

BAB I

PENDAHULUAN

A. Latar Belakang

Teknik kehidupan sehari-hari, lebih mudah mengenali seseorang lewat wajahnya dibandingkan lewat sidik jari atau iris mata. Istilah dalam proses pengenalan wajah dapat disebut sebagai *covert deployment*, yaitu suatu objek tidak merasa kalau sedang dikenali lewat wajahnya. Hal ini tidak mungkin dilakukan lewat biometrik lain seperti sidik jari, iris mata yang memerlukan kerjasama dari objek untuk menyodorkan jari atau menempatkan mata pada alat pemindai (A. S. Nugroho, 2017). Oleh karena itu, pengenalan wajah menjadi sangat penting untuk sistem pemantauan dan penanggulangan kriminal dan juga bisa diterapkan untuk sistem lain yang berbasis pengenalan wajah seperti sistem absensi dengan pengenalan wajah (R. Hartanto dan M. N. Adji, 2018).

Sistem pengenalan wajah pada skripsi ini akan diimplementasikan untuk sistem absensi. Sistem absensi menggunakan pengenalan wajah merupakan sistem absensi yang menerapkan teknik biometrik untuk pencatatan absensi yang kemudian masuk ke dalam *database system* (F. Syuhada, I. G. P. Suta Wijaya, dan F. Bimantoro, 2018). Disamping itu, peneliti memilih metode pembelajaran *deep learning* untuk *face recognition* ini karena berdasarkan pada masalah yang sering dihadapi pada sistem pengenalan wajah pada umumnya yaitu yang tingkat akurasi pengenalan wajahnya itu masih rendah.

Sistem absensi sangat banyak digunakan di sekolah-sekolah, universitas-universitas maupun perusahaan-perusahaan. Sistem absensi yang digunakan kebanyakan masih menggunakan metode konvensional yaitu melalui lembar atau *sheet* dalam bentuk *hardcopy* yang kemudian di tanda tangani dan dikumpulkan kembali setelah ditanda tangani. Proses pengolahan data absensi mahasiswa dilakukan dengan memasukkan data absensi mahasiswa tersebut secara manual kedalam sistem informasi akademik lalu dikelola melalui proses komputerisasi. Banyaknya masalah dan kekurangan dari sistem absensi melalui metode konvensional seperti membutuhkan waktu yang cukup lama dalam proses penandatanganan mahasiswa di dalam perpustakaan sehingga membuat antrian yang Panjang. Oleh karena itu, peneliti ingin melakukan penelitian dengan menerapkan teknologi *face recognition* untuk sistem absensi ini, yang diharapkan dapat kehadiran atau absensi mahasiswa dapat dilakukan dengan cepat dan lebih efisien.

Dalam pembuatan sistem absensi dengan pengenalan wajah ini, peneliti akan mengumpulkan foto wajah dengan sampel Data yang digunakan dalam penelitian ini adalah data gambar wajah sebagian mahasiswa di fakultas Teknik informatika Universitas Muhammadiyah Parepare. Angkatan 2017 dengan setiap orangnya memiliki gambar wajah dengan ekspresi wajah. Data yang telah dikumpulkan selanjutnya akan di *resize* karena ukuran data sebelumnya terlalu besar dan berat sistem juga akan sulit dalam melakukan pengolahan data training. Setelah itu, dataset tersebut dibagi menjadi data latih (*train*) dan data uji (*test*), proses training data pada penelitian ini menerapkan metode pembelajaran *deep learning* dengan salah satu modelnya yang paling sesuai untuk pembuatan sistem

face recognition yaitu model *Open CV*. Di dalam prakteknya, para mahasiswa dapat melakukan proses absensi bisa dengan posisi wajah dengan menghadapkan wajah di depan kamera *Webcam*. Setelah itu, setiap mahasiswa yang telah mendaftarkan kehadirannya, maka data-data mahasiswa tersebut seperti nama, NIM, jurusan serta tahun Angkatan akan masuk kedalam *database* yang ada di dalam sistem absensi dan sebagai pertanda kehadiran setiap mahasiswa secara *real time* yang nantinya akan menghasilkan keluaran nilai.

B. Rumusan Masalah

Berdasarkan uraian latar belakang yang telah diuraikan, maka akan dibuat suatu penelitian pada masalah yang sedang dihadapi oleh Perpustakaan UM Parepare. Adapun latar belakang masalah yang dapat diidentifikasi sebagai berikut :

1. Bagaimana menganalisa, merancang dan membangun sebuah sistem absensi menggunakan pengenalan wajah berbasis *Open CV*?
2. Bagaimana hasil analisis pengenalan wajah dengan metode *Open CV* pada *framework Tensorflow* untuk mengetahui suatu tingkat keakuratan yang dihasilkan?

C. Batasan Masalah

Adapun agar pembahasan menjadi lebih terarah dan tidak menyimpang dan juga sesuai dengan latar belakang yang sudah diuraikan sebelumnya, maka penulis membatasi permasalahan penelitian pada Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*. Maka penulis membatasi masalahnya sebagai berikut :

1. Merancang dan membangun sistem absensi perpustakaan menggunakan teknologi *Face Recognition* berbasis *Open CV*.
2. Uji coba dan implementasi suatu sistem hanya di lingkup Universitas Muhammadiyah Parepare Program Studi Teknik Informatika.
3. Sistem ini tidak membahas tentang Fakultas atau program Studi lainnya.
4. Memiliki sistem absensi yang cepat tanpa harus menunggu sehingga lebih mudah dan praktis untuk mahasiswa.
5. Aplikasi ini dibangun menggunakan Bahasa pemrograman *Tensorflow.js*, *JavaScript* dan *HTML* serta basis data *Mongodb*.
6. Aplikasi yang dibangun berupa sistem berbasis *Web*.

D. Tujuan Penelitian

Berdasarkan rumusan masalah yang telah didefinisikan sebelumnya tujuan dari penelitian dan perancangan aplikasi ujian komprehensif Program Studi Teknik Informatika UM Parepare dalam penelitian ini adalah :

1. Merancang dan membangun sebuah sistem absensi menggunakan pengenalan wajah berbasis *Open CV*.
2. Untuk mengetahui hasil analisis pengenalan wajah dengan metode *Open CV* pada *framework Tensorflow* untuk mengetahui suatu tingkat keakuratan yang dihasilkan.

E. Manfaat Penelitian

Adapun Salah satu fungsi dari penelitian adalah bisa memberikan manfaat dan kegunaan yang secara nyata akan diterima oleh berbagai pihak yang terkait

dengan obyek penelitian seperti dunia akademis secara luas. Dalam penelitian kali ini peneliti berharap bisa memberikan kegunaan, antara lain :

1. Bagi Peneliti

- a. Meningkatkan kemampuan berfikir tentang konsep seperti apa yang harus penulis angkat kedalam penelitian, tentunya berkaitan dekat dengan keadaan di Universitas yang menjadi tempat penelitian.
- b. Meningkatkan kemampuan untuk memecahkan masalah yang timbul dari konsep yang sebelumnya disusun.
- c. Meningkatkan wawasan mengenai langkah-langkah dan tata cara membuat sistem yang berguna yang dapat diukur melalui keefektifan dalam penggunaan dan keefisienan dalam pemanfaatan sistem.
- d. Memenuhi syarat kelulusan untuk jenjang Strata Satu (S1) pada Fakultas Teknik Informatika, Universitas Muhammadiyah Parepare.

2. Bagi Dosen dan Mahasiswa

- a. Memiliki sistem yang baru sebagai media absensi perpustakaan pada aktivitas di UM Parepare.
- b. Memiliki sistem yang dapat mempermudah dosen dan mahasiswa/i UM Parepare.
- c. Meminimalisir tenaga, waktu dan biaya dalam melaksanakan *presensi* mahasiswa/i.

3. Bagi Universitas / Perguruan Tinggi

- a. Menciptakan Mahasiswa dan Mahasiswi yang mampu menerapkan sebuah sistem absensi elektronik di Lembaga Pendidikan atau perguruan tinggi yang menjadi tempat penelitian.
- b. Dapat membuat perguruan tinggi dikenal lebih luas oleh institusi swasta atau negeri yang dijadikan objek penelitian.
- c. Menjadi referensi bagi Mahasiswa/i yang akan melakukan penelitian khususnya untuk judul yang berkaitan dengan aplikasi berbasis *Website*.

F. Sistematika Penulisan

Dalam Sistematika penulisan skripsi ini dibagi ke dalam lima bab yang masing-masing menjelaskan bagian-bagian yang berbeda namun tetap memiliki korelasi satu dengan yang lain. Setiap bab-nya terdiri dari beberapa sub tersendiri. Diawali dengan bab pendahuluan dan di akhiri dengan bab penutup yang berisi kesimpulan dan saran-saran. Secara singkat akan diuraikan sebagai berikut :

BAB I : PENDAHULUAN

Pada bab ini akan dijelaskan tentang latar belakang penulisan, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II : TINJAUAN PUSTAKA

Bab ini berisi uraian tentang landasan teori-teori dari perancangan “Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*”.

BAB III : METODOLOGI PENELITIAN

Dalam bab ini membahas mengenai metode penelitian yang digunakan dalam merancang dan membangun sistem. Pada bab ini juga membahas tentang identifikasi masalah, analisis kebutuhan aplikasi yang meliputi kebutuhan masukan, proses dan keluaran aplikasi yang akan dibangun. Penjelasan mengenai proses perancangan dengan menggunakan rancangan dan UML.

BAB IV : HASIL DAN PEMBAHASAN

Pada bab ini membahas mengenai suatu hasil dari tahap analisis dan perancangan sesuai dengan metode yang digunakan pada alat dan sistem yang dibuat serta hasil dari pengujian.

BAB V : PENUTUP

Pada bab akhir ini berisi tentang kesimpulan dari penelitian yang telah dilakukan dan saran-saran yang akan diberikan untuk pengembangan lebih lanjut agar tercapai hasil yang lebih maksimal.

DAFTAR PUSTAKA

LAMPIRAN

BAB II

TINJAUAN PUSTAKA

A. Penelitian Terdahulu

Berbagai penelitian sebelumnya merupakan hal yang sangat perlu dan dapat dijadikan sebagai data pendukung. Salah satu data pendukung yang menurut peneliti perlu dijadikan bagian tersendiri adalah penelitian terdahulu yang relevan dengan permasalahan yang sedang dibahas dalam penelitian ini. Dalam hal ini, fokus penelitian terdahulu yang dijadikan acuan adalah terkait dengan masalah teknologi informasi. Oleh karena itu, peneliti melakukan kajian terhadap beberapa hasil penelitian berupa tugas akhir/ skripsi atau jurnal melalui *internet*. Beberapa penelitian sebelumnya yang diambil oleh peneliti sebagai bahan pertimbangan dan sumber referensi yang berhubungan dengan judul penelitian ini diantaranya sebagai berikut.

1. Sumatri Multi Lahan (2021) dengan skripsi berjudul **“Aplikasi Peminjaman Buku Pada Perpustakaan Menggunakan Kartu Tanda Siswa (KTS)”**.

Dalam penelitian ini peneliti berfokus menggunakan suatu aplikasi desktop *delphi* dengan terintegrasi dengan *RFID* dan *Barcode* sehingga data yang tersimpanpun lebih terjamin kerapiahannya karena bersifat aplikasi yang datanya pun tersimpan dengan aman. Selanjutnya pada bagian dari alatnya digunakan *Mikrokontroller Arduino Uno* dan *RFID* serta *Barcode Reader* yang sebagai penghubung alat dan pembaca data. Data-data yang telah diinput akan disimpan ke dalam Database *SQL* sehingga data yang disimpan dapat aman dan mudah

untuk ditemukan serta teratur. Adapun pengujian sistemnya mendapatkan hasil yang sesuai kebutuhan yang telah ditargetkan.

2. Murni (2018) dengan skripsi yang berjudul “**Aplikasi Notifikasi Masa Berakhir Peminjaman Buku Pada Perpustakaan Umum Melalui SMS Gateway**”. Masalah pokok yang dibahas dalam penelitian ini adalah bagaimana merancang suatu aplikasi notifikasi masa berakhir peminjaman buku melalui *sms gateway*. Sistem ini dirancang menggunakan bahasa pemrograman VB dengan *MySQL* sebagai database. Hasil Penelitian menunjukkan bahwa sistem aplikasi notifikasi masa berakhir peminjaman buku pada perpustakaan melalui *sms gateway* yang telah dibangun ternyata dapat menanggulangi permasalahan yang ada pada sistem lama, sehingga sistem yang baru ini lebih efisien dan efektif dan hasilnya akan lebih optimal.
3. Muh. Amin (2021) dengan skripsi yang berjudul “**Sistem Penelusuran Lokasi Buku Pada Perpustakaan Universitas Muhammadiyah Parepare Berbasis Web**”. Berdasarkan hasil penelitian yang dilakukan oleh peneliti tersebut Munculah gagasan untuk membuat aplikasi penelusuran buku pada perpustakaan UM Parepare berbasis *web* dengan Metode yang digunakan dalam pembuatan aplikasi ini adalah *Waterfall*. Bahasa pemrograman yang digunakan untuk membuat aplikasi ujian web ini adalah *PHP* dengan menggunakan *framework Laravel*. Berdasarkan hasil implementasi sistem ini memungkinkan pengguna untuk mencari katalog buku secara online, mengecek ketersediaan, dan menemukan lokasi fisik buku dengan lebih mudah.

B. Tinjauan Teori

1. Definisi Sistem

a. Pengertian Sistem

Sistem dapat merupakan sesuatu yang abstrak maupun yang berwujud. Terdapat beberapa pendapat para ahli mengenai pengertian sistem. Davis (1984) Sebuah system terdiri dari bagian-bagian yang saling berkaitan yang beroperasi bersama untuk mencapai beberapa sasaran atau maksud. Sedangkan menurut Mcleod (2001) Sistem adalah himpunan dari unsur-unsur yang saling berkaitan sehingga membentuk suatu kesatuan yang utuh dan terpadu. Menurut Suartini Bambang, sistem adalah kesatuan yang terdiri dari elemen-elemen yang berkaitan untuk berinteraksi satu sama lain dalam rangka menciptakan hasil atau tujuan tertentu. Ada dua pendekatan dalam mendefinisikan sistem, yaitu :

- 1) Pendekatan yang menekankan pada prosedur Dalam pendekatan prosedur, Sistem sebagai suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau menyelesaikan suatu sasaran tertentu. Prosedur itu sendiri adalah urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa (*what*) yang harus dikerjakan, siapa (*who*) yang mengerjakannya, kapan (*when*) dikerjakan, dan bagaimana (*how*) mengerjakannya (Jogiyanto 2005).
- 2) Pendekatan yang Menekankan pada Elemen dan/atau Komponen Sistem : Pendekatan Elemen/ Komponen dalam pendekatan elemen, Sistem

didefinisikan sebagai sekumpulan elemen yang salingberinteraksi satu dengan yang lain untuk mencapai suatu tujuan tertentu (Cushing, Davis, Murdick, Fuller, Ross, etc). Pendekatan ini sifatnya lebih luas dan lebih banyak diterima oleh berbagaikalangan. Suatu sistem yang dibuat tentunya memiliki maksud tertentu. Sistem dibuat untuk mencapai suatu tujuan (*goal*) dan sasaran (*objective*). Tujuan (*goal*) meliputi ruang lingkup luas.

Sistem adalah kumpulan atau himpunan dari unsur atau variabel-variabel yang saling berkaitan, saling berinteraksi, dan saling tergantung satu sama lain untuk mencapai tujuan. Sistem menurut Romney dan Steinbart (2015). Sistem adalah rangkaian dari dua atau lebih komponen-komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Sebagian besar sistem terdiridari subsistem yang lebih kecil yang mendukung sistem yang lebih besar. Sistem tersebut terdiri atas input, proses dan output. Input adalah data masukan dan output adalah data yang dihasilkan berupa informasi.

b. Karakteristik Sistem

Suatu sistem mempunyai karateristik atau sifat-sifat tertentu, karakteristik sistem yang paling umum dimiliki sebuah sistem adalah sebagai berikut :

- 1) Komponen atau elemen (*Components*) : Suatu sistem yang terdiri dari komponen-komponen yang saling berinteraksi, yang artinya saling bekerjasama membentuk suatu kesatuan.
- 2) Batasan sistem (*Boundary*) : Batasan sistem merupakan daerah yang

membatasi antara sistem yang satu dengan sistem yang lain atau dengan lingkungan luarnya. Adanya batas sistem, maka sistem dapat membentuk suatu kesatuan karena dengan batas sistem ini, fungsi dan tugas sub sistem satu dengan yang lainnya berbeda tetapi saling berinteraksi.

- 3) Lingkungan Luar Sistem (*Environment*) : Lingkungan luar sistem adalah segala sesuatu diluar batas sistem yang mempengaruhi operasi suatu sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan merugikan. Lingkungan luar sistem yang bersifat menguntungkan harus dipelihara dan dijaga supaya tidak hilang pengaruhnya. Sedangkan, lingkungan yang bersifat merugikan harus dihilangkan supaya tidak mengganggu operasi sistem.
- 4) Penghubung sistem (*Interface*) : Penghubung sistem merupakan suatu media (penghubung) antara subsistem dengan subsistem lainnya yang membentuk suatu kesatuan, sehingga sumber-sumber daya mengalir dari subsistem yang satu ke subsistem lainnya. Dengan kata lain, melalui penghubung, output dari subsistem akan menjadi input bagi subsistem lainnya.
- 5) Masukan (*Input*) : *Input* adalah energi atau sesuatu dimasukkan ke dalam suatu sistem yang dapat berupa masukan yaitu energi yang dimasukkan supaya sistem dapat beroperasi atau masukan sinyal yang merupakan energi yang diproses untuk menghasilkan suatu luaran.
- 6) Luaran (*Output*) : *Output* merupakan hasil dari energi yang diolah dan

diklasifikasikan menjadi *output*.

- 7) Pengolah (Process) : Suatu sistem mempunyai bagian pengolah yang akan mengubah input menjadi output.
- 8) Sasaran (*Objektive*) : Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil apabila mengenai sasaran dan tujuannya.

c. Syarat-Syarat Sistem

Menurut (Kumar, 2011), syarat- syarat sistem seperti berikut :

- 1) Sistem harus dibentuk untuk menyelesaikan suatu tujuan.
- 2) Elemen sistem harus mempunyai rencana yang ditetapkan.
- 3) Adanya hubungan diantara elemen sistem.

2. Absensi

Absensi atau kartu jam hadir ialah dokumen yang mencatat jam hadir setiap karyawan di perusahaan. Catatan jam hadir karyawan tersebut dapat berupa daftar hadir biasa, dapat juga pula berbentuk kartu hadir yang diisi dengan mesin pencatat waktu. Pekerjaan mencatat waktu pada dasarnya dapat dipisahkan menjadi 2 (dua) bagian yakni pencatatan waktu hadir (*attendance time keeping*) dan juga pencatatan waktu kerja.

Pencatatan jam hadir pada kartu jam hadir yang dilakukan oleh pada setiap pegawai atau pekerja bisa mempengaruhi gaji bersih atau *take home pay* yang akan diterima oleh si pegawai atau pekerja setiap bulannya. Karena jika pegawai atau pekerja lupa ataupun tidak mencatatkan jam hadirnya pada kartu jam

hadir akan dapat mempengaruhi komponen-komponen yang ada pada gaji, terutama sekali pada pos tunjangan, Dikarena tunjangan yang diberikan perusahaan pada setiap pegawai atau pekerjatergantungan dari beberapa banyak pegawai atau pekerja hadir pada jam kerja. Seperti tunjangan makan dan juga transportasi.

Apabila pegawai atau pekerja tidak mencatatkan jam hadirnya pada kartu jam hadir maka tunjangan makan dan hingga transpotasinya yang diterima pegawai atau pekerja setiap bulannya akan berkurang dan juga akan mempengaruhi gaji bersih yang diterima pegawai atau pekerja tersebut. Pencatatan waktu hadir tersebut dimaksudkan untuk mengumpulkan data mengenai jumlah jam hadir pegawai atau pekerja dalam suatu periode pembayaran dan juga kadang-kadang juga tentang tarif upah untuk pekerjaan yang dilakukan. Pencatatan waktu kerja dimaksudkan untuk dapat mencatat jam kerja sesungguhnya yang digunakan oleh pegawai atau pekerja dalam setiap pekerjaan (job) atau departemennya. Catatan waktu kerja tersebut dapat digunakan untuk mengecek catatan waktu hadir serta juga mendapatkan data produksi yang diperlukan untuk dapat distribusi upah serta gaji dan juga perhitungan intensif.

3. Pengenalan Wajah (Face Recognition)

Identifikasi (pengenalan) wajah atau *face recognition* adalah sebuah tugas yang dikerjakan oleh manusia secara rutin dan mudah dalam kehidupan sehari-hari. Penelitian dan pengembangan ilmu pengenalan wajah berkembang secara otomatis atas dasar ketersediaan *desktop* kuat dan rendah biaya serta *embedded system* yang telah menciptakan minat yang sangat besar dalam pengolahan citra digital dan video. Motivasi penelitian dan pengembangan dari pengenalan wajah termasuk

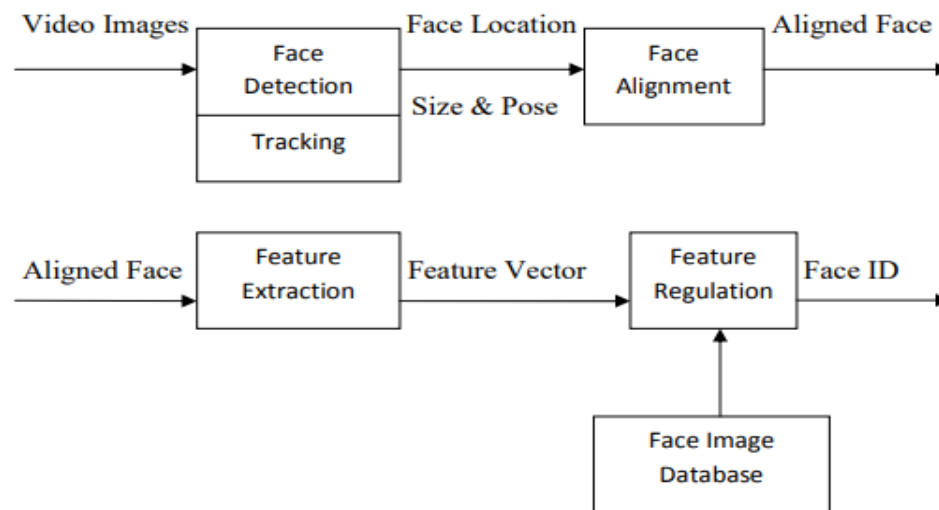
dalam lingkup otentikasi biometric, pengawasan, interaksi manusia komputer, dan manajemen multimedia (Li & Jain, 2005:1).

Pengenalan wajah adalah salah satu ilmu yang terdapat di dalam *computer vision*, di mana sebuah komputer dapat menganalisa suatu citra wajah yang terdapat di dalam sebuah gambar dan dapat menemukan identitas atau data diri dari citra wajah tersebut dengan membandingkan terhadap data-data citra wajah yang sudah disimpan sebelumnya di dalam database. Pada umumnya *face recognition* dilakukan dari sisi depan dengan pencahayaan yang merata ke seluruh wajah. Akan tetapi muncul beberapa permasalahan, seperti posisi wajah, skala atau jarak wajah, orientasi, umur, dan ekspresi wajah. Sistem *face recognition* pada umumnya mencakup empat modul utama (Li & Jain, 2005:2), yaitu: deteksi, *alignment*, ekstraksi fitur dan pencocokan. Proses lokalisasi dan normalisasi (deteksi wajah dan *alignment*) adalah langkah-langkah sebelum proses pengenalan wajah (ekstraksi fitur wajah dan pencocokan) dilakukan.

Deteksi wajah adalah langkah awal untuk melakukan identifikasi wajah atau *face recognition*. Sebuah pendeteksi wajah yang ideal seharusnya mampu mengidentifikasi dan menemukan lokasi dan luas semua wajah yang ada di dalam sebuah gambar tanpa memperhatikan pose, skala, orientasi, umur, dan ekspresi (Li & Jain, 2005:13). Deteksi wajah melakukan segmentasi area citra wajah dengan bagian latar (*background*).

Proses *alignment* bertujuan untuk memperoleh akurasi yang lebih baik dan tinggi untuk lokalisasi dan normalisasi citra wajah sebab deteksi wajah menyediakan batas lokasi dan skala dari setiap citra wajah yang dapat terdeteksi.

Setelah sebuah wajah dilakukan normalisasi, ekstraksi fitur dilakukan untuk mengambil data yang efektif yang berguna untuk memisahkan antara citra wajah dan orang-orang yang berbeda satu sama lain dan cukup stabil untuk bermacam-macam geometric dan fotometrik. Pencocokan wajah dilakukan dengan cara melakukan pencocokan fitur yang telah diekstraksi dari citra wajah masukan dengan kumpulan data latihan dan uji coba citra wajah yang telah tersimpan sebagai database wajah.



Gambar 2. 1. Proses Pengenalan Wajah

Citra wajah direpresentasikan sebagai sebuah susunan pixel dengan dimensi tinggi. *Face recognition*, dan penelitian *computer vision* secara umum, telah mengamati pertumbuhan minat dalam teknik yang menerapkan aljabar dan fitur statistik untuk melakukan ekstraksi dan analisis kasus jenis ini. Analisa komputer untuk citra wajah dipengaruhi oleh sinyal visual (cahaya yang terpantul pada permukaan wajah) yang disimpan oleh sebuah sensor digital sebagai sebuah susunan dari nilai pixel. Nilai pixel ini menyimpan warna atau hanya intensitas cahaya. *Array pixel* dari citra wajah berukuran $m \times n$ dapat disimpan dalam bentuk

sebuah trik (contoh: vektor) di dalam sebuah ruang citra *Face Detection Tracking Face Alignment Feature Extraction Feature Regulation Face Image Database* mendimensi dengan menulis nilai-nilai pixel-nya dalam urutan tetap. Masalah utama dari data multidimensi adalah dimensionalitasnya, jumlah koordinat yang diperlukan untuk menspesifikasikan sebuah titik data. Jumlah dimensionalitas ruang tersebut, yang dibuat dari representasi pixel $m \times n$, adalah jumlah yang sangat tinggi bahkan untuk sebuah ukuran citra wajah yang sangat sederhana. Metode pengenalan wajah untuk mengoperasikan representasi ini mengalami beberapa kesulitan. Menurut Shakhnarovich & Moghaddam, (2005, p. 141), beberapa dari hambatan ini juga disebut curse dari dimensionalitas.

- a. Menangani dimensionalitas yang tinggi, terutama dalam konteks pengenalan berdasarkan kecocokan memerlukan biaya yang mahal secara komputasi.
- b. Untuk metode parametrik, jumlah parameter yang diperlukan untuk berkembang secara eksponensial berdasarkan dimensionalitasnya. Terkadang jumlah nya lebih banyak dari jumlah citra yang disediakan untuk latihan dan uji coba.
- c. Untuk metode non-parametrik, kompleksitas dari sampel cukup tinggi.

Menurut Turk (2005:3), pengenalan wajah bisa dilihat sebagai suatu cara untuk secara tepat mengenali citra dari sebuah wajah, dengan menggunakan data dari wajah yang telah lebih dahulu dikenal. Pengenalan wajah memiliki semua hambatan dalam pengenalan yang berdasarkan pemrosesan citra. Dikarenakan citra yang digunakan dapat berubah-ubah secara drastis disebabkan

beberapa faktor yang rumit dan membingungkan, seperti faktor pencahayaan, posisi kamera, pengaturan kamera, dan noise.

Jadi hasil dari *face recognition* akan berupa informasi yang dikenal atau tidak sebagai wajah dengan sebelumnya membandingkan dengan informasi dari wajah yang diketahui. Proses *face recognition* ini memiliki permasalahan dari pencahayaan, posisi kamera, parameter kamera dan *noise* yang didapatkan pada sebuah citra.

4. JavaScript

JavaScript adalah bahasa pemrograman tingkat tinggi dan dinamis. *JavaScript* populer di internet dan dapat bekerja di sebagian besar penjelajah web populer seperti *Google Chrome*, *Internet Explorer (IE)*, *Mozilla Firefox*, *Netscape* dan *Opera*. Kode *JavaScript* dapat disisipkan dalam halaman web menggunakan tag *SCRIPT*. *JavaScript* merupakan salah satu teknologi inti *World Wide Web* selain *HTML* dan *CSS*. *JavaScript* membantu membuat halaman web interaktif dan merupakan bagian aplikasi web yang esensial.

Awalnya hanya diimplementasi sebagai *client-side* dalam penjelajah web, kini *engine JavaScript* disisipkan ke dalam perangkat lunak lain seperti dalam *server-side* dalam server web dan basis data, dalam program *non web* seperti perangkat lunak pengolah kata dan pembaca *PDF*, dan sebagai *runtime environment* yang memungkinkan penggunaan *JavaScript* untuk membuat aplikasi *desktop* maupun *mobile*. *Javascript* pertama kali dikembangkan pada pertengahan dekade 90an. *Javascript* dapat disisipkan di dalam dokumen *HTML* ataupun

dijadikan dokumen tersendiri yang kemudian diasosiasikan dengan dokumen lain yang dituju. (Flanagan, 2006)

5. Tensorflow.js

TensorFlow™ adalah koleksi *software open source* untuk komputasi numerik yang menggunakan grafik aliran data. *Node* pada grafik menunjukkan operasi matematika, sedangkan tepi-tepi grafik menunjukkan susunan data multidimensi (*Tensor*) yang dikomunikasikan antartepi grafik. Arsitekturnya yang fleksibel dapat dimanfaatkan untuk menerapkan komputasi ke satu atau beberapa CPU atau GPU pada *desktop*, server, atau perangkat seluler dengan cukup menggunakan satu *API*. TensorFlow awalnya dikembangkan oleh para peneliti dan *engineer* yang bekerja di Google Brain Team dalam organisasi riset *Machine Intelligence Google*, untuk keperluan menjalankan *machine learning* dan riset jaringan *deep neural network*. Namun, sistem ini juga cukup untuk diterapkan pada aneka domain lainnya. TensorFlow ini digunakan sebagai *backend* dari Keras, yang artinya adalah TensorFlow sebagai bagian belakang layar dari sebuah aplikasi atau *website* dengan menggunakan bahasa pemrograman *Python* dan dikombinasikan dengan *library Keras* dari *deep learning*. (Deka M Wildan, 2014)

6. Vue.js

Vue js ialah *progressive framework* yang digunakan untuk membangun tampilan halaman *website*. Berbeda dengan *framework* lain yang menggunakan prinsip MVC (*Model View Controller*) *Vue js* hanya difokuskan untuk membangun tampilan/ hanya bekerja pada *view layer*. Dalam hal lainnya *Vue js* juga dapat

membantu programmer untuk membuat *web* dengan aplikasi *single page* yang canggih dan dapat di kombinasikan dengan *library browser* lainnya.

Selain itu *Vue js* menggunakan template *sintaks* berbasis HTML yang memungkinkan pengguna untuk mendeklarasikan data/ *state* kedalam DOM. Semua *template Vue js* adalah HTML yang valid yang dapat diuraikan oleh *browser* sesuai spesifikasi dan parser HTML. (Madcoms Madiun, 2012)

7. Node.js

Node.js merupakan salah satu peranti pengembang yang bisa digunakan untuk membuat aplikasi berbasis *Cloud*. *Node.js* dikembangkan dari engine *JavaScript* yang dibuat oleh *google* untuk browser *Chrome/ Chromium* (V8) ditambah dengan *libUV* serta beberapa pustaka internal lainnya. Dengan menggunakan *Node.js*, semua pengembangan akan dilakukan menggunakan *JavaScript*, baik pada sisi klien maupun server. *Node.js* dibuat pertama kali oleh Ryan Dahl (twitter.com/ryah) dan sampai saat ini dikembangkan oleh komunitas sebagai perangkat lunak bebas dengan pendanaan utama dari *Joyent*, perusahaan tempat Ryan Dahl bekerja. (Purnomosidi, 2013 : 15)

8. MongoDB

MongoDB adalah database dokumen dengan skalabilitas dan fleksibilitas yang anda inginkan dengan *query* dan pengindeksan yang anda butuhkan. *MongoDB* merupakan salah satu database *NoSQL*, yang merupakan sebuah konsep penyimpanan data tanpa perlu adanya *table schema* dan tidak ada bahasa *sql* yang terlibat dalam pemakaian database.

Unit terkecil pada *MongoDB* adalah *document*, lalu kumpulan dari *document* adalah *collection*. Dokumen dalam *MongoDB* memiliki atribut yang berbeda-beda dengan dokumen yang lainnya walaupun dalam satu *collection*. Metode penyimpanan yang digunakan oleh *MongoDB* adalah *object-oriented* dan dukungan tipe *index* yang berbeda untuk *lookups* terhadap data tertentu dan memiliki kemampuan *clustering* secara *default*.

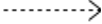
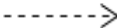
9. UML (Unified Modelling Language)

N, Sora (2020), *Unified Modelling Language (UML)* adalah sekumpulan alat yang digunakan untuk melakukan abstraksiterhadap sebuah sistem atau perangkat lunak berbasis objek. *UML* merupakan singkatan dari *Unified Modeling Language*. *UML* juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena developer harus melakukan penelusuran dan mempelajari kode program *UML* juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu developer ke developer lainnya. Tidak hanya antar developer terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya *UML*. Dalam *UML* sendiri terdapat beberapa diagram yaitu :

a. Use Case Diagram

Adapun simbol-simbol *Use Case Diagram* antara lain :

Tabel 2. 1. Simbol *Use Case Diagram*

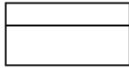
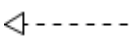
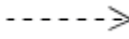
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna <i>Mainkan</i> ketika berinteraksi dengan <i>Use Case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>Use Case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>Use Case</i> target memperluas perilaku dari <i>Use Case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor

NO	GAMBAR	NAMA	KETERANGAN
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

b. *Class Diagram*

Adapun simbol-simbol *Class Diagram* antara lain :

Tabel 2. 2. Simbol *Class Diagram*

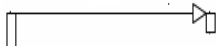
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri

NO	GAMBAR	NAMA	KETERANGAN
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

c. *Sequence Diagram*

Adapun simbol-simbol *Sequence Diagram* antara lain :

Tabel 2. 3. Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

d. *StateChart Diagram*

Adapun simbol-simbol *StateChart Diagram* antara lain :

Tabel 2. 4. Simbol *StateChart Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>State</i>	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		<i>Initial Pseudo State</i>	Bagaimana objek dibentuk atau diawali
3		<i>Final State</i>	Bagaimana objek dibentuk dan dihancurkan

NO	GAMBAR	NAMA	KETERANGAN
4		<i>Transition</i>	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		<i>Node</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

e. Activity Diagram

Adapun simbol-simbol *Activity Diagram* antara lain :

Tabel 2. 5. Simbol *Activity Diagram*

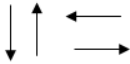
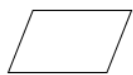
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

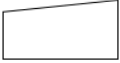
10. Flowchart

Arifianto Rahmad (2017), *Flowchart* adalah adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu

program. Berikut ini adalah beberapa simbol yang digunakan dalam menggambar suatu *Flowchart* :

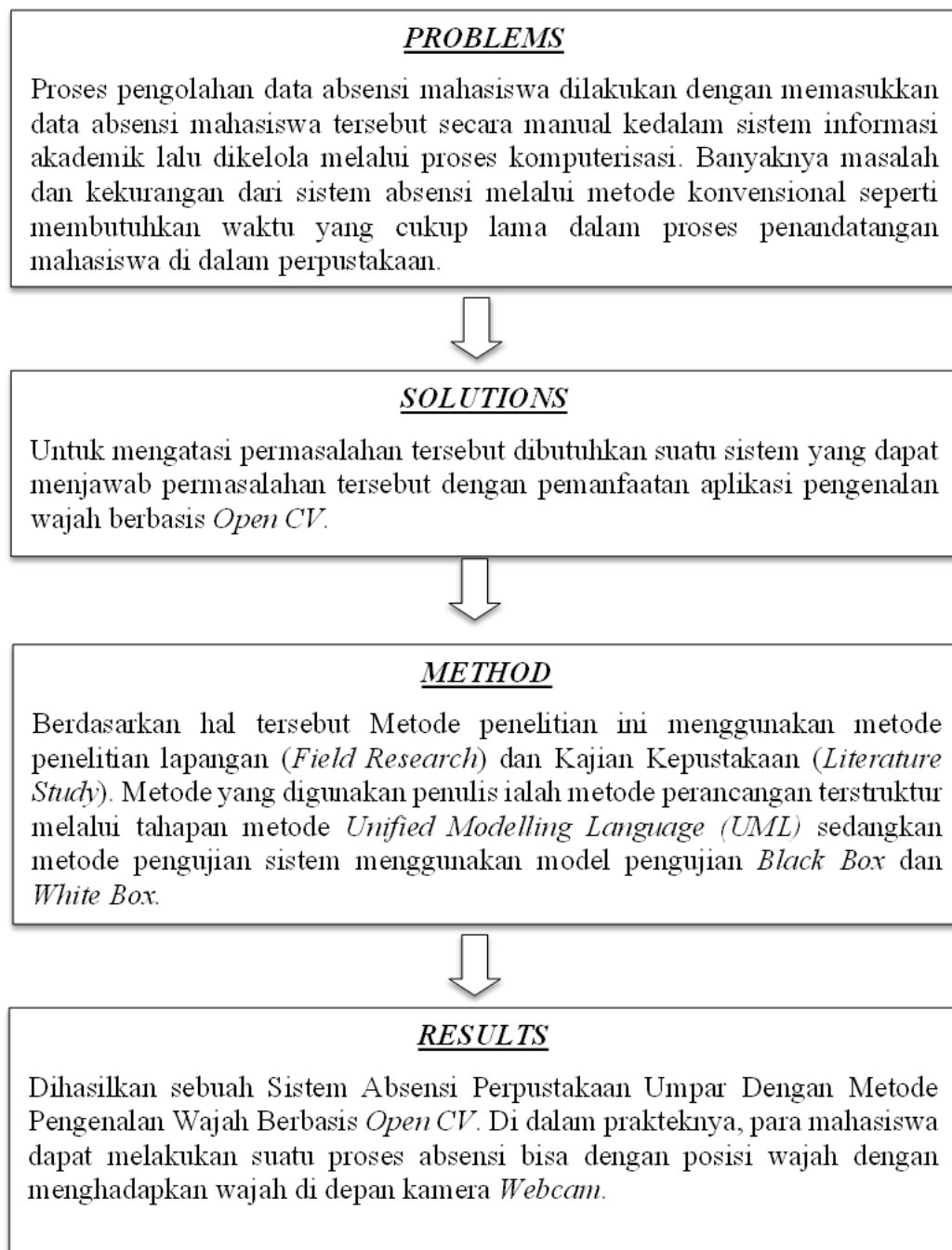
Tabel 2. 6. Simbol *Flowchart*

Simbol	Nama dan Fungsinya
	Terminal Point Symbol / Simbol Titik Terminal Menunjukkan permulaan (start) atau akhir (stop) dari suatu proses.
	Flow Direction Symbol / Simbol Arus adalah simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain (<i>connecting line</i>). Simbol ini juga berfungsi untuk Menunjukkan garis alir dari proses.
	Processing Symbol / Simbol Proses digunakan untuk Menunjukkan kegiatan yang dilakukan oleh komputer. Pada bidang industri (proses produksi barang), simbol ini menggambarkan kegiatan inspeksi atau yang biasa dikenal dengan simbol inspeksi
	Decision Symbol / Simbol Keputusan merupakan simbol yang digunakan untuk memilih proses atau keputusan berdasarkan kondisi yang ada. Simbol ini biasanya ditemui pada <i>Flowchart</i> program.
	Input-Output / Simbol Keluar-Masuk Menunjukkan proses <i>input-output</i> yang terjadi tanpa bergantung dari jenis peralatannya.
	Predefined Process / Simbol Proses Terdefinisi merupakan simbol yang digunakan untuk Menunjukkan pelaksanaan suatu bagian prosedur (<i>sub-proses</i>).
	Connector (On-page) Simbol ini fungsinya adalah untuk menyederhanakan hubungan antar simbol yang letaknya berjauhan atau rumit bila dihubungkan dengan garis dalam satu halaman
	Connector (Off-page) Sama seperti <i>on-page connector</i> , hanya saja simbol ini digunakan untuk menghubungkan simbol dalam halaman berbeda. label dari simbol ini dapat menggunakan huruf atau angka

Simbol	Nama dan Fungsinya
	<i>Preparation Symbol / Simbol</i> Persiapan merupakan simbol yang digunakan untuk mempersiapkan penyimpanan di dalam storage.
	<i>Manual Input Symbol</i> digunakan untuk <i>Menunjukkan</i> input data secara manual menggunakan online keyboard.
	<i>Manual Operation Symbol / Simbol Kegiatan</i> Manual digunakan untuk <i>Menunjukkan</i> kegiatan/proses yang tidak dilakukan oleh komputer.
	<i>Document Symbol</i> Jika Anda menemukan simbol ini artinya input berasal dari dokumen dalam bentuk kertas, atau output yang perlu dicetak di atas kertas.
	<i>Multiple Documents</i> sama seperti <i>document symbol</i> hanya saja dokumen yg digunakan lebih dari satu dalam simbol ini
	<i>Display Symbol</i> adalah simbol yang menyatakan penggunaan peralatan output, seperti layar monitor, printer, plotter dan lain sebagainya
	<i>Delay Symbol</i> sesuai dengan namanya digunakan untuk <i>Menunjukkan</i> proses delay (<i>Menunggu</i>) yang perlu dilakukan. Seperti <i>Menunggu</i> surat untuk diarsipkan

C. Kerangka Pikir

Untuk lebih memperjelas kerangka pikir maka digambarkan dalam bentuk diagram sebagai berikut :



Gambar 2. 2. Kerangka Pikir

BAB III

METODOLOGI PENELITIAN

A. Tempat dan Waktu Penelitian

1. Penelitian yang dilakukan oleh penulis bertempat di Perpustakaan Universitas Muhammadiyah Parepare (Umpar) Jl. Jend. Ahmad Yani No.km.6, Bukit Harapan, Kec. Soreang, Kota Parepare, Sulawesi Selatan 91112.
2. Pelaksanaan penelitian selama ($\pm$) 2 bulan.

B. Metode Penelitian

Untuk membantu kelancaran pengumpulan data, maka penulis menggunakan beberapa metode antara lain :

a. Metode Analisis

1. Tinjauan Pustaka (*Literature Review*), dilakukan dengan cara mengumpulkan data-data secara teoritis yang berhubungan dengan masalah yang menjadi objek penelitian dari literatur-literatur serta buku-buku dari perpustakaan yang digunakan sebagai landasan teori.
2. Pengamatan (*Observation*), proses mengumpulkan data-data yang diperlukan dengan mengadakan kunjungan secara langsung di Universitas Muhammadiyah Parepare, mengamati langsung proses pelaksanaan ujian komprehensif serta bertemu dengan pihak terkait, data-data yang diperoleh kemudian dituangkan dalam bentuk kata-kata maupun skema.

b. Wawancara (*Interview*)

Metode ini dengan cara mengumpulkan data-data yang diperlukan dengan mengadakan wawancara secara langsung dengan Ketua Perpustakaan, Sekretaris Perpustakaan, Staff Perpustakaan serta pihak-pihak yang terkait.

c. Kajian Pustaka

Tahapan penelitian yang dimaksud dalam penelitian ini ada beberapa tahapan yaitu persiapan penelitian, pengumpulan data analisis, perancangan, pengujian dan implementasi. Adapun Uraian dari tahapan tersebut adalah sebagai berikut :

1. Persiapan Penelitian

Pada tahapan ini peneliti melakukan persiapan penelitian. Persiapan penelitian yang dimaksud adalah menyiapkan buku-buku, artikel-artikel tentang topik penelitian serta *software* yang digunakan selama penelitian.

2. Studi Literature

Pada tahapan ini peneliti melakukan apa yang disebut dengan kajian pustaka, yaitu mempelajari buku-buku referensi dan hasil penelitian sejenis sebelumnya yang pernah dilakukan oleh orang lain. Tujuannya ialah untuk mendapatkan landasan teori mengenai masalah yang akan diteliti. Teori merupakan pijakan bagi peneliti untuk memahami persoalan yang diteliti dengan benar dan sesuai dengan kerangka berpikir ilmiah.

C. Metode Pengumpulan Data

Pada tahap ini peneliti melakukan wawancara dengan pihak yang terkait dengan penelitian, serta melakukan pencatatan dan pengamatan langsung di tempat penelitian.

1. Analisis

Pada tahap analisis, peneliti melakukan analisa terhadap sistem yang di terapkan sekarang berdasarkan kemudian merumuskan masalah yang menjadi pokok penelitian sehingga dapat dibuat alternatif pemecahan masalah.

2. Perancangan

Peneliti kemudian merancang aplikasi yang ingin dibuat berdasarkan alternatif pemecahan masalah.

3. Pengujian

Setelah melakukan perancangan, peneliti kemudian menguji hasil perancangan yang telah dibuat. Jika hasil perancangan terdapat kekurangan atau kelemahan maka kembali ke tahap analisis.

4. Implementasi

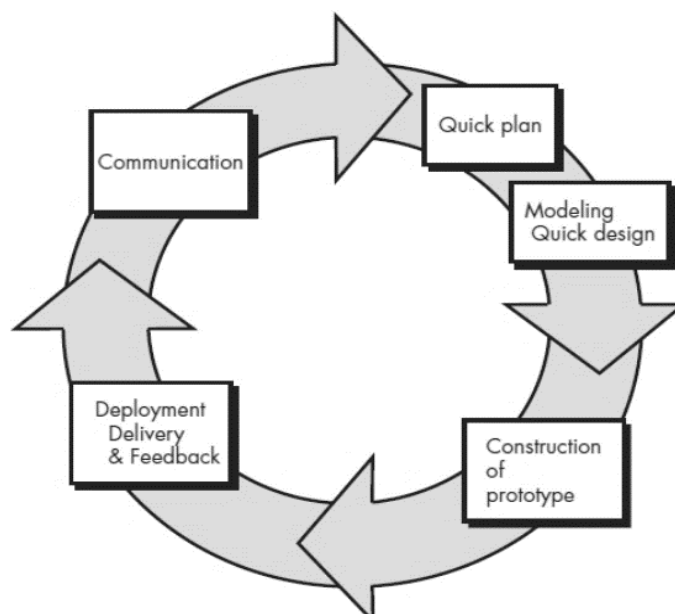
Setelah pada perancangan tidak terdapat kekurangan maka aplikasi siap untuk digunakan oleh *user*.

D. Metode Pengembangan Sistem

Metode Pengembangan sistem merupakan suatu kerangka kerja mengenai bagaimana proses pelaksanaan Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*, adapun metode pengembangan yang digunakan yaitu *Prototype* hal ini dikarenakan pada Teknik Informatika UM

Parepare belum sama sekali memiliki aplikasi yang berkaitan dengan ujian komprehensif sehingga sangat cocok menggunakan metode *Prototype*.

Menurut Pressman (2012), dalam melakukan perancangan sistem yang akan dikembangkan dapat menggunakan metode *Prototype*. Metode ini cocok digunakan untuk mengembangkan perangkat-perangkat lunak yang akan dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna, dalam hal ini pengguna dari perangkat yang dikembangkan adalah Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*. Tahap berikutnya membuat rancangan kilat yang selanjutnya akan dievaluasi kembali sebelum diproduksi secara benar. Metode pengembangan sistem yang digunakan adalah model *prototype* dengan 5 (lima) tahapan yaitu *communication*, *quick plan*, *modeling quick design*, *construction of prototipe*, *deployment, delivery & feedback*.



Gambar 3. 1. Model Prototype

Berikut ini adalah tahapan-tahapan dalam metode *Prototype* adalah sebagai berikut :

1. *Communication*

Pada tahap ini, antara pelanggan dengan tim pengembang perangkat lunak berkomunikasi mengenai spesifikasi suatu kebutuhan yang diinginkan. Tim pengembang melakukan pengumpulan data yang akan digunakan dalam pembuatan sistem yang dibutuhkan.

2. *Quick Plan*

Tim pengembang memberikan gambaran besar mengenai sistem yang akan dibuat kepada perpustakaan dan sub bidang staff. Dalam hal ini sistem yang akan dibuat adalah Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*, kemudian pengembang merancang penggunaan spesifikasi *hardware* dan *software* yang akan digunakan dalam pembuatan sistem.

3. *Modeling Quick Design*

Pada tahap ini, pengembang membuat perancangan sistem menggunakan *use case diagram*, *activity diagram*, *class diagram*, *sequence diagram* dan *user interface*.

4. *Construction of Prototype*

Setelah tahap pemodelan, maka pengembang mulai melakukan pengkodean program. Bahasa pemrograman yang digunakan adalah *Tensorflow.js*, *JavaScript* dan *HTML* serta basis data *Mongodb*.

5. *Deployment, Delivery & Feedback*

Tahap pengkodean program dibarengi oleh tahapan implementasi dan pengujian sistem. Sistem diuji dengan metode *black box testing* dengan pengujian fungsional. Setelah sistem lulus uji coba, maka sistem yang baru diberikan kepada pelanggan dan pengembang mengajari pelanggan dalam penggunaan sistem yang baru. Jika ada kekurangan atau penambahan kebutuhan sistem, maka pihak terkait akan mengkomunikasikan kembali dengan tim pengembang.

E. Sumber Data Penelitian

Sumber data pada penelitian ini adalah menggunakan *Field Research* atau penelitian lapangan yang merupakan cara mengumpulkan data melalui pengamatan secara langsung orang-orang atau keadaan lokasi penelitian yang dapat dijadikan acuan pembahasan dalam masalah ini. Adapun jenis dan sumber data yang dibutuhkan dalam penelitian ini adalah sebagai berikut :

1. Data Primer yaitu data yang diperoleh melalui wawancara yang dilakukan langsung dengan responden yang dapat mewakili beberapa Sumber data. Dalam penelitian ini merupakan subjek penelitian utama yakni Mahasiswa/i Informatika Universitas Muhammadiyah Parepare.
2. Data sekunder yaitu merupakan data yang diperoleh melalui studi kepustakaan atau dari berbagai literatur dengan menelaah buku-buku dan tulisan-tulisan atau internet, jurnal hukum, serta peraturan perundang-undangan yang relevan dengan permasalahan yang diteliti.

F. Alat dan Bahan

Tahap desain dilakukan untuk menentukan ruang lingkup pembuatan aplikasi yang ditangani, yang selanjutnya dijabarkan dalam bentuk diagram alir. Dalam pembuatan program aplikasi tersebut penulis membutuhkan perangkat keras (*Hardware*) dan perangkat lunak (*Software*). Adapun *Hardware* dan *Software* yang digunakan adalah :

1. Perangkat keras (*Hardware*)

Perangkat keras yang digunakan untuk membuat aplikasi menggunakan PC (*Personal Komputer*) dengan spesifikasi sebagai berikut :

- a. *Processor Intel Core i7 – 7500U Processor (up to 4.0GHz)*
- b. *Kapasitas Memory 8 GB DDR 4 2400MHz*
- c. *Kapasitas Hardisk 1 TB 5400 RPM*
- d. *Layar Monitor 14.0” LED backlit FHD*

2. Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam pembuatan aplikasi adalah :

- a. *Sistem Operasi Windows 10*
- b. *XAMPP Control Panel dari Web Server*
- c. *Javascript Sebagai Bahasa Web Program*
- d. *MonggoDB untuk Database*
- e. *Visual Studio Code yaitu Tools*
- f. *Vue.js, adobe Photoshop dan bootstrap*

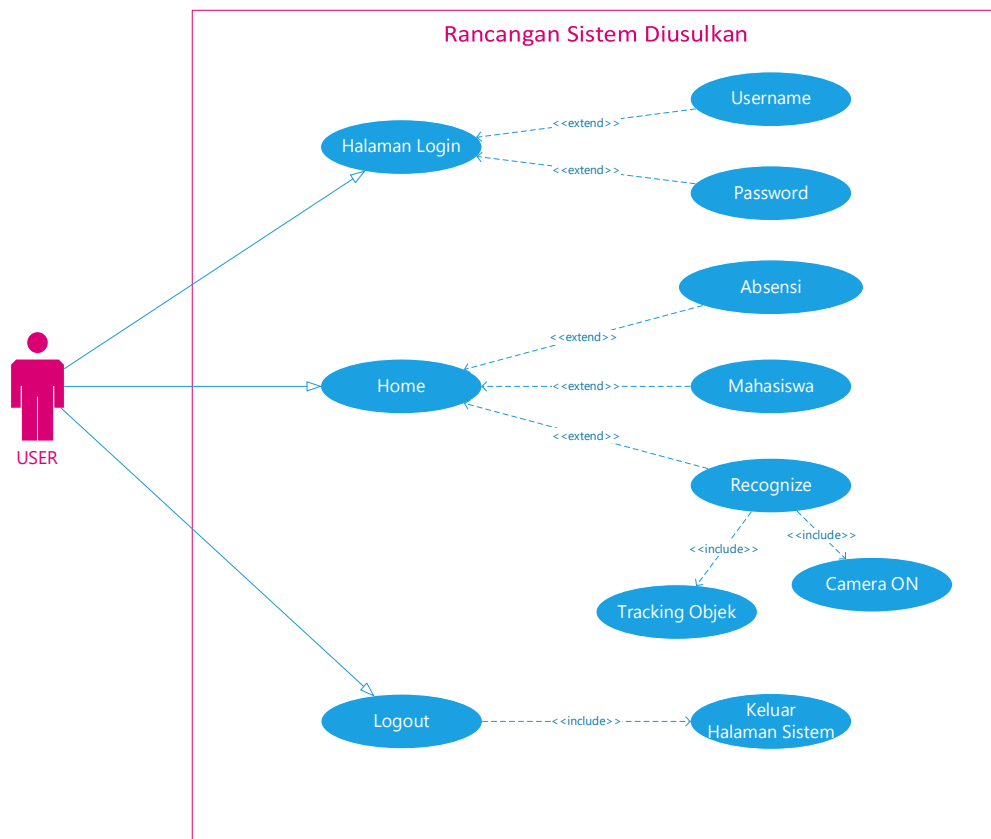
BAB IV

HASIL DAN PEMBAHASAN

A. Rancangan Sistem

1. Analisis Sistem Diusulkan.

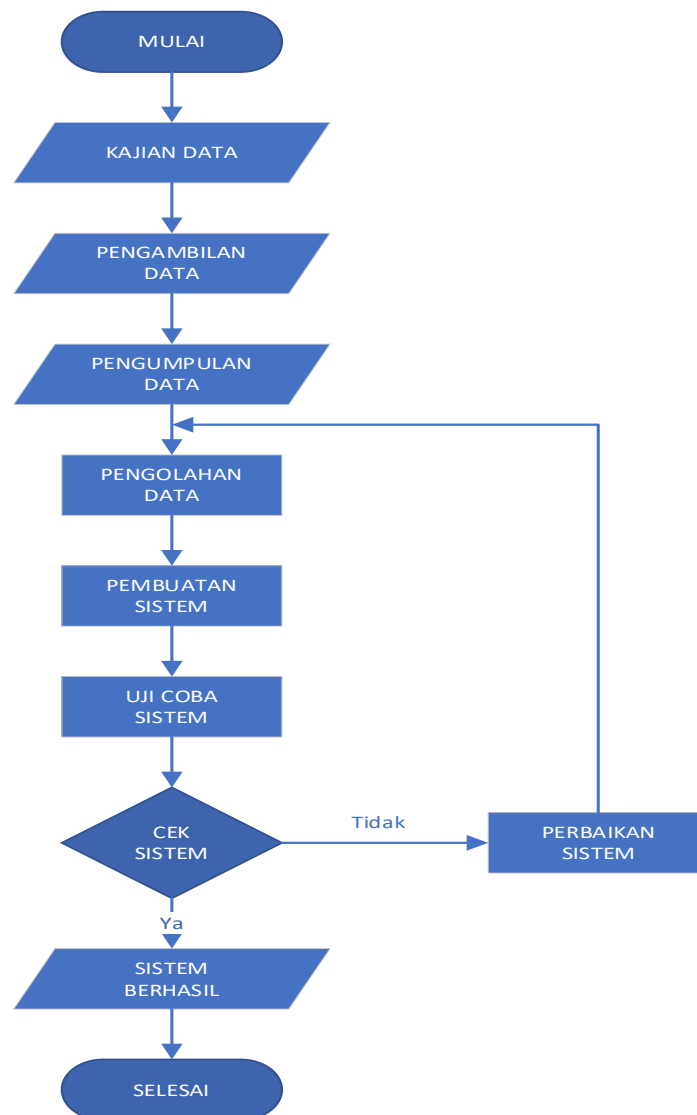
Analisis sistem diusulkan menggunakan *Use Case Diagram*. *Use case diagram* menggambarkan interaksi antara *actor* dengan proses atau sistem yang dibuat. *Use case* ini meliputi semua hal yang ada pada sistem, sedangkan *actor* meliputi semua hal yang ada diluar sistem. Aktor termasuk seseorang atau apa saja yang berhubungan dengan sistem yang dibangun.



Gambar 4. 1 Use Case Diagram Sistem Diusulkan

2. Flowchart

Flowchart merupakan bagan yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan dan menjelaskan urutan dari prosedur-prosedur yang ada di dalam alur penelitian. Berikut gambaran *flowchart* alur aplikasi dapat dilihat sebagai berikut :



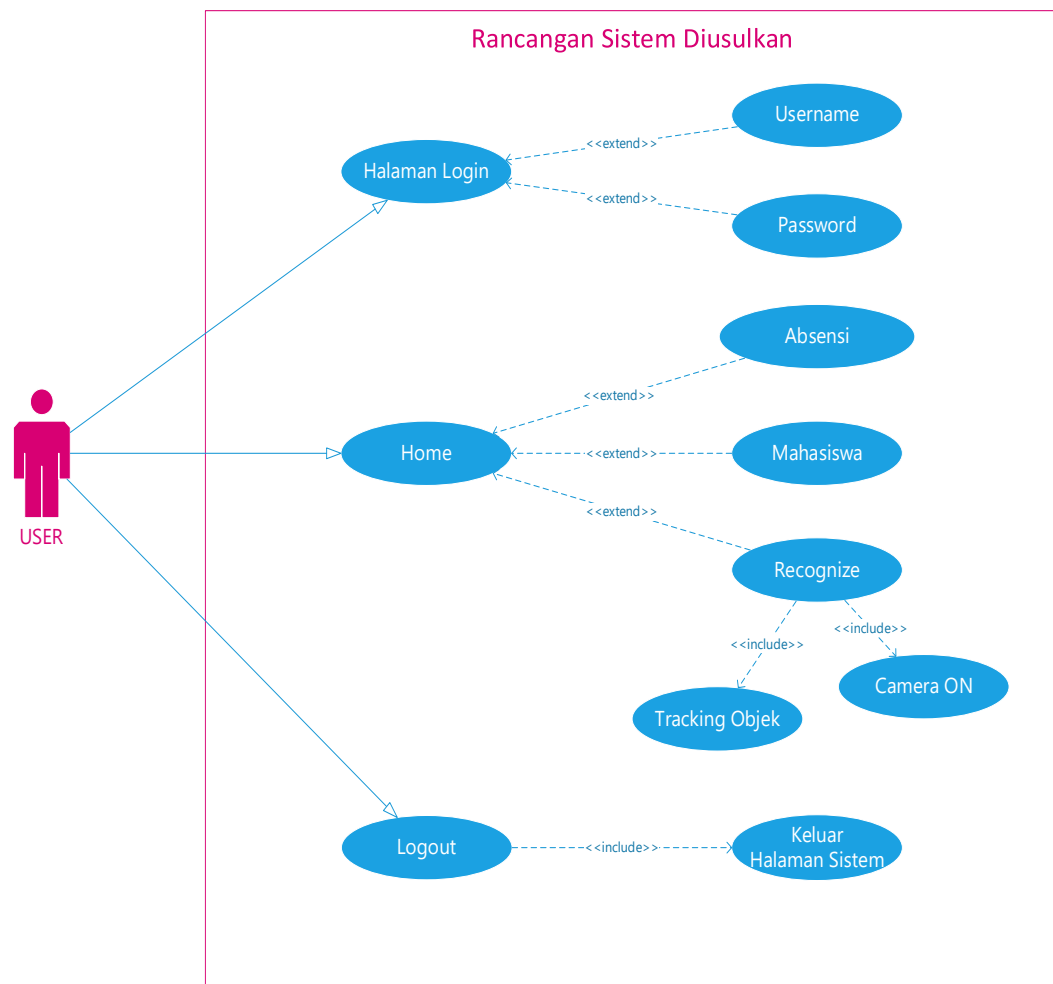
Gambar 4. 2 *Flowchart Sistem*

B. Analisis Aliran Data UML

Analisis aliran data yang diusulkan ini bertujuan mengetahui proses informasi, dalam analisis sistem penulis menggunakan suatu pengembangan orientasi objek sehingga menggunakan *Use Case Diagram*, *Activity Diagram*, dan *Sequence diagram*.

1. Use Case Diagram

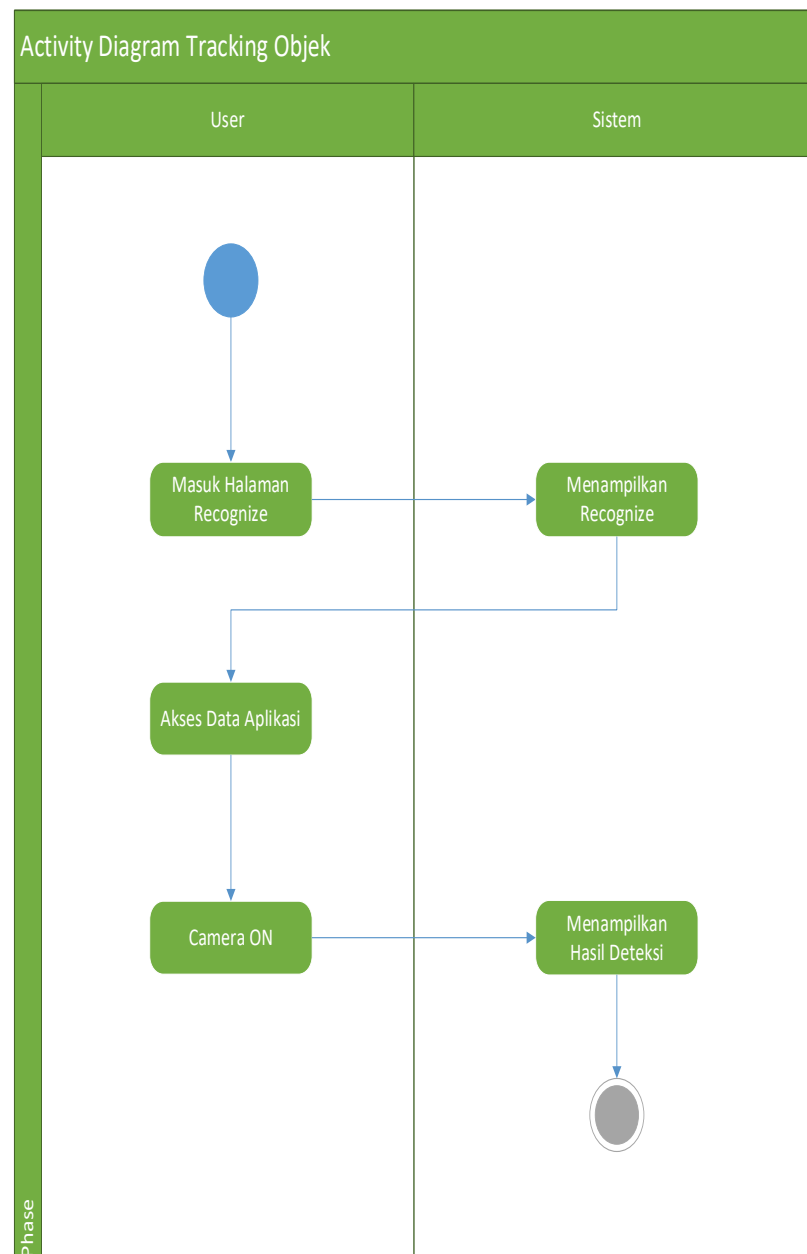
Use Case Diagram berfungsi untuk menjelaskan alur sistem jika dilihat menurut pandangan orang yang berada diluar sistem (aktor).



Gambar 4. 3 Use Case Diagram User

2. Activity Diagram

Activity diagram adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. *Activity diagram* ini menjelaskan tentang aktifitas-aktifitas yang terjadi dalam sebuah aliran proses pada sebuah *system*.

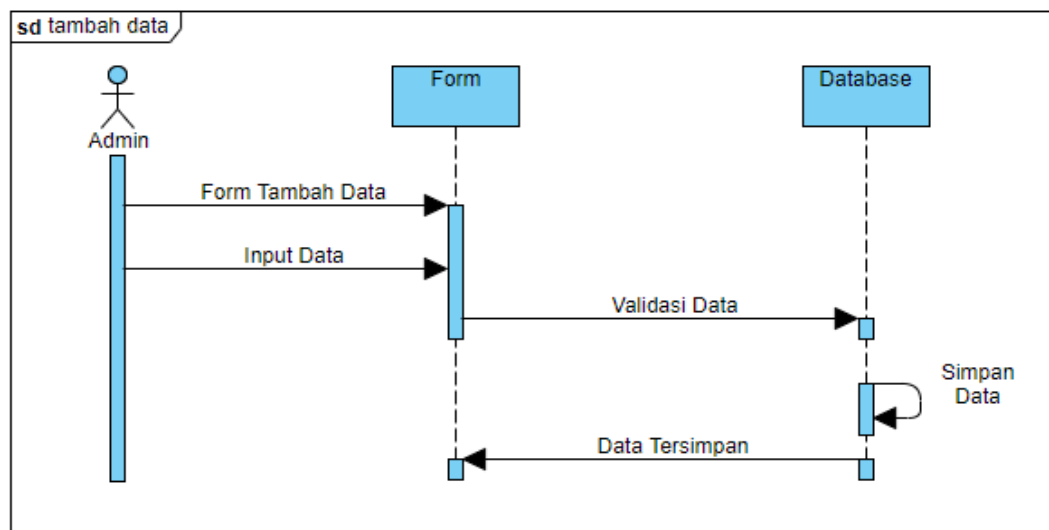


Gambar 4. 4 Activity Diagram Tracking Objek

3. Sequence Diagram

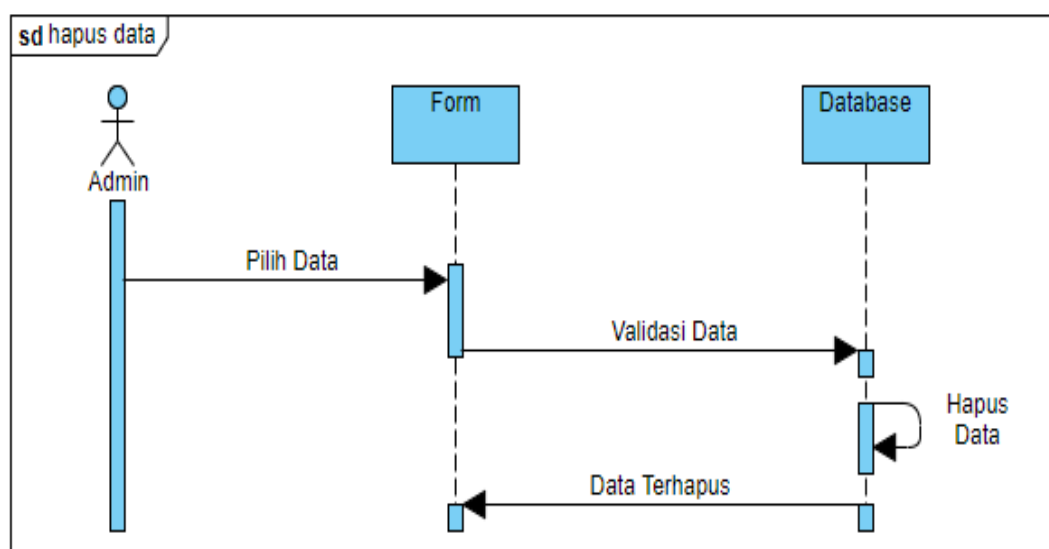
Sequence Diagram merupakan salah satu diagram *Interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan *message* (pesan) apa yang dikirim dan kapan pelaksanaannya.

a. Sequence Diagram Tambah Data



Gambar 4. 5 Sequence Diagram Tambah Data

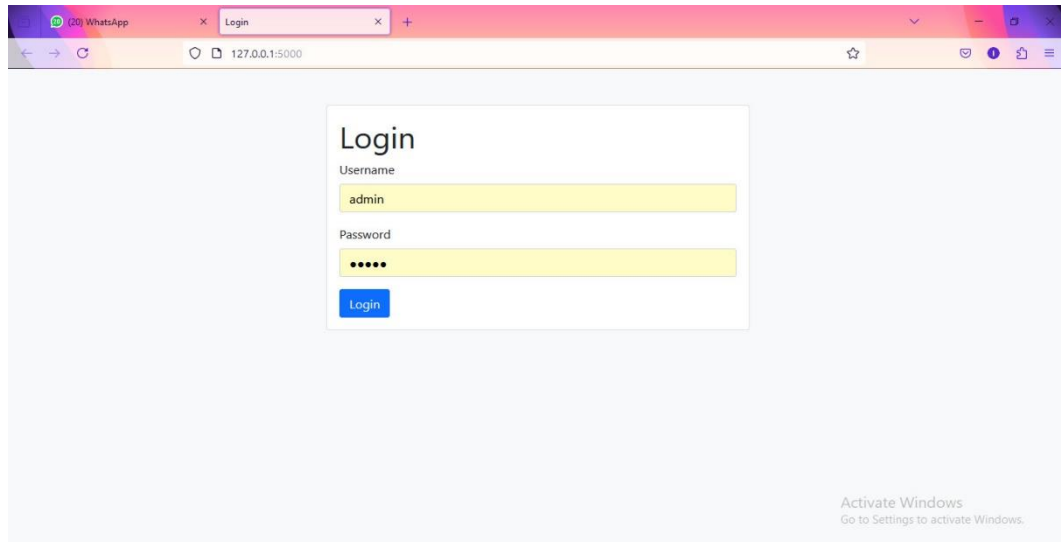
b. Sequence Diagram Hapus Data



Gambar 4. 6 Sequence Diagram Hapus Data

C. Detail Aplikasi

1. Halaman *Login*

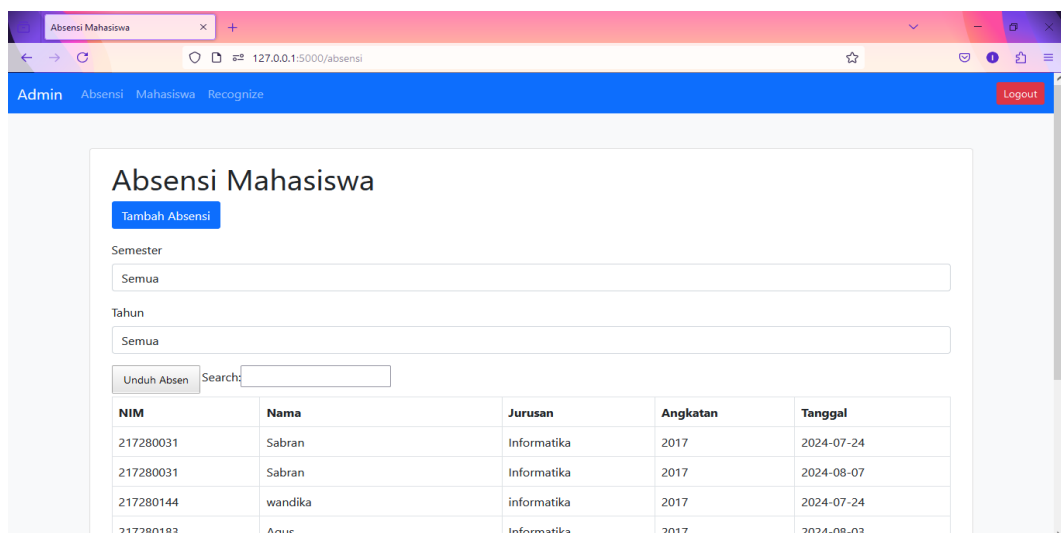


Activate Windows
Go to Settings to activate Windows.

Gambar 4. 7 Halaman *Login*

Halaman *login* merupakan halaman *login admin*. *Login* merupakan tampilan awal yang ada pada sistem, yang digunakan *admin* untuk masuk ke halaman selanjutnya.

2. Halaman Absensi

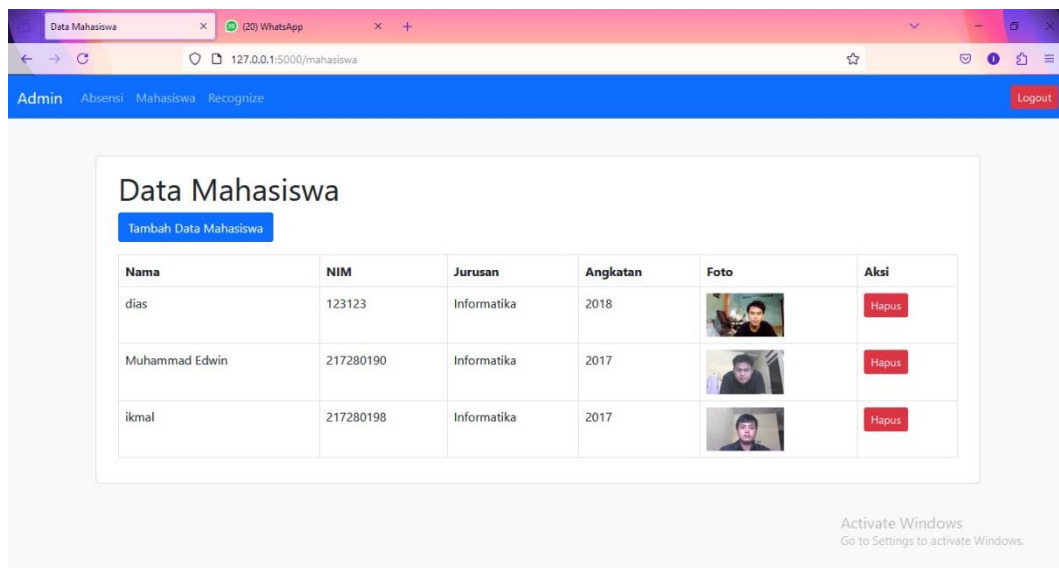


NIM	Nama	Jurusan	Angkatan	Tanggal
217280031	Sabran	Informatika	2017	2024-07-24
217280031	Sabran	Informatika	2017	2024-08-07
217280144	wandika	informatika	2017	2024-07-24
217280183	Aqus	Informatika	2017	2024-08-03

Gambar 4. 8 Halaman *Form Absensi*

Pada menu halaman *form* absensi terdapat pengaksesan untuk *admin*. Halaman ini menampilkan *form* inputan data mahasiswa dan table daftar absensi mahasiswa. Pada halaman ini juga terdapat menu tambah, *edit* dan hapus data mahasiswa.

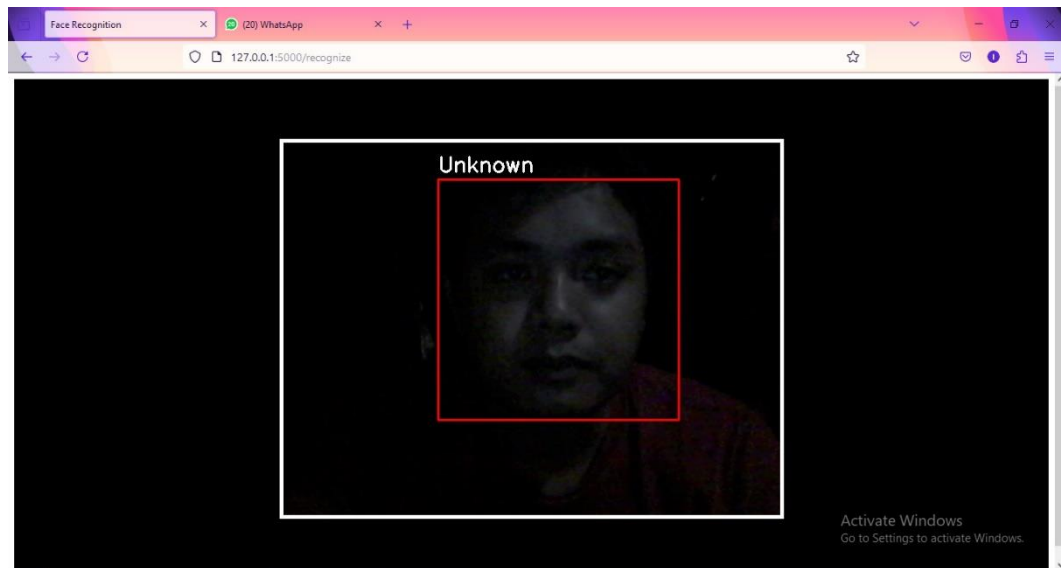
3. Halaman Mahasiswa



Gambar 4. 9 Halaman Mahasiswa

Pada menu halaman *form* mahasiswa terdapat pengaksesan untuk *admin*. Halaman ini menampilkan *form* inputan data mahasiswa dan table daftar mahasiswa. Pada halaman ini juga terdapat menu tambah, *edit* dan hapus data mahasiswa.

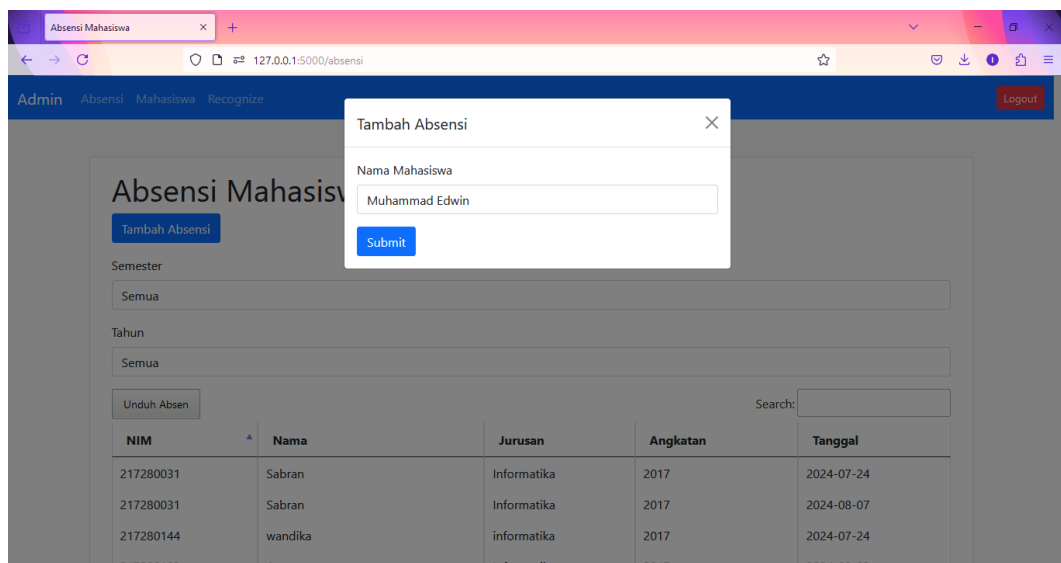
4. Halaman *Recognize*



Gambar 4. 10 Halaman *Recognize*

Pada menu halaman *form recognize* terdapat suatu pengaksesan untuk pengguna. Pada halaman ini merupakan menu *form* yang berada di halaman *home*. *Form recognize* digunakan sebagai absensi menggunakan pengenalan wajah.

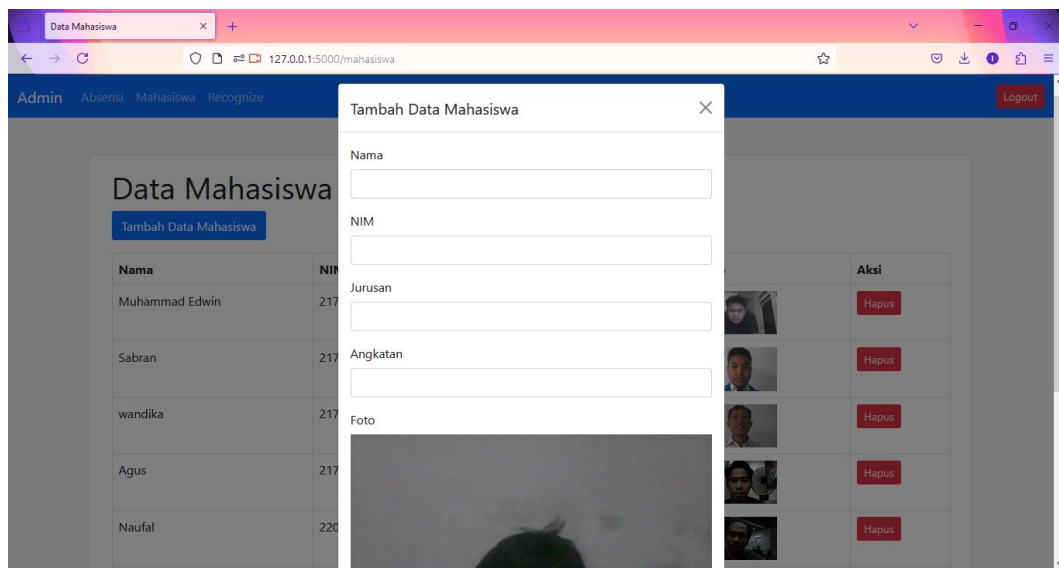
5. Halaman Tambah Data Absensi



Gambar 4. 11 Halaman Tambah Data Absensi

Pada Halaman ini akan tampil ketika *admin* menekan *button* “tambah absensi” pada *form* halaman absensi. Pada halaman ini merupakan tampilan menu tambah data mahasiswa yang hanya bisa diakses oleh *admin*. Halaman menu ini digunakan *admin* untuk memasukkan data-data mahasiswa.

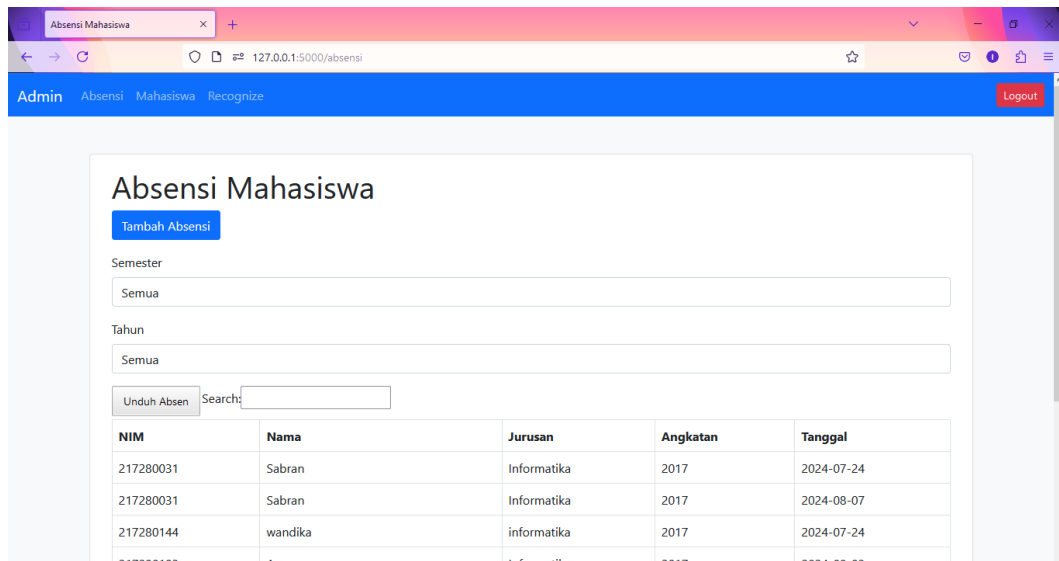
6. Halaman Tambah Data Mahasiswa



Gambar 4. 12 Halaman Tambah Data Mahasiswa

Pada Halaman ini akan tampil ketika *admin* menekan *button* “tambah data mahasiswa” pada *form* halaman absensi. Pada halaman ini merupakan tampilan menu tambah data mahasiswa yang hanya bisa diakses oleh *admin*. Halaman menu ini digunakan *admin* untuk memasukkan data-data mahasiswa berdasarkan Nama, Nim, Jurusan, Angkatan dan *Image*.

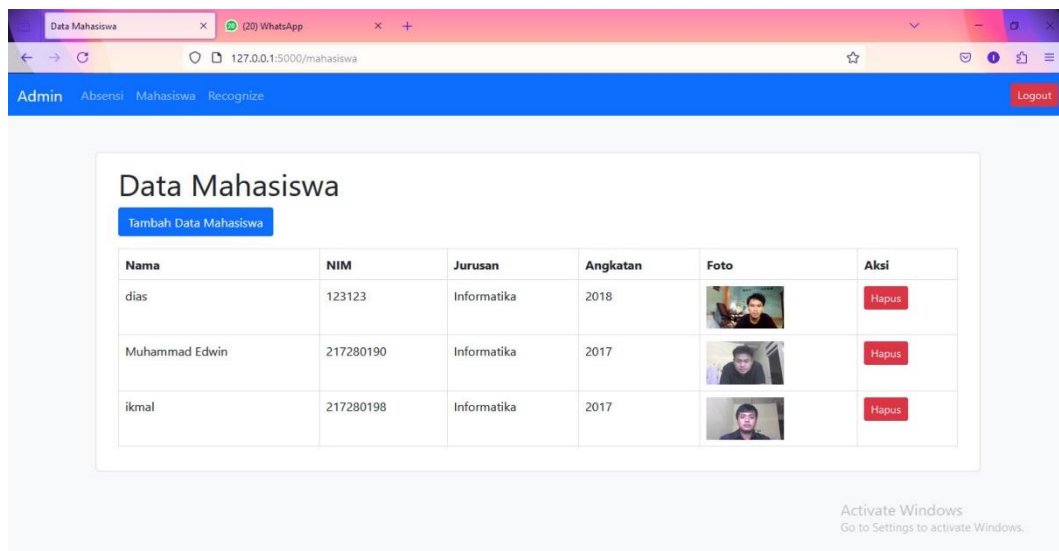
7. Halaman Daftar Absensi



Gambar 4. 13 Halaman Daftar Absensi

Pada menu halaman daftar absensi terdapat pengaksesan untuk *admin*. Pada halaman ini merupakan menu *form* yang berada di halaman *home*. Halaman ini menampilkan tabel daftar absensi berdasarkan ID, Nama Mahasiswa dan Tanggal.

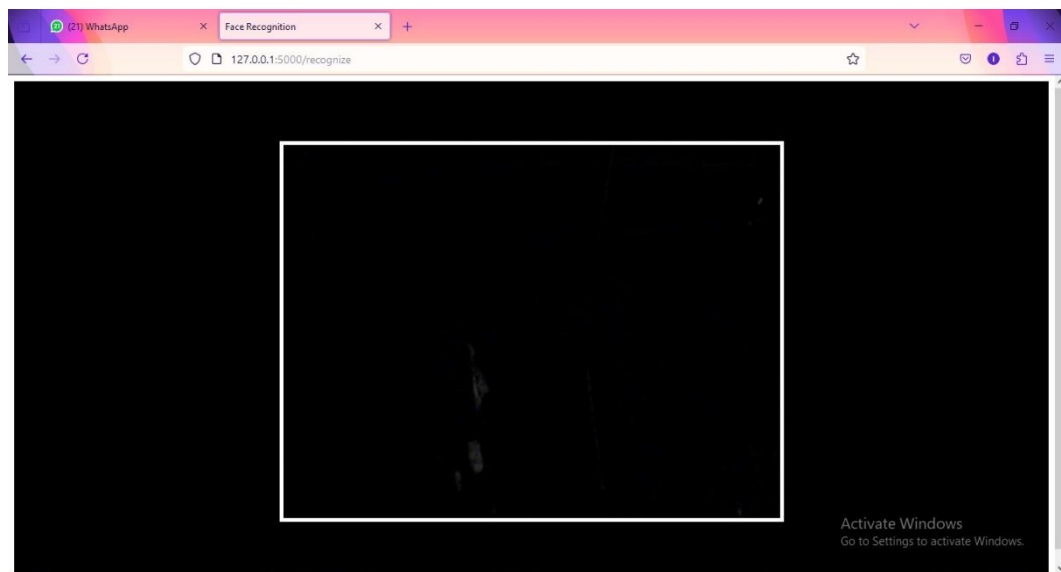
8. Halaman Daftar Mahasiswa



Gambar 4. 14 Halaman Daftar Mahasiswa

Pada menu halaman daftar mahasiswa terdapat pengaksesan untuk *admin*. Pada halaman ini merupakan menu *form* yang berada di halaman *home*. Halaman ini menampilkan table daftar mahasiswa berdasarkan Nama, Nim, Jurusan, Angkatan dan *Image*.

9. Halaman Menu Deteksi

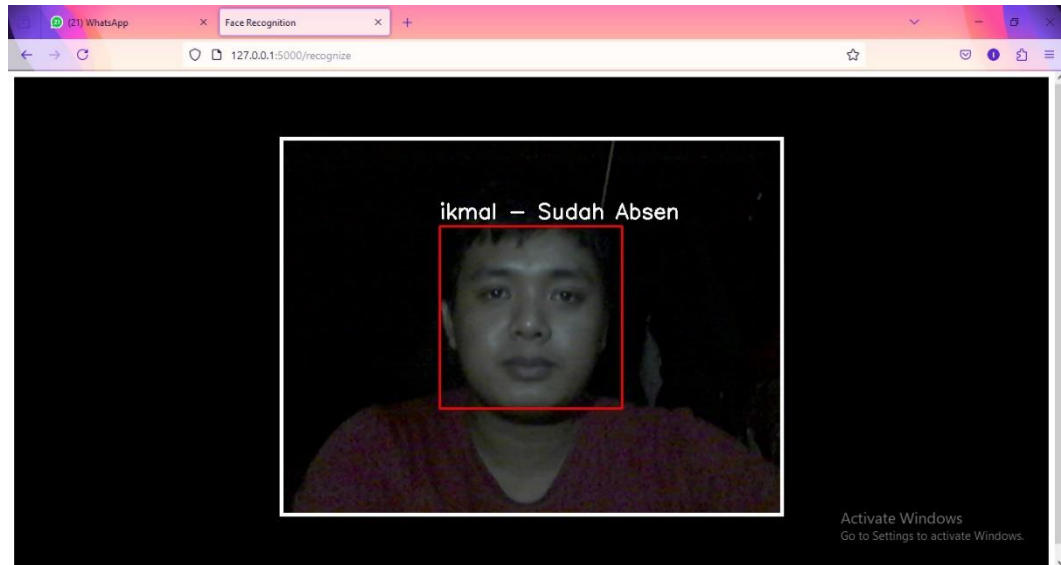


Gambar 4. 15 Halaman Menu Deteksi

Pada halaman menu deteksi ini terdapat pengaksesan untuk pengguna yakni *user*. Halaman deteksi merupakan halaman utama pengguna, halaman ini digunakan untuk melakukan absensi mahasiswa menggunakan pengenalan wajah.

D. Rancangan Output Sistem

Perancangan *output* program merupakan perancangan yang dilakukan untuk mengetahui informasi dan data absensi perpustakaan berupa Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*.



Gambar 4. 16 Halaman Hasil Deteksi

Gambar ini merupakan tampilan hasil *tracking* uang kertas dan kemudian sistem secara otomatis akan melakukan *image processing* menggunakan *open CV* kemudian ditampilkan pada pengenalan wajah yang lainnya sebagai hasil akhir absensi.

E. Implementasi

Adapun kebutuhan yang digunakan untuk merancang dan membangun sebuah sistem dengan judul “Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*”.

1. Kebutuhan Perangkat Keras (*Hardware*)

Spesifikasi *hardware* pengguna aplikasi ini adalah sebagai berikut :

Tabel 4. 1 Spesifikasi *Hardware* Komputer

Jenis	Spesifikasi
<i>Notebook/Komputer</i>	Lenovo Ideapad 100
<i>Processor</i>	<i>Processor Intel Core i7</i>
<i>RAM</i>	8 GB
<i>HDD</i>	1 TB

Tabel 4. 2 Spesifikasi *Device User*

Jenis	Spesifikasi
<i>Jenis Smartphone</i>	<i>Redmi Note 8 Pro</i>
Layar	5,2 inch'
Ram	4 gb

2. Kebutuhan Perangkat Lunak (*Software*)

Spesifikasi *software* pengguna aplikasi ini sebagai berikut :

Tabel 4. 3 Spesifikasi *Software* Komputer

Jenis	Spesifikasi
Sistem Operasi	<i>Windows 10</i>
<i>Tools</i>	<i>Xampp, Visual Studio Code</i>

Tabel 4. 4 Spesifikasi *Software Device User*

Jenis	Spesifikasi
<i>Versi Android</i>	<i>Android 8.1 (Oreo)</i>
<i>Versi MIUI</i>	<i>MIUI 10 Global 9.6.27 Beta</i>

F. Metode pengujian

Pengujian sistem merupakan proses eksekusi suatu program atau sistem secara keseluruhan. Pengujian menunjukkan bahwa fungsi dari perangkat lunak bekerja sesuai dengan spesifikasi dan bahwa persyaratan kinerja telah dipenuhi.

Metode pengujian yang digunakan ada *black-box*. Pengujian *blackbox* perangkat lunak dilakukan untuk memperlihatkan bahwa fungsi telah beroperasi sepenuhnya dan secara bersamaan pula mencari kesalahan pada setiap fungsi.

G. Teknik Pengujian

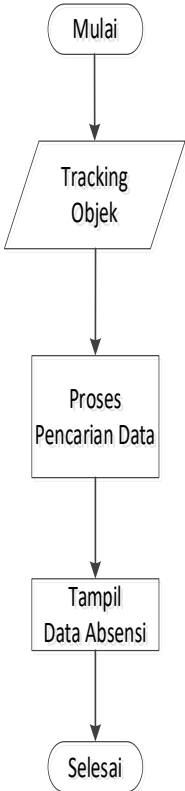
Pengujian *black-box* berkaitan dengan pengujian yang dilakukan pada *interface* perangkat lunak. Meskipun didesain untuk mengungkapkan kesalahan, pengujian *black-box* digunakan untuk memperlihatkan bahwa fungsi-fungsi perangkat lunak adalah operasional, bahwa *input* diterima dengan baik dan *output* dihasilkan dengan tepat, dan integritas informasi *external* (seperti *file* data) dipelihara. Pengetesan sistem merupakan pengetesan program secara keseluruhan, adapun tahap yang dilakukan sebagai berikut :

1. Mekanisme pengujian adalah melakukan pengetesan program aplikasi dengan cara menjalankan aplikasi, pengetesan pemasukan data, perubahan data, dan penghapusan data serta pengetesan terhadap fungsi tombol yang terdapat pada masing-masing halaman apakah berfungsi dengan baik.
2. Hasil yang diperoleh :

Sistem Absensi Perpustakaan Umbar sesuai usulan dapat berjalan dengan lancar sesuai dengan yang diharapkan.

3. Pengujian *White Box*

Tabel 4. 5 Pengujian Menu Deteksi

Deteksi Wajah Absensi	
Flowchart	Flowgraph
 <pre> graph TD A([Mulai]) --> B[/Tracking Objek/] B --> C[Proses Pencarian Data] C --> D[Tampil Data Absensi] D --> E([Selesai]) </pre>	 <pre> graph TD 1((1)) --> 2((2)) 2 --> 3((3)) 3 -- R1 --> 4((4)) 4 --> 5((5)) </pre>

Dari gambar *flowgraph* validasi pengujian menu deteksi diatas dapat dilakukan proses perhitungan sebagai berikut:

- 1) Menghitung *Cyclomatic Complexity* $V(G)$ dari Egde dan *Node*:

Dengan Rumus : $V(G) = E - N + 2$

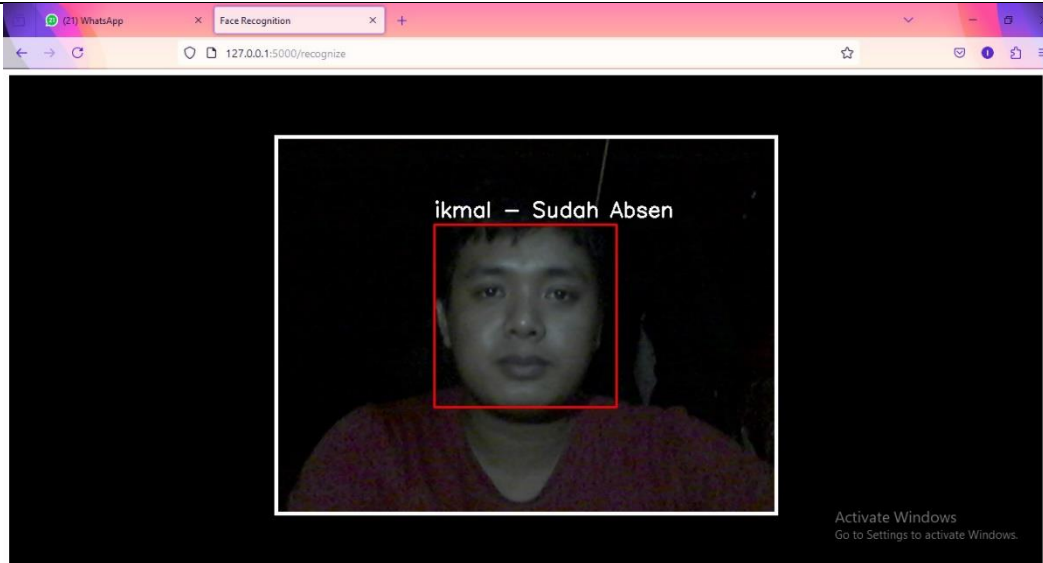
Dengan E (*edge*) = 4

N (*node*) = 5

Predikat *Node* (P) = 0

4. Pengujian *Black Box*

Tabel 4. 7 *BlackBox* Halaman Deteksi Wajah

Test Faktor	Hasil	Kesimpulan
Jika <i>user</i> melakukan <i>tracking</i> wajah pada <i>webcam</i>	✓	Informasi, tampil halaman absensi <i>user</i>
Screen Shot 		

BAB V

PENUTUP

A. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah diuraikan pada bab-bab sebelumnya, Dihasilkannya sebuah Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*. Aplikasi yang dibuat sudah bebas dari kesalahan karena diuji dengan menggunakan dua teknik pengujian yaitu pengujian *Black box* dan pengujian *White box*. Maka dapat ditarik kesimpulan sebagai berikut :

1. Penelitian ini berhasil membangun suatu Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV*. Di dalam prakteknya para mahasiswa/i dapat melakukan proses absensi bisa dengan posisi wajah dengan menghadapkan wajah di depan kamera *Webcam*. Setelah itu, setiap mahasiswa yang telah mendaftarkan kehadirannya, maka data-data mahasiswa tersebut akan masuk kedalam *database* yang ada di dalam sistem absensi dan sebagai pertanda kehadiran setiap mahasiswa secara *real time* yang nantinya akan menghasilkan keluaran nilai.
3. Analisis pengenalan wajah dengan *OpenCV* melibatkan pengumpulan dan preprocessing data, ekstraksi fitur dengan model pra-latih, pelatihan dan evaluasi model neural network, serta optimasi faktor-faktor seperti kualitas data, arsitektur model, parameter pelatihan, augmentasi data, dan teknik regularisasi untuk mencapai tingkat keakuratan yang tinggi.

B. Saran-saran

Dalam Sistem Absensi Perpustakaan Umpar Dengan Metode Pengenalan Wajah Berbasis *Open CV* masih terdapat beberapa kekurangan, sehingga penulis memberikan saran terhadap bagian-bagian tertentu, yaitu :

1. Hendaknya sistem yang penulis rancang dapat dikembangkan sehingga dalam pengolahan datanya dapat lebih dijelaskan secara mendetail, agar pengolahan data dan pembuatan laporan dapat dilakukan lebih efektif dan efisien.
2. Sistem Aplikasi yang telah dikembangkan penulis sebaiknya dipelihara dengan sebaik-baiknya dan di perbaharui sesuai dengan kebutuhan yang ada serta melakukan *backup* data pada jangka waktu tertentu untuk menghindari kemungkinan data hilang atau rusak.

DAFTAR PUSTAKA

- A.S. Nugroho. (2017). “*Mengenal Teknologi Pengenalan Wajah (1)*,” Diperoleh dari:<https://ptik.bppt.go.id/berita-ptik/44-mengenal-teknologi-pengenalan-wajah-1> (Diakses : 1 November 2021)
- A. Varshney, S. Singh, S. Srivastava, S. Chaudhary dan Tanuja. (2019) “*Automated attendance system using face recognition*,” in *Int Research Journal of Engineering and Technology* (IRJET), pp. 7118–7123.
- D. Putra (2010). “*Pengolahan citra digital*”. Penerbit Andi. Hal 26-27.
- F. Syuhada, I. G. P. Suta Wijaya, dan F. Bimantoro. (2018). “*Pengenalan Wajah Untuk Sistem Kehadiran Menggunakan Metode Eigenface dan Euclidean Distance*,” *J. Comput. Sci. Informatics Eng.*, vol. 2, no. 1, pp. 64–69.
- Komputerbanguye. (2016) “*pengertian Visual Studio*” Diperoleh dari:[//komputerbanguye.blogspot.co.id/2016/11/pengertian-microsoft-visual-studio-dan-kegunaanya.html](http://komputerbanguye.blogspot.co.id/2016/11/pengertian-microsoft-visual-studio-dan-kegunaanya.html).
- Knowledge dan Research (2019) “*Tensorflow Oleh Google Brain*” Diperoleh dari:<https://mmsi.binus.ac.id/2019/11/26/tensorflow-oleh-googlebrain> (Diakses : 10 November 2021).
- K. R. Arpiandi. (2017). “*Mengenal Teknologi Deep Learning dan Sejarahnya*,” Diperoleh dari:<https://www.codepolitan.com/mengenal-teknologi-deeplearning-dan-sejarahnya-59aea44b5f64> (Diakses : 3 November 2021).
- Kadek Dwi Pradnyani Novianti (2015). “*Peningkatan Recall dan Precision pada Penelusuran Informasi Pustaka Berbasis Semantik*”. *Konferensi Nasional Sistem & Informatika: STMIK STIKOM*. Hal 704-706.
- M. Y. F. Siddiqui and Sukesha. (2015). “*Face recognition using original and symmetrical face images*,” in *Proc. 2015 on 1st Int. Conf. on Next Generation Computing Technologies, NGCT*, pp. 898–902, September. 4-5.
- MB. Herlambang. (2019) “*Deep Learning: Convolutional Neural Networks (aplikasi)*”, Diperoleh dari:<https://www.megabagus.id/deep-learning-convolutional-neural-networks-aplikasi/2/> (Diakses : 5 November 2021).
- M. Naveenkumar dan A. Vadiel. (2015). “*OpenCV for Computer Vision Applications*,” in *Proc. of Nat. Conf. on Big Data and Cloud Computing*, pp. 52–56.

R. Hartanto dan M. N. Adji. (2018). *"Face recognition for attendance system detection," in.*

Proc. 2018 10th Int. Conf. Inf. Technol. Electr. Eng. Smart Technol. Better Soc. ICITEE 2018, pp. 376–381.

Royani Darma Nurfitra, Gunawan Ariyanto (2018). *"Implementasi Deep Learning Berbasis Tensorflow Untuk Pengenalan Sidik Jari"*. Program Studi Informatika Universitas Muhammadiyah Surakarta (UMS) Surakarta, Indonesia.

Solih. (2006). *"pengertian UML"* Diperoleh dari: [https://repository.bsi.ac.id/index.php/unduh/item/5795/BAB II.pdf](https://repository.bsi.ac.id/index.php/unduh/item/5795/BAB%20II.pdf) (Diakses : 5 November 2021).

Sugiono. (2006). *metode penelitian kuantitatif* Diperoleh dari: <https://penerbitdeepublish.com/metode-penelitian-kualitatif/> (Diakses : 8 November 2021)

T. Suhesti. (2014). *"Bahasa Pemrograman Python,"* Diperoleh dari: <https://docplayer.info/49238665-Bahasa-pemrograman-python.html> (Diakses: 8 November 2021).

T. Mantoro, M. A. Ayu, dan Suhendi. (2018). *"Multi-Faces Recognition Process Using Haar Cascades and Eigenface Methods," Int. Conf. Multimed. Comput. Syst. -Proceedings*, pp. 1–5.