

**OTOMATISASI PRESENSI MENGGUNAKAN *GLOBAL POSITIONING SYSTEM (GPS)***

**SKRIPSI**

Oleh :

**KHARIS ROSYADI**

**NIM. 10650112**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2015**

**OTOMATISASI PRESENSI MENGGUNAKAN GLOBAL POSITIONING  
SYSTEM (GPS)**

**SKRIPSI**

**Diajukan kepada :  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk Memenuhi Salah Satu Persyaratan  
Dalam Memperoleh Gelar Sarjana Komputer (S. Kom)**

**Oleh :**

**KHARIS ROSYADI  
NIM. 10650112**

**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2015**

**HALAMAN PERSETUJUAN**  
**OTOMATISASI PRESENSI MENGGUNAKAN GLOBAL POSITIONING**  
**SYSTEM (GPS)**

**SKRIPSI**

**Oleh :**

Nama : Kharis Rosyadi  
NIM : 10650112  
Jurusan : Teknik Informatika  
Fakultas : Sains dan Teknologi

Telah Disetujui, 4 Mei 2015

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Zainal Abidin, M. Kom**  
NIP. 19760613 200501 1 001

**Totok Chamidy, M. Kom**  
NIP. 19691222 200604 1 001

Mengetahui,

**Ketua Jurusan Teknik Informatika**

**Dr. Cahyo Crysdian**  
NIP. 19740424 200901 1 008

**HALAMAN PENGESAHAN**  
**OTOMATISASI PRESENSI MENGGUNAKAN GLOBAL POSITIONING**  
**SYSTEM (GPS)**

SKRPSI

Oleh :

**KHARIS ROSYADI**

**NIM. 10650112**

Telah Dipertahankan Di Depan Dewan Penguji Skripsi  
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer (S. Kom)

Tanggal, 9 Juli 2015

**Susunan Dewan Penguji :**

	<b>Tanda Tangan</b>
<b>1. Penguji Utama</b> : <u>Syahiduz Zaman, M.Kom</u> 19700502 200501 1 005	( )
<b>2. Ketua Penguji</b> : <u>M. Ainul Yaqin, M.Kom</u> 19761013 200604 1 004	( )
<b>3. Sekretaris</b> : <u>Zainal Abidin, M.Kom</u> 19760613 200501 1 004	( )
<b>4. Anggota Penguji</b> : <u>Totok Chamidy, M.Kom</u> 19691222 200604 1 001	( )

Mengetahui,  
**Ketua Jurusan Teknik Informatika**

**Dr. Cahyo Crysdiان**  
**NIP. 19740424 200901 1 008**

**SURAT PERNYATAAN  
ORISINALITAS PENELITIAN**

Saya yang bertanda tangan dibawah ini:

Nama : Kharis Rosyadi  
NIM : 10650112  
Fakultas/ Jurusan : Sains dan Teknologi/ Teknik Informatika  
Judul Penelitian : Otomatisasi Presensi Menggunakan *Global Positioning System* (GPS)

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur penjiplakan, maka saya bersedia untuk mempertanggung jawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 11 Mei 2015

Yang Membuat Pernyataan,

Kharis Rosyadi

NIM. 10650112

MOTO

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Dari Ali r.a:

كُنْ عِنْدَ اللَّهِ خَيْرَ النَّاسِ، وَكُنْ عِنْدَ النَّفْسِ شَرَّ النَّاسِ،

وَكُنْ عِنْدَ النَّاسِ رَجُلًا مِّنَ النَّاسِ

*“Jadilah engkau orang yang paling bagus menurut Allah dan yang paling jelek di matamu sendiri dan jadilah yang sewajarnya di mata orang lain”*

## HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Dengan menyebut nama Allah Yang Maha Pengasih dan Maha Penyayang.*

*Alhamdulillah... sujud syukur kepada Allah SWT, penguasa semesta alam.*

*Atas karunia, hidayah, petunjuk dan kemudahan yang Engkau berikan dalam menyelesaikan skripsi ini.*

*Sholawat dan salam selalu terlimpahkan keharibaan*

*Revolusioner Akbar Rasulullah Muhammad SAW*

### **Karya ini Saya Persembahkan Kepada:**

Bapak (Sofwan B.A) sosok ayah yang membanggakan dan yang telah menahan rasa sakitnya selama saya mengerjakan skripsi ini agar konsentrasi saya tidak terganggu dan yang selalu bekerja keras untuk keempat anaknya. Ibu (Siti Aminah) tercinta yang telah memberikan cinta, kasih sayang, doa, nasehat dan dukungan yang tak pernah ada hentinya demi kesuksesan saya. Adik (M. Lutfi Jauhar, Duratul Asfia, Alfin Nafisa) yang selalu memberikan support penuh terhadap ujian saya, kalian luar biasa.

Kepada Kyai Mukhroji Azhar, yang telah memberikan ilmu, membimbing dan mengarahkan saya menjadi lebih baik hingga sekarang dan selalu jadi panutan saya.

Keluarga Besar Tahsinul Akhlak, yang telah menemani saya menuntut ilmu hingga sekarang dan besok.

Sahabat dan teman-teman di Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang khususnya di Jurusan Teknik Informatika angkatan 2010 terima kasih untuk kebahagiaannya. Kalian luar biasa.

## KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

*Bismillahirrahmaanirrahim*

Alhamdulillah, puji syukur kehadiran Allah *Subhanahu Wa Ta'ala*, yang senantiasa memberikan perlindungan serta melimpahkan rahmat, taufik, dan hidayah-Nya kepada penulis sehingga dapat menyelesaikan skripsi dengan judul, “Otomatisasi Presensi Menggunakan Global Positioning System(GPS)” Tak lupa, shalawat serta salam kepada senantiasa tercurahkan kepada junjungan kita, Baginda Nabi Muhammad *Shalallahu 'Alaihi Wasallam*.

Dengan segala kerendahan hati, penulis menyadari bahwa dalam menyelesaikan skripsi ini tidak lepas dari peran berbagai pihak yang telah banyak memberikan bantuan, bimbingan dan dorongan. Dalam kesempatan ini penulis ingin mengucapkan terima kasih yang tak terhingga khususnya kepada:

1. Prof. Dr H. Mudjia Rahardjo, M.Si, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. drh Hj Bayyinatul Muchtaromah, M.Si, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Cahyo Crysdiyan selaku Ketua Jurusan Teknik Informatika yang telah memotivasi, membantu dan memberikan penulis arahan yang baik dan benar dalam menyelesaikan penulisan skripsi ini.
4. Zainal Abidin, M.Kom, selaku dosen pembimbing I yang telah mencurahkan waktunya dalam memberikan bimbingan, motivasi, nasihat dalam menyelesaikan karya ilmiah ini. Semoga Allah SWT melimpahkan Rahmat-Nya kepada beliau sekeluarga.
5. Totok Chamidy, M.Kom, selaku dosen pembimbing II yang telah memberikan bimbingan dan nasihatnya dalam penyusunan laporan karya ilmiah ini. Semoga Allah SWT melimpahkan Rahmat-Nya kepada beliau sekeluarga.
6. Muhammad Faisal, M.T, selaku dosen wali yang selalu memberi masukan, nasihat, motivasi dan ilmu pengetahuannya kepada penulis. Semoga Allah SWT melimpahkan Rahmat-Nya kepada beliau sekeluarga

7. Seluruh Dosen Teknik Informatika Universitas Islam Negeri Malang yang telah mengajar penulis selama empat tahun lamanya, dan memberikan dukungan untuk menyelesaikan penulisan skripsi ini.
8. Asatidz-Asatidzah yang telah mendidik dan membimbing baik dalam ilmu umum maupun ilmu agama mulai dari anak-anak hingga bisa seperti sekarang ini.
9. Orangtua, adik tersayang yang telah banyak memberikan doa, motivasi dan dorongan dalam penyelesaian skripsi ini.
10. Sahabat-sahabat dan teman-teman di Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
11. Dan semua pihak yang terlibat baik secara langsung maupun tidak langsung yang tidak dapat disebutkan satu persatu. Terima Kasih atas semua bantuan dan motivasi yang telah diberikan.

Penulis menyadari sepenuhnya bahwa sebagai manusia biasa tentunya tidak akan luput dari kekurangan dan keterbatasan. Maka dengan segenap kerendahan hati, penulis mengharapkan saran dan kritik yang dapat menyempurnakan penulisan ini sehingga dapat bermanfaat dan berguna untuk pengembangan ilmu pengetahuan.

Malang, 11 Mei 2015

Penulis

## DAFTAR ISI

HALAMAN JUDUL .....	ii
HALAMAN PERSETUJUAN .....	iii
HALAMAN PENGESAHAN .....	iv
HALAMAN PERNYATAAN .....	v
HALAMAN MOTO .....	vi
HALAMAN PERSEMBAHAN .....	vii
KATA PENGANTAR .....	viii
DAFTAR ISI .....	x
DAFTAR GAMBAR .....	xiii
DAFTAR TABEL .....	xiv
ABSTRAK .....	xv
ABSTRACT .....	xvi
<b>BAB I PENDAHULUAN</b> .....	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Identifikasi Masalah .....	5
1.3. Batasan Masalah .....	5
1.4. Tujuan Penelitian .....	5
1.5. Manfaat Penelitian .....	6
1.6. Sistematika Penulisan .....	6
<b>BAB II LANDASAN TEORI</b> .....	<b>7</b>
2.1. Manajemen Kinerja .....	7
2.2. Presensi .....	8
2.3. Smartphone .....	9
2.4. Android .....	10
2.4.1. Arsitektur Android .....	12
1. Linux Kernel .....	12
2. Libraries .....	13
3. Application Framework .....	14
4. Applications .....	15
5. Android Software Development Kit (SDK) .....	15
2.4.2. Komponen Aplikasi Android .....	15
1. Activity .....	15
2. Service .....	16
3. Broadcast Receiver .....	16
4. Content Provider .....	17

2.4.3. Struktur Projek Android .....	17
1. Root Content .....	18
2. Resource Directory .....	18
2.5. Eclipse .....	18
2.6. Android Developer Tool .....	19
2.7. Global Positioning System (GPS) .....	19
2.8. Location Based Service (LBS) .....	20
2.9. Web Server .....	20
2.10. JSON .....	21
2.11. PHP .....	24
2.12. MySQL .....	25
2.13. SQLite .....	26
2.13.1. Fitur-fitur SQLite .....	27
2.13.2. Integrasi dengan bahasa lain .....	28
2.14. XML .....	29
<b>BAB III ANALISA DAN PERANCANGAN SISTEM</b> .....	<b>32</b>
3.1. Deskripsi Sistem .....	32
3.2. Perancangan Sistem .....	34
3.2.1. Diagram Konteks .....	34
3.2.2. Data Flow Diagram level 1 .....	34
3.3. Perancangan Database .....	35
3.3.1. Tabel Presensi .....	36
3.3.2. Tabel Registrasi .....	36
3.3.3. Tabel Pegawai .....	37
3.3.4. Tabel Rekap Presensi .....	37
3.3.5. Tabel Android .....	38
3.4. Perancangan Flowchart .....	38
3.4.1. Flowchart Proses Pengambilan Lokasi .....	38
3.4.2. Flowchart Proses Penentuan Area Lokasi .....	39
3.4.3. Flowchart Proses Sinkronisasi Web Server .....	40
3.5. Desain Interface .....	41
3.5.1. Tampilan Aplikasi Android .....	41
3.5.1.1. Tampilan Halaman Utama .....	41
3.5.1.2. Tampilan Halaman History .....	42
3.5.2. Tampilan Web .....	43
3.5.2.1. Tampilan halaman login .....	43
3.5.2.2. Tampilan halaman daftar lokasi .....	44
3.5.3. Penghitungan Area Lokasi .....	45
3.5.4. Penghitungan Jarak Kampus .....	45
<b>BAB IV IMPLEMENTASI DAN UJI COBA</b> .....	<b>47</b>
4.1. Implementasi .....	47
4.1.1. Implementasi Algoritma .....	47
4.1.1.1. Tahapan Akuisisi Data Lokasi .....	47
4.1.1.2. Tahapan Penentuan Manual Area Lokasi .....	48
4.1.1.3. Tahapan Proses Sinkronisasi ke Webserver .....	50
4.1.1.4. Tahapan Proses Otomatisasi Aplikasi .....	51
4.1.1.5. Tahapan Penghitungan Manual Jarak Kampus .....	51

4.1.2. Implementasi Interface .....	52
4.1.2.1. Tampilan Awal Aplikasi Android .....	52
4.1.2.2. Tampilan Menu History .....	53
4.1.2.3. Tampilan Menu Jarak Lokasi .....	54
4.1.2.4. Tampilan Menu Settings Aplikasi .....	54
4.1.2.5. Tampilan Halaman Login Admin di Web .....	55
4.1.2.6. Tampilan Halaman Registrasi User .....	55
4.1.2.7. Tampilan Halaman Lihat User .....	56
4.1.2.8. Tampilan Halaman Laporan.....	56
4.1.2.9. Tampilan Halaman Laporan Detail User .....	57
4.1.2.10. Tampilan Halaman Daftar Lokasi Per Periode .....	58
4.2. Pengujian Sistem .....	58
4.2.1. Rencana Pengujian .....	58
4.2.2. Pengujian .....	59
4.2.2.1. Pengambilan Data Lokasi .....	59
4.2.2.2. Menentukan Area Lokasi Kampus .....	60
4.2.2.3. Melakukan Proses Otomatisasi Presensi .....	61
4.2.2.4. Menampilkan Data Lokasi Di Web Server .....	66
4.2.2.5. Menentukan Jarak Lokasi Kampus .....	67
4.3. Presensi Menurut Kajian Dalam Al-Qur'an dan Hadits .....	68
BAB V PENUTUP .....	72
5.1. Kesimpulan .....	72
5.2. Saran .....	72
DAFTAR PUSTAKA .....	73

## DAFTAR GAMBAR

Gambar 2.1. Diagram Arsitektur Android .....	12
Gambar 2.2. Contoh Class Activity .....	16
Gambar 2.3. Contoh Class Service .....	16
Gambar 2.4. Contoh Class Broadcast Receiver .....	17
Gambar 2.5. Contoh Class ContentProvider .....	17
Gambar 2.6. Struktur data object JSON .....	22
Gambar 2.7. Struktur data array JSON .....	22
Gambar 2.8. Struktur data value JSON .....	22
Gambar 2.9. Struktur data string JSON .....	23
Gambar 2.10. Struktur data angka JSON .....	23
Gambar 2.11. Contoh representasi JSON untuk suatu objek .....	24
Gambar 3.1. Koordinat Area Universitas .....	33
Gambar 3.2. Context Diagram .....	34
Gambar 3.3. Data Flow Diagram level 1 .....	35
Gambar 3.4. Entity Relational Database .....	36
Gambar 3.5. Flowchart proses pengambilan lokasi .....	39
Gambar 3.6. Flowchart proses penentuan area lokasi .....	40
Gambar 3.7. Flowchart proses sinkronisasi web server .....	41
Gambar 3.8. Tampilan halaman utama aplikasi .....	42
Gambar 3.9. Tampilan halaman history .....	43
Gambar 3.10. Tampilan halaman login web .....	44
Gambar 3.11. Tampilan halaman daftar lokasi .....	44
Gambar 4.1. Pseudocode pengambilan longitude dan latitude .....	48
Gambar 4.2. Proses penentuan area lokasi .....	49
Gambar 4.3. Pseudocode proses penentuan area lokasi .....	50
Gambar 4.4. Pseudocode proses sinkronisasi ke web server .....	50
Gambar 4.5. Pseudocode proses otomatisasi presensi .....	51
Gambar 4.6. Tampilan awal aplikasi presensi di android .....	53
Gambar 4.7. Tampilan history aplikasi di android .....	53
Gambar 4.8. Tampilan menu jarak lokasi .....	54
Gambar 4.9. Tampilan menu settings .....	54
Gambar 4.10. Tampilan Halaman login web server .....	55
Gambar 4.11. Tampilan Halaman Registrasi User .....	55
Gambar 4.12. Tampilan Halaman Lihat User .....	56
Gambar 4.13. Tampilan Halaman Laporan .....	57
Gambar 4.14. Tampilan Halaman Detail User .....	57
Gambar 4.15. Tampilan Halaman Daftar Lokasi Per Periode .....	58
Gambar 4.16. Tampilan jarak lokasi kampus .....	68

## DAFTAR TABEL

Tabel 2.1. Tabel versi Android dan versi kernel linux yang digunakan .....	13
Tabel 3.1. Tabel Presensi .....	36
Tabel 3.2. Tabel Registrasi .....	37
Tabel 3.3. Tabel Pegawai .....	37
Tabel 3.4. Tabel Rekap Presensi .....	37
Tabel 3.5. Tabel Android .....	38
Tabel 4.1. Rencana Pengujian .....	59
Tabel 4.2. Hasil Uji Input Lokasi .....	60
Tabel 4.3. Hasil Penentuan Area Lokasi .....	60
Tabel 4.4. Hasil otomatisasi pada perangkat Acer Z160 .....	61
Tabel 4.5. Hasil otomatisasi pada perangkat Xperia J .....	62
Tabel 4.6. Hasil otomatisasi pada perangkat Cross A26 .....	62
Tabel 4.7. Hasil otomatisasi pada perangkat Xperia E .....	63
Tabel 4.8. Hasil otomatisasi pada perangkat Redmi 2 .....	63
Tabel 4.9. Hasil otomatisasi pada perangkat Asus Zenfone 4 .....	64
Tabel 4.10. Hasil otomatisasi pada perangkat Asus Zenfone 4 .....	64
Tabel 4.11. Hasil otomatisasi pada perangkat Xperia L .....	65
Tabel 4.12. Hasil otomatisasi pada perangkat Xperia M .....	65
Tabel 4.13. Hasil otomatisasi pada perangkat Lenovo A6000 .....	66

## ABSTRAK

Rosyadi, Kharis 2015. **Otomatisasi Presensi Menggunakan *Global Positioning System (GPS)***. Skripsi Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing: (I) Zainal Abidin, M.Kom dan (II) Totok Chamidy, M.Kom

Kata Kunci: *Presensi, Pelacakan, Global Positioning System (GPS)*

Kedisiplinan menjadi salah satu sumber penilaian kinerja karyawan suatu organisasi atau instansi. Kedisiplinan terbagi menjadi 2, yaitu disiplin terhadap waktu dan disiplin terhadap tingkah laku atau perbuatan dalam melaksanakan prosedur kerja yang ada. Disiplin dalam melaksanakan pekerjaan berdampak pada kepuasan kerja, keselamatan, produktivitas, dan stabilitas organisasi.

Presensi merupakan salah satu tolak ukur utama untuk menjunjung tinggi nilai kedisiplinan karyawan/ anggota dalam suatu organisasi/ instansi maupun perusahaan. Penggunaan mesin presensi pada era digital ini sudah menjadi bagian dalam pelaksanaan organisasi. Salah satu perangkat presensi yang populer adalah mesin presensi *fingerprint*. Mesin ini dinilai memberikan kemudahan dan keakuratan data dengan user yang terlibat menggunakan sidik jari, yang sulit untuk melakukan pemalsuan. Namun dalam prakteknya masih ada beberapa kasus dimana karyawan menyalahgunakan mesin ini dengan melakukan tindakan indisipliner, yakni menggunakan untuk *check-in* pada waktu masuk jam kerja dan *check-out* pada jam pulang namun karyawan tersebut tidak berada di tempat ketika jam kerja.

Penelitian ini berusaha memberikan terobosan baru penggunaan mesin presensi berbasis GPS menggunakan perangkat android untuk melakukan *tracking* karyawan secara *real-time*. Sehingga bisa memantau lokasi aktivitas terkini karyawan yang bersangkutan untuk melakukan presensi. Aplikasi ini bisa mendefinisikan posisi karyawan di masing-masing gedung Universitas. Pelacakan ini akan menggunakan bantuan *web-server* untuk menyimpan data aktivitas lokasi karyawan yang akan berguna untuk rekap presensi.

Hasil dari penelitian ini menunjukkan bahwa perangkat android yang secara *default* terdapat GPS bisa melakukan *tracking* lokasi karyawan yang bisa menunjukkan posisi karyawan secara *real-time* yang berarti bisa berfungsi sebagai mesin presensi.

## ABSTRACT

Rosyadi, Kharis 2015. **Automation Presence Using Global Positioning System (GPS)**. Thesis. Informatics Department of Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University, Malang.

Adviser: (I) Zainal Abidin, M.Kom dan (II) Totok Chamidy, M.Kom

Keywords: *Presence, Tracking, Global Positioning System (GPS)*

Discipline became one of the sources of performance appraisal of an organization or agency. Discipline is divided into two, namely disciplined with time and discipline to behavior or actions in implementing the existing work procedures. Discipline in carrying out the work have an impact on job satisfaction, safety, productivity, and stability of the organization.

Presence is one of the main benchmark for upholding the values of discipline employees / members in an organization / institution or company. The use of the engine's presence in the digital age has become part of the implementation of the organization. One of the devices is a popular Presence Presence fingerprint machine. This machine is considered to provide the ease and accuracy of the data with the user involved using finger scales, which are difficult to counterfeit. However, in practice there are still some cases where employees are abusing this machine with misconduct, which is used to check in at the time of entering work hours and check-out at home, but the employee is not in place when working hours.

This study seeks to provide new breakthrough use GPS-based presence engine using android devices to track employees in real-time. So that it can monitor the location of the current activities of the employees concerned to carry out presence. This application can define the position of employees in each University building. This tracking will use the help of a web-server to store the location of employee activity data that will be useful to recap presence.

Results from this study indicate that the default android devices that are GPS can perform location tracking employees who can show the position of employees in real-time which means it can function as salty presence.

## **BAB I**

### **PENDAHULUAN**

#### **1.1. Latar Belakang**

Perkembangan informasi saat ini tumbuh dengan sangat pesat, hampir seluruh layanan diciptakan untuk mempermudah kehidupan manusia. Salah satu bentuk perkembangan tersebut adalah kemudahan seseorang untuk memperoleh data-data posisi spasial orang lain. Layanan ini bisa dimanfaatkan untuk menemukan lokasi seorang pegawai atau karyawan secara *real-time* guna meningkatkan motivasi dan etos kerja dan menghindari kemungkinan kecurangan yang dilakukan pegawai atau karyawan yang bersangkutan yang telah diatur dalam manajemen kinerja. Manajemen kinerja merupakan suatu cara untuk mendapatkan hasil yang lebih baik bagi organisasi kelompok dan individu dengan memahami dan mengolah kinerja sesuai dengan target yang telah direncanakan dengan standar dan persyaratan kompetensi yang telah ditentukan (Tohari. 2002). Manajemen kinerja bertujuan untuk menciptakan budaya pada individu dan kelompok dalam memikul tanggung jawab bagi usaha dalam meningkatkan proses kerja dan kemampuan yang berkesinambungan yang menggunakan presensi sebagai salah satu standar utama penilaian kedisiplinan.

Presensi<sup>1</sup> sebagai tolak ukur utama untuk menjunjung tinggi nilai kedisiplinan karyawan/ anggota dalam suatu organisasi/ instansi maupun perusahaan. Penggunaan mesin presensi pada era digital ini sudah menjadi bagian dalam pelaksanaan organisasi. Dalam penerapannya di berbagai organisasi ataupun perusahaan presensi terbagi menjadi beberapa macam, yakni mesin presensi kartu

---

<sup>1</sup> Presensi adalah kehadiran (lihat Kamus Besar Bahasa Indonesia tahun 2008). Sehingga presensi bisa diartikan sebagai suatu kegiatan yang dilakukan oleh suatu organisasi/ instansi maupun perusahaan sebagai standar untuk menilai kedisiplinan anggotanya berdasarkan kuantitas kehadiran

Presensi berbeda dengan absensi yang berarti ketidakhadiran (lihat Kamus Besar Bahasa Indonesia tahun 2008), absensi adalah suatu cara yang dilakukan untuk menghitung jumlah ketidakhadiran anggotanya. Namun pada kenyataannya seringkali kata absensi digunakan sebagai standar kuantitas kehadiran.

manual, mesin presensi *magnetic card*, mesin presensi biometrik, dan mesin presensi *fingerprint*.

Seiring perkembangan teknologi penggunaan beberapa perangkat presensi di beberapa media memiliki beberapa kelemahan. Penggunaan mesin presensi kartu manual dan *magnetic card* memungkinkan kemudahan presensi namun kelemahannya adalah adanya kartu yang harus dibawa anggota untuk melakukan presensi sehingga apabila kartu tidak dibawa atau hilang akan membuat karyawan yang bersangkutan tidak bisa melakukan presensi, dan keadaan ini merugikan kedua belah pihak dalam suatu organisasi/ instansi maupun perusahaan. Selain itu kelemahan yang sangat rentan terjadi kecurangan adalah apabila kartu presensi dititipkan oleh rekan kerja sehingga presensi bisa dilakukan oleh rekan kerja yang bersangkutan tanpa sepengetahuan sistem.

Mesin presensi biometrik dan mesin presensi *fingerprint* memiliki beberapa kelemahan diantaranya rentan terhadap distorsi yang disebabkan oleh kotoran, rentan terhadap penyusutan menggunakan jari palsu (Sravya, 2012). Sistem ini juga akan berjalan lambat apabila dipakai dalam jangka waktu yang panjang karena sensor yang berfungsi untuk mendeteksi karakteristik pegawai/ anggota akan mengalami penyusutan pemakaian seiring berjalannya waktu. Begitu juga akan terjadi antrian yang panjang apabila pegawai atau anggota organisasi/ instansi maupun perusahaan datang secara bersamaan dalam satu waktu tertentu sehingga mengakibatkan proses presensi menjadi lebih lama.

Untuk mengatasi beberapa kelemahan pada mesin presensi terdahulu maka dalam penulisan skripsi ini dibuatlah sistem presensi menggunakan *Location Based Service*. Perangkat ini memanfaatkan *Global Positioning System* (GPS) yang dapat melacak informasi spasial karyawan atau pegawai dalam area Universitas. Sistem akan membatasi cakupan area Universitas dengan area non-Universitas berupa titik koordinat yang diambil langsung dari Google Map API. Ketika perangkat aktif maka sistem akan mengirimkan informasi spasial karyawan dengan memanfaatkan *Location Based Service* dan server akan mendapat

koordinat dimana posisi karyawan yang bersangkutan berada. Sehingga apabila karyawan berada pada area Universitas maka karyawan dinyatakan *check-in* oleh sistem dan karyawan akan memperoleh notifikasi telah melakukan presensi. Namun ketika karyawan berada di area non-Universitas maka sistem akan mengirimkan notifikasi pada karyawan bahwa perangkat keluar dari area Universitas. Skema ini melakukan *tracking* dalam *range* minimal 30 menit sehingga informasi spasial karyawan bisa terdeteksi oleh sistem secara otomatis.

Perangkat *mobile smartphone* Android menjadi salah satu alternatif pilihan yang bisa memenuhi kebutuhan tersebut. Dengan memanfaatkan teknologi *Global Positioning System (GPS)*, sistem presensi karyawan dapat melakukan pelacakan posisi karyawan Universitas secara tepat dan akurat. Sehingga perangkat ini membantu meminimalisir kemungkinan kecurangan kedisiplinan karyawan.

Pembahasan kedisiplinan ini sesuai dengan pernyataan Al-Quran dalam surat An Nisa ayat 59, yang berbunyi:

يَا أَيُّهَا الَّذِينَ ءَامَنُوا أَطِيعُوا اللَّهَ وَأَطِيعُوا الرَّسُولَ وَأُولِي الْأَمْرِ مِنْكُمْ <sup>ص</sup> فَإِنْ تَنَزَعْتُمْ فِي شَيْءٍ فَرُدُّوهُ إِلَى اللَّهِ وَالرَّسُولِ إِنْ كُنْتُمْ تُؤْمِنُونَ بِاللَّهِ وَالْيَوْمِ الْآخِرِ <sup>ج</sup> ذَلِكَ خَيْرٌ وَأَحْسَنُ تَأْوِيلًا.

*Hai orang-orang yang beriman, taatilah Allah dan taatilah Rasul (Nya), dan ulil amri di antara kamu. kemudian jika kamu berlainan Pendapat tentang sesuatu, Maka kembalikanlah ia kepada Allah (Al Quran) dan Rasul (sunnahnya), jika kamu benar-benar beriman kepada Allah dan hari kemudian. yang demikian itu lebih utama (bagimu) dan lebih baik akibatnya.*

Allah SWT memerintahkan kita untuk taat (disiplin) terhadap kepada-Nya dalam hal ini taat menjalankan perintah-Nya dan menjauhi larangan-Nya. Allah juga memerintahkan kita untuk taat pada rasul dan pemerintah terkait. Dalam kaitannya dengan penulisan skripsi ini taat pada hal ini dapat dilakukan dengan taat kepada instansi yaitu mengikuti segala aturan yang ditetapkan instansi. Nilai taat tersebut dapat dicerminkan pada diri karyawan dengan berdisiplin, termasuk

disiplin dalam melaksanakan pekerjaan sesuai jam kerja yang telah ditetapkan instansi terkait. Cermin taat juga dapat dinilai dari bagaimana seorang karyawan memikul tugas yang telah menjadi tanggung jawabnya. Seperti yang telah dijelaskan pada hadits di bawah ini:

عَنْ عَبْدِ اللَّهِ بْنِ عُمَرَ يَقُولُ سَمِعْتُ رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ كُلُّكُمْ رَاعٍ وَكُلُّكُمْ مَسْئُولٌ عَنْ رَعِيَّتِهِ الْإِمَامُ رَاعٍ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ وَالرَّجُلُ رَاعٍ فِي أَهْلِهِ وَهُوَ مَسْئُولٌ عَنْ رَعِيَّتِهِ وَالْمَرْأَةُ رَاعِيَةٌ فِي بَيْتِ زَوْجِهَا وَمَسْئُولَةٌ عَنْ رَعِيَّتِهَا وَالْخَادِمُ رَاعٍ فِي مَالِ سَيِّدِهِ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ قَالَ وَحَسِبْتُ أَنْ قَدْ قَالَ الرَّجُلُ رَاعٍ فِي مَالِ أَبِيهِ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ وَكُلُّكُمْ رَاعٍ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ (رواه البخاري ومسلم والترمذي)

*Dari Abdullah bin Umar r.a., ia berkata: Aku mendengar Rasulullah SAW bersabda: Kamu semua adalah pemimpin dan harus bertanggung jawab atas kepemimpinannya. Seseorang imam adalah pemimpin dan harus bertanggung jawab atas kepemimpinannya. Seseorang suami adalah pemimpin bagi istrinya adalah seorang pemimpin dalam rumah tangga dan harus bertanggung jawab atas kepemimpinannya. Pembantu adalah pemelihara terhadap harta tuannya, dia harus bertanggung jawab atas kepemimpinannya. Abdullah berkata: Saya kira (Rasulallah) bersabda juga dan seseorang anak adalah pemelihara milik orang tuanya, dia harus bertanggung jawab atas peliharaannya itu. Dan kamu semua adalah pemimpin dan harus bertanggung jawab atas kepemimpinannya. (HR Bukhari, Muslim dan Tirmidzi)*

Tugas adalah amanat, apapun jabatan yang ada pada diri seorang seseorang termasuk karyawan, maka harus mempertanggungjawabkan tugas yang telah dibebankan kepadanya dihadapan yang dipimpin dan di dalam pengadilan Allah kelak. Tak seorangpun mampu melepaskan diri dari tanggung jawab itu. Oleh karena itu, dia harus bersikap adil dan bijaksana dalam menjalankan tugasnya. Apabila lengah dan mengabaikan tanggung jawabnya, maka celakalah dia sebab akan menyengsarakannya dan yang telah dipimpinnya. Sehingga dalam

praktiknya kewajiban seorang karyawan adalah untuk melakukan tugas dan tanggung jawabnya terhadap instansi terkait.

Pembuatan aplikasi presensi pada skripsi ini juga bertujuan untuk menjunjung tinggi nilai kedisiplinan dan tanggung jawab yang ada pada karyawan. Yaitu dengan melakukan presensi berbasis GPS yang mana ini akan sangat menuntungkan pihak instansi yang bisa membantu memantau aktivitas karyawan melalui perangkat *android*-nya. Dan harapannya karyawan bisa menghormati aturan instansi mengenai kedisiplinan, dan tanggung jawab serta menjunjung tinggi nilai kedisiplinan. Sehingga pada akhirnya etos kerja dan produktifitas instansi meningkat dan terciptalah suasana kerja yang kondusif untuk bekerja.

### **1.2. Identifikasi Masalah**

Berdasarkan latar belakang yang telah diuraikan di atas identifikasi masalah untuk melacak aktifitas posisi karyawan selama jam kerja secara *real-time* menggunakan *smartphone* Android.

### **1.3. Batasan Masalah**

Adapun batasan masalah pada penulisan skripsi ini adalah:

- a. Aplikasi dibuat di atas *platform* Android sehingga hanya dapat dijalankan pada perangkat yang menggunakan *platform* Android.
- b. Ruang lingkup objek penelitian adalah Universitas Islam Negeri Maulana Malik Ibrahim Malang
- c. Sistem digunakan untuk mendeteksi informasi spasial karyawan/ pegawai Universitas
- d. Sistem ini terhubung dengan *web server* secara *online* dan melakukan pendeteksian secara periodik menurut pengaturan waktu yang ditetapkan.

### **1.4. Tujuan Penelitian**

Tujuan perancangan sistem ini adalah dalam skripsi ini adalah bagaimana membuat suatu sistem presensi karyawan menggunakan *Global Positioning System* (GPS).

### **1.5. Manfaat Penelitian**

Manfaat pembuatan aplikasi presensi ini adalah membuat sistem presensi otomatis yang lebih akurat datanya berbasiskan lokasi pengguna secara *real-time*.

### **1.6. Sistematika Penulisan**

Penulisan skripsi ini tersusun dalam lima bab dengan sistematika penulisan sebagai berikut:

#### **BAB I PENDAHULUAN**

Pendahuluan, membahas tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan penulisan skripsi, metodologi, dan sistematika penulisan.

#### **BAB II LANDASAN TEORI**

Landasan teori berisikan beberapa teori yang mendasari dalam penyusunan tugas akhir ini. Adapun yang dibahas dalam bab ini adalah dasar teori yang berkaitan dengan pembahasan mengenai presensi berbasis lokasi.

#### **BAB III ANALISA DAN PERANCANGAN**

Menganalisa kebutuhan sistem untuk membuat aplikasi meliputi langkah-langkah pembuatan aplikasi presensi untuk melakukan akses lokasi berbasis GPS. Dan mensinkronkan dengan *web server online*.

#### **BAB IV IMPLEMENTASI DAN UJI COBA**

Menjelaskan mengenai pengujian aplikasi berbasis lokasi yang secara real-time dan diunggah ke *web server online*.

#### **BAB V PENUTUP**

Berisi kesimpulan dan saran

## BAB II

### LANDASAN TEORI

#### 2.1. Manajemen Kinerja

Kata Manajemen Kinerja merupakan penggabungan dari kata manajemen dan kinerja. Manajemen berasal dari kata *to manage* yang berarti mengatur. Menurut George R Terry dalam bukunya *Principles of Management*, Manajemen merupakan suatu proses yang menggunakan metode ilmu dan seni untuk menerapkan fungsi-fungsi perencanaan, pengorganisasian, pengarahan dan pengendalian pada kegiatan-kegiatan dari sekelompok manusia yang dilengkapi dengan sumber daya/faktor produksi untuk mencapai tujuan yang sudah ditetapkan lebih dahulu, secara efektif dan efisien. Sedangkan menurut John R Schermerhorn Jr dalam bukunya *Management*, manajemen adalah proses yang mencakup perencanaan, pengorganisasian, pengarahan dan pengendalian terhadap penggunaan sumber daya yang dimiliki, baik manusiadan *material* untuk mencapai tujuan.

Dari beberapa definisi manajemen yang diberikan oleh para ahli, dapat disimpulkan manajemen mencakup tiga aspek, yaitu:

1. Pertama : manajemen sebagai proses
2. Kedua : adanya tujuan yang telah ditetapkan
3. Ketiga : mencapai tujuan secara efektif dan efisien

Kata kinerja merupakan singkatan dari kinetika energi kerja yang padanannya dalam bahasa Inggris adalah *performance*, yang sering diindonesiakan menjadi kata performa.

Dengan demikian, kinerja adalah kesediaan seseorang atau kelompok orang untuk melakukan sesuatu kegiatan dan menyempurnakannya sesuai dengan tanggung jawabnya dengan hasil seperti yang diharapkan.

Dari kedua kata manajemen dan kinerja, jika digabungkan menjadi satu kata baru yaitu Manajemen Kinerja (*Performance Management*). Manajemen kinerja merupakan suatu cara untuk mendapatkan hasil yang lebih baik bagi organisasi kelompok dan individu dengan memahami dan mengolah kinerja sesuai dengan target yang telah direncanakan dengan standar dan persyaratan kompetensi yang telah ditentukan (Tohari. 2002). Manajemen kinerja bertujuan untuk menciptakan budaya pada individu dan kelompok dalam memikul tanggung jawab bagi usaha dalam meningkatkan proses kerja dan kemampuan yang berkesinambungan yang menggunakan presensi sebagai salah satu standar utama penilaian kedisiplinan.

## 2.2. Presensi

Adanya kesalahpahaman penggunaan istilah kehadiran dan ketidakhadiran yang beredar di masyarakat menyebabkan perlunya bagi penulis menjelaskan perbedaan istilah yang sesungguhnya

### 1. Presensi

Presensi adalah kehadiran (lihat Kamus Besar Bahasa Indonesia tahun 2008). Sehingga presensi bisa diartikan sebagai suatu kegiatan yang dilakukan oleh suatu organisasi/instansi maupun perusahaan sebagai standar untuk menilai kedisiplinan anggotanya berdasarkan kuantitas kehadiran.

### 2. Absensi

Presensi berbeda dengan absensi yang berarti ketidakhadiran (lihat Kamus Besar Bahasa Indonesia tahun 2008), absensi adalah suatu cara yang dilakukan untuk menghitung jumlah ketidakhadiran anggotanya. Namun pada kenyataannya seringkali kata absensi digunakan sebagai standar kuantitas kehadiran.

Definisi ketidakhadiran (*absenteism*) adalah kegagalan untuk melapor pada waktu kerja. Dengan kata lain ketidakhadiran merupakan kegagalan seorang karyawan untuk hadir di tempat kerja pada hari kerja. Ketidakhadiran berbeda dengan terlambat (*lateness*) atau lamban (*tardiness*) yang menunjukkan kegagalan untuk

datang tepat waktu. Dari penjelasan diatas dapat disimpulkan bahwa penggunaan istilah yang benar adalah kata “presensi” yang mengandung pemahaman sebagai satu bentuk kehadiran seseorang di sebuah tempat untuk memenuhi kewajibannya. Secara umum terdapat dua jenis sistem presensi yang jamak digunakan, yaitu:

#### A. Presensi Manual

Sistem ini dalam proses pendataannya masih menggunakan tulisan tangan (tanda tangan).

#### B. Presensi Otomatis

Sistem ini dalam proses pendataannya sudah menggunakan alat bantu komputerisasi. Untuk sebagian institusi yang memiliki jumlah karyawan berskala besar, beberapa diantaranya sudah menggunakan barcode reader, fingerprint, bahkan retinal-scan sebagai alat penunjang efektifitas kinerja perusahaan.

Dalam aplikasi ini presensi dipakai dalam objek kehadiran pegawai yang berkenaan dengan tugas dan kewajibannya. Pada umumnya instansi atau lembaga selalu memperhatikan pegawainya untuk datang dan pulang tepat waktu, sehingga pekerjaan tidak tertunda. Ketidakhadiran seorang pegawai berbanding lurus terhadap produktivitas kerja, sehingga presensi menjadi salah satu tolak ukur profesionalisme karyawan terhadap kinerja suatu organisasi atau instansi untuk bisa mencapai tujuan secara optimal.

### 2.3. *Smartphone*

*Smartphone* adalah suatu *mobile device* yang tertanam sistem operasi sehingga memiliki kemampuan tingkat tinggi dan memiliki kemampuan menyerupai komputer (wikipedia.org). *Smartphone* adalah kombinasi antara telepon genggam, *Personal Digital Assistant (PDA)*, *media player*, *digital camera*, dan *Global Positioning System (GPS)*. Selain fitur-fitur tersebut, di dalam *smartphone* tertanam pula Bluetooth dan Wi-Fi serta kemampuan untuk mengakses internet dengan baik.

*Smartphone* merupakan *mobile device* untuk masa depan yang menawarkan berbagai fungsi kemudahan dalam kemampuan nirkabel, daya komputasi, dan penyimpanan *on-board*. Sehingga secara tidak langsung *smartphone* memiliki kemampuan untuk mempermudah kegiatan manusia.

Beberapa sistem operasi yang terdapat dalam *smartphone* adalah Symbian OS, iOS, Blackberry OS, Windows Mobile, Palm OS, Firefox OS, dan Android. Dengan kelebihan dan kekurangan yang dimiliki masing-masing sistem operasi berbeda.

#### 2.4. Android

Android adalah sebuah sistem operasi untuk perangkat *mobile* yang mencakup aplikasi sistem operasi, *middleware* dan aplikasi. Android SDK menyediakan fitur dan API yang diperlukan untuk mulai mengembangkan aplikasi pada *platform* Android dengan menggunakan bahasa pemrograman Java.

Menurut DiMarzio (2008, p6), Android, sebagai suatu sistem, adalah sistem operasi berbasis Java yang berjalan pada kernel Linux 2.6. Sistem ini sangat ringan dan berfitur lengkap.

Android dikembangkan oleh Open Handset Alliance, Android membawa inovasi dengan internet dan keterbukaan terhadap perangkat *mobile*. Android memberikan sebuah *software* yang lengkap untuk aplikasi pada telepon seluler : sistem operasi, *middleware* dan aplikasi kunci dalam *mobile*. Android memiliki empat karakteristik sebagai berikut:

##### 1. Terbuka

Android dibangun untuk memungkinkan developers membuat aplikasi *mobile* yang menarik yang bisa mengambil keuntungan penuh dari semua *handset* yang ditawarkan dengan maksud untuk menjadi aplikasi yang benar-benar terbuka. Sebagai contoh, sebuah aplikasi dapat memanggil salah satu dari fungsi inti telepon seluler seperti membuat panggilan, mengirim pesan teks, atau menggunakan kamera, memungkinkan developer untuk menciptakan pengalaman yang lebih kaya dan lebih kohesif bagi pengguna. Android dibuat di atas Linux

Kernel terbuka dan memanfaatkan mesin virtual khusus yang dirancang untuk mengoptimalkan memori dan sumber daya hardware dalam lingkungan mobile. Android merupakan open source, bebas dikembangkan untuk menyertakan teknologi baru yang mutakhir yang baru muncul. Platform ini akan terus berkembang seiringnya komunitas developer yang bekerja sama untuk membangun aplikasi mobile yang inovatif.

#### 2. Semua aplikasi diciptakan sama

Android tidak membedakan antara aplikasi inti dari telepon seluler dan aplikasi pihak ketiga (*third-party application*). Semua aplikasi dapat dibangun dan untuk memiliki akses yang sama terhadap kemampuan telepon seluler dalam menyediakan berbagai aplikasi dan layanan kepada pengguna. Dengan perangkat yang dibuat berbasis *platform* Android, pengguna dapat sepenuhnya menyesuaikan telepon seluler sesuai keinginan mereka.

#### 3. Memecahkan hambatan pada aplikasi

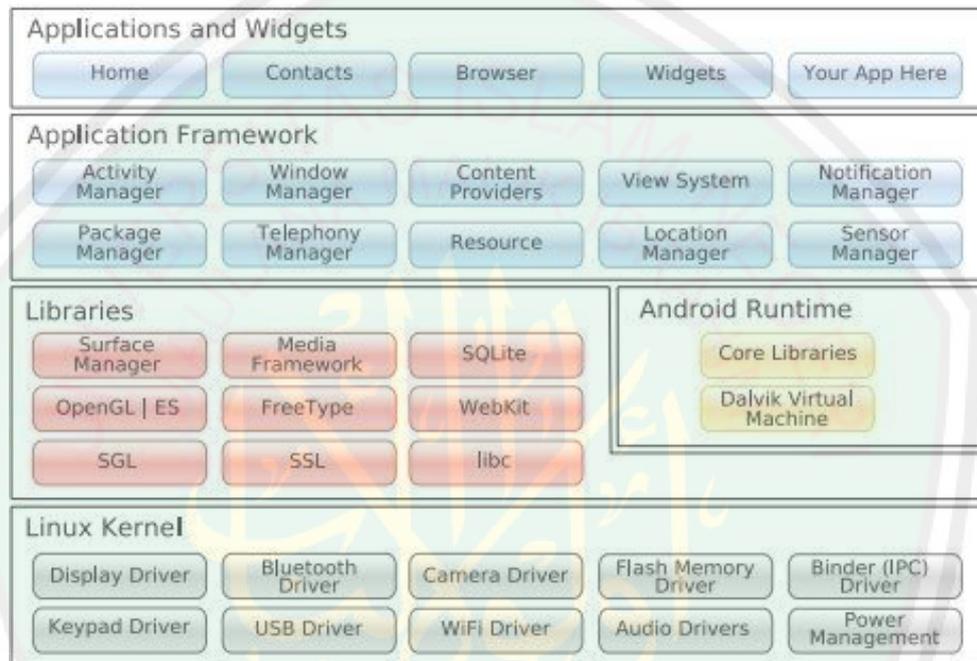
Android memecah hambatan untuk membangun aplikasi baru dan inovatif. Sebagai contoh, seorang developer dapat menggabungkan informasi dari web dengan data pada telepon seluler individu – seperti kontak pengguna, kalender, atau lokasi geografis – untuk memberikan pengalaman pengguna yang lebih relevan.

#### 4. Pengembangan aplikasi yang cepat dan mudah

Android menyediakan akses yang luas kepada developer untuk menggunakan berbagai libraries yang diperlukan dan tools yang dapat dipakai dalam pembuatan aplikasi. Sebagai contoh, Android memungkinkan developer untuk mendapatkan lokasi dari perangkat, dan memungkinkan perangkat tersebut untuk berkomunikasi satu sama lain sehingga memungkinkan aplikasi sosial peer-to-peer.

### 2.4.1. Arsitektur Android

Android merupakan sekumpulan tumpukan *software* yang berjalan pada satu sistem yang terintegrasi dalam suatu *device*. Setiap satu lapisan menghimpun beberapa program untuk mendukung sistem-sistem spesifik dari sistem operasi.



Gambar 2.1 Diagram Arsitektur Android.

#### 1. Linux Kernel

Tumpukan paling bawah dalam arsitektur Android adalah Kernel Linux. Kernel linux tidak pernah berinteraksi langsung dengan pengguna dan pengembang aplikasi, namun kernel menjadi bagian terpenting dari keseluruhan sistem yang bekerja. Kernel linux memiliki beberapa fungsi dalam sistem Android, diantaranya:

- 1) *Hardware Abstraction*
- 2) Program pengelolaan memori
- 3) Pengaturan keamanan
- 4) *Power Management Software*

- 5) *Other Hardware Drivers* (*Drivers* adalah program yang mengatur perangkat *hardware*)
- 6) *Support for Shared Libraries*
- 7) *Network Stack*

Seiring berkembangnya android, kernel linux juga mengalami perkembangan pula. Berikut ini adalah tabel perbedaan versi android dan versi kernelnya

*Tabel 2.1. Tabel versi Android dan versi kernel linux yang digunakan*

Versi Android	Versi kernel linux
1.0	2.6.25
1.5 (Cupcake)	2.6.27
1.6 (Donut)	2.6.29
2.2 (Froyo)	2.6.32
2.3 (Gingerbread)	2.6.35
3.0 (Honeycomb)	2.6.36
4.0.x (Ice Cream Sandwich)	3.0.1
4.1/4.2 (Jelly Bean)	3.0.31

## 2. Libraries

Lapisan berikutnya dalam arsitektur Android adalah *Library*. *Library* menerima seperangkat instruksi untuk memandu perangkat dalam menangani perbedaan jenis data. Contohnya, *playback* dan *recording* adalah fitur dari format audio dan video yang dipandu oleh *Media Framework Library*.

Berikut adalah aplikasi-aplikasi yang terdapat pada *Library*:

- 1) *Surface Manager*: Mengatur *interface* aplikasi
- 2) SGL: Grafis 2D
- 3) Open GLES: 3D Library

4) *Media Framework*: Mendukung *playbacks* dan *recording* sebagai variasi dari format audio, video, dan gambar.

5) *Free Type*: sebagai proses *rendering* font

6) WebKit: Mesin pencari

7) libc (System C libraries)

8) SQLite

9) Open SSL

Satu lapisan dengan *library* terdapat lapisan *Android Runtime* yang berisi librari-librari Java. Programmer aplikasi Android membangun aplikasi mereka menggunakan bahasa pemrograman java disinilah letak librari-librari tersebut. Selain itu juga terdapat *Dalvik Virtual Runtime*, adalah *software* yang bertanggung jawab untuk semua aplikasi yang berjalan pada perangkat Android yang bersifat *open-source*. Selain itu juga melakukan tugas:

- 1) Mengoptimalkan kebutuhan memori yang rendah
- 2) Didesain untuk mengijinkan penggunaan *Virtual Machine* misalnya menjalankan aplikasi sekali
- 3) Melakuakan Register sesuai dengan *Virtual Machine*

### 3. *Application Framework*

Aplikasi Android secara langsung berinteraksi dengan lapisan ini dalam arsitektur Android. Program ini mengelola fungsi dasar telepon, seperti pengelolaan sumber daya, pengelolaan panggilan suara, dan lain-lain.

Aplikasi-aplikasi penting dalam lapisan Application Framework:

- 1) Activity Manager, mengelola aktivitas daur hidup dari aplikasi.
- 2) Content Provider, mengelola *sharing* data antar aplikasi.

- 3) Telephony Manager, mengelola semua panggilan suara.
- 4) Location Manager, mengelola lokasi menggunakan *Global Positioning System*(GPS), jadi secara default seluruh perangkat Android mendukung penggunaan GPS.
- 5) Resource Manager, mengelola seluruh sumber daya yang digunakan di sistem Android.

#### 4. Applications

Lapisan ini adalah lapisan paling tinggi dalam arsitektur Android. Secara umum, pengguna perangkat Android berinteraksi dengan lapisan ini. Seperti penggunaan fungsi dasar telepon, membuat panggilan telepon, mengakses mesin pencari, dan lain-lain. Selain pengguna, lapisan ini juga digunakan oleh banyak pengembang dan *programmer*.

#### 5. Android Software Development Kit(SDK)

*Android Software Development Kit*(SDK) adalah suatu program aplikasi yang disediakan google secara resmi yang menyediakan API libraries dan kebutuhan developer tools untuk membangun, menguji, dan men-*debug* aplikasi Android.

##### 2.4.2. Komponen Aplikasi Android

Aplikasi android di tulis dalam bahasa pemrograman bahasa java. Kode java dikompilasi bersama dengan data file resource yang dibutuhkan oleh aplikasi, dimana prosesnya di-*package* oleh tools yang dinamakan *apt tools* kedalam paket android sehingga menghasilkan file dengan ekstensi apk. File apk itulah yang kita sebut dengan aplikasi, dan nantinya dapat di install di perangkat mobile.

Ada empat jenis komponen pada aplikasi android yaitu :

##### 1. Activity

Suatu activity akan menyajikan *user interface* (UI) kepada pengguna, sehingga pengguna dapat melakukan interaksi. Sebuah aplikasi android bisa jadi hanya memiliki satu activity, tetapi umumnya aplikasi memiliki banyak activity

tergantung pada tujuan aplikasi dan desain dari aplikasi tersebut. Satu activity biasanya akan dipakai untuk menampilkan aplikasi atau yang bertindak sebagai user interface (UI) saat aplikasi diperlihatkan kepada user. Untuk pindah dari satu activity ke activity lain kita dapat melakukannya dengan satu event, misalnya click tombol, memilih opsi atau menggunakan triggers tertentu. Secara hirarki sebuah windows activity dinyatakan dengan method activity.

Sebuah Activities di implementasikan sebagai dari class Activity seperti berikut :

```
public class MainActivity extends Activity {
}

```

**Gambar 2.2 Contoh class activity**

## 2. Service

Service tidak memiliki graphic user interface (GUI), tetapi *service* berjalan secara *background*, sebagai contoh dalam memainkan music atau mengambil data dari jaringan, tetapi service harus berada dalam kelas induknya. Misalnya, media player sedang memutar lagu dari list yang ada, aplikasi ini akan memiliki dua atau lebih activity yang memungkinkan user untuk memilih lagu misalnya, atau menulis sms sambil media player sedang jala. Untuk menjaga musik tetap dijalankan, activity player dapat menjalankan service. Service dijalankan pada thread utama dari proses aplikasi.

Sebuah Services di implementasikan sebagai dari class Services seperti berikut :

```
public class MyService extends Service{
}

```

**Gambar 2.3 Contoh class Service**

## 3. Broadcast Receiver

Broadcast Receiver berfungsi menerima dan bereaksi untuk menyampaikan notifikasi. Contoh broadcast seperti notifikasi zona waktu berubah, baterai low,

gambar telah selesai diambil oleh kamera, atau perubahan referensi bahasa yang digunakan. Broadcast receiver tidak memiliki user interface (UI), tetapi memiliki sebuah activity untuk merespon informasi yang mereka terima, atau mungkin menggunakan notification manager untuk memberi tahu kepada pengguna, seperti lampu latar atau vibrating perangkat dan lain sebagainya.

Sebuah Broadcast Receivers di implementasikan sebagai subclass dari class BroadcastReceivers seperti berikut :

```
public class MyReceiver extends BroadcastReceiver {
}

```

**Gambar 2.4 Contoh class Broadcast Receiver**

#### 4. Content Provider

Content provider membuat kumpulan aplikasi data secara spesifik sehingga bisa digunakan oleh aplikasi lain. Data disimpan dalam file sistem seperti database SQLite. Content provider menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu activity, misalnya ketika menggunakan aplikasi yang membutuhkan peta, atau aplikasi yang membutuhkan untuk mengakses data kontak dan navigasi.

Sebuah Content Providers di implementasikan sebagai subclass dari class ContentProviders dan harus menerapkan satu set standar API yang memungkinkan aplikasi lain untuk melakukan transaksi.

```
public class MyContentProvider extends ContentProvider {
}

```

**Gambar 2.5 Contoh class ContentProvider**

#### 2.4.3 Struktur Projek Android

Project Android dibangun berdasarkan direktori yang spesifik seperti saat kita membuat project Java. Untuk membantu memahaminya, kita mulai dengan penjelasan di bawah.

## 1. Root Content

Ketika kita membuat sebuah project Android, kita akan memiliki 5 item utama dalam direktori rootproject kita.

- 1.1. AndoidManifest.xml, adalah file XML yang mendeskripsikan aplikasi yang dibangun dankomponen (activities, services, etc) apa saja yang disediakan oleh aplikasi.
- 1.2. build.xml, adalah script Ant untuk meng-compile aplikasi dan menginstallnya dalam device
- 1.3. bin/, yang memuat aplikasi setelah di-compile
- 1.4. src/, yang memuat file-file source code java untuk pembuatan aplikasi
- 1.5. res/, yang memuat “seluruh sumber”, seperti icon, GUI Layout, dan sejenisnya.
- 1.6. assets/, yang memuat file-file static lainnya yang ingin dikemas bersama aplikasi untuk di-deploy ke dalam device

## 2. Resources Directory

Dalam project Android, terdapat direktori res/ yang memuat “sumber-sumber” (file tetap seperti gambar yang dikemas bersama dengan aplikasi). Beberapa sub direktori yang akan dijumpai atau dibuat di bawah direktori res/ terdiri dari:

1. res/drawable/ untuk gambar (PNG, JPEG, dll)
2. res/layout/ untuk spesifikasi UI layout yang dibuat dengan XML
3. res/raw/ for general-purpose files (misalnya CSV File of Account Information)
4. res/values/ untuk menyimpan nilai strings, dimensions dan sejenisnya
5. res/xml/ for other general purpose XML Files you wish to ship.

## 2.5. Eclipse

Eclipse adalah sebuah IDE(Integrated Development Environment) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform. Eclipse bersifat *multiplatform*, dapat dijalankan diberbagai sistem operasi desktop. *Multi-language*, mendukung berbagai macam bahasa pemrograman seperti: java,

C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya. Multi-role, bisa digunakan sebagai perangkat pengembangan web, dokumentasi, maupun *test* perangkat lunak.

## 2.6. Android Developer Tool(ADT)

*Android Developer Tool*(ADT) adalah plugin yang menyediakan perangkat-perangkat untuk pengembangan aplikasi Android yang ter-integrasi dengan Eclipse IDE. ADT memiliki keunggulan dalam pengaksesan GUI(*Graphical User Interface*)sehingga pengembang aplikasi dapat dengan mudah membuat *prototipe*, melakukan, dan membangun antarmuka aplikasi.

Berikut ini adalah beberapa fitur unggulan dari ADT:

1. Terintegrasi dengan project aplikasi Android untuk melakukan pembuatan, pembangunan, pemaketan, pemasangan, dan melakukan *debug* program dengan mudah.
2. Peralatan SDK yang terintegrasi dengan Eclipse
3. Bahasa pemrograman java dan Editor XML,
4. Dokumentasi untuk Android APIs

## 2.7. Global Positioning System(GPS)

*Global Positioning System*(GPS) merupakan perangkat yang memberikan informasi kepada pengguna menurut posisi, navigasi, dan waktu. Sistem ini terdiri dari 3 segmen: segmen angkasa, segmen pengendali, dan segmen pengguna.

- a. Segmen angkasa(Space segment), terdiri dari kumpulan satelit yang memancarkan sinyal radio kepada pengguna.
- b. Segmen pengendali(Control Segment), segmen yang berfungsi untuk mengontrol orbit satelit, lokasi, ketinggian, dan kecepatan. Segmen ini mengirimkan kepada alat navigasi kita berupa data ephemeris, yaitu data yang sudah dikoreksi dengan tepat dari satelit.

- c. Segmen pengguna (User Segment), adalah perangkat GPS yang digunakan oleh pengguna untuk setiap aspek kehidupan modern.

Tingkat akurasi perangkat GPS bergantung penuh terhadap satelit sehingga perangkat ini tidak akan bisa bekerja secara maksimal apabila mengalami gangguan pada sinyal satelit. Beberapa gangguan yang dapat mengurangi kekuatan sinyal adalah kondisi geografi, hutan, air, gedung-gedung, serta alat-alat elektronik yang dapat mengeluarkan gelombang elektromagnetik.

### **2.8. Location Based Service (LBS)**

*Location Based Service (LBS)* adalah layanan *mobile* yang memiliki kemampuan untuk menyediakan informasi secara *real-time* berdasar lokasi pengguna. LBS saat ini banyak digunakan diberbagai media sosial sebagai layanan hiburan. Dengan kemudahan dalam pengaksesan perangkat *mobile* terutama *mobile network* dengan menggunakan posisi geografis dari perangkat *mobile*. Fitur ini disebut *geo-tagging*.

*Geographic Information System (GIS)* adalah jantung dari *Location Based Service (LBS)* dalam menyediakan semua fungsi dalam LBS (Priyanka, 2012). Pertama kita mungkin mengirimkan informasi lokasi pada bagian kontrol. Pada bagian ini menyediakan layanan yang biasa digunakan seperti pada aplikasi *tracking* lokasi. Kedua, menggunakan informasi lokasi untuk membuat keputusan dalam komunikasi, seperti user yang menonaktifkan aplikasi *instant messaging* ketika mengendara secara otomatis. Mengirim informasi lokasi untuk mengontrol dalam *men-tracking* lokasi biasanya mewakili koordinat geospasial atau *men-tracking* alamat penduduk.

### **2.9. Web Server**

*Web server* merupakan sebuah proses yang berjalan pada komputer *host*, yang menunggu *request* dari HTTP masuk. *Web server* menyediakan akses terhadap informasi dan layanan yang disimpan dalam web. (Eaglestone dan Ridley, 2001, p209)

Menurut Hoffer, Prescott, dan Topi (2009, p469), *web server* digunakan untuk memproses permintaan dari klien dan kemudian mengembalikan halaman HTML kepada klien.

## 2.10. JSON

JSON (JavaScript Object Notation) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data.

JSON terbuat dari dua struktur:

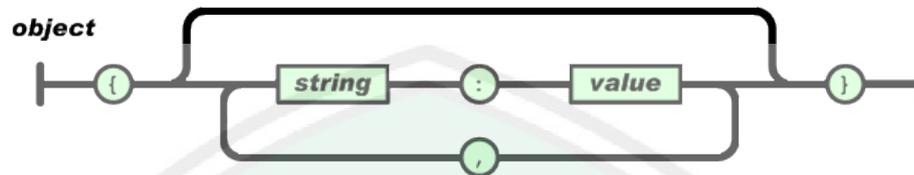
1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (object), rekaman (record), struktur (struct), kamus (dictionary), tabel hash (hash table), daftar berkunci (keyed list), atau associative array.
2. Daftar nilai terurutkan (an ordered list of values). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (array), vektor (vector), daftar (list), atau urutan (sequence).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemrograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini.

JSON menggunakan bentuk sebagai berikut:

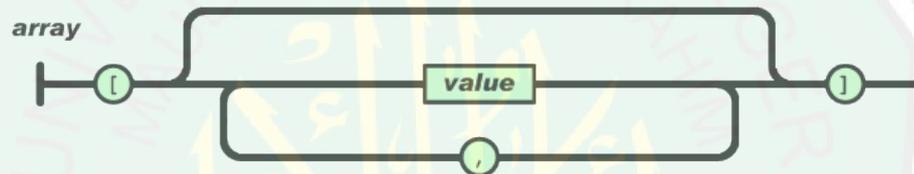
1. Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap

nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).



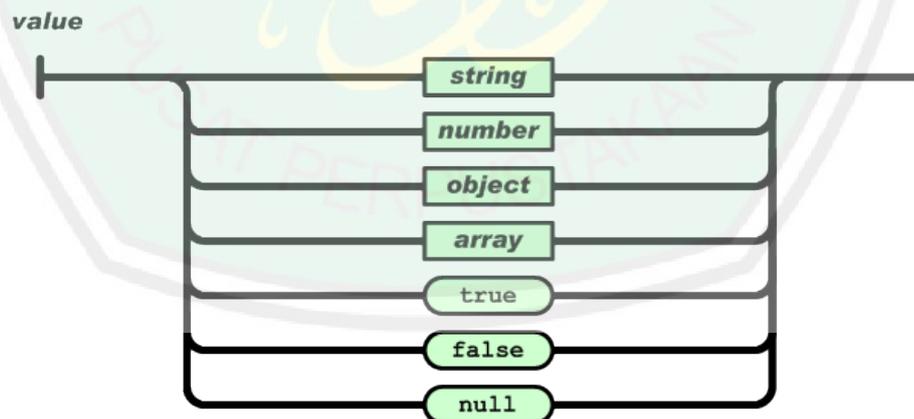
Gambar 2.6 Struktur data object JSON

- Array adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [ (kurung kotak buka) dan diakhiri dengan ] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).



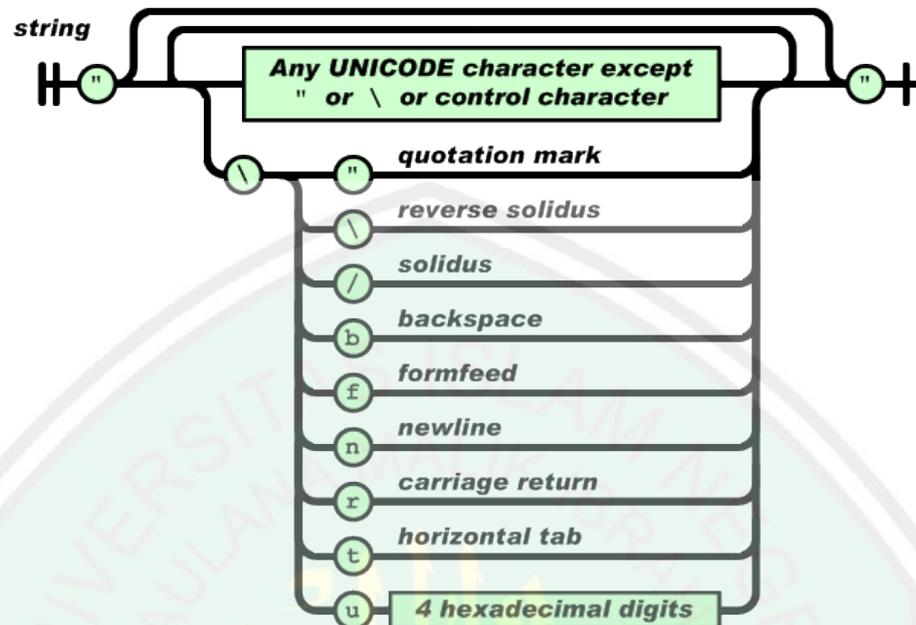
Gambar 2.7 Struktur data array JSON

- Nilai (value) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat.



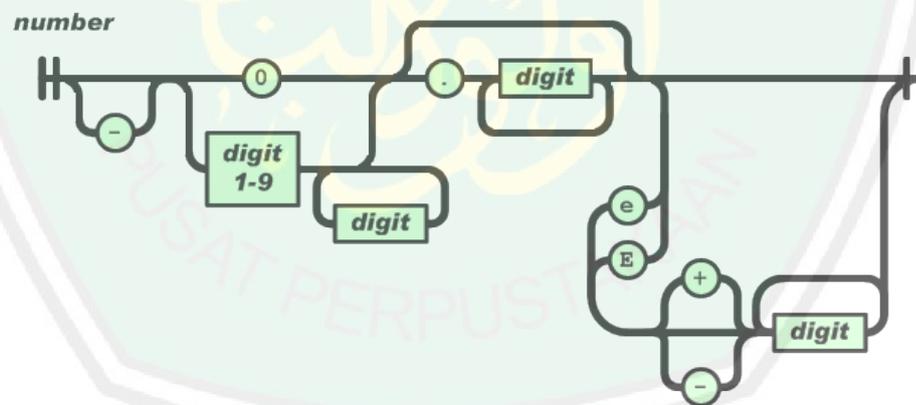
Gambar 2.8 Struktur data value JSON

- String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan backslash escapes "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.



Gambar 2.9 Struktur data string JSON

5. Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan.



Gambar 2.10 Struktur data angka JSON

6. Spasi kosong (whitespace) dapat disisipkan di antara pasangan tanda-tanda tersebut, kecuali beberapa detail encoding yang secara lengkap dipaparkan oleh bahasa pemrograman yang bersangkutan.

```
{
  "namaLokasi": "UIN",
  "lokasi": {
    "longitude": "112.9071331",
    "latitude": "-7.9759387"
  }
}
```

**Gambar 2.11 Contoh representasi JSON untuk suatu objek**

## 2.11. PHP

PHP adalah bahasa script di sisi server yang didesain secara khusus untuk jaringan. Dalam halaman HTML, bisa dimasukkan kode PHP yang akan dieksekusi tiap kali halaman di kunjungi. Kode PHP akan di terjemahkan di jaringan server dan mengenerate HTML atau output lain yang dilihat oleh pengunjung.

Menurut dokumen resmi PHP, PHP singkatan dari PHP Hypertext Preprocessor. PHP merupakan bahasa berbentuk skrip yang di tempatkan dalam server dan diproses di server. Hasilnya yang di kirimkan ke klien, tempat pemakai menggunakan browser.

Secara khusus, PHP di rancang untuk membentuk web dinamis. Artinya ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, anda dapat menampilkan isi database ke halaman Web. Pada prinsipnya, PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (Active Server Page), Cold Fusion ataupun Perl.

Kelahiran PHP bermula saat Ramus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi tool yang disebut "Personal Home Page". Paket inilah yang menjadi cikal-bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI Versi 2. Pada versi inilah pemrograman dapat menempelkan kode terstruktur di dalam tag HTML. Yang menarik, kode PHP juga bisa berkomunikasi dengan database dan melakukan perhitungan-perhitungan yang kompleks sambil jalan.

Pada saat ini, PHP cukup populer sebagai piranti pemrograman Web, terutama di lingkungan Linux. Walaupun demikian, PHP sebenarnya juga dapat berfungsi pada server-server yang berbasis UNIX, Windows NT, dan Macintosh. Bahkan versi untuk Windows 95/98 juga tersedia.

Pada awalnya, PHP di rancang untuk diintegrasikan dengan Web server Apache. Namun, belakangan ini PHP juga dapat bekerja dengan Web server seperti PWS (Personal Web Server), IIS (Internet Information Server), dan Xitami. Untuk mencoba PHP, anda tidak perlu menggunakan komputer berkelas server. Dengan hanya komputer biasa, anda bisa mempelajari dan mempraktekan PHP.

## 2.12. MySQL

MySQL adalah *database server relasional* yang gratis di bawah lisensi GNU General Public License. Dengan sifatnya yang *Open Source*, memungkinkan user juga untuk melakukan modifikasi pada *source code*-nya untuk memenuhi kebutuhan spesifik mereka sendiri. MySQL merupakan database server *multi - user* dan *multi – threaded* yang tangguh (robust). Dengan memiliki banyak feature MySQL bias bersaing dengan *database* komersial sekalipun. Tidak mengejutkan, MySQL menjadi *database* pilihan untuk banyak pengguna PHP.

MySQL dikembangkan oleh MySQL AB, sebuah perusahaan komersial yang membangun layanan bisnisnya melalui *database* MySQL. Awal mula pengembangan MySQL adalah penggunaan *mySQL* untuk koneksi ke table menggunakan rutin level rendah (ISAM) setelah beberapa pengujian diperoleh kesimpulan *mSQL* tidak cukup cepat dan fleksibel untuk memenuhi kebutuhan. Sehingga dihasilkan suatu antarmuka SQL baru pada *database* tetapi dengan API yang mirip *mSQL*. API ini dipilih sedemikian rupa sehingga memudahkan *porting* kode. Tentang penggunaan nama MySQL sampai saat ini belum jelas betul. *Base directory* dan sejumlah *library* serta tool pada saat pengembangan telah diawali dengan “my” selama sekitar 10 tahun. Bagaimanapun pemberian nama MySQL sampai saat ini masih menjadi sebuah “misteri”.

MySQL adalah suatu sistem manajemen *database*. Untuk menambahkan, mengakses, dan memproses data yang tersimpan pada suatu database computer anda memerlukan sistem manajemen *database* seperti MySQL. Karena computer sangat unggul dalam menangani sejumlah besar data, system manajemen database memainkan suatu peranan yang peting dalam komputansi, baik sebagai *utility stand-alone* maupun bagian dari aplikasi lainnya.

MySQL adalah system manajemen database relasional. Suatu database relasional menyimpan data dalam table-tabel terpisah. Hal ini memungkinkan kecepatan dan fleksibilitas. Table-tabel yang dihubungkan dengan relasi yang ditentukan membuatnya bias mengkombinasikan data dari beberapa table pada suatu permintaan. Bagian SQL dari kata MySQL berasal dari “*Structured Query Language*” bahasa paling umum yang dipergunakan untuk mengakses *database*. Konektivitas, kecepatan dan keamanannya membuat MySQL cocok untuk pengaksesan *database* pada internet. MySQL merupakan sistem *client/server* yang terdiri dari SQL server *multiheraded* yang memungkinkan *backed* yang berbeda, sejumlah program *client* dan *library* yang berbeda, *tool administrative*, dan beberapa antarmuka pemrograman. MySQL juga tersedia sebagai *library* yang bisa digabungkan ke aplikasi.

### 2.13. SQLite

SQLite itu merupakan sebuah Database yang bersifat ACID-compliant dan memiliki ukuran pustaka kode yang relatif kecil, ditulis dalam bahasa C. SQLite merupakan proyek yang bersifat public domain yang dikerjakan oleh D. Richard Hipp. SQLite adalah sebuah open source database yang telah ada cukup lama, cukup stabil, dan sangat terkenal pada perangkat kecil, termasuk Android. Android menyediakan database relasional yang ringan untuk setiap aplikasi menggunakan SQLite. Aplikasi dapat mengambil keuntungan dari itu untuk mengatur relational database engine untuk menyimpan data secara aman dan efisien. Untuk Android, SQLite dijadikan satu di dalam Android runtime, sehingga setiap aplikasi Android dapat membuat basis data SQLite. Karena

SQLite menggunakan antarmuka SQL, cukup mudah untuk digunakan orang-orang dengan pengalaman lain yang berbasis database. Terdapat beberapa alasan mengapa SQLite sangat cocok untuk pengembangan aplikasi Android, yaitu:

Database dengan konfigurasi nol. Artinya tidak ada konfigurasi database untuk para developer. Ini membuatnya relatif mudah digunakan. Tidak memiliki server. Tidak ada proses database SQLite yang berjalan. Pada dasarnya satu set libraries menyediakan fungsionalitas database. Single-file database. Ini membuat keamanan database secara langsung. Open source. Hal ini membuat developer mudah dalam pengembangan aplikasi.

### **2.13.1. Fitur-fitur SQLite**

Pustaka SQLite mengimplementasikan hampir seluruh elemen-elemen standar yang berlaku pada SQL-92, termasuk transaksi yang bersifat atomic, konsistensi basisdata, isolasi, dan durabilitas (dalam bahasa Inggris lebih sering disebut ACID), trigger, dan kueri-kueri yang kompleks. Tidak ada pengecekan tipe sehingga data bisa dientrikan dalam bentuk string untuk sebuah kolom bertipe integer. Beberapa kalangan melihat hal ini sebagai sebuah inovasi yang menambah nilai guna dari sebuah basisdata, utamanya ketika digunakan dalam bahasa pemrograman berbasis script (PHP, Perl), sementara kalangan lain melihat hal tersebut sebagai sebuah kekurangan.

Beberapa proses ataupun thread dapat berjalan secara bersamaan dan mengakses basisdata yang sama tanpa mengalami masalah. Hal ini disebabkan karena akses baca data dilakukan secara paralel. Sementara itu akses tulis data hanya bisa dilakukan jika tidak ada proses tulis lain yang sedang dilakukan; jika tidak, proses tulis tersebut akan gagal dan mengembalikan kode kesalahan (atau bisa juga secara otomatis akan mencobanya kembali sampai sejumlah nilai waktu yang ditentukan habis). Hanya saja ketika sebuah tabel temporer dibuat, mekanisme penguncian pada proses multithread akan menyebabkan masalah. Update yang terkini (versi 3.3.4) dikatakan telah memperbaiki masalah ini.

Sebuah program yang mandiri dinamakan `sqlite` disediakan dan bisa digunakan untuk mengeksekusi kueri dan manajemen file-file basisdata `SQLite`. Program tersebut juga merupakan contoh implementasi penulisan aplikasi yang menggunakan pustaka `SQLite`

### 2.13.2. Integrasi dengan bahasa lain

- A. `SQLite` termasuk dalam framework `REALbasic`, yang memungkinkan aplikasi yang dikembangkan dengan menggunakan `REALbasic` dapat memanfaatkan basisdata `SQLite`.
- B. Pustaka `SQLite` bisa digunakan secara langsung pada bahasa `C/C++`, namun untuk `Tcl` dan beberapa bahasa pemrograman berbasis script juga tersedia.
- C. Modul `DBI/DBD` untuk `Perl` juga tersedia pada `CPAN`, `DBD:SQLite`, namun modul ini bukanlah antarmuka dengan `SQLite` melainkan memasukkan `SQLite` secara keseluruhan dalam modul tersebut.
- D. Modul `Python` juga tersedia (`PySQLite`) yang diimplementasikan pada `DB API Python` versi 2.0 (PEP 249)
- E. `PHP` dimulai dengan `PHP5` telah memasukkan `SQLite`, versi `PHP4` sebelumnya bisa juga digunakan untuk mengakses `SQLite`, namun modul `SQLite` tidak dimasukan secara standar bawaan.
- F. Dimulai `Lazarus` versi 0.9.8 dan `Free Pascal` 2.0.0, `SQLite` didukung untuk digunakan oleh para programmer.
- G. Meskipun `borland` tidak memaketkan `SQLite` secara standar bawaan, `Delphi` sudah mendukung `SQLite` juga menggunakan pustaka yang dibuat oleh pihak ketiga (`Aducom` dan `Zeos`). Banyak program administrasi basisdata `SQLite` baik komersil ataupun bebas yang menawarkan kemudahan-kemudahan dalam manajemen basisdata `SQLite` terdapat di internet dibuat dengan menggunakan bahasa ini.
- H. `SQLite` juga termasuk dalam paket yang dibundle secara standar bawaan pada `Mac OS X`, dan digunakan sebagai salah satu pilihan mekanisme penyimpanan data pada `API Apple`.

## 2.14. XML

XML (eXtensible Markup Language) merupakan bahasa web turunan dari SGML (Standart Generalized Markup Language) yang ada sebelumnya. XML hampir sama dengan HTML, dimana keduanya sama-sama turunan dari SGML. Teknologi XML dikembangkan mulai tahun 1966 dan mendapatkan pengakuan dari Worl Wide Web Consortium (W3C) pada bulan Februari 1998. Sedangkan SGML sendiri telah dikembangkan pada awal tahun 1980-an. Pada saat HTML dikembangkan pada tahun 1990, para penggagas XML mengadopsi bagian paling penting SGML dan dengan berpedoman pada pengembangan HTML menghasilkan bahasa markup yang tidak kalah hebatnya dengan SGML.

Secara sederhana XML adalah suatu bahasa yang digunakan untuk mendeskripsikan dan memanipulasi dokumen secara terstruktur. Secara teknis XML didefinisikan sebagai suatu bahasa meta-markup yang menyediakan format tertentu untuk dokumen-dokumen yang mempunyai data terstruktur. Bahasa markup adalah mekanisme untuk mengenal secara terstruktur di dokumen. XML adalah suatu aplikasi profil dari SGML. Seperti yang didefinisikan dari ISO 8879, SGML adalah cara standart dan vendor-independent.

XML tidak mempunyai definisi secara tepat karena ada yang berpendapat bahwa XML bukanlah suatu bahasa pemrograman, melainkan XML merupakan sintaks yang digunakan untuk menjelaskan bahasa markup lain (Dournaee,2002), sehingga dinamakan meta-language. Meskipun demikian pendapat yang XML bukan merupakan bahasa markup, didasarkan bahwa XML merupakan bahasa markup terpisah untuk tujuan terpisah. Selain itu XML bukanlah solusi semua hal untuk tujuan semua user. Sedangkan peran dari markup itu sendiri berupa:

1. Markup dapat menambah maksud arti (semantic) suatu data.
2. Dapat memisahkan data.
3. Dapat mendefinisikan peran data.
4. Dapat mendefinisikan batasan data.

#### 5. Dapat menfenisikan keterhubungan data.

XML merupakan sebuah himpunan bagian (subset) dari SGML yang bertujuan agar SGML secara generik dapat melayani, menerima, dan memproses di dalam web dengan cara seperti yang dimungkinkan HTML saat ini. XML didesain untuk kemudahan implementasi dan interoperabilitas dengan SGML maupun HTML. XML adalah bahasa markup yang dirancang untuk penyampaian informasi melalui World Wide Web (www) atau sering disebut web saja.

XML merubah cara kita berpikir untuk mengembangkan suatu software terutama aplikasi web. Masalah yang kita sekarang adalah bagaimana caranya untuk bertukar informasi antar satu aplikasi dengan aplikasi lain. Kadang kolaborasi antara satu aplikasi dengan aplikasi yang lain masih harus ditentukan dengan spesifikasi aplikasi tersebut. Padahal seharusnya kita hanya perlu mendapatkan informasi data bukan mengerti cara kerja aplikasi lain itu, disinilah visi internet yang belum tercapai. Visi ini adalah dunia internet dimana PC, server, smart devices dan internet-based device dapat berkolaborasi tanpa ada halangan. Bisnis-bisnis akan dapat bertukar data menyediakan customized dan comprehensive solusi kepada customer. Dan yang paling utama adalah informasi yang dibutuhkan dapat diakses dari mana saja dan dengan computing device, platform, atau aplikasi yang kita gunakan.

XML dapat memungkinkan pertukaran informasi atau data antar device (server, PCs, smart device, aplikasi, dan situs web). Data ini akan menjadi independent (unlocked), memudahkannya untuk diorganisir, diprogram, dan dirubah, dan ditukar antar situs web atau aplikasi apa saja. Karena kebutuhan ini, maka makin banyak teknologi berbasis XML yang keluar. Contohnya adalah SOAP (Simple Project Acces Protocol) dan UDDI (Universal Description Discovery and Integration). SOAP memungkinkan aplikasi-aplikasi untuk berkomunikasi melalui standart internet protocol. Sedangkan UDDI memberikan bisnis-bisnis suatu cara standart untuk menjelaskan servis mereka dan cara koneksi secara otomatis. Dari sini munculah web service, yaitu suatu tipe baru

software yang dapat dipakai berulang-ulang (seperti komponen software) dengan pengecualian service ini dapat diakses darimana saja melalui internet.



### BAB III ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas mengenai perancangan sistem pembuatan aplikasi otomatisasi presensi menggunakan *Global Positioning System*(GPS). Pada bab ini akan dibahas meliputi deskripsi sistem, perancangan sistem, perancangan database, perancangan *flowchart*, dan perancangan interface.

#### 3.1. Deskripsi Sistem

Aplikasi ini merupakan alat untuk melakukan pelacakan user menggunakan perangkat android secara otomatis. Pelacakan dilakukan menggunakan *Location Based Service* dengan mengambil titik koordinat lokasi perangkat user melalui GPS. Untuk mendapatkan titik koordinat area Universitas sistem menggunakan Google Maps API dengan memberikan parameter koordinat Universitas.

Dalam penentuan lokasi user yang telah melakukan *check-in* atau *check-out*, sistem membedakan area Universitas dengan area Non-Universitas. Area Universitas adalah seluruh area yang berada dalam lingkungan Universitas, area non-Universitas adalah area yang berada diluar lingkungan Universitas. Rumusan yang digunakan untuk membedakan lokasi tersebut adalah:

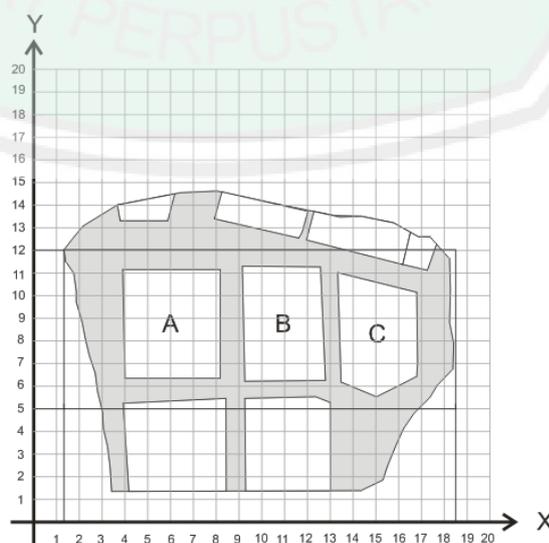
$$(Ax \leq Px \leq Bx) \text{ AND } (Cx \geq Px \geq Dx)$$

$$(Ay \leq Py \leq By) \text{ AND } (Cy \geq Py \geq Dy)$$

Ketika lokasi perangkat user( $P_{x,y}$ ) ada di area yang lebih dari titik A dan kurang dari titik B dan di area yang lebih dari D dan kurang dari C maka perangkat user ada di dalam area Universitas dan sistem akan secara otomatis mendeteksinya sebagai perangkat yang *check-in*. Sebaliknya apabila perangkat user tidak dalam kondisi tersebut maka perangkat user ada di luar Universitas sistem akan secara otomatis mendeteksinya sebagai perangkat yang *check-out*. Dalam area Universitas sendiri dibagi dalam beberapa region sesuai dengan gedung yang terdapat dalam Universitas. Pembagian region ini bertujuan untuk

pemantauan aktivitas perangkat user didalam Universitas dan dalam pergedung di dalam Universitas. Misal, apabila ada aktivitas user berada di gedung A maka sistem akan otomatis melakukan pencatatan terhadap aktivitas tersebut sesuai dengan tempat user berada. Skema *tracking* ini berjalan dalam *range* minimal 1 menit dan dapat diatur sesuai kebutuhan pengguna seperti 5 menit, 30 menit, dan 60 menit sehingga informasi spasial pengguna bisa terdeteksi oleh sistem secara otomatis dan *real-time* dengan nilai akurasi jarak yang bisa diketahui. Kemudian data lokasi pengguna yang telah di-*tracking* akan terunggah ke web sever online dan secara langsung bisa dipantau oleh admin. Sehingga *user* tidak memiliki kesempatan untuk melakukan kecurangan dalam pencatatan presensi berupa data lokasi yang berjalan *real-time*.

Sistem ini juga bisa mendeteksi ketika perangkat suatu user aktif secara bersamaan melakukan aktivitas yang sama dalam waktu yang sama maka sistem akan menduga user telah melakukan kecurangan dengan menipikan perangkat pada user lain. Nilai ini akan didapat ketika nilai *similarity* dihitung menurut aktivitas dan waktu yang dilakukan secara bersamaan. Sistem juga bisa menganalisis aktivitas user ketika user berada pada kondisi *check-out* dalam waktu tertentu. Sehingga user bisa terdeteksi apabila user meninggalkan area Universitas pada jam kerja user yang dicurigai tidak kembali ke are Universitas.



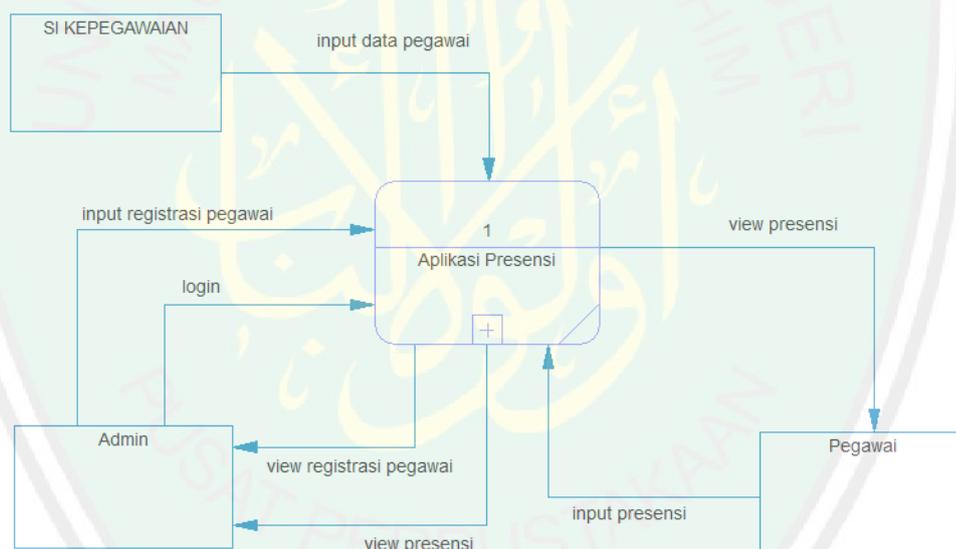
**Gambar 3.1** Koordinat area Universitas

## 3.2 Perancangan Sistem

### 3.2.1. Diagram Konteks

Diagram konteks menghubungkan antara aplikasi dengan entitas. Selain itu, diagram konteks merupakan diagram paling awal yang menjelaskan sistem secara umum. Aliran data dalam diagram konteks memodelkan masukan ke dalam sistem dan keluaran dari sistem.

Gambar 3.2 menjelaskan mengenai arus aplikasi presensi untuk mendapatkan data lokasi pengguna yang melibatkan 1 entitas, yaitu pegawai. Pegawai merupakan pengguna aplikasi yang bisa melihat data presensi dan melihat jarak lokasi dengan kampus ketika pegawai berada diluar kampus.

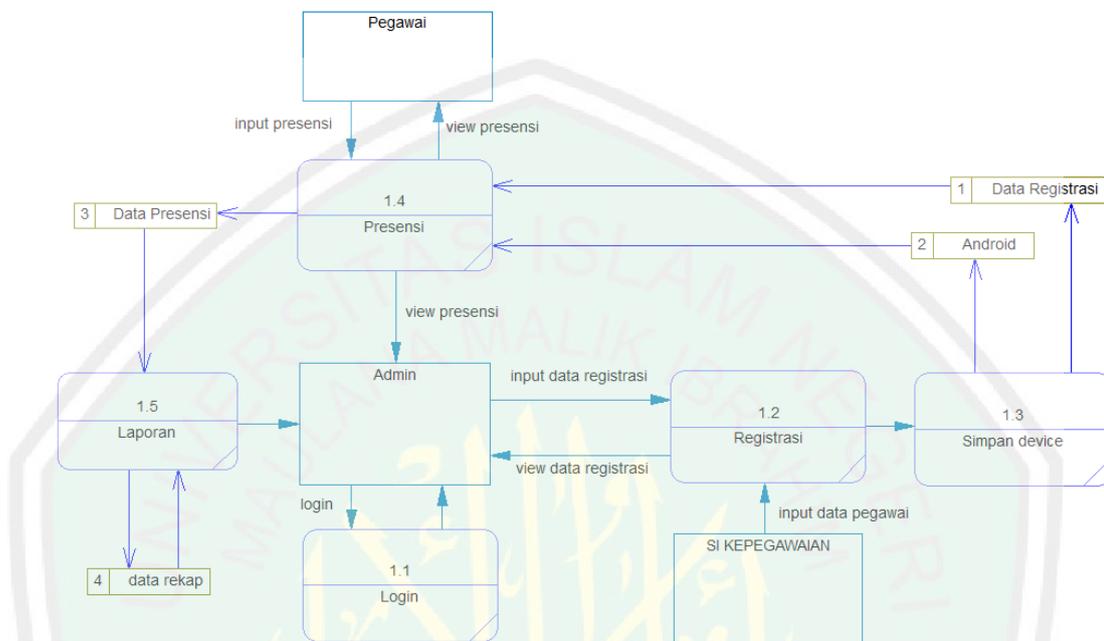


Gambar 3.2. Context Diagram aplikasi presensi

### 3.2.2. Data Flow Diagram level 1

Gambar 3.3 menjelaskan bagian proses yang lebih rinci dari proses secara umum yang ada pada diagram konteks sebelumnya. Terdapat dua proses pada data flow diagram level 1 diatas, antara lain: view data presensi dan view jarak lokasi. View data presensi untuk melihat data presensi yang telah diinputkan oleh user selama kurun waktu sesuai interval yang telah diatur. Sedangkan view jarak lokasi adalah melihat jarak lokasi user dengan lokasi kampus berada, sesuai dengan jarak

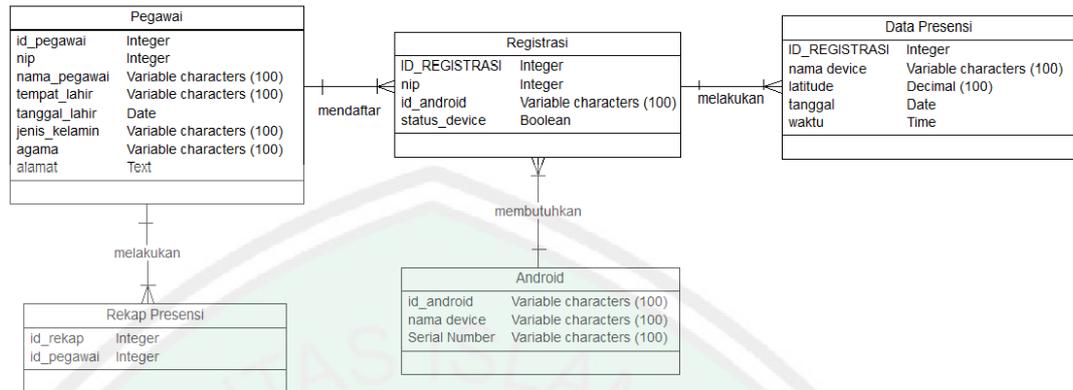
sebenarnya antara dua titik. Yang dihitung menggunakan algoritma jarak *euclidean*.



Gambar 3.3. Data Flow Diagram level 1

### 3.3. Perancangan Database

Kebutuhan tabel database pada aplikasi terdiri dari 5 tabel, yakni tabel registrasi, tabel android, tabel presensi, tabel rekap presensi, dan tabel pegawai. Tabel pegawai memiliki 8 field: id\_pegawai, nip, nama pegawai, tempat lahir, tanggal lahir, jenis kelamin, agama, dan alamat. Tabel registrasi memiliki 4 field: id registrasi, nip, id android, status device. Tabel presensi memiliki 5 field: id\_registrasi, nama device, id\_android, longitude, latitude, tanggal, waktu. Dan tabel rekap memiliki 2 field: id\_rekap, dan id\_pegawai. Sedangkan pada tabel android memiliki 3 field: id\_android, nama device, serial number.



Gambar 3.4. Entity Relational Database aplikasi presensi

### 3.3.1. Tabel Presensi

Tabel user berfungsi untuk menyimpan data lokasi user berupa: id, id android, nama lokasi, hari, tanggal, jam, longitude, dan latitude

Tabel 3.1 Tabel Presensi

Field	Tipe data
Id_registrasi	int(5)
Nama device	varchar (100)
Longitude	varchar (100)
tanggal	Date
Waktu	Time
Nama lokasi	varchar (100)

### 3.3.2. Tabel Registrasi

Tabel registrasi adalah tabel yang berfungsi menyimpan data registrasi pegawai. Susunan *field*-nya sebagai berikut:

Tabel 3.2 Tabel Registrasi

Field	Tipe data
Id_registrasi	Integer

Field	Tipe data
Nip	varchar (100)
id_android	Date
status_device	Date

### 3.3.3. Tabel Pegawai

Tabel pegawai adalah tabel yang berfungsi menyimpan data pegawai. Susunan *field*-nya sebagai berikut:

Tabel 3.3 Tabel Pegawai

Field	Tipe data
id_pegawai	Integer
Nip	varchar (100)
Nama_pegawai	varchar (100)
Tempat lahir	varchar (100)
Tanggal lahir	Date
Jenis kelamin	varchar (100)
Agama	varchar (100)
Alamat	Text

### 3.3.4. Tabel Rekap Presensi

Tabel rekap presensi digunakan untuk menyimpan data rekap pegawai yang telah melakukan presensi per periode. Susunan *field*-nya sebagai berikut:

Tabel 3.4 Tabel Rekap Presensi

Field	Tipe data
Id_rekap	int(5)
Id_pegawai	varchar (100)

### 3.3.5. Tabel Android

Tabel android berfungsi untuk menyimpan data perangkat pegawai. *Field* pada tabel admin berupa: id, username, password

Tabel 3.5 Tabel Android

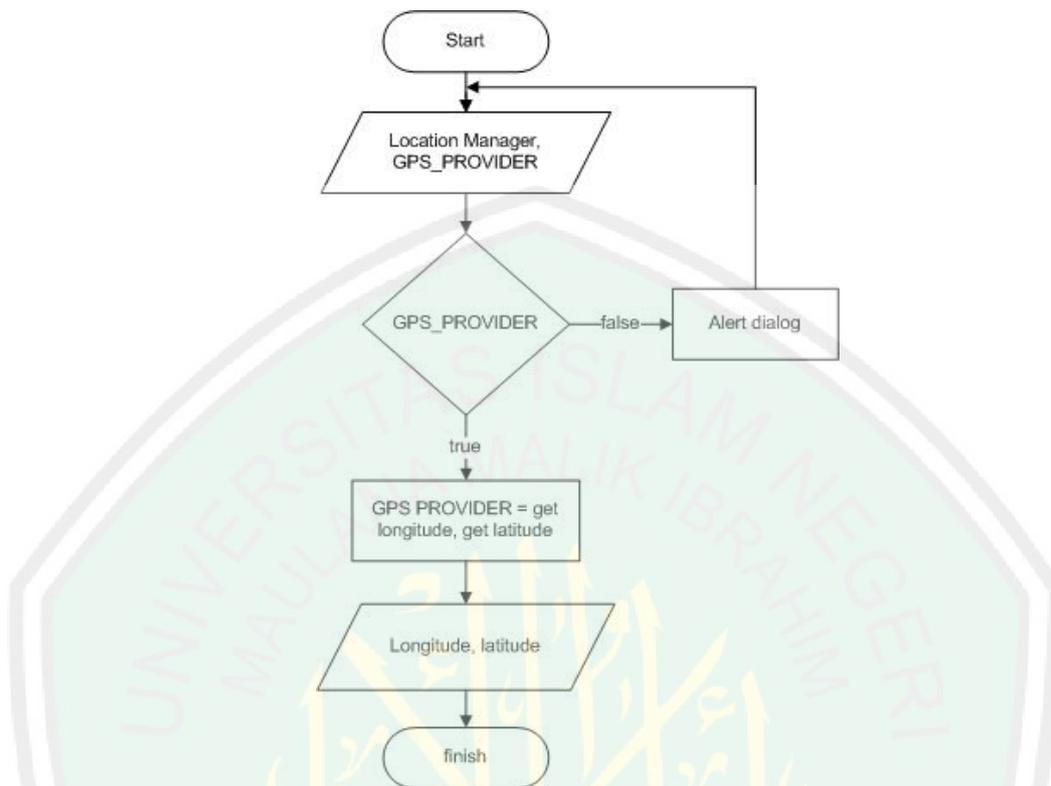
Field	Tipe data
Id	int(5)
Nama device	varchar (100)
Serial number	varchar (100)

### 3.4. Perancangan Flowchart

Perancangan flowchart pada aplikasi ini memiliki empat bagian, yakni: flowchart proses pengambilan lokasi, flowchart penentuan nama lokasi, flowchart proses sinkronisasi ke web server.

#### 3.4.1. Flowchart Proses Pengambilan Lokasi

Flowchart pada gambar 3.5 merupakan alur proses pengambilan lokasi melalui *GPS PROVIDER*. Untuk memanggil *GPS PROVIDER* dibutuhkan perintah untuk meng-*import* kelas Location Manager. Setelah *GPS PROVIDER* tersedia maka aplikasi bisa mengambil data lokasi berbasis GPS dengan nilai output *longitude* dan *latitude*. Nilai *longitude* dan *latitude* inilah yang akan dihitung untuk menentukan nilai nama area lokasi dan data *longitude* dan *latitude* ini tidak ditampilkan (*hidden*) pada *Textview* pada halaman awal aplikasi. Namun nilai *longitude* dan *latitude*-nya masih tersimpan di database. Nama lokasi yang ditampilkan di *Textview* adalah nama lokasi yang telah diproses pada penghitungan penentuan area lokasi.



Gambar 3.5. Flowchart proses pengambilan lokasi

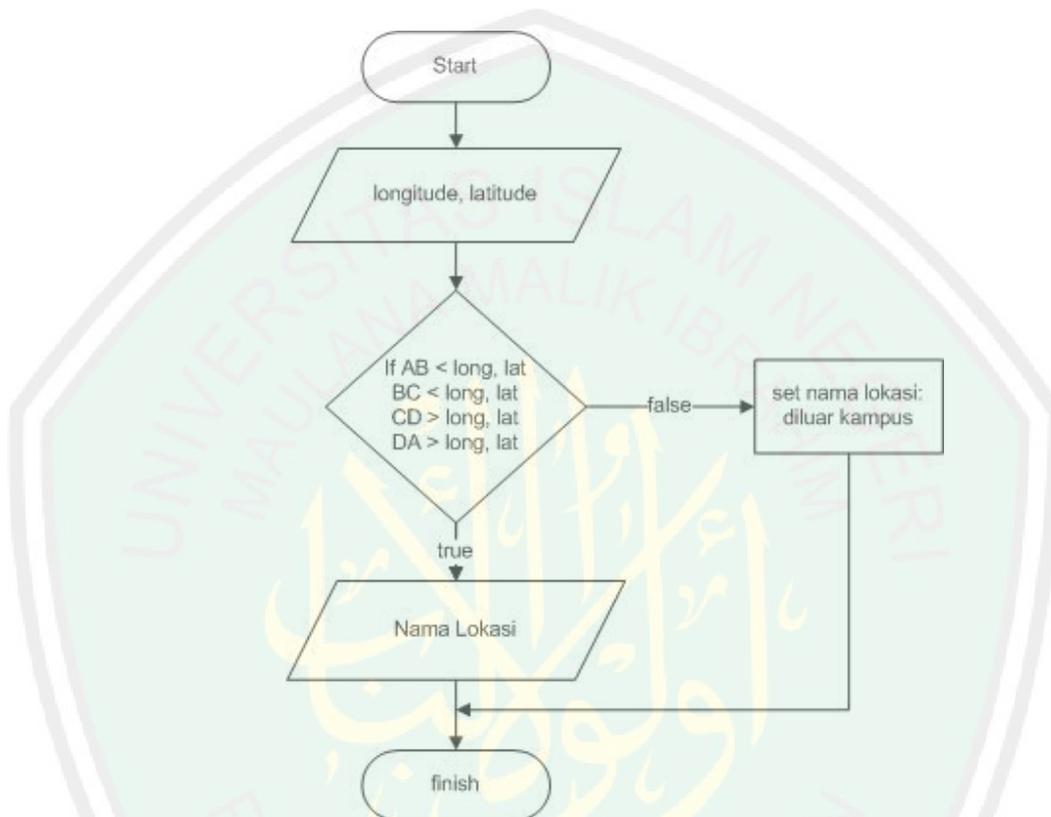
### 3.4.2. Flowchart Proses Penentuan Area Lokasi

Flowchart pada gambar 3.6 merupakan penjelasan proses untuk menentukan area lokasi. Dengan nilai input adalah *longitude* dan *latitude* yang telah dijabarkan sebelumnya. Untuk menentukan area lokasi maka *longitude* dan *latitude* harus memenuhi empat kondisi ini, yakni:

- A. Garis AB: if  $Ax \leq longitude \ \&\& \ Bx \leq longitude$
- B. Garis BC: if  $Cx \geq longitude \ \&\& \ Dx \geq longitude$
- C. Garis CD: if  $Ay \leq latitude \ \&\& \ Dy \leq latitude$
- D. Garis DA: if  $By \geq latitude \ \&\& \ Cy \geq latitude$

Yakni *longitude* dan *latitude* berada dalam suatu area gedung dengan nilai koordinat titik yang telah diketahui sebelumnya. Keempat kondisi ini juga berlaku untuk masing-masing gedung atau tempat untuk memudahkan aplikasi melakukan *tracking*. Dalam *tracking* lokasi area kampus akan menampilkan data-data area lokasi, seperti: gedung saintek, gedung perpustakaan, gedung b perkuliahan,

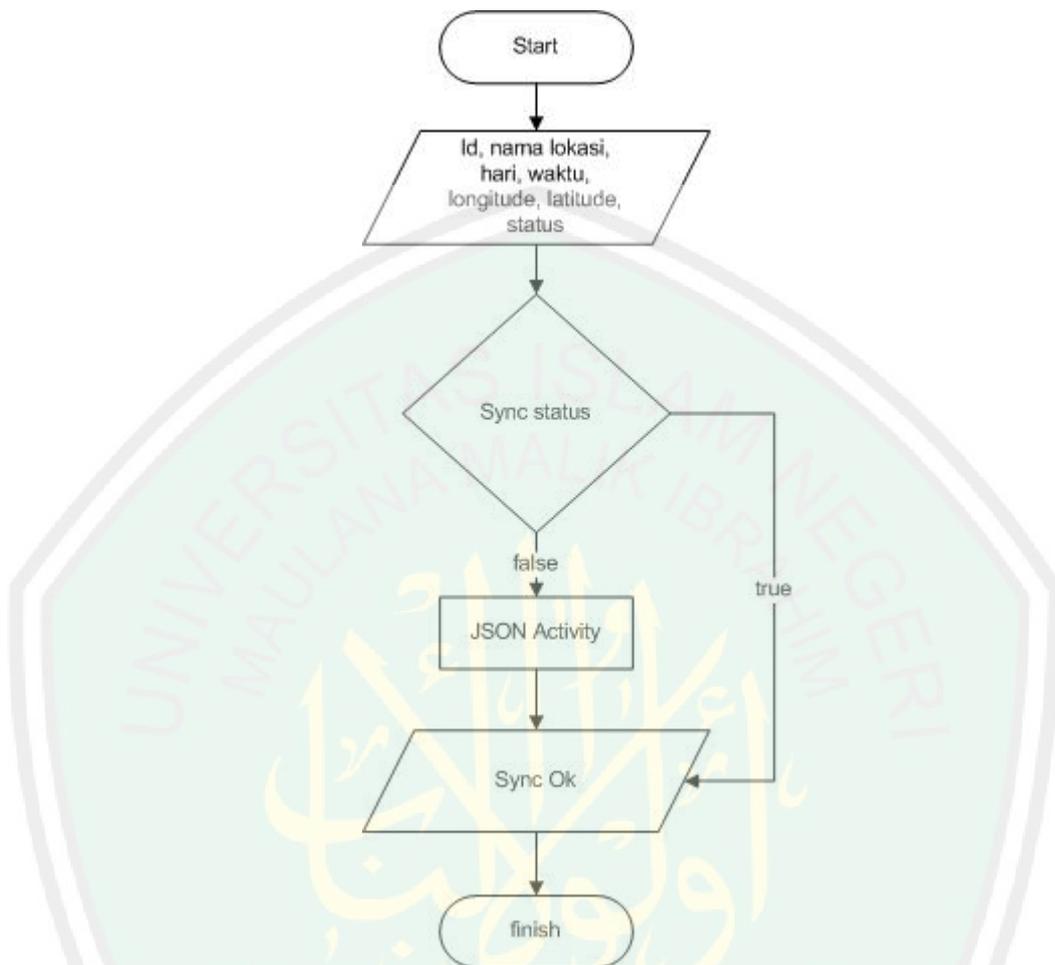
gedung c, dan lain sebagainya. Namun apabila penghitungan nilai *longitude* dan *latitude* tidak memenuhi keempat kondisi tersebut maka secara *default* lokasi akan berstatus DILUAR KAMPUS.



Gambar 3.6. Flowchart proses penentuan area lokasi

### 3.4.3. Flowchart Proses Sinkronisasi Webservice

Flowchart pada gambar 3.7 menjelaskan mengenai proses sinkronisasi data lokasi ke web server. Dibutuhkan pendeklarasian database sqlite berupa data lokasi yang meliputi id, id android, nama lokasi, longitude, latitude, hari, waktu, dan status sebagai data input. Pendeklarasian tabel status berguna untuk mengetahui status data sqlite pada aplikasi yang di proses dalam kelas *DBHelper*. Ketika data status bernilai yes, maka data lokasi sudah disinkronkan dengan web server. Namun ketika status bernilai no maka aplikasi akan meneruskan ke kelas JSON Activity yang akan memproses data lokasi untuk disinkronkan ke web server. Kemudian *DBHelper* akan meng-*update* tabel status sehingga bernilai *no*. Kemudian kelas *DBHelper* akan memunculkan notifikasi yang menunjukkan bahwa sinkronisasi berhasil.



Gambar 3.7. Flowchart proses sinkronisasi web server

### 3.5. DESAIN INTERFACE

Perencanaan desain *interface* pada aplikasi ini memiliki 2 bagian, yakni: *interface* aplikasi android dan *interface* web. Tampilan *interface* android berupa tampilan yang ada pada aplikasi android. Sedangkan tampilan *interface* web berupa tampilan *interface* di *web service*.

#### 3.5.1 TAMPILAN APLIKASI ANDROID

##### 3.5.1.1 TAMPILAN HALAMAN UTAMA

Gambar 3.6 merupakan halaman utama aplikasi yang berisi data lokasi, tanggal, dan jam pengguna terkini. Sehingga aplikasi selalu terupdate setiap pengguna membuka aplikasi. Dibawah juga terdapat tampilan history yang disimpan pada database lokal yang belum terupload ke webserver. Pada bagian ini

pengguna akan diminta untuk melakukan sinkronisasi history ke web server ketika data lokasi belum tersinkronisasi dengan menggunakan menu pada bagian bawah aplikasi. Pada menu tersebut juga terdapat tombol untuk mengukur jarak lokasi pengguna dengan kampus ketika pengguna berada diluar area kampus. Data lokasi yang tersinkron dengan webserver sudah termasuk data id android pada perangkat android pegawai yang bersangkutan. ID android digunakan untuk mengidentifikasi pegawai yang bersangkutan karena ID android sifatnya *unique*.



**Gambar 3. 8. Tampilan halaman utama aplikasi**

### 3.5.1.2. TAMPILAN HALAMAN HISTORY

Gambar 3.7 merupakan halaman *history* yang berisi seluruh data aktivitas *user* yang sudah tersinkronisasi ke *web server*. Sehingga untuk mendapatkan data tersebut *user* harus terhubung ke internet. Data *history* tersebut meliputi data lokasi, hari, dan tanggal.



Gambar 3. 9. Tampilan halaman history

### 3.5.2 Tampilan Web

#### 3.5.2.1. Tampilan halaman login

Admin merupakan pemegang hak akses penuh aplikasi ini, terutama pada sisi web server. Karena fungsi administrator adalah untuk melakukan manajemen data user yang terinput ke *web server*.

Gambar 3.8 merupakan form halaman login untuk admin. Form ini merupakan penamanan agar tidak semua orang bisa masuk ke dalam administrator. Untuk masuk ke dalam halaman admin, admin harus memasukkan *username* dan *password*.



Gambar 3. 10. Tampilan halaman login web

### 3.5.2.2. Tampilan halaman Daftar Lokasi

Gambar 3.9 merupakan halaman admin yang menampilkan data lokasi yang telah diinputkan oleh pegawai melalui aplikasi secara otomatis berdasarkan interval waktu yang telah diatur. Data ini berupa ID Android, Nama Lokasi, Longitude, Latitude, Hari, Jam. Pada halaman ini admin bisa membuat laporan harian, bulanan, dan tahunan dari aktivitas pegawai.

ID	ID Android	Nama lokasi	Longitude	Latitude	Hari	Jam

Gambar 3. 11. Tampilan halaman daftar lokasi

### 3.5.3 Penghitungan Area Lokasi

Penghitungan area lokasi dilakukan dengan mengetahui nilai *longitude* dan *latitude user*, kemudian dilakukan proses AND antara nilai lokasi *user* dengan nilai lokasi kampus yang telah dideklarasikan.

Inputan berupa *longitude* dan *latitude user*: 3,2

Nilai Titik A(x, y) = ( 1, 1)

Nilai Titik B(x, y) = ( 1, 5)

Nilai Titik C(x, y) = ( 5, 1)

Nilai Titik D(x, y) = ( 5, 5)

*User* akan dianggap berada pada area lokasi jika berada didalam area dengan memenuhi kondisi:

Kondisi 1, garis AB: if  $A_x \leq longitude \ \&\& \ B_x \leq longitude$

Kondisi 2, garis BC: if  $C_x \geq longitude \ \&\& \ D_x \geq longitude$

Kondisi 3, garis CD: if  $A_y \leq latitude \ \&\& \ D_y \leq latitude$

Kondisi 4, garis DA: if  $B_y \geq latitude \ \&\& \ C_y \geq latitude$

Setelah semua kondisi terpenuhi dan bernilai benar maka *user* dianggap berada pada area lokasi tersebut.

### 3.5.4 Penghitungan Jarak Kampus

Penghitungan jarak menggunakan Jarak Euclidean yang menghasilkan nilai jarak dua titik. Dengan rumus:

$$D = \sqrt{x_p^2 - x_k^2 + y_p^2 - y_k^2} * 111.319$$

Keterangan :  $D = \text{Jarak}$

$x_p = \text{longitude user}$

$x_k = \text{longitude kampus}$

$y_p = \text{latitude user}$

$y_k = \text{latitude kampus}$

Nilai 111.319 adalah nilai satu derajat longitude pada garis equator. Sehingga nilai longitude dan latitude bisa diketahui jarak sebenarnya.

Inputan berupa *longitude* dan *latitude user*: 112.57487, -7.938198333

Nilai *longitude* dan *latitude* kampus: 112.607991, -7.951112

Maka hasil penghitungannya:

$$D = \sqrt{(112.57487 - 112.607991)^2 + (-7.938198 - (-7.951112))^2} * 111.319$$

$$D = \sqrt{0.0010969344 + 0.000166771396} * 111.319$$

$$D = 0.03554863986146305429416589037582 * 111.319$$

$$D = 3.9572390407382057409722527507459$$



## BAB IV IMPLEMENTASI DAN HASIL UJI COBA

### 4.1 Implementasi

Bab ini membahas mengenai implementasi dan dari sistem presensi yang telah dibangun dan uji coba terhadap aplikasi presensi. Tujuan dilakukannya pengujian adalah untuk mengetahui nilai keakuratan sinyal gps pada perangkat android dan dapat berjalan baik pada aplikasi tersebut. Selain itu juga untuk mengetahui apakah aplikasi yang dibangun dapat berjalan sesuai dengan apa yang sudah direncanakan.

#### 4.1.1 Implementasi Algoritma

Pada bagian ini membahas implementasi algoritma pada masing-masing bagian aplikasi. Implementasi ini terdiri dari beberapa bagian, yakni:

##### 4.1.1.1 Tahapan Akuisisi Data Lokasi

Pengambilan data lokasi merupakan tahap awal sebelum dilakukannya proses penghitungan area lokasi terhadap longitude dan latitude area lokasi. Pertama yaitu mengambil data *longitude* dan *latitude* dengan menggunakan GPS yang sudah tertanam pada perangkat android.

Gambar 4.1 merupakan *pseudo code* proses pengambilan data lokasi melalui GPS. Untuk bisa mengambil data lokasi berupa longitude dan latitude dibutuhkan pemanggilan kelas Location Manager yang sudah tertanam di perangkat android. Kelas Location Manager sendiri merupakan kelas yang menyediakan akses *location service*, yang memungkinkan aplikasi untuk memperoleh pembaruan lokasi perangkat secara berkala. Setelah kelas Location Manager dipanggil maka aplikasi memeriksa apakah kondisi GPS\_PROVIDER pada perangkat tersedia. Apabila GPS tidak tersedia maka aplikasi memanggil kelas alert dialog yang memberikan peringatan untuk *user* menghidupkan GPS\_PROVIDER. Ketika GPS\_PROVIDER sudah tersedia maka kelas Location Manager bisa melakukan pengambilan data lokasi berupa *longitude* dan *latitude*.

```

Deklarasi:
LocationManager, boolean GPS_PROVIDER
    if(GPS_PROVIDER) then
        if (getLocation) then
            location = getLongitude
            location = getLatitude
        endif
    else alert dialog
    endif
Write longitude, latitude

```

Gambar 4.1 Pseudo code pengambilan longitude dan latitude

#### 4.1.1.2 Tahapan Penentuan Manual Area Lokasi

Penghitungan area lokasi dilakukan dengan mengetahui nilai *longitude* dan *latitude user*, kemudian dilakukan proses AND antara nilai lokasi *user* dengan nilai lokasi kampus yang telah dideklarasikan.

Proses penentuan area lokasi dengan menggunakan parameter longitude dan latitude *user* yang dibandingkan dengan nilai longitude dan latitude area yang telah ditetapkan sebelumnya. Perbandingan ini menggunakan nilai AND dengan satu output bernilai benar. Area lokasi yang digunakan sebagai sample adalah lokasi gedung Saintek UIN MALIKI. Dengan nilai longitude dan latitude terdapat pada bagian masing-masing pojok gedung Saintek UIN MALIKI.

Titik pertama disebut titik A gedung berada pada lokasi pojok sebelah barat selatan dengan nilai longitude dan latitude: 112.606783, -7.951041. Sedangkan untuk titik kedua disebut titik B berada di pojok utara sebelah barat dengan nilai longitude dan latitude: 112.606760, -7.950468. Sedangkan untuk titik ketiga disebut titik C berada di pojok utara sebelah timur dengan nilai longitude dan latitude: 112.607308, -7.950450. Sedangkan untuk titik ketiga disebut titik D berada di pojok selatan sebelah timur dengan nilai longitude dan latitude: 112.607326, -7.951022. Untuk lebih mudah memahami bisa memahami area lokasi gedung Saintek UIN MALIKI bisa dilihat pada gambar 4.2

Penghitungan nilai lokasi dapat dilakukan dengan menggunakan inputan longitude dan latitude pada perangkat yang digunakan user. Pada kasus ini inputan berupa *longitude* dan *latitude user*: 112.607019, -7.950703

Dengan nilai pada masing-masing titik gedung sebagai berikut:

Nilai Titik A(x, y) = (112.606783, -7.951041)

Nilai Titik B(x, y) = (112.606760, -7.950468)

Nilai Titik C(x, y) = (112.607308, -7.950450)

Nilai Titik D(x, y) = (112.607326, -7.951022)

Pada posisi lokasi titik koordinat user. *User* akan dianggap berada pada area lokasi jika berada didalam area dengan memenuhi kondisi:

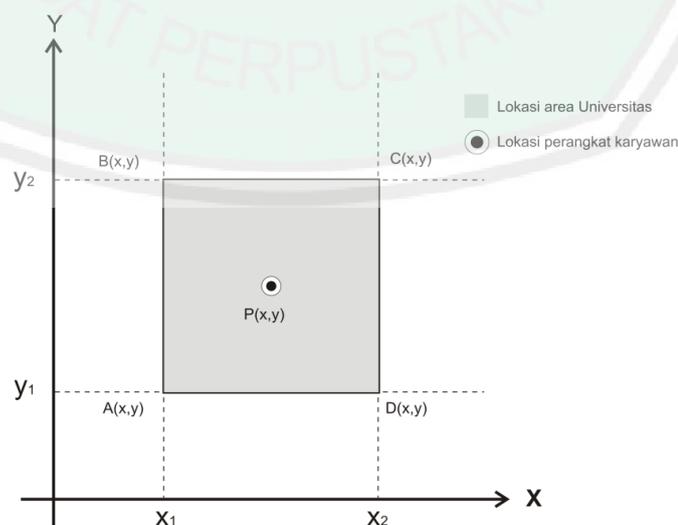
Kondisi 1, garis AB: if  $A_x \leq \text{longitude}$  &&  $B_x \leq \text{longitude}$

Kondisi 2, garis BC: if  $C_x \geq \text{longitude}$  &&  $D_x \geq \text{longitude}$

Kondisi 3, garis CD: if  $A_y \leq \text{latitude}$  &&  $D_y \leq \text{latitude}$

Kondisi 4, garis DA: if  $B_y \geq \text{latitude}$  &&  $C_y \geq \text{latitude}$

Setelah semua kondisi terpenuhi dan bernilai benar maka *user* dianggap berada pada area lokasi Gedung Saintek UIN MALIKI. *Pseudocode* Pada gambar 4.3 memperjelas jalannya algoritma pendeteksian area lokasi menggunakan *longitude* dan *latitude*.



Gambar 4.2 Proses penentuan area lokasi

```

Deklarasi:
double longitude, latitude
double Ax, Ay, Bx, By, Cx, Cy, Dx, Dy
String nama_lokasi
get longitude, latitude
if (Ax >= longitude && Bx <= longitude)
    && (Cx >= longitude && Dx > longitude)
    && (Ay <= latitude && Dy <= latitude)
    && (By >= latitude && Cy > latitude)
    set nama_lokasi
endif

```

Gambar 4.3 *Pseudo code* proses penentuan area lokasi

#### 4.1.1.3 Tahapan Proses Sinkronisasi ke Webserver

Proses sinkronisasi database sqlite ke MySQL dimulai dengan pemanggilan kelas *AsyncHttpClient*. Kemudian memanggil kelas *ArrayList* yang berisi database SQLite dari tabel *users*. Kemudian akan dilakukan pengecekan apakah tabel *users* berisi data lokasi, jika tidak maka *user* akan menerima notifikasi bahwa belum ada data lokasi yang bisa disinkronkan atau data kosong. Namun apabila data tersedia maka akan dilakukan pengecekan apakah data lokasi sudah tersinkronisasi. Setelah data lokasi sudah disinkronisasi *user* mendapat notifikasi bahwa data lokasi sudah tersinkronisasi. Apabila data lokasi pada database sqlite belum tersinkronisasi maka akan dilanjutkan peng-*upload*-an data lokasi berupa id, id android, nama lokasi, longitude, latitude, hari, waktu, dan status.

```

Deklarasi AsyncHttpClient, ArrayList, dbHelper
if ArrayList != 0
    if dbHelper != 0
        ambil data dari sqlite
        post ke web server with json
        if succes
            set status yes, set notifikasi berhasil
        else
            set status no, set notifikasi gagal
    else
        set notifikasi sudah tersinkronisasi
else
    Data tidak tersedia

```

Gambar 4.4 *Pseudo code* proses sinkronisasi ke web server

#### 4.1.1.4 Tahapan Proses Otomatisasi Aplikasi

Proses otomatisasi aplikasi menggunakan *Handler* sebagai *thread* yang telah tersedia di pemrograman java yang digunakan android. Thread ini berfungsi untuk melakukan eksekusi perintah pengaksesan lokasi yang telah diatur oleh sistem. Aplikasi ini memanggil thread dengan kelas *Handler*, yang mengijinkan aplikasi untuk mengirim dan memproses pesan dan me-*Runnable* objek untuk dieksekusi.

Gambar 4.10 merupaka *pseudo code* proses otomatisasi presensi. Proses otomatis ini menggunakan *Handler*, kelas *Handler* harus di deklarasikan dulu sebelum dipanggil. Setelah melakukan pendeklarasian *Handler* maka metode *Runnable* dapat dipanggil yang berfungsi untuk memanggil metode *run*. Pemanggilan metode *run* ini berfungsi untuk melakukan proses *thread*. Didalam metode *run* terdapat nilai interval yang menunjukkan jeda tiap *thread* yang dieksekusi dalam satuan milidetik(ms). Sedangkan *thread* yang dieksekusi adalah metode update lokasi. Di dalam metode update lokasi terdapat metode pemanggilan data lokasi dan menampilkan data dalam bentuk *Textview*.

```

Deklarasi Handler
call method Runnable
    call method run
        set interval
        update lokasi
    end run
end Runnable
  
```

Gambar 4.5 *Pseudo code* proses otomatisasi presensi

#### 4.1.1.5 Tahapan Penghitungan Manual Jarak Kampus

Penghitungan jarak menggunakan Jarak Euclidean yang menghasilkan nilai jarak dua titik. Dengan rumus:

$$D = \sqrt{(x_p - x_k)^2 + (y_p - y_k)^2} * 111.319$$

Keterangan :  $D = \text{Jarak}$   
 $x_p = \text{longitude user}$   
 $x_k = \text{longitude kampus}$   
 $y_p = \text{latitude user}$   
 $y_k = \text{latitude kampus}$

Nilai 111.319 adalah nilai satu derajat longitude pada garis equator. Sehingga nilai longitude dan latitude bisa diketahui jarak sebenarnya.

Inputan berupa *longitude* dan *latitude user*: 112.57487, -7.938198333

Nilai *longitude* dan *latitude kampus*: 112.607991, -7.951112

Maka hasil penghitungannya:

$$D = \sqrt{(112.57487 - 112.607991)^2 + (-7.938198 - (-7.951112))^2} * 111.319$$

$$D = \sqrt{0.0010969344 + 0.000166771396} * 111.319$$

$$D = 0.03554863986146305429416589037582 * 111.319$$

$$D = 3.9572390407382057409722527507459$$

#### 4.1.2 Implementasi Interface

##### 4.1.2.1 Tampilan Awal Aplikasi Android

Gambar 4.1 merupakan tampilan awal aplikasi presensi di android yang hanya bisa diakses oleh *user*. Pada bagian atas terdapat tombol menu settings yang berfungsi untuk melakukan pengaturan terhadap interval pengecekan lokasi serta menyalakan dan mematikan fungsi akses lokasi. Kemudian dibawahnya empat baris teks yang berisi data lokasi, diantaranya nama lokasi, hari dan tanggal *check-in*, waktu *check-in*, dan ID Android. Pada bagian ini *user* akan mendapat data lokasi terbaru setelah gps diaktifkan oleh *user*. Dibawahnya merupakan daftar dari history aktifitas terkini *user* yang belum tersinkronisasi oleh *web server*, pada bagian ini *user* akan mendapat notifikasi ketika status sinkronisasi. Apabila data sudah tersinkron dengan web server maka *user* akan dapat perintah untuk mensinkronkan dengan tombol sinkronisasi. Kemudian, pada bagian bawah terdapat empat tombol, yakni tambah lokasi, *history*, sinkronisasi, dan jarak. Tombol tambah lokasi digunakan untuk menambahkan lokasi ke dalam database *web server* secara manual. Tombol *history* digunakan untuk melihat catatan *check-in user* yang telah disinkronkan dengan *web server*. Tombol sinkronisasi

digunakan untuk melakukan sinkronisasi data yang telah ditambahkan oleh *user* ke *web server online*. Sedangkan tombol jarak digunakan untuk mengetahui jarak *user* dengan kampus ketika *user* berada diluar area kampus. Penghitungan jarak ini menggunakan jarak *euclidean*.



Gambar 4.6 Tampilan awal Aplikasi Presensi di Android

#### 4.1.2.2 Tampilan Menu History

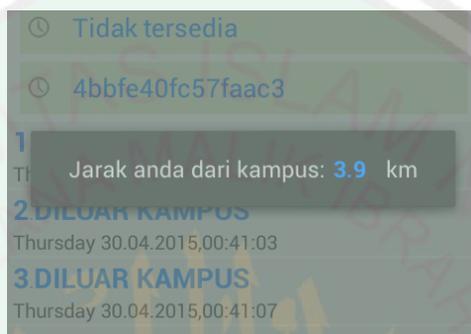
Gambar 4.2 merupakan tampilan history lokasi yang telah diinputkan oleh *user*. Data ini diambil dari *webserver online* yang langsung tersinkron dengan aplikasi *user*. Pada menu ini *user* hanya diberi kewenangan untuk melihat detail lokasi pengguna dengan informasi hari dan waktu ketika dia melakukan *check-in* ke dalam kampus tanpa bisa melakukan perubahan termasuk edit dan delete.



Gambar 4.7 Tampilan history aplikasi presensi di android

#### 4.1.2.3 Tampilan Menu Jarak Lokasi

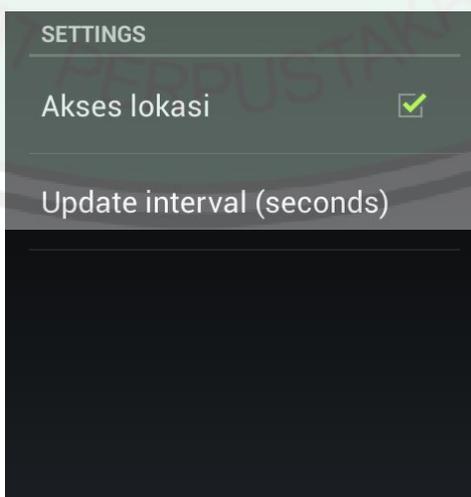
Gambar 4.3 merupakan tampilan menu jarak lokasi dari kampus ketika *user* berada di luar lokasi kampus. Penghitungan jarak ini menggunakan metode jarak euclidean dimana lokasi *user* yang dinamis berupa *longitude* dan *latitude* dihitung dengan satu titik data lokasi kampus berupa *longitude* dan *latitude* yang statis.



Gambar 4.8 Tampilan menu jarak lokasi

#### 4.1.2.4 Tampilan Menu Settings Aplikasi

Pada gambar 4.4 ini merupakan halaman menu *settings* aplikasi. Halaman menu *settings* berfungsi untuk melakukan pengaturan aplikasi, yakni mengaktifkan akses lokasi dan mengatur interval deteksi dengan nilai rentang 1 menit, 5 menit, 10 menit, 30 menit, dan 60 menit sehingga aplikasi bisa berjalan secara otomatis dengan rentang waktu yang telah dipilih oleh *user*.



Gambar 4.9 Tampilan Menu Settings

#### 4.1.2.5 Tampilan Halaman Login Admin di Web

Gambar 4.10 merupakan tampilan halaman login admin pada web server. Admin diminta untuk memasukkan *username* dan *password* yang sesuai untuk dapat melakukan login ke *web server*.

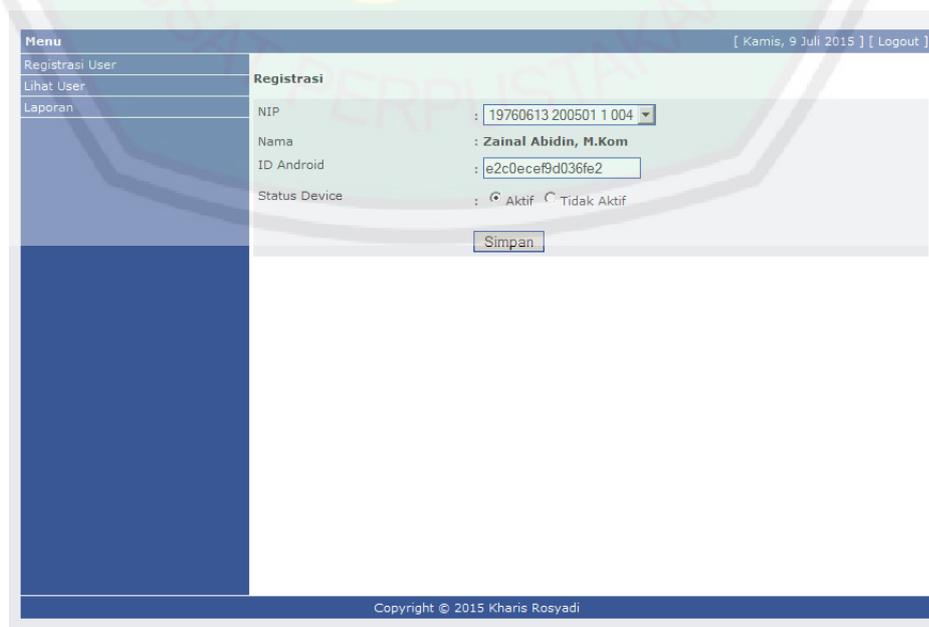


Sistem Presensi Pegawai UIN Maulana Malik Ibrahim Malang	
Username	<input type="text"/>
Password	<input type="password"/>
<input type="button" value="LOGIN"/>	

Gambar 4.10 Tampilan halaman login *web server*

#### 4.1.2.6 Tampilan Halaman Registrasi User

Gambar 4.11 merupakan tampilan halaman Registrasi User. Pendaftaran user dilakukan dengan inputan berupa nip pegawai yang diambil dari Sistem Kepegawaian. Kemudian setelah memilih nip maka akan muncul nama pegawai yang bersangkutan dan diinputkan pula id android pegawai sesuai perangkat yang pegawai miliki yang sedang aktif.



Menu	Registrasi
Registrasi User	NIP : 197606132005011004
Lihat User	Nama : Zainal Abidin, M.Kom
Laporan	ID Android : e2c0ecef9d036fe2
	Status Device : <input checked="" type="radio"/> Aktif <input type="radio"/> Tidak Aktif
	<input type="button" value="Simpan"/>

Copyright © 2015 Kharis Rosyadi

Gambar 4.11 Tampilan halaman Registrasi User

#### 4.1.2.7 Tampilan Halaman Lihat User

Gambar 4.12 merupakan tampilan halaman Lihat User. Halaman ini berisi user yang telah terdaftar di sistem presensi dengan keterangan id android pegawai yang sedang aktif. Pada bagian aksi per user admin dapat mengupdate data id android user yang aktif dan admin akan masuk pada menu edit user. Admin juga dapat menghapus user yang bersangkutan.

No.	NIP	Nama Pegawai	ID Android	Aksi
1	196607201997031004	JOHANES ROBBY WAANI ST	5de48862f7a7f4da	Edit   Hapus
2	196411191997031002	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT	9d6994e2647ed62a	Edit   Hapus
3	195907101983031030	DARMADJI SUHEDI ST.MT	d06b5639b93bbb92	Edit   Hapus
4	197907072005021001	IBNU KURNIAWAN ST	30c17c1009bbf7fc	Edit   Hapus
5	19760613 200501 1 004	Zainal Abidin, M.Kom	e2c0ecef9d036fe2	Edit   Hapus

Gambar 4.12 Tampilan halaman Lihat User

#### 4.1.2.8 Tampilan Halaman Laporan

Gambar 4.13 merupakan tampilan halaman laporan, halaman ini berfungsi untuk merekap laporan presensi yang telah dilakukan user. Pada halaman ini terdapat menu inputan untuk memilih tanggal yang akan dilihat rekapan-nya sesuai tanggal yang telah dipilih. Pada menu bawah terdapat daftar user yang telah melakukan presensi. Apabila nama user dipilih maka akan masuk ke menu detail presensi yang telah dilakukan user. Pada bagian ini admin dapat melihat detail aktifitas presensi per user yang menampilkan nama lokasi user, tanggal, dan jam lokasi.

Menu		[ Kamis, 9 Juli 2015 ] [ Logout ]			
Registrasi User	<b>Laporan Presensi Per Periode</b> Periode: <input type="text"/> s/d <input type="text"/> <input type="button" value="Tampilkan"/>				
Lihat User					
Laporan					
No.	Tanggal	Waktu	Nama Lokasi	Nama Pegawai	
1	2015-05-03	13:06:35	SAINTEK	JOHANES ROBBY WAANI ST	
2	2015-05-03	13:08:35	SAINTEK	JOHANES ROBBY WAANI ST	
3	2015-05-03	13:09:35	SAINTEK	JOHANES ROBBY WAANI ST	
4	2015-05-03	13:10:35	SAINTEK	JOHANES ROBBY WAANI ST	
5	2015-05-03	13:16:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT	
6	2015-05-03	13:17:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT	
7	2015-05-03	13:18:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT	
8	2015-05-03	13:19:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT	
9	2015-05-03	13:20:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT	
10	2015-05-03	13:25:29	SAINTEK	DARMADJI SUHEDI ST.MT	
11	2015-05-03	13:26:29	SAINTEK	DARMADJI SUHEDI ST.MT	
12	2015-05-03	13:27:29	SAINTEK	DARMADJI SUHEDI ST.MT	
13	2015-05-03	13:28:29	SAINTEK	DARMADJI SUHEDI ST.MT	
14	2015-05-03	13:29:29	SAINTEK	DARMADJI SUHEDI ST.MT	
15	2015-05-03	13:34:22	Perpustakaan	IBNU KURNIAWAN ST	
16	2015-05-03	13:35:22	Perpustakaan	IBNU KURNIAWAN ST	
17	2015-05-03	13:36:22	Perpustakaan	IBNU KURNIAWAN ST	
18	2015-05-03	13:37:22	Perpustakaan	IBNU KURNIAWAN ST	
19	2015-05-03	13:38:22	Perpustakaan	IBNU KURNIAWAN ST	
20	2015-05-03	14:15:42	Perpustakaan	Zainal Abidin, M.Kom	
21	2015-05-03	14:16:42	Perpustakaan	Zainal Abidin, M.Kom	

Gambar 4.13 Tampilan halaman laporan

#### 4.1.2.9 Tampilan Halaman Laporan Detail User

Gambar 4.13 merupakan tampilan halaman detail user. Pada menu ini admin dapat melihat data lokasi user dan dapat mengkonversinya ke dalam format .xls.

Menu		[ Kamis, 9 Juli 2015 ] [ Logout ]				
Registrasi User	<b>Detail Aktivitas Presensi Pegawai</b> NIP : 19760613 200501 1 004 ID Android : e2c0ecef9d036fe2 Nama : Zainal Abidin, M.Kom <input type="button" value="Simpan ke .xls"/>					
Lihat User						
Laporan						
No.	Nama Lokasi	Hari	Waktu	Longitude	Latitude	
1	Perpustakaan	2015-05-03	14:15:42	112.607670	-7.950703	
2	Perpustakaan	2015-05-03	14:16:42	112.607670	-7.950703	
3	Perpustakaan	2015-05-03	14:17:42	112.607670	-7.950703	
4	Perpustakaan	2015-05-03	14:18:42	112.607670	-7.950703	
5	Perpustakaan	2015-05-03	14:19:42	112.607670	-7.950703	

Copyright © 2015 Kharis Rosyadi

Gambar 4.14 Tampilan halaman detail user

#### 4.1.2.10 Tampilan Halaman Daftar Lokasi Per Periode

Gambar 4.15 merupakan halaman daftar lokasi pada web server pada halaman admin. Halaman ini berfungsi untuk mengetahui seluruh data lokasi yang telah disinkronkan oleh *user* ke web server. Data ini meliputi ID Android, hari dan tanggal, waktu, nama lokasi, longitude, dan latitude *user*.

Menu					
Registrasi User	[ Kamis, 9 Juli 2015 ] [ Logout ]				
Lihat User	Hasil pencarian data pada tanggal 2015-05-03 sampai tanggal 2015-07-05				
Laporan	No.	Nama Lokasi	Hari	Waktu	Nama Pegawai
	1	SAINTEK	2015-05-03	13:28:29	DARMADJI SUHEDI ST.MT
	2	Perpustakaan	2015-05-03	14:17:42	Zainal Abidin, M.Kom
	3	SAINTEK	2015-05-03	13:09:35	JOHANES ROBBY WAANI ST
	4	SAINTEK	2015-05-03	13:27:29	DARMADJI SUHEDI ST.MT
	5	Perpustakaan	2015-05-03	14:16:42	Zainal Abidin, M.Kom
	6	SAINTEK	2015-05-03	13:08:35	JOHANES ROBBY WAANI ST
	7	SAINTEK	2015-05-03	13:26:29	DARMADJI SUHEDI ST.MT
	8	Perpustakaan	2015-05-03	14:15:42	Zainal Abidin, M.Kom
	9	SAINTEK	2015-05-03	13:06:35	JOHANES ROBBY WAANI ST
	10	SAINTEK	2015-05-03	13:25:29	DARMADJI SUHEDI ST.MT
	11	Perpustakaan	2015-05-03	13:38:22	IBNU KURNIAWAN ST
	12	SAINTEK	2015-05-03	13:20:11	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
	13	Perpustakaan	2015-05-03	13:37:22	IBNU KURNIAWAN ST
	14	SAINTEK	2015-05-03	13:19:11	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
	15	Perpustakaan	2015-05-03	13:36:22	IBNU KURNIAWAN ST
	16	SAINTEK	2015-05-03	13:18:11	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
	17	Perpustakaan	2015-05-03	13:35:22	IBNU KURNIAWAN ST
	18	SAINTEK	2015-05-03	13:17:11	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
	19	Perpustakaan	2015-05-03	13:34:22	IBNU KURNIAWAN ST
	20	Perpustakaan	2015-05-03	14:19:42	Zainal Abidin, M.Kom
	21	SAINTEK	2015-05-03	13:16:11	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
	22	SAINTEK	2015-05-03	13:29:29	DARMADJI SUHEDI ST.MT
	23	Perpustakaan	2015-05-03	14:18:42	Zainal Abidin, M.Kom
	24	SAINTEK	2015-05-03	13:10:35	JOHANES ROBBY WAANI ST

Copyright © 2015 Kharis Rosyadi

Gambar 4.15 Tampilan halaman daftar lokasi per periode pada web servers

## 4.2 Pengujian Sistem

Pengujian sistem merupakan suatu langkah yang harus dilakukan untuk menjelaskan mengenai pengoperasian perangkat lunak yang telah dibangun. Pengujian tersebut dibagi beberapa item yang masing-masing item akan diuji.

### 4.2.1 Rencana Pengujian

Dalam perencanaan pengujian ini, pengujian akan dibagi dalam beberapa item. Dari hasil pengujian tersebut akan didapatkan hasil pengujian yang nantinya dapat ditarik kesimpulan dari pengujian tersebut. Pengujian ini dilakukan dengan tujuan apakah aplikasi yang dibangun sudah berjalan dengan baik dan sesuai dengan yang penulis harapkan

#### 4.2.2 Pengujian

Pengujian yang dilakukan menitikberatkan pada hasil keluaran dari kendali masukan. Pada kendali masukan, pengujian dikatakan berhasil apabila keluaran sesuai dengan kendali masukan. Berdasarkan rencana pengujian sistem ini yang telah disusun, maka dapat dilakukan pengujian sebagai berikut:

**Tabel 4.1** Rencana Pengujian

Item Uji	Butir Uji
Pengambilan data lokasi	Mengambil data lokasi berupa <i>longitude</i> dan <i>latitude</i>
Menentukan area lokasi kampus	Menentukan area lokasi berdasarkan <i>longitude</i> dan <i>latitude</i> didalam area kampus, gedung/ bangunan
Melakukan proses sinkronisasi ke <i>web server</i>	Melakukan proses sinkronisasi database <i>sqlite</i> dengan MySQL
Melakukan proses otomatisasi presensi	Melakukan proses otomatisasi presensi berdasarkan interval waktu pengecekan lokasi
Menampilkan data lokasi di <i>web server</i>	Menampilkan data berupa id android, nama, <i>longitude</i> , <i>latitude</i> , hari, dan jam di <i>web server</i>
Menentukan jarak lokasi kampus	Menampilkan jarak kampus apabila lokasi berada diluar kampus

##### 4.2.2.1 Pengambilan Data Lokasi

Hasil pengujian input data lokasi dapat dilihat pada tabel 4.2. Yang memiliki butir uji, Masukkan perangkat android ke lokasi, menyalakan fungsi akses lokasi, mengatur interval akses lokasi.

**Tabel 4.2** Hasil uji input data lokasi

<b>Kasus dan Hasil Uji</b>			
<b>Butir Uji</b>	<b>Hasil yang diharapkan</b>	<b>Hasil pengamatan</b>	<b>Kesimpulan</b>
Masukkan perangkat android ke lokasi	Sistem mampu mendeteksi perubahan nama lokasi	Perangkat mampu mendeteksi nama lokasi berbasis <i>longitude</i> dan <i>latitude</i>	[√] sukses [ ] gagal
Menyalakan fungsi akses lokasi	Sistem mampu menyalakan fungsi akses lokasi	Perangkat mampu menyalakan dan menghentikan akses lokasi	[√] sukses [ ] gagal
Mengatur interval akses lokasi	Sistem mampu mengatur interval waktu pengaksesan lokasi	Perangkat mampu melakukan akses lokasi berdasarkan waktu interval yang telah diatur	[√] sukses [ ] gagal

#### 4.2.2.2 Menentukan Area Lokasi Kampus

Tabel 4.3 merupakan hasil dari penentuan area lokasi kampus dengan berbagai perangkat uji, pengujian dilakukan pada hari sabtu, 2 Mei 2015 di Gedung Saintek dan Gedung Perpustakaan. Berikut hasil pengujian area lokasi kampus dengan berbagai perangkat:

**Tabel 4.3** Hasil penentuan area lokasi

<b>ID</b>	<b>ID Android</b>	<b>Nama Lokasi</b>	<b>Hari</b>	<b>Jam</b>	<b>Keterangan</b>
1	4bbfe40fc57faac3	Saintek	Sabtu	13:03	[√] sukses [ ] gagal
2	5de48862f7a7f4da	Saintek	Sabtu	13:06	[√] sukses [ ] gagal
3	9d6994e2647ed62a	Saintek	Sabtu	13:16	[√] sukses [ ] gagal

ID	ID Android	Nama Lokasi	Hari	Jam	Keterangan
4	d06b5639b93bbb92	Saintek	Sabtu	13:25	<input checked="" type="checkbox"/> sukses <input type="checkbox"/> gagal
5	30c17c1009bbf7fc	Perpustakaan	Sabtu	13:34	<input checked="" type="checkbox"/> sukses <input type="checkbox"/> gagal
6	9043d45ef251c6b0	Perpustakaan	Sabtu	13:43	<input checked="" type="checkbox"/> sukses <input type="checkbox"/> gagal
7	29f41c7a4e7d0480	Perpustakaan	Sabtu	13:51	<input checked="" type="checkbox"/> sukses <input type="checkbox"/> gagal
8	75869bbd4faa3f1a	Perpustakaan	Sabtu	14:04	<input checked="" type="checkbox"/> sukses <input type="checkbox"/> gagal
9	e2c0ecef9d036fe2	Perpustakaan	Minggu	10:15	<input checked="" type="checkbox"/> sukses <input type="checkbox"/> gagal
10	a622bd5768771888	Perpustakaan	Minggu	10:20	<input checked="" type="checkbox"/> sukses <input type="checkbox"/> gagal

#### 4.2.2.3 Melakukan Proses Otomatisasi Presensi

Berikut ini hasil otomatisasi aplikasi presensi pada berbagai perangkat yang berbeda. Otomatisasi ini ditunjukkan dengan nilai interval 1 menit pada masing-masing perangkat.

##### A. Acer Z160

Tabel 4.4 merupakan tabel data lokasi pada perangkat Acer Z160 yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.4** Hasil otomatisasi pada perangkat Acer Z160

No	ID Android	Nama Lokasi	Hari	Jam
1	4bbfe40fc57faac3	Saintek	Sabtu	13:03:41
2	4bbfe40fc57faac3	Saintek	Sabtu	13:04:41
3	4bbfe40fc57faac3	Saintek	Sabtu	13:05:41

No	ID Android	Nama Lokasi	Hari	Jam
4	4bbfe40fc57faac3	Saintek	Sabtu	13:06:41
5	4bbfe40fc57faac3	Saintek	Sabtu	13:07:41

### B. Sony Xperia J

Tabel 4.5 merupakan tabel data lokasi pada perangkat Sony Xperia J yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.5** Hasil otomatisasi pada perangkat Sony Xperia J

No	ID Android	Nama Lokasi	Hari	Jam
1	5de48862f7a7f4da	Saintek	Sabtu	13:06:35
2	5de48862f7a7f4da	Saintek	Sabtu	13:07:35
3	5de48862f7a7f4da	Saintek	Sabtu	13:08:35
4	5de48862f7a7f4da	Saintek	Sabtu	13:09:35
5	5de48862f7a7f4da	Saintek	Sabtu	13:10:35

### C. Cross A26

Tabel 4.6 merupakan tabel data lokasi pada perangkat Cross A26 yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.6** Hasil otomatisasi pada perangkat Cross A26

No	ID Android	Nama Lokasi	Hari	Jam
1	9d6994e2647ed62a	Saintek	Sabtu	13:16:11
2	9d6994e2647ed62a	Saintek	Sabtu	13:17:11
3	9d6994e2647ed62a	Saintek	Sabtu	13:18:11
4	9d6994e2647ed62a	Saintek	Sabtu	13:19:11

No	ID Android	Nama Lokasi	Hari	Jam
5	9d6994e2647ed62a	Saintek	Sabtu	13:20:11

#### D. Sony Xperia E

Tabel 4.7 merupakan tabel data lokasi pada perangkat Sony Xperia E yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.7** Hasil otomatisasi pada perangkat Xperia E

No	ID Android	Nama Lokasi	Hari	Jam
1	d06b5639b93bbb92	Saintek	Sabtu	13:25:29
2	d06b5639b93bbb92	Saintek	Sabtu	13:26:29
3	d06b5639b93bbb92	Saintek	Sabtu	13:27:29
4	d06b5639b93bbb92	Saintek	Sabtu	13:28:29
5	d06b5639b93bbb92	Saintek	Sabtu	13:29:29

#### E. Redmi 2

Tabel 4.8 merupakan tabel data lokasi pada perangkat Redmi 2 yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.8** Hasil otomatisasi pada perangkat Redmi 2

No	ID Android	Nama Lokasi	Hari	Jam
1	30c17c1009bbf7fc	Perpustakaan	Sabtu	13:34:22
2	30c17c1009bbf7fc	Perpustakaan	Sabtu	13:35:22
3	30c17c1009bbf7fc	Perpustakaan	Sabtu	13:36:22
4	30c17c1009bbf7fc	Perpustakaan	Sabtu	13:37:22
5	30c17c1009bbf7fc	Perpustakaan	Sabtu	13:38:22

#### F. Asus Zenfone 4

Tabel 4.9 merupakan tabel data lokasi pada perangkat Asus Zenfone 4 yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.9** Hasil otomatisasi pada perangkat Asus Zenfone 4

No	ID Android	Nama Lokasi	Hari	Jam
1	9043d45ef251c6b0	Perpustakaan	Sabtu	13:43:46
2	9043d45ef251c6b0	Perpustakaan	Sabtu	13:43:46
3	9043d45ef251c6b0	Perpustakaan	Sabtu	13:43:46
4	9043d45ef251c6b0	Perpustakaan	Sabtu	13:43:46
5	9043d45ef251c6b0	Perpustakaan	Sabtu	13:43:46

#### G. Asus Zenfone 4

Tabel 4.10 merupakan tabel data lokasi pada perangkat Asus Zenfone 4 yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.10** Hasil otomatisasi pada perangkat Asus Zenfone 4

No	ID Android	Nama Lokasi	Hari	Jam
1	29f41c7a4e7d0480	Perpustakaan	Sabtu	13:51:30
2	29f41c7a4e7d0480	Perpustakaan	Sabtu	13:52:30
3	29f41c7a4e7d0480	Perpustakaan	Sabtu	13:53:30
4	29f41c7a4e7d0480	Perpustakaan	Sabtu	13:54:30
5	29f41c7a4e7d0480	Perpustakaan	Sabtu	13:55:30

## H. Sony Xperia L

Tabel 4.11 merupakan tabel data lokasi pada perangkat Sony Xperia L yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.11** Hasil otomatisasi pada perangkat Sony Xperia L

No	ID Android	Nama Lokasi	Hari	Jam
1	75869bbd4faa3f1a	Perpustakaan	Sabtu	14:04:05
2	75869bbd4faa3f1a	Perpustakaan	Sabtu	14:05:05
3	75869bbd4faa3f1a	Perpustakaan	Sabtu	14:06:05
4	75869bbd4faa3f1a	Perpustakaan	Sabtu	14:07:05
5	75869bbd4faa3f1a	Perpustakaan	Sabtu	14:08:05

## I. Sony Xperia M

Tabel 4.12 merupakan tabel data lokasi pada perangkat Sony Xperia M yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.12** Hasil otomatisasi pada perangkat Sony Xperia M

No	ID Android	Nama Lokasi	Hari	Jam
1	e2c0ecef9d036fe2	Perpustakaan	Minggu	10:15:42
2	e2c0ecef9d036fe2	Perpustakaan	Minggu	10:16:42
3	e2c0ecef9d036fe2	Perpustakaan	Minggu	10:17:42
4	e2c0ecef9d036fe2	Perpustakaan	Minggu	10:18:42
5	e2c0ecef9d036fe2	Perpustakaan	Minggu	10:19:42

#### J. Lenovo A6000

Tabel 4.13 merupakan tabel data lokasi pada perangkat Lenovo A6000 yang terinput secara otomatis pada web server dengan nilai interval 1 menit.

**Tabel 4.13** Hasil otomatisasi pada perangkat Lenovo A6000

No	ID Android	Nama Lokasi	Hari	Jam
1	a622bd5768771888	Perpustakaan	Minggu	10:20:39
2	a622bd5768771888	Perpustakaan	Minggu	10:21:39
3	a622bd5768771888	Perpustakaan	Minggu	10:22:39
4	a622bd5768771888	Perpustakaan	Minggu	10:23:39
5	a622bd5768771888	Perpustakaan	Minggu	10:24:39

#### 4.2.2.4 Menampilkan Data Lokasi Di Web Server

Data lokasi *user* tersimpan di *web server online*. Sebelum data bisa ditampilkan maka *user* harus menginputkan lokasinya pada *web server* dengan nilai interval sesuai pengaturan yang ditetapkan oleh *user* sendiri dengan nilai 1 menit, 5 menit, 10 menit, 15 menit, 30 menit, dan 60 menit. Untuk menampilkan data lokasi di *web server online*, hanya bisa dilakukan oleh admin untuk alasan keamanan. Namun *user* dapat pula melihat data lokasi yang telah diinputkannya di dalam aplikasi dengan menu history.

Gambar 4.16 merupakan tampilan tabel pada interface web server yang menunjukkan data lokasi yang sudah tersinkronisasi. Data lokasi ini berupa id, id android, nama lokasi, hari, waktu, longitude, dan latitude. Pada halaman ini admin bisa melakukan manajemen data berupa update data, tambah data, dan hapus data. Selain itu admin juga bisa membuat pelaporan aktivitas *user*, laporan ini bisa per hari, per minggu, maupun perbulan.

No.	Tanggal	Waktu	Nama Lokasi	Nama Pegawai
1	2015-05-03	13:06:35	SAINTEK	JOHANES ROBBY WAANI ST
2	2015-05-03	13:08:35	SAINTEK	JOHANES ROBBY WAANI ST
3	2015-05-03	13:09:35	SAINTEK	JOHANES ROBBY WAANI ST
4	2015-05-03	13:10:35	SAINTEK	JOHANES ROBBY WAANI ST
5	2015-05-03	13:16:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
6	2015-05-03	13:17:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
7	2015-05-03	13:18:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
8	2015-05-03	13:19:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
9	2015-05-03	13:20:11	SAINTEK	IR. WILLIAM TAVIPONOVA GLORICUS KEREH MT
10	2015-05-03	13:25:29	SAINTEK	DARMADJI SUHEDI ST.MT
11	2015-05-03	13:26:29	SAINTEK	DARMADJI SUHEDI ST.MT
12	2015-05-03	13:27:29	SAINTEK	DARMADJI SUHEDI ST.MT
13	2015-05-03	13:28:29	SAINTEK	DARMADJI SUHEDI ST.MT
14	2015-05-03	13:29:29	SAINTEK	DARMADJI SUHEDI ST.MT
15	2015-05-03	13:34:22	Perpustakaan	IBNU KURNIAWAN ST
16	2015-05-03	13:35:22	Perpustakaan	IBNU KURNIAWAN ST
17	2015-05-03	13:36:22	Perpustakaan	IBNU KURNIAWAN ST
18	2015-05-03	13:37:22	Perpustakaan	IBNU KURNIAWAN ST
19	2015-05-03	13:38:22	Perpustakaan	IBNU KURNIAWAN ST
20	2015-05-03	14:15:42	Perpustakaan	Zainal Abidin, M.Kom
21	2015-05-03	14:16:42	Perpustakaan	Zainal Abidin, M.Kom

Gambar 4.16 Tampilan data lokasi di web server

#### 4.2.2.5 Menentukan Jarak Lokasi Kampus

Jarak lokasi kampus digunakan untuk mengetahui lokasi user dengan kampus dengan nilai inputan berupa titik lokasi kampus dan dihitung dengan jarak *euclidean* dengan posisi lokasi user menggunakan GPS. Posisi ini hanya menghitung lokasi *user* dengan kampus berdasar pada garis lurus, bukan berdasar jarak jalan yang harus antara *user* dengan kampus. Fitur ini tidak berfungsi langsung pada *web server* karena tidak tersimpan di *database* dan hanya fitur tambahan untuk mengetahui lokasi kampus dengan *user*.

Gambar 4.10 merupakan tampilan menu jarak lokasi kampus ketika *user* berada diluar area kampus. Fitur ini dimaksudkan untuk mengetahui jarak antara *user* dengan kampus, sehingga *user* dapat mengetahui jarak dari *user* ke kampus.



Gambar 4.13 Tampilan jarak lokasi kampus

#### 4.3 Presensi Menurut Kajian Dalam Al-Qur'an dan Hadits

Presensi merupakan tolak ukur utama suatu instansi untuk menjunjung tinggi nilai kedisiplinan karyawan/ anggotanya. Dalam prakteknya presensi juga bisa dipakai untuk mengetahui tingkat kinerja karyawan yang bersangkutan. Dimana untuk menunjang etos kerja yang positif dalam suatu instansi presensi sangat krusial perannya, karena ketika seorang performa karyawan menurun dalam presensinya maka hal itu juga berakibat pada produktivitas instansi yang bersangkutan. Hingga saat ini presensi memiliki beberapa alat yang bisa dipraktekkan, seperti: presensi manual, presensi menggunakan sidik jari, presensi menggunakan biometrik, maupun presensi menggunakan kartu. Meskipun demikian tetap saja ada karyawan yang masih mampu melakukan penyelewengan terkait masalah kedisiplinannya. Karena presensi tersebut tidak berjalan tidak secara *real-time* yang berakibat beberapa *user* dengan bebas melakukan penyelewengan.

Didalam Al-Quran disebutkan bahwa Allah SWT memerintahkan disiplin dalam arti ketaatan pada peraturan yan ditetapkan, yakni surat An Nisa 59 yang berbunyi:

يَا أَيُّهَا الَّذِينَ ءَامَنُوا أَطِيعُوا اللَّهَ وَأَطِيعُوا الرَّسُولَ وَأُولِي الْأَمْرِ مِنْكُمْ صَل فَإِن تَنَزَعْتُمْ فِي شَيْءٍ فَرُدُّوهُ إِلَى اللَّهِ وَالرَّسُولِ إِن كُنتُمْ تُؤْمِنُونَ بِاللَّهِ وَالْيَوْمِ الْآخِرِ ذَلِكَ خَيْرٌ وَأَحْسَنُ تَأْوِيلًا.

*Hai orang-orang yang beriman, taatilah Allah dan taatilah Rasul (Nya), dan ulil amri di antara kamu. kemudian jika kamu berlainan Pendapat tentang sesuatu, Maka kembalikanlah ia kepada Allah (Al Quran) dan Rasul (sunnahnya), jika kamu benar-benar beriman kepada Allah dan hari kemudian. yang demikian itu lebih utama (bagimu) dan lebih baik akibatnya.*

Dalam ayat tersebut dijelaskan bahwa Pertama, para mukmin diperintahkan untuk taat kepada Allah, yakni untuk menjalankan perintah-Nya dan menjauhi larangan-Nya serta tunduk dan patuh pada segala ketentuan dalam al-Quran dan as-Sunnah. Kedua untuk menaati Rasulullah SAW yang diutus dengan membawa risalah dari Allah Swt sehingga wajib di taati. Ketiga perintah menaati ulil amri yakni ulama', ahli fikih, dan pemerintah termasuk disini dalam instansi yang berwenang diatas kita. Tindakan taat ini dimaksudkan atas komitmen kita untuk menjunjung tinggi nilai kedisiplinan dalam suatu instansi.

Cerminan disiplin juga selaras dengan nilai yang diemban seorang karyawan terhadap beban yang ditanggungnya dengan penuh tanggung jawab. Seperti dijelaskan pada hadits:

عَنْ عَبْدِ اللَّهِ بْنِ عُمَرَ يَقُولُ سَمِعْتُ رَسُولَ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ رَاعٍ كُلُّكُمْ رَاعٍ وَكُلُّكُمْ مَسْئُولٌ عَنْ رَعِيَّتِهِ الْإِمَامُ رَاعٍ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ وَالرَّجُلُ رَاعٍ فِي أَهْلِهِ وَهُوَ مَسْئُولٌ عَنْ رَعِيَّتِهِ وَالْمَرْأَةُ رَاعِيَةٌ فِي بَيْتِ زَوْجِهَا وَمَسْئُولَةٌ عَنْ رَعِيَّتِهَا وَالْخَادِمُ رَاعٍ فِي مَالِ سَيِّدِهِ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ قَالَ وَحَسِبْتُ أَنْ قَدْ قَالَ الرَّجُلُ رَاعٍ فِي مَالِ أَبِيهِ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ وَكُلُّكُمْ رَاعٍ وَمَسْئُولٌ عَنْ رَعِيَّتِهِ (رواه البخاري ومسلم والترمذي)

*Dari Abdullah bin Umar r.a., ia berkata: Aku mendengar Rasulullah SAW bersabda: Kamu semua adalah pemimpin dan harus bertanggung jawab atas*

*kepemimpinannya. Seseorang imam adalah pemimpin dan harus bertanggung jawab atas kepemimpinannya. Seseorang suami adalah pemimpin bagi istrinya adalah seorang pemimpin dalam rumah tangga dan harus bertanggung jawab atas kepemimpinannya. Pembantu adalah pemelihara terhadap harta tuannya, dia harus bertanggung jawab atas kepemimpinannya. Abdullah berkata: Saya kira (Rasulallah) bersabda juga dan seseorang anak adalah pemelihara milik orang tuanya, dia harus bertanggung jawab atas peliharaannya itu. Dan kamu semua adalah pemimpin dan harus bertanggung jawab atas kepemimpinannya. (HR Bukhari, Muslim dan Tirmidzi)*

Hadits ini menjelaskan kepada kita bahwa setiap manusia diberi tugas memimpin atau menjaga. Baik kaitannya dengan dirinya sendiri maupun dengan orang lain. Secara pribadi, seorang diberi tugas untuk menjaga dirinya sendiri. Pemuka atau imam diberi tugas memimpin rakyatnya. Suami bertugas memimpin dan menjaga istrinya. Seorang istri diberi amanat memimpin anak-anak suaminya. Pembantu diberi tugas menjaga harta atau kekayaan tuannya. Rektor diberi tugas melaksanakan pekerjaannya memimpin kampus.

Tugas adalah amanat, apapun jabatan yang ada pada diri seorang seseorang termasuk karyawan, maka harus mempertanggungjawabkan tugas yang telah dibebankan kepadanya dihadapan yang dipimpin dan di dalam pengadilan Allah kelak. Tak seorangpun mampu melepaskan diri dari tanggung jawab itu. Oleh karena itu, dia harus bersikap adil dan bijaksana dalam menjalankan tugasnya. Apabila lengah dan mengabaikan tanggung jawabnya, maka celakalah dia sebab akan menyengsarakannya dan yang telah dipimpinnya. Sehingga dalam praktiknya kewajiban seorang karyawan adalah untuk melakukan tugas dan tanggung jawabnya terhadap instansi terkait.

Selama karyawan bekerja sesuai dengan apa yang menjadi tanggung jawabnya maka dia akan mendapatkan apa yang telah diusahakannya. Seperti dalam firman Allah SWT dalam surat An najm ayat 39, berikut:

وَأَنْ لَّيْسَ لِلْإِنْسَانِ إِلَّا مَا سَعَى

*Dan bahwasanya seorang manusia tiada memperoleh selain apa yang telah diusahakannya.*

Dalam keterangan ayat tersebut di atas maka sudah jelas bahwa manusia mempunyai keharusan untuk berusaha dan mengubah diri sendiri dari kemunduran dan keterbelakangan menuju kepada kemajuan. Suatu prestasi kerja tidak akan mudah diraih oleh seseorang melainkan atas usahanya sendiri dibarengi dengan rasa optimis dan kerja keras. Ayat ini bermaksud untuk menuntun kita agar kita menerima dengan lapang dada dari apa yang kita kerjakan sesuai dengan hasil yang kita terima. Kita dilarang menambahi maupun mengurangi dari yang kita peroleh dari hasil orang lain.

Dalam kaitannya dengan skripsi ini nilai kedisiplinan wajib dilakukan oleh seluruh civitas dalam suatu instansi. Dengan adanya aplikasi ini kehadiran seorang karyawan dapat dipantau secara *real-time* untuk mendukung kedisiplinan instansi. Hal ini selaras dengan nilai ketaatan yang ada dalam kandungan Surat An Nisa ayat 59 yang memerintahkan wajib kepada kita untuk taat kepada Allah SWT, Rasulullah SAW, maupun ulul amri, ulama, dan pemerintah.

## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Dari pendefinisian masalah serta hasil analisa, implementasi dan uji coba yang telah diteliti dapat disimpulkan bahwa:

1. Presensi dapat dilakukan dengan menggunakan perangkat yang mendukung *Global Positioning System* (GPS) dengan melakukan *tracking* pada perangkat tersebut. Dalam kasus ini perangkat yang digunakan adalah perangkat berbasis sistem operasi android.
2. Penentuan area lokasi dapat dilakukan dengan menggunakan lokasi koordinat menggunakan *longitude* dan *latitude* yang sudah terlacak pada perangkat. Dengan melakukan perbandingan nilai lokasi perangkat dan area lokasi yang ditetapkan.
3. Berdasarkan hasil uji coba aplikasi penentuan jarak lokasi bekerja dengan baik dan sesuai dengan jarak kampus dan jarak *user*.
4. Berdasarkan hasil uji coba aplikasi dapat berjalan dengan baik pada perangkat android yang berbeda. Sesuai dengan kompatibilitas struktur aplikasi yang digunakan.

#### 5.2 Saran

Untuk penelitian kedepan, saran yang dapat diberikan untuk pengembangan yang mungkin dapat dilakukan pada aplikasi presensi ini adalah:

1. Aplikasi yang dapat menunjukkan data lokasi berupa objek 3D, sehingga nilai keakuratan lebih tinggi ketika seorang *user* berada di suatu gedung yang bertingkat.
2. Aplikasi yang dapat menganalisis dugaan kepada *user* ketika *user* menitipkan perangkat pada *user* lain.

**DAFTAR PUSTAKA**

- Priyanka Shah, Ruta Gadgil, Neha Tamhankar. 2012. *Location Based Reminder Using GPS For Mobile (Android)*. India
- Manav Singhal, Anupam Shukla. 2012. *Implementation of Location based Service in Android using GPS and Web Service*. India
- Winarko, E. 2006. *Perancangan Database Dengan PowerDesigner 6.32*. Jakarta: Prestasi Pustaka.
- Safaat, N. (2012). *Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Penerbit Informatika.
- Departemen Pendidikan Nasional. 2008. *Kamus Besar Bahasa Indonesia Pusat Bahasa Edisi Keempat*. 2008. Jakarta: PT Gramedia Pustaka Utama
- Departemen Agama Republik Indonesia. 2005. *Al-Qur'an Terjemahan*. Jakarta: PT Syamil Cipta Media
- Tohari, Ahmad. 2002. *Pemahaman Praktis Manajemen Sumber Daya Manusia*. Bandung: Mandar Maju.
- Hermawan S, Stephanus. 2011. *Mudah Membuat Aplikasi Android*. Binus University
- Felker, Donn. 2011. *Android Applications Development For Dummies*. Willey Publishing Inc. Canada
- Tamada, R. 2012. *Android JSON Parsing Tutorial*.  
<http://www.androidhive.info/2012/01/android-json-parsing-tutorial/> [accessed 13 April 2015]