

BAB II

TINJAUAN PUSTAKA

2.1 Absensi

Menurut Istijanto (2012), absensi adalah ketidak hadiran karyawan saat yang bersangkutan dijadwalkan bekerja. Jumlah absesn kerja dalam perusahaan menggambarkan pertukaran benefit antar karyawan dan perusahaan yang menggaji. Tentu saja, perusahaan tidak ingin rugi akibat tingginya absen karyawan.

Ada tiga ukuran absen, yaitu hilangnya waktu kerja, frekuensi atau tingkat keseringan absen, dan jumlah dalam jangka pendek. Hilangnya waktu kerja adalah jumlah total jam atau hari kerja yang hilang selama setahun. Frekuensi adalah jumlah total terjadinya absen dalam setahun, tapna memandang lama jangka waktu. Absen dalam jangka pendek adalah jumlah absen 1-2 hari kerja selama setahun.

2.2 Monitoring

Menurut Moerdiyanto (2009), *Monitoring* merupakan aktivitas yang dilakukan pimpinan untuk melihat, memonitor jalannya organisasi selama kegiatan berlangsung, dan menilai ketercapaian tujuan, melihat factor pendukung dan penghambat pelaksanaan program. Dalam *monitoring (monitoring)* dikumpulkan data dan dianalisis, hasil analisis diinterpretasikan dan dimaknakan sebagai masukan bagi pimpinan untuk mengadakan perbaikan.

2.3 Near Field Communication (NFC)

Menurut Indri Neforawati, M. Irdam Fareza, dan Vida juniarti (2015:14), *Near Field Communication (NFC)* adalah salah satu teknologi konektivitas wireless jarak dekat yang memungkinkan interaksi dua arah antar perangkat elektronik yang lebih aman dan simple. NFC juga memungkinkan penggunaanya untuk melakukan transaksi secara contactless, mengakses konten digital dan melakukan koneksi dengan perangkat elektronik hanya dengan satu sentuhan. NFC atau Near-Field Communication adalah salah satu teknologi komunikasi

antar gadget terbaru yang menggunakan sistem RFID (Radio Frequency Identification) untuk saling bertukar data dalam jarak dekat, sekitar 4 inci. Teknologi NFC (*Near Field Communication*), merupakan prinsip komunikasi nirkabel standar yang memungkinkan dua perangkat untuk bertukar data saat dalam posisi berdekatan satu sama lain. NFC bekerja hanya dalam jarak sangat dekat, hingga 20cm (7,87 inci) secara teori, meskipun jarak operasi aktual yang optimal adalah sekitar 4cm (1,57 inci) atau kurang, berbeda dengan Bluetooth, yang dapat memasang perangkat sampai jarak 50 meter.

Sedangkan Menurut Sadewa Chandra dan Tri Yanti Nurochmah (2014:8), *Near Field Communication* (NFC) merupakan satu set teknologi nirkabel atau wireless jarak dekat yang biasanya membutuhkan jarak 4cm atau kurang untuk berkomunikasi. NFC beroperasi pada frekuensi 13.65 MHz dengan rata-rata kecepatan transfer 106 Kbps – 848 Kbps. NFC selalu melibatkan Frekuensi Radio (RF) yang dapat mengaktifkan target yang didekatkan pada perangkat yang memiliki fitur NFC juga. Hal ini memungkinkan target NFC untuk dapat mengidentifikasi faktor-faktor yang sangat sederhana dari device seperti tag, stiker, ataupun kartu yang tidak memerlukan baterai.

2.4 Radio Frequency Identification (RFID)

Definisi menurut (Maryono, 2010) identifikasi dengan frekuensi radio adalah teknologi untuk mengidentifikasi seseorang atau objek benda menggunakan transmisi frekuensi radio, khususnya 125kHz, 13.65Mhz atau 800-900MHz. RFID menggunakan komunikasi gelombang radio untuk secara unik mengidentifikasi objek atau seseorang Terdapat beberapa pengertian RFID menurut (Maryono, 2010) yaitu :

- a. RFID (Radio Frequency Identification) adalah sebuah metode identifikasi dengan menggunakan sarana yang disebut label RFID atau transponder (tag) untuk menyimpan dan mengambil data jarak jauh.
- b. Label atau transponder (tag) adalah sebuah benda yang bisa dipasang atau dimasukkan di dalam sebuah produk, hewan atau bahkan manusia dengan tujuan untuk identifikasi menggunakan gelombang radio. Label RFID terdiri atas mikrochip .

2.5 Siswa

Dalam kamus Besar Bahasa Indonesia, menjelaskan bahwa siswa adalah murid (terutama pada tingkat dasar, menengah, pelajar). Siswa atau Peserta Didik menurut ketentuan umum Undang-undang RI No. 20 Tahun 2003 tentang Sistem Pendidikan Nasional, Peserta didik adalah anggota masyarakat yang berusaha mengembangkan potensi diri melalui proses pembelajaran yang tersedia pada jalur, jenjang dan jenis pendidikan tertentu.

2.6 Pengenalan Desain Android

2.6.1 Android

Menurut Nazruddin Safaat (2012), Android adalah salah satu platform sistem operasi yang digemari masyarakat karena sifatnya yang open source sehingga memungkinkan pengguna untuk melakukan pengembangan. Android merupakan generasi baru platform mobile berbasis linux yang mencakup sistem operasi, middleware, dan aplikasi.

Sedangkan Menurut Afrida, Amnah, dan Wibowo (2018 : 52), Android merupakan sebuah sistem operasi yang berbasis linux untuk perangkat portable seperti smartphone dan komputer tablet. Android menyediakan platform terbuka bagi programmer untuk mengembangkan aplikasi sendiri pada berbagai perangkat dengan sistem operasi android.

2.6.2 Android SDK

Menurut Safaat (2012), Android SDK adalah tools API (Application Programming Interface) yang diperlukan untuk mengembangkan aplikasi pada platform Android yang menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, middleware dan aplikasi kunci yang di-release oleh Google. Saat ini disediakan Android SDK (Software Development Kit) sebagai alat bantu dan API untuk mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. Sebagai platform Android aplikasi-netral, Android memberi Anda kesempatan untuk membuat Aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan Handphone/Smartphone.

2.6.3 ADT (*Android Development Tools*)

Menurut Safaat (2012) Android Development Tools (ADT) adalah *plug-in* yang didesain untuk IDE Eclipse yang memberi kita kemudahan dalam mengembangkan aplikasi Android dengan menggunakan IDE Eclipse. Dengan menggunakan ADT untuk Eclipse, ini akan memudahkan kita dalam membuat aplikasi *project* Android, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya. Selain itu kita juga dapat melakukan *running* aplikasi menggunakan Android SDK melalui Eclipse. Mengembangkan aplikasi Android dengan menggunakan ADT di Eclipse sangat mudah dan baik untuk memulai mengembangkan aplikasi Android. Semakin tinggi *platform* Android yang kita gunakan, maka penggunaan ADT dianjurkan untuk memakai versi yang lebih baru, karena munculnya *platform* baru diikuti oleh munculnya versi ADT yang terbaru.

2.6.4 Java

Menurut Rosa dan Shalahuddin (2019), Java dikembangkan oleh perusahaan Sun Microsystems. Java adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer *standalone* ataupun pada lingkungan jaringan. Java 2 adalah generasi kedua dari java *platform*.

Java berdiri di atas sebuah mesin *interpreter* yang diberi nama *Java Virtual Machine* (JVM). JVM inilah yang akan membaca *bytecode* dari file *.class* dari suatu program sebagai representasi langsung program yang berisi bahasa mesin. Java disebut juga sebagai bahasa pemrograman yang *portable* karena dapat dijalankan pada berbagai sistem operasi, asalkan pada sistem operasi tersebut terdapat JVM. Java merupakan bahasa pemrograman objek murni karena semua kode pemrogramannya dibungkus di kelas.

2.6.5 PHP (*Hypertext Preprocessor*)

Menurut Bangsawan (2015), PHP adalah singkatan dari "PHP: Hypertext Preprocessor", yaitu bahasa pemrograman yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML. PHP diciptakan oleh Rasmus Lerdorf pertama kali tahun 1994. Pada awalnya PHP adalah singkatan dari "Personal Home Page Tools". Selanjutnya diganti menjadi FI ("Forms Interpreter"). Sejak versi 3.0, nama bahasa ini diubah menjadi PHP: Hypertext Preprocessor

dengan singkatannya PHP. PHP versi terbaru adalah versi ke-5. Berdasarkan survey Netcraft pada bulan Desember 1999, lebih dari sejuta site menggunakan PHP, di antaranya adalah NASA, Mitsubishi, dan RedHat.

2.6.6 MySql

Menurut Bangsawan (2015), MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: database management system) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus di mana penggunaannya tidak cocok dengan penggunaan GPL. Dalam bahasa SQL pada umumnya informasi tersimpan dalam tabel-tabel yang secara logik merupakan struktur dua dimensi yang terdiri atas baris (row atau record) dan kolom (column atau field). Adapun dalam sebuah database dapat terdiri atas beberapa tabel. Jadi, MySQL tetap menggunakan tabel, baris, dan kolom. Sebuah database dalam MySQL mengandung beberapa tabel dan satu tabel dalam database terdiri atas sejumlah baris dan kolom. MySQL juga merupakan pengolahan database yang populer.

2.7 Metode Perangkat Lunak Yang digunakan

2.7.1 Prototype

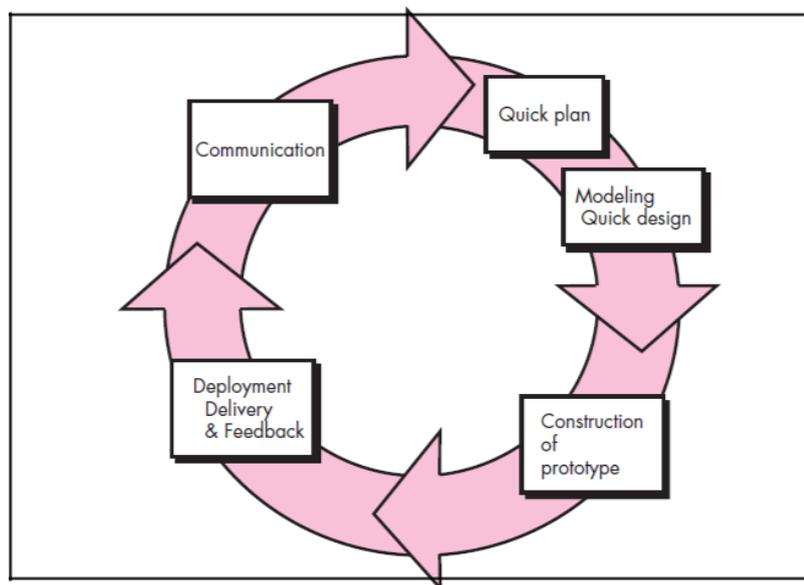
Prototype adalah proses pembuatan model sederhana *software* yang memungkinkan pengguna memiliki gambaran dasar tentang program serta melakukan pengujian awal. *Prototype* memberikan fasilitas bagi pengembang dan pengguna untuk saling berinteraksi selama proses pembuatan, sehingga pengembang dapat dengan mudah memodelkan perangkat lunak yang akan di buat. Metode ini cocok digunakan untuk mengembangkan sebuah perangkat lunak yang dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna. Kemudian membuat sebuah rancangan kilat yang selanjutnya akan dievaluasi kembali sbelum di produksi secara benar *Prototype*, bukanlah merupakan sesuatu yang lengkap, tetapi sesuatu yang harus dievaluasi dan dimodifikasi kembali. Segala perubahan dapat terjadi pada saat *prototype* dibuat untuk memenuhi

kebutuhan pengguna dan saat yang sama memungkinkan pengembangan untuk lebih memahami kebutuhan pengguna secara baik. (Pressman, 2012).

Tahapan-tahapan dalam metode *Prototype* :

1. Komunikasi (*Communication*) : Pengumpulan data awal, yaitu komunikasi dengan klien dan *user* untuk menentukan kebutuhan.
2. Perencanaan Cepat (*Quick Plan*) : Pembuatan perencanaan analisis terhadap kebutuhan pengguna.
3. Pemodelan Perancangan Cepat (*Modeling Quick Design*) : Membuat rancangan desain program.
4. Pembentukan *Prototype* (*Construction of prototype*) : Pembuatan aplikasi berdasarkan dari pemodelan desain yang telah dibuat.
5. Penyerahan Sistem dan Umpan Balik (*Development Delivery and Feedback*) : Memproduksi perangkat secara benar sehingga dapat digunakan oleh pengguna.

Berikut ini merupakan Diagram Alur Prototipe dapat dilihat pada gambar 2.1.



Gambar 2.1 : Diagram Alur Metode Prototype

Pada Gambar 2.1, Tahap pertama ialah *communication* dan pengumpulan data awal yaitu tahap suatu perencanaan yang dilakukan, mulai dari menciptakan dan melaksanakan proses untuk memastikan bahwa perencanaan tersebut berkualitas tinggi, terpercaya,

efisiensi biaya. Tahap kedua adalah *quick plan* yaitu analisis terhadap kebutuhan pengguna. Tahap ketiga adalah *modelling quick design* yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali. Tahap keempat adalah *construction of prototype* adalah pembuatan perangkat *prototype* termasuk pengujian dan penyempurnaan. Tahap kelima adalah *deployment, delivery, and feedback* adalah tahap penyerahan sistem ke pengguna dan umpan balik.

2.8 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem peranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. UML sendiri juga memberikan standar penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. UML sebagai sebuah bahasa yang memberikan tatanan penulisan kata-kata dalam ‘MS Word’ untuk kegunaan komunikasi. Sebuah bahasa model adalah sebuah bahasa yang mempunyai konsep aturan penulisan serta secara fisik mempresentasikan dari sebuah sistem. (Jihad, 2015).

Sedangkan menurut Marini (2019), UML (*Unified Modeling Language*) adalah melakukan sebuah perencanaan berorientasi obyek biasanya menggunakan model yang dikenal dengan *Unified Modeling Language* (UML) yang merupakan sebuah bahasa pemodelan objek standar sebagai ganti dari pendekatan atau metode berorientasi objek standar. *Unified Modeling Language* (UML) adalah satu kumpulan konveksi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek.

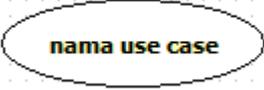
1. *Use Case Diagram*

Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat”. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan use case. (Sukamto dan Shalahuddin, 2013:155).

Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna. Komponen-komponen *use case* dapat dilihat pada tabel 2.1.

Tabel 2.1 Simbol Use Case Diagram

(Sumber: Rosa dan Shalahuddin, 2019)

Simbol	Deksripsi
<p>Use Case</p> 	<p>Fungsionalitas yang disediakan <i>system</i> sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>.</p>
<p>Actor</p> 	<p>Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan di buat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang. biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Asosiasi (Association)</p> 	<p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p>
<p>Ekstensi (Extend)</p> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang di tambahkannya.</p>
<p>Generalisaasi (Generalization)</p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>

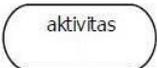
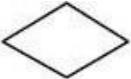
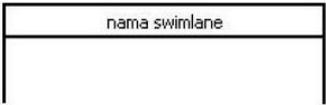
2. Activity Diagram

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. (Sukanto dan Shalahuddin, 2013:161)

Berikut adalah simbol-simbol yang dapat dilihat pada diagram aktivitas table 2.2.

Tabel 2.2 Komponen Activity Diagram
Simbol Activity Diagram

(Sumber: Rosa dan Shalahuddin, 2019)

Simbol	Deksripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

3. Class Diagram

Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. (Sukamto dan Shalahuddin, 2013:141). Penjelasan simbol *class diagram* dapat dilihat pada tabel 2.3.

Tabel 2.3 Simbol Class Diagram
(Sumber: Rosa dan Shalahuddin, 2019)

Simbol	Deksripsi
<p>Kelas</p> 	Kelas pada stuktur sistem.
<p>Antarmuka / <i>Interface</i></p> 	Sama dengan konsep interface dalam pemograman berorientasi objek.
<p>Asosiasi / <i>Association</i></p> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
<p>Asosiasi berarah / <i>Directed Association</i></p> 	Relasi antar kelas dengan makna kelas satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
<p>Generalisasi</p> 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
<p>Kebergantungan / <i>Dependency</i></p> 	Relasi antar kelas dengan makna kebergantungan antarkelas.
<p>Agregasi / <i>Aggregation</i></p> 	Relas antarkelas dengan makna semuabagian (whole-part).

2.9 Pengujian Perangkat Lunak

2.9.1 *Black Box Testing* (Pengujian kotak hitam)

Yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi, masukan, dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Pengujian kotak hitam dilakukan dengan membuat kasus uji yang bersifat mencoba semua fungsi dengan memakai perangkat lunak apakah sesuai dengan spesifikasi yang dibutuhkan. Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah (Rosa dan Shalahuddin, 2019)

2.10 Penelitian Terkait

2.10.1 Rancang Bangun System Informasi Monitoring pada PT. Sukanda Djaya Lampung Selatan Berbasis Website dan SMS Gateway

Pada penelitian (Halimah, Ema, Winda, dan Cornelius, 2018) menghasilkan tentang pembuatan system monitoring untuk mempermudah memonitoring keadaan barang dan penundaan pembayaran atau transaksi oleh customer terhadap pembelian barang melalui hutang dari PT. Sukanda Djaya yang fungsinya juga untuk mengirimkan pesan peringatan melalui SMS Gateway kepada customer yang melakukan pembelian melalui website resmi perusahaan tersebut, sehingga memudahkan dalam penagihan hutang secara tidak langsung kepada customer.

2.10.2 Model Aplikasi Monitoring System Absensi Sidik Jari Sebagai Pendukung Keputusan Untuk Penilaian Kinerja Karyawan

Pada penelitian (Junaidi, Ladyca, dan Adhitya, 2015) menghasilkan tentang pembuatan system monitoring berdasarkan absensi melalui sidik jari karyawan yang hasil datanya akan digunakan sebagai pendukung pengambilan keputusan penilaian kinerja terhadap karyawan dan mengurangi tingkat manipulasi data dari karyawan yang sering melakukan mangkir sehingga mereka menitipkan absensinya ke rekan kerjanya, untuk itu sistem ini dibangun agar data yang digunakan bisa lebih valid.

2.10.3 Rancang Bangun Aplikasi Sistem Informasi Monitoring Absensi Mahasiswa Politeknik Negeri Jakarta Menggunakan Teknologi NFC Pada Android

Pada penelitian (Indri, Irdam, dan Vida, 2015) menghasilkan tentang bagaimana membuat sistem absensi menggunakan media kartu NFC yang diintegrasikan datanya dengan database, sehingga kartu tersebut yang digunakan untuk melakukan absensi sebagai tanda bahwa mereka mahasiswa kampus tersebut. Rancangan aplikasi ini juga digunakan sebagai cara untuk memonitoring mahasiswa tersebut, agar diketahui mahasiswa tersebut mengikuti kegiatan belajar mengajar atau tidak. Monitoring sistem ini dalam bentuk surat peringatan kepada mahasiswa yang tidak masuk atau mengikuti perkuliahan selama kurang lebih 1 minggu full tanpa adanya alasan, dan surat peringatan ini langsung dikirimkan kepada orang tua/wali mahasiswa tersebut agar mendapatkan efek jera, dan dilakukan sanksi sesuai dengan peraturan yang telah ditetapkan.

2.10.4 Perancangan Sistem Informasi Monitoring Progres Pengadaan Barang Pada Bagian Pengadaan Dan Logistik Institut Informatika Dan Bisnis Darmajaya

Pada penelitian (Sutedi, 2011) menghasilkan tentang pembuatan sistem monitoring yang lebih baik dari sistem sebelumnya yang memudahkan dalam input dan output data yang dibutuhkan sehingga dilakukan penelitian tersebut untuk melakukan perubahan terhadap algoritma sistem dan databasenya untuk melakukan monitoring lebih mudah dari sistem sebelumnya yang sudah ada.