing* hingga proses ekstraksi. Kemudian setiap data wajah dibuat dalam bentuk 1D. Bentuk 1D dari setiap wajah kemudian dicari nilai rataan dari *flatvektor* tersebut. Penjelasan lebih lanjut dapat dilihat pada gambar dibawah ini :



Gambar 2.20 Penyusunan *Flatvektor*

2.2.14.3.*Perhitungan Rataan flatvector*

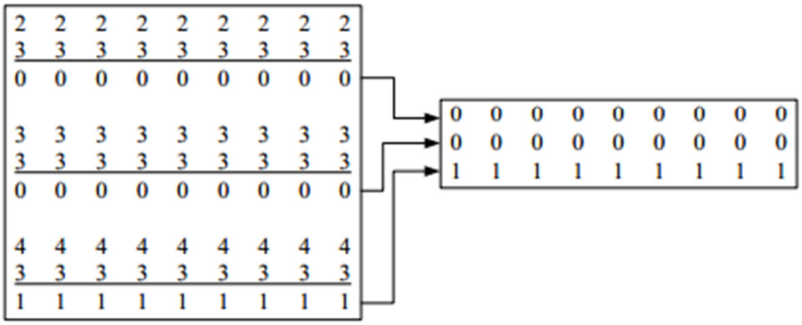
Setelah didapat *flatvektor,* langkah selanjutnya adalah menghitung rataan *flatvektor*. Rataan dihitung dengan menjumlahkan urutan nilai piksel yang bersesuaian, kemudian dibagi dengan jumlah gambar. Berikut ilustrasi pembuatan rataan flatvektor.



Gambar 2.21 Contoh Perhitungan Rataan *flatvektor*

2.2.14.4.*Proyeksikan ke facespace*

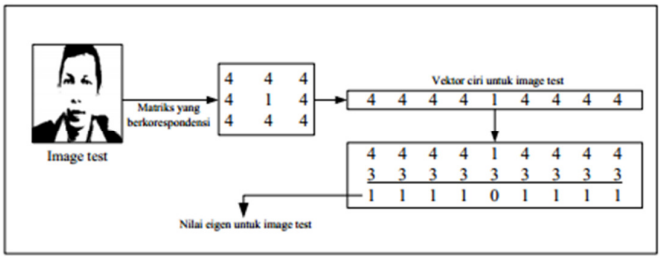
Proses selanjutnya adalah memproyeksikan masing-masing vektor kedalam *facespace*. *Facespace* berisi nilai piksel masing-masing gambar dikurangi dengan nilai rataan *flatvektor*. Namun, apabila ditemukan nilai pengurangan yang menghasilkan nilai negatif, nilai pengurangan menjadi 0.



Gambar 2.22 Proyeksi ke facespace

2.2.14.5.*Ektraksi PCA*

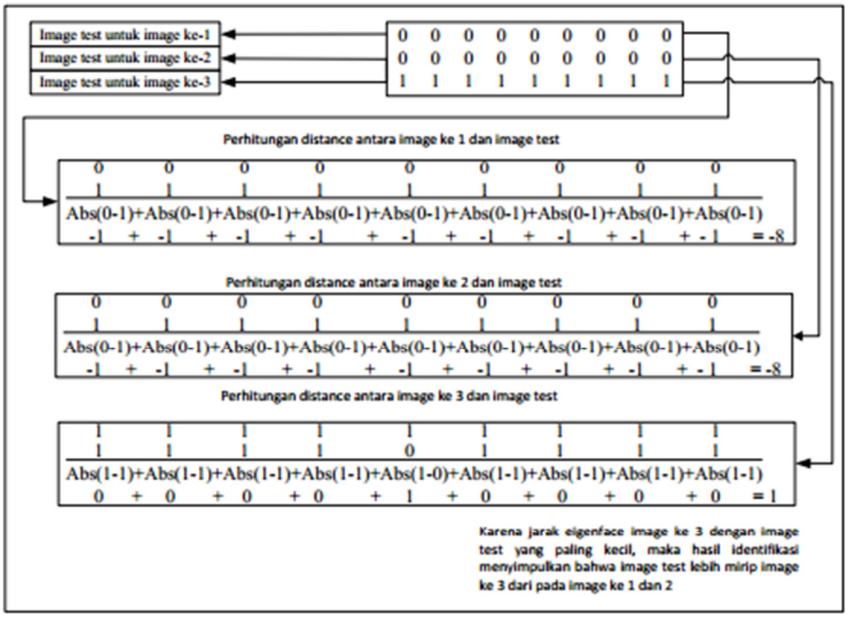
Untuk mengidentifikasi data wajah, hasil dari konversi vektor 1D pada tabel dilakukan pengurangan terhadap nilai rataan *flatvektor*. Untuk lebih jelasnya, berikut proses ekstraksi PCA



Gambar 2.23 Pembentukan *flatvektor*

2.2.14.6.*Cari Distance Minimum*

Mencari nilai minimum distance dengan cara membandingkan penjumlahan hasil eigenface dengan nilai eigen dari masing masing gambar yang terdapat pada data training. Hasil penjumlahan merupakan nilai absolute dari selisih nilai eigen itu sendiri.



Gambar 2.24 Mencari *Distance Minimum*

2.2.15. OpenCV Dan EmguCV

Open CV adalah *library* yang didalamnya terdapat berbagai fungsi yang dipergunakan dalam *computer vision*. Open CV ditulis dengan bahasa C dan C++ yang dapat berjalan di Linux, Windows dan MacOS. Open CV telah lama dipergunakan para peneliti dan pengembang dalam berbagai bidang *computer vision*, seperti inspeksi produk, *medical imaging*, *camera callibration*, *security*, *user interface*, *stereo vision* dan robotika. (Denny 2012)

EmguCV adalah wrapper .Net untuk OpenCV. Emgu CV berperan untuk menjembatani C# dan OpenCV. Dengan EmguCV, fungsi-fungsi dalam OpenCV dapat dipanggil melalui bahasa pemrograman yang *compatible* dengan .NET seperti C#, VB, dan VC++. EmguCV adalah *library* yang ditulis dengan C# yang mana tentunya lebih *safe* karena pembuatan *object* atau pun *reference* di-*manage* oleh *garbage collector*. (Utama 2016)

Emgu CV terdiri dari 2 layer, yaitu *basic layer* dan *second layer*. *Basic layer* mengandung fungsi, struktur, dan enumerasi yang secara langsung merefleksikan apa yang ada di OpenCV. Sehingga fungsi-fungsi pada OpenCV dapat dengan mudah dipanggil dengan bahasa pemrograman C#. Sedangkan *second layer*

mengandung kelas-kelas yang memanfaatkan keunggulan teknologi .NET. (Utama 2016)

2.2.16. *Lux Meter*

Alat ukur cahaya (*lux meter*) adalah alat yang digunakan untuk mengukur besarnya intensitas cahaya di suatu tempat. Besarnya intensitas cahaya ini perlu untuk diketahui karena pada dasarnya manusia juga memerlukan penerangan yang cukup. Untuk mengetahui besarnya intensitas cahaya ini maka diperlukan sebuah sensor yang cukup peka dan linier terhadap cahaya. Semakin jauh jarak antara sumber cahaya ke sensor maka akan semakin kecil nilai yang ditunjukkan lux meter. Ini membuktikan bahwa semakin jauh jaraknya maka intensitas cahaya akan semakin berkurang Alat ini didalam memperlihatkan hasil pengukurannya menggunakan format digital yang terdiri dari rangka, sebuah sensor. Sensor tersebut diletakan pada sumber cahaya yang akan diukur intenstasnya. (Tridinamika 2016)

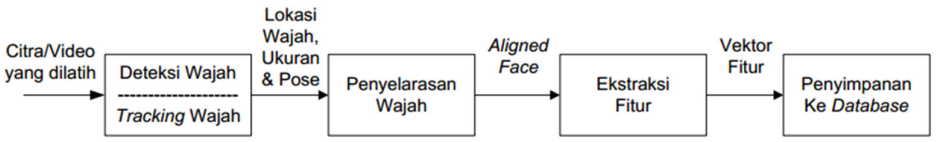
2.2.17. Proses Pengenalan wajah

Secara umum, meskipun didasari dengan metode-metode yang berbeda proses pengenalan wajah terdiri dari dua proses utama yaitu, proses pelatihan citra dan proses pengenalan citra. Kedua proses ini memiliki tahapan-tahapan yang hampir sama, perbedaannya terletak pada hasil ekstraksi fiturnya.

Proses pelatihan citra wajah memiliki tahapan-tahapan sebagai berikut:

1. Pendeteksian wajah atau *tracking* wajah.
2. Penyelarasan wajah.
3. Ekstraksi fitur.
4. Penyimpanan fitur ke database.

Untuk lebih jelasnya, diagram alur pelatihan wajah adalah sebagai berikut :

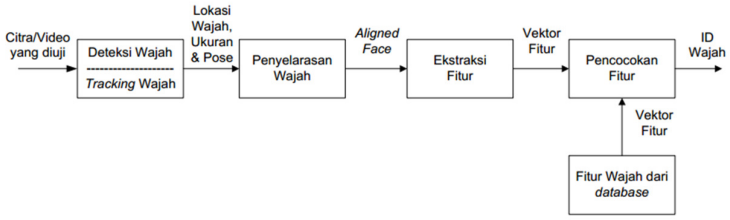


Gambar 2.25 Proses Pelatihan Citra Wajah

Sedangkan untuk proses pengenalan citra, tahapan tahapan yang dilalui adalah :

1. Pendeteksian wajah atau tracking wajah.
2. Penyelarasan wajah.
3. Ekstraksi fitur.
4. Pencocokan fitur yang diuji dengan fitur yang dilatih

Untuk lebih jelasnya, diagram alur pengenalan wajah adalah sebagai berikut



Gambar 2.26 Proses Pengenalan Citra Wajah

2.2.17.1. Pendeteksian Wajah

Pendeteksian wajah adalah proses yang mensegmentasi area wajah dengan latar belakang dari suatu masukan citra. Proses ini bekerja dengan cara memeriksa citra yang dimasukan, apakah memiliki citra wajah atau tidak, jika memiliki, maka akan dilakukan pemisahan dengan cara memotong citra wajah dari latar belakang citra yang dimasukan. Jika masukan berbentuk video, proses yang dilakukan adalah proses *face tracking*. Secara umum, proses *face tracking* dan proses pendeteksian wajah mempunyai fungsi yang sama. Perbedaannya terletak pada proses pendeteksiannya saja, jika pada masukan berbentuk citra, sistem berjalan *offline* sehingga dapat menggunakan proses pendeteksian wajah, sedangkan pada masukan video, sistem berjalan secara *online* atau *real-time* yang membutuhkan pendeteksian secara langsung maka proses yang digunakan adalah proses *face tracking*. *Face tracking* yang dilakukan menggunakan metode viola-jone atau *haar cascade*.

2.2.17.2.Penyelarasan Wajah

Pada proses pendeteksian wajah, citra wajah yang didapatkan masih berupa perkiraan kasar atau masih memiliki kualitas yang cukup buruk seperti ukuran

yang berbeda dengan ukuran normal, faktor pencahayaan yang kurang atau lebih, kejelasan citra yang buruk dan sebagainya. Maka perlu diadakannya proses penyelarasan.

Proses penyelarasan wajah merupakan proses yang bertujuan untuk menormalisasi citra wajah yang didapatkan dari proses pendeteksian wajah. Proses ini terdiri dari tahapan-tahapan sebagai berikut:

2.2.17.3.Ekstrasi Wajah

Proses ekstraksi wajah merupakan proses untuk mendapatkan citra wajah. Dalam proses ekstraksi terdapat dua buah proses, yaitu deteksi wajah menggunakan *haar cascade* dan *resize.* Algoritma *haar-cascade* digunakan untuk menandai atau memilih area yang terdeteksi memiliki wajah. kemudian wajah tersebut yang nantinya akan disimpan kedalam database.

2.2.17.4.Penyimpanan Fitur Wajah

Proses penyimpanan fitur, merupakan tahapan terakhir dari proses pelatihan citra wajah. Proses ini bertugas untuk menyimpan vektor fitur hasil ekstraksi citra wajah beserta identitas vektor fitur tersebut kedalam database, yang nantinya akan digunakan sebagai pembanding citra wajah yang diuji pada proses pengenalan wajah.

2.2.17.5. Pencocokan Fitur Wajah

Pencocokan fitur adalah tahapan proses yang hanya terdapat pada proses pengenalan citra wajah. Proses ini merupakan proses perbandingan fitur yang telah diekstrak dari citra uji dengan fitur citra wajah dari database, yang sebelumnya telah melalui proses pelatihan citra. Proses perbandingan ini dilakukan menggunakan metode pengenalan pola, salah satunya adalah metode nearest neighbours.

Dari proses perbandingan fitur tersebut akan menghasilkan nilai jarak terdekat yang menandakan nilai fitur citra uji hampir menyamai dengan fitur citra latih. Nilai jarak ini akan menjadi nilai masukan untuk nilai kemiripan citra.

Proses pencocokan dilakukan dengan membandingkan nilai eigenface suatu gambar dengan nilai nilai rataan iegenface pada data training. Perbandingan akan melihat sebara besar kesalahan atau tingkat error ketika membandingkan, semakin sedikit tingkat kesalahan, maka semakin bagus, data citra semakin mirip.

2.2.18. *Short Message Service* (SMS)

Short Message Service (SMS) adalah layanan komponen pesan teks dari telepon, web, atau sistem komunikasi mobile lainnya. SMS Menggunakan protokol komunikasi standar untuk memungkinkan telepon tetap atau perangkat ponsel untuk bertukar pesan teks singkat.

Sebuah pesan sms maksimal terdiri dari 140 bytes, dengan kata lain sebuah pesan bisa memuat 140 karakter 8- bit, 160 karakter 7- bit atau 70 karakter 16- bit untuk bahasa Jepang, bahasa Mandarin dan bahasa Korea yang memakai Hanzi (Aksara Kanji / Hanja). Selain 140 bytes ini ada data- data lain y ang termasuk. Adapula beberapa metode untuk mengirim pesan yang lebih dari 140 bytes, tetapi seorang pengguna harus membayar lebih dari sekali.

Pesan - pesan sms dikirim dari sebuah telepon genggam ke pusat pesan (SMSC dalam bahasa Inggris), di sini pesan di simpan dan mencoba mengirimnya selama beberapa kali. Setelah sebuah waktu yang telah ditentukan, biasanya 1 hari 19 atau 2 hari, lalu pesan dihapus. Seorang pengguna bisa mendapatkan konfirmasi dari pusat pesan ini.

2.2.19. *Short Message Service Gateway* (SMS *Gateway*)

Short Message Service (SMS) *gateway* adalah *mekanisme* interaksi atau pengolahan pesan SMS sehinggan dapat dikirim dan diterima. SMS *gateway* memfasilitasi dan merampingkan proses pesan teks untuk organisasi, dan sering melakukan beberapa konversi ke format yang berbeda.

SMS *Gateway* berfungsi sebagai penghubung yang melakukan *relay* sms antara ESME (*External Short Message Entitiy*) dan SMSC dan sebaliknya. Komunikasi antara ESME dan SMS *Gateway* dapat menggunakan protokol SMPP atau dengan HTTP, sementara ke SMSC menggunakan Protokol SMPP (*Short Message Peer to Peer*).

ESME adalah entitas luar yang dapat berupa *server* aplikasi penyedia layanan (*Application Service Provider*) yang dimiliki oleh *Content Provider*, aplikasi Perbankan, *server polling*, dan lain-lain yang dapat menerima pesan, memproses pesan dan mengirim *respons* atas pesan yang masuk , serta perangkat lain seperti *email gateway*, WAP *proxy server* , Voice mail *server*.

SMPP adalah protokol yang cukup populer yang terbuka, protokol standar industri yang dirancang untuk menyediakan antarmuka komunikasi data yang fleksibel untuk transfer data pesan singkat antar entitas SMS, seperti ESME dan SMSC. SMPP yang banyak digunakan sekarang adalah SMPP v3.4. SMPP cukup populer dipakai karena memberikan kapasitas yang besar dan kecepatan transfer sms yang cepat.

Beberapa keuntungan dalam menggunakan SMS Gateway adalah :

1. Pengetikan pesan lebih mudah menggunakan peranti masukan keyboard.
2. Kemudahan dan kecepatan manipulasi konten pesan (dapat menggunakan copy dan paste).
3. Kemudahan dalam memasukan data nomor telepon tujuan.
4. Kemudahan penggunaan hanya satu nomor perusahaan, sehingga mudah diingat dan mempermudah dalam sosialasi.

2.2.20. Pengujian

Pengujian dilakukan untuk mengetahui kinerja dari sistem yang telah dirancang, apakah sudah sesuai dengan yang diharapkan atau tidak seperti yang diharapkan. Pengujian ini akan dilakukan dengan pengujian software secara berulang dengan beberapa faktor pengujian, kemudian akan dilakukan analisa pengambilan data-data hasil pengujian tersebut

2.2.20.1. *Black box*

Pengujian pada *black box* tidak terlalu memperhatikan struktur internal (*source code*) dari program, tetapi lebih kepada menemukan kesalahan pada program.

2.2.20.2. Pengujian Beta

Pengujian beta adalah pengujian yang dilakukan secara objektif dengan kata lain pengujian ini adalah pengujian secara langsung di lapangan atau tempat dimana aplikasi yang dibuat diimplementasikan. Untuk pengujian beta, dilakukan dengan cara melakukan wawancara terhadap pihak terkait yang menggunakan sistem.

2.2.20.3. UAT *(User Acceptance Test)*

User Acceptance Test, merupakan metode pengujian dimana klien atau user ikut serta dalam melakukan pengujian produk untuk melakukan validasi produk sesungguhnya sesuai dengan kebutuhannya. Pengujian dilakukan di lokasi klien atau tempat pemasanagan produk. (TutorialsPoint 2016)

User Acceptance Test merupakan langkah terakhir sebelum meluncurkan aplikasi. Biasanya pengguna akhir yang akan menggunakan aplikasi tersebut yang akan melakuka pengujian sebelum pengguna tersebut menerima aplikasi untuk digunakan. Pengujian memberikan keyakinan pada pengguna akhir bahwa aplikasi yang disampaikan kepada mereka telah memenuhi kebutuhan mereka. Pengujian juga dapat membantu dalam meperbaikin bug yang berkaitan dengan aplikasi. Atribut-atribut yang melekat pada UAT, dapat dilihat pada tabel berikut :

Tabel 2.3 Atribut dalam *User Acceptance Test*

No	Kriteria	*Test Plan*	Pelaporan
1.	Fungsional Kebenaran dan Kelengkapan	Pengenalan	Identifikasi Pelaporan
2.	Integritas data	Kategori Penerimaan Pengujian	Hasil Pelaporan
3.	Konversi Data	Lingkungan Operasi	Variasi
4.	Kegunaan	ID Kasus Pengujian	Rekomendasi
5.	Prestasi	Judul Pengujian	Rangkuman daftar tugas selanjutnya
6.	Aktualitas	Tujuan Pengujian	Keputusan Disetujui
7.	Kerahasiaan dan Ketersediaan	Prosedur Pengujian	
8.	Installability dan upgradability	Jadwal pengujian	
9.	Skalabilitas	*Resources*	
10.	Dokumentasi		