

**PERANCANGAN SISTEM ABSENSI PENGENALAN WAJAH
MAHASISWA PRODI TEKNOLOGI INFORMASI
BERBASIS *WEBSITE***

TUGAS AKHIR

Diajukan Oleh:

ABDUL RAFAR

NIM. 190705076

**Mahasiswa Fakultas Sains dan Teknologi
Prodi Teknologi Informasi**



**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI AR-RANIRY
BANDA ACEH
1444 H/2023**

LEMBAR PERSETUJUAN

PERANCANGAN SISTEM ABSENSI PENGENALAN WAJAH MAHASISWA PRODI TEKNOLOGI INFORMASI BERBASIS *WEBSITE*

TUGAS AKHIR

Diajukan Kepada Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Ar-Raniry Banda Aceh
Sebagai Salah Satu Beban Studi Memperoleh Gelar Sarjana (S1)
dalam Ilmu Teknologi Informasi

Oleh:

Abdul Rafar

NIM. 190705076

**Mahasiswa Fakultas Sains dan Teknologi
Program Studi Teknologi Informasi**

Disetujui untuk Dimunaqasyahkan Oleh:

Pembimbing I,


Khairan AR, M.Kom
NIP. 198607042014031001

Pembimbing II,


Mulkan Fadhli, S.T., M.T
NIP. 198811282020121006

Mengetahui

Ketua Program Studi Teknologi Informasi


Malahayati, MT
NIP. 198301272015032003

LEMBAR PENGESAHAN
PERANCANGAN SISTEM ABSENSI PENGENALAN WAJAH
MAHASISWA PRODI TEKNOLOGI INFORMASI
BERBASIS *WEBSITE*

TUGAS AKHIR

Telah Diuji Oleh Panitia Ujian Munaqasah Tugas Akhir
Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh dan Dinyatakan Lulus
Serta Diterima Sebagai Salah Satu Beban Studi Program Sarjana (S-1)
Dalam Prodi Teknologi Informasi

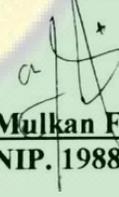
Pada Hari/Tanggal: Rabu, 8 Mei 2024 M
29 Syawal 1445 H
di Darussalam, Banda Aceh
Panitia Ujian Munaqasah Tugas Akhir

Ketua,



Khairan AR, M.Kom
NIP. 198607042014031001

Sekretaris,



Mulkan Fadhli, S.T., M.T
NIP. 198811282020121006

Penguji I,



Malahayati, MT
NIP. 198301272015032003

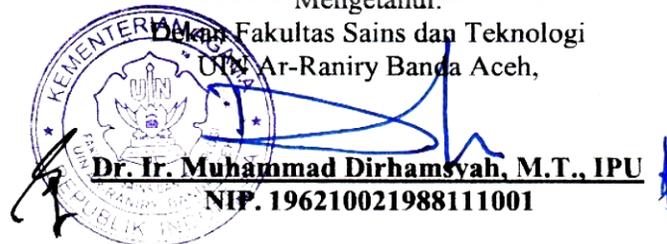
Penguji II,



Hendri Ahmadian, M.I.M
NIP. 198301042014031002

Mengetahui:

Dekan Fakultas Sains dan Teknologi
UIN Ar-Raniry Banda Aceh,



Dr. Ir. Muhammad Dirhamsyah, M.T., IPU
NIP. 196210021988111001

LEMBAR PERNYATAN KEASLIAN

Yang bertanda tangan di bawah ini :

Nama : Abdul Rafar
NIM : 190705076
Program studi : Teknologi Informasi
Fakultas : Sains dan Teknologi
Judul : Perancangan Sistem Absensi Pengenalan Wajah Mahasiswa Prodi Teknologi Informasi Berbasis *Website*

Dengan ini menyatakan bahwa dalam penulisan skripsi ini, saya:

1. Tidak menggunakan ide orang lain tanpa mampu mengembangkan dan mempertanggungjawabkan;
2. Tidak melakukan plagiasi terhadap naskah karya orang lain;
3. Tidak menggunakan karya orang lain tanpa menyebutkan sumber asli atau tanpa izin pemilik karya;
4. Tidak memanipulasi dan memalsukan data;
5. Mengerjakan sendiri karya ini dan mampu bertanggungjawab atas karya ini.

Bila dikemudian hari ada tuntutan dari pihak lain atas karya saya, dan telah melalui pembuktian yang dapat dipertanggungjawabkan dan ternyata memang ditemukan bukti bahwa saya telah melanggar pernyataan ini, maka saya siap dikenai sanksi berdasarkan aturan yang berlaku di Fakultas Sains dan Teknologi UIN Ar-Raniry Banda Aceh.

Demikian pernyataan ini saya buat dengan sesungguhnya dan tanpa paksaan dari pihak manapun.

Banda Aceh, 8 Mei 2024

Yang Menyatakan,



Abdul Rafar

ABSTRAK

Nama : Abdul Rafar
NIM : 190705076
Program studi : Teknologi Informasi
Judul : Perancangan Sistem Absensi Pengenalan Wajah Mahasiswa Prodi Teknologi Informasi Berbasis *Website*
Tanggal Sidang : 8 Mei 2024 M
Jumlah Halaman : 76 Halaman
Pembimbing I : Khairan AR, M.Kom.
Pembimbing II : Mulkan Fadhli, S.T., M.T
Kata Kunci : Teknologi Informasi, Sistem Sistem Absensi Wajah, Website.

Program Studi Teknologi Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Ar-Raniry, Banda Aceh, saat ini menghadapi beberapa permasalahan terkait absensi mahasiswa. Salah satunya adalah praktik "titip absen" yang sering terjadi, di mana mahasiswa yang tidak hadir akan meminta temannya untuk mencatatkan kehadiran mereka. Selain itu, absensi online juga rentan terhadap kecurangan karena mahasiswa dapat melakukan absensi dari mana saja, bahkan di luar kampus atau di luar ruang kuliah.

Untuk mengatasi masalah ini, peneliti berencana untuk mengembangkan Sistem Absensi wajah otomatis berbasis video menggunakan Webcam. Dengan sistem ini, mahasiswa harus hadir secara langsung di kampus untuk melakukan absensi karena kehadiran wajah tidak dapat diwakilkan oleh orang lain atau dilakukan dari rumah. Sistem ini akan menggunakan metode FaceMatcher untuk meningkatkan akurasi pengenalan wajah mahasiswa pada saat absensi..

Kata Kunci : Teknologi Informasi, Sistem Pengarsipan Surat Dengan Laravel, Website

KATA PENGANTAR

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِیْمِ

Puji Syukur saya panjatkan kehadiran Allah SWT dan junjungan Nabi Besar Muhammad saw, karena dengan rahmat, *ridha*, pengetahuan, dan penerangan alam, sehingga saya dapat menyusun Proposal tugas akhir. Shalawat dan salam semoga tercurahkan kepada junjungan Nabi Besar Muhammad SAW beserta keluarganya, para sahabatnya dan seluruh umatnya yang selalu istiqamah hingga akhir zaman.

Proposal tugas akhir ini menjelaskan tentang perancangan dan “sistem absensi pengenalan wajah otomatis mahasiswa berbasis video menggunakan webcam study kasus: prodi teknologi informasi”. Proposal ini merupakan salah satu persyaratan dalam menyelesaikan tugas akhir bagi mahasiswa/i Program Studi Teknologi Informasi, Fakultas Sains dan Teknologi, Universitas Islam Negeri Ar-Raniry.

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah membantu penulis dalam penyelesaian proposalnya, penulis juga mendapatkan banyak pengetahuan dan wawasan baru yang sangat berarti. Oleh karena itu, penulis tidak lupa mengucapkan terima kasih kepada :

1. Orang tua dan keluarga penulis, yang senantiasa mendoakan dan memberi support kepada penulis dalam menyelesaikan proposal tugas akhir ini.
2. Ibu Malahayati, MT Selaku Ketua Program Studi Teknologi Informasi.
3. Bapak Khairan Ar, M.kom selaku pembimbing laporan tugas akhir Prodi Teknologi Informasi UIN Ar-Raniry. Yang telah meluangkan waktunya untuk membimbing dan mengarahkan penulis dalam membuat proposal tugas akhir ini sehingga saya dapat menyelesaikan penulisan proposal tugas akhir.
4. Bapak Mulkan Fadhli, M.T selaku pembimbing kedua laporan tugas akhir Prodi Teknologi Informasi UIN Ar-Raniry. Yang telah meluangkan waktunya untuk membimbing dan mengarahkan penulis dalam membuat proposal tugas akhir ini sehingga saya dapat menyelesaikan penulisan proposal tugas akhir
5. Bapak Hendri Ahmadian, S.Si., M.I.M selaku dosen wali akademik saya. Banyak ilmu, arahan dan dukungan selama saya menulis laporan tugas akhir ini.

6. Semua pihak yang telah membantu penulis dalam menyelesaikan proposal tugas akhir.

Akhir kata semoga laporan ini dapat dinilai dengan sebaik-baiknya dan menjadi dokumen yang dapat memberikan masukan dan penilaian dalam sistem perguruan tinggi serta masukan bagi dunia pendidikan. Dan mohon maaf jika terdapat kesalahan dan kekurangan dalam laporan ini.

Banda Aceh, 11 Mei 2024
Penulis,

Abdul Rafar



DAFTAR ISI

LEMBAR PERSETUJUAN.....	i
LEMBAR PENGESAHAN	ii
LEMBAR PERNYATAN KEASLIAN.....	iii
ABSTRAK.....	iv
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	2
1.4 Batasan Masalah.....	3
1.5 Manfaat Penelitian.....	3
BAB II TINJAUAN PUSTAKA.....	4
2.1 Landasan Rancangan.....	4
2.2 Landasan Teori Dasar.....	4
2.2.1 Sistem.....	5
2.2.2 Informasi.....	5
2.2.3 Sistem Informasi.....	5
2.2.4 Komponen Sistem Informasi.....	6
2.3 Teori pengembangan sistem.....	6
2.3.1 <i>FaceMatcher</i>	6
2.3.2 <i>User Interface (UI)</i>	8
2.3.3 <i>Face Recognition</i>	8
2.3.4 <i>Dynamic Time Warping</i>	9
2.3.5 Tools Bahasa Pemrograman.....	9
2.3.6 Basis data (<i>Database</i>).....	10
2.3.7 UML (<i>Unified Modelling Language</i>).....	12

2.4 Penelitian Terdahulu.....	14
BAB III METODOLOGI PENELITIAN.....	16
3.1 Pendekatan Penelitian.....	16
3.1 Waktu dan Tempat penelitian.....	16
3.2 Alat dan Bahan	16
3.3 Tahapan penelitian	17
3.4 Tahapan Metode <i>FaceMatcher</i>	18
3.5 Populasi dan Sampel	20
3.6 Teknik Pengumpulan data	21
3.6.1 Observasi	21
3.6.2 Studi literatur	21
3.7 Analisis Kebutuhan	21
3.7.1 Kebutuhan Perangkat Lunak (<i>Software</i>).....	22
3.7.2 Kebutuhan Perangkat Keras (<i>Hardware</i>)	22
3.8 Perancangan Sistem.....	22
3.8.1 <i>Use Case Diagram</i>	23
3.8.2 <i>Activity Diagram</i>	24
3.8.3 <i>Sequence Diagram</i>	33
3.8.4 <i>Class Diagram</i>	39
3.8.5 <i>Entity Relationship Diagram</i>	40
3.8.6 <i>Database</i>	41
BAB IV HASIL DAN PEMBAHASAN	43
4.1 Implementasi Metode <i>FaceMatcher</i>	43
4.2 Implementasi Sistem dan Pembahasan.....	48
4.2.1 Tampilan Halaman Admin.....	48
4.2.2 Tampilan Halaman Mahasiswa.....	54
4.3 Evaluasi Pengujian Sistem	56
4.3.1 Hasil Pengujian Kompabilitas	57
4.3.2 Hasil Pengujian <i>Error Handling</i>	58
BAB V KESIMPULAN DAN SARAN.....	60
5.1 Kesimpulan.....	60

5.2 Saran.....	60
DAFTAR PUSTAKA	61
LAMPIRAN.....	65



DAFTAR GAMBAR

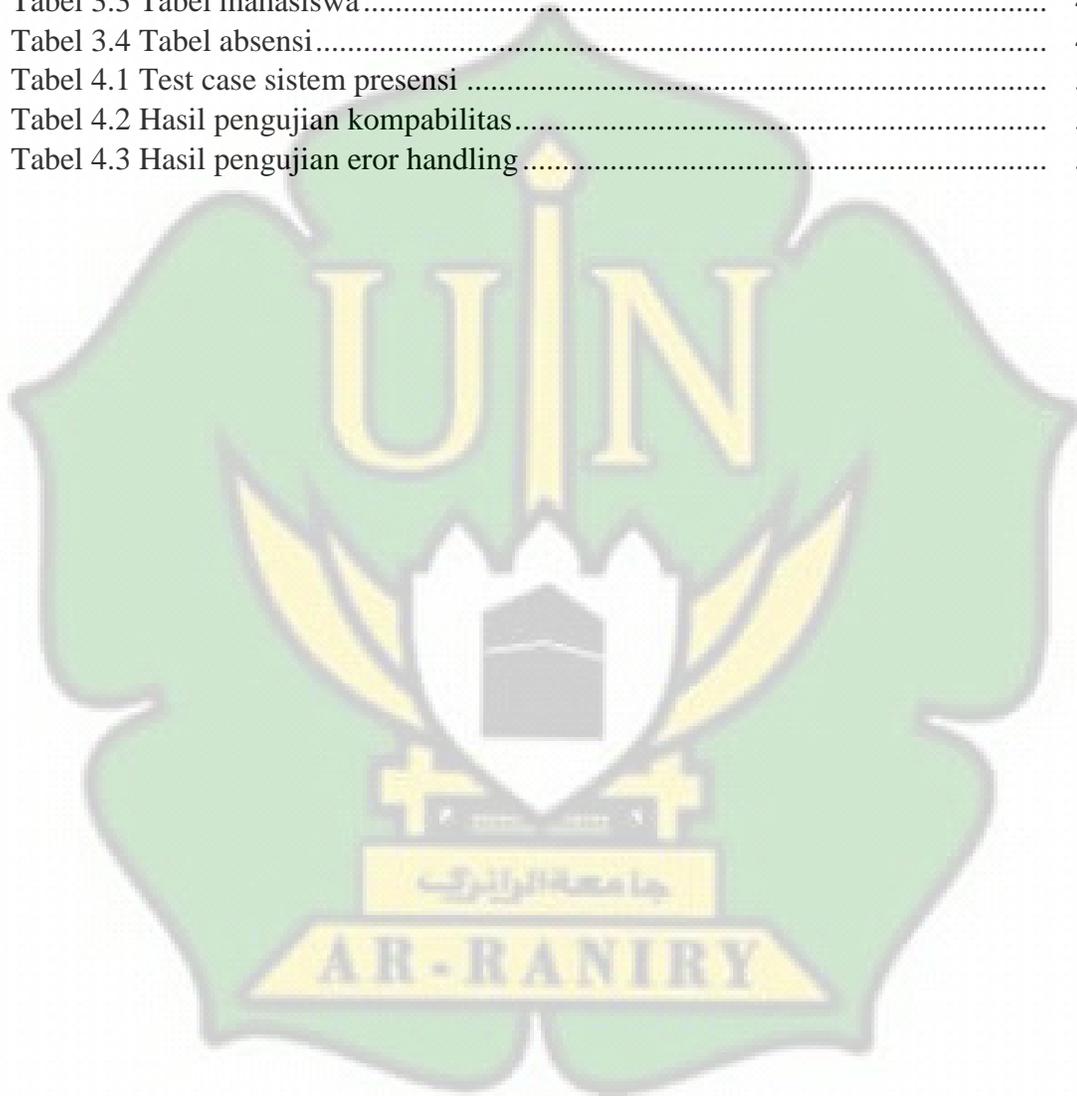
Gambar 3.1 Tahapan Penelitian	18
Gambar 3.2 Tahapan Identifikasi.....	18
Gambar 3.3 Tahapan Verifikasi	19
Gambar 3.4 Rumus Slovin	20
Gambar 3.5 Use Case Diagram.....	23
Gambar 3.6 Activity Diagram login.....	24
Gambar 3.7 Activity Diagram Data Mahasiswa	25
Gambar 3.8 Activity Diagram Tambah data	26
Gambar 3.9 Activity Diagram Ubah data	27
Gambar 3.10 Activity Diagram Hapus data.....	28
Gambar 3.11 Activity Diagram Cari data	29
Gambar 3.12 Activity Diagram Data absen hari in.....	30
Gambar 3.13 Activity Diagram Data semua absensi	31
Gambar 3.14 Activity Diagram Absen masuk	32
Gambar 3.15 Activity Diagram Absen pulang.....	33
Gambar 3.16 Sequence Diagram login	34
Gambar 3.17 Sequence Diagram Data mahasiswa	35
Gambar 3.18 Sequence Diagram Tambah data.....	35
Gambar 3.19 Sequence Diagram Tambah data.....	36
Gambar 3.20 Sequence Diagram Tambah data.....	36
Gambar 3.21 Sequence Diagram Tambah data.....	37
Gambar 3.22 Sequence Diagram Data absen hari ini	37
Gambar 3. 23 Sequence Diagram Data semua absensi	38
Gambar 3.24 Sequence Diagram Absen masuk.....	38
Gambar 3.25 Sequence Diagram Absen masuk.....	39
Gambar 3.26 Class diagram	40
Gambar 3.27 Entity Relationship.....	41
Gambar 4.1 Face Detection.....	44
Gambar 4.2 Feature Extraction 1	44
Gambar 4.3 Feature Extraction 2	45
Gambar 4.4 Descriptor Comparison 1	45
Gambar 4.5 Descriptor Comparison 2	46
Gambar 4.6 Descriptor Comparison 3	46
Gambar 4.7 Matching Face 1	47
Gambar 4.8 Matching Face 2	48
Gambar 4.9 Halaman login	49
Gambar 4.10 Halaman home.....	49
Gambar 4.11 Halaman data mahasiswa	50
Gambar 4.12 Tambah mahasiswa	51

Gambar 4.13 Tambah data wajah.....	51
Gambar 4.14 Halaman absen hari ini.....	52
Gambar 4.15 Tambah absensi.....	53
Gambar 4.16 Data semua absensi.....	53
Gambar 4.17 Halaman absensi mahasiswa.....	54
Gambar 4.18 Absen masuk.....	55
Gambar 4.19 Absen keluar.....	55



DAFTAR TABEL

Tabel 2.1 Notasi Entity-Relation Diagram	12
Tabel 2.2 Notasi Unifed Modelling Language.....	14
Tabel 3.1 Alat dan Bahan.....	17
Tabel 3.2 Tabel user.....	41
Tabel 3.3 Tabel mahasiswa.....	42
Tabel 3.4 Tabel absensi.....	42
Tabel 4.1 Test case sistem presensi	57
Tabel 4.2 Hasil pengujian kompabilitas.....	58
Tabel 4.3 Hasil pengujian eror handling.....	59



BAB I

PENDAHULUAN

1.1 Latar Belakang

Absensi merupakan suatu proses pengambilan data untuk mengetahui jumlah ketidakhadiran dalam suatu kegiatan. Absensi digunakan dalam banyak hal namun, dalam hal ini adalah absensi mahasiswa. Absensi kehadiran merupakan bagian peranan penting karena salah satu penunjang utama dalam mendukung dan memotivasi kegiatan yang dilakukan di dalamnya.

Berdasarkan pengamatan dan observasi yang dilakukan penulis di setiap ruang belajar di Uin Ar-ranniry yang lebih spesifik di Fakultas Sains dan Teknologi, saat ini presensi yang dilakukan mahasiswa masih dengan cara konvensional yaitu dengan menandatangani buku maupun kertas absen setiap harinya atau ada juga yang menggunakan absensi *online* melalui *website*. Kedua metode absen tersebut memang sudah biasa dilakukan oleh para mahasiswa dan mudah untuk dilakukan. Namun, dalam praktiknya, sistem absensi mahasiswa tersebut tidak begitu efisien dan efektif. Masih banyak terdapat kekurangan yang ditemui. Seperti pada absensi dengan metode konvensional munculnya fenomena "titip absen" di mana jika seorang mahasiswa absen karena alasan yang tidak dapat ditoleransi, maka dia akan meminta temannya untuk menandatangani nama mahasiswa yang absen tersebut. Praktik tersebut sudah menjadi kebiasaan yang umum terjadi di kalangan mahasiswa. Atau jika melakukan absensi secara *online* melalui *website* kecurangan yang dapat ditemui adalah mahasiswa dapat melakukan absen dimana saja karena *website* siakad tidak mendukung fitur gps untuk melihat lokasi mahasiswa sedang berada dalam kawasan kampus atau tidak.

Berdasarkan permasalahan tersebut, maka dari itu peneliti ingin setidaknya dapat meminimalisir kegiatan kecurangan yang dapat dilakukan oleh para mahasiswa dengan cara membuat Sistem Absensi pengenalan wajah mahasiswa menggunakan *webcam*, yang mana nantinya mahasiswa kalau ingin melakukan absensi

kehadiran harus datang ke kampus untuk melakukannya karena absensi dengan wajah ini tidak bisa diwakilkan oleh temannya dan juga tidak bisa dilakukan dirumah.

Metode *FaceMatcher* merupakan arsitektur yang efisien sering digunakan dalam aplikasi dunia nyata. Arsitektur ini menggunakan konvolusi terpisah yang lebih dalam daripada konvolusi standar yang digunakan dalam arsitektur sebelumnya untuk membangun model yang lebih ringan. Selain itu, *FaceMatcher* memperkenalkan dua *hyperparameter* global baru yaitu *pengganda lebar* dan *pengganda resolusi* yang memungkinkan pengembang model untuk mengatur latensi atau akurasi demi kecepatan dan ukuran yang lebih rendah sesuai dengan kebutuhan mereka (Singh, 2024).

Dengan menggunakan metode *FaceMatcher* diharapkan dapat meningkatkan akurasi data wajah dari mahasiswa pada absensi pengenalan wajah ini. Sistem absensi wajah ini nantinya akan dibangun dengan beberapa fitur seperti: pengumpulan data wajah secara elektronik, klasifikasi absensi berdasarkan waktu dan tanggal serta melakukan absensi harian seperti absen masuk dan absen pulang yang nantinya akan otomatis disimpan ke dalam *database*.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang terjadi, peneliti ingin mahasiswa agar lebih disiplin dalam hal terutama kehadiran. Maka dapat dirumuskan dari permasalahan tersebut, yaitu: Bagaimana cara membangun sebuah sistem absensi berbasis pengenalan wajah menggunakan metode *facematcher* agar dapat mengurangi tingkat kecurangan?

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang sistem absensi wajah untuk mahasiswa Prodi Teknologi Informasi berbasis *website*.
2. Menguji fungsi dari sistem absensi wajah untuk mengetahui keberhasilan dalam menjalankannya.

3. Menganalisis pengujian data wajah untuk mengetahui tingkat akurasi keberhasilan mendeteksi wajah pada sistem yang telah dibangun.

1.4 Batasan Masalah

Agar pembahasan sesuai dengan tema judul yang telah ditentukan dan latar belakang yang telah jelaskan, maka penulis membatasi masalah hanya dengan membangun aplikasi sistem Absensi wajah untuk mahasiswa Prodi Teknologi informasi dengan jadwal yang sama setiap minggunya, dengan menggunakan metode *FaceMatcher* dalam bahasa pemograman *Php*. Bantuan *software* yang akan digunakan ialah *Visual Studio Code*.

1.5 Manfaat Penelitian

Manfaat penelitian ini memiliki beberapa manfaat berdasarkan pada latar belakang, rumusan masalah, dan tujuan penelitian yang telah dijelaskan.

1. Bagi penulis, diharapkan penelitian ini dapat menambah wawasan serta pengalaman penulis dalam membangun sebuah sistem.
2. Bagi instansi, diharapkan hasil dari penelitian ini mudah mudahan dapat membantu dalam menagatasi kecurangan absensi.

BAB II

TINJAUAN PUSTAKA

2.1 Landasan Rancangan

Proses penentuan dan proses data yang diperlukan oleh sistem baru disebut perancangan. Keuntungan dari tahap perancangan sistem adalah untuk memberikan gambaran rancangan bangun yang lengkap sebagai pedoman. Untuk *programmer* yang mengembangkan aplikasi, tahap ini mencakup desain *hardware* atau *software*, *database*, dan aplikasi, sesuai dengan bagian sistem yang dikomputerisasikan (Hanadhito Riswantoro, 2019).

Menurut (Donald A. Norman) Seorang pakar teknologi dan psikologi mengungkapkan bahwa perancangan merupakan suatu proses yang mempertimbangkan kebutuhan, keterampilan, tujuan, serta tuntutan lingkungan dalam menciptakan produk atau sistem yang optimal untuk digunakan.

Menurut (Johan Van Der Zwart) Seorang pakar di bidang desain grafis menjelaskan bahwa perancangan merupakan suatu proses yang melibatkan pemikiran kreatif, analisis, dan pengembangan ide guna menciptakan produk atau sistem yang efisien dan efektif.

Berdasarkan dari beberapa pendapat diatas perancangan merupakan tahap berikutnya setelah analisis sistem. Tujuan dari tahap ini adalah membuat rancangan yang memenuhi kebutuhan yang ditentukan selama tahap analisis.

2.2 Landasan Teori Dasar

Adapun landasan teori dasar yang mendukung dalam penelitian ini adalah sebagai berikut :

2.2.1 Sistem

Sistem dapat didefinisikan sebagai sekelompok subsistem, komponen, atau elemen yang bekerja sama untuk mencapai tujuan yang sama dan menghasilkan *output* yang sudah ditentukan sebelumnya. Selain itu, sistem dapat didefinisikan sebagai sekelompok komponen yang saling berhubungan, bekerja sama untuk mencapai tujuan dengan menerima *input* dan menghasilkan *output* dalam proses transformasi yang teratur (Ari Nur Rokhman, 2020).

Secara terminologi, sistem digunakan dalam berbagai cara yang beragam sehingga sulit untuk memberikan definisi yang mencakup semua penggunaannya dan cukup singkat untuk memahami maksudnya. Hal ini disebabkan oleh variasi pemahaman tentang sistem yang bergantung pada sudut pandang individu yang mencoba mendefinisikannya. Misalnya, menurut hukum, sistem dianggap sebagai serangkaian aturan yang membatasi, baik dalam kapasitas sistem itu sendiri maupun lingkungan di mana sistem tersebut beroperasi untuk menjamin keadilan dan kesesuaian (Ridho, 2018).

2.2.2 Informasi

Dalam sebuah organisasi atau perusahaan, informasi memiliki peran yang krusial dalam mendukung proses pengambilan keputusan oleh manajemen. Secara garis besar, informasi adalah data yang telah diolah menjadi bentuk yang lebih bermakna bagi penerimanya. (fauzi 2017, 2018).

Secara umum, informasi dapat diartikan sebagai hasil dari proses pengolahan data yang kemudian disajikan dalam bentuk yang lebih bermanfaat dan memiliki makna yang lebih penting bagi penerimanya. Informasi ini menggambarkan kejadian-kejadian nyata yang digunakan sebagai dasar dalam pengambilan keputusan.

2.2.3 Sistem Informasi

Sistem informasi organisasi dapat menyediakan informasi kepada semua tingkatan organisasi kapan saja diperlukan. Sistematika ini mengumpulkan, mengambil, mengubah, mengolah, dan menyampaikan informasi enam dengan

menggunakan peralatan sistem informasi atau sistem informasi lainnya. orang. Dapat juga dijelaskan bahwa digital imaging adalah pemrosesan dan terjemahan digital dari gambar dengan menggunakan komputer. Bayangan itu sendiri merupakan fungsi dari intensitas cahaya yang dipantulkan pada bidang dua dimensi.

2.2.4 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen berikut berdasarkan komponen fisik penyusunnya (Sutanta, 2011):

1. Perangkat keras

Perangkat keras yang termasuk dalam sistem informasi adalah perangkat yang digunakan oleh sistem komputer untuk menerima dan menerima input dan output.

2. Perangkat lunak

Dalam sistem informasi, perangkat lunak terdiri dari program—program komputer yang terdiri dari sistem informasi (*operating system* atau *OS*), bahasa pemrograman (*programming language*), dan program aplikasi.

3. Prosedur

Prosedur mencakup prosedur pengoperasian sistem informasi, manual, dan dokumen yang berisi aturan-aturan yang terkait dengan sistem informasi.

4. Manusia

Ada beberapa peran yang dimainkan oleh manusia dalam suatu sistem informasi. Mereka termasuk operator, programmer, sistem analis, manajer, teknisi, administrator, basis data, dan individu lain yang terlibat dalam sistem tersebut.

2.3 Teori pengembangan sistem

Adapun beberapa teori pendukung dalam pengembangan sistem penelitian ini adalah sebagai berikut :

2.3.1 FaceMatcher

Metode *FaceMatcher* ini menggunakan *FaceAPI.js* untuk melakukan pengenalan wajah. *FaceAPI.js* adalah sebuah pustaka JavaScript yang menyediakan alat-alat untuk mendeteksi wajah, mengenali wajah, dan mengekstraksi landmark

wajah. Pustaka ini dibangun di atas *TensorFlow.js*, sehingga memudahkan pengguna dalam bekerja dengan model pembelajaran mendalam untuk tugas-tugas yang berkaitan dengan wajah. Dengan menggunakan *FaceAPI.js*, pengguna dapat melakukan berbagai operasi seperti mendeteksi wajah dalam gambar, mengekstraksi landmark wajah, dan mengenali wajah dengan membandingkan deskriptornya (Suthar, 2023).

FaceMatcher di *FaceAPI.js* adalah alat tingkat tinggi yang mempermudah proses pengenalan wajah dalam gambar. Alat ini memungkinkan Anda untuk membuat model pengenalan wajah dengan kumpulan deskriptor yang diberi label, dan kemudian menggunakan model ini untuk mencari kecocokan terbaik untuk deskriptor wajah yang diberikan.

Cara Kerja *FaceMatcher* pada dasar Pencocokan wajah melibatkan beberapa langkah utama (Collins, 2023), seperti:

1. *Face Detection*, dilakukan dengan menggunakan algoritme visi komputer yang terlatih untuk mengenali fitur wajah.
2. *Feature Extraction*, dilakukan setelah wajah terdeteksi langkah ini bertujuan untuk mengekstrak berbagai fitur dari gambar wajah, seperti posisi mata, lebar hidung, panjang hidung, bentuk mulut, dan kontur wajah.
3. *Face Comparison*, fitur-fitur dari gambar akan dibandingkan untuk menentukan kemiripan dengan menghitung skor kemiripan antara kedua set fitur. Semakin tinggi skor kemiripannya, semakin besar kemungkinan kedua wajah tersebut milik orang yang sama.
4. *Quality Assessment*, penilaian kualitas dilakukan untuk menentukan kualitas keseluruhan gambar wajah dan kemungkinan informasi yang akurat dan dapat diandalkan tentang wajah individu yang terkandung dalam gambar tersebut.

Dengan menggunakan metode *FaceMatcher* ini memiliki manfaat yang berbeda dibandingkan dengan metode lain seperti contohnya:

1. Alur Kerja yang Lebih Sederhana: *FaceMatcher* menyederhanakan alur kerja dengan menghilangkan sebagian besar kompleksitas yang terkait, sehingga memudahkan pengenalan wajah dalam gambar.
2. Efisiensi yang Tinggi: Metode ini mampu mencocokkan deskriptor wajah dengan koleksi deskriptor yang sudah diberi label secara efisien, membantu Anda menemukan kecocokan terbaik dengan cepat.
3. Fleksibilitas yang Luas: Anda dapat menggunakan deskriptor yang sudah diberi label dari gambar-gambar yang berbeda, sehingga metode ini cocok untuk berbagai skenario pengenalan wajah.

2.3.2 User Interface (UI)

User interface ialah ilmu yang mempelajari tentang tata letak grafis suatu tampilan dari *interface* sebuah sistem. Pengelompokan yang termasuk UI yaitu seperti tombol, gambar, teks, dan semua item yang berinteraksi dengan penggunanya. Adanya UI dalam pengembangan produk digital dapat lebih memudahkan para *developer* dalam merancang produknya sebelum dilakukan pengkodean dan di publikasikan ke user (Ariawan et al., 2020).

2.3.3 Face Recognition

Face recognition ialah sebuah teknologi untuk pengenalan wajah yang mana bisa mendeteksi wajah seseorang dengan memindai wajahnya secara gambar maupun video *realtime*. Cara kerja dari *face recognition* ini sendiri adalah sebagai berikut :

a. Pendeteksi wajah

Tahapan pertama yang dilakukan teknologi *face recognition* adalah mendeteksi wajah seseorang berupa gambar, video, atau secara *real-time*.

b. Pengukuran

Tahapan ini teknologi *face recognition* akan mengukur lekukan pada wajah dan akan dibuatkan *template* -nya.

c. Analisis wajah

Teknologi ini akan mengenali wajah seseorang secara spesifik, mulai dari jarak antar kedua mata, ukuran hidung, bentuk dagu dan lain-lain

d. Verifikasi dan identifikasi

Tahapan terakhir pada teknologi ini, algoritma akan berusaha mengonfirmasi identitas dari seseorang yang sudah terdeteksi wajahnya. Tahap inilah yang digunakan untuk dilakukan verifikasi keamanan yang akan digunakan. *Recognition* memanfaatkan basis data yang sudah terekam sebelumnya, demi memastikan bahwa pengguna perangkat tersebut adalah penggunanya.

2.3.4 *Dynamic Time Warping*

Dynamic time warping adalah suatu metode yang digunakan untuk menghitung jalur lintasan optimal antara dua sekuens data dengan tujuan untuk mendapatkan *warping path* dan jarak antara kedua sekuens tersebut sebagai hasil keluaran. Algoritma digunakan untuk menghitung jarak antara dua deretan dan juga mencari nilai pembengkokan jalur kedua. Algoritma ini dimulai dengan menghitung jarak lokal antara elemen dari dua rangkaian menggunakan berbagai metode perhitungan jarak yang berbeda. Suatu metode yang umum digunakan untuk menghitung jarak adalah dengan mengukur jarak mutlak antara nilai dari dua elemen.

2.3.5 Tools Bahasa Pemrograman

Berikut merupakan *tools* bahasa pemrograman yang penulis gunakan pada penelitian ini, sebagai berikut:

a. *Hypertext Preprocessor* (Php)

Php merupakan bahasa pemrograman yang bersifat *open source*, sehingga *user* bebas memodifikasi dan mengembangkan sesuai dengan kebutuhannya. Php sendiri menjalankan instruksi pemrograman selama proses *runtime*, hasil dari instruksi ini pasti akan berbeda tergantung pada data yang diproses.

b. *Face-API.js*

Face-API.js ialah sebuah pustaka JavaScript yang diciptakan oleh Vincent Muhler, yang memiliki fungsi untuk mendeteksi wajah melalui browser. Mengacu pada (Wayan Wardani & Yoka Fathoni, n.d.) *Face-API.js* dibangun dengan menggunakan API inti *Tensorflow.js* dan menggunakan beberapa *Convolutional*

Neural Networks (CNN) untuk mendukung tugas seperti mendeteksi wajah, mengidentifikasi wajah, ekspresi wajah, usia, dan jenis kelamin.

c. *Visual Studio Code* (VSC)

Visual Studio Code merupakan salah satu editor kode yang biasa digunakan oleh para *programmer*. VSC menjadi salah satu *code editor* dengan fitur yang sangat lengkap dan menarik untuk digunakan.

d. *Hypertext Markup Language* (HTML)

Web browser seperti *google chrome* dan *browser* lainnya akan membaca dokumen HTML yang merupakan bahasa pengformatan teks yang berisi tag-tag yang digunakan untuk membuat kerangka web *pages* pada sebuah jaringan komputer yang dikenal sebagai *World Wide Web* (www) atau biasa disebut dengan *web* saja.

e. *JavaScript*

JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman web lebih hidup dan interaktif. Jika sebelumnya Anda hanya mengenal HTML dan CSS, sekarang Anda tahu JavaScript dapat meningkatkan kinerja halaman web. Kamu juga dapat menggunakan JavaScript ini untuk membuat aplikasi, tools, atau bahkan *game* web.

2.3.6 Basis data (*Database*)

a. *ERD* (*Entity-Relation Diagram*)

ERD atau diagram entitas hubungan adalah metode yang digunakan untuk menggambarkan data atau objek yang dibuat berdasarkan entitas dan hubungan antar entitas tersebut dengan menggunakan beberapa simbol. Diagram nota grafis yang digunakan dalam proses pembuatan *database* bertujuan untuk menjelaskan struktur *database* dengan lebih mudah dipahami. ERD memiliki peran penting sebagai alat bantu dalam pembuatan *database* dan memberikan gambaran mengenai cara kerja *database* yang dibuat (khoulah afifah dkk, 2022).

ERD dibagi dalam 3 elemen, yaitu sebagai berikut:

1) Entitas

Entitas merupakan obyek yang tersedia di dalam basis data, entitas tersebut dapat berupa individu, objek, atau keadaan terkait dengan suatu informasi.

2) Atribut

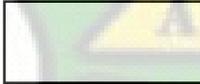
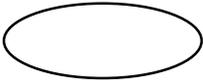
Atribut digunakan untuk menerangkan data yang terdapat dalam suatu objek atau substansi. Biasanya, atribut ditempatkan di dalam tabel entitas atau dapat dipisahkan dari tabel tersebut.

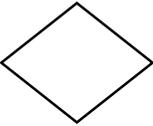
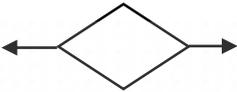
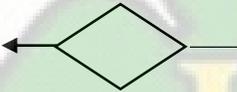
3) Relasi

Penggunaan relasi untuk dapat menghubungkan antara entitas entitas dua atau lebih.

Ada beberapa macam relasi yang dimiliki oleh ERD, yaitu:

- *One to One* : hubungan di antara dua tabel data dimana setiap baris data pada tabel pertama hanya memiliki kaitan dengan satu baris data pada tabel kedua.
- *One to Many* : hubungan di mana setiap baris data dalam tabel pertama terhubung dengan lebih dari satu baris data dalam tabel kedua..
- *Many to Many* : hubungan di mana setiap baris data pada tabel pertama memiliki keterhubungan dengan lebih dari satu baris pada tabel kedua.

Notasi	Nama	Keterangan
	Entitas	Mewakili suatu objek yang berwujud nyata dan dapat dibedakan dengan objek lainnya.
	Atribut / field	Kepemilikan dari suatu entitas adalah fitur yang digunakan untuk menggambarkan sifat-sifat dari entitas tersebut.

	Relasi	Menyatakan hubungan antara beberapa entitas yang memiliki perbedaan.
	Relasi 1 : 1	Hubungan yang ada menunjukkan bahwa setiap entitas dalam himpunan pertama hanya dapat berhubungan dengan satu entitas dalam himpunan kedua, dan sebaliknya.
	Relasi 1 : N	Hubungan yang diberikan menunjukkan koneksi setiap entitas himpunan pertama dapat berelasi banyak entitas pada himpunan entitas kedua, tetapi tidak sebaliknya.
	Relasi N : N	Hubungan yang diberikan menunjukkan bahwa terdapat hubungan antara banyak entitas pada himpunan pertama dengan entitas pada himpunan kedua.

Tabel 2.1 Notasi *Entity-Relation Diagram*

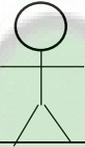
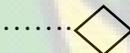
2.3.7 UML (*Unified Modelling Language*)

UML (*Unified Modeling Language*) ialah bahasa yang standar pada perancangan dan pendokumentasian dari sistem. Bahasa pemodelan visual UML digunakan untuk berkomunikasi dan menggunakan diagram untuk memodelkan sistem sekaligus teks-teks pendukung. UML digunakan sebagai pemodelan (Jantce TJ Sitinjak et al., 2020).

Adapun diagram UML yang digunakan adalah sebagai berikut:

1. *Activity* diagram untuk menggambarkan proses urutan aktifitas dalam sebuah sistem yang akan dibuat.
2. *Use case* diagram menggambarkan interaksi antara sistem dengan user (pengguna). Juga merupakan tahap awal dalam proses perancangan sistem.

3. *Class* diagram menggambarkan sekumpulan kelas dalam sebuah sistem, interface dan kolaborasi terhubungnya antara satu dengan yang lain antara masing-masing bagian.
4. *Sequence* diagram ialah interaksi proses yang terjadi didalam perangkat lunak yang diatur dalam urutannya.

Simbol	Nama	Keterangan
	<i>Actor</i>	Menunjukkan peran orang, alat ketika berhubungan dengan <i>use case</i>
	<i>Use case</i>	Akan digunakan untuk menggambarkan penghubung antara <i>actor</i> dan <i>use case</i>
	<i>Generalization</i>	Untuk ikut serta dalam skenario penggunaan, <i>actor</i> harus memperjelas spesialisasinya
	<i>Interface</i>	Sebuah <i>class</i> yang tidak memiliki implementasi namun berisi kumpulan operasi
	<i>Relization</i>	Hubungan elemen yang bertanda panah akan merelasikan ke elemen yang ada bagian tanda panah
	<i>Interaction</i>	Digunakan untuk alir pesan antar objek
	<i>Association</i>	penghubung antar <i>actor</i> dengan <i>use case</i>

	<i>Dependency</i>	Hubungan yang menunjukkan perubahan pada satu elemen yang mempengaruhi elemen yang lain.
	<i>Note</i>	Tampilan fisik saat program berjalan menunjukkan representasi sumber daya komputasi.
	<i>Class</i>	Merupakan objek yang memiliki sifat dan Tindakan.
	<i>Package</i>	Sebuah tempat yang berguna saat mengelompokkan elemen dari suatu sistem yang bangun.

Tabel 2.2 Notasi *Unified Modelling Language*

2.4 Penelitian Terdahulu

Penelitian terdahulu kumpulan sumber-sumber penelitian yang digunakan dan diakses sebagai dukungan dan referensi oleh peneliti untuk menjalankan studi mereka dan mencegah hasil yang serupa dengan penelitian sebelumnya. Di samping itu, juga dapat digunakan sebagai sumber ide untuk penelitian lain yang bertujuan menghasilkan temuan baru atau memperbarui dengan menggunakan metode dan pendekatan ilmiah yang berbeda.

Pengembangan yang dilakukan oleh (Jannah, 2023) dalam penelitiannya membangun Web Presensi Karyawan Industri Ketas Jaya menggunakan Metode *Convolutional Neural Network* (CNN) telah dibangun untuk mengurangi kemungkinan penularan virus antar karyawan. Teknologi *touchless*, yang mengidentifikasi absensi melalui pengenalan wajah, dapat meningkatkan akurasi dan efisiensi proses absensi. Selain itu, mereka memiliki kemampuan untuk meningkatkan produktivitas karyawan di sektor kertas Jaya melalui peningkatan keamanan aset data dan pemeliharaan aset sumber daya manusia yang berkualitas

Pengembangan yang dilakukan oleh (Wayan Wardani & Yoka Fathoni, n.d.) dalam penelitiannya yang berjudul “Perancangan Absensi Berbasis *Face Recognition* Pada Desa Sokaraja Lor Menggunakan *Platform* Android” hasil penelitian ini telah berhasil membangun sistem absensi yang dapat digunakan oleh para pegawai dan pejabat desa untuk lebih mudah mengelola dan melihat siapa saja yang datang tepat waktu dan terlambat.

Pengembangan yang dilakukan oleh (Christyanto et al., 2022) dalam penelitiannya yang berjudul “Pengembangan Aplikasi Android Presensi Kehadiran Realtime menggunakan Pengenalan Wajah dengan Model *Facenet*” berdasarkan penelitian yang dilakukan, aplikasi Android yang dikembangkan yaitu dengan menggunakan Bahasa Kotlin. Implementasi *Website* menggunakan *platform* Bubble untuk membuat tampilan *interface* dan sambungan ke *database online*..

Pengembangan yang dilakukan oleh (Hartiwi et al., 2020) dalam penelitiannya yang berjudul “Sistem Manajemen Absensi dengan Fitur Pengenalan Wajah dan GPS Menggunakan YOLO pada Platform Android” hasil penelitiannya didapat akurasi sebesar 0.93435 dan tersendah masih dalam range 93%, sedangkan nilai –rata-rata akurasi adalah 93.26%. Dari 20 data penilaian yang dilakukan sistem Sistem Manajemen Absensi Dengan Fitur Pengenalan Wajah dan GPS menggunakan YOLO pada Platform Android. Penelitian ini membuktikan pemanfaatan tool library dengan algoritma You Only Look Once (YOLO) merupakan metode terpopuler dalam dunia pengenalan wajah dan terbukti tangguh dan sangat baik untuk saat ini.

Pengembangan yang dilakukan oleh (Amari, 2023) dalam pengujiannya Metode *euclidean distance* dapat digunakan dengan baik untuk klasifikasi wajah dalam sistem *login* dan absensi. Perangkat android memengaruhi kinerja sistem. Metode *euclidean distance* dalam sistem pengenalan wajah menggunakan perangkat Android versi 12, RAM 4, dan penyimpanan 64GB menunjukkan akurasi 100% dan kecepatan 2,5 hingga 3 detik, sementara perangkat Android versi 11, RAM 7, dan penyimpanan 128GB menunjukkan akurasi 100% dan kecepatan 2 hingga 2,5 detik. Metode ini juga dapat melakukan klasifikasi pengenalan wajah siswa yang menggunakan kacamata dengan akurasi 97% dan kecepatan 2 hingga 3 detik.

BAB III

METODOLOGI PENELITIAN

3.1 Pendekatan Penelitian

Pendekatan penelitian yang digunakan oleh peneliti adalah penelitian kuantitatif untuk menentukan hasilnya. Penelitian kuantitatif adalah penelitian yang bermula dari teori dan berkembang menjadi cara mengumpulkan data numerik di lapangan.

Menurut (Sugiyono, 2019) pendekatan kuantitatif juga disebut sebagai pendekatan positivistic, yang digunakan untuk menginvestigasi populasi atau sampel tertentu. Pengambilan sampel umumnya dilakukan secara acak, pengumpulan data menggunakan instrumen penelitian, analisis data bersifat kuantitatif/statistik dengan tujuan menguji hipotesis yang telah ditetapkan. Penelitian kuantitatif ini digunakan untuk mendapatkan informasi tentang hasil pengukuran instrumen yang telah divalidasi.

3.1 Waktu dan Tempat penelitian

Waktu penelitian ini, yaitu penelitian tentang bagaimana membangun sebuah sistem absensi otomatis mahasiswa berbasis video dimulai pada bulan Agustus tahun 2023 sampai dengan sekarang.

Lokasi penelitian dilakukan dalam lingkungan kampus Program Studi Teknologi Informaasi, Universitas Islam Negeri Ar-ranniry, Banda Aceh.

3.2 Alat dan Bahan

Pada penelitian ini, Keseluruhan alat dan bahan adalah sebagai berikut :

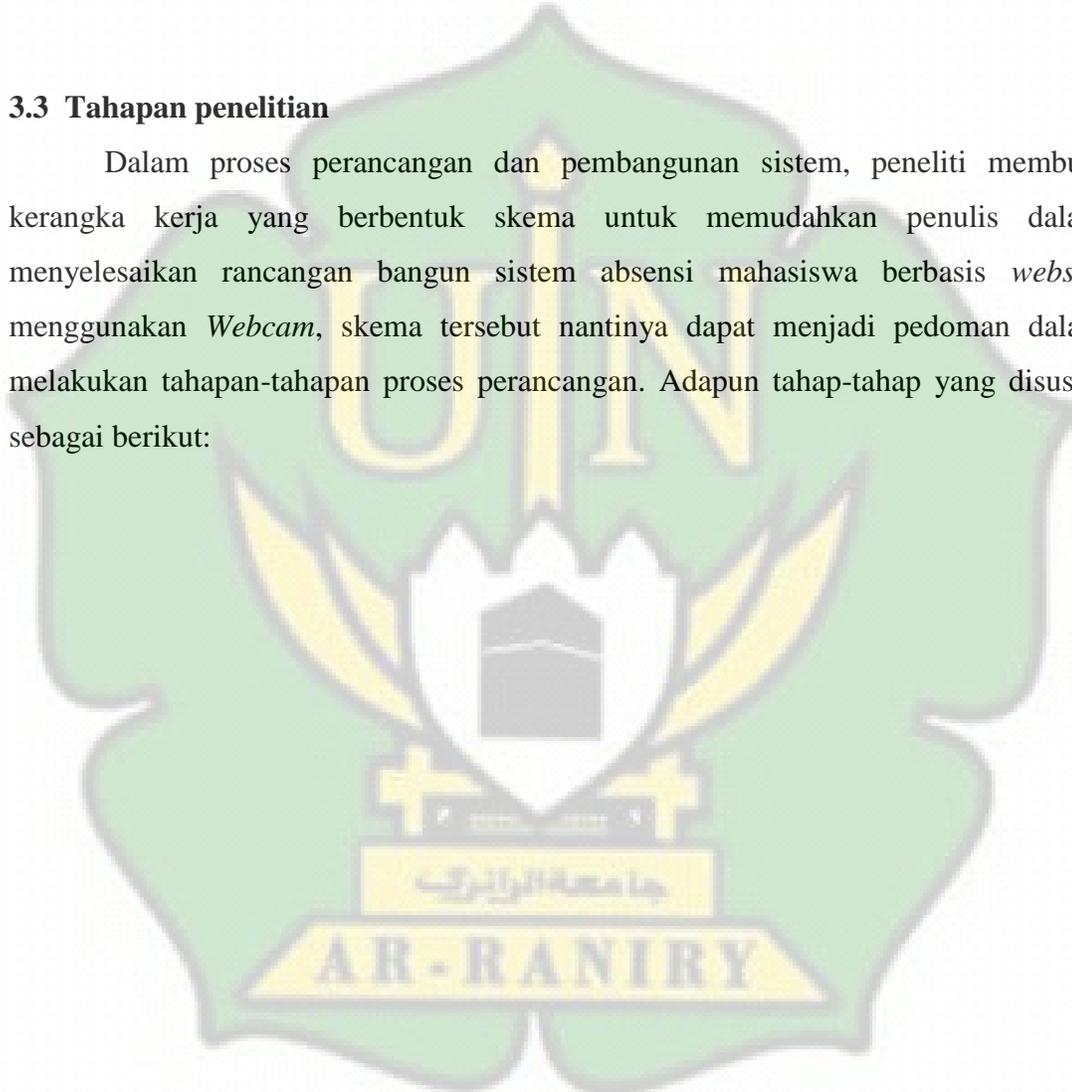
No	Alat dan Bahan	Nama	Jumlah
1	Perangkat keras (<i>Hardware</i>)	- Komputer/ Laptop	1
		- <i>Webcam</i>	1
2	Perangkat Lunak (<i>Software</i>)	- <i>Php</i>	1

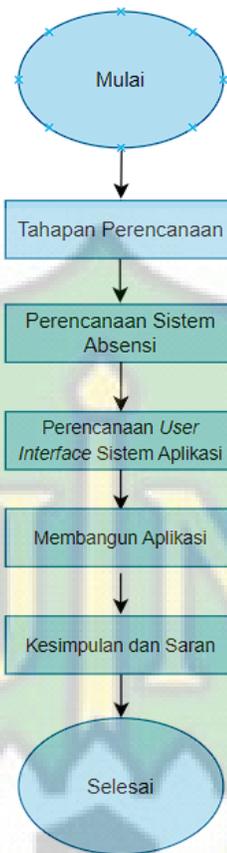
		- <i>Visual Studio Code</i>	1
		- <i>Face-API.js</i>	1
		- <i>Sistem Operasi Windows 10 or above</i>	1

Tabel 3.1 Alat dan Bahan

3.3 Tahapan penelitian

Dalam proses perancangan dan pembangunan sistem, peneliti membuat kerangka kerja yang berbentuk skema untuk memudahkan penulis dalam menyelesaikan rancangan bangun sistem absensi mahasiswa berbasis *website* menggunakan *Webcam*, skema tersebut nantinya dapat menjadi pedoman dalam melakukan tahapan-tahapan proses perancangan. Adapun tahap-tahap yang disusun sebagai berikut:





Gambar 3.1 Tahapan Penelitian

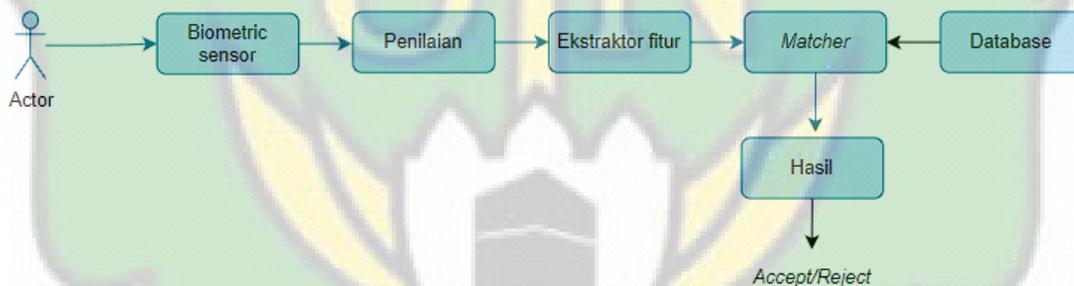
3.4 Tahapan Metode *FaceMatcher*

Dalam proses perancangan dan pembangunan sistem, peneliti menciptakan skema alur kerja untuk metode *FaceMatcher* agar memudahkan pemahaman terhadap proses metode *Facematcher* itu sendiri. Berikut adalah tahapan-tahapannya:



Gambar 3.2 Tahapan Identifikasi

Seperti yang terlihat pada gambar diatas mahasiswa melakukan pendaftaran untuk pertama kali agar wajah dapat di diketahui oleh sistem. Sistem biometrik mengidentifikasi individu berdasarkan fitur atau karakteristik yang dimiliki oleh individu. Prosesnya dimulai dengan menangkap fitur, seperti mengambil gambar wajah dan iris mata untuk pengenalan wajah. Data yang terkumpul kemudian diolah menggunakan fungsi matematika untuk membuat *template* biometrik (Edlis, 2009). Kemudian dilanjutkan oleh sistem untuk menilai wajah tersebut agar dapat dilakukannya ekstraksi deskriptor pada wajah mahasiswa yang telah diambil. Selanjutnya sistem akan menyimpan semua data wajah mahasiswa ke dalam *database* yang sudah disiapkan.



Gambar 3.3 Tahapan Verifikasi

Setelah dilakukannya tahapan identifikasi maka akan dilanjutkan dengan tahapan verifikasi seperti yang terlihat pada gambar diatas. mahasiswa melakukan pengenalan wajah yang sudah dikenali oleh sistem. Sistem akan membaca biometrik dan deskriptor wajah dari mahasiswa tersebut dan mencocokkannya dengan data yang sudah tersimpan di *database*. Jika wajah sesuai dengan data yang ada di *database* maka akan keluar hasil “berhasil” tapi kalau wajah tidak cocok maka akan keluar hasil “gagal”.

3.5 Populasi dan Sampel

Populasi merupakan suatu wilayah generalisasi yang terdiri dari obyek dan subyek yang memiliki kualitas dan ciri tertentu yang telah ditetapkan oleh peneliti untuk dipelajari dan ditarik kesimpulan. Populasi tidak hanya terbatas pada manusia, melainkan mencakup semua karakteristik yang dimiliki oleh subjek atau objek penelitian.

Sampel merupakan bagian dari keseluruhan populasi dan memiliki karakteristik yang sama dengan populasi tersebut. Oleh karena itu, ketika populasi sangat besar, peneliti tidak mungkin untuk mengkaji seluruh populasi karena keterbatasan seperti waktu dan dana. Untuk mengestimasi ukuran sampel dari populasi yang diketahui jumlahnya, digunakan rumus Slovin, sebagai berikut :

$$n = \frac{N}{1 + N (e)^2}$$

Gambar 3.4 Rumus Slovin

n= ukuran sampel

N = ukuran populasi

e = persentase kelonggaran kesalahan pengambilan sampel yang masih bisa ditoleransi

e = 0,05

Jumlah populasi yang diketahui adalah 520 mahasiswa. Dengan menggunakan rumus Slovin untuk menentukan sampel dengan populasi yang diketahui (N), tingkat kepercayaan 95% dan tingkat kesalahan 5% (e), dapat dihitung jumlah sampel sebagai berikut :

$$n = \frac{N}{1 + N (e)^2}$$
$$n = \frac{505}{1 + 505 (0,05)^2}$$
$$n = \frac{505}{2.2625} = 223$$

Oleh karena itu, jumlah orang yang menjadi sampel adalah 223. Metode sampling yang akan digunakan dalam penelitian ini adalah *Simple Random Sampling* dengan kriteria inklusif. Teknik ini disebut sederhana karena pengambilannya dilakukan secara acak tanpa memperhatikan strata. Adapun, kriteria inklusif untuk sampel adalah sebagai berikut :

1. Merupakan mahasiswa Prodi Teknologi Informasi UIN Ar-raniry.
2. Pernah/sedang mengikuti belajar di lab.

3.6 Teknik Pengumpulan data

Pengumpulan data ialah suatu metode wajib dipenuhi dalam melakukan suatu penelitian. Tujuannya adalah untuk mendapatkan data yang valid dan benar, maka penulis menggunakan beberapa metode untuk pengumpulan data sebagai berikut :

3.6.1 Observasi

Tahapan pengumpulan informasi dan data-data awal, peneliti melakukan observasi untuk mengumpulkan informasi yang diperlukan untuk mengembangkan sistem absensi otomatis mahasiswa berbasis video di Prodi TI. Observasi dilakukan pada prodi TI bertepatan pada tanggal 10 Agustus 2023. Adapun hasilnya sebagai berikut: Bagaimana seluruh mahasiswa pada Prodi TI masih menggunakan absensi *convensional* atau dengan melakukan absensi *online* melalui *website*.

3.6.2 Studi literatur

Metode pengumpulan yang dilakukan penulis untuk mencari berbagai data-data mengenai sistem absensi wajah mahasiswa. Informasi yang digunakan sebagai refensi yang didapatkan dari membaca dan mempelajari buku-buku, jurnal, artikel penelitian dan berbagai sumber lainnya yang berhubungan dengan sistem informasi yang diangkat penulis.

3.7 Analisis Kebutuhan

Analisa untuk kebutuhan merupakan langkah awal untuk pengembangan perangkat lunak yang tujuannya memperoleh informasi dan spesifikasi tentang perangkat lunak yang sesuai dengan *user* atau klien. Kedua belah pihak, yaitu *user* dan *creator* dari perangkat lunak sangat dibutuhkan pada tahapan ini. Informasi dari *user*

nantinya akan terlibat untuk menjadi rujukan dalam membuat desain dari sebuah perangkat lunak.

Adapun beberapa kebutuhan yang dibutuhkan oleh sistem untuk analisis yaitu seperti menganalisis kebutuhan perangkat lunak (*Software*) dan analisis kebutuhan perangkat keras (*hardware*). Langkah-langkah yang dibutuhkan oleh sistem adalah sebagai berikut:

3.7.1 Kebutuhan Perangkat Lunak (*Software*)

Software yang telah digunakan pada pengembangan sistem wajah mahasiswa pada penelitian ini adalah sebagai berikut:

- a. *Visual Studio Code*
- b. *Php*
- c. *Navicat / Xampp*

3.7.2 Kebutuhan Perangkat Keras (*Hardware*)

Spesifikasi dari perangkat keras (*hardware*) yang digunakan oleh penulis untuk pengembangan sistem absensi wajah mahasiswa pada penelitian ini adalah sebagai berikut:

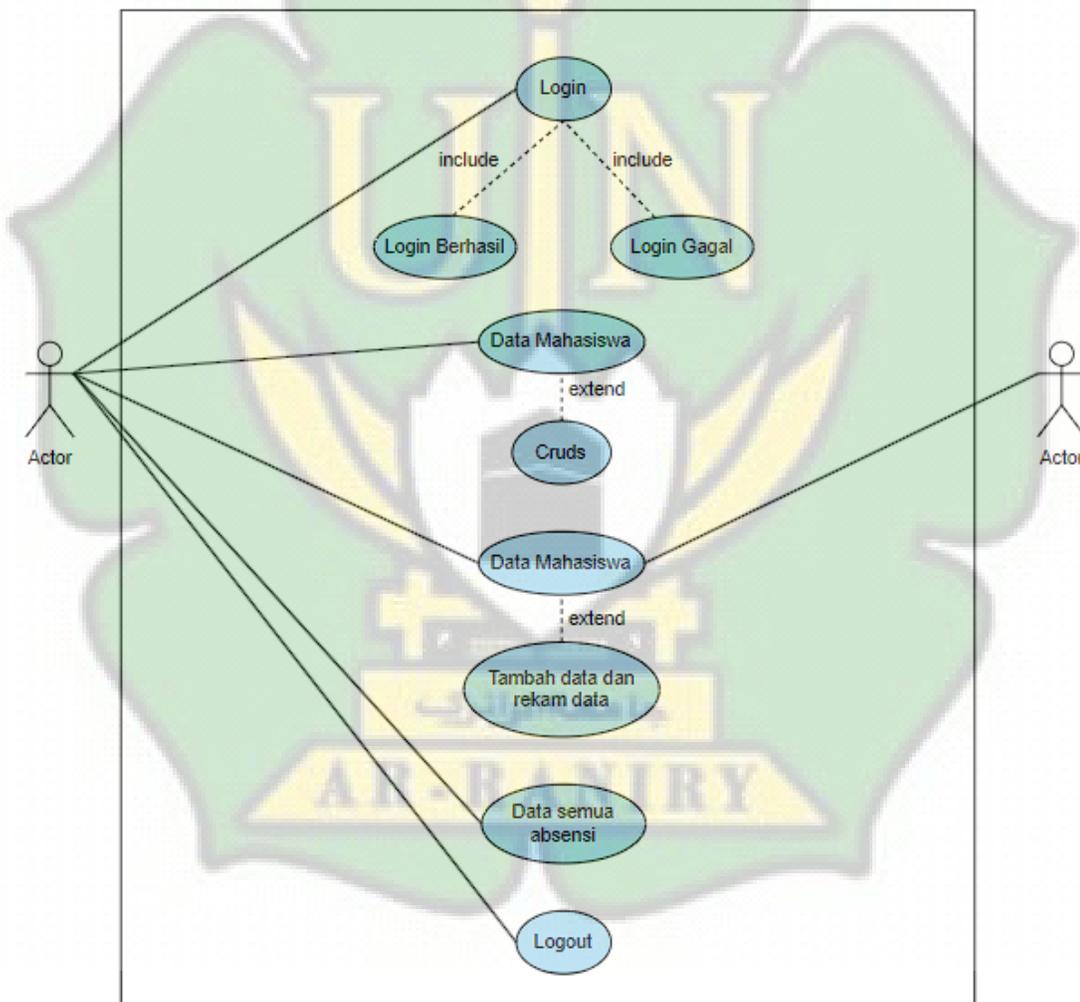
Laptop	:	Asus
Processor	:	AMD Ryzen 5 3550H with Radeon Vega Mobile 2.10 GHz
RAM	:	16 GB
Penyimpanan	:	512 GB
Monitor	:	14 inch
Sistem Operasi	:	Windows 11 Home

3.8 Perancangan Sistem

Perancangan sistem ialah proses yang dilakukan setelah terjadinya komunikasi antara *client* dalam pembuatan sebuah sistem. Ada beberapa hal yang diperlukan untuk merancang sistem absensi wajah mahasiswa adalah sebagai berikut:

3.8.1 Use Case Diagram

Use case ini sebagai gambaran proses yang nantinya akan dirancang pada sistem. Sistem tersebut akan melakukan interaksi dengan *actor* dan *user* untuk menjelaskan tentang keseluruhan sistem. Mengacu pada (Nym, 2022) gambar *usecase diagram* dibawah merupakan penggunaan umum untuk sistem absensi wajah mahasiswa, dengan menunjukkan fungsi umum yang dapat dilakukan oleh sistem. Hal ini didasarkan pada saat yang diperlukan dalam memeriksa kehadiran para mahasiswa.



Gambar 3.5 Use Case Diagram

Berdasarkan use case diagram pada gambar 3.5 diatas, sistem absensi pengenalan wajah memiliki beberapa kegiatan, yaitu sebagai berikut :

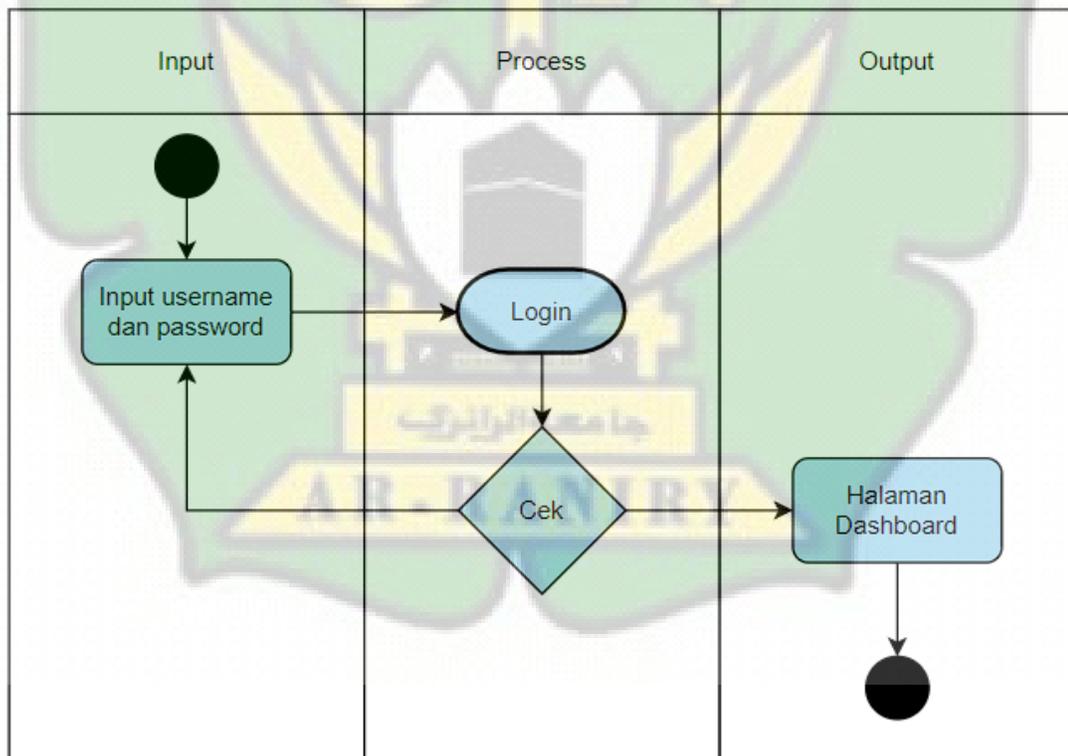
1. Admin dapat mengakses login, *dashboard*, data mahasiswa, data absen hari ini, semua data absensi dan seluruh fitur yang ada pada sistem absensi pengenalan wajah.
2. Mahasiswa dapat mengakses halaman data mahasiswa saja yang digunakan untuk melakukan absen masuk dan absen pulang.

3.8.2 Activity Diagram

Activity digram merupakan rancangan aliran aktivitas atau aliran kerja dalam sebuah sistem yang akan dijalankan (Rizky, 2019).

a. Activity diagram – Login

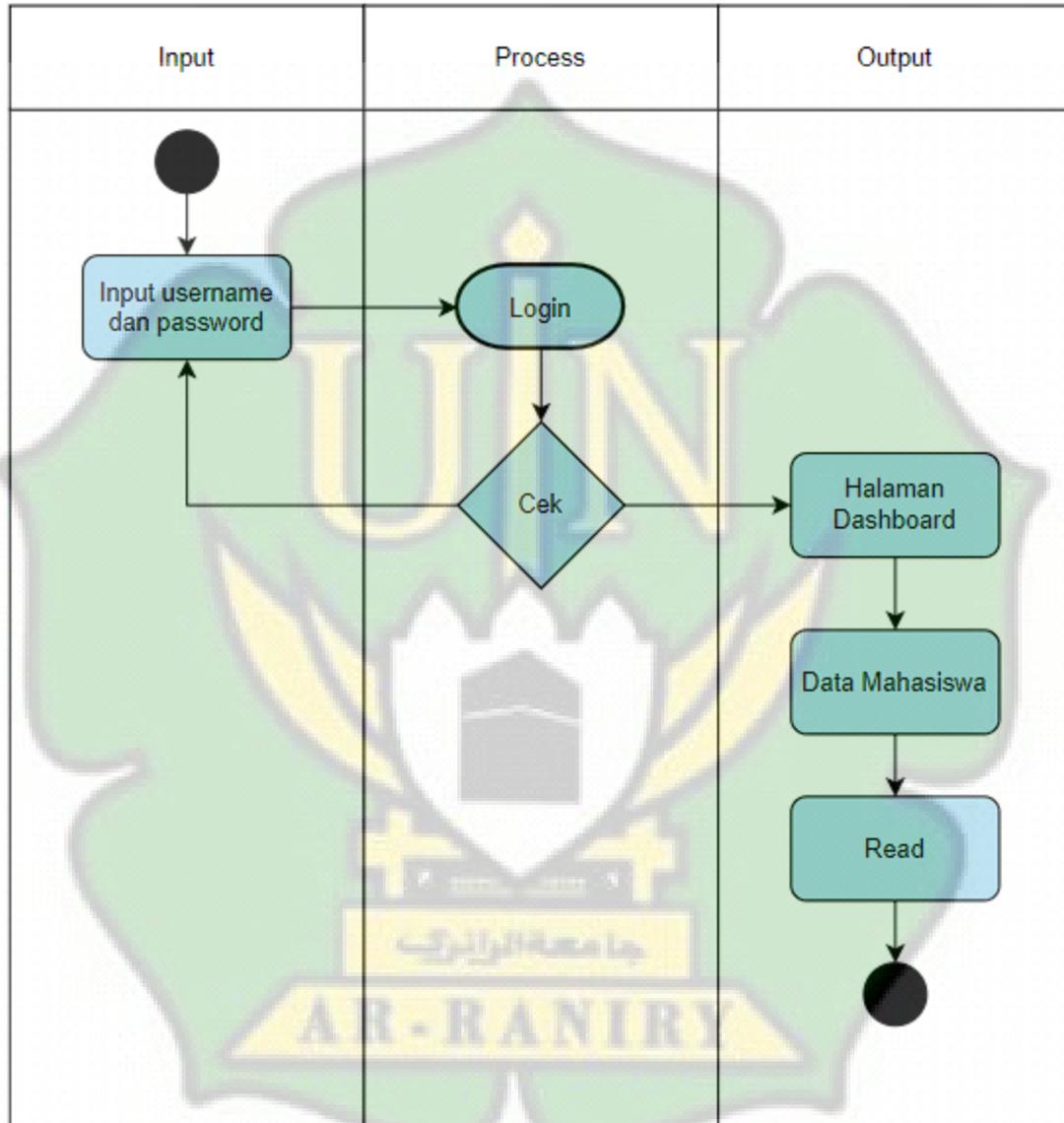
Administrator dapat memasukkan *username* dan *password* guna memungkinkan administrator dapat mengakses sistem. Setelah proses validasi data berhasil, administrator akan diarahkan ke menu utama sistem atau *dashboard*.



Gambar 3.6 Activity Diagram login

b. *Activity diagram* – Data Mahasiswa

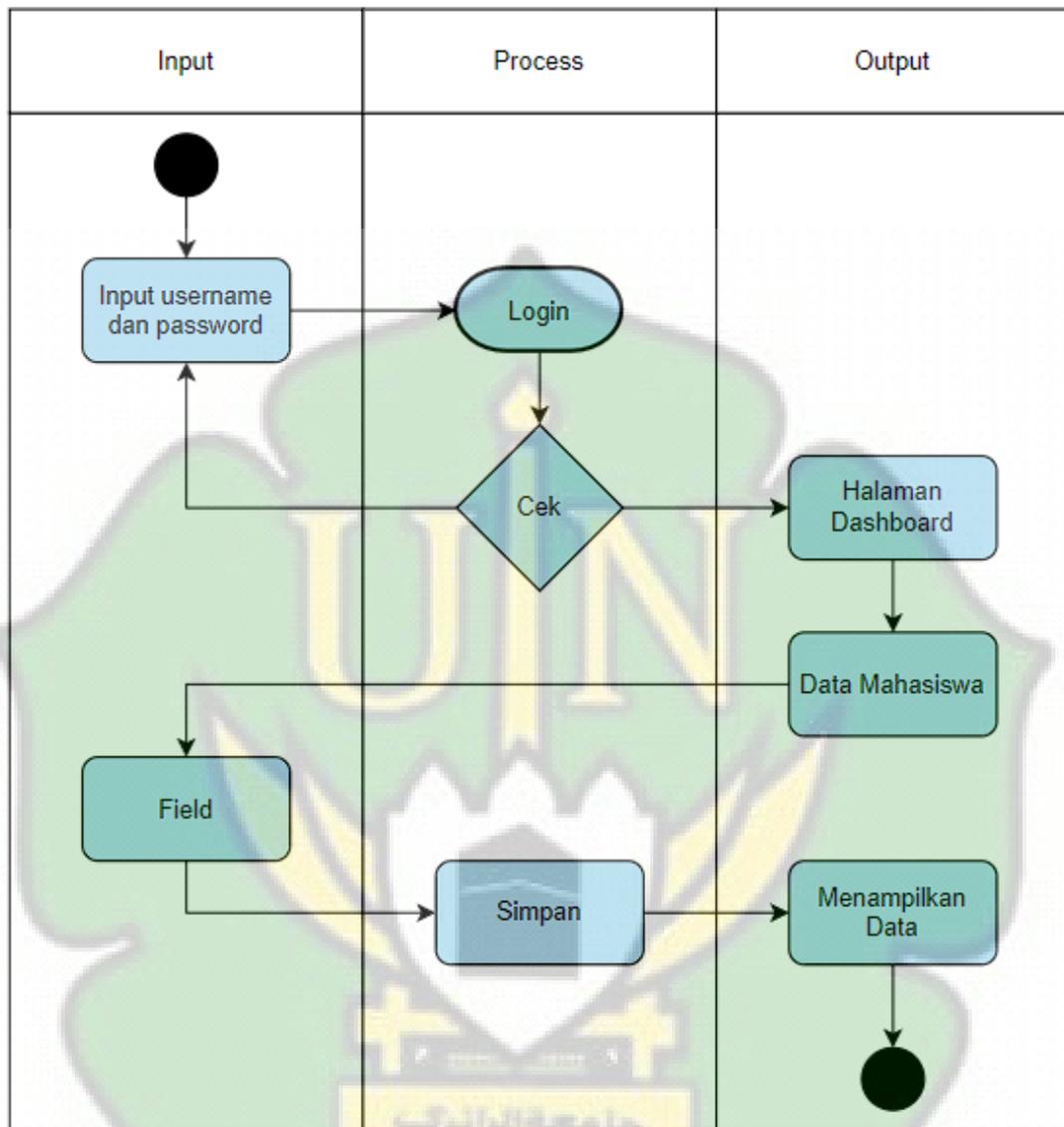
Setelah masuk ke halaman *dashboard* admin dapat mengakses halaman data mahasiswa.



Gambar 3.7 *Activity Diagram* Data Mahasiswa

c. *Activity diagram* – Tambah data

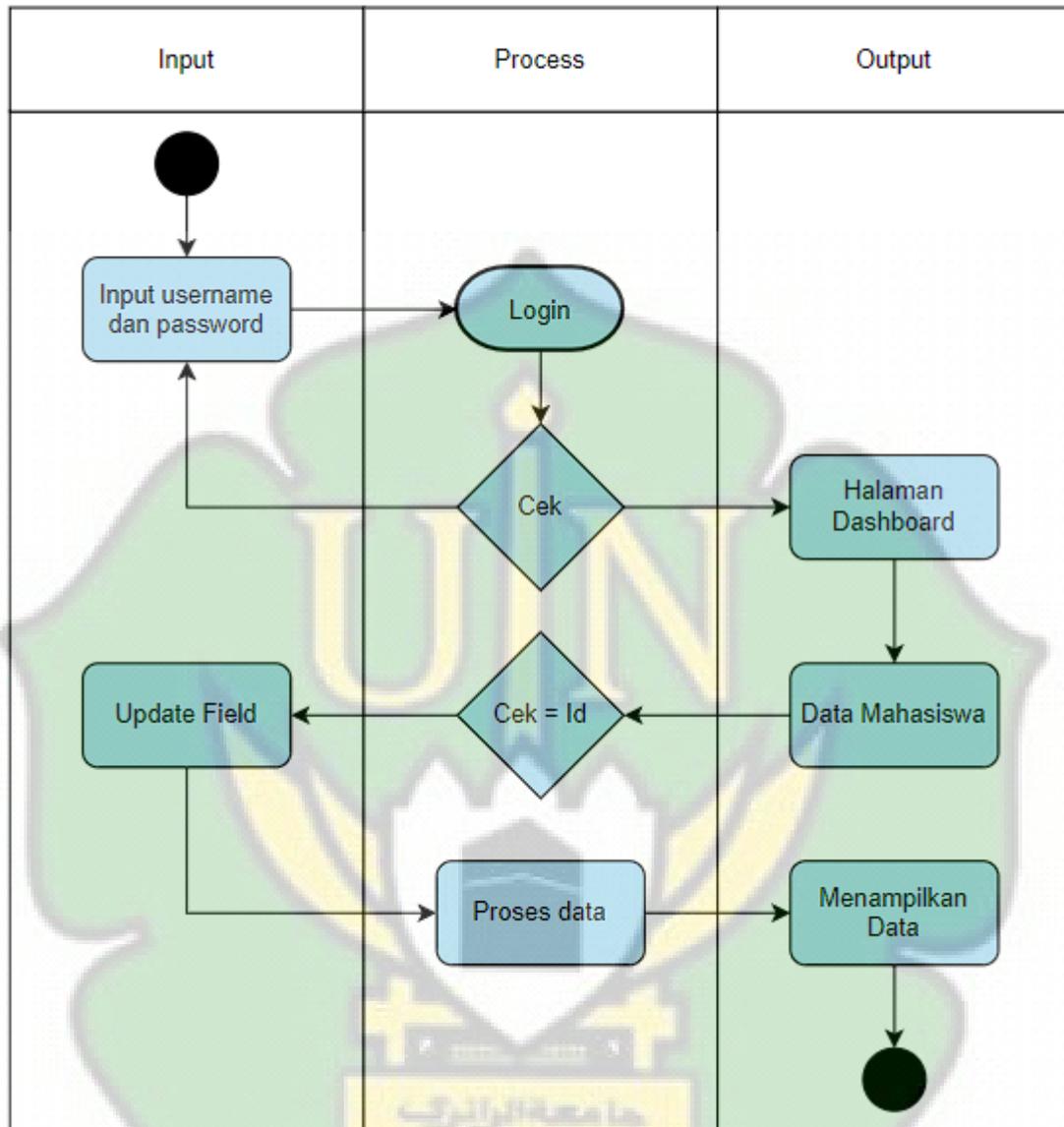
Setelah masuk halaman data mahasiswa, admin dapat melakukan tambah data mahasiswa berupa nama, nim, unit, dan angkatan.



Gambar 3.8 Activity Diagram Tambah data.

d. *Activity diagram* – Ubah data

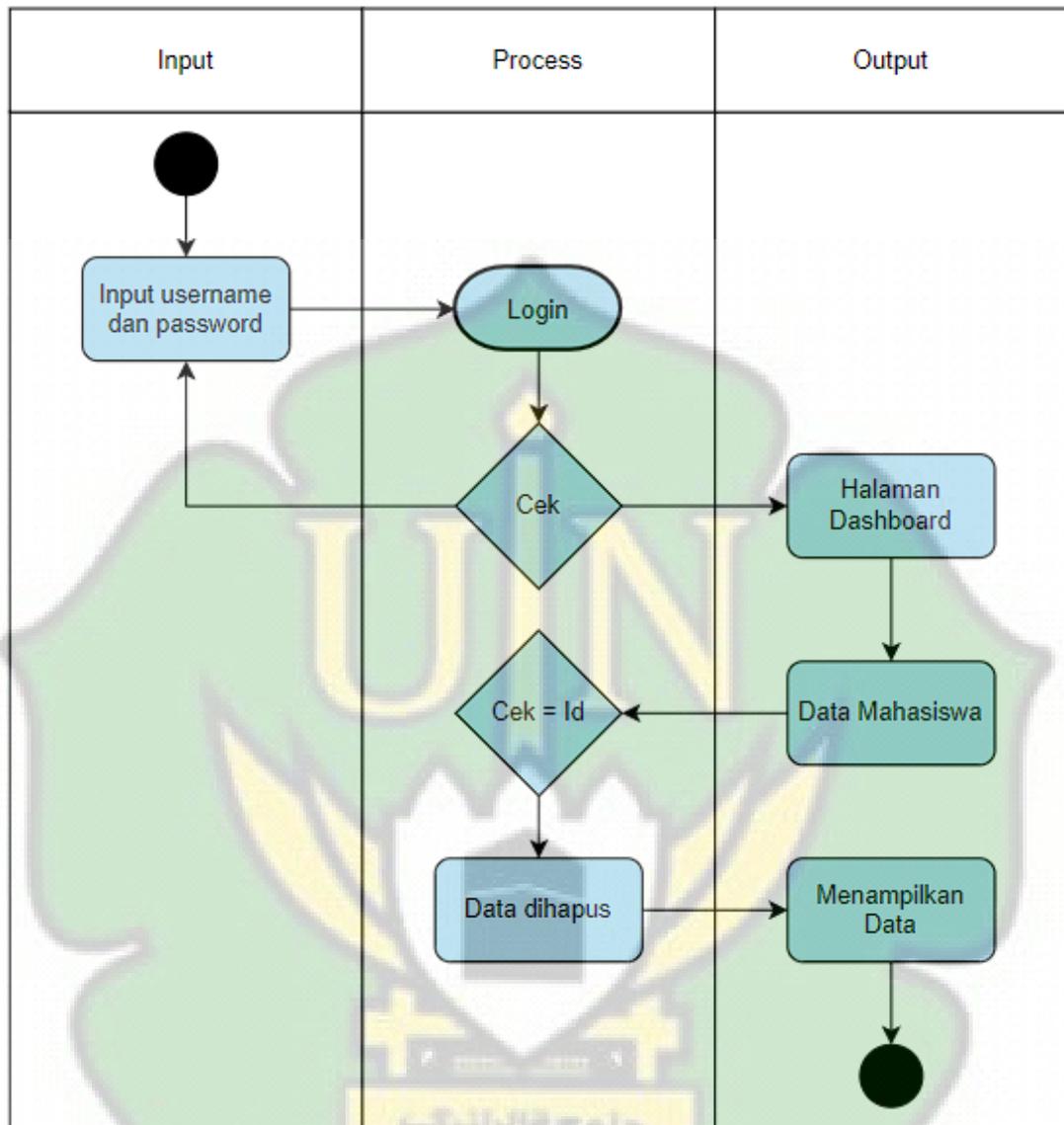
Aktivasi yang dilakukan oleh admin adalah untuk mengubah data atau meng-*edit* data mahasiswa.



Gambar 3.9 Activity Diagram Ubah data

e. Activity diagram – Hapus data

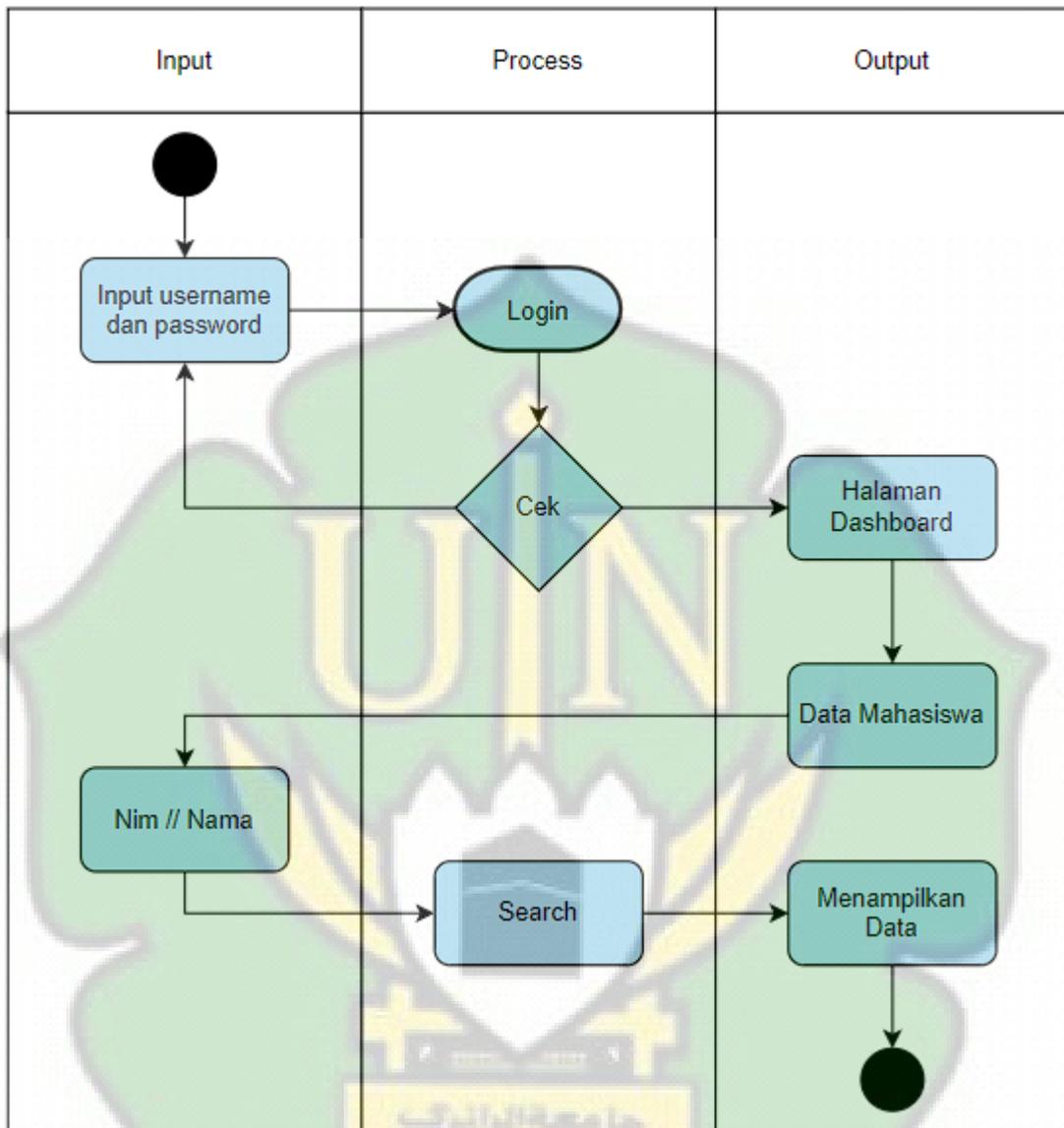
Aktivasi yang dilakukan oleh admin adalah untuk menghapus data mahasiswa.



Gambar 3.10 Activity Diagram Hapus data

f. Activity diagram – Cari data

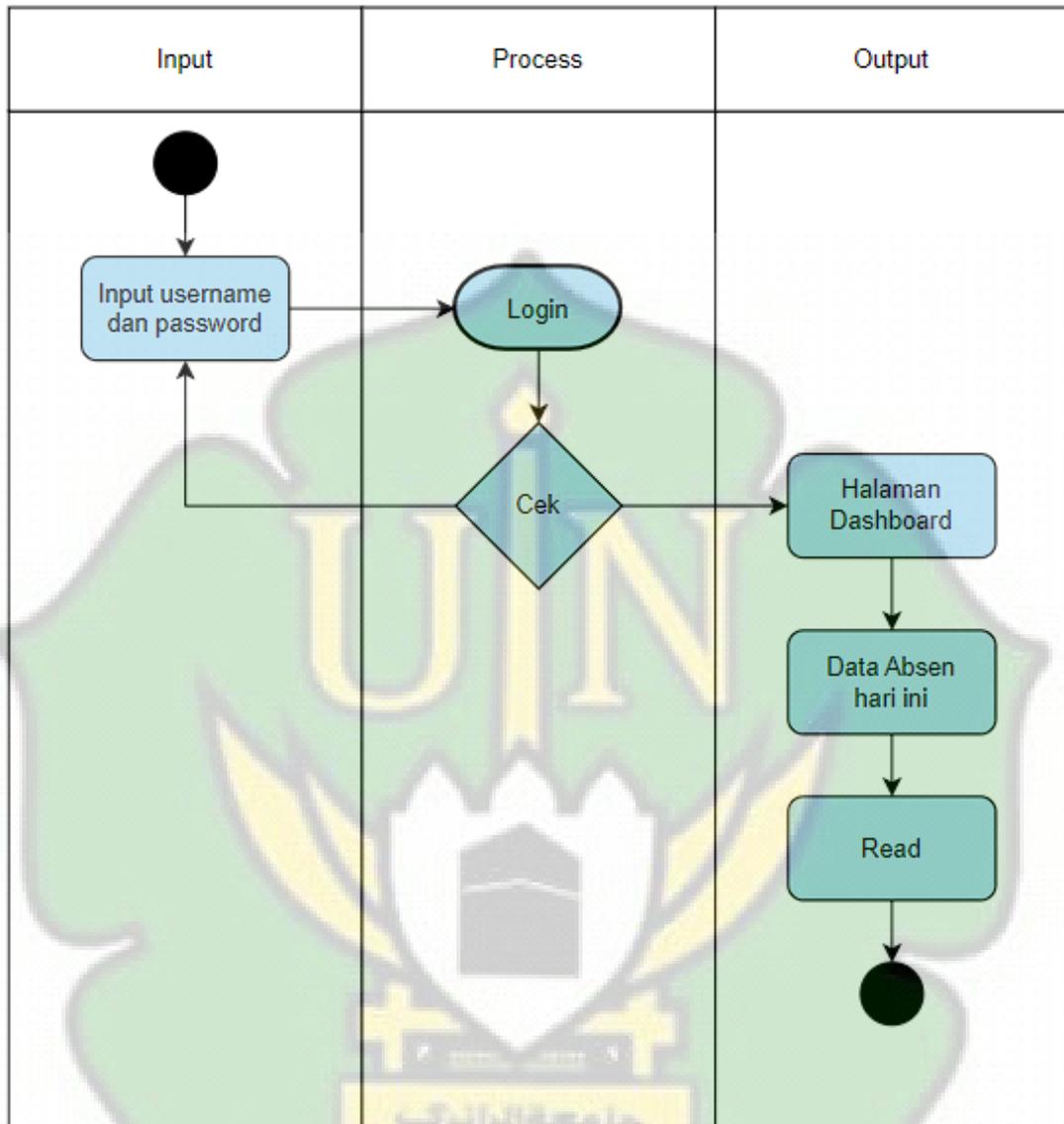
Aktivasi yang dilakukan oleh admin adalah untuk mencari data dari mahasiswa.



Gambar 3.11 Activity Diagram Cari data

g. Activity diagram – Data Absen Hari Ini

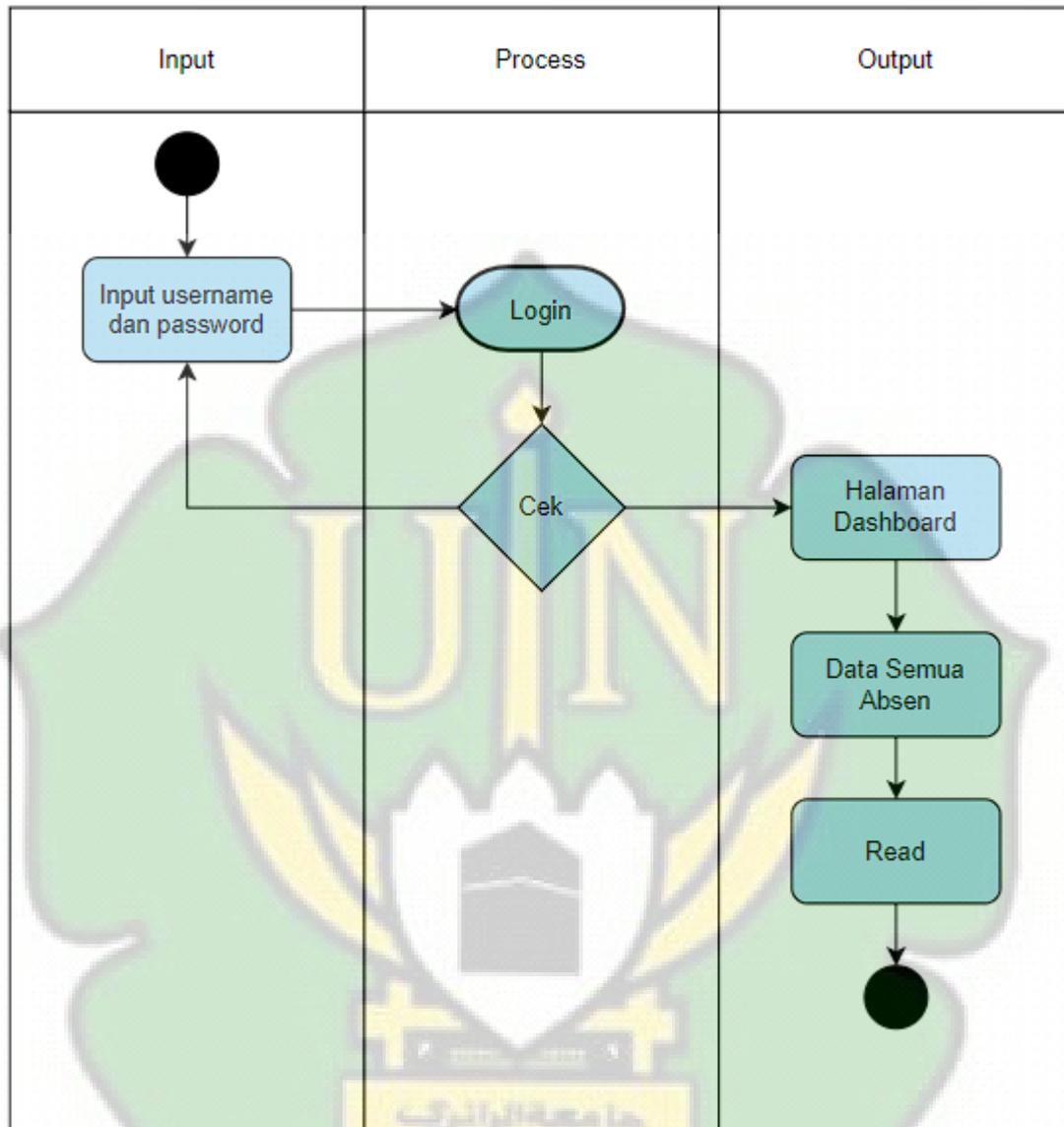
Setelah mengakses halaman data mahasiswa admin juga dapat mengakses halaman data absen hari ini.



Gambar 3.12 Activity Diagram Data absen hari in

h. Activity diagram – Data Semua Absensi

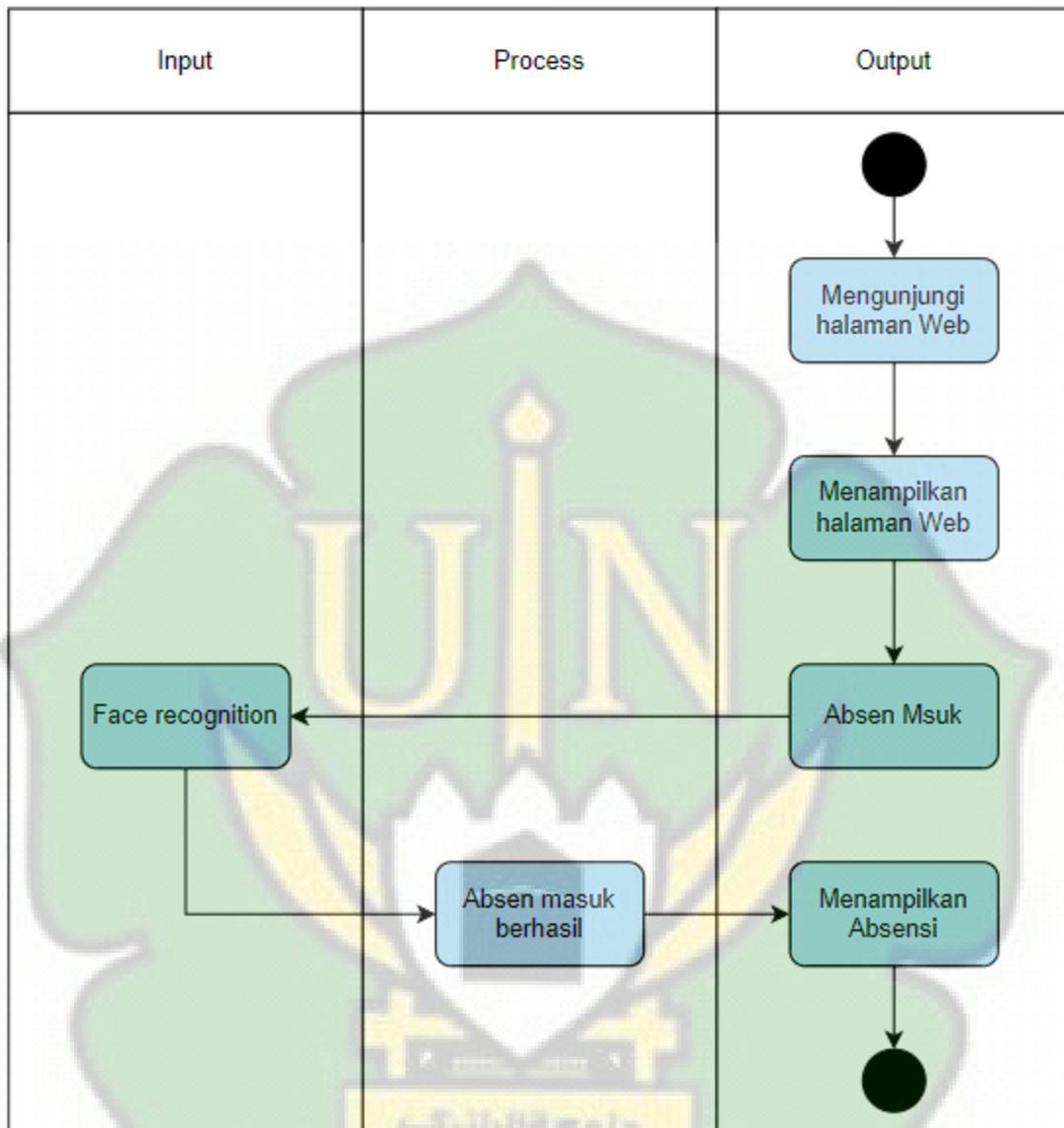
Setelah mengakses halaman data absen hari ini admin juga dapat mengakses halaman data semua absensi.



Gambar 3.13 Activity Diagram Data semua absensi

i. Activity diagram – Absen Masuk

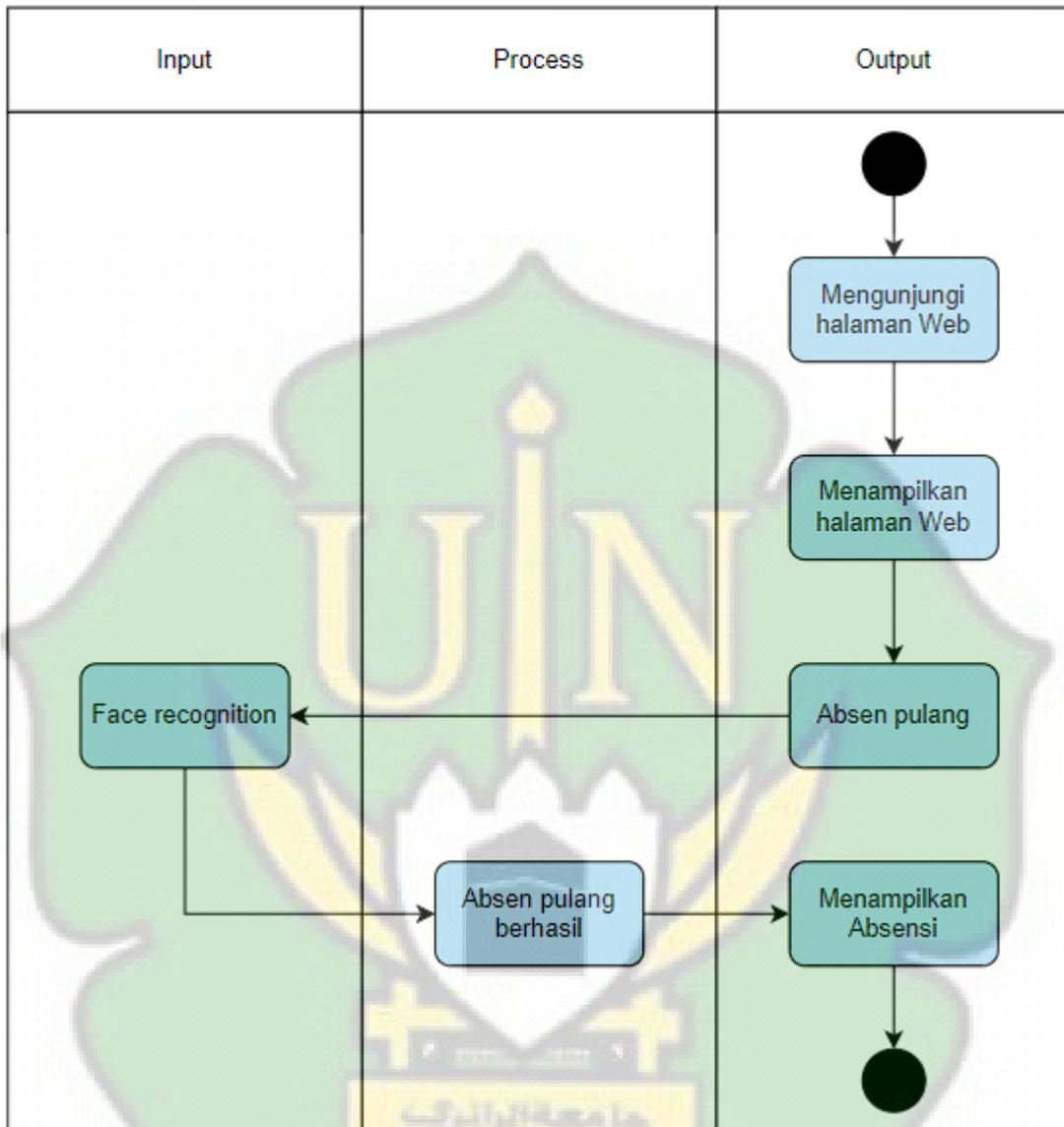
Aktivasi yang dilakukan oleh mahasiswa untuk melakukan absen masuk mata kuliah terkait.



Gambar 3.14 Activity Diagram Absen masuk

j. Activity diagram – Absen Pulang

Aktivasi yang dilakukan oleh mahasiswa untuk melakukan absen pulang mata kuliah terkait.



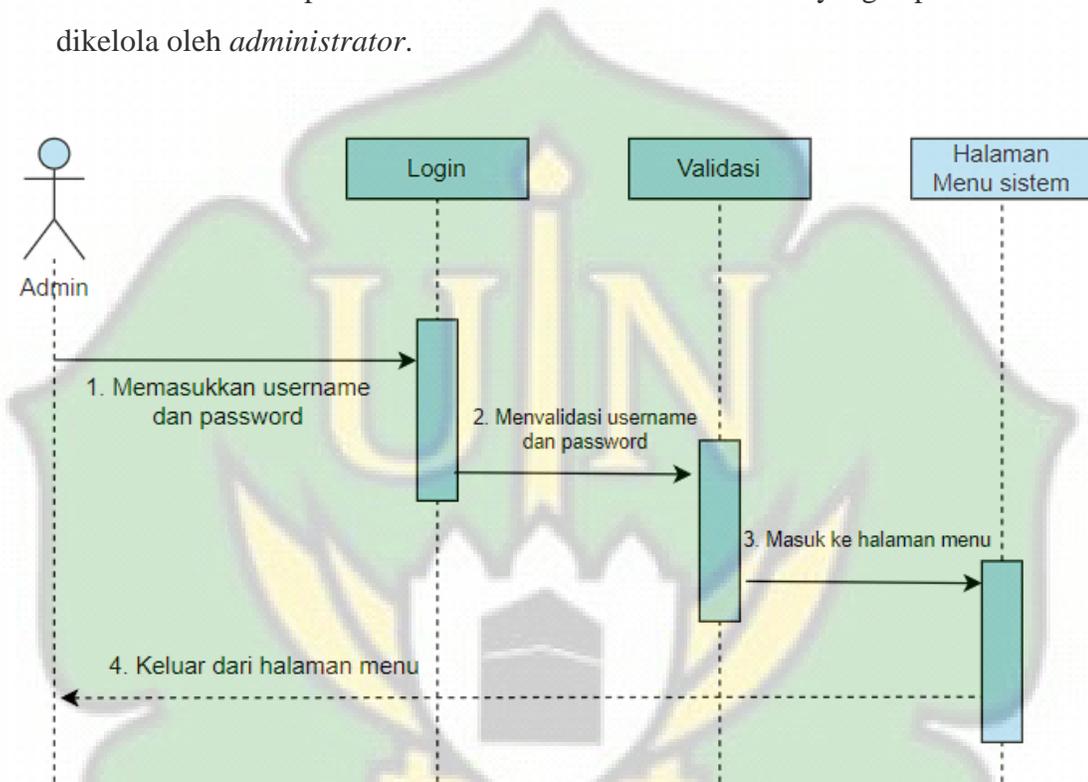
Gambar 3.15 Activity Diagram Absen pulang

3.8.3 Sequence Diagram

Sequence diagram merupakan sebuah gambaran yang menunjukkan interaksi proses yang dilakukan oleh mahasiswa dan admin terhadap sistem (Amanda Widya Pramesti, 2020).

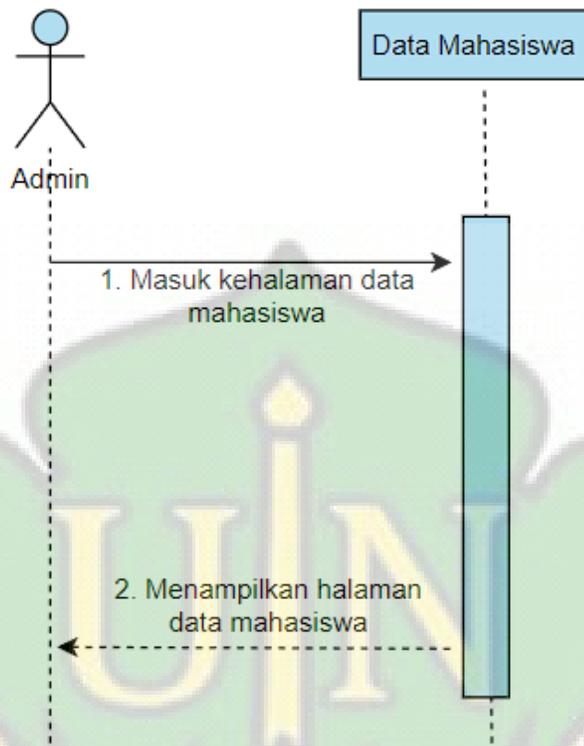
a. *Sequence Diagram – Login*

Tahap awal yang harus dilakukan oleh administrator adalah memasukkan *username* dan *password* ke dalam kolom *input*. Setelah itu, sistem akan memeriksa apakah data administrator yang dimasukkan benar atau salah. Setelah validasi data, sistem akan menampilkan halaman beranda atau *dashboard* yang dapat diakses dan dikelola oleh *administrator*.



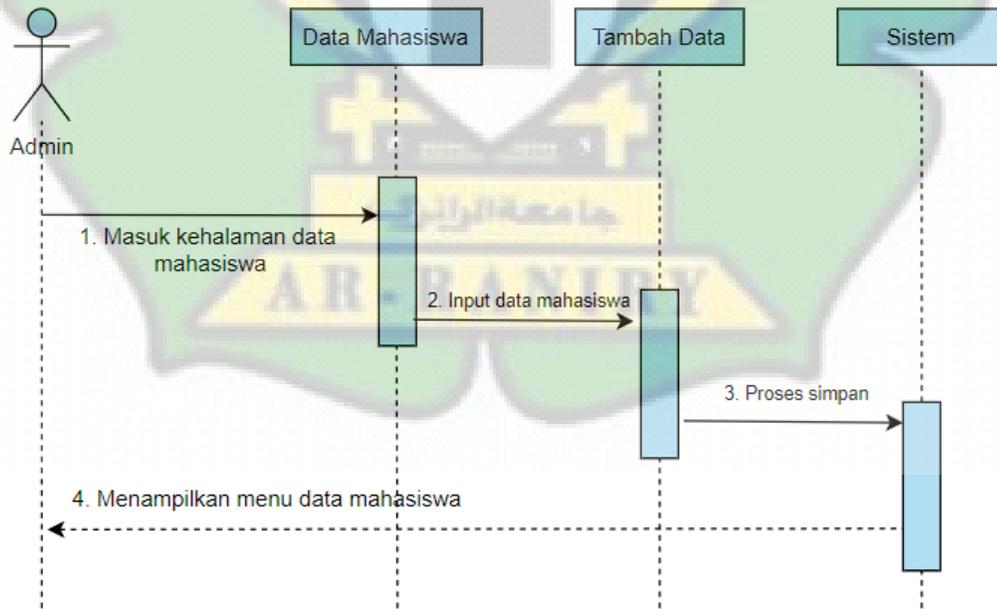
Gambar 3.16 *Sequence Diagram login*

b. *Sequence Diagram – Data Mahasiswa*



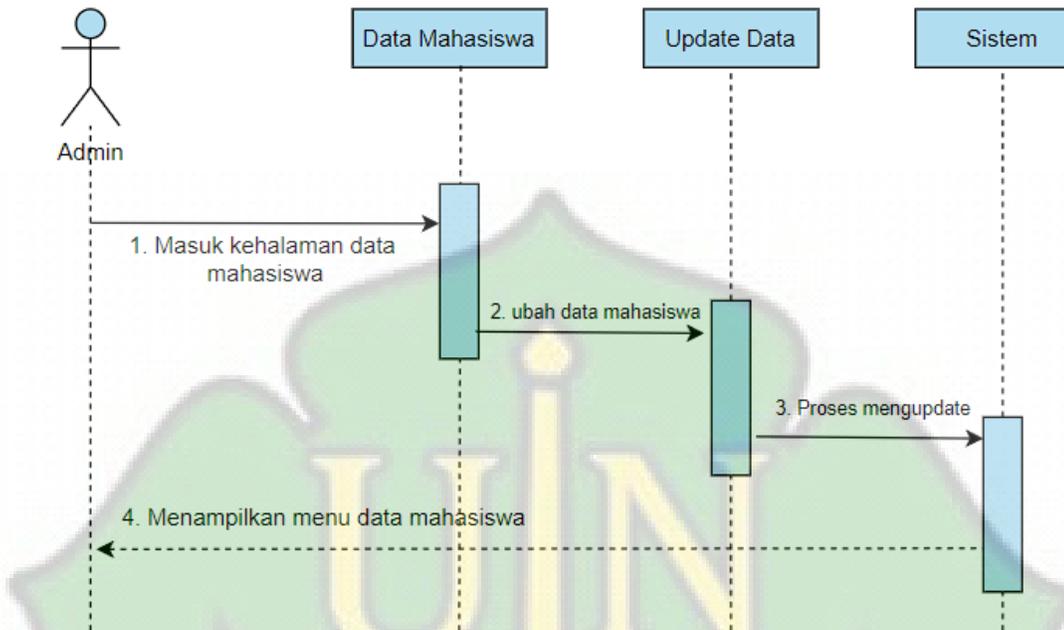
Gambar 3.17 *Sequence Diagram* Data mahasiswa

c. *Sequence Diagram* – Tambah Data



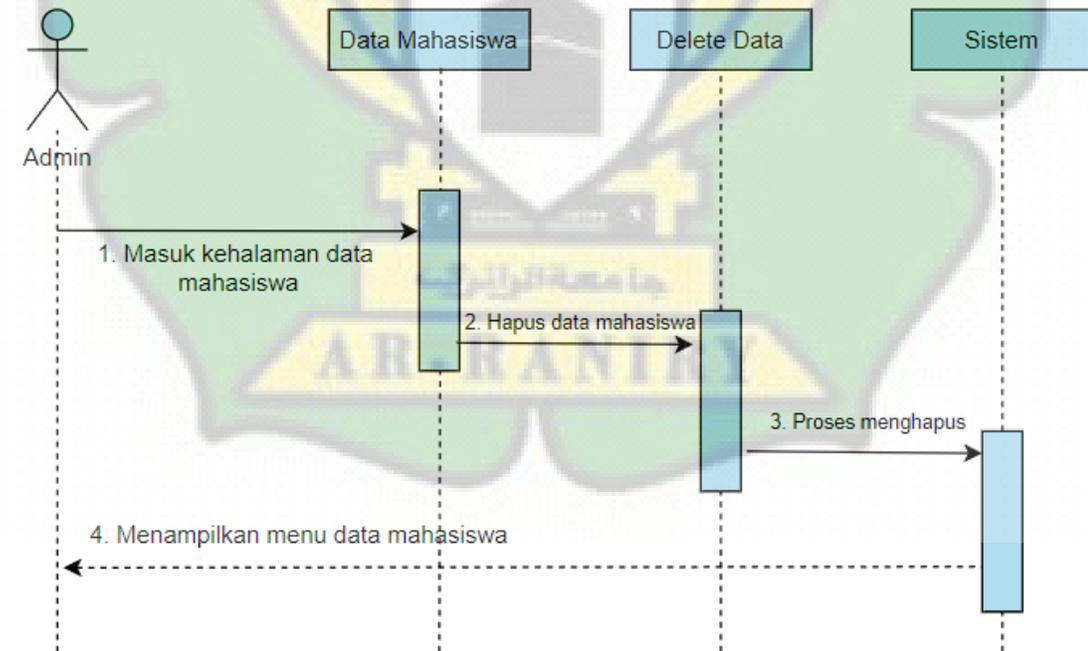
Gambar 3.18 *Sequence Diagram* Tambah data

d. *Sequence Diagram – Ubah Data*



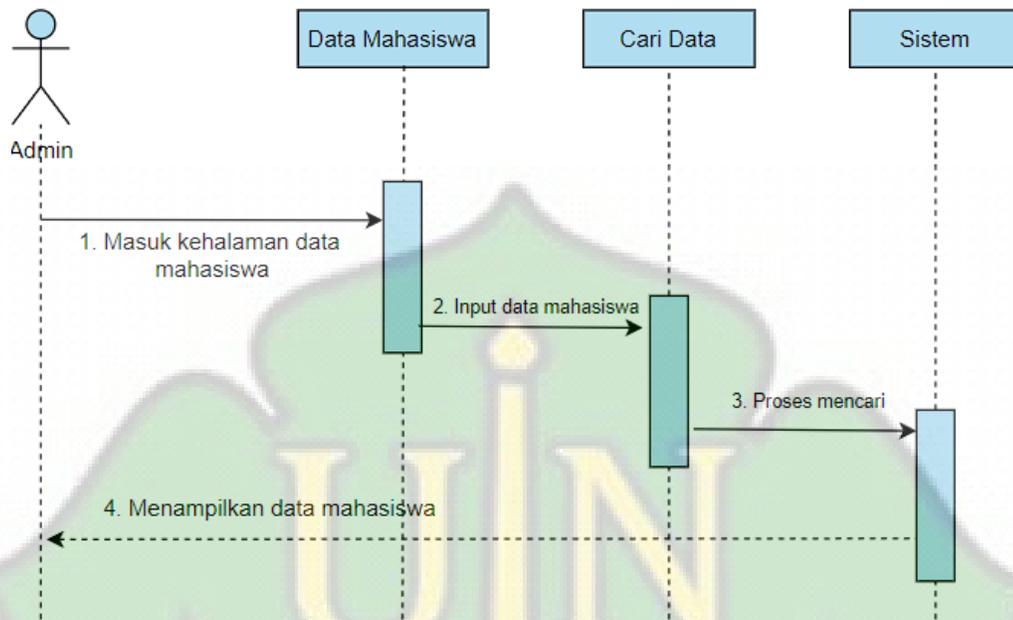
Gambar 3.19 *Sequence Diagram* Tambah data

e. *Sequence Diagram – Hapus Data*



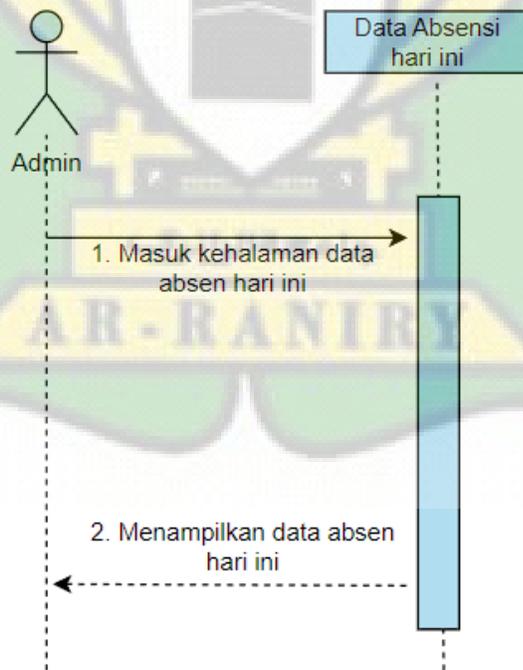
Gambar 3.20 *Sequence Diagram* Tambah data

f. *Sequence Diagram* – Cari Data



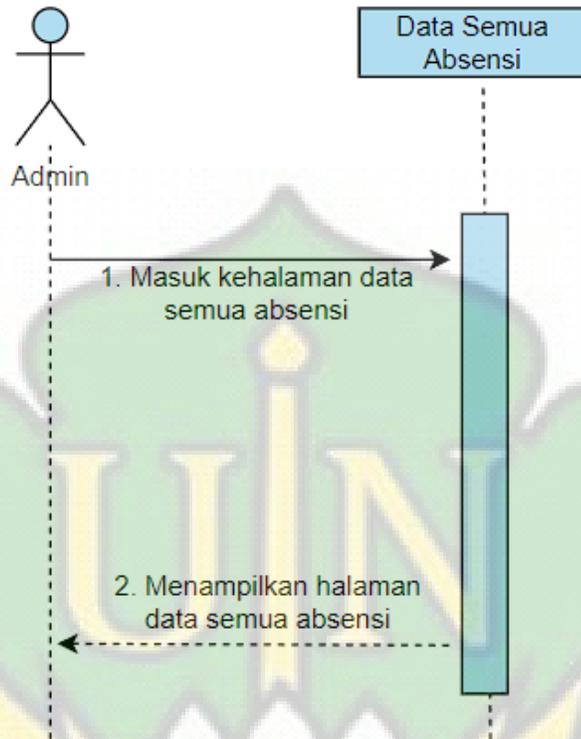
Gambar 3.21 *Sequence Diagram* Tambah data

g. *Sequence Diagram* – Data Absen Hari Ini



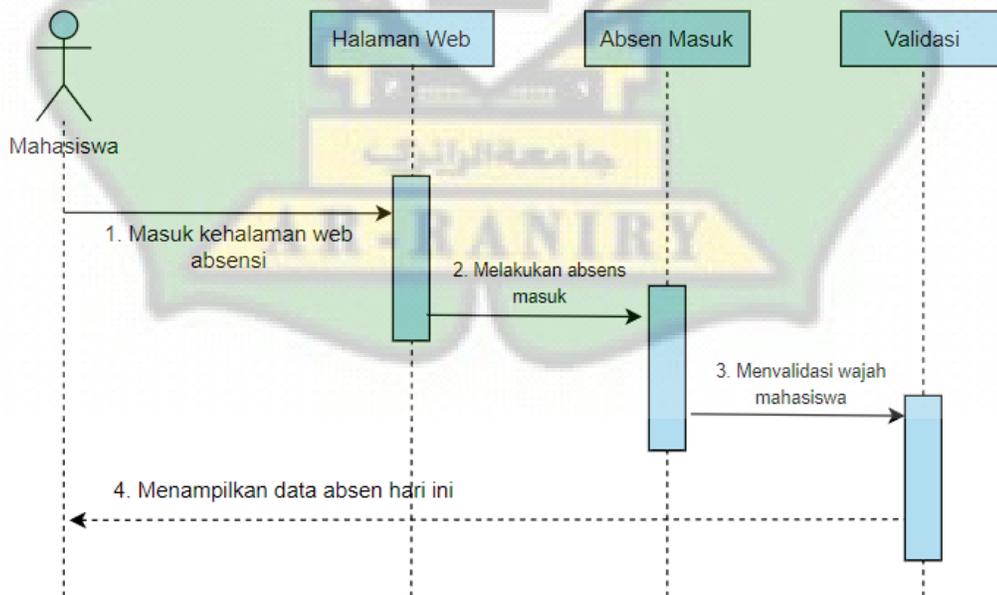
Gambar 3.22 *Sequence Diagram* Data absen hari ini

h. *Sequence Diagram* – Data Semua Absensi



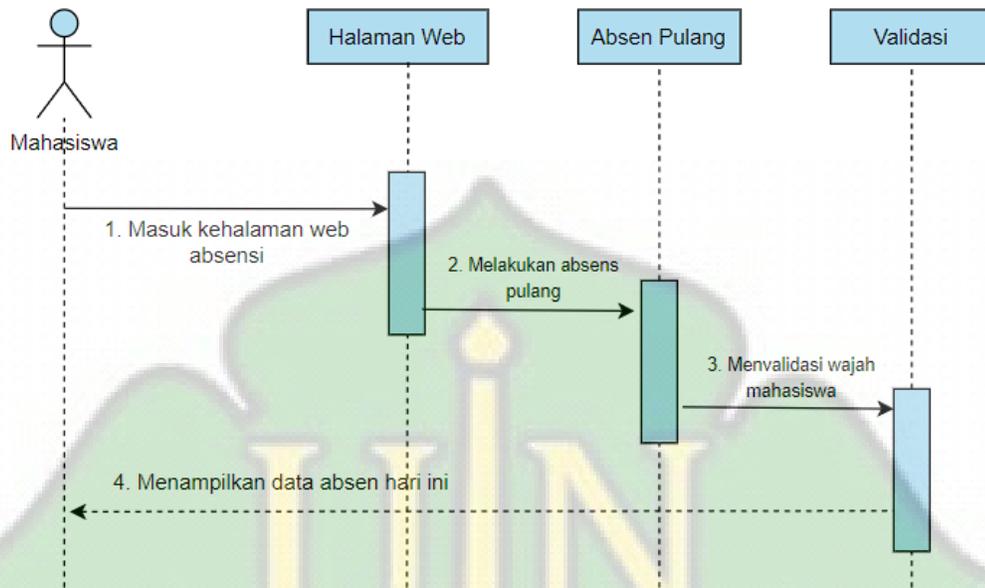
Gambar 3. 23 *Sequence Diagram* Data semua absensi

i. *Sequence Diagram* – Absen Masuk



Gambar 3.24 *Sequence Diagram* Absen masuk

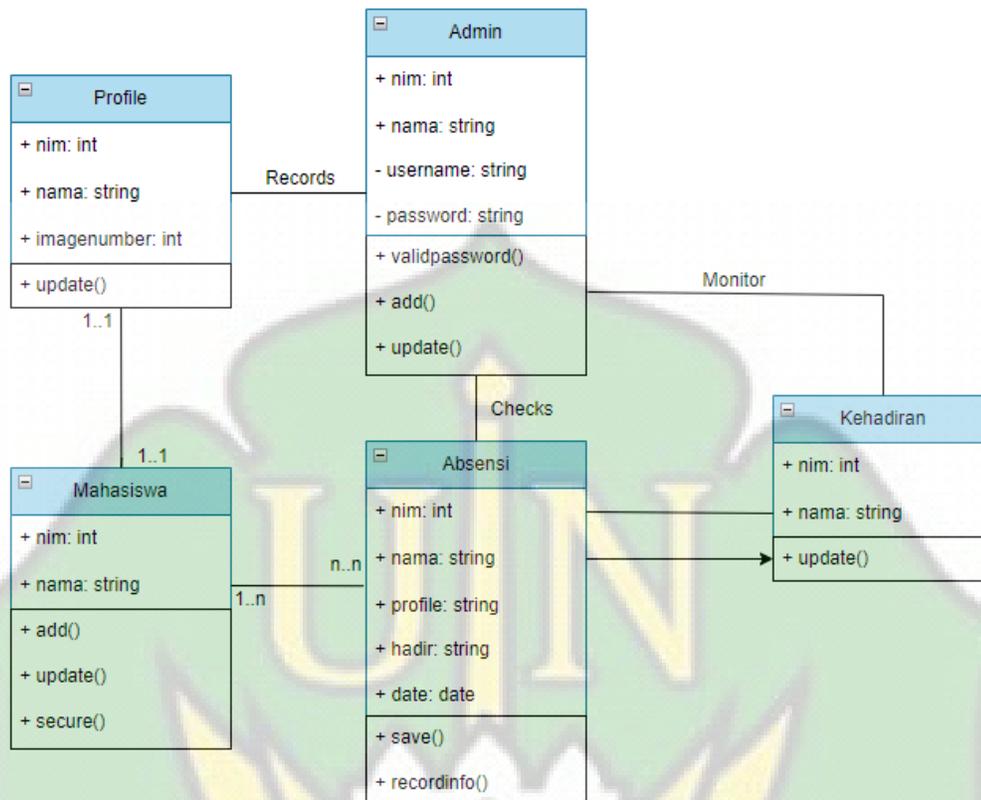
j. *Sequence Diagram – Absen Pulang*



Gambar 3.25 *Sequence Diagram Absen masuk*

3.8.4 Class Diagram

Class Diagram ini berguna sebagai contoh gambaran aktivitas yang akan dikerjakan oleh penulis kepada sistem. Adapun sebagai berikut *class diagram* sistem absensi wajah mahasiswa :

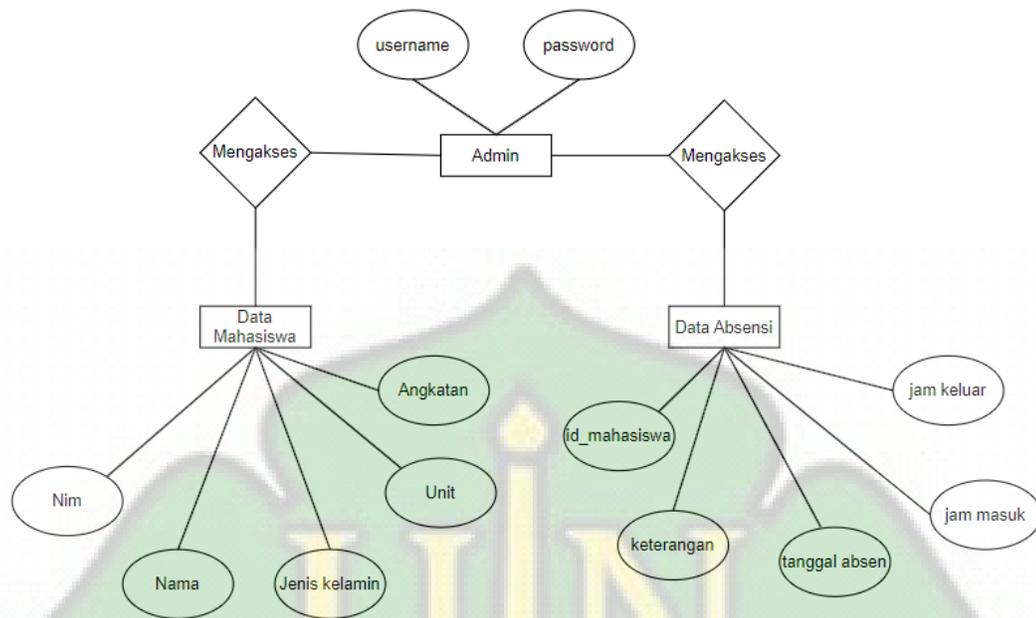


Gambar 3.26 Class diagram

Gambar diatas (*Face Recognition Attendance System | Shailersolution, n.d.*) menunjukkan gagasan sederhana tentang cara kerja *class diagram* pada sistem absensi wajah mahasiswa. Bagian yang diidentifikasi untuk Sistem Kehadiran Pengenalan Wajah adalah profil, mahasiswa, dan kehadiran.

3.8.5 Entity Relationship Diagram

ERD (Entity Relationship Diagram) atau diagram hubungan entitas adalah sebuah diagram yang digunakan untuk merancang sebuah database dengan menunjukkan koneksi atau hubungan antara objek dan entitas beserta atribut-atributnya secara rinci. Dengan menggunakan ERD, sistem *database* yang sedang dibentuk dapat diilustrasikan dengan lebih terstruktur dan teratur.



Gambar 3.27 Entity Relationship

3.8.6 Database

Database merupakan kumpulan data yang dikelola secara sistematis dan saling terkait. *Database* juga memungkinkan pengguna untuk dengan mudah mencari, menyimpan, dan menghapus informasi (Budi, 2021) . Dapat dilihat pada tabel dibawah ini perancangan *database* Sistem Absensi Mahasiswa yang dibuat dengan phpmyadmin.

a. Tabel *User*

Field name	Data Type	Length	Description
id	bigint	20	Primary key
nama_user	varchar	255	
username	varchar	255	
password	varchar	255	
role	enum		

Tabel 3.2 Tabel *user*

b. Tabel Mahasiswa

Field name	Data Type	Length	Description
Id	bigint	20	Primary key
nim	varchar	255	
nama_mahasiswa	varchar	255	
jenis_kelamin	enum		
unit	varchar	255	
angkatan	varchar	255	
image	varchar	255	

Tabel 3.3 Tabel mahasiswa

c. Tabel Absensi

Field name	Data Type	Length	Description
id	bigint	20	Primary key
id_mahasiswa	bigint	20	
tgl_absen	date		
jam_masuk	varchar	30	
jam_keluar	varchar	30	
keterangan	enum		

Tabel 3.4 Tabel absensi

BAB IV

HASIL DAN PEMBAHASAN

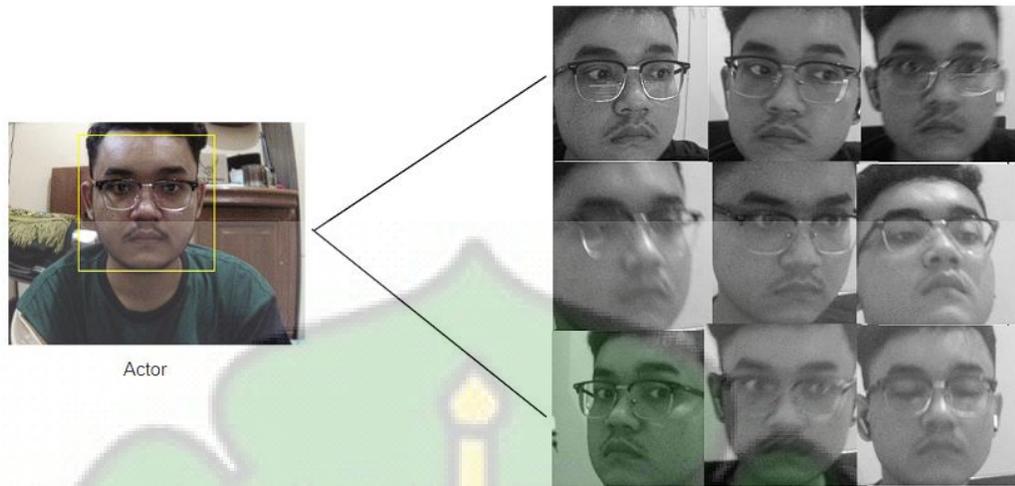
4.1 Implementasi Metode *FaceMatcher*

Untuk dapat menjalankan pengenalan wajah pada sistem absensi ini peneliti menggunakan *FaceApi.js* untuk memanggil fungsi- fungsi *face recognition*. Kemudian, digunakannya metode *FaceMatcher* untuk mencocokkan wajah mahasiswa dengan yang ada di *database*. Proses untuk memanggil kamera terdapat pada halaman absensi yang digunakan untuk mengidentifikasi foto wajah sebagai bukti kehadiran.

Untuk menghasilkan tingkat akurasi yang sempurna memang bisa dikatakan mustahil. Namun, setidaknya dengan menggunakan metode *FaceMatcher* tingkat akurasi pengenalan wajah sudah mampu mengenali wajah dengan sangat baik. Untuk mendapatkan hasil yang akurat, pengenalan wajah akan dilakukan dengan mencocokkan deskriptor.

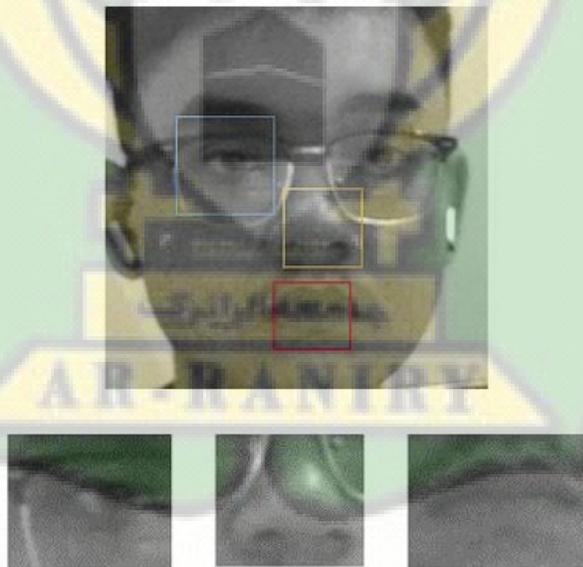
Sistem pengenalan wajah menggunakan deskriptor untuk merepresentasikan wajah. Deskriptor ini pada dasarnya adalah pengkodean matematis yang menangkap karakteristik unik wajah, seperti jarak antara mata, bentuk garis rahang, dan posisi hidung. Berikut ini adalah perincian umum tentang cara kerja pencocokan wajah dengan JSON deskriptor:

1. *Face Detection*: Tahap awal sistem adalah mendeteksi wajah dalam sebuah gambar. Proses ini umumnya dilakukan dengan menggunakan model pencocokan descriptor wajah menggunakan kumpulan data yang telah tersedia.

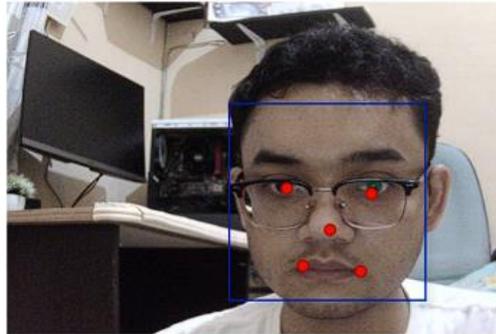


Gambar 4.1 *Face Detection*

2. *Feature Extraction*: Setelah wajah berhasil terdeteksi, sistem akan mengekstrak fitur-fitur dari wajah tersebut. Fitur-fitur ini kemudian digunakan untuk menghasilkan deskriptor wajah.



Gambar 4.2 *Feature Extraction 1*



Gambar 4.5 *Descriptor Comparison 2*

Seperti terlihat pada gambar diatas, yang mana diuji dengan menggunakan dua wajah orang yang sama dalam kondisi yang berbeda sistem mampu membaca fitur dari objek yang terkait dengan akurat. Fitur wajah tersebut dibaca dalam file json pada deskriptor. Dapat dilihat hasil pencocokan deskriptor pada gambar dibawah ini.

```
async function createFaceMatcher(data) {
  const labeledFaceDescriptors = await Promise.all(data.parent.map(className => {
    const descriptors = [];
    for (var i = 0; i < className._descriptors.length; i++) {
      descriptors.push(className._descriptors[i]);
    }
    return new faceapi.LabeledFaceDescriptors(className._label, descriptors);
  }));
  return new faceapi.FaceMatcher(labeledFaceDescriptors, 0.6);
}
```

Gambar 4.6 *Descriptor Comparison 3*

Fungsi ini pertama-tama membuat larik objek faceapi.LabeledFaceDescriptors menggunakan larik data.parent. Untuk setiap className di data.parent, fungsi ini membuat larik deskriptor dengan mengulangi ulang className._descriptors. Kemudian membuat objek faceapi.LabeledFaceDescriptors baru menggunakan className._label dan larik deskriptor. Setelah semua objek faceapi.LabeledFaceDescriptors dibuat, fungsi tersebut membuat objek faceapi.FaceMatcher baru menggunakan larik deskriptor wajah berlabel dan nilai toleransi 0,6. Objek faceapi.FaceMatcher kemudian dapat digunakan untuk membandingkan deskriptor wajah dan menemukan kecocokan

terbaik. Nilai toleransi menentukan seberapa dekat kecocokan yang dibutuhkan. Nilai yang lebih dekat ke 1 berarti pencocokan yang lebih dekat diperlukan, sementara nilai yang lebih dekat ke 0 berarti pencocokan yang lebih jauh dapat diterima.

4. *Matching*: Jika jarak antara dua deskriptor berada di bawah ambang batas tertentu, maka sistem akan menentukan bahwa wajah dalam gambar tersebut cocok dengan salah satu wajah yang ada dalam database.

```
async function start() {
  $.ajax({
    datatype: 'json',
    url: "<?php echo site_url('Base/tampilwajah') ?>",
    data: ""
  }).done(async function(data) {
    if(data.length > 2){
      var json_str = "{\\"parent\":" + data + "\"}"
      content = JSON.parse(json_str)
      for (var x = 0; x < Object.keys(content.parent).length; x++) {
        for (var y = 0; y < Object.keys(content.parent[x]._descriptors).length; y++) {
          var results = Object.values(content.parent[x]._descriptors[y])
          content.parent[x]._descriptors[y] = new Float32Array(results)
        }
      }
      faceMatcher = await createFaceMatcher(content);
    }
    waitingDialog.hide()
    $('#parent1').show()
    $('#parent2').show()
    run();
  });
}
```

Gambar 4.7 *Matching Face 1*

Kode ini memuat model pengenalan wajah dari URL menggunakan pustaka faceapi, dan kemudian menggunakan model tersebut untuk melakukan pengenalan wajah pada set data yang diambil dari server melalui permintaan AJAX. Fungsi createFaceMatcher dipanggil dengan kumpulan data yang diproses sebagai argumen. Fungsi ini membuat objek FaceMatcher menggunakan pustaka faceapi, yang dapat digunakan untuk membandingkan wajah dan menemukan kecocokan terbaik.



Gambar 4.8 *Matching Face 2*

4.2 Implementasi Sistem dan Pembahasan

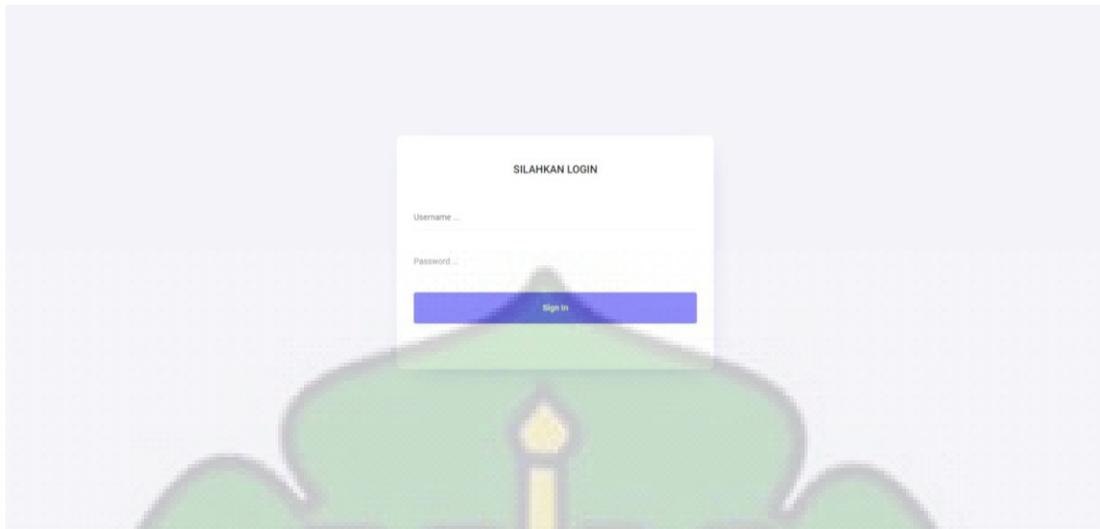
Implementasi dan pengujian sistem pemantauan kehadiran siswa dilakukan dengan tujuan untuk menemukan permasalahan pada sistem sebelum digunakan oleh pengguna. Jika pada pengujian memperlihatkan kerentanan di bagian sistem, pengguna dapat mengembangkan bagian sistem tersebut (Rahmatil Ula, 2002). Tahap implementasi sistem adalah proses membangun sistem sesuai dengan rencana atau desain sistem yang telah dipersiapkan atau dibuat terlebih dahulu, agar dapat mengoperasikan dan menguji sistem yang baru, serta memastikan keefektifan sistem. Sesuai dengan fungsinya, tahapan ini dapat dibagi menjadi beberapa bagian.

4.2.1 Tampilan Halaman Admin

Tampilan halaman admin merupakan halaman yang hanya dapat diakses oleh admin saja.

a. Tampilan Halaman Login

Tampilan login merupakan halaman awal bagi admin untuk dapat bisa mengakses website sistem absensi wajah mahasiswa.



Gambar 4.9 Halaman *login*

b. Tampilan Halaman *Home*

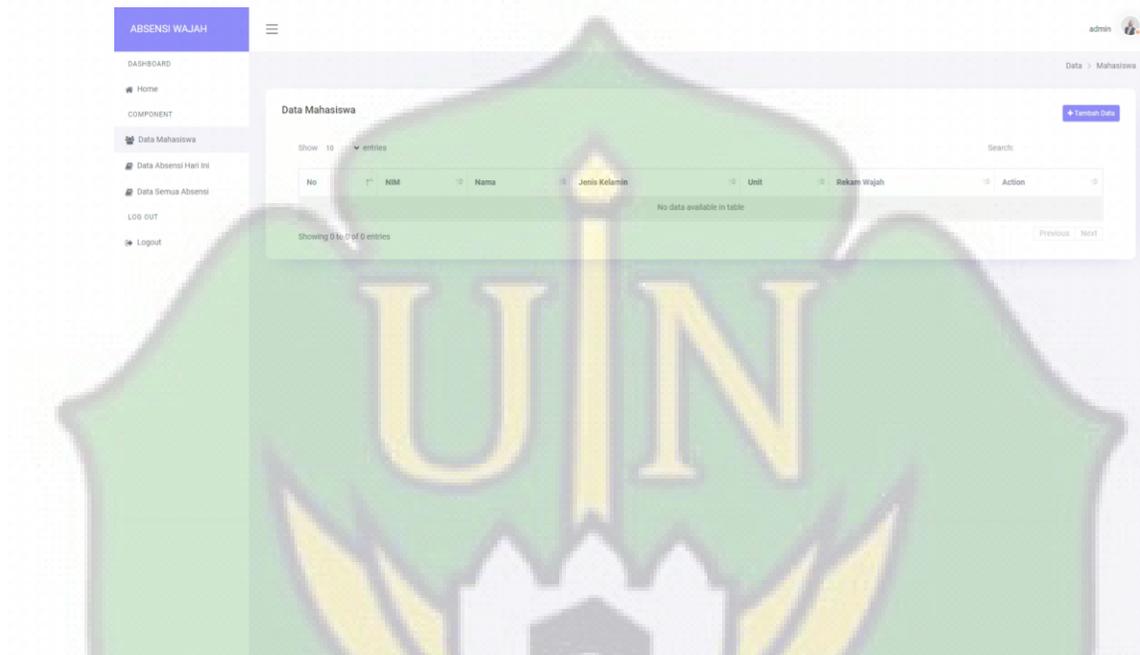
Halaman *home* ini menampilkan 3 (tiga) informasi penting yaitu: total data mahasiswa, data absensi harian, dan data seluruh absensi.



Gambar 4.10 Halaman *home*

c. Tampilan Halaman Data Mahasiswa

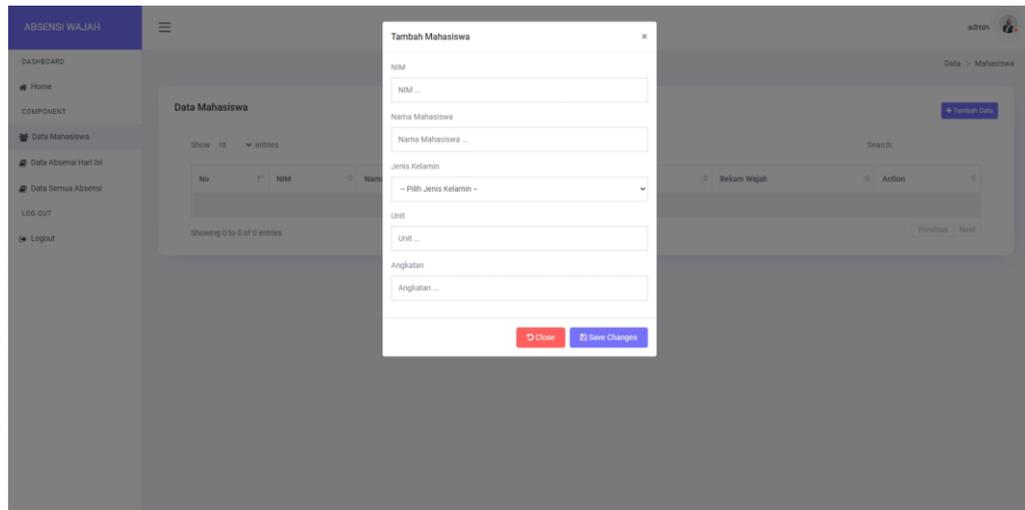
Halaman data mahasiswa ini akan menampilkan tabel yang berisi nama, nim, unit dan jenis kelamin seluruh mahasiswa yang telah mendaftar.



Gambar 4.11 Halaman data mahasiswa

- Tampilan Tambah Mahasiswa

Tambah mahasiswa merupakan halaman yang dapat di akses oleh admin untuk menambakan seluruh mahasiswa Prodi Teknologi Informasi kedalam sistem absensi wajah ini.



Gambar 4.12 Tambah mahasiswa

- Tampilan Tambah Data Wajah

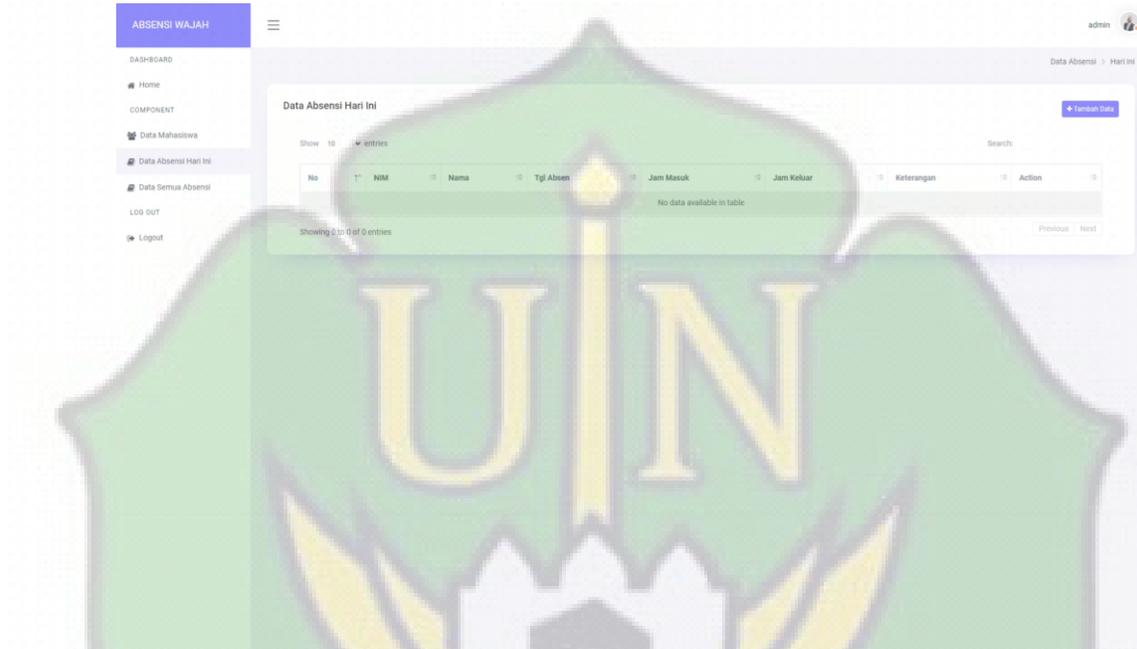
Tambah data wajah dapat diakses pada tab “*action*”, pada halaman ini mahasiswa dapat melakukan pendaftaran wajah pada kamera yang telah disediakan, kemudian sistem akan mengambil 5 (lima) *sample* wajah dari masing- masing mahasiswa untuk disimpan ke *database*.



Gambar 4.13 Tambah data wajah

d. Tampilan Data Absen Hari Ini

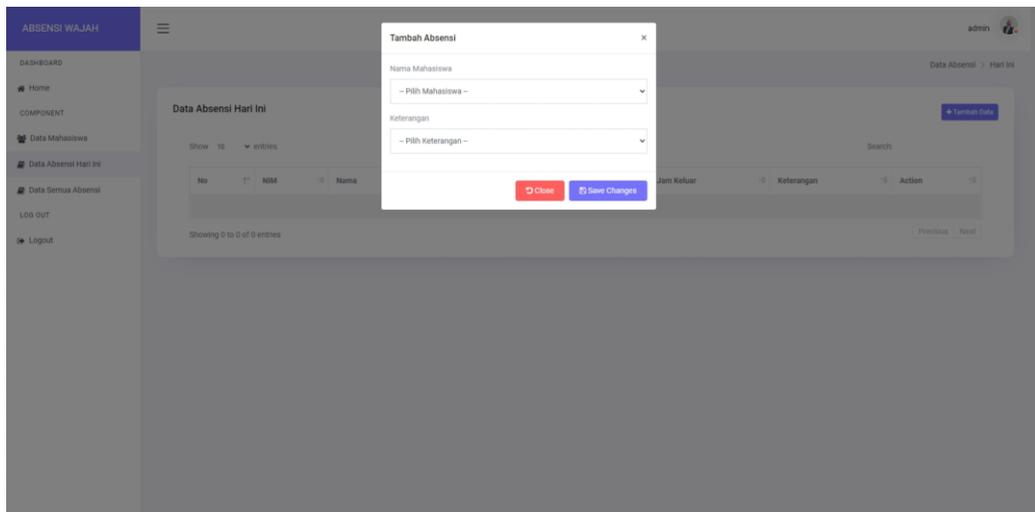
Halaman data absen hari ini akan menampilkan tabel yang berisi nama, nim, tanggal absen, absen masuk, absen keluar dan keterangan seluruh mahasiswa yang telah melakukan absensi harian.



Gambar 4.14 Halaman absen hari ini

- Tampilan Tambah Absensi

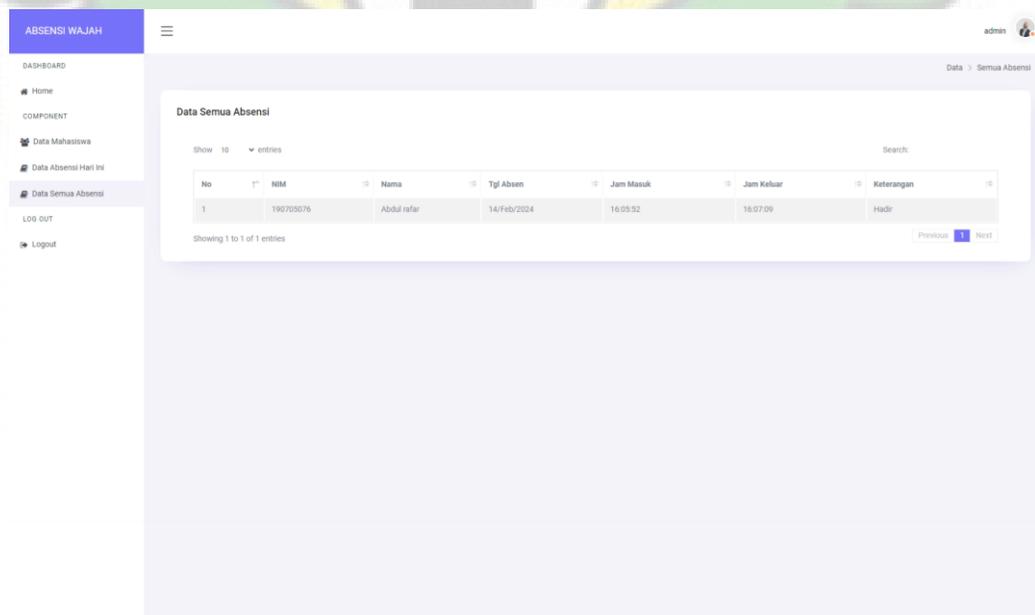
Tambah absensi dapat diakses pada tab “tambah data”, pada halaman ini admin dapat menambahkan absen manual apabila mahasiswa yang bersangkutan tidak dapat berhadir pada hari tersebut. Admin dapat mengisi keterangan hadir, izin, sakit, ataupun alpa.



Gambar 4.15 Tambah absensi

e. Tampilan Data Semua Absensi

Halaman data semua absensi akan menampilkan tabel yang berisi nama, nim, tanggal absen, absen masuk, absen keluar dan keterangan dari seluruh mahasiswa.



Gambar 4.16 Data semua absensi

4.2.2 Tampilan Halaman Mahasiswa

Tampilan halaman mahasiswa merupakan halaman yang hanya dapat diakses oleh mahasiswa.

- Tampilan Halaman Absensi Mahasiswa

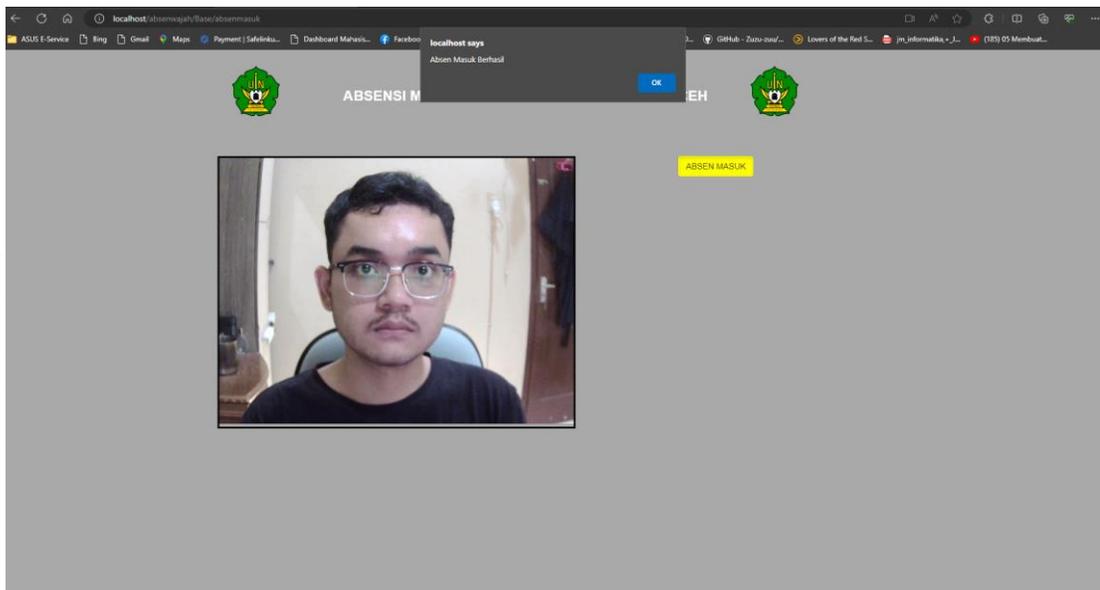
Halaman absensi mahasiswa merupakan halaman awal yang dapat diakses mahasiswa. Pada halaman ini akan menampilkan 2 (dua) pilihan *action* untuk absen masuk dan absen keluar. Serta pada halaman ini juga akan menampilkan tabel yang berisi nama, nim, tanggal absen, absen masuk, absen keluar dan keterangan seluruh mahasiswa yang telah melakukan absensi harian.



Gambar 4.17 Halaman absensi mahasiswa

- Tampilan Absen Masuk

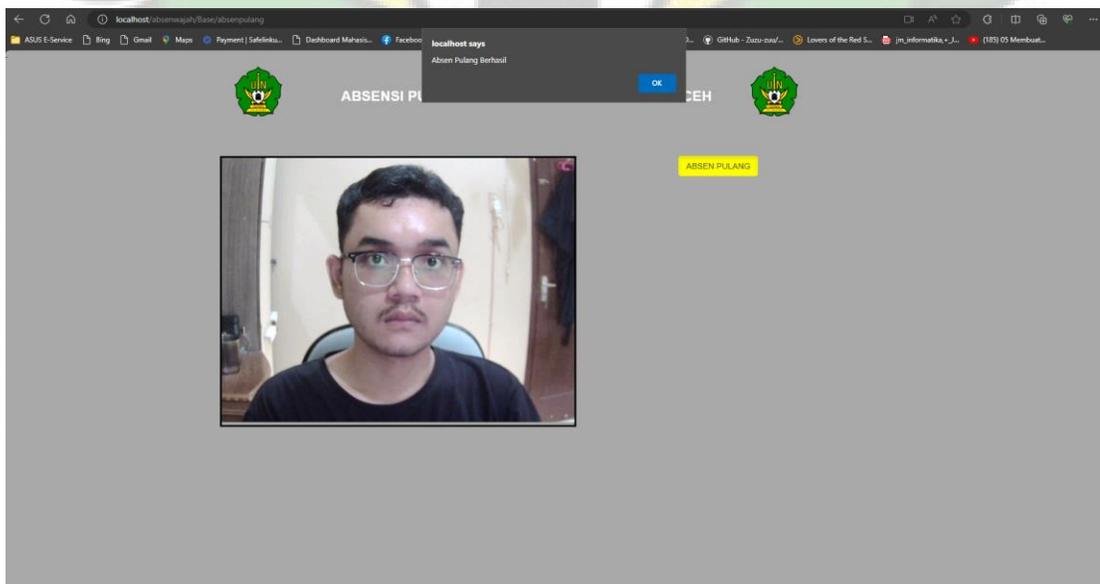
Tampilan absen masuk merupakan halaman yang dapat diakses melalui tab “absen masuk” pada halaman absensi mahasiswa. Pada halaman ini mahasiswa akan melakukan absensi pada sistem melalui kamera yang telah disediakan. Apabila berhasil terdeteksi maka akan muncul notifikasi absen masuk berhasil.



Gambar 4.18 Absen masuk

- Tampilan Absen Keluar

Tampilan absen keluar merupakan halaman yang dapat diakses melalui tab “absen keluar” pada halaman absensi mahasiswa. Pada halaman ini mahasiswa akan melakukan absensi untuk pulang. Apabila berhasil terdeteksi maka akan muncul notifikasi absen keluar berhasil.



Gambar 4.19 Absen keluar

4.3 Evaluasi Pengujian Sistem

Pengujian ini melibatkan beberapa bagian dari inti alur sistem presensi. Ini dimulai dengan admin mengelola data sistem dan kemudian dilanjutkan dengan tindakan siswa untuk berdiri di aplikasi web secara langsung. Maka sistem akan mengaktifkan kamera untuk mengambil *frame*, saat sistem dalam posisi *idle* akan selalu siap untuk pengenalan wajah mahasiswa. Saat *frame* dari kamera menunjukkan wajah dan jarak yang dideteksi sesuai dengan kriteria, hasil pengenalan wajah akan dilanjutkan. Jika wajah tidak terdeteksi, sistem akan kembali ke langkah awal pada *idle state*. Sistem akan mengidentifikasi klasifikasi wajah menggunakan *database* yang telah dibuat (Industri & Indonesia, 2023). Jika identitas mahasiswa dianggap sesuai dengan *database*, mereka dapat melakukan absensi. Jika tidak, sistem akan kembali ke *idle state*. Dapat dilihat pada Tabel 4.1 menunjukkan berbagai uji coba yang dilakukan untuk memastikan bahwa alur sistem sesuai dengan rancangan awal.

No	Skenario	Tindakan	Hasil	Keterangan
1	<i>Login</i>	Admin menggunakan <i>username</i> dan <i>password</i> untuk mengakses aplikasi <i>web</i> .	Berhasil	Admin telah memiliki <i>username</i> dan <i>password</i>
2	Registrasi Wajah	Mahasiswa mendaftarkan wajah ke aplikasi <i>web</i>	Berhasil	
3	Kelola <i>User</i>	Data mahasiswa yang akan hadir dapat ditambahkan dan diubah oleh admin.	Berhasil	
4	Kelola Absensi	Admin dapat menambahkan mahasiswa dan keterangan kehadiran	Berhasil	

5	Deteksi Wajah	Sistem mendeteksi wajah pada gambar siswa dengan Face-API.js	Berhasil	Ketika sistem mendeteksi wajah siswa, tombol "rekam wajah" akan muncul.
6	Pengenalan Wajah	Face-API.js akan mengidentifikasi gambar wajah yang ditangkap dan dideteksi dari foto.	Berhasil	Dipengaruhi oleh perangkat masing-masing, mungkin akan terjadinya <i>lag</i>
7	Melihat Status Presensi	Mahasiswa melihat status presensinya di halaman presensi	Berhasil	Hasil presensi juga akan menunjukkan status waktu presensi mahasiswa

Tabel 4.1 Test case sistem presensi

4.3.1 Hasil Pengujian Kompabilitas

Pengujian ini dilakukan oleh mahasiswa, dengan cara mencoba melakukan absensi menggunakan sistem Absensi *web* ini.

Proses	Keadaan	Keterangan
<i>System web</i>	<ul style="list-style-type: none"> Menggunakan <i>browser</i> berbeda. Menggunakan versi <i>browser</i> berbeda. Menggunakan Laptop / Komputer berbeda. Menggunakan perangkat <i>webcam</i> berbeda. 	Dengan menggunakan versi browser terbaru, perangkat komputer yang mumpuni, dan <i>webcam</i> yang bagus akan mempercepat proses jalannya sistem.
Absensi	Pencahayaan : <ul style="list-style-type: none"> Terang Gelap Remang - remang 	Proses deteksi sangat dipengaruhi oleh pencahayaan. wajah, pencahayaan yang lebih baik meningkatkan kemampuan sistem untuk mendeteksi wajah.

Absensi	Jarak wajah dengan webcam : <ul style="list-style-type: none"> • 1 meter • 2 meter 	Jika jarak antara wajah dan <i>webcam</i> semakin dekat, maka sistem dapat mendeteksi wajah dengan lebih cepat.
Absensi	Penggunaan Aksesoris : <ul style="list-style-type: none"> • Kacamata • Topi • Masker 	Penggunaan topi dan kacamata tidak mempengaruhi proses deteksi wajah, tetapi menggunakan masker dapat mengganggu proses deteksi karena aksesoris tersebut menutupi mulut dan hidung, membuat sistem tidak dapat melihat wajah.
Absensi	Melakukan Absensi secara bersamaan.	Dengan melakukan absensi secara bersamaan sistem akan membaca sekaligus mahasiswa yang sedang melakukan absensi.

Tabel 4.2 Hasil pengujian kompatibilitas

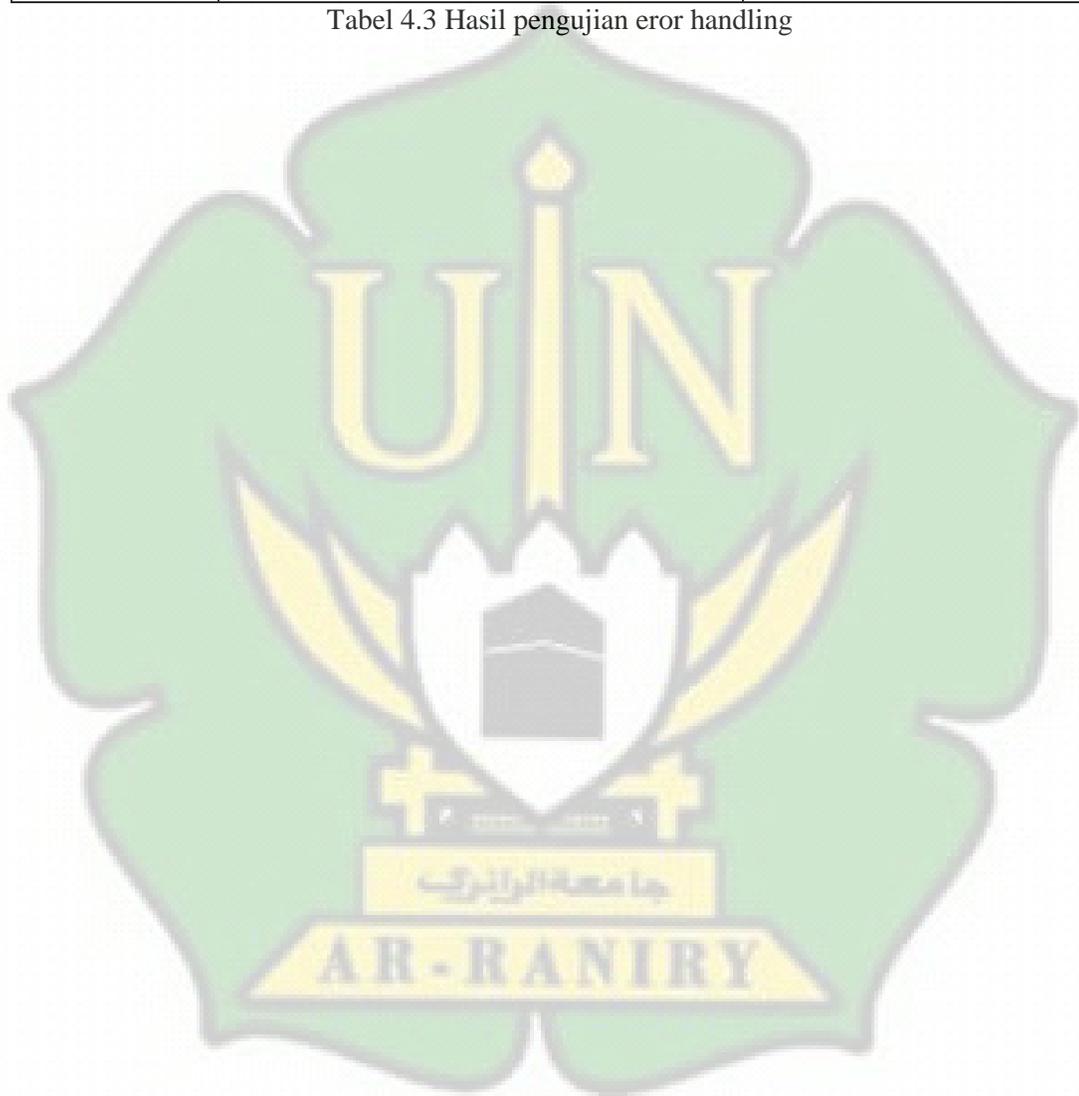
4.3.2 Hasil Pengujian *Error Handling*

Pengujian ini dilakukan untuk melihat keadaan sistem saat mengalami masalah ketika dijalankan.

Proses	Keadaan	Keterangan
Absensi	Dengan melakukan absensi secara bersamaan sistem akan membaca sekaligus mahasiswa yang sedang melakukan absensi.	Pegelola harus mendapatkan perangkat keras baru jika perangkat komputer rusak.
Absensi	Adanya keadaan atau kondisi yang mengubah wajah mahasiswa, membuat sistem tidak dapat	Anda dapat melakukan absen secara manual Admin dapat melakukan absensi secara manual

	mengidentifikasi gambar yang diambil.	kepada mahasiswa tersebut.
<i>Login</i>	<ul style="list-style-type: none"> • Lupa <i>username</i> • Lupa <i>password</i> 	Mengganti <i>password</i> atau <i>username</i> dapat dilakukan dengan cara mengakses database oleh developer,

Tabel 4.3 Hasil pengujian eror handling



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dari perancangan sistem absensi berbasis *website* ini, maka didapatkan kesimpulan yaitu, hasil penelitian menunjukkan bahwa metode pengenalan wajah menggunakan *FaceMatcher* memiliki akurasi penuh seratus persen. Ini berarti sistem dapat mengenali wajah mahasiswa dengan sangat tepat dan mengurangi risiko kecurangan dalam pencatatan kehadiran. Namun, metode ini membutuhkan waktu yang lebih lama untuk diklasifikasikan karena keterbatasan kecepatan dari perangkat, yang berarti lebih banyak data yang akan dimuat.

5.2 Saran

Dari hasil Implementasi Metode *FaceMatcher* pada pembuatan sistem absensi berbasis *website* menggunakan *Framework codeigniter*, maka saran yang dapat diberikan yaitu citra wajah yang diambil tidak hanya dari depan saja dengan mendeteksi bagian mata, mulut, dan hidung. Diharapkan penelitian lanjutan dapat membuat sistem mampu untuk mendeteksi objek 3D agar sistem dapat membedakan gambar atau foto dengan wajah asli. Deteksi objek 3D memungkinkan sistem untuk mengenali perbedaan antara wajah manusia dengan gambar atau patung wajah palsu

DAFTAR PUSTAKA

- Amanda Widya Pramesti. (2020). *PERANCANGAN SISTEM INFORMASI ABSENSI SISWA BERBASIS WEB PADA SMA ADITYA KARYA KABUPATEN TANGERANG*. <https://Widuri.Raharja.Info/>.
<https://widuri.raharja.info/index.php?title=SI1511489063>
- Amari, R. O. (2023). *IMPLEMENTASI PENGENALAN WAJAH DENGAN METODE EUCLIDEAN DISTANCE PADA SISTEM LOGIN DAN ABSENSI*. 31–41.
- Ari Nur Rokhman. (2020). Rancangan Bangun Aplikasi Absensi Karyawan Menggunakan Location Base Service (LBS) Berbasis Android. *Sustainability (Switzerland)*, 14(2), 1–4. <http://www.unpcdc.org/media/15782/sustainable-procurement-practice.pdf>
<https://europa.eu/capacity4dev/unep/document/briefing-note-sustainable-public-procurement>
<http://www.hpw.qld.gov.au/SiteCollectionDocuments/ProcurementGuideIntegratingSustainability>
- Budi. (2021). *Database Absensi Siswa*. Berkasbelajar.Github.Io.
<https://berkasbelajar.github.io/contoh-database-absensi-siswa/>
- Christyanto, N. E., Muhammad, E., Jonemaro, A., & Yudistira, N. (2022). *Pengembangan Aplikasi Android Presensi Kehadiran Realtime menggunakan Pengenalan Wajah dengan Model Facenet* (Vol. 6, Issue 10). <http://j-ptiik.ub.ac.id>
- Collins, C. (2023). *Face Matching: How It Works and What the Output Means*. Medium.Com. <https://mr-collins-llb.medium.com/face-matching-how-it-works-and-what-the-output-means-90a4cb87832d>
- Edlis, ummu fadhella. (2009). *Biometrik, Sistem Biometrik*.
- Face Recognition Attendance System | Shailersolution*. (n.d.).

<https://www.shailersolutions.com/face-recognition-attendance-system/>

fauzi 2017. (2018). Karakteristik Sistem. *Journal of Chemical Information and Modeling*, 53(9), 8–24.

Hanadhito Riswantoro. (2019). Perancangan Prosedur Pengeluaran Kas Pada Mini Market Syar'e Mart. *Perancangan Sistem Informasi Terpadu Pemerintah Daerah Kabupaten Paser*, 53(9), 1689–1699.

https://www.google.com/url?sa=t&source=web&rct=j&url=https://dspace.uui.ac.id/bitstream/handle/123456789/10087/04.2%2520BAB%25202.pdf%3Fsequence%3D5%26isAllowed%3Dy&ved=2ahUKEwjehMmkypv4AhXk1TgGHa_XB0YQFnoECAUQAQ&usg=AOvVaw1Iu90bd51U2wDzhgnhR08w

Hartiwi, Y., Rasywir, E., Pratama, Y., & Jusia, P. A. (2020). Sistem Manajemen Absensi dengan Fitur Pengenalan Wajah dan GPS Menggunakan YOLO pada Platform Android. *Jurnal Media Informatika Budidarma*, 4, 1235–1242. <https://doi.org/10.30865/mib.v4i4.2522>

ichi.pro. (2020). *Menerapkan Sistem Pengenalan Wajah dengan face-api.js — Bagian II*. ICHI.PRO. <https://ichi.pro/id/menerapkan-sistem-pengenalan-wajah-dengan-face-api-js-bagian-ii-79527285810375>

Industri, F. T., & Indonesia, U. I. (2023). *DETEKSI WAJAH PADA CITRA DIGITAL*.

Jannah, dkk. (2023). Implementasi Web Presensi Karyawan Industri Kertas Jaya Menggunakan Metode Convolutional Neural Network. *Ejournal.Unesa.Ac.Id*, 1–12. <https://ejournal.unesa.ac.id/index.php/jurnal-manajemen-informatika/article/view/56899%0Ahttps://ejournal.unesa.ac.id/index.php/jurnal-manajemen-informatika/article/view/56899/44641>

Jantce TJ Sitinjak, D. D., Maman, ., & Suwita, J. (2020). Analisa Dan Perancangan Sistem Informasi Administrasi Kursus Bahasa Inggris Pada Intensive English Course Di Ciledug Tangerang. *Insan Pembangunan Sistem Informasi Dan Komputer (IPSIKOM)*, 8(1). <https://doi.org/10.58217/ipsikom.v8i1.164>

- Nym. (2022). *Use Case Diagram for Face Recognition Attendance System*. Itsourcecode. <https://itsourcecode.com/uml/face-recognition-attendance-system-use-case-diagram/>
- Rahmatil Ula. (2002). *RANCANG BANGUN SISTEM INFORMASI ABSENSI SISWA BERBASIS WEB DI SMKS MAHYAL ULUM AL-AZIZIYAH*.
- Ridho. (2018). Sistem Informasi. *Gender and Development*, 120(1), 0–22.
http://www.uib.no/sites/w3.uib.no/files/attachments/1._ahmed-affective_economies_0.pdf<http://www.laviedesidees.fr/Vers-une-anthropologie-critique.html>[http://www.cairn.info](http://www.cairn.info.lama.univ-amu.fr/resume.php?ID_ARTICLE=CEA_202_0563%5Cnhttp://www.cairn.info).
- Rizky, M. (2019). *UML Diagram : Activity Diagram*. <https://Socs.Binus.Ac.Id>.
<https://socs.binus.ac.id/2019/11/22/uml-diagram-activity-diagram/>
- Singh, N. (2024). *MobileNet V1 Architecture*. Iq.Opengenus.Org.
<https://iq.opengenus.org/mobilenet-v1-architecture/>
- Sugiyono. (2019). *Yeni Marchna Sari Purba, 2021 IMPLEMENTASI PROGRAM DIKLAT BERJENJANG TINGKAT DASAR DALAM JARINGAN UNTUK MENINGKATKAN KOMPETENSI PENDIDIK PAUD Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu*. 23.
- Sutanta. (2011). *Komponen sistem informasi*.
- Suthar, H. (2023). *Comparing FaceMatcher and euclideanDistance Methods in FaceAPI.js for Face Recognition*. Medium.Com.
<https://hellohimesh.medium.com/comparing-facematcher-and-euclidean-distance-methods-in-faceapi-js-for-face-recognition-d1533cd0ee39#:~:text=What is FaceMatcher%3F The FaceMatcher method in FaceAPI.js,the best match for a given face descriptor.>
- Wayan Wardani, N., & Yoka Fathoni, M. (n.d.). *Perancangan Absensi Berbasis Face Recognition Pada Desa Sokaraja Lor Menggunakan Platform Android*.

<http://jurnal.mdp.ac.id>



LAMPIRAN

```
<!DOCTYPE html>
<html>
<head>
  <title> <?php echo $title ?> </title>

  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min
.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js">
</script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/js/bootstrap.min.js"></script>-
  <script src="<?php echo base_url('assets/faceapi/face-api.js')
?>"></script>
</head>

<body style="background-color:#A9A9A9 ;">

  <div id="parent1">
    <div class="container">

      <div class="row">
        <div class="col-sm-2 col-12">
          
        </div>
        <div class="col-sm-7 col-12">
          <h3 style="padding-top: 25px;color:
white;"><center><b>REKAM WAJAH MAHASISWA UIN AR-RANIRY
ACEH</b></center></h3>
        </div>
        <div class="col-sm-2 col-12">
          
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```

```

</div>
<br/><br/><br/>
<div class="row">
    <div class="col-sm-7 col-12">
        <div class="margin" style="position: relative;
float:left; border: 3px solid black;">
            <video id="vidDisplay" style="height: 450px; width:
600px; display: inline-block; vertical-align: baseline;"
onloadedmetadata="onPlay(this)" autoplay="true"></video>
            <canvas id="overlay" style="position: absolute;
top: 0; left: 0;" width = "600" height = "450"/>
        </div>
    </div>
    <div class="col-sm-5 col-12">
        <center>
            <div id="parent2" style="float:left;">
                <button id="register" class="btn btn-light"> DAFTAR
WAJAH </button>
                <br><br>
                <span style="font-size: 25px; font-family:Lucida
Console,Monaco,monospace; font-weight: bold;color: white;"> Wajah
Mahasiswa </span><br></br>
                <img id = "prof_img" style="height:200px; width:
200px; border: 3px solid black; border-radius: 10px;" ></img><br><br>

                <div id="reg_disp" style="display: none;">
                    <input id = "id" type="hidden" value="<?php echo
$mahasiswa->id ?>">
                    <input id = "lname" type="hidden" value="<?php
echo $mahasiswa->nim ?>">

                    <input type="text" style="width:500px; height:
30px; border-radius: 5px; border:1px solid black;text-align: center;"
value="<?php echo $mahasiswa->nim.'-'. $mahasiswa-
>nama_mahasiswa ?>" readonly>
                    <br/><br/>
                    <button id="capture" class="btn btn-block btn-
primary"> Simpan Wajah</button><br><br>
                    <div id="tries" style="font-size: 23px; font-
family:Lucida Console,Monaco,monospace; font-weight: bold;color:
white;">Pengenalan Wajah : </div>
                </div>
            </div>

```

```

        <div id="log_disp">
            <div id="logname" style="font-size: 35px; font-
weight: bold; width: 570px; white-space: pre-wrap; text-align:
center;"></div><br>
                <div style="width: 200px; border: 3px solid
black;"></div><br>
            </div>
        </div>
    </center>
</div>
</div>
</div>
</div>
</div>
</body>

<script>
    var waitingDialog = waitingDialog || (function ($) {
        var $dialog = $(
            '<div class="modal fade" data-backdrop="static" data-
keyboard="false" tabindex="-1" role="dialog" aria-hidden="true"
style="padding-top:15%; overflow-y:visible;">' +
                '<div class="modal-dialog modal-m">' +
                    '<div class="modal-content">' +
                        '<div class="modal-header"><h3 style="margin:0;"></h3></div>'
+
                    '<div class="modal-body">' +
                        '<div class="progress progress-striped active"
style="margin-bottom:0;"><div class="progress-bar" style="width:
100%"></div></div>' +
                        '</div>' +
                    '</div></div></div>');

        return {
            show: function (message, options) {
                if (typeof options === 'undefined') {
                    options = {};
                }
                if (typeof message === 'undefined') {
                    message = 'Loading';
                }
                var settings = $.extend({

```

```

        dialogSize: 'm',
        progressType: '',
        onHide: null
    }, options);
    $dialog.find('.modal-dialog').attr('class', 'modal-
dialog').addClass('modal-' + settings.dialogSize);
    $dialog.find('.progress-bar').attr('class', 'progress-bar');
    if (settings.progressType) {
        $dialog.find('.progress-bar').addClass('progress-bar-' +
settings.progressType);
    }
    $dialog.find('h3').text(message);
    if (typeof settings.onHide === 'function') {
        $dialog.off('hidden.bs.modal').on('hidden.bs.modal', function
(e) {
            settings.onHide.call($dialog);
        });
    }
    $dialog.modal();
},
hide: function () {
    $dialog.modal('hide');
}
};
})(jQuery);
</script>

<script>

//-----GLOBAL VARIABLE FOR FACE MATCHER-----
-----
var faceMatcher = undefined
//-----
-----

waitingDialog.show('Initializing data...', {dialogSize: 'sm',
progressType: 'success'})
$("#parent1").hide();
$("#parent2").hide();
Promise.all([

```

```

        faceapi.nets.faceRecognitionNet.loadFromUri("<?php echo
base_url('models') ?>"),
        faceapi.nets.faceLandmark68Net.loadFromUri("<?php echo
base_url('models') ?>"),
        faceapi.nets.ssdMobilenetv1.loadFromUri("<?php echo
base_url('models') ?>"),
        faceapi.nets.tinyFaceDetector.loadFromUri("<?php echo
base_url('models') ?>")
    ]).then(start)

    async function start() {
        $.ajax({
            datatype: 'json',
            url: "<?php echo site_url('Mahasiswa/tampilwajah') ?>",
            data: ""
        }).done(async function(data) {
            if(data.length > 2){
                var json_str = "{\"parent\":\"" + data + "\""
                content = JSON.parse(json_str)
                for (var x = 0; x < Object.keys(content.parent).length; x++)
            {
                for (var y = 0; y <
Object.keys(content.parent[x]._descriptors).length; y++) {
                    var results =
Object.values(content.parent[x]._descriptors[y])
                    content.parent[x]._descriptors[y] = new
Float32Array(results)
                }
            }
            faceMatcher = await createFaceMatcher(content);
        }
        waitingDialog.hide()
        $('#parent1').show()
        $('#parent2').show()
        run();
    });
}

// Create Face Matcher
async function createFaceMatcher(data) {
    const labeledFaceDescriptors = await
Promise.all(data.parent.map(className => {
        const descriptors = [];

```

```

    for (var i = 0; i < className._descriptors.length; i++) {
      descriptors.push(className._descriptors[i]);
    }
    return new faceapi.LabeledFaceDescriptors(className._label,
descriptors);
  )))
  return new faceapi.FaceMatcher(labeledFaceDescriptors,0.6);
}

async function onPlay() {
  const videoEl = $('#vidDisplay').get(0)
  if(videoEl.paused || videoEl.ended )
    return setTimeout(() => onPlay())

  $("#overlay").show()
  const canvas = $('#overlay').get(0)

  if($("#register").hasClass('active'))
  {
    const options = getFaceDetectorOptions()
    const result = await faceapi.detectSingleFace(videoEl,
options)
    if (result) {
      const dims = faceapi.matchDimensions(canvas, videoEl, true)
      dims.height = 450
      dims.width = 600
      canvas.height = 450
      canvas.width = 600
      const resizedResult = faceapi.resizeResults(result, dims)
      faceapi.draw.drawDetections(canvas, resizedResult)
    }
    else{
      $("#overlay").hide()
    }
  }

  if($("#login").hasClass('active'))
  {
    if(faceMatcher != undefined){
      //-----FACE RECOGNIZE-----
-
      const input = document.getElementById('vidDisplay')

```

```

        const displaySize = { width: 600, height: 450 }
        faceapi.matchDimensions(canvas, displaySize)
        const detections = await
faceapi.detectAllFaces(input).withFaceLandmarks().withFaceDescriptors()
        const resizedDetections = faceapi.resizeResults(detections,
displaySize)
        const results = resizedDetections.map(d =>
faceMatcher.findBestMatch(d.descriptor))
        results.forEach((result, i) => {
            const box = resizedDetections[i].detection.box
            const drawBox = new faceapi.draw.DrawBox(box, { label:
result.toString() })
            drawBox.draw(canvas)
            var str = result.toString()
            rating = parseFloat(str.substring(str.indexOf('(') +
1, str.indexOf(')')))
            str = str.substring(0, str.indexOf('('))
            str = str.substring(0, str.length - 1)
            if(str != "unknown"){
                if(rating < 0.5){
                    if(str == $("#log_name").text()){
                        console.log("Match TRUE!")
                        match = true;
                        //$("#logname").html(str)
                        //$("#prof_img").attr('src',"http://localho
st/data/" + str + "/image0.png")
                    }
                }
            }
        })
        //-----
    }
}

setTimeout(() => onPlay())
}

async function run() {
    const stream = await navigator.mediaDevices.getUserMedia({ video:
{} })
    const videoEl = $('#vidDisplay').get(0)
    videoEl.srcObject = stream

```

```

}

// tiny_face_detector options
let inputSize = 160
let scoreThreshold = 0.4

function getFaceDetectorOptions() {
    return new faceapi.TinyFaceDetectorOptions({ inputSize,
scoreThreshold });
}

async function load_neural(){
    waitingDialog.show('Initializing neural data....', {dialogSize:
'sm', progressType: 'success'})
    $.ajax({
        datatype: 'json',
        url: "<?php echo site_url('Mahasiswa/tampilwajah') ?>",
        data: ""
    }).done(async function(data) {
        if(data.length > 2){
            var json_str = "{\"parent\":\"" + data + "\""
            content = JSON.parse(json_str)
            console.log(content)
            for (var x = 0; x < Object.keys(content.parent).length; x++)
            {
                for (var y = 0; y <
Object.keys(content.parent[x]._descriptors).length; y++) {
                    var results =
Object.values(content.parent[x]._descriptors[y]);
                    content.parent[x]._descriptors[y] = new
Float32Array(results);
                }
            }
            faceMatcher = await createFaceMatcher(content);
        }
        waitingDialog.hide()
    });
}
</script>
<script>

```

```

$(document).ready(async function(){

    var counter = 5;
    const descriptions = [];

    // -----Initialize-----
    // $("#login").css('background-color','yellow');
    // $("#login").addClass('active');
    $("#register").css('background-color','white');
    $("#register").addClass('active');

    if($("#login").hasClass('active')){
        $("#reg_disp").hide();
        $("#log_disp").show();
    }
    else if($("#register").hasClass('active')){
        $("#reg_disp").show();
        $("#log_disp").hide();
    }
    //-----

    $("#login").click(function(){
        $.ajax({
            datatype: 'json',
            url: "<?php echo site_url('Mahasiswa/tampilwajah') ?>",
            data: ""
        }).done(function(data) {
            labeled = JSON.parse(data)
        });
        $(this).css('background-color','yellow')
        $("#register").css('background-color','yellow')
        $(this).addClass('active')
        $("#register").removeClass('active')
        $("#reg_disp").hide()
        $("#log_disp").show()
        $("#prof_img").removeAttr('src')
        $("#lname").val()
        $("#logname").html('')
        counter = 5
        description = []
        $("#tries").html("Pengenalan Wajah : " + counter)
    });

```

```

$("#register").click(function(){
    $(this).css('background-color','yellow')
    $("#login").css('background-color','white')
    $(this).addClass('active')
    $("#login").removeClass('active')
    $("#reg_disp").show()
    $("#log_disp").hide()
    $("#prof_img").removeAttr('src')
    $("#lname").val()
    $("#logname").html('')
    counter = 5
    description = []
    $("#tries").html("Pengenalan Wajah : " + counter)
});

$("#tries").html("Pengenalan Wajah : " + counter)

$("#capture").click(async function(){
    var data = $("#lname").val();
    const label = data;

    if($("#lname").val()){
        if($("#register").hasClass('active')){

            if(counter <= 5 && counter >= 0 ){
                var canvas = document.createElement('canvas');
                var context = canvas.getContext('2d');
                var video = document.getElementById('vidDisplay');
                context.drawImage(video, 0, 0, 600, 350);
                var capURL = canvas.toDataURL('image/png');
                var canvas2 = document.createElement('canvas');
                canvas2.width = 1200;
                canvas2.height = 800;
                var ctx = canvas2.getContext('2d');
                ctx.drawImage(video, 0, 0, 1200, 800);
                var new_image_url = canvas2.toDataURL();
                var img = document.createElement('img');
                img.src = new_image_url;
                document.getElementById("prof_img").src = img.src;

                const detections = await
                faceapi.detectSingleFace(img).withFaceLandmarks().withFaceDescriptor();

```

```

        if( detections != null){
            descriptions.push(detections.descriptor);
            var descrip = descriptions;
            counter--;
            $("#tries").html("Pengenalan Wajah : " + counter)
            if(counter == 0){
                // Save Image
                $.ajax({
                    type: "POST",
                    url: "<?php echo site_url('Mahasiswa/simpanwajah')
?>",
                    data: {image: img.src ,path: data, id:
$("#id").val()}
                }).done(function(o) {
                    console.log('Image Saved');
                });

                waitingDialog.show('Processing data.....',
{dialogSize: 'sm', progressType: 'success'})
                var postData = new faceapi.LabeledFaceDescriptors(label,
descrip);
                $.ajax({
                    url: '<?php echo site_url('Mahasiswa/simpanjson')
?>',
                    type: 'POST',
                    data: { myData: JSON.stringify(postData) },
                    datatype: 'json'
                })
                .done(async function (data) {
                    load_neural()
                    alert("Done!")
                    console.log("Success!")
                    waitingDialog.hide()
                    counter = 5
                    $("#tries").html("Pengenalan Wajah : " + counter)
                    $("#lname").val()
                    $("#prof_img").removeAttr('src')
                    window.close();
                    window.location.href = '<?php echo
site_url('Mahasiswa') ?>';
                })
                .fail(function (jqXHR, textStatus, errorThrown) {

```

```
        alert("Error due to internet connection! Please try  
again!");  
    });  
    const descriptions = [];  
    }  
    }  
    else{  
        alert("No FACE detected!");  
    }  
    }  
    else{  
        alert("Done Learning!");  
        counter = 5;  
        const descriptions = [];  
    }  
    }  
    }  
    });  
});  
</script>  
  
</html>
```

