

**RANCANG BANGUN APLIKASI *ANDROID* ABSENSI DAN  
MANAJEMEN PENUGASAN KARYAWAN BERBASIS *REACT NATIVE*  
(STUDI KASUS: PT. QUEEN NETWORK NUSANTARA)**

**(Skripsi)**

**Oleh**

**RAMADHANI ASROFA KEMAL**



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
2023**

**RANCANG BANGUN APLIKASI *ANDROID* ABSENSI DAN  
MANAJEMEN PENUGASAN KARYAWAN BERBASIS *REACT NATIVE*  
(STUDI KASUS: PT. QUEEN NETWORK NUSANTARA)**

**Oleh**

**RAMADHANI ASROFA KEMAL**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Mencapai Gelar  
SARJANA TEKNIK**

**Pada**

**Program Studi Teknik Informatika  
Fakultas Teknik**



**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS LAMPUNG  
2023**

## ABSTRAK

### **RANCANG BANGUN APLIKASI *ANDROID* ABSENSI DAN MANAJEMEN PENUGASAN KARYAWAN BERBASIS *REACT NATIVE* (STUDI KASUS: PT. QUEEN NETWORK NUSANTARA)**

Oleh

**RAMADHANI ASROFA KEMAL**

Organisasi harus mengelola sumber daya manusia dengan baik untuk mencapai tujuan. Sumber daya manusia (SDM) yang baik adalah SDM yang memiliki kemampuan untuk mengelola waktu yang baik. PT. Queen Network Nusantara belum menerapkan sistem absensi sehingga menyebabkan jam kerja pegawai perusahaan menjadi tidak teratur. Penelitian ini bertujuan untuk merancang dan membangun aplikasi *android* yang memfasilitasi absensi dan manajemen penugasan karyawan di PT. Queen Network Nusantara. Dalam penelitian ini, metode *Personal Extreme Programming* (PXP) digunakan untuk mengembangkan aplikasi secara cepat dan fleksibel. Dalam tahap perencanaan, kebutuhan sistem diidentifikasi melalui *User Stories*. Kemudian, dilakukan perancangan sistem berdasarkan analisis kebutuhan menggunakan *Unified Modelling Language* (UML) dengan *Use Case Diagram* dan *Activity Diagram*. Setelah itu, dilakukan implementasi kode program menggunakan *Framework React Native*. Terakhir, dilakukan pengujian menggunakan teknik *Black Box Testing*. Penelitian berhasil mengembangkan 9 fitur pada aplikasi absensi dan telah diuji menggunakan 33 skenario pengujian serta hasilnya sesuai dengan yang diharapkan.

Kata kunci: Absensi, *React Native*, *Android*, Layanan Berbasis Lokasi

## ABSTRACT

### **DESIGN AND DEVELOPMENT OF *ANDROID* APPLICATION FOR ATTENDANCE AND TASK MANAGEMENT BASED ON *REACT NATIVE* (CASE STUDY: PT. QUEEN NETWORK NUSANTARA)**

By

**RAMADHANI ASROFA KEMAL**

*Organizations must manage human resources well to achieve their goals. Good human resources (HR) are HR who can manage time well. PT. Queen Network Nusantara has not implemented an absence system, causing the company's employees' working hours to become irregular. This study aims to design and build an Android application that facilitates employee attendance and assignment management at PT. Queen of the Archipelago Network. In this study, the Personal Extreme Programming (PXP) method is used to develop applications quickly and flexibly. At the planning stage, system requirements are identified through User Stories. Then do the system design based on needs analysis using the Unified Modeling Language (UML) with Use Case Diagrams and Activity Diagrams. After that, implementation of the program code is carried out using the React Native Framework. Finally, testing is carried out using the Black Box Testing technique. The research succeeded in developing 9 features in the attendance application and has been tested using 33 test scenarios, and the results are as expected.*

*keywords: Attendance, React Native, Android, Location Based Service*

Judul Skripsi : **RANCANG BANGUN APLIKASI *ANDROID*  
ABSENSI DAN MANAJEMEN PENUGASAN  
KARYAWAN BERBASIS *REACT NATIVE*  
(STUDI KASUS: PT. QUEEN NETWORK  
NUSANTARA)**

Nama Mahasiswa : **Ramadhani Asrofa Kemal**

Nomor Pokok Mahasiswa : 1915061028

Program Studi : Teknik Informatika

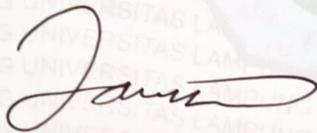
Fakultas : Teknik

**MENYETUJUI**

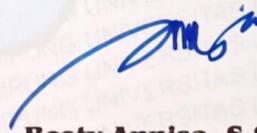
**1. Komisi Pembimbing**

Pembimbing Utama

Pembimbing Pendamping



**Ir. Meizano Ardhi Muhammad, S.T., M.T.**  
NIP 19810528 201212 1 001

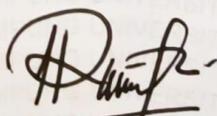


**Resty Annisa, S.ST., M.Kom.**  
NIP 19900830 201903 2 019

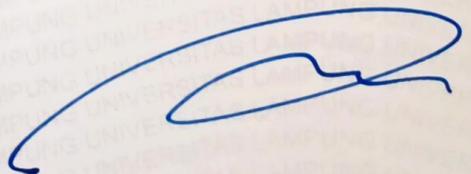
**2. Mengetahui**

Ketua Jurusan  
Teknik Elektro

Ketua Program Studi  
Teknik Informatika



**Herlinawati, S.T., M.T.**  
NIP 19710314 199903 2 001



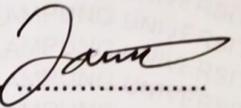
**Mona Arif Muda, S.T., M.T.**  
NIP 19711112 200003 1 002

## MENGESAHKAN

### 1. Tim Penguji

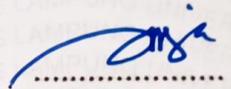
Ketua

: **Ir. Meizano Ardhi Muhammad, S.T., M.T.**



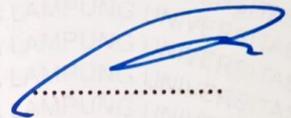
Sekretaris

: **Resty Annisa, S.ST., M.Kom.**



Penguji

: **Mona Arif Muda, S.T., M.T.**



### 2. Dekan Fakultas Teknik



**Dr. Eng. Ir. Helmy Fitriawan, S.T., M.Sc. }**

NIP.19750928 200112 1 002

Tanggal Lulus Ujian Skripsi : **2 Agustus 2023**

## SURAT PERNYATAAN

Saya yang bertandatangan di bawah ini, menyatakan bahwa skripsi saya dengan judul “Rancang Bangun Aplikasi *Android* Absensi Dan Manajemen Penugasan Karyawan Berbasis *React Native* (Studi Kasus: PT. Queen Network Nusantara)” dibuat oleh saya sendiri. Semua hasil yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila di kemudian hari terbukti bahwa skripsi ini merupakan salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan hukum atau akademik yang berlaku.

Bandar Lampung,  
Pembuat pernyataan,

2023



Ramadhani Asrofa Kemal  
NPM. 1915061028

## RIWAYAT HIDUP



Penulis dilahirkan di Lampung, pada tanggal 25 November 2001. Penulis merupakan anak pertama dari pasangan Bapak Mintarja dan Ibu Arya Puspa. Penulis menyelesaikan pendidikannya di SD Negeri 1 Kibang pada tahun 2013, SMP Negeri 1 Kibang pada tahun 2016, dan SMK Negeri 3 Metro pada tahun 2019. Pada tahun 2019 penulis terdaftar sebagai mahasiswa Program Studi Teknik Informatika, Jurusan Teknik Elektro, Fakultas Teknik Universitas Lampung melalui jalur SBMPTN. Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan, antara lain:

1. Menjadi anggota biasa Himpunan Mahasiswa Teknik Elektro Universitas Lampung, Departemen Sosial dan Kewirausahaan, Divisi Kewirausahaan pada tahun 2019.
2. Menjadi anggota biasa Himpunan Mahasiswa Teknik Elektro Universitas Lampung, Departemen Komunikasi dan Informasi, Divisi Media Informasi pada tahun 2020.
3. Melaksanakan Kuliah Kerja Nyata di Desa Rajabasa Lama, Kecamatan Labuhan Ratu, Kabupaten Lampung Timur, Provinsi Lampung pada bulan Januari sampai dengan Februari 2022.
4. Mengikuti program Studi Independen Kampus Merdeka dari Kementerian Pendidikan dan Budaya dengan mengambil kelas *How to be an Ideal Top Search React JS Front End Engineer* di PT Marka Kreasi Persada pada tahun 2022.
5. Melakukan kerja praktik di PT. Queen Network Nusantara sebagai *Front End Developer* pada tahun 2022.

## **MOTTO**

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.”

**(Q.S. Al-Baqarah : 286)**

“Ilmu tanpa amal adalah kegilaan, dan amal tanpa ilmu adalah kesia-siaan.”

**(Imam Ghazali)**

“Hiduplah seolah engkau mati besok. Belajarlah seolah engkau hidup selamanya.”

**(Mahatma Gandhi)**

“Pendidikan Memiliki Akar yang Pahit, tapi Buahnya Manis.”

**(Aristoteles)**

## PERSEMBAHAN

*Bismillaahirrohmaanirrahiim,  
Dengan mengharapkan ridho dari Allah SWT,  
Kupersembahkan karyaku ini untuk orang-orang yang kusayangi  
dengan setulus hati.*

*Orangtua tercinta,*

*Keluargaku,*

*Teman-Temanku,*

*Dan*

*Orang-orang yang telah membantu hidupku*

*Terimakasih untuk semuanya,  
Kalian adalah hartaku yang paling berharga*

## SANWACANA

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayat-Nya sehingga penulis dapat menyelesaikan penyusunan skripsi/tugas akhir ini dengan judul “Rancang Bangun Aplikasi *Android* Absensi Dan Manajemen Penugasan Karyawan Berbasis *React Native* (Studi Kasus: PT. Queen Network Nusantara)”. Dalam pelaksanaan dan pembuatan skripsi/tugas akhir ini penulis menerima dukungan baik secara moril maupun materil yang sangat berharga dari berbagai pihak. Oleh karena itu, penulis ingin mengucapkan terima kasih kepada semua pihak yang telah membantu, khususnya kepada:

1. Kedua orangtuaku, Bapak Mintarja dan Ibu Arya Puspa tercinta serta seluruh keluarga penulis yang tidak hentinya mendo’akan serta memberikan dorongan semangat dan materi;
2. Bapak Dr. Eng. Helmy Fitriawan, S.T., M.Sc., selaku Dekan Fakultas Teknik Universitas Lampung;
3. Ibu Herlinawati, S.T., M.T. selaku Ketua Jurusan Teknik Elektro Universitas Lampung;
4. Bapak Mona Arif Muda, S.T., M.T. selaku Ketua Program Studi Teknik Informatika Universitas Lampung dan selaku Penguji yang telah membantu proses kelancaran pengerjaan penelitian dan memberikan banyak saran serta masukan terhadap penelitian ini;
5. Bapak Ir. Meizano Ardhi Muhammad, S.T., M.T. selaku Pembimbing Utama yang selalu meluangkan waktunya untuk memberikan bimbingan dan dukungan serta memudahkan penulis dalam menyelesaikan penelitian ini;
6. Ibu Resty Annisa, S. ST., M. Kom, selaku Pembimbing Pendamping yang selalu memberikan dukungan serta bimbingan agar menjadi lebih baik;

7. Ibu Yessi Mulyani, S.T., M.T. selaku Pembimbing Akademik yang selalu memberikan bimbingan dan dukungan kepada penulis selama menyelesaikan perkuliahan.
8. Mbak Rika selaku Admin Program Studi Teknik Informatika yang telah banyak membantu penulis dalam segala urusan administrasi selama perkuliahan;
9. Seluruh dosen dan staf Jurusan Teknik Informatika Unila yang memberi masukan dan mempermudah proses pembuatan skripsi / tugas akhir ini.
10. Ni Nengah Rizki Noventy yang selalu memberi dukungan dengan tulus sehingga penulis dapat menyelesaikan skripsi ini hingga tuntas.
11. Sahabat Laut, Sanjaya Ilham Kurnia, M. Iqbal Syahbana, Surya Saputra, M. Irfan Fadhila, Royyan Fajrul Falah, Rachmat Agung Ananda, M. Naufal Ashiddiq, Nabila Tusyifa, Presillia Br Sianturi, Rachel Adella Ferninda, Ridha Aisyiyah Riandani, dan Tania Argadini yang selalu memberikan saran, dukungan, dan bantuan kepada penulis selama menjalani perkuliahan.
12. Keluarga besar Teknik Elektro Angkatan 2019 yang telah menjadi teman seperjuangan sejak mahasiswa baru. Terimakasih telah mewarnai masa perkuliahan penulis.

Penulis berharap agar laporan ini dapat menjadi referensi bagi pengembangan keilmuan di bidang teknik informatika. Oleh karena itu, semoga penelitian ini bermanfaat bagi yang membacanya.

Bandar Lampung,

2023

Penulis,



Ramadhani Asrofa Kemal

## DAFTAR ISI

	Halaman
<b>PERSEMBAHAN.....</b>	<b>i</b>
<b>SANWACANA .....</b>	<b>ii</b>
<b>DAFTAR ISI.....</b>	<b>iv</b>
<b>DAFTAR GAMBAR.....</b>	<b>vii</b>
<b>DAFTAR TABEL .....</b>	<b>x</b>
<b>I. PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Tujuan Penelitian.....	2
1.3 Rumusan Masalah .....	3
1.4 Batasan Masalah.....	3
1.5 Sistematika Penulisan.....	3
<b>II. TINJAUAN PUSTAKA.....</b>	<b>5</b>
2.1 Manajemen Sumber Daya Manusia .....	5
2.2 PT. Queen Network Nusantara.....	6
2.3 Sistem Informasi.....	7
2.4 Absensi .....	8
2.5 Aplikasi <i>Mobile</i> .....	9
2.6 <i>Android</i> .....	10
2.7 <i>Framework React Native</i> .....	12
2.8 <i>Javascript</i> .....	13

2.9	<i>Location Based Service (LBS)</i> .....	13
2.10	<i>Global Position System (GPS)</i> .....	15
2.11	<i>Haversine Formula</i> .....	16
2.12	<i>Google Maps API</i> .....	17
2.13	<i>Expo Camera</i> .....	18
2.14	<i>Visual Studio Code</i> .....	19
2.15	<i>Personal Extreme Programming</i> .....	19
2.16	<i>Unified Modelling Language</i> .....	20
2.17	<i>Black Box Testing</i> .....	20
2.18	<i>State of The Art</i> .....	21
2.18.1	<i>Research on Mobile Phone Attendance Positioning System Based on Campus Network</i> .....	21
2.18.2	<i>Multifactor Authentication on Mobile Secure Attendance System..</i>	22
2.18.3	<i>Android Application for Presence Recognition based on Face and Geofencing</i> .....	22
<b>III.</b>	<b>METODOLOGI PENELITIAN</b> .....	<b>24</b>
3.1	Waktu dan Tempat .....	24
3.2	Alat dan Bahan Penelitian .....	24
3.2.1	Alat Penelitian.....	24
3.2.2	Bahan Penelitian.....	25
3.3	Tahapan Penelitian .....	25
3.3.1	<i>Planning (Perencanaan)</i> .....	26
3.3.2	<i>Design (Perancangan)</i> .....	26
3.3.3	<i>Coding (Pengkodean)</i> .....	26
3.3.4	<i>Testing (Pengujian)</i> .....	26
3.4	Perencanaan ( <i>Planning</i> ) .....	26

3.4.1	<i>User Stories</i> .....	27
3.4.2	Analisis Kebutuhan .....	28
3.5	Perancangan ( <i>Design</i> ).....	28
3.5.1	<i>Use Case Diagram</i> .....	28
3.5.2	<i>Activity Diagram</i> .....	29
3.5.3	Perancangan Antarmuka .....	40
<b>IV.</b>	<b>HASIL DAN PEMBAHASAN</b> .....	<b>51</b>
4.1	Hasil.....	51
4.1.1	Pengkodean ( <i>Coding</i> ) Iterasi 1 .....	51
4.1.2	Pengujian ( <i>Testing</i> ) Iterasi 1 .....	67
4.1.3	Pengkodean ( <i>Coding</i> ) Iterasi 2.....	73
4.1.4	Pengujian ( <i>Testing</i> ) Iterasi 2 .....	80
4.2	Pembahasan .....	89
4.2.1	Metode <i>Personal Extreme Programming</i> .....	89
4.2.2	Aplikasi <i>Android</i> Absensi dan Penugasan PT. Queen Network Nusantara.....	90
<b>BAB V</b>	<b>SIMPULAN DAN SARAN</b> .....	<b>93</b>
5.1	Simpulan.....	93
5.2	Saran.....	93
<b>DAFTAR PUSTAKA</b>	.....	<b>94</b>

## DAFTAR GAMBAR

	Halaman
Gambar 2.2.1 Logo PT. Queen Network Nusantara .....	6
Gambar 2.2.2 Foto Kantor PT. Queen Network Nusantara .....	6
Gambar 2.4.1 Contoh alat sistem absensi menggunakan identifikasi sidik jari.....	9
Gambar 2.6.1 Logo <i>Android</i> .....	10
Gambar 2.7.1 Logo <i>React</i> .....	13
Gambar 2.9.1 Teknologi <i>Location Based Services</i> .....	14
Gambar 2.10.1 <i>Global Positioning System</i> Menggunakan Satelit.....	16
Gambar 3.3.1 Tahapan Penelitian .....	25
Gambar 3.5.1 <i>Use Case Diagram</i> Sebelum Pegawai <i>Login</i> .....	28
Gambar 3.5.2 <i>Use Case Diagram</i> Setelah Pegawai <i>Login</i> .....	29
Gambar 3.5.3 <i>Activity Diagram Login</i> .....	30
Gambar 3.5.4 <i>Activity Diagram Logout</i> .....	31
Gambar 3.5.5 <i>Activity Diagram</i> Melakukan Absensi .....	32
Gambar 3.5.6 <i>Activity Diagram</i> Mengajukan Cuti .....	33
Gambar 3.5.7 <i>Activity Diagram</i> Melihat Notifikasi.....	34
Gambar 3.5.8 <i>Activity Diagram</i> Mengubah Profil.....	35
Gambar 3.5.9 <i>Activity Diagram</i> Mengubah <i>Password</i> .....	36
Gambar 3.5.10 <i>Activity Diagram</i> Melihat Riwayat Kehadiran.....	37
Gambar 3.5.11 <i>Activity Diagram</i> Melihat Jadwal.....	37
Gambar 3.5.12 <i>Activity Diagram</i> Membuat Tugas .....	38
Gambar 3.5.13 <i>Activity Diagram</i> Melihat dan Melaporkan Tugas.....	39
Gambar 3.5.14 Halaman <i>Login</i> .....	40
Gambar 3.5.15 Halaman Beranda .....	41
Gambar 3.5.16 Halaman Kehadiran.....	42

Gambar 3.5.17 Halaman Tugas.....	43
Gambar 3.5.18 Halaman Akun.....	44
Gambar 3.5.19 Halaman Notifikasi .....	45
Gambar 3.5.20 Halaman Cuti.....	46
Gambar 3.5.21 Halaman Ubah Profil.....	47
Gambar 3.5.22 Halaman Ubah Sandi.....	48
Gambar 3.5.23 Halaman Absensi.....	49
Gambar 3.5.24 Halaman Kamera.....	50
Gambar 4.1.1 Implementasi <i>Input Field</i> .....	51
Gambar 4.1.2 Formik pada <i>Login</i> .....	52
Gambar 4.1.3 Fungsi <i>login User</i> .....	52
Gambar 4.1.4 Cek <i>Login User</i> .....	53
Gambar 4.1.5 <i>Button Logout</i> Akun.....	53
Gambar 4.1.6 Fungsi <i>Logout</i> Akun.....	54
Gambar 4.1.7 <i>Component Camera</i> .....	55
Gambar 4.1.8 Fungsi <i>handleFacesDetected</i> .....	55
Gambar 4.1.9 Fungsi <i>takePicture</i> .....	56
Gambar 4.1.10 <i>Component MapView</i> .....	57
Gambar 4.1.11 Fungsi <i>getLocation</i> .....	58
Gambar 4.1.12 Fungsi <i>getAddress</i> .....	58
Gambar 4.1.13 Fungsi <i>Haversine</i> .....	59
Gambar 4.1.14 Validasi Absensi.....	60
Gambar 4.1.15 <i>Component CardTotal</i> .....	61
Gambar 4.1.16 List Riwayat Cuti .....	61
Gambar 4.1.17 Fungsi <i>handleSubmit</i> Cuti.....	62
Gambar 4.1.18 List Notifikasi.....	62
Gambar 4.1.19 Implementasi Tampilan Foto Profil .....	63
Gambar 4.1.20 Implementasi Tampilan Profil Data Pribadi.....	63
Gambar 4.1.21 <i>Form Input</i> Data Profil.....	64
Gambar 4.1.22 <i>Handle Input Value Component Form</i> .....	64
Gambar 4.1.23 Fungsi <i>handleSubmit</i> Ubah <i>Password</i> .....	65
Gambar 4.1.24 Validasi Form Ubah <i>Password</i> .....	65

Gambar 4.1.25 List Riwayat Kehadiran.....	66
Gambar 4.1.26 <i>Fetching</i> Data Riwayat Kehadiran .....	67
Gambar 4.1.27 Fungsi <i>Get</i> Data Tanggal .....	74
Gambar 4.1.28 List Jadwal.....	74
Gambar 4.1.29 <i>Fetching</i> Data Jadwal .....	74
Gambar 4.1.30 Hasil Tampilan Fitur Jadwal .....	75
Gambar 4.1.31 List Tugas.....	76
Gambar 4.1.32 Fungsi <i>handleSubmit</i> Tugas .....	77
Gambar 4.1.33 Fungsi <i>handleSubmit</i> Pelaporan Hasil Tugas.....	78
Gambar 4.1.34 <i>Fetch</i> Data Riwayat Tugas .....	79
Gambar 4.1.35 Hasil Tampilan Detail Tugas .....	80
Gambar 4.1.36 Pengujian <i>Login Android</i> Versi 6.....	83
Gambar 4.1.37 Pengujian Absensi Dengan Wajah Asli <i>Android</i> Versi 6.....	84
Gambar 4.1.38 Pengujian Absensi Dengan Foto Wajah <i>Android</i> Versi 6.....	85
Gambar 4.1.39 Pengujian <i>Login Android</i> Versi 13.....	86
Gambar 4.1.40 Pengujian Absensi Dengan Wajah Asli <i>Android</i> Versi 13.....	87
Gambar 4.1.41 Pengujian Absensi Dengan Foto Wajah <i>Android</i> Versi 13.....	88
Gambar 4.1.42 Versi <i>Android</i> .....	89
Gambar 4.2.1 Aplikasi E-Gawi di <i>Playstore</i> .....	91

**DAFTAR TABEL**

	Halaman
Tabel 2.6.1 Versi <i>Android</i> .....	11
Tabel 3.1.1 Jadwal Penelitian .....	24
Tabel 3.2.1 Alat Penelitian.....	24
Tabel 3.4.1 <i>User Story</i> .....	27
Tabel 4.1.1 <i>Blackbox Testing Login</i> .....	68
Tabel 4.1.2 <i>Blackbox Testing Logout</i> .....	68
Tabel 4.1.3 <i>Blackbox Testing</i> Fitur Absensi Ketika Verifikasi Lokasi.....	69
Tabel 4.1.4 <i>Blackbox Testing</i> Fitur Absensi Ketika Verifikasi Wajah .....	70
Tabel 4.1.5 <i>Blackbox Testing</i> Fitur Cuti .....	70
Tabel 4.1.6 <i>Blackbox Testing</i> Fitur Notifikasi .....	71
Tabel 4.1.7 <i>Blackbox Testing</i> Fitur Ubah Profil .....	71
Tabel 4.1.8 <i>Blackbox Testing</i> Fitur Ubah <i>Password</i> .....	72
Tabel 4.1.9 <i>Blackbox Testing</i> Fitur Riwayat Kehadiran .....	73
Tabel 4.1.10 <i>Blackbox Testing</i> Fitur Jadwal .....	81
Tabel 4.1.11 <i>Blackbox Testing</i> Fitur Tugas .....	81
Tabel 4.1.12 Pengujian Sistem Operasi .....	82

## I. PENDAHULUAN

### 1.1 Latar Belakang

Organisasi harus mengelola sumber daya manusia dengan baik untuk mencapai tujuan. Sumber daya manusia (SDM) yang baik adalah SDM yang memiliki kemampuan untuk mengelola waktu yang baik. Efektifitas kerja memiliki hubungan yang positif dengan manajemen waktu, semakin baik manajemen waktu maka semakin baik pula efektifitas kerjanya. [1]

PT. Queen Network Nusantara (QNN) merupakan perusahaan yang bergerak dibidang teknologi dan penyedia layanan internet di Bandar Lampung, saat ini QNN sudah memiliki kurang lebih 30 pegawai namun belum menerapkan sistem absensi di dalam perusahaannya. Hal ini menyebabkan jam kerja pegawai perusahaan menjadi tidak teratur. Waktu kerja di Indonesia diatur dalam UU NO.13 Tahun 2003 Tentang Ketenagakerjaan, sebagaimana waktu kerja yang diatur adalah selama 40 jam dalam 1 minggu. [2] Salah satu cara memastikan waktu kerja yang cukup adalah dengan absensi, absensi yang valid dapat membantu memastikan kehadiran pegawai sesuai dengan jadwal kerja yang telah ditetapkan oleh pemerintah. QNN membutuhkan sebuah aplikasi yang memiliki fitur untuk melakukan absensi dan memiliki fitur tambahan seperti melakukan pelaporan tugas.

Sistem yang dimaksud adalah sistem yang digunakan untuk melakukan absensi yang dapat dilakukan dimana saja, mencatat semua riwayat kehadiran pegawai, semua pencatatan dilakukan secara otomatis oleh sistem, dan terdapat verifikasi wajah untuk memastikan bahwa absensi dilakukan oleh pegawai itu sendiri. Sistem penugasan yang dimaksud adalah sistem yang memiliki fitur pemberian tugas dan pencatatan riwayat tugas yang dikerjakan.

PT. Queen Network Nusantara menginginkan absensi menggunakan batasan lokasi dalam pengambilan absensi, absensi dapat dilakukan di dalam area kantor dengan jarak 25 meter dari titik koordinat kantor. Hal ini dapat dicapai dengan cara mendapatkan lokasi dari pegawai dengan menggunakan layanan LBS (*Location Based Service*), dan menghitung jaraknya dengan mengimplementasikan *Haversine Formula*. Sistem informasi yang dapat mencakup semua kebutuhan tersebut adalah aplikasi *mobile*.

Perkembangan *mobile smartphone* di Indonesia sangat pesat, pada awal 2023 sudah ada 353,8 juta pengguna aktif *mobile smartphone*, angka tersebut setara dengan 128 persen dari total populasi di Indonesia yaitu 276,4 juta pada Januari 2023. [3] Aplikasi *mobile* yang dirancang khusus untuk pengelolaan karyawan dapat memudahkan proses pengelolaan absensi dan penugasan pegawai. Dengan menerapkan teknologi tersebut, perusahaan dapat memperbaiki proses kerja dengan melihat kembali riwayat pekerjaan sebagai bahan evaluasi, meningkatkan koordinasi antar pegawai dengan adanya fitur pelaporan tugas secara terbuka untuk semua penerima tugas, dan mempercepat pengambilan keputusan. Aplikasi ini cocok dibuat dengan menggunakan *Framework React Native* karena lingkungan sistem yang sudah ada dalam perusahaan menggunakan *Framework React*.

Penelitian ini penting dilakukan karena PT. Queen Network Nusantara memiliki permasalahan dalam manajemen absensi dan penugasan karyawan yang mengakibatkan adanya potensi kesalahan. Sehingga penelitian harus dilakukan agar dapat memberikan solusi yang tepat sesuai dengan kebutuhan perusahaan.

## **1.2 Tujuan Penelitian**

Tujuan yang ingin dicapai adalah merancang dan membangun aplikasi *android* absensi dan manajemen penugasan karyawan menggunakan *Framework React Native*.

### 1.3 Rumusan Masalah

Rumusan masalah yang dapat diidentifikasi dalam penelitian ini adalah bagaimana merancang dan membangun aplikasi *android* absensi dan manajemen penugasan karyawan menggunakan *React Native* dengan metode PXP.

### 1.4 Batasan Masalah

Dalam penelitian ini, pembatasan masalah meliputi hal-hal sebagai berikut:

1. Spesifikasi aplikasi berdasarkan permintaan PT. Queen Network Nusantara
2. *Output* dari penelitian ini adalah sebuah aplikasi *android*. Penelitian tidak menyediakan *output* berupa aplikasi *backend* atau *service API*.

### 1.5 Sistematika Penulisan

Sistematika penulisan skripsi / tugas akhir ini terdiri dari 5 (lima) bab sebagai berikut:

#### **BAB I : PENDAHULUAN**

Dalam bab ini memuat latar belakang, tujuan penelitian, rumusan masalah, batasan masalah dan sistematika penulisan.

#### **BAB II : TINJAUAN PUSTAKA**

Dalam bab ini memuat penjelasan tentang PT. Queen Network Nusantara, dasar teori Manajemen Sumber Daya Manusia, Sistem Informasi, Absensi, Aplikasi *Mobile*, *Android*, *Framework React Native*, *Javascript*, *Location Based Service (LBS)*, *Global Positioning System (GPS)*, *Haversine Formula*, *Google Maps API*, *Expo Camera*, *Visual Studio Code*, *Personal Extreme Programming*, *Unified Modelling Language*, *Black Box Testing*, dan *State of The Art*.

#### **BAB III : METODOLOGI PENELITIAN**

Dalam bab ini memuat waktu dan tempat penelitian, alat dan bahan penelitian, tahapan penelitian, dan tahapan penelitian. Metode yang digunakan adalah metode *Personal Extreme Programming* yang terdiri dari *Planning*, *Design*, *Coding*, dan *Testing*.

**BAB IV : HASIL DAN PEMBAHASAN**

Dalam bab ini memuat hasil dan pembahasan yang didapat dalam penelitian ini.

**BAB V : KESIMPULAN DAN SARAN**

Dalam bab ini memuat kesimpulan dari hasil penelitian dan saran-saran mengenai perbaikan dan pengembangan lebih lanjut.

## II. TINJAUAN PUSTAKA

### 2.1 Manajemen Sumber Daya Manusia

Manajemen sumber daya manusia (MSDM) merupakan proses perencanaan, pengorganisasian, pengarahan, dan pengendalian SDM di dalam suatu organisasi atau perusahaan. Tujuannya adalah untuk memastikan bahwa organisasi memiliki tenaga kerja yang berkualitas, terampil, dan produktif yang dapat membantu mencapai tujuan strategis organisasi. Menurut Handoko, Manajemen sumber daya manusia merupakan seleksi, penarikan, pengembangan, pemeliharaan, dan penggunaan SDM untuk mencapai tujuan, baik tujuan-tujuan individu maupun organisasi. [4]

Terdapat bermacam pendapat mengenai pengertian manajemen sumber daya manusia, diantaranya ada yang mengartikan sebagai *manpower management* serta ada yang menyetarakan dengan pengertian sumber daya manusia dengan personal (personalia, kepegawaian, dan lainnya). Akan tetapi, pada manajemen sumber daya manusia yang mungkin tepat adalah *human resources management*, dengan demikian secara sederhana pengertian Manajemen Sumber Daya Manusia adalah Mengelola Sumber Daya Manusia. [5]

Menurut Schuler didalam buku Manajemen Sumber Daya Manusia oleh Edy Soetrisno, setidaknya MSDM memiliki tiga tujuan utama [5], yakni:

1. Memperbaiki tingkat produktifitas.
2. Memperbaiki kualitas kehidupan kerja.
3. Meyakinkan organisasi telah memenuhi aspek legal.

## 2.2 PT. Queen Network Nusantara

PT. Queen Network Nusantara merupakan perusahaan bergerak dalam jasa penyedia layanan internet, *backbone*, *maintenance* jaringan internet, multimedia, perangkat lunak, instalasi, dan konstruksi yang didirikan pada tahun 2016 oleh Bapak Supriyanto dan berpusat di Bandar Lampung. Saat ini PT. Queen Network Nusantara sudah memiliki wilayah cakupan di berbagai area di Provinsi Lampung, seperti Metro, Pesawaran, Pringsewu, Tanggamus, Gn. Sugih, dan Lampung Timur.



Gambar 2.2.1 Logo PT. Queen Network Nusantara

Salah satu produk yang dimiliki PT. Queen Network Nusantara adalah produk jaringan internet dari rumah ke rumah atau disebut FTTH (*Fiber to the Home*) yang sudah tersebar di beberapa daerah di kota Bandar Lampung, salah satunya adalah di Kecamatan Sukarame.



Gambar 2.2.2 Foto Kantor PT. Queen Network Nusantara

Nilai nilai perusahaan PT. Queen Network Nusantara adalah nilai nilai yang berorientasi pada kemitraan. Visi dari PT. Queen Network Nusantara adalah “Kami memiliki komitmen yang besar untuk maju dan berkembang”. Misi dari PT. Queen

Network Nusantara adalah selalu menghargai dan menjaga kepercayaan mitra dalam setiap kerja sama yang dijalani, hubungan baik antar mitra akan selalu kami bina walaupun masa kerja sama telah berakhir, yang dijabarkan sebagai berikut:

1. Memberikan solusi yang efektif dan efisien pada setiap jasa yang kami berikan.
2. Mengkaji dengan seksama setiap masalah yang dihadapi guna memberikan solusi yang tepat guna.
3. Memberikan solusi guna meminimalisir resiko bisnis yang dihadapi tanpa mempengaruhi kegiatan usaha itu sendiri. [6]

### **2.3 Sistem Informasi**

Sistem informasi adalah suatu sistem yang terdiri dari elemen-elemen yang saling terkait dan berinteraksi untuk mengumpulkan, mengolah, menyimpan, mengambil, dan menyebarkan informasi guna mendukung pengambilan keputusan, pengendalian, dan operasi suatu organisasi atau bisnis. Sistem informasi adalah gabungan terstruktur dari individu, perangkat keras, perangkat lunak, jaringan komunikasi, dan sumber daya data yang bekerja bersama untuk mengumpulkan, mengolah, dan menyampaikan informasi di dalam suatu organisasi. [7]

Sistem informasi dapat diartikan sebagai sekelompok subsistem yang saling terkait, bekerja bersama untuk membentuk satu kesatuan, berinteraksi, dan saling bekerjasama dengan metode tertentu dalam melakukan fungsi pengolahan data. Sistem ini menerima input berupa data, mengolahnya, dan menghasilkan output dalam bentuk informasi yang digunakan sebagai dasar untuk pengambilan keputusan yang memiliki nilai signifikan, baik pada saat itu maupun di masa mendatang. [8]

Sistem yang baik memiliki karakteristik sebagai berikut [8]:

#### **1. Komponen**

Suatu sistem yang terdiri dari sejumlah komponen yang berinteraksi satu sama lain, saling bekerja sama membentuk satu kesatuan. Komponen sistem ini dapat berupa subsistem atau bagian-bagian dari sistem tersebut.

2. Batasan sistem (*boundary*)

Batasan sistem adalah wilayah yang memisahkan suatu sistem dari sistem lainnya atau dari lingkungan luar. Batasan sistem tersebut memungkinkan sistem dipandang sebagai suatu kesatuan. Batasan sistem menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem (*enviromtent*)

Lingkungan luar sistem adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan dan merugikan sehingga harus dijaga dan dikendalikan.

4. Penghubung sistem (*interface*)

Penghubung sistem adalah sarana yang menghubungkan satu subsistem dengan subsistem lainnya. Melalui penghubung ini, memungkinkan aliran sumber daya dari satu subsistem ke subsistem lainnya.

5. Masukkan sistem (*input*)

Masukkan sistem adalah energi yang dimasukkan kedalam sistem, yang dapat berupa perawatan (*maintenance input*) dan masukkan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran.

6. Keluaran sistem (*output*)

Keluaran sistem adalah hasil dari proses pengolahan dan pengelompokan energi yang menghasilkan informasi yang berguna dan sisa pembuangan.

## 2.4 Absensi

Absensi adalah suatu pencatatan kehadiran yang merupakan bagian dari kegiatan pelaporan dalam sebuah institusi atau organisasi. Absensi diatur dan disusun sedemikian rupa sehingga dapat dengan mudah dicari dan digunakan oleh pihak yang membutuhkannya. [9]



Gambar 2.4.1 Contoh alat sistem absensi menggunakan identifikasi sidik jari

Absensi merupakan proses pencatatan kehadiran atau ketidakhadiran seseorang pada suatu kegiatan, biasanya absensi digunakan dalam konteks kehadiran karyawan dalam sebuah perusahaan. Absensi memiliki tujuan untuk memastikan dan memonitor kehadiran orang-orang yang seharusnya hadir pada suatu kegiatan. Absensi dapat dilakukan dengan berbagai cara, seperti menggunakan perekaman manual dengan buku absensi, menggunakan sistem perekaman elektronik, dan menggunakan identifikasi biometrik seperti sidik jari atau pemindai wajah.

Absensi memiliki peranan penting dalam MSDM, seperti memonitor kehadiran dan ketidakhadiran karyawan secara efektif, sehingga manajer dapat mengambil tindakan yang tepat dengan mengidentifikasi karyawan yang sering tidak masuk atau terlambat. Selain itu, dengan adanya absensi dapat memperkecil kemungkinan kecurangan dan penipuan terkait kehadiran karyawan.

## 2.5 Aplikasi *Mobile*

Aplikasi merupakan sebuah program komputer atau perangkat lunak yang dirancang untuk melakukan suatu tugas tertentu pada komputer atau perangkat *mobile*. Sedangkan *mobile* merujuk pada kemampuan untuk bergerak atau berpindah pindah dengan mudah. Dalam konteks teknologi informasi, *mobile* merupakan perangkat elektronik portabel yang bisa dibawa kemana mana, seperti ponsel pintar dan tablet.

Berdasarkan pengertian tersebut, dapat disimpulkan bahwa aplikasi *mobile* merupakan sebuah program perangkat lunak yang dirancang khusus untuk dijalankan pada perangkat *mobile*, seperti ponsel pintar dan tablet.

## 2.6 *Android*

*Android* merupakan OS berbasis Linux yang dirancang untuk perangkat layar sentuh seperti *smartphone* dan tablet. Awalnya *Android* dikembangkan oleh *Android, Inc.*, dengan dukungan dari Google, Google kemudian membelinya pada tahun 2005. [10]



Gambar 2.6.1 Logo *Android*

*Android* secara sederhana bisa diartikan sebagai sebuah *software* yang digunakan pada perangkat *mobile* yang mencakup sistem operasi, *middleware*, dan aplikasi kunci yang dirilis oleh Google. Pengembangan aplikasi pada *platform Android* ini menggunakan dasar bahasa pemrograman Java. *Platform* pengembangan aplikasi *Android* bersifat *open-source*, sehingga siapapun dapat mengembangkan kemampuan untuk membangun aplikasi yang kaya dan inovatif. Bahkan seorang pengembang *Android* dapat membuat aplikasi yang bervariasi, kemudian menjual untuk keuntungan pribadi tanpa lisensi ke produsen atau vendor tertentu. [11]

Dalam perkembangannya, sistem operasi *android* mengalami banyak perubahan versi, diantaranya dapat dilihat pada Tabel 2.6.1 [12]:

Tabel 2.6.1 Versi *Android*

Versi	SDK / API Level	Nama Kode	Tahun
<i>Android 14</i>	Level 34	<i>Upside Down Cake</i>	TBD
<i>Android 13</i>	Level 33	<i>Tiramisu</i>	2022
<i>Android 12</i>	Level 32 ( <i>Android 12L</i> )	<i>Snow Cone</i>	2021
	Level 31 ( <i>Android 12</i> )		
<i>Android 11</i>	Level 30	<i>Red Velvet Cake</i>	2020
<i>Android 10</i>	Level 29	<i>Quince Tart</i>	2019
<i>Android 9</i>	Level 28	<i>Pie</i>	2018
<i>Android 8</i>	Level 27 ( <i>Android 8.1</i> )	<i>Oreo</i>	2017
	Level 26 ( <i>Android 8.0</i> )		
<i>Android 7</i>	Level 25 ( <i>Android 7.1</i> )	<i>Nougat</i>	2016
	Level 24 ( <i>Android 7.0</i> )		
<i>Android 6</i>	Level 23	<i>Marshmallow</i>	2015
<i>Android 5</i>	Level 22 ( <i>Android 5.1</i> )	<i>Lollipop</i>	2014
	Level 21 ( <i>Android 5.0</i> )		
<i>Android 4</i>	Level 20 ( <i>Android 4.4W</i> )	<i>KitKat</i>	2013
	Level 19 ( <i>Android 4.4</i> )		
	Level 18 ( <i>Android 4.3</i> )	<i>Jelly Bean</i>	2012
	Level 17 ( <i>Android 4.2</i> )		
	Level 16 ( <i>Android 4.1</i> )		
	<i>Android 3</i>	Level 15 ( <i>Android 4.0.3 – 4.0.4</i> )	<i>Ice Cream</i>
Level 14 ( <i>Android 4.0.1 – 4.0.2</i> )		<i>Sandwich</i>	
<i>Android 3</i>	Level 13 ( <i>Android 3.2</i> )	<i>Honeycomb</i>	2010
	Level 12 ( <i>Android 3.1</i> )		
	Level 11 ( <i>Android 3.0</i> )		
<i>Android 2</i>	Level 10 ( <i>Android 2.3.3 – 2.3.7</i> )	<i>Gingerbread</i>	2010
	Level 9 ( <i>Android 2.3.0 – 2.3.2</i> )		
	Level 8 ( <i>Android 2.2</i> )	<i>Froyo</i>	2009
	Level 7 ( <i>Android 2.1</i> )	<i>Eclair</i>	
	Level 6 ( <i>Android 2.0.1</i> )		

	Level 5 ( <i>Android 2.0</i> )		
<i>Android 1</i>	Level 4 ( <i>Android 1.6</i> )	<i>Donut</i>	
	Level 3 ( <i>Android 1.5</i> )	<i>Cupcake</i>	
	Level 2 ( <i>Android 1.1</i> )	<i>Petit Four</i>	
	Level 1 ( <i>Android 1.0</i> )		2008

*Android SDK (Software Development Kit)* merupakan *tools API (Application Programming Interface)* yang diperlukan untuk memulai pengembangan aplikasi pada *platform Android*. Pada *Android SDK* ini terdiri dari *debugger, libraries, handset emulator*, dan dokumentasi. SDK memungkinkan pengembang membuat aplikasi untuk *platform Android SDK, Android* mencakup proyek sampel dengan kode sumber, perangkat pengembangan, emulator dan perpustakaan yang diperlukan untuk membangun aplikasi *Android*. [13]

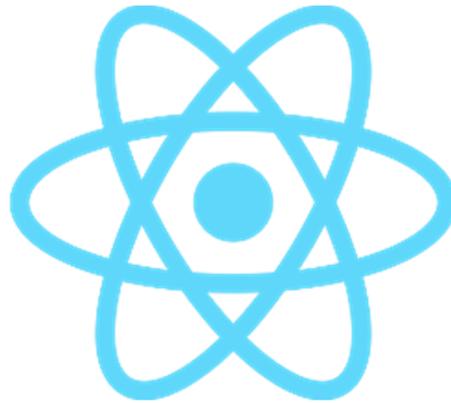
Dalam mengembangkan aplikasi mobile absensi dan penugasan karyawan, digunakan *android* versi 13 dengan SDK level 33.

## 2.7 *Framework React Native*

*React Native* adalah *framework opensource* yang dibuat oleh facebook dan dirilis pada tahun 2015, *React Native* merupakan *framework* yang digunakan untuk membuat aplikasi *multi-platform* dengan bahasa *javascript*, sesuai dengan penjelasan di situs resminya “*Learn once, write anywhere*”. Pada tahun 2018, *React Native* memiliki jumlah kontributor tertinggi ke-2 untuk setiap repositori di GitHub. Saat ini, *React Native* didukung oleh individu dan perusahaan di seluruh dunia termasuk Callstack, Expo, Infinite Red, Microsoft, dan Software Mansion. [14]

*React Native* merupakan sebuah *Framework Javascript* untuk membangun aplikasi *mobile Android* dan iOS. *React Native* memiliki dasar dari React, yaitu *Library Javascript* milik Facebook yang digunakan untuk membangun antarmuka, berbeda dengan *React* yang menargetkan *web browser*, *React Native* menargetkan *platform mobile*. Sama seperti *React* untuk *web*, *React Native* ditulis menggunakan campuran antara *javascript* dan *markup* mirip XML, yang disebut JSX. Lalu *React*

*Native* ini juga memaparkan antarmuka *javascript* untuk *platform API* dimana pengembang dalam membangun aplikasi ini dapat mengakses fitur-fitur seperti kamera, lokasi, dll yang ada pada ponsel. [15]



Gambar 2.7.1 Logo *React*

*React Native* menggunakan *JavaScript* untuk membangun aplikasi *mobile* dengan tampilan dan fungsionalitas seperti aplikasi *native*. Dalam *React Native*, *JavaScript* digunakan untuk membuat UI dan mengatur logika aplikasi. *Framework* ini memiliki banyak dukungan dan *library* yang memudahkan pengembangan aplikasi *mobile* dengan *JavaScript*.

## 2.8 *Javascript*

*Javascript* adalah bahasa yang memiliki bentuk berupa kumpulan skrip yang berfungsi untuk memberikan tampilan agar lebih interaktif pada dokumen web. *Javascript* merupakan bahasa skrip yang dipakai untuk menciptakan halaman web yang dapat berinteraksi dengan pengguna dan dapat merespon *event* yang terjadi pada halaman. *Javascript* dikembangkan oleh Brenden Eich di Netscape pada tahun 1995. *Javascript* merupakan bahasa pada sisi klien yang didesain pada browser, ia dibangun secara langsung ke dalam browser. [16]

## 2.9 *Location Based Service (LBS)*

*Location Based Service (LBS)*, yang juga dikenal sebagai Layanan Berbasis Lokasi, adalah istilah yang merujuk pada teknologi yang digunakan untuk melacak lokasi perangkat yang sedang digunakan. LBS merupakan layanan informasi yang dapat diakses melalui *mobile device* dengan menggunakan *mobile network*, dan

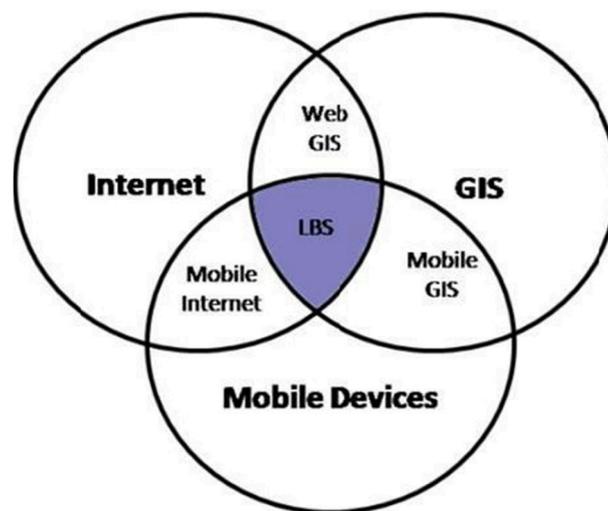
dilengkapi kemampuan untuk memanfaatkan lokasi dari *mobile device* tersebut [17]. LBS memiliki dua unsur, yakni:

a. *Location Manager (API Maps)*

*Application Programming Interface (API) Maps* menyediakan fasilitas untuk menampilkan, memanipulasi maps/peta dan fitur lainnya seperti menampilkan satelit, jalan, atau gabungan keduanya.

b. *Location Provider (API Location)*

*API Location* adalah teknologi yang digunakan untuk mencari lokasi yang digunakan oleh perangkat. API ini berhubungan dengan data GPS (*Global Positioning System*) dan data lokasi *real-time*. *API Location* berada pada paket *android* yaitu dalam paket *android.location*. *Location Manager* dapat menentukan lokasi saat ini, *track* gerakan/perpindahan, dan kedekatan dengan suatu lokasi dengan mendeteksi perpindahan [17]



Gambar 2.9.1 Teknologi *Location Based Services*

Dalam menggunakan LBS memerlukan beberapa elemen penting, yaitu [17]:

- a. *Mobile Device*, yaitu alat yang digunakan untuk meminta informasi yang dibutuhkan. Biasanya perangkat yang memungkinkan yaitu PDA, Mobile Phone, Laptop, dan perangkat lainnya yang mempunyai fasilitas navigasi.
- b. *Communication Network*, merupakan jaringan selular yang mengirimkan data pengguna dan permintaan layanan.

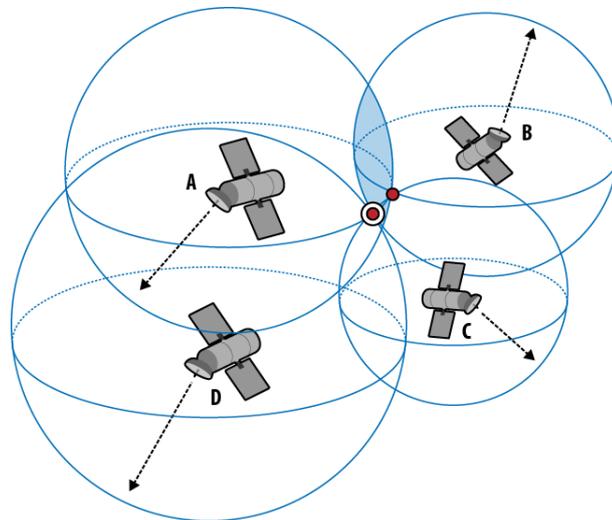
- c. *Positioning Component*, ketika mengolah layanan posisi, penting untuk menentukan posisi pengguna. Posisi pengguna bisa diperoleh melalui jaringan komunikasi atau menggunakan *Global Positioning System* (GPS). Selain GPS, media untuk mengambil posisi dapat ditentukan melalui menara seluler dan kombinasi antara GPS dan menara seluler (aGPS).
- d. *Service and Application Provider*, adalah penyedia layanan pengguna selular yang bertanggung jawab untuk memproses layanan. *Provider* ini melakukan proses komputasi seperti menemukan rute perjalanan, mencari informasi tentang tempat terdekat, serta mengakses database eksternal seperti *Yellow Pages* atau *Google API* untuk menghasilkan informasi yang dibutuhkan oleh pengguna.
- e. *Data and Content Provider*, yaitu pihak yang menyediakan data yang dapat diminta oleh pengguna. Penyedia layanan ini tidak selalu menyimpan seluruh data dan informasi yang mereka olah, karena data dan informasi tersebut seringkali berasal dari pengembang atau pihak ketiga yang memiliki otoritas untuk menyimpannya.

## 2.10 *Global Position System (GPS)*

*Global Positioning System* (GPS) merupakan suatu kumpulan satelit dan sistem kontrol yang memungkinkan sebuah penerima GPS untuk mendapatkan lokasinya di permukaan bumi 24 jam sehari. [18] *Global Positioning System* (GPS) digunakan untuk menentukan posisi pengguna di permukaan bumi dengan sinkronisasi sinyal satelit. Sistem ini menggunakan minimal 4 satelit yang mengirimkan sinyal gelombang mikro ke Bumi. Sinyal kemudian diterima oleh alat penerima dan digunakan untuk menentukan posisi, arah, kecepatan, dan waktu [19]

Pesawat penerima GPS menggunakan sinyal satelit untuk melakukan triangulasi posisi yang hendak ditentukan dengan cara mengukur lama perjalanan waktu sinyal dikirimkan dari satelit, kemudian mengalikannya dengan kecepatan cahaya untuk menentukan secara tepat berapa jauh pesawat penerima GPS dari setiap satelit. Dengan mengunci sinyal yang ditransmit oleh satelit minimum 3 sinyal dari satelit yang berbeda, pesawat penerima GPS dapat menghitung posisi tetap sebuah titik

yaitu posisi lintang dan bujur bumi (*Latitude & Longitude*) atau sering disebut dengan *2D fix*. Penguncian sinyal satelit yang keempat membuat pesawat 7 penerima GPS dapat menghitung posisi ketinggian titik tersebut terhadap muka laut rata-rata (Mean Sea/Level) atau disebut *3D fix* dan keadaan ini yang ideal untuk melakukan navigasi atau GPS juga bisa diartikan adalah singkatan dari *Global Positioning System* yang merupakan sistem untuk menentukan posisi dan navigasi secara global dengan menggunakan satelit dan metode Triangulasi. [20] Dalam penelitian ini, GPS digunakan untuk mengambil posisi *latitude* dan *longitude* dari smartphone yang dipakai.



Gambar 2.10.1 *Global Positioning System* Menggunakan Satelit

### 2.11 *Haversine Formula*

*Haversine Formula* adalah salah satu metode yang digunakan untuk menghitung jarak antara dua buah koordinat pada peta dua dimensi dengan mengabaikan tinggi gunung dan kedalaman lembah. *Haversine formula* biasa juga dikenal dengan istilah *great circle distance*, rumus ini melakukan perhitungan dari titik utama ke titik tujuan dengan fungsi trigonometri dengan menggunakan *latitude* dan *longitude* yang pada akhirnya akan menghasilkan jarak antara kedua titik. Metode ini digunakan untuk aplikasi dalam sistem informasi geografis [21].

Formula ini pertama kali ditemukan oleh Jamez Andrew di tahun 1805, dan digunakan pertama kali oleh Josef de Mendoza y Ríos di tahun 1801. Istilah *haversine* ini sendiri diciptakan pada tahun 1835 oleh Prof. James Inman. Josef de Mendoza y Ríos menggunakan *haversine* pertama kali dalam penelitiannya tentang “Masalah Utama Astronomi Nautical”, Proc. Royal Soc, Dec 22. 1796. *Haversine* digunakan untuk menemukan jarak antar bintang [22]. Berikut merupakan rumus *haversine* [23]:

$$d = 2r \arcsin \left( \sqrt{\sin^2 \left( \frac{\varphi_2 - \varphi_1}{2} \right) + \cos \varphi_1 \cdot \cos \varphi_2 \cdot \sin^2 \left( \frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

Dimana:

- d : Jarak antara dua titik (*Distance*)
- r : Jari – jari (*Radius*) Bumi, 6371 Km
- $\varphi$  : *Latitude*, dalam bentuk radian
- $\lambda$  : *Longitude*, dalam bentuk radian

Penerapan Rumus *haversine* dalam bidang IT dapat dibuat dalam bentuk berikut ini [24]:

$$a = \sin^2 \left( \frac{\Delta\varphi}{2} \right) + \cos \varphi_1 \cdot \cos \varphi_2 \cdot \sin^2 \left( \frac{\Delta\lambda}{2} \right)$$

$$c = 2 \cdot \operatorname{atan2} \left( \sqrt{a}, \sqrt{1-a} \right)$$

$$d = r \cdot c$$

Hasil dari perhitungan rumus *haversine* akan menghasilkan jarak antara kedua buah titik dalam bentuk kilometer. Dalam penelitian ini, rumus *haversine* diterapkan untuk menentukan jarak antara titik koordinat pengguna dengan titik koordinat kantor.

## 2.12 Google Maps API

*Application Programming Interface* (API) adalah kumpulan fungsi-fungsi bahasa pemrograman yang disediakan oleh aplikasi atau layanan agar bisa diintegrasikan

dengan aplikasi atau website yang dibuat. API memiliki tujuan untuk menghapus “*clueless*” dari sebuah sistem dengan membuat blok besar yang terdiri dari software di seluruh dunia dan menggunakan kembali perintah, fungsi, class, atau protokol yang mereka atau API miliki. Sehingga programmer tidak perlu membuang waktu untuk membuat infrastruktur dan akan menghemat waktu kerja. [25]

*Google Maps* adalah sebuah layanan peta digital yang disediakan oleh Google dan bersifat *open-source*. *Google Maps* dapat dikembangkan sesuai dengan keinginan *developer* dengan menggunakan *library* yang telah disediakan oleh Google. Selain itu, *Google Maps* dapat diintegrasikan ke aplikasi lain baik berbasis *mobile* maupun berbasis *web* dengan menggunakan *Google Maps API*.

*Google Maps API* adalah layanan yang dibuat oleh Google untuk memanfaatkan *Google Maps* dalam pengembangan aplikasi [26]. Untuk dapat menggunakan *Google Maps API* diperlukan *token (API Keys)* yang nantinya akan disisipkan pada *code* aplikasi atau sistem yang dibangun. *Token* atau *API Keys* adalah kode unik yang dikeluarkan oleh Google untuk memberikan izin akses *Google Maps API*. *Token* tersebut didapatkan dari halaman resmi *Google Maps*. *Google Maps API* sendiri memungkinkan user atau *developer* untuk menampilkan peta suatu wilayah, memberikan arahan rute tercepat kepada user, menampilkan alamat dan posisi *latitude longitude* dari sebuah tempat dan dapat memberikan perhitungan waktu dari satu rute ke rute lainnya. Dengan *Google Maps API* waktu dan biaya dapat dihemat untuk membangun peta digital yang dapat ditambahkan pada *web* yang dibangun. [27] *Google Maps API* digunakan untuk menentukan nama alamat berdasarkan nilai *latitude* dan *longitude* (koordinat) yang didapatkan dari GPS.

### **2.13 Expo Camera**

*Expo Camera* adalah modul kamera yang disediakan oleh Expo, sebuah *platform* pengembangan aplikasi yang berbasis *React Native*. *Expo Camera* memungkinkan pengembang untuk mengakses fitur kamera pada perangkat *mobile*, seperti mengambil foto dan merekam video. Modul tersebut juga mampu mendeteksi wajah. [28]

### 2.14 *Visual Studio Code*

*Visual Studio Code* adalah perangkat lunak *editor source code* yang dibuat oleh Microsoft untuk Windows, MacOS, dan Linux. *Visual Studio Code* memiliki fitur fitur pendukung untuk *debugging*, *intelligent code complete*, *syntax highlighting*, *code refactoring*, *snippet*, dan *embedded Git*. *Visual Studio Code* pertama kali diperkenalkan pada tanggal 29 April 2015. Pada 18 November 2015, *Visual Studio Code* dirilis di bawah Lisensi MIT, dan sudah tersedia di GitHub. *Visual studio code* digunakan sebagai satu satunya *code editor* yang dipakai selama pengembangan aplikasi.

### 2.15 *Personal Extreme Programming*

Metode *Personal Extreme Programming* atau dikenal dengan metode PXP. Metode ini dicetuskan oleh Kent Beck, seorang pakar *software engineering*. *Personal Extreme Programming* merupakan model pengembangan perangkat lunak yang menyederhanakan tahapan pengembangan sistem sehingga menjadi lebih efisien, adaptif, dan fleksibel. Nilai dasar metode *Personal Extreme Programming*:

1. *Communication*, Memfokuskan komunikasi yang baik antara programmer dengan user maupun antar programmer.
2. *Courage*, Pengembang perangkat lunak harus selalu memiliki keyakinan, keberanian dan integritas dalam melakukan tugasnya.
3. *Simplicity*, Lakukan semua dengan sederhana.
4. *Feedback*, Mengandalkan feedback sehingga dibutuhkan anggota tim yang berkualitas.
5. *Quality Work*, Proses berkualitas berimplikasi pada perangkat lunak yang berkualitas sebagai hasil akhirnya. [29]

PXP membalikkan proses perangkat lunak konvensional, daripada merencanakan, menganalisis, dan merancang untuk sesuatu yang terlalu jauh, PXP melakukan pengurangan biaya dengan melakukan aktivitasnya sedikit demi sedikit selama pengembangan perangkat lunak [30]. PXP merupakan model pengembangan perangkat lunak yang tergolong dalam pendekatan agile. Perbedaan antara PXP

dengan model pengembangan lain adalah metode ini sangat cocok dan terbuka terhadap perubahan kebutuhan sistem yang cepat.

### **2.16 Unified Modelling Language**

*Unified Modeling Language* (UML) adalah bahasa standar untuk membuat cetak biru atau model suatu perangkat lunak. *Unified Modeling Language* adalah proses mendokumentasikan dan mespesifikasikan sistem menggunakan bahasa grafis untuk mengembangkan sistem. [31]

UML berfungsi sebagai alat untuk menggambarkan dan mengkomunikasikan desain sistem secara lebih jelas dan efektif, sehingga memudahkan para pengembang perangkat lunak untuk memahami kebutuhan bisnis dan teknis suatu sistem. UML digunakan untuk memodelkan berbagai jenis sistem, seperti sistem informasi, sistem pengolahan transaksi, dan sistem berorientasi objek. UML terdiri dari sejumlah diagram yang dapat digunakan untuk merepresentasikan berbagai aspek dari sebuah sistem, diantaranya:

1. Diagram *Use Case*, merupakan diagram untuk memvisualisasikan relasi antara aktor dengan sistem secara global dan lebih difokuskan pada fungsionalitas sistem. [31]
2. Diagram *Activity*, merupakan diagram sebagai visualisasi dari kegiatan pada *use case*, proses bisnis, logika dan relasi antar aktor bersama *work flow use case*. [31]

### **2.17 Black Box Testing**

*Black Box Testing* merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak, penguji tidak memiliki pengetahuan tentang desain internal dan tidak memiliki akses ke *source code*. Teknik ini digunakan untuk memastikan bahwa semua input yang dibutuhkan oleh sistem diterima dengan cara yang ditentukan dan memberikan *output* yang benar [32]. Keuntungan penggunaan metode *Blackbox Testing* adalah [33]:

1. Penguji tidak harus mengetahui tentang bahasa pemrograman tertentu.

2. Pengujian dilakukan dari perspektif pengguna untuk membantu mengidentifikasi ambiguitas atau inkonsistensi dalam spesifikasi persyaratan.
3. Programmer dan tester keduanya saling bergantung satu sama lain.

### **2.18 State of The Art**

*State of the art* merupakan hasil analisa penelitian sebelumnya yang memiliki konsep berkaitan dengan sistem yang dikembangkan. Penelitian sebelumnya digunakan sebagai bahan analisa guna memperkaya pembahasan referensi penelitian, serta membedakannya dengan penelitian yang saat ini sedang dilakukan. Penelitian ini disertakan beberapa referensi yang berhubungan dengan absensi *mobile* yang telah dijelaskan pada poin Penelitian Terkait. Penelitian dilakukan pada pembuatan aplikasi *mobile*, terdapat tim lain yang mengerjakan bagian *backend* sistem dan *service API*, salah satunya adalah fitur pengenalan wajah. Terdapat beberapa penelitian terkait yang dijadikan sebagai perbandingan dan acuan pada penelitian ini. Berikut merupakan beberapa penelitian terkait:

#### **2.18.1 Research on Mobile Phone Attendance Positioning System Based on Campus Network**

Penelitian dengan judul “*Research on Mobile Phone Attendance Positioning System Based on Campus Network*”. Diambil dari *International Conference on Smart Grid and Electrical Automation (ICSGEA)*, diteliti oleh Zhen Xu, Ping Chen, Weihua Zhang, Xing Liu, dan Heqing Wu pada tahun 2019 di China yang membahas mengenai implementasi metode absensi menggunakan aplikasi *mobile* dan sistem penentuan posisi berdasarkan jaringan kampus dalam verifikasi siswa. Penelitian ini mengusulkan metode untuk merekam dan melacak pengguna jaringan kampus secara online dengan menghubungkan informasi seperti *IP Adress*, *MAC Address*, akun pengguna, dan *port switch*. Hasil penelitian ini menyelesaikan dua masalah utama, yaitu verifikasi identitas dan sistem penentuan posisi absensi *mobile*. [34]

Penelitian dengan judul “*Research on Mobile Phone Attendance Positioning System Based on Campus Network*” menggunakan jaringan kampus sebagai salah satu verifikasi absensi, sehingga absensi dapat dilakukan didalam area tertentu. Hal ini

yang kemudian dikembangkan dalam penelitian ini yaitu sistem yang dapat melakukan absensi dari dalam area yang ditentukan ataupun luar area jika diizinkan, sehingga membuat pegawai yang banyak bekerja di luar area dapat melakukan absensi dengan mudah.

### **2.18.2 *Multifactor Authentication on Mobile Secure Attendance System***

Penelitian dengan judul “*Multifactor Authentication on Mobile Secure Attendance System*”. Diambil dari *International Conference on ICT for Smart Society (ICISS)*, diteliti oleh Subroto Budhi Utomo dan Bayu Hendradjaya pada tahun 2018 di Bandung, Indonesia yang membahas mengenai sistem absensi *mobile* yang dapat digunakan untuk melakukan pencatatan kehadiran serta pemantauan karyawan selama jam kerja. Penelitian ini telah merancang proses autentikasi dengan menggunakan *Secure Android ID* sebagai pengganti *username* dan sidik jari sebagai pengganti *password*, GPS dan geofence juga digunakan untuk memastikan pegawai benar-benar hadir di tempat yang telah ditentukan, penambahan modul *anti emulator* dan *anti fake location* juga dilakukan dalam sistem ini. [35]

Penelitian dengan judul “*Multifactor Authentication on Mobile Secure Attendance System*” menggunakan *Secure Android ID* dan sidik jari, sehingga pegawai dapat melakukan absensi pada satu *device* saja, kelebihan dari sistem ini adalah keamanan dari akun pengguna lebih terjaga karena autentikasi didesain pada satu *device* saja, namun memiliki kekurangan ketika *device* pengguna mengalami kerusakan, pengguna harus mendaftarkan *device* kembali untuk melakukan absensi. Hal ini yang menjadi perbedaan dengan penelitian yang dilakukan, pada penelitian yang dilakukan digunakan metode autentikasi menggunakan *username* dan *password*, sehingga dapat memudahkan pengguna untuk melakukan absensi melalui *device* mana saja.

### **2.18.3 *Android Application for Presence Recognition based on Face and Geofencing***

Penelitian dengan judul “*Android Application for Presence Recognition based on Face and Geofencing*”. Diambil dari *International Seminar on Application for*

*Technology of Information and Communication (iSemantic)*, diteliti oleh Alvin Syarifudin Shahab dan Riyanarto Samo pada tahun 2020 di Surabaya, Indonesia yang membahas mengenai pengembangan sistem absensi yang fleksibel dan mempercepat proses absensi dengan menggunakan aplikasi mobile berbasis *geofencing* dan pengenalan wajah. Sistem mengambil gambar dari karyawan saat karyawan berada di dalam area *geofencing*. Data wajah dikirim ke server untuk pengenalan menggunakan PCA dan k-NN dengan hasil akurasi 90% dan waktu pemrosesan 1,5 detik lebih baik dari pengenalan LBPH dan NN. [36]

Penelitian dengan judul “*Android Application for Presence Recognition based on Face and Geofencing*” menggunakan pengenalan wajah sebagai verifikasi absensi. Hal ini memiliki persamaan dengan penelitian yang dilakukan, penelitian menggunakan pengenalan wajah sebagai salah satu verifikasi absensi.

### III. METODOLOGI PENELITIAN

#### 3.1 Waktu dan Tempat

Penelitian dilaksanakan mulai dari bulan Januari 2023 sampai dengan bulan Mei 2023 yang bertempat di kantor PT. Queen Network Nusantara.

Tabel 3.1.1 Jadwal Penelitian

No	Aktivitas	Jan-23	Feb-23	Mar-23	Apr-23	Mei-23
1	Penentuan Kebutuhan					
2	<i>Planning</i>					
3	<i>Design</i>					
4	<i>Coding</i>					
5	<i>Testing</i>					
6	Iterasi Sistem					
7	Pelaporan					

#### 3.2 Alat dan Bahan Penelitian

##### 3.2.1 Alat Penelitian

Adapun alat yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.2.1 berikut:

Tabel 3.2.1 Alat Penelitian

No.	Nama Alat	Spesifikasi	Deskripsi
1.	Laptop	Prosesor Intel Core i5-8250U, Memory 12GB, VGA Nvidia GeForce MX 130, sistem operasi Windows	Perangkat keras untuk membangun aplikasi
2.	<i>Smartphone Android</i>	<i>Android</i> Versi 13	Perangkat keras untuk meng- <i>install</i> aplikasi

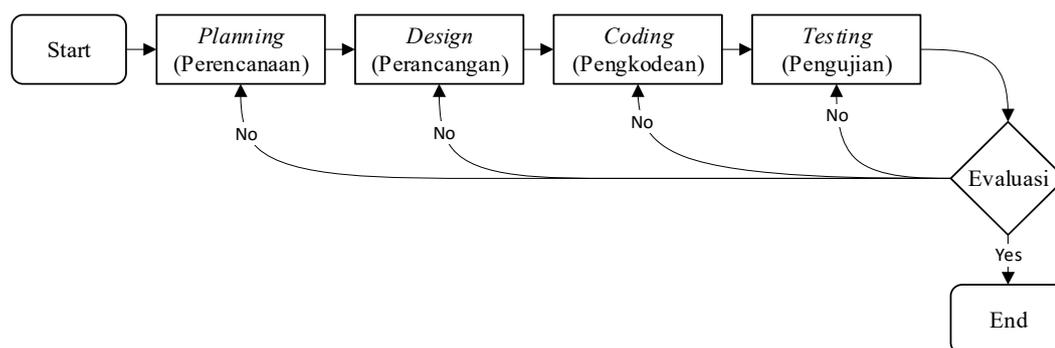
3.	Visual Studio Code	Versi 1.78.2	Perangkat lunak untuk menuliskan baris kode aplikasi
4.	<i>React Native</i>	Versi 0.70.5	<i>Framework</i> untuk membuat aplikasi <i>multifplatform</i>
5.	Figma	Versi 28.4.12.3	Perangkat lunak untuk merancang antarmuka aplikasi
6.	<i>Expo Camera</i>	Versi 13.1.0	<i>Library</i> untuk merekam dan deteksi wajah

### 3.2.2 Bahan Penelitian

Bahan yang digunakan dalam penelitian adalah data pegawai PT. Queen Network Nusantara dan akses database yang disediakan oleh PT. Queen Network Nusantara.

### 3.3 Tahapan Penelitian

Rancang bangun aplikasi *mobile* absensi dan manajemen penugasan karyawan ini dilakukan dengan berbagai tahapan dengan tujuan untuk memudahkan penulisan penelitian. Penelitian dilakukan menggunakan metode *Personal Extreme Programming*. Adapun tahapan yang dilakukan dalam penelitian dapat dilihat pada Gambar 3.3.1.



Gambar 3.3.1 Tahapan Penelitian

Berdasarkan gambar tersebut, penjabaran metode yang digunakan berdasarkan tahapan penelitian adalah sebagai berikut:

### **3.3.1 *Planning* (Perencanaan)**

Pada tahap pertama, yaitu tahap dimana dijelaskan semua aspek kebutuhan sistem yang ingin dicapai di masa mendatang. Setiap kebutuhan tersebut dirumuskan dalam bentuk *User Stories*.

### **3.3.2 *Design* (Perancangan)**

Pada tahap ini, dilakukan pembuatan model sistem berdasarkan hasil analisa kebutuhan pada tahap perencanaan. Hal ini bertujuan untuk memberikan gambaran visual bagaimana sistem akan berfungsi. Pemodelan sistem yang digunakan yaitu *Unified Modelling Language* (UML) menggunakan *Use Case Diagram* dan *Activity Diagram*.

### **3.3.3 *Coding* (Pengkodean)**

Pada tahap ini, merupakan implementasi kode program berdasarkan rancangan model sistem dan kebutuhan klien. Kode yang dibuat harus selalu diuji dan diperbaiki ketika ditemukan kesalahan. Dalam pembangunan aplikasi ini, menggunakan *Framework React Native*.

### **3.3.4 *Testing* (Pengujian)**

Pada tahap ini, dilakukan pengujian terhadap aplikasi. Hal ini dilakukan untuk memastikan aplikasi dapat berfungsi dengan baik dan menemukan kesalahan-kesalahan. Pengujian dilakukan menggunakan metode *black-box testing*, dan berfokus pada fungsionalitas dari aplikasi.

## **3.4 Perencanaan (*Planning*)**

Pada tahap ini, dilakukan analisa untuk menentukan kebutuhan *user* dan kebutuhan sistem yang akan dibangun. Kebutuhan sistem diperoleh dari hasil identifikasi dari *user story*.

### 3.4.1 User Stories

Pada bagian ini, menggambarkan kebutuhan dari pengguna yang dituliskan dalam bentuk naratif sederhana agar kebutuhan pengguna dapat dipahami dengan jelas. Berikut merupakan tabel *user stories* pegawai.

Tabel 3.4.1 *User Story*

No.	Aktor	User Story
1.	Pegawai	Sebagai seorang pegawai saya ingin aplikasi dapat melakukan otentikasi, sehingga tidak ada campur tangan dari pihak lain
		Sebagai seorang pegawai saya ingin aplikasi dapat memudahkan saya untuk mencatat waktu masuk dan keluar kerja, serta memberikan kemudahan untuk mengajukan izin atau cuti melalui aplikasi tersebut
		Sebagai seorang pegawai saya ingin aplikasi memiliki fitur notifikasi sebagai pengingat.
		Sebagai seorang pegawai saya ingin aplikasi dapat mengubah <i>password</i> dan profil untuk mempersonalisasi akun saya
		Sebagai seorang pegawai saya ingin aplikasi dapat menampilkan catatan kehadiran, sehingga saya dapat memantau kehadiran setiap bulannya
		Sebagai seorang pegawai saya ingin aplikasi dapat melihat jadwal kegiatan di beberapa hari kedepan dan ringkasan kegiatan yang saya lakukan pada bulan ini
		Sebagai seorang pegawai saya ingin aplikasi yang dapat membagikan tugas kepada anggota tim dan dapat melihat tugas yang harus saya kerjakan
		Sebagai seorang pegawai saya ingin aplikasi dapat melihat status tugas dan riwayat pelaporan tugas, sehingga saya dapat memantau dan memberikan <i>feedback</i> kepada anggota tim

### 3.4.2 Analisis Kebutuhan

Berdasarkan hasil identifikasi *user stories*, maka didapatkan kebutuhan dari sistem sebagai berikut:

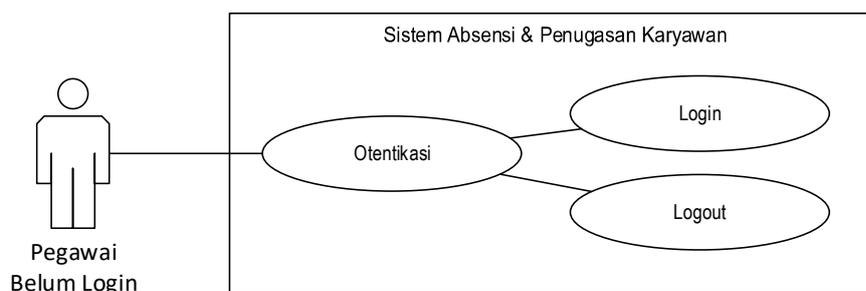
1. Pengguna dapat melakukan otentikasi aplikasi.
2. Pengguna dapat mencatat waktu masuk dan keluar (melakukan absensi).
3. Pengguna dapat mengajukan izin atau cuti melalui aplikasi.
4. Pengguna dapat menerima pemberitahuan atau pengingat melalui notifikasi.
5. Pengguna dapat mengubah *password* akun.
6. Pengguna dapat mengubah profil akun.
7. Pengguna dapat melihat catatan riwayat kehadiran secara lengkap setiap bulan.
8. Pengguna dapat melihat jadwal kegiatan dibeberapa hari kedepan.
9. Pengguna dapat membagikan tugas kepada anggota tim.
10. Pengguna dapat melihat dan melaporkan hasil tugas yang dikerjakan.

### 3.5 Perancangan (*Design*)

Pada tahap ini setelah kebutuhan didapatkan, selanjutnya dilakukan tahap perancangan dengan melakukan pemodelan sistem menggunakan *Unified Modelling Language* (UML). Diagram yang digunakan diantaranya adalah *use case diagram* dan *activity diagram*. Setelah dilakukan pemodelan menggunakan UML, selanjutnya dilakukan perancangan antarmuka.

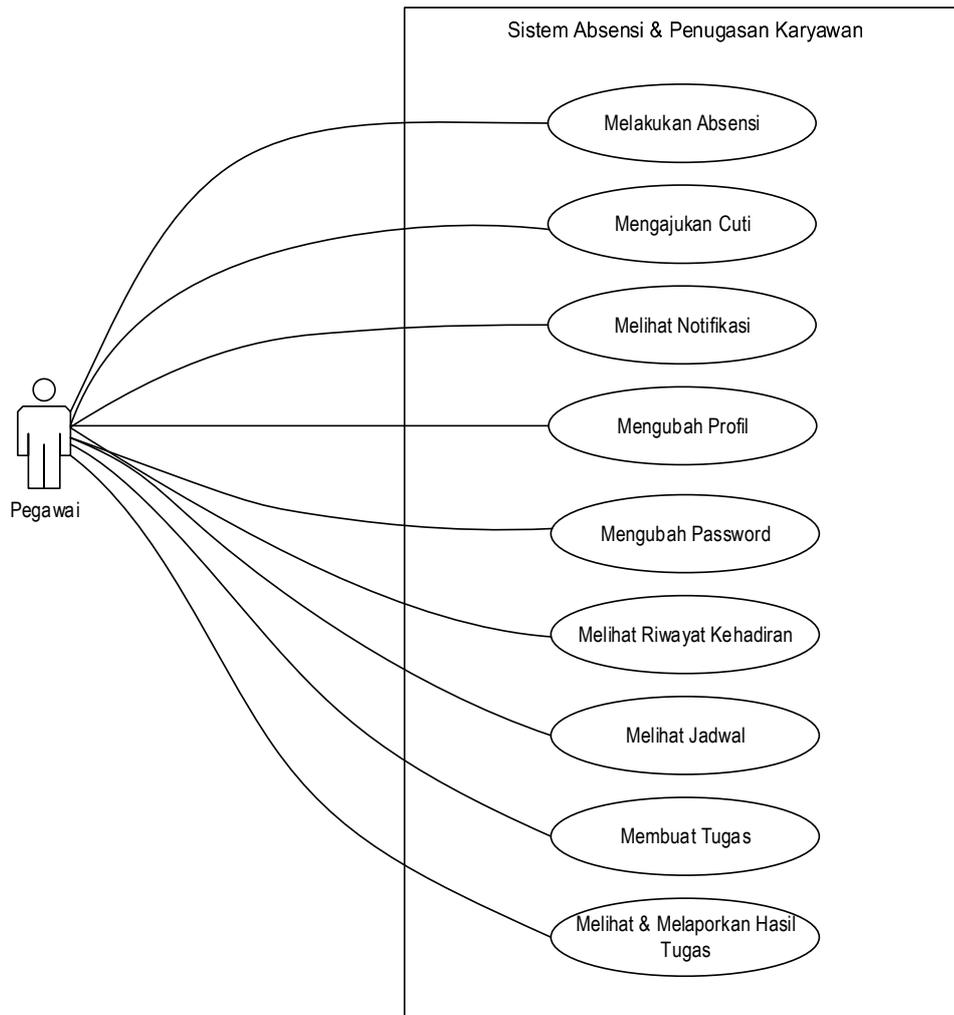
#### 3.5.1 Use Case Diagram

*Use case diagram* digunakan untuk menggambarkan interaksi antara pengguna atau aktor yang terlibat dengan sistem aplikasi, serta menggambarkan berbagai fitur fungsionalitas yang diperlukan oleh pengguna. *Use case diagram* dapat dilihat pada Gambar 3.5.1.



Gambar 3.5.1 *Use Case Diagram* Sebelum Pegawai Login

*Use case diagram* setelah pengguna *login* memiliki fitur melakukan absensi, mengajukan cuti, melihat notifikasi, mengubah profil & password, melihat riwayat kehadiran, melihat jadwal, membuat & membagikan tugas, melihat & melaporkan tugas, melihat status & riwayat pelaporan tugas, memberikan *feedback* tugas. *Use case diagram* setelah pengguna *login* dapat dilihat pada Gambar 3.5.2.



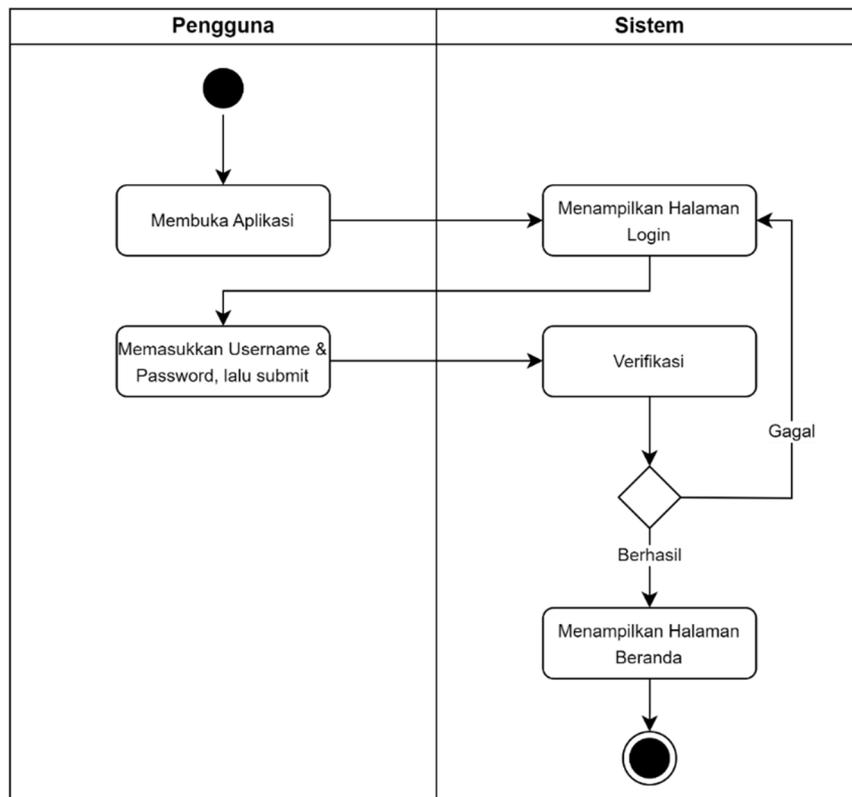
Gambar 3.5.2 *Use Case Diagram* Setelah Pegawai *Login*

### 3.5.2 Activity Diagram

*Activity diagram* digunakan untuk menggambarkan aktivitas atau proses bisnis dari sebuah fitur didalam sistem. Berikut adalah *activity diagram* yang ada pada aplikasi absensi dan penugasan karyawan.

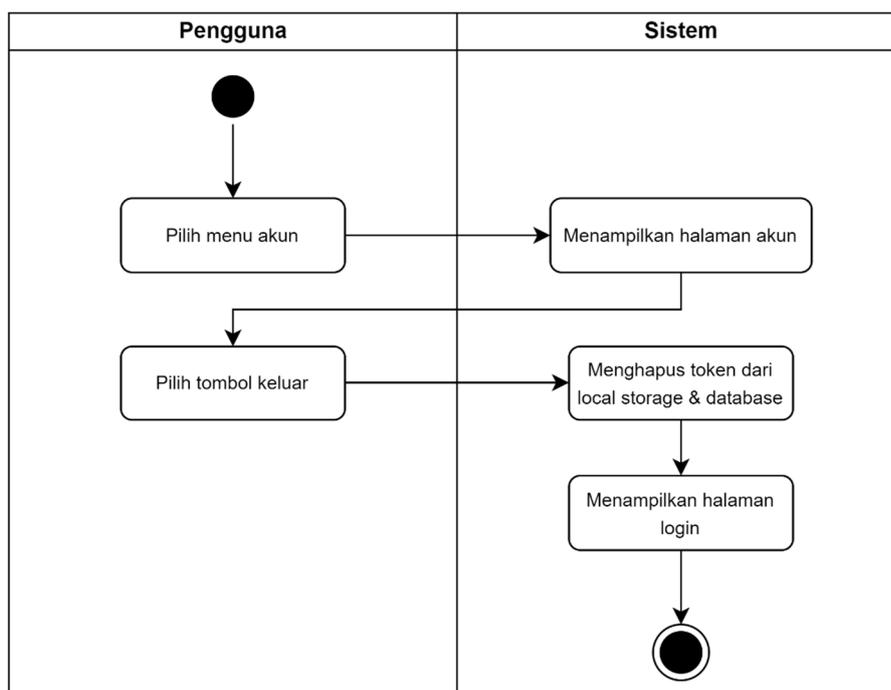
### 1. Activity Diagram Otentikasi

*Activity diagram* Otentikasi dibagi menjadi dua, yaitu *login* dan *logout*. *Activity diagram login* menjelaskan alur proses pengguna untuk masuk ke dalam sistem. Pengguna memulai proses dengan masuk ke aplikasi, kemudian aplikasi akan menampilkan halaman *login*. Pengguna memasukkan *username* dan *password* lalu menekan tombol submit, setelah disubmit sistem akan memverifikasi data yang dikirim. Jika gagal maka, sistem akan tetap berada didalam halaman *login* dan menampilkan pesan kesalahan. Jika berhasil, maka sistem akan masuk ke sistem dan menampilkan halaman utama aplikasi. *Activity diagram login* dapat dilihat pada Gambar 3.5.3.



Gambar 3.5.3 Activity Diagram Login

*Activity diagram logout* menjelaskan alur proses pengguna untuk keluar dari aplikasi. Pengguna memulai dengan memilih menu akun, sistem akan menampilkan halaman akun. Pengguna memilih tombol keluar, lalu sistem akan menghapus *login token* dari *local storage* dan database sistem, lalu sistem akan menampilkan halaman *login*. *Activity diagram logout* dapat dilihat pada Gambar 3.5.4.



Gambar 3.5.4 *Activity Diagram Logout*

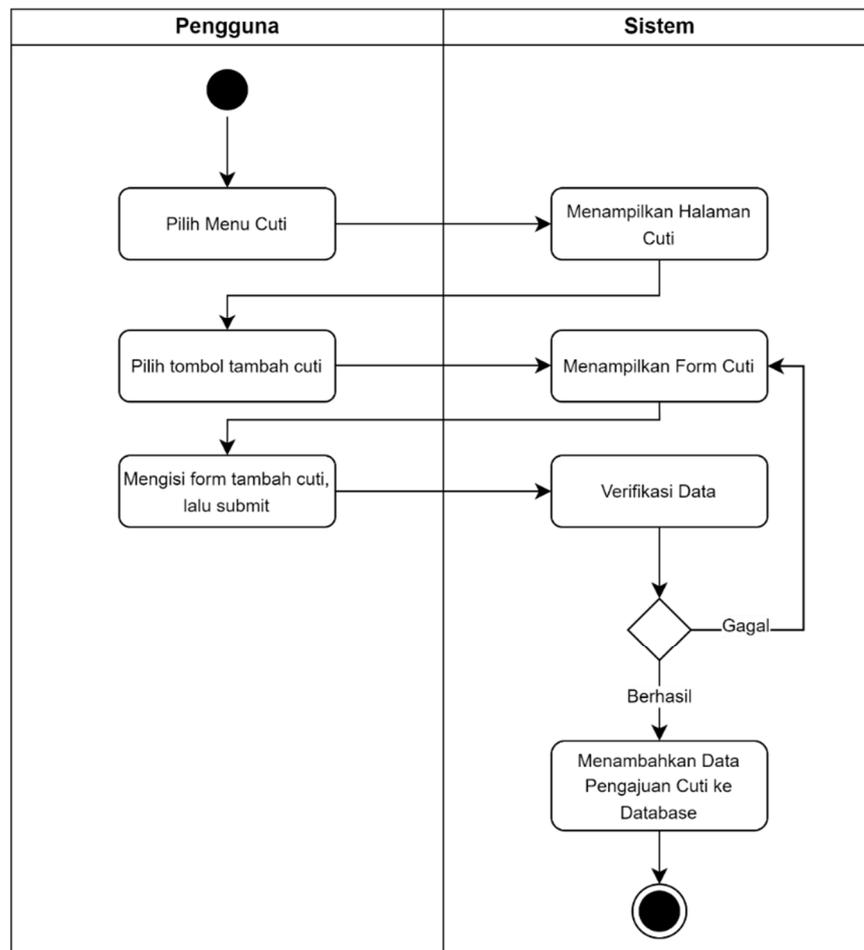
## 2. *Activity Diagram* Melakukan Absensi

*Activity diagram* melakukan absensi menjelaskan alur proses pengguna dalam melakukan pencatatan kehadiran. Pengguna memulai dengan memilih menu absensi, sistem akan menampilkan peta yang berisi lokasi kantor dan lokasi pengguna saat ini. Pengguna memilih tombol masuk jika akan melakukan absensi masuk, dan memilih tombol keluar jika akan melakukan absensi keluar. Sistem akan melakukan verifikasi lokasi pengguna, jika berada di luar area kantor, maka sistem akan melakukan verifikasi izin akun pengguna untuk melakukan absensi diluar kantor, jika tidak diizinkan absensi diluar kantor, maka sistem akan menampilkan pesan tidak dapat melakukan absensi, jika diizinkan absensi diluar kantor, maka pengguna harus melampirkan catatan atau alasan dan dapat melanjutkan proses absensi. Jika pengguna berada di dalam area kantor, sistem akan langsung menampilkan halaman kamera. Selanjutnya pengguna mengambil gambar wajah dan sistem melakukan verifikasi wajah pengguna, jika wajah berbeda, maka sistem akan menampilkan pesan kesalahan wajah berbeda dan pengguna diwajibkan mengambil gambar kembali, jika wajah sama, maka sistem akan menyimpan data absensi kedalam database sistem. *Activity diagram* melakukan absensi dapat dilihat pada Gambar 3.5.5.



### 3. *Activity Diagram* Mengajukan Cuti

*Activity diagram* mengajukan cuti menjelaskan alur proses pengguna dalam melakukan pengajuan cuti. Pengguna memulai dengan memilih menu cuti, sistem akan menampilkan halaman cuti, pengguna memilih tombol tambah pada halaman cuti, lalu sistem akan menampilkan *form* pengajuan cuti, selanjutnya pengguna melakukan pengisian *form* cuti dan menekan tombol *submit*, selanjutnya sistem akan memverifikasi data *form*, jika verifikasi gagal, maka sistem tetap akan menampilkan form, jika verifikasi berhasil, maka sistem menambahkan data pengajuan ke dalam database. *Activity diagram* mengajukan cuti dapat dilihat pada Gambar 3.5.6.

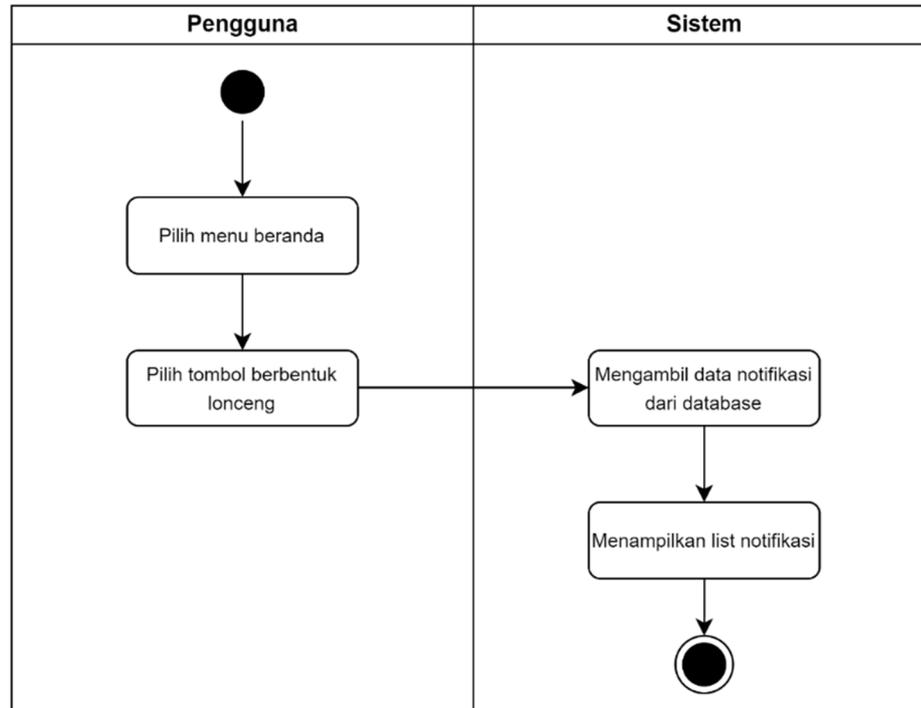


Gambar 3.5.6 *Activity Diagram* Mengajukan Cuti

### 4. *Activity Diagram* Melihat Notifikasi

*Activity diagram* melihat notifikasi menjelaskan alur proses pengguna dalam melihat riwayat notifikasi. Pengguna memulai dengan memilih menu beranda, lalu

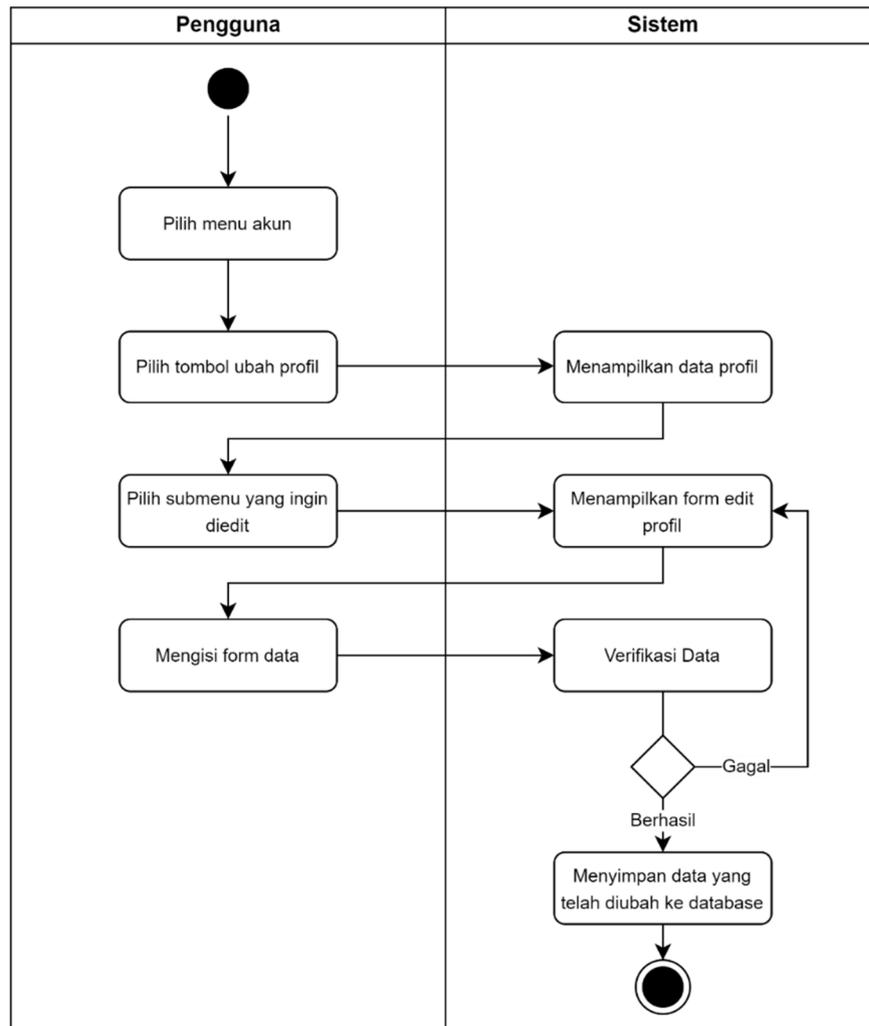
memilih tombol lonceng yang ada pada halaman tersebut. Sistem akan mengambil data notifikasi dari database, dan menampilkan notifikasi dalam bentuk list. *Activity diagram* melihat notifikasi dapat dilihat pada Gambar 3.5.7.



Gambar 3.5.7 *Activity Diagram* Melihat Notifikasi

##### 5. *Activity Diagram* Mengubah Profil

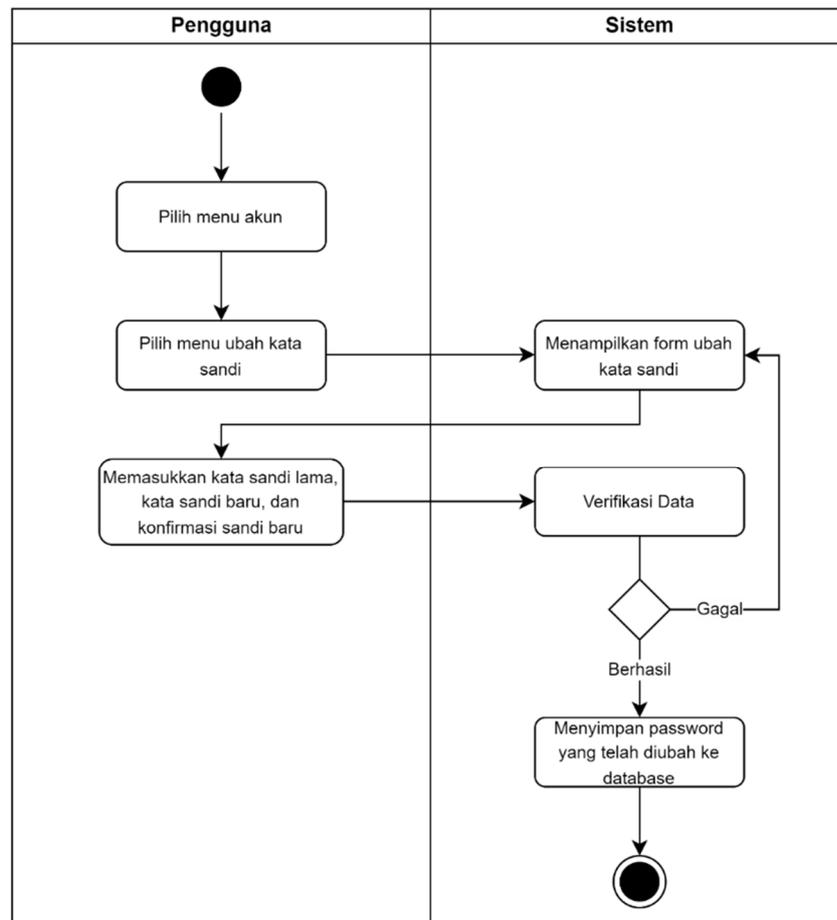
*Activity diagram* mengubah profil menjelaskan alur proses pengguna dalam melakukan perubahan profil. Pengguna memulai dengan memilih menu akun, lalu pada halaman akun pengguna memilih tombol ubah profil, maka sistem akan menampilkan halaman profil, lalu pengguna memilih submenu profil yang ingin diubah, lalu sistem akan menampilkan form sesuai dengan submenu yang dipilih. Pengguna melakukan pengisian form data dan submit data, lalu sistem akan memverifikasi data yang disubmit sebelumnya, jika verifikasi gagal, maka sistem akan tetap menampilkan halaman pengisian form dan menampilkan pesan kesalahan pengisian form, jika verifikasi berhasil maka sistem akan menyimpan data yang telah diubah ke dalam database. *Activity diagram* mengubah profil dapat dilihat pada Gambar 3.5.8.



Gambar 3.5.8 *Activity Diagram* Mengubah Profil

#### 6. *Activity Diagram* Mengubah Password

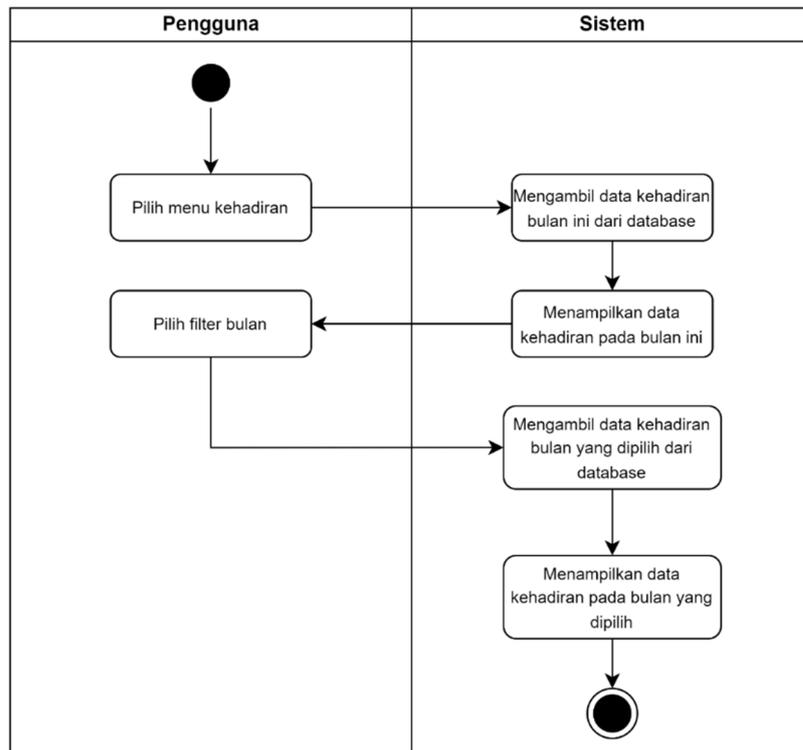
*Activity diagram* mengubah password menjelaskan alur proses pengguna dalam melakukan perubahan password. Pengguna memulai dengan memilih menu akun, lalu pada halaman akun pengguna memilih menu ubah kata sandi, lalu sistem akan menampilkan form ubah kata sandi, lalu pengguna mengisi kata sandi lama, kata sandi baru, dan konfirmasi kata sandi baru. Sistem melakukan verifikasi terhadap data yang dikirim, jika verifikasi gagal, maka sistem akan tetap menampilkan halaman pengisian form dan menampilkan pesan kesalahan pengisian form, jika verifikasi berhasil maka sistem akan menyimpan password yang baru ke dalam database. *Activity diagram* mengubah *password* dapat dilihat pada Gambar 3.5.9.



Gambar 3.5.9 Activity Diagram Mengubah Password

#### 7. Activity Diagram Melihat Riwayat Kehadiran

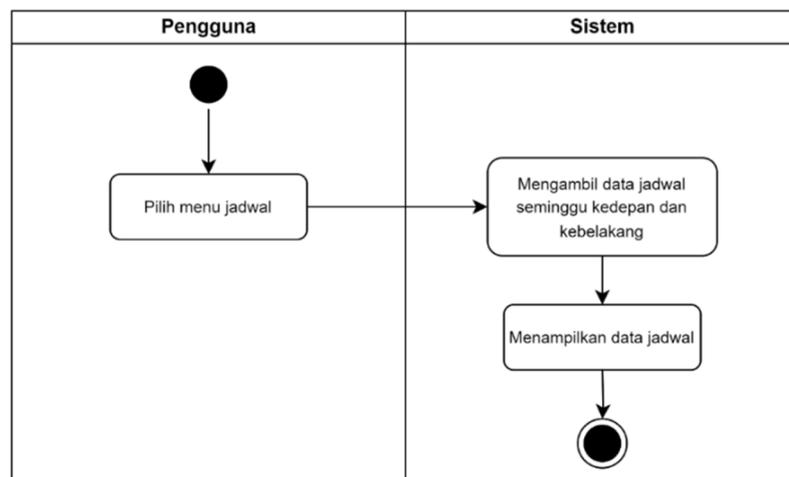
*Activity diagram* melihat riwayat kehadiran menjelaskan alur proses pengguna untuk melihat riwayat kehadiran. Pengguna memulai dengan memilih menu kehadiran, lalu sistem akan melakukan pengambilan data kehadiran pengguna pada bulan ini dari database dan menampilkan list data riwayat kehadiran pada bulan tersebut. Jika pengguna ingin melihat riwayat pada bulan sebelumnya, maka pengguna memilih filter bulan yang akan dilihat, setelah memilih filter bulan, sistem akan mengambil data kehadiran pada bulan yang dipilih dari database dan menampilkan data list data riwayat kehadiran pada bulan yang dipilih. *Activity diagram* melihat riwayat kehadiran dapat dilihat pada Gambar 3.5.10.



Gambar 3.5.10 *Activity Diagram* Melihat Riwayat Kehadiran

#### 8. *Activity Diagram* Melihat Jadwal

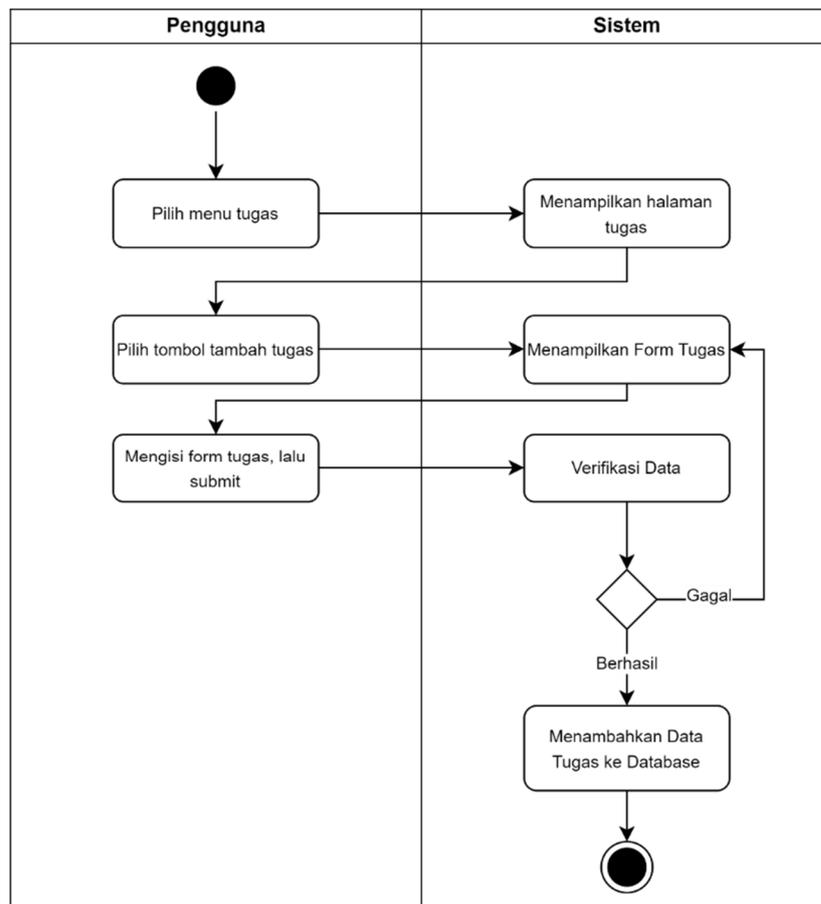
*Activity diagram* melihat jadwal menjelaskan alur proses pengguna dalam untuk melihat jadwal. Pengguna memilih menu jadwal, lalu sistem akan melakukan pengambilan data jadwal seminggu kedepan dan seminggu kebelakang dari hari ini. Selanjutnya sistem akan menampilkan data jadwal. *Activity diagram* melihat jadwal dapat dilihat pada Gambar 3.5.11.



Gambar 3.5.11 *Activity Diagram* Melihat Jadwal

### 9. Activity Diagram Membuat Tugas

*Activity diagram* membuat tugas menjelaskan alur proses pengguna dalam melakukan pembuatan tugas. Pengguna memulai dengan memilih menu tugas, lalu sistem akan menampilkan halaman tugas, lalu pengguna memilih tombol tambah tugas pada halaman tersebut, lalu sistem akan menampilkan halaman form tugas, selanjutnya pengguna mengisi form tugas dan menekan tombol submit setelah form terisi. Sistem melakukan verifikasi data form tugas yang dikirim, jika verifikasi gagal maka sistem akan tetap menampilkan halaman form tugas dan menampilkan pesan kesalahan form tugas, jika verifikasi berhasil maka sistem akan menambahkan data tugas ke dalam database. *Activity diagram* membuat tugas dapat dilihat pada Gambar 3.5.12.

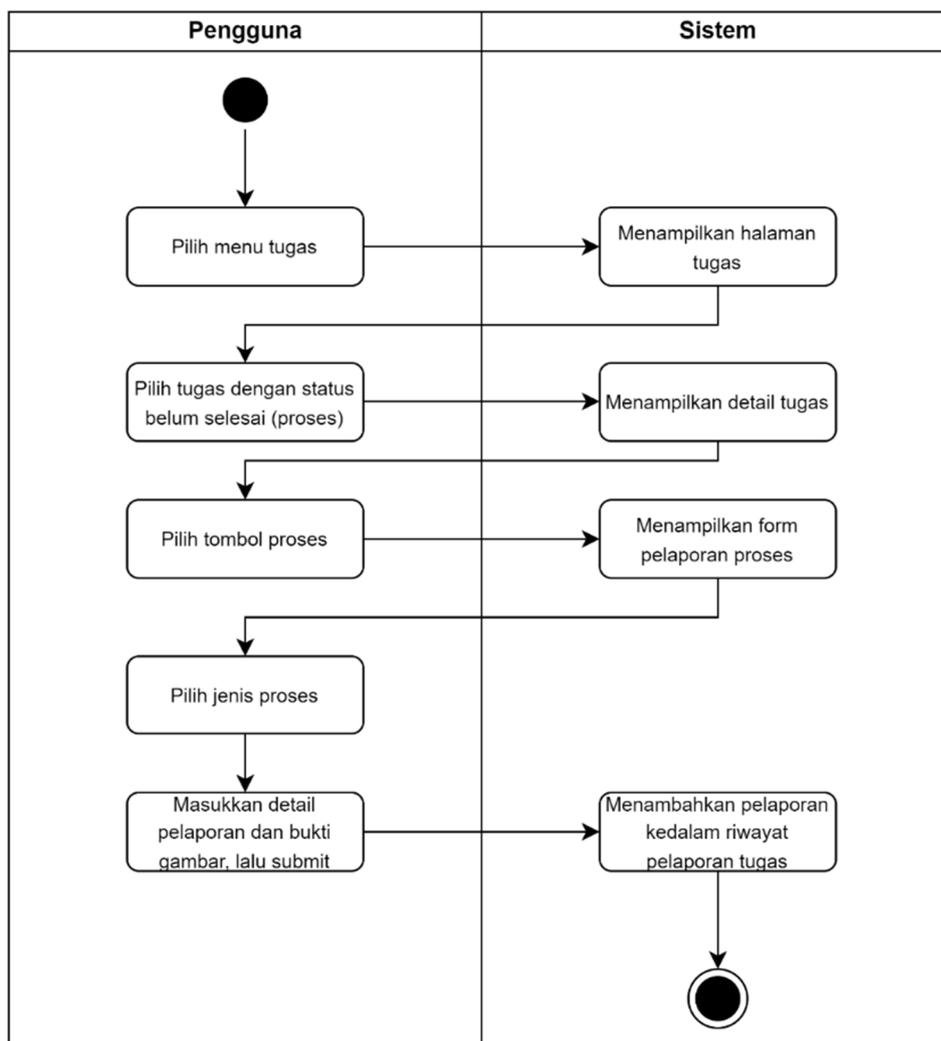


Gambar 3.5.12 Activity Diagram Membuat Tugas

### 10. Activity Diagram Melihat dan Melaporkan Tugas

*Activity diagram* melihat dan melaporkan tugas menjelaskan alur proses pengguna untuk melihat dan melaporkan proses tugas. Pengguna memulai dengan memilih

menu tugas, sistem akan menampilkan halaman tugas. Pengguna memilih sebuah tugas dengan status belum selesai atau masih dalam proses, sistem akan menampilkan detail tugas yang dipilih. Pengguna menekan tombol proses, lalu sistem akan menampilkan form pelaporan tugas. Pengguna memilih jenis proses dan memasukkan detail pelaporan serta bukti pelaporan tugas, lalu pengguna menekan tombol submit, sistem akan menambahkan pelaporan tersebut ke dalam riwayat pelaporan tugas, dan dapat dilihat oleh pengirim dan setiap pengguna yang menerima tugas tersebut. *Activity diagram* melihat dan melaporkan tugas dapat dilihat pada Gambar 3.5.13.



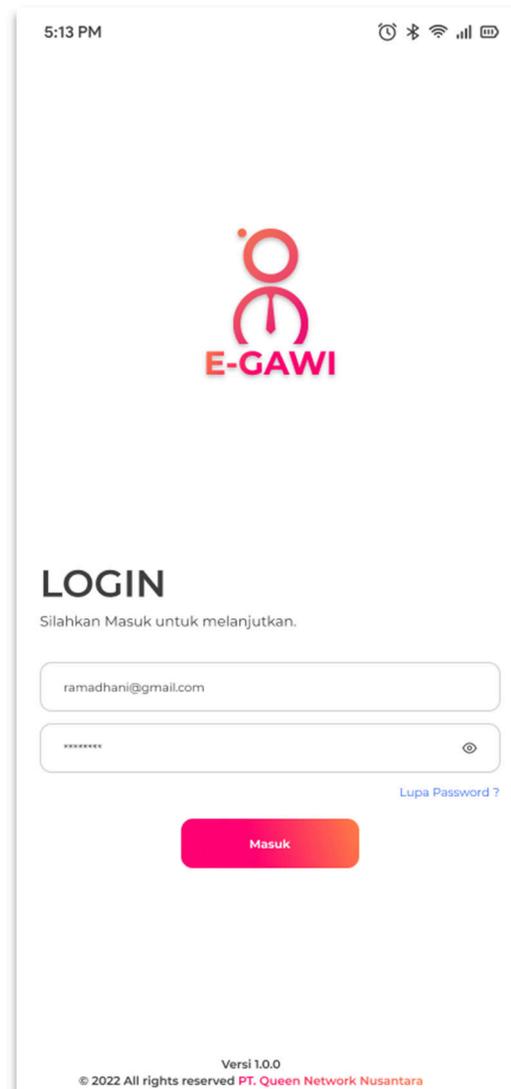
Gambar 3.5.13 *Activity Diagram* Melihat dan Melaporkan Tugas

### 3.5.3 Perancangan Antarmuka

Perancangan antarmuka adalah desain tampilan sistem yang akan dibangun, antarmuka sistem merupakan sebuah perantara antara pengguna dengan sistem yang ada.

#### 1. Halaman *Login*

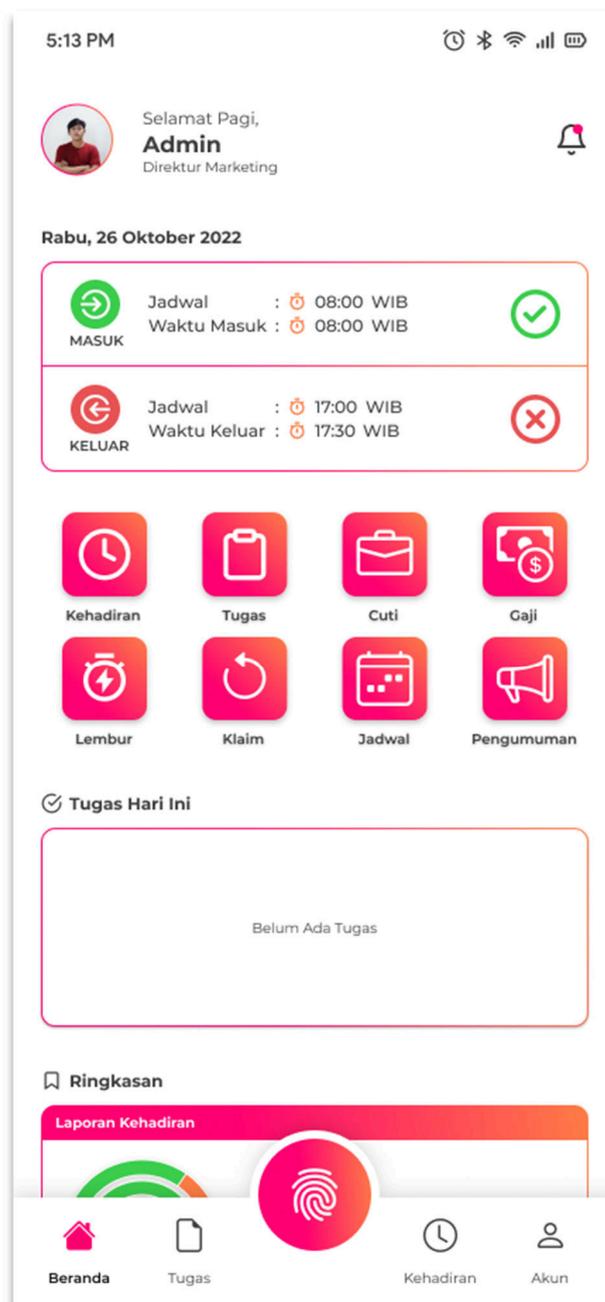
Halaman *login* merupakan pintu masuk ke dalam aplikasi. Pada halaman ini terdapat *field input username* dan *password*, dan sebuah tombol submit form. Pengguna diharuskan mengisi *form* dan melakukan *submit* untuk masuk ke dalam aplikasi. Halaman *login* ditunjukkan pada Gambar 3.5.14.



Gambar 3.5.14 Halaman *Login*

## 2. Halaman Beranda

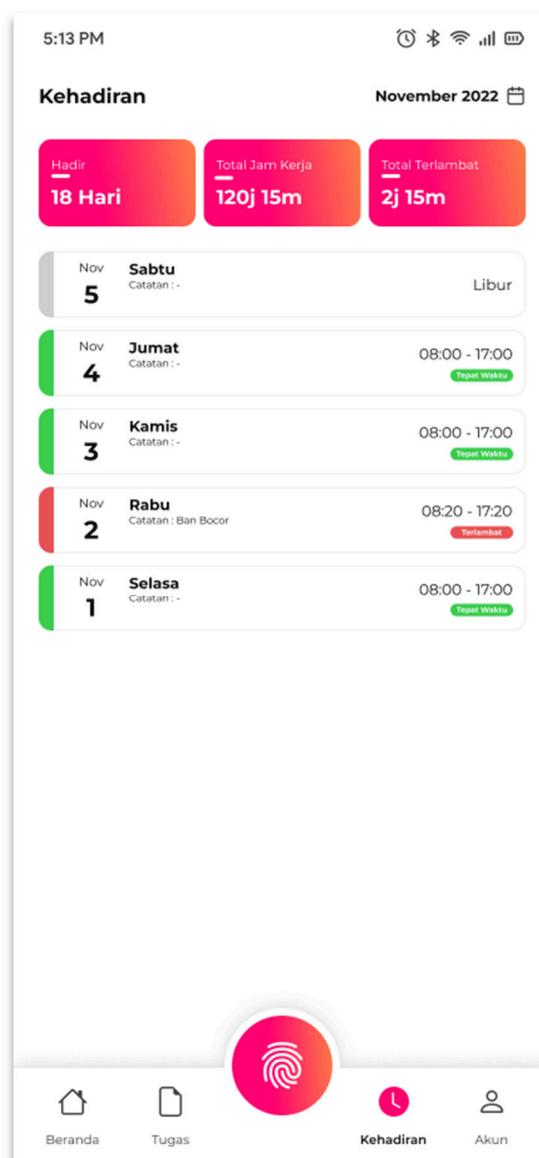
Halaman Beranda merupakan halaman utama dalam aplikasi. Pada halaman ini terdapat berbagai menu dan fitur-fitur didalamnya, seperti menu kehadiran, tugas, cuti, dan lain sebagainya. Pada halaman ini juga terdapat ringkasan kehadiran dan tugas yang dapat dikerjakan pada hari ini, selain itu juga terdapat tombol riwayat notifikasi. Halaman beranda ditunjukkan pada Gambar 3.5.15.



Gambar 3.5.15 Halaman Beranda

### 3. Halaman Kehadiran

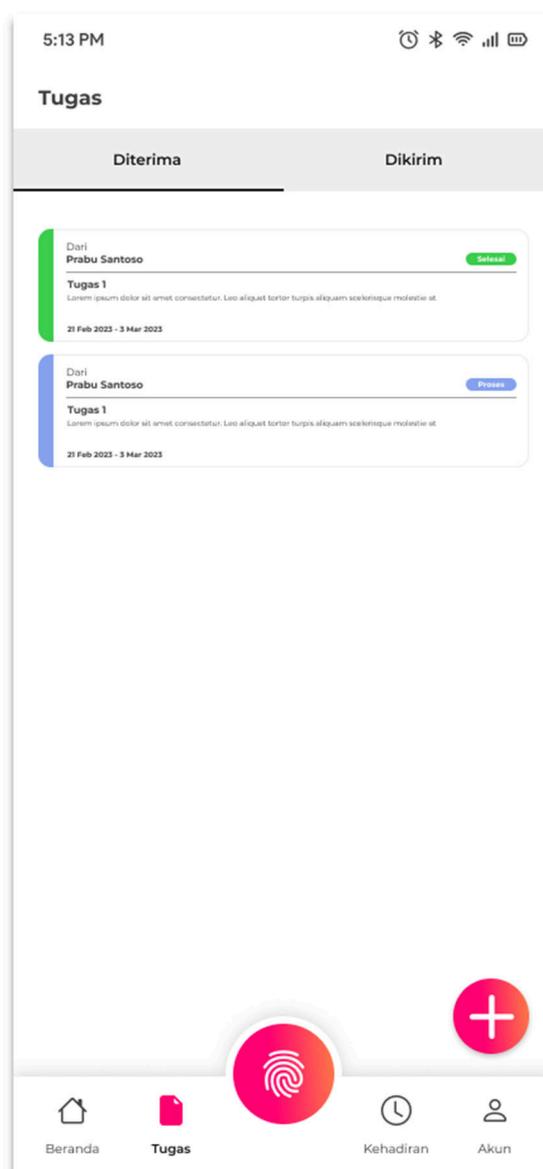
Halaman kehadiran merupakan halaman untuk melihat rangkuman dan riwayat kehadiran pada tiap bulan. Pada halaman ini menampilkan jumlah kehadiran, total jam kerja, dan total keterlambatan pada tiap bulannya, kehadiran tiap hari dicatat dengan list dari berbagai *card* ditandai dengan warna yang berbeda, yakni warna hijau menandakan hadir tepat waktu, warna merah menandakan hadir terlambat, dan warna abu-abu menandakan hari libur, tidak hadir atau belum melakukan absensi. Pada halaman ini terdapat *filter* bulan untuk melihat riwayat kehadiran pada bulan tertentu. Halaman kehadiran ditunjukkan pada Gambar 3.5.16.



Gambar 3.5.16 Halaman Kehadiran

#### 4. Halaman Tugas

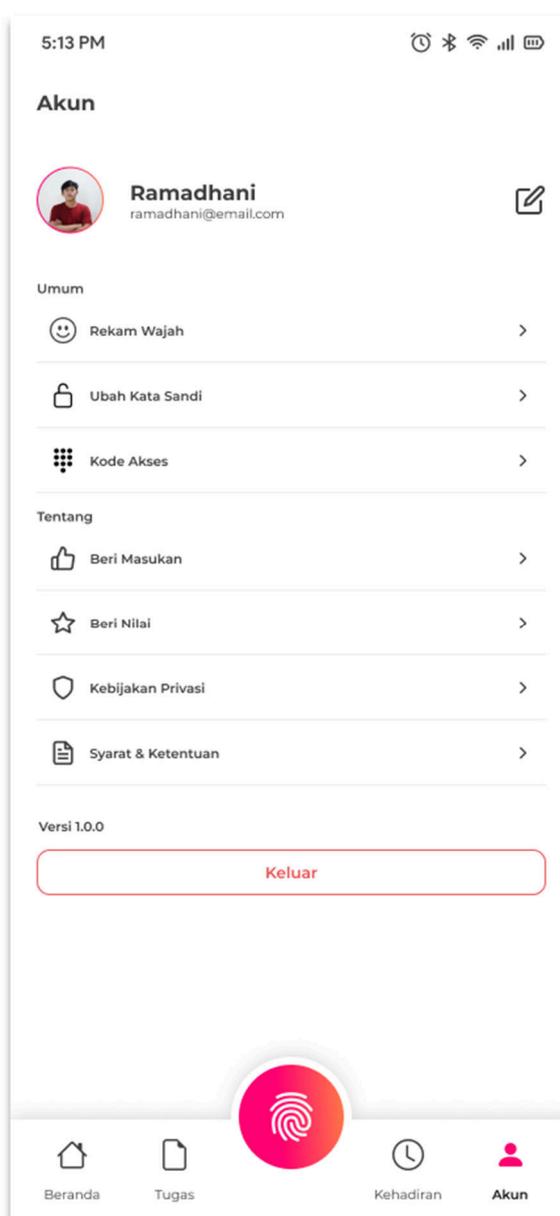
Halaman tugas merupakan halaman untuk melihat tugas-tugas yang diberikan kepada pengguna dan dikirim oleh pengguna. Pada halaman ini menampilkan dua submenu untuk menampilkan list tugas yang diterima dan dikirim. List tugas berisi *card* dengan detail dan warna yang berbeda, warna hijau menandakan bahwa status tugas sudah selesai dan warna biru menandakan bahwa tugas masih dalam proses pengerjaan. Pada halaman ini terdapat tombol tambah yang digunakan untuk berpindah ke halaman *form* tugas. Halaman tugas ditunjukkan pada Gambar 3.5.17.



Gambar 3.5.17 Halaman Tugas

## 5. Halaman Akun

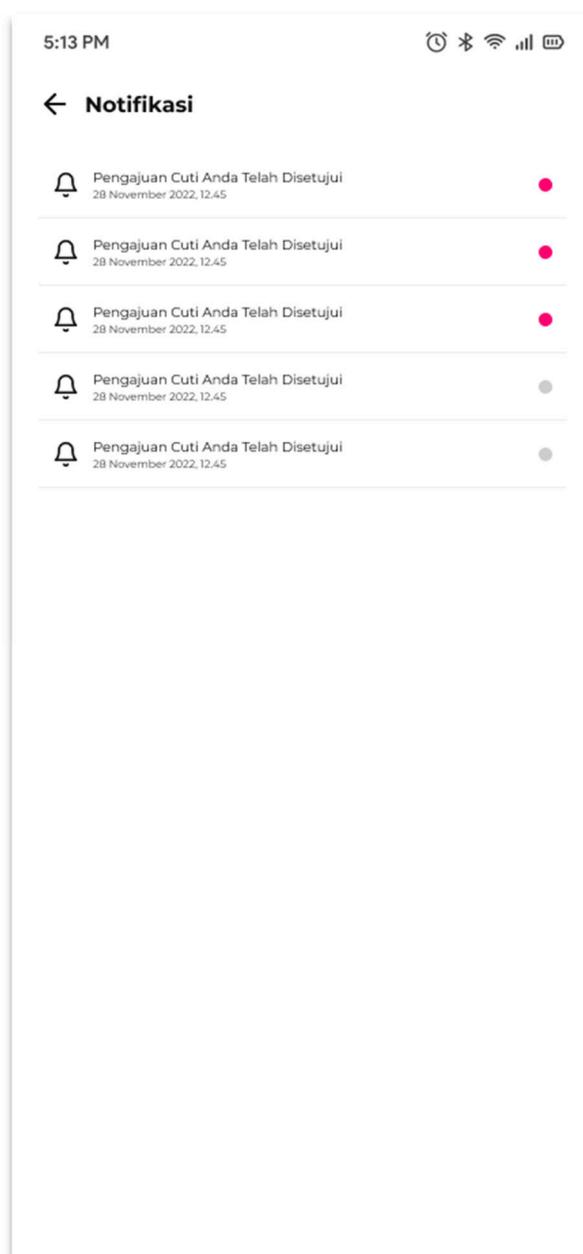
Halaman akun merupakan halaman untuk melihat pengaturan yang berhubungan dengan akun dan aplikasi. Pada halaman ini menampilkan menu yang dibagi menjadi 2 bagian, yaitu menu umum yang berisi, rekam wajah, ubah kata sandi, dan kode akses, selanjutnya menu tentang berisi menu beri masukan, beri nilai, kebijakan privasi, dan syarat & ketentuan. Pada terdapat tombol sunting yang digunakan untuk berpindah ke halaman ubah profil dan tombol keluar untuk *logout* dari akun. Halaman akun ditunjukkan pada Gambar 3.5.18.



Gambar 3.5.18 Halaman Akun

## 6. Halaman Notifikasi

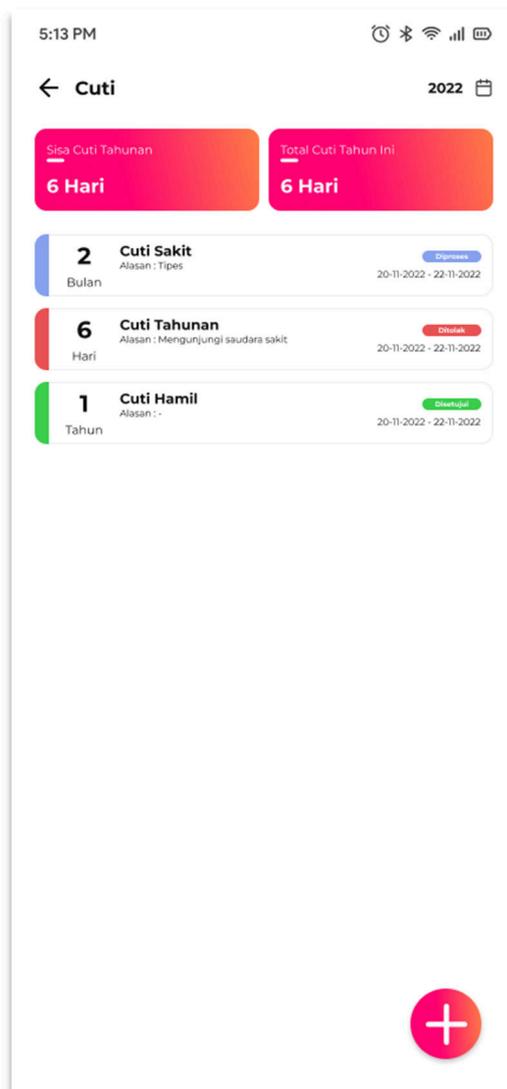
Halaman notifikasi merupakan halaman untuk melihat *list* riwayat notifikasi. Pada halaman ini menampilkan *list* notifikasi yang berisi penjelasan, waktu dan status notifikasi. Status notifikasi ditandai dengan lingkaran yang memiliki dua jenis warna yang berbeda, warna merah muda menandakan bahwa notifikasi belum dibaca, sedangkan warna abu-abu menandakan bahwa notifikasi sudah dibaca. Halaman notifikasi ditunjukkan pada Gambar 3.5.19.



Gambar 3.5.19 Halaman Notifikasi

## 7. Halaman Cuti

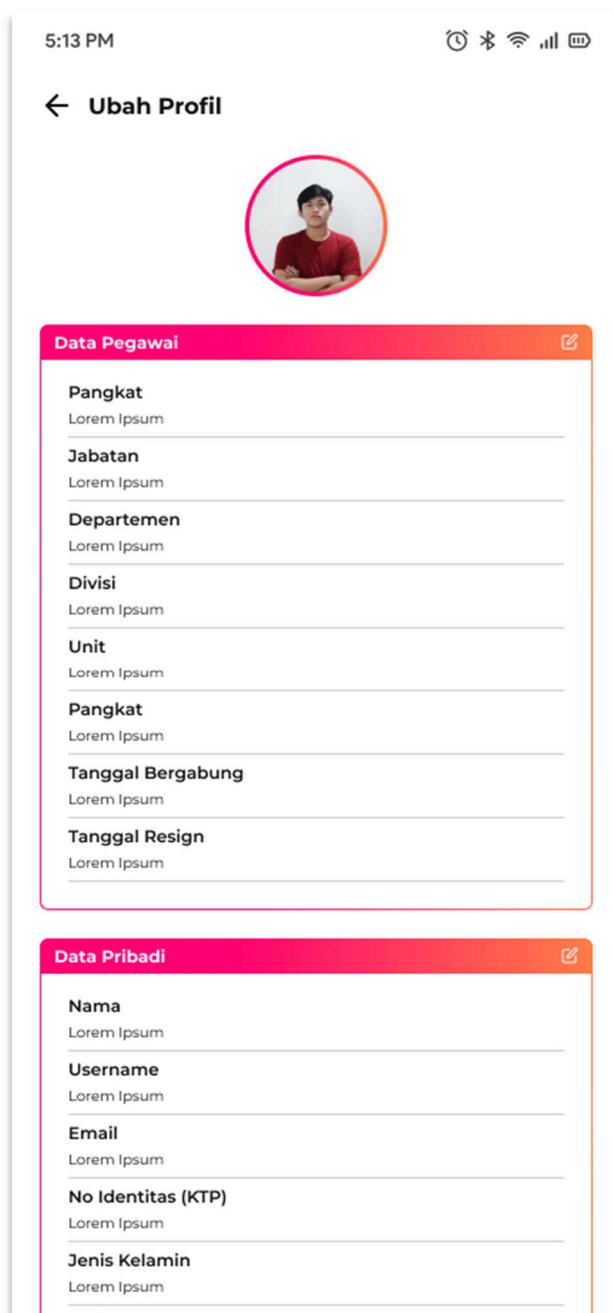
Halaman cuti merupakan halaman untuk mengajukan cuti. Pada halaman ini menampilkan sisa cuti tahunan dan total cuti tahunan yang diajukan. Setiap cuti ditampilkan dalam bentuk *card* yang berisi jenis cuti, alasan cuti, durasi cuti, dan status pengajuan cuti. Status pengajuan cuti memiliki tiga status yang berbeda ditandai dengan warna yang berbeda, warna biru menandakan bahwa cuti sedang diproses dan menunggu validasi cuti oleh HRD, warna merah menandakan bahwa pengajuan cuti ditolak, dan warna hijau menandakan bahwa pengajuan cuti diterima. Pada halaman cuti terdapat *filter* tahun untuk melihat pengajuan cuti tiap tahunnya dan terdapat tombol tambah untuk berpindah ke halaman *form* pengajuan cuti. Halaman cuti ditunjukkan pada Gambar 3.5.20.



Gambar 3.5.20 Halaman Cuti

## 8. Halaman Ubah Profil

Halaman ubah profil merupakan halaman untuk mengubah profil pengguna. Pada halaman ini menampilkan foto profil dan sub profil, untuk mengubah melihat dan mengubah foto profil dilakukan dengan dengan menekan foto, sedangkan untuk mengubah sub profil dilakukan dengan menekan tombol sunting yang ada pada tiap sub menu profilnya. Halaman ubah profil ditunjukkan pada Gambar 3.5.21.



5:13 PM

← Ubah Profil



**Data Pegawai**

**Pangkat**  
Lorem Ipsum

**Jabatan**  
Lorem Ipsum

**Departemen**  
Lorem Ipsum

**Divisi**  
Lorem Ipsum

**Unit**  
Lorem Ipsum

**Pangkat**  
Lorem Ipsum

**Tanggal Bergabung**  
Lorem Ipsum

**Tanggal Resign**  
Lorem Ipsum

**Data Pribadi**

**Nama**  
Lorem Ipsum

**Username**  
Lorem Ipsum

**Email**  
Lorem Ipsum

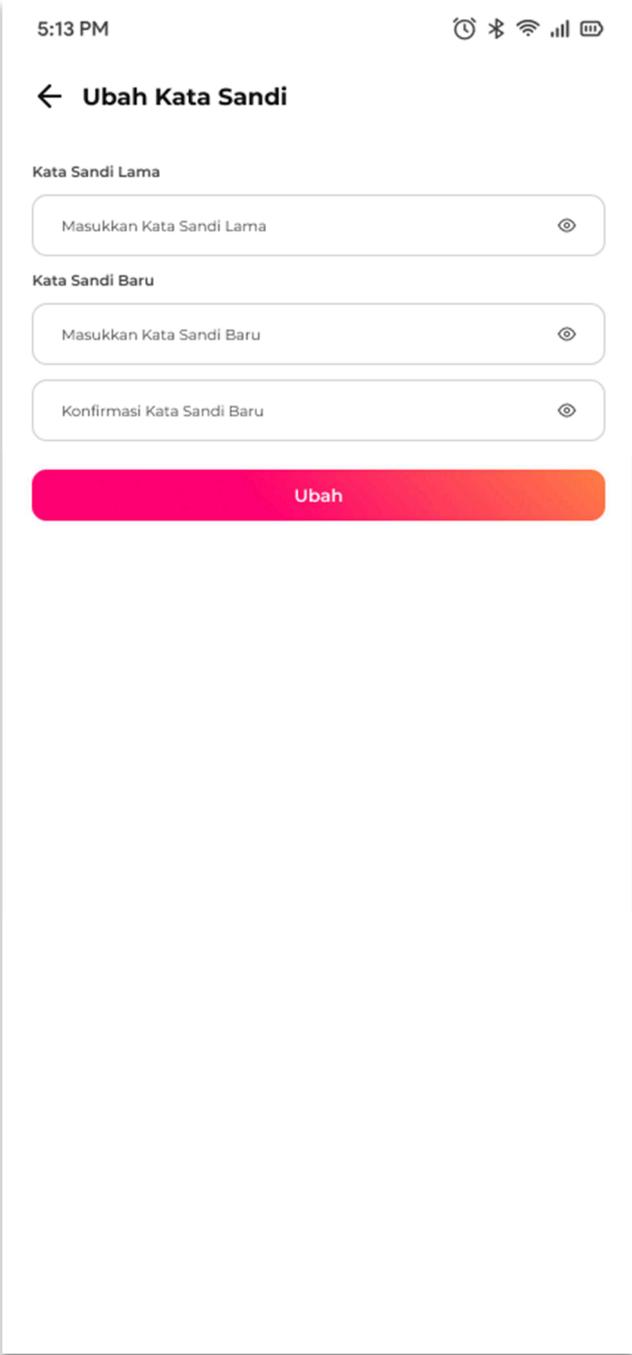
**No Identitas (KTP)**  
Lorem Ipsum

**Jenis Kelamin**  
Lorem Ipsum

Gambar 3.5.21 Halaman Ubah Profil

## 9. Halaman Ubah Sandi

Halaman ubah sandi merupakan halaman yang digunakan untuk mengubah sandi pengguna. Pada halaman ini menampilkan *field input* kata sandi lama, kata sandi baru, konfirmasi kata sandi baru, dan tombol ubah. Halaman ubah sandi ditunjukkan pada Gambar 3.5.22.

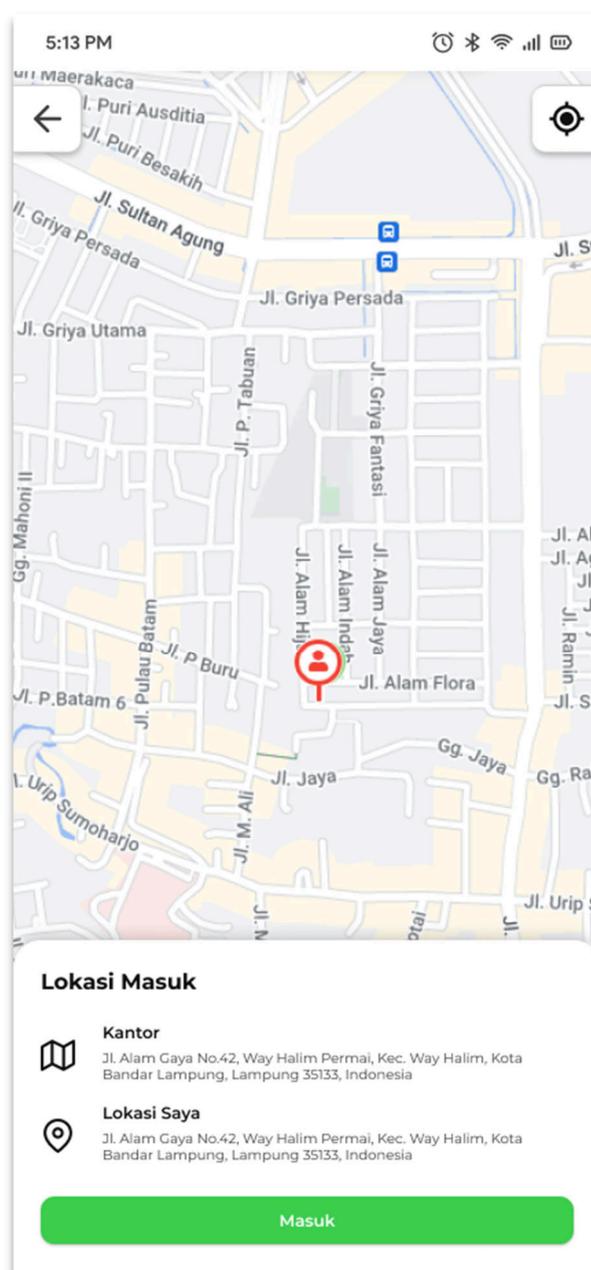


The screenshot displays a mobile application interface for changing a password. At the top, the status bar shows the time as 5:13 PM and various system icons. Below the status bar, there is a back arrow and the title 'Ubah Kata Sandi'. The main content area contains three input fields, each with a label and a placeholder text: 'Kata Sandi Lama' with 'Masukkan Kata Sandi Lama', 'Kata Sandi Baru' with 'Masukkan Kata Sandi Baru', and 'Konfirmasi Kata Sandi Baru' with 'Konfirmasi Kata Sandi Baru'. Each input field has a small eye icon on the right side to toggle visibility. At the bottom of the form, there is a prominent red button with the text 'Ubah'.

Gambar 3.5.22 Halaman Ubah Sandi

## 10. Halaman Absensi

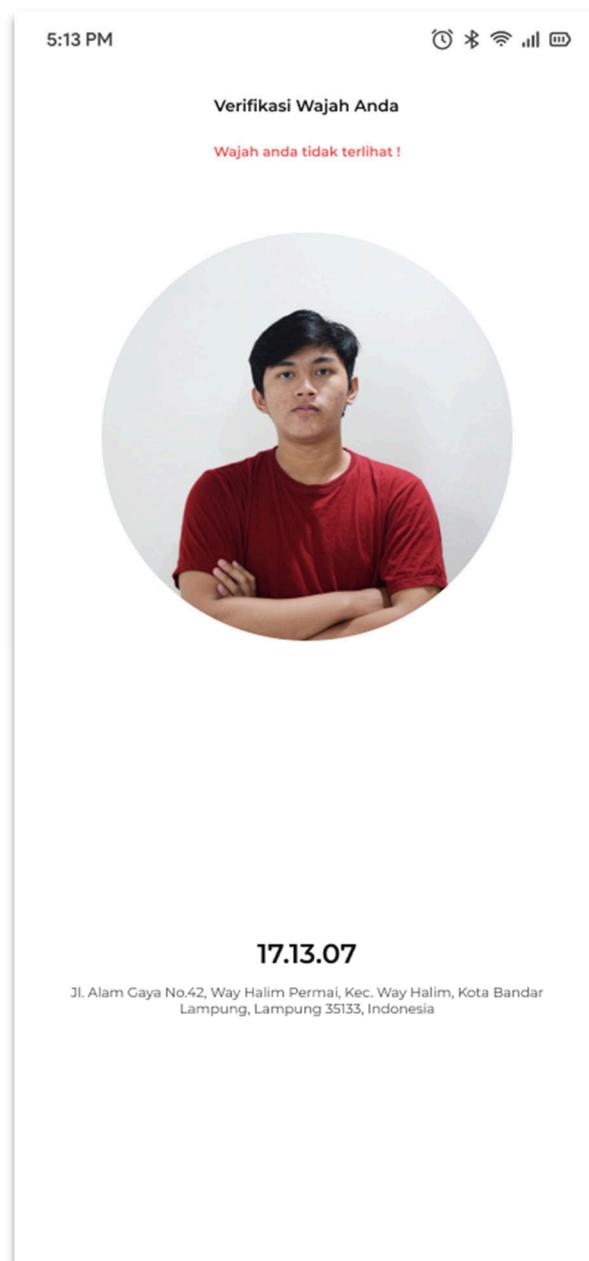
Halaman absensi merupakan halaman yang digunakan untuk melakukan pencatatan kehadiran atau absensi. Pada halaman ini menampilkan *maps* lokasi kantor dan lokasi pengguna, alamat kantor dan pengguna saat ini, dan tombol masuk atau keluar untuk submit lokasi dan melanjutkan ke tahap absensi selanjutnya. Pada halaman ini terdapat tombol kembali dan tombol lokasi untuk *refresh* lokasi pengguna. Halaman absensi ditunjukkan pada Gambar 3.5.23.



Gambar 3.5.23 Halaman Absensi

## 11. Halaman Kamera

Halaman kamera merupakan halaman yang digunakan untuk mengambil foto wajah untuk verifikasi absensi. Pada halaman ini menampilkan kamera, alamat, waktu saat ini, dan menampilkan pesan kondisi kamera. Untuk mengambil foto dilakukan dengan menampilkan wajah yang jelas dengan posisi tersenyum. Halaman kamera ditunjukkan pada Gambar 3.5.24.



Gambar 3.5.24 Halaman Kamera

## **BAB V SIMPULAN DAN SARAN**

### **5.1 Simpulan**

Berdasarkan hasil dari penelitian yang telah disampaikan, didapatkan kesimpulan sebagai berikut:

1. Penelitian yang dilakukan berhasil merancang dan membangun aplikasi *Android* absensi dan penugasan karyawan menggunakan Framework *React Native* di PT. Queen Network Nusantara. Aplikasi tersebut telah diuji menggunakan 33 skenario pengujian, dan hasilnya sesuai dengan yang diharapkan.
2. Aplikasi absensi dan penugasan karyawan telah berhasil menyediakan 9 fitur dan digunakan pada PT. Queen Network Nusantara dengan minimal spesifikasi *android* versi 6 (*Marshmallow*).
3. Pengujian yang dilakukan pada fitur absensi bagian pengenalan wajah telah berhasil mengenali wajah pengguna, pada pengujian tersebut ditemukan kekurangan yaitu *library Expo Camera* yang dipakai tidak dapat membedakan antara wajah asli dengan wajah pada hasil foto.

### **5.2 Saran**

Adapun saran untuk pengembangan selanjutnya adalah sebagai berikut:

1. Melakukan pengembangan aplikasi lebih lanjut dengan menambahkan fitur penghitungan gaji, pengajuan lembur, pengumuman, dan lupa password.
2. Melakukan pengembangan pengenalan wajah dengan mengambil foto wajah tampak depan, depan kanan, dan depan kiri untuk meminimalisir terjadinya kecurangan dalam absensi.
3. Melakukan pengembangan untuk versi iOS sehingga karyawan yang tidak memiliki *android* dapat menggunakan aplikasi absensi.

## DAFTAR PUSTAKA

- [1] Kholisa, N., Hubungan Manajemen Waktu Dengan Efektivitas Kerja Karyawan, *Journal of Social and Industrial Psychology*, vol. 1, no. 1, hal. 56–60, 2012.
- [2] Indonesia, R., *Undang-Undang Republik Indonesia Nomor 13 Tahun 2003 Tentang Ketenagakerjaan*. Indonesia, 2003.
- [3] We Are Social dan Meltwater, Digital 2023: Global Overview Report, 2023. [Daring]. Tersedia di: <https://datareportal.com/reports/digital-2023-global-overview-report>. [Diakses: 31-Mei-2023].
- [4] Handoko, H., *Manajemen Personalia dan Sumberdaya Manusia*. Yogyakarta: BPFE, 2016.
- [5] Edy Sutrisno, *Manajemen Sumber Daya Manusia*. Jakarta: Kencana, 2016.
- [6] PT. Queen Network Nusantara, Company Profile. .
- [7] Anggraeni, E.Y. dan Irviani, R., *Pengantar Sistem Informasi*. Risanto, E. (editor), Yogyakarta: Andi Offset, 2017.
- [8] Sutanta, E., *Basis Data Dalam Tinjauan Konseptual*. Yogyakarta: Andi Offset, 2011.
- [9] Setiawan, E.B. dan Kurniawan, B., Perancangan Sistem Absensi Kehadiran Perkuliahan dengan Menggunakan Radio Frequency Identification (RFId), *Jurnal CoreIT*, vol. 1, no. 2, hal. 44–49, 2015.
- [10] Enterprise, J., *Mengenal Dasar-Dasar Pemrograman Android*. Jakarta: Elex Media Komputindo, 2015.
- [11] EMS, T., *Pemrograman Android dalam Sehari*. Jakarta: Elex Media Komputindo, 2015.
- [12] Eugene, B., Android API Levels, 2022. [Daring]. Tersedia di: <https://apilevels.com/>. [Diakses: 16-Apr-2023].

- [13] Maiyana, E., Pemanfaatan Android Dalam Perancangan Aplikasi Kumpulan Doa, *Jurnal Sains dan Informatika*, vol. 4, no. 1, hal. 54–67, 2018.
- [14] Meta Platforms, React Native Learn once, write anywhere. [Daring]. Tersedia di: <https://reactnative.dev/>. [Diakses: 16-Apr-2023].
- [15] Eisenman, B., *Learning React Native: Building Native Mobile Apps with JavaScript*. O'Reilly Media, Inc., 2016.
- [16] Sianipar, R.H., *Pemrograman Javascript: Teori Dan Implementasi*. Penerbit Informatika, 2015.
- [17] Budiman, E., Pemanfaatan Teknologi Location Based Service Dalam Pengembangan Aplikasi Profil Kampus Universitas Mulawarman Berbasis Mobile, *Jurnal Ilmiah ILKOM*, vol. 8, no. 3, hal. 137–144, 2016.
- [18] Susanty, W., Astari, I.N. dan Thamrin, T., Aplikasi GIS Menggunakan Metode Location Based Service (LBS) Berbasis Android, *Explore: Jurnal Sistem informasi dan telematika*, vol. 10, no. 1, hal. 53–58, juni 2019.
- [19] Santoso, K.I., Aplikasi Location Based Service Layanan Kesehatan Kota Magelang Berbasis Android, *Infokam*, vol. 12, no. 1, hal. 18–27, 2016.
- [20] Samsir dan Sitorus, J.H.P., Perancangan Sistem Monitoring Lokasi Kendaraan Menggunakan GPS U-Blox Berbasis Android, *Jurnal Bisantera Informatika (JBI)*, vol. 5, no. 1, hal. 1–10, 2021.
- [21] Alam, C.N., Manaf, K., Atmadja, A.R. dan Aurum, D.K., Implementation of haversine formula for counting event visitor in the radius based on Android application, dalam: *2016 4th International Conference on Cyber and IT Service Management*, hal. 1–6, 2016.
- [22] Yulianto, Ramadiani dan Kridalaksana, A.H., Penerapan Formula Haversine Pada Sistem Informasi Geografis Pencarian Jarak Terdekat Lokasi Lapangan Futsal, *Jurnal Ilmiah Ilmu Komputer*, vol. 13, no. 1, hal. 14–21, 2018.
- [23] Suryana, A., Reynaldi, F., Pratama, F., Ginanjar, G., Indriansyah, I. dan Hasman, D., Implementation of haversine formula on the limitation of e-voting radius based on android, dalam: *Proceedings - 2018 4th International Conference on Computing, Engineering, and Design, ICCED 2018*, hal. 218–223, 2019.
- [24] Aldya, A.P., Rahmatulloh, A. dan Fachurroji, M., Haversine Formula Untuk

- Membatasi Jarak Pada Aplikasi Presensi Online, *Jurnal Informatika Sains dan Teknologi*, vol. 4, no. 2, hal. 171–180, 2019.
- [25] Tulach, J., *Practical API design: Confessions of a java framework architect*. Berkeley, CA: Apress, 2008.
- [26] Masykur, F., Implementasi Sistem Informasi Geografis Menggunakan Google Maps Api Dalam Pemetaan Asal Mahasiswa, *Jurnal SIMETRIS*, vol. 5, no. 2, hal. 181–186, 2014.
- [27] Paunsyah, H., Mubarak, H. dan Shofa, R.N., Penentuan Jalur Terpendek menggunakan Google Maps API pada Sistem Informasi Geografis (SIG) Panti Sosial di Kota Tasikmalaya, *Innovation in Research of Informatics (INNOVATICS)*, vol. 1, no. 1, hal. 1–6, 2019.
- [28] Expo, Expo Docs. [Daring]. Tersedia di: <https://docs.expo.dev/versions/latest/sdk/camera/>. [Diakses: 12-Jul-2023].
- [29] Fatoni, A. dan Dwi, D., Rancang Bangun Sistem Extreme Programming Sebagai Metodologi Pengembangan Sistem, *Prosisko*, vol. 3, no. 1, hal. 17–20, 2016.
- [30] Beck, K., Embracing change with extreme programming, *Computer*, vol. 32, no. 10, hal. 70–77, 1999.
- [31] Mulyani, S., *Analisis dan Perancangan Sistem Informasi Manajemen Keuangan Daerah: Notasi Pemodelan Unified Modeling Language (UML)*. Bandung: Abdi Sistematika, 2016.
- [32] Roohullah Jan, S., Tauhid Ullah Shah, S., Ullah Johar, Z., Shah, Y. dan Khan, F., An Innovative Approach to Investigate Various Software Testing Techniques and Strategies, *International Journal of Scientific Research in Science, Engineering and Technology*, vol. 2, no. 2, hal. 682–689, 2016.
- [33] Tri Snadhika Jaya, Pengujian Aplikasi dengan Metode Blackbox Testing Boundary Value Analysis, *Jurnal Informatika Pengembangan IT (JPIT)*, vol. 3, no. 2, hal. 45–48, 2018.
- [34] Xu, Z., Chen, P., Zhang, W., Liu, X. dan Wu, H., Research on mobile phone attendance positioning system based on campus network, dalam: *Proceedings - 2019 International Conference on Smart Grid and Electrical Automation, ICSGEA 2019*, hal. 387–389, 2019.

- [35] Utomo, S.B. dan Hendradjaya, B., Multifactor Authentication on Mobile Secure Attendance System, dalam: *Proceeding - 2018 International Conference on ICT for Smart Society: Innovation Toward Smart Society and Society 5.0, ICISS 2018*, 2018.
- [36] Shahab, A.S. dan Sarno, R., Android application for presence recognition based on face and geofencing, dalam: *Proceedings - 2020 International Seminar on Application for Technology of Information and Communication: IT Challenges for Sustainability, Scalability, and Security in the Age of Digital Disruption, iSemantic 2020*, hal. 208–213, 2020.