

Rancang Bangun Aplikasi Absensi Online Berbasis Android Menggunakan Metode Deep Learning Pada PT. Pelabuhan Indonesia III (Persero)

Ardian Permana Putera¹, Puteri Noraisya Primandari, S.ST., M.IM²

Teknik Informatika, Universitas 17 Agustus 1945 Surabaya
Jl. Semolowaru No.45, Menur Pumpungan, Kec. Sukolilo, Surabaya

¹ardianpermana.ptr@gmail.com

²puterinoraisya@untag-sby.ac.id

Abstract

The development of Good Corporate Governance (GCG) has a great impact on the company, one of the important things that must be implemented by the company is the management of attendance and good management of employee work hours so that system clarity and accountability of company organs can be managed correctly. At PT Pelabuhan Indonesia III (Persero), attendance is done using a finger print or eye scan machine, which in its use often causes long queues in the morning, because employees must take turns to make attendance, not to mention the risk of losing data if the machine is damaged. A system that is appropriate, safe and flexible is needed to change the company's needs. Deep Learning is a Neural Network that works by imitating the performance of the human brain. In this study, a mobile android based attendance system is designed that is capable of capturing detected face images for face verification using Deep Learning and provides location limits for attendance only at work units using Geofencing. This system is also built using the React Native Framework, the strength of this framework is that it can be used for Android and iOS (cross-platform app) and also although it is based on Javascript, the output of this framework produces native applications.

Keywords: Deep Learning, Attendance, Face Recognition, GCG, React Native, android

Abstrak

Perkembangan Good Corporate Governance (GCG) memberikan dampak pengaruh yang besar bagi perusahaan, salah satu hal penting yang harus diimplementasikan oleh perusahaan adalah pengelolaan absensi serta manajemen jam kerja karyawan yang baik agar kejelasan sistem dan pertanggungjawaban organ perusahaan dapat terkelola dengan benar. Pada PT Pelabuhan Indonesia III (Persero), absensi dilakukan menggunakan mesin finger print, dimana dalam penggunaannya seringkali menyebabkan antrian panjang, karena karyawan harus secara bergantian untuk melakukan absensi, belum lagi resiko kehilangan data apabila mesin rusak. Dibutuhkan sistem yang flexibel, tepat guna, dan aman terhadap perubahan kebutuhan perusahaan. Deep Learning merupakan suatu metode pembelajaran yang dilakukan mesin untuk dapat memahami dan mengklasifikasi suatu objek, yakni utamanya dalam aplikasi yang dibangun ini adalah wajah yang ditangkap dalam bentuk citra. Deep Learning adalah Neural Network yang bekerja dengan cara menirukan kinerja otak manusia. Pada penelitian ini, dirancang sistem absensi berbasis mobile android yang mampu melakukan pengambilan citra wajah yang terdeteksi untuk verifikasi wajah menggunakan Deep Learning dan memberikan batasan lokasi untuk absensi hanya pada unit kerja dengan menggunakan Geofencing. Sistem ini juga dibangun menggunakan Framework React Native, kelebihan framework ini adalah dapat digunakan untuk android dan iOS (cross-platform app) dan juga walaupun berbasis Javascript, luaran dari framework ini menghasilkan aplikasi native

Kata Kunci: Deep Learning, Absensi, Verifikasi Wajah, GCG, React Native, android

1. PENDAHULUAN

Perkembangan *Good Corporate Governance* (GCG) memberikan dampak pengaruh yang besar bagi perusahaan, salah satu hal penting yang harus diimplementasikan oleh perusahaan adalah pengelolaan absensi serta manajemen jam kerja karyawan yang baik agar kejelasan sistem dan pertanggungjawaban organ perusahaan dapat dikelola dengan benar

Pada PT Pelabuhan Indonesia III (Persero), absensi dilakukan menggunakan mesin *finger print* atau scan mata, dimana dalam penggunaannya seringkali menyebabkan antrian panjang di pagi hari, karena karyawan harus secara bergantian untuk melakukan absensi, belum lagi resiko kehilangan data apabila mesin rusak saat data absensi belum tersinkron ke database perusahaan. Dibutuhkan sistem yang flexibel, tepat guna, dan aman terhadap perubahan kebutuhan perusahaan.

Deep Learning merupakan suatu metode pembelajaran yang dilakukan mesin untuk dapat memahami dan mengklasifikasi suatu objek, yakni utamanya dalam aplikasi yang dibangun ini adalah wajah yang ditangkap dalam bentuk citra. *Deep Learning* adalah *Neural Network* yang bekerja dengan cara menirukan kinerja otak manusia. Penelitian ini akan memanfaatkan library yang mendukung komputasi dalam *Deep Learning*, yaitu *face-api.js* yang dikembangkan oleh Vincent Mühler yang dapat berjalan diatas library *TensorFlow*. *TensorFlow* merupakan library yang dikembangkan oleh Google untuk kepentingan penelitian dan pengembangan dalam lingkup *Machine Learning* khususnya *Deep Learning* dan *Artificial Intellegent*.

Pada penelitian ini, dirancang sistem absensi berbasis mobile android yang mampu melakukan pengambilan citra wajah yang terdeteksi untuk verifikasi wajah menggunakan *Deep Learning* fungsi ini memastikan karyawan yang absen adalah sesuai dan tidak diwakilkan orang lain, sehingga dapat memastikan validitas data absensi serta memberikan batasan lokasi untuk absensi hanya pada unit kerja karyawan dengan menggunakan *Geofencing*. Sistem ini juga dibangun menggunakan *Framework React Native*, kelebihan *framework* ini adalah dapat digunakan untuk android dan *iOS* (*cross-platform app*) dan juga walaupun berbasis *Javascript*, luaran dari *framework* ini menghasilkan aplikasi native, sedangkan untuk *Rest Api Face Recognition* yang dibangun menggunakan *Node Js*.

2. METODE PENELITIAN

2.1 Proses Pengambilan Data Penelitian



Gambar 1 Design Penelitian

Proses penelitian baik mulai pengambilan data, perancangan, pembangunan dan pengujian hingga dokumentasi aplikasi dapat dilihat melalui gambar 1, detail tahapan penelitian adalah sebagai berikut:

1. Studi Literatur
Melakukan pencarian teori pendukung penelitian sebagai bahan kajian untuk merancang Aplikasi absensi Online Berbasis Android dengan *Face Recognition*
2. Data Penelitian
Data dari studi literatur akan dilakukan analisa sebagai bahan pertimbangan guna melakukan perancangan dan pengembangan aplikasi
3. Perancangan Aplikasi absensi Online Berbasis Android dengan *Face Recognition*
Merancang sistem Aplikasi absensi Online Berbasis Android berdasarkan data yang telah didapat.
4. Pembangunan Aplikasi
Melakukan pengembangan dari perancangan Aplikasi Absensi Online berbasis Android menggunakan metode *Deep Learning*
5. Pengujian Aplikasi
Tahap selanjutnya setelah pembangunan aplikasi selesai dilakukan, sistem akan dilakukan pengujian sebagai parameter kelayakan aplikasi
6. Dokumentasi Aplikasi
Melakukan dokumentasi baik mulai tahap perancangan hingga selesai pembangunan terhadap aplikasi yang telah dibuat

2.2 Perancangan Proses

2.2.1 Perancangan Data

Perancangan data dari aplikasi MyIntan dibuat untuk memberi penjelasan berkaitan dengan data yang dipakai dan dihasilkan oleh aplikasi ini. Perancangan data yang dilakukan meliputi penentuan jenis file yang digunakan, ekstensi file yang dipakai, ukuran file, serta lokasi penyimpanan dari file yang dipakai dan dihasilkan oleh aplikasi ini.

Jenis data input yang digunakan oleh aplikasi adalah gambar atau *image* berekstensi .JPG. Jenis data ini digunakan sebagai inputan yang akan dikirim ke *Rest Api Face Registration Service*, setelah data gambar tersebut dikirim, *Rest Api Face Registration Service* akan memproses data tersebut dan akan mengekstrak *Face Descriptor* dalam bentuk *Feature Vector* yang memiliki 128 titik *vector* wajah, kemudian data array *feature vector* wajah ini akan disimpan lagi dalam file berbentuk *Json Object (.json)* beserta *label* nama user, file inilah yang digunakan sebagai dataset wajah pada penggunaan kasus mendaftarkan dataset wajah.

Penyimpanan dataset wajah MyIntan akan disimpan di penyimpanan server *Face Registration Service*, hal ini dimungkinkan karena dataset wajah yang tersimpan berbentuk *array Feature Vector* dan disimpan dalam file *Json Object (.json)* ini mempunyai ukuran sangat kecil < 200 kb untuk asumsi penyimpanan 40 dataset wajah dalam satu file *Json Object (.json)* per user seperti pada gambar 2. Data file *Json Object (.json)* ini akan disimpan dalam sub folder uploads, nama folder tersebut sesuai dengan nama dari user login.

```

1  |
2  |
3  |
4  |
5  |
6  |
7  |
8  |
9  |
10 |
11 |
12 |
13 |
14 |
15 |
16 |
17 |
18 |
19 |
20 |
21 |
22 |
23 |
24 |
25 |
26 |
27 |
28 |
29 |
30 |
31 |
32 |
33 |
34 |
35 |
36 |
37 |
38 |
39 |
40 |
41 |
42 |
43 |

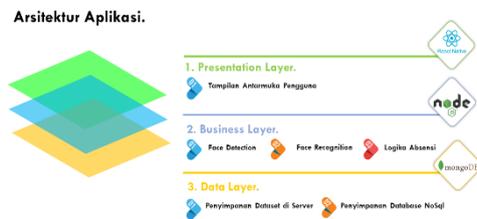
```

Gambar 2 Isi Dataset wajah file .json untuk user ardian

2.2.2 Perancangan Arsitektur Perangkat Lunak

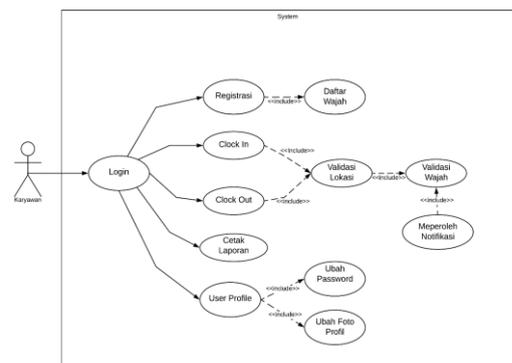
Arsitektur yang dibangun menggunakan arsitektur 3 lapis, yaitu *Data Layer*, *Business Layer*, dan *Presentation Layer*. Lapisan *Data* bertugas untuk menangani *database* aplikasi serta penyimpanan file dataset wajah user. Basis data yang digunakan adalah MongoDB yang berjalan di cloud server MongoDB Atlas, sedangkan Lapisan *Business* bertugas menjembatani logika yang diproses aplikasi termasuk proses absensi karyawan. Untuk Lapisan *Presentation* bertugas menangani *interface* pengguna. Data Layer untuk dataset wajah akan ditangani oleh penyimpanan di dalam server yang diatur oleh file system dari sistem operasi Windows di server. Gambar arsitektur aplikasi dapat dilihat pada gambar 3.

Pada lapisan Business Layer, digunakan RESTful API web service sebagai jembatan komunikasi atau pertukaran data antara sistem client dengan sistem server. Lapisan Business Layer melakukan pengolahan API ini supaya dapat dipakai oleh aplikasi. API yang diolah oleh aplikasi akan menggunakan tipe JSON. Pengolahan API ini dapat dilihat pada hasil semua implementasi kasus penggunaan.



Gambar 3 Arsitektur Aplikasi MyIntan

2.2.3 Perancangan Diagram Kasus Penggunaan

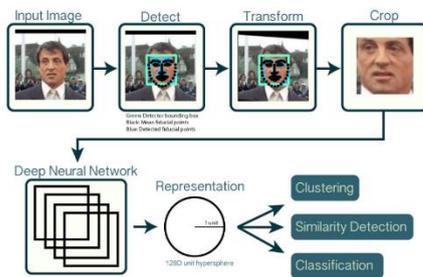


Gambar 4 Use Case Diagram Aplikasi

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Metode Deep Learning pada Aplikasi

Dalam penggunaan metode *Deep Learning* untuk melakukan verifikasi wajah, sistem akan memulai dari deteksi wajah dari input citra, kemudian melakukan *cropping dan alignment* wajah yang telah dideteksi, hingga melakukan ekstraksi *Feature Vector* wajah yang berbentuk 128 array, dari hasil ini akan dapat diolah lagi untuk melakukan klustering, klasifikasi, hingga *similarity detection* menggunakan perbandingan *euclidean distance* antar 2 *Feature Vector* wajah. Secara ringkas bagan arsitektur aplikasi dapat di lihat seperti pada gambar 5.



Gambar 5 Alur Penggunaan Deep Learning pada aplikasi

3.2 Implementasi Fitur Face Detection

Tahap awal dalam melakukan verifikasi wajah atau pengenalan wajah, adalah melakukan deteksi wajah dari citra input. Metode deteksi wajah yang digunakan dalam menangkap wajah dalam citra pada aplikasi yang dibangun adalah menggunakan *SSD (Single Shot Multibox Detector)* yang sudah dijelaskan pada sub bab sebelumnya. Sistem akan menampilkan kotak wajah yang didalamnya berisikan skor probabilitas kesesuaian wajah, ilustrasi implementasi pendeteksian wajah pada aplikasi My Intan dapat dilihat pada gambar 6 dan 7.



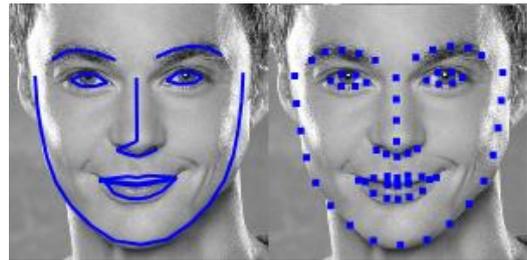
Gambar 7 Face Detection Menu Registrasi



Gambar 6 Face Detection Menu Kehadiran

3.3 Implementasi Fitur Face Landmark Detection dan Face Alignment

Tahap kedua sebelum wajah yang sudah dideteksi ini di proses pada *face recognition network*, sistem akan melakukan *cropping wajah dan alignment* sehingga wajah yang sudah di crop akan diposisikan menjadi *center* atau menengah. Untuk hal itu sistem akan melakukan metode *CNN* sederhana untuk mengembalikan 68 titik landmark wajah.



Gambar 8 Ilustrasi Face Landmark

Dari posisi titik-titik landmark wajah ini, wajah yang sudah dicrop dapat diproses untuk memosisikannya hingga menengah. Hasil dari simulasi *Face Landmark* dan *Face Alignment* dapat dilihat pada gambar dibawah ini, simulasi input citra wajah pada saat proses deteksi wajah (kiri) dibandingkan dengan simulasi hasil wajah yang telah di crop dan *alignment* (kanan).



Gambar 9 Simulasi Face Landmark Detection



Gambar 10 Simulasi Face Alignment dan Cropping

3.4 Implementasi Fitur Face Recognition

Setelah didapatkan wajah yang sudah dicrop dan dilakukan *alignment* hingga menjadi *center* atau menengah, citra ini akan diproses oleh jaringan *face recognition services* berdasarkan arsitektur *ResNet-34* yang diimplementasikan dalam *dlib* untuk dilakukan ekstraksi data wajah. Jaringan saraf *face recognition services* ini telah dilatih untuk belajar memetakan berbagai fitur wajah pada manusia

dan menerjemahkannya ke dalam deskriptor wajah (vektor dengan nilai 128).



Gambar 11 Simulasi hasil Face Descriptor

Untuk membandingkan dua wajah, deskriptor wajah yang terekstrak akan dibandingkan dengan dataset deskriptor wajah yang telah tersimpan sebelumnya. Lebih tepatnya, sistem akan menghitung *Euclidean distance* (jarak *Euclidean*) antara dua deskriptor wajah dan menilai apakah dua wajah tersebut serupa berdasarkan pada nilai *threshold* (untuk ukuran gambar wajah 150 x 150, **0.6** adalah nilai *threshold* yang bagus). Untuk menghitung *Euclidean distance* untuk n-dimensional vector adalah sebagai berikut:

$$d(p, q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \dots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}$$

Keterangan:

p1 = nilai vektor deskriptor wajah ke 1 (dataset wajah)

q1 = nilai vektor deskriptor wajah 2 (input wajah)

Simulasi *Face Recognition* atau verifikasi wajah dengan menggunakan nilai *Euclidean distance* dapat dilihat pada tabel gambar berikut

Tabel 1 Simulasi Uji Verifikasi Wajah (Valid Skenario)



Tabel 2 Simulasi Uji Verifikasi Wajah (Invalid Skenario)

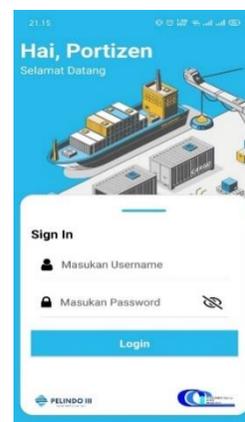


Pada result hasil uji simulasi verifikasi wajah valid dan invalid, sistem dapat mengetahui kesesuaian wajah dari *Euclidean distance*, batas maksimal pengenalan wajah yang dikonfigurasi di sistem adalah 0.6, apabila *distance* melebihi value yang ditentukan, sistem akan tidak mengenali wajah tersebut lalu mengembalikan nilai *unknown*. Dapat dilihat juga, sistem juga melakukan klasifikasi lain seperti *gender classification* dan *expressions classification* pada citra wajah input.

3.5 Implementasi Antarmuka Aplikasi

Implementasi dari antarmuka aplikasi mobile ini dikembangkan menggunakan *Framework React Native* berbasis *javascript* yang dibangun dan diimplementasikan dalam lingkungan kerja *Visual Studio Code*.

3.5.1 Halaman Login



Gambar 12 Implementasi Halaman Login

3.5.2 Halaman HomeScreen Utama

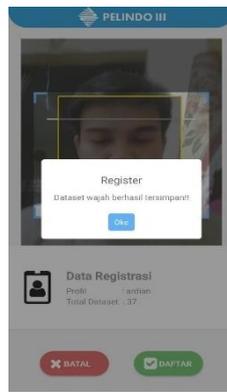


Gambar 13 Implementasi Halaman Homescreen

3.5.3 Halaman Daftar Wajah



Gambar 15 Implementasi Halaman Daftar Wajah



Gambar 14 Pendaftaran Wajah Berhasil

3.5.4 Halaman My Intan (Kehadiran)

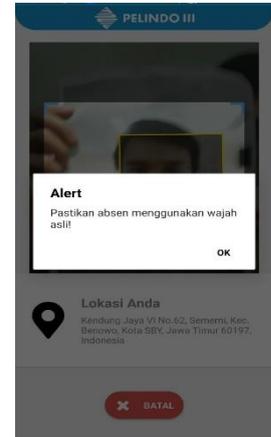


Gambar 16 Implementasi Halaman My Intan (Kehadiran)

Pada Halaman ini aplikasi akan mendeteksi tanggal, jam dan lokasi terakhir sebagai input saat absensi Clock In/Out, dalam menu ini juga telah dilengkapi pengecekan untuk Device yang menggunakan fake gps dan validasi jarak absensi sesuai unit kerja karyawan

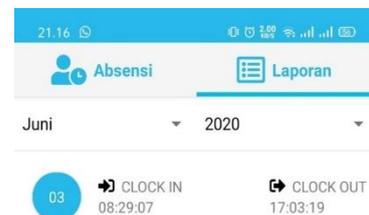


Gambar 18 Implementasi Halaman Absensi untuk Verifikasi Wajah



Gambar 17 Liveness Detector

3.5.5 Halaman Laporan Kehadiran



Gambar 19 Implementasi Halaman Laporan Kehadiran

3.5.6 Halaman Laporan Profil



Gambar 20 Implementasi Halaman Profil

3.6 Pengujian Aplikasi

Pengujian aplikasi yang akan diujikan dalam penelitian ini meliputi pengujian fungsional pada aplikasi, pengujian kegunaan, dan pengujian eksekusi waktu. Hal ini digunakan untuk memastikan bahwa sistem sudah berjalan sesuai dengan kebutuhan. Lingkungan pengujian dalam uji fungsionalitas ini dipakai untuk semua kasus penggunaan. Spesifikasi lingkungan pengujian, akan dijelaskan pada tabel 3.

Spesifikasi	Deskripsi
Jenis Perangkat	Smartphone
Merek Perangkat	Realme 3 Pro 2019
Sistem Operasi	Android Q (Android 10)
Memori Internal	64GB
Memori RAM	4GB

Tabel 3 Spesifikasi Perangkat Pengujian

3.6.1 Pengujian Fungsionalitas

Pengujian fungsionalitas digunakan untuk melakukan pengecekan hasil output dari aplikasi, pengujian dilakukan dengan metode black box, detil pengujian dapat dilihat pada tabel 4.

Tabel 4 Pengujian Fungsionalitas

Modul	Kasus dan Hasil Pengujian		
	Data Masukan	Yang Diharapkan	Hasil
Login	Username dan password telah diisi benar	Menampilkan Halaman <i>Homescreen</i>	valid
	Username dan password telah diisi salah	Menampilkan notifikasi tidak diijinkan masuk	valid
Daftar Wajah	Melakukan pendaftaran dataset wajah	Dataset wajah tersimpan	valid
Kehadiran	Validasi <i>Fake Gps</i>	Tombol aksi <i>Clock In/Out Disable</i> dan muncul notif penggunaan <i>fake gps</i>	valid
	Validasi Lokasi Absensi	Tombol aksi <i>Clock In/Out Disable</i> dan muncul notif	valid

		radius absensi diluar jangkuan	
	Lihat <i>Last Activity</i>	Aktivitas riwayat clock In/Out dapat ditampilkan	valid
	Cetak Laporan	Menampilkan laporan sesuai periode	valid
Clock In/Out	Melakukan Absensi Clock In/Out dengan verifikasi wajah	Berhasil melakukan absensi	valid
Profil	Merubah Password	Berhasil merubah password	valid
	Merubah Foto Profil	Berhasil merubah foto profil	valid

3.6.2 Pengujian Kegunaan

Pengujian kegunaan ini dilakukan untuk mengetahui tingkat kegunaan aplikasi terhadap pengguna. Kegunaan yang diuji meliputi kemudahan karyawan dalam melakukan pendaftaran dataset wajah, proses melakukan absensi Clock In/Clock Out, dan proses verifikasi wajah karyawan. Pengujian dilakukan dengan memberikan kesempatan pada pengguna untuk mencoba sendiri mengoperasikan aplikasi yang telah dikembangkan dengan arahan dari pengembang aplikasi. Pengujian ini dilakukan dengan dibantu 6 karyawan, data user untuk login adalah sesuai dengan user login yang digunakan untuk oleh karyawan untuk login kedalam aplikasi kepegawaian, sehingga tidak perlu dilakukan konfigurasi tambahan. Pengguna diminta untuk melakukan pengisian kuesioner setelah melakukan uji coba aplikasi, detil kuesioner dapat dilihat pada tabel 5.

Tabel 5 Kuesioner

1. Menurut saya, aplikasi ini dapat dioperasikan dengan mudah	1 2 3 4 5
STS	0 0 0 0 0 SS
2. Menurut saya, tampilan aplikasi ini user friendly dan menarik	1 2 3 4 5
STS	0 0 0 0 0 SS

3. Saya pikir aplikasi mudah untuk digunakan (user Menurut saya, lebih mudah dan efisien melakukan absensi menggunakan aplikasi ini daripada melakukan absensi menggunakan mesin fingerprint)	1 2 3 4 5	STS	O	O	O	O	O	SS
4. Menurut saya, mudah untuk melakukan pendaftaran wajah	1 2 3 4 5	STS	O	O	O	O	O	SS
5. Menurut saya, lokasi saat melakukan absensi sudah sesuai dan akurat	1 2 3 4 5	STS	O	O	O	O	O	SS
6. Menurut saya, jam absensi saat melakukan absensi adalah sesuai dengan jam yang berlaku di tempat kerja	1 2 3 4 5	STS	O	O	O	O	O	SS
7. Menurut saya, penggunaan fitur verifikasi wajah saat melakukan absensi, mudah untuk digunakan dan membuat data absensi lebih aman	1 2 3 4 5	STS	O	O	O	O	O	SS
8. Menurut saya, proses absensi untuk Clock In maupun Clock Out pada aplikasi berjalan dengan cepat	1 2 3 4 5	STS	O	O	O	O	O	SS
9. Menurut saya, hasil absensi dengan fitur verifikasi wajah memiliki hasil yang akurat	1 2 3 4 5	STS	O	O	O	O	O	SS

Tabel 6 Hasil Kuesioner

Respon	Penilaian								
	1	2	3	4	5	6	7	8	9
1	4	4	4	5	5	5	4	4	4
2	4	5	4	4	4	5	4	5	5
3	5	5	5	5	5	5	5	5	5
4	5	5	5	5	5	5	5	5	5
5	5	5	4	4	5	5	4	3	5
6	5	6	4	5	4	5	5	5	5
Rata-rata	4.6	4.8	4.5	4.6	4.8	4.8	4.5	4.5	4.8
Total rata-rata						4.6			

Berdasarkan hasil kuesioner pada Tabel 6 ditunjukkan bahwa nilai rata-rata penilaian responden terhadap kegunaan dari aplikasi ini menghasilkan nilai total rata-rata skala 4.6 dari skala penilaian 5 sehingga dapat diambil kesimpulan bahwa aplikasi ini mudah digunakan dan dapat memberikan manfaat bagi pengguna.

3.6.3 Pengujian Waktu Eksekusi

Pengujian waktu eksekusi ini dilakukan untuk mengetahui tingkat kecepatan aplikasi dalam melakukan proses absensi baik Clock In maupun Clock Out oleh karyawan, pada kasus penggunaan melakukan absensi Clock In dan Clock Out. Pengujian waktu eksekusi ini dibagi menjadi dua pengujian, pengujian pertama dilakukan untuk menghitung lama proses absensi dimulai dari aplikasi mendapatkan input citra wajah, mengupload data ke server, memverifikasi wajah hingga proses menyimpan data absensi dan menampilkan pesan notifikasi absensi. Untuk pengujian kedua, dilakukan untuk menghitung performa Face Recognition Service dalam memverifikasi wajah terhitung dari data citra wajah sudah terupload ke server hingga wajah telah berhasil terverifikasi. Pengguna aplikasi pada pengujian ini sama dengan pengguna pada pengujian kegunaan.

Tabel 7 Uji Eksekusi Waktu

Respon	Smartphone	Eksekusi Server	Eksekusi Perangkat
1	Samsung Galaxy Note Fe (Android X)	1.12 detik	5.7 detik
2	Realme 5 Pro (Android X)	1.07 detik	4.9 detik
3	Redmi Note 5 Pro (Android Oreo)	1.24 detik	4.8 detik
4	Redmi Note 5 Pro (Android Oreo)	1.48 detik	8.2 detik
5	Samsung A52 (Android X)	1.26 detik	5.23 detik
6	Samsung A21 (Android X)	1.19 detik	6.7 detik

Bersarakan hasil pengujian waktu eksekusi didapatkan pada Tabel 7, terdapat perbedaan waktu lama eksekusi pada perangkat yang digunakan oleh tiap pengguna. Waktu eksekusi perangkat tercepat dari seluruh pengujian adalah selama 4.8 detik dan paling lama 8.2 detik, sedangkan untuk waktu eksekusi di server menunjukkan waktu tercepat untuk melakukan verifikasi wajah adalah selama 1.07 detik dan paling lama 1.48 detik. Hasil ini menunjukkan bahwa lama waktu eksekusi di perangkat ini didasarkan pada spesifikasi dan kecepatan jaringan internet perangkat untuk melakukan upload citra wajah, setelah citra wajah ini terupload, sistem Face Recognition Service dapat melakukan verifikasi wajah rata-rata dalam kurun waktu 1.2 detik.

Hasil ini menunjukkan bahwa verifikasi wajah menggunakan metode Deep Learning sangat memuaskan, namun sistem juga memiliki batasan, yaitu lama waktu eksekusi juga dipengaruhi dengan jaringan perangkat.

4. SIMPULAN

Dari hasil pengujian yang dilakukan, dapat ditarik kesimpulan bahwa penggunaan aplikasi absensi android menggunakan verifikasi wajah dengan metode Deep Learning pada PT. Pelabuhan Indonesia III (Persero) dapat diterima dengan baik oleh pengguna, hal ini ditunjukkan dari hasil yang didapat dalam questioner dengan nilai total rata-rata 4.6 dari skala total 5, setelah pengguna diminta untuk melakukan uji coba aplikasi menggunakan perangkat pribadi. Performa aplikasi saat melakukan verifikasi wajah menggunakan metode Deep learning menunjukkan hasil yang memuaskan, dengan kecepatan eksekusi server tercepat dalam waktu 1.2 detik sedangkan eksekusi perangkat tercepat dalam waktu 4.8 detik.

Dengan dibangunnya Aplikasi Absensi Online Berbasis Android Menggunakan Metode Deep Learning Pada PT. Pelabuhan Indonesia III (Persero) diharapkan membantu manajemen dalam mengurangi penumpukan karyawan dipagi hari saat melakukan absensi, hal ini juga sejalan dalam era pandemi untuk menjaga jarak antar karyawan dalam area lingkungan kerja. Data absensi yang dikirimkan oleh aplikasi ini telah divalidasi atas lokasi absensi sesuai dengan unit kerja karyawan dengan memanfaatkan fitur geofencing, serta verifikasi wajah yang dilengkapi liveness detector, sehingga menjamin

validitas data absensi.

Adapun beberapa hal yang dapat dikembangkan adalah mengembangkan aplikasi MyIntan agar dapat menjadi aplikasi *Personal Assistance* Karyawan, yang didalamnya sudah melengkapi kebutuhan karyawan atas semua data kepegawaian dan membuat mekanisme pendaftaran wajah agar dataset yang diambil lebih variatif / ekspresi wajah tidak monoton, sehingga dapat meminimalisir redundan data.

DAFTAR PUSTAKA

Jurnal:

- [1] I Putu Putra Yana Wardana, I A Dwi Giriantari, Made Sudarma. Aplikasi Verifikasi Wajah Untuk Absensi Pada Platform Android Dengan Menggunakan Algoritma Fisherface Register: Teknologi Elektro. 2016; 15(2), 45-52.
- [2] Husain, Al, Prastian, A. H. P., Ramadhan, Andre. Perancangan Sistem Absensi Online Menggunakan Android Guna Mempercepat Proses Kehadiran Karyawan Pada PT. Sintech Berkah Abadi. Register: Technomedia Journal (TMJ).2017; 2(1), 105-116.
- [3] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. arXiv:1512.03385 [cs], 2015. arXiv: 1512.03385.
- [4] Y. Hu, A. E. G. Huber, J. Anumula, and S. Liu. Overcoming the vanishing gradient problem in plain recurrent networks. CoRR, abs/1801.06105, 2018.

Halaman Internet:

- [1]Mühler, Vincent 2018. face-api.js - JavaScript API for Face Recognition in the Browser with tensorflow.js. <https://itnext.io/face-api-js-javascript-api-for-face-recognition-in-the-browser-with-tensorflow-js-bcc2a6c4cf07>(diakses tanggal 1 Maret 2020)
- [2]Goyal, Asha 2019. Deeper into Deep Neural Networks. <https://towardsdatascience.com/deeper-into-deep-neural-networks-6bf658df2060> (diakses tanggal 10 Maret 2020)
- [3]Jamilu Auwalu Adamu 2019. Superintelligent Deep Learning Artificial Neural Networks. https://www.researchgate.net/figure/How-Neural-Network-relate-with-Human-Brain-Source-Google-Images-2019_fig1_338015239 (diakses tanggal 10 Maret 2020)