



BAB II

TINJAUAN PUSTAKA

Dalam bab ini akan diterapkan mengenai tinjauan pustaka yang bertujuan untuk membahas permasalahan yang diambil. Sesuai judul dalam pembuatan Tugas Akhir, yang akan dibuat penulis terdiri dari pengertian-pengertian sebagai berikut :

2.1 Teori Judul

2.1.1 Aplikasi

Menurut Jogiyanto (1999) aplikasi adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*.

Menurut Rachmad Hakim S, aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (game), dan sebagainya.

Dari defisini diatas, aplikasi ialah suatu perangkat lunak di dalam perangkat computer yang bertugas untuk mengolah data dari proses input menjadi output, bermain game, dan lain-lain.

2.1.2 Monitoring

Monitoring adalah proses pengumpulan dan analisis informasi berdasarkan indikator yang ditetapkan secara sistematis dan kontinu tentang kegiatan program sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program kegiatan itu selanjutnya, Mardiani dalam Aprisa dan Monalisa (2015:50).

Monitoring merupakan program yang terintegrasi, bagian penting dipraktek manajemen yang baik dan arena itu merupakan bagian integral di manajemen sehari-hari, Cassely dan Kumar (1987).

Monitoring adalah mekanisme yang sudah menyatu untuk memeriksa bahwa semua berjalan sesuai direncanakan dan memberi kesempatan agar penyesuaian dapat dilakukan secara metodologis (Oxfam, 1995).

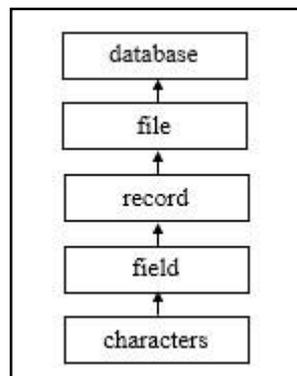


Dari definisi di atas, Monitoring berarti suatu proses dimana informasi dikumpulkan menjadi satu dan dikontrol agar berjalan sesuai yang direncanakan dan terintegrasi sehingga dapat dilakukan tindakan koreksi untuk penyempurnaan program kegiatan itu selanjutnya.

2.2 Teori Khusus

2.2.1 Database

Pengertian database menurut Bambang Hariyanto dalam jurnal Minarni dan Susanti (2014:105) adalah :”kumpulan data (elementer) yang secara logic berkaitan dalam mempresentasikan fenomena/fakta secara terstruktur dalam domain tertentu untuk mendukung aplikasi dalam system tertentu”. Dari definisi diatas maka dapat disimpulkan bahwa database adalah kumpulan dari suatu data yang saling berhubungan satu dengan yang lainnya, yang berdasarkan sebuah skema atau struktur tertentu, yang kelak dapat dimanfaatkan kembali dengan cepat dan mudah. Suatu database terdiri dari *file*, *record*, *field data*, dan *characters*.



Gambar 2.1 Jenjang Data

a. *Characters*

merupakan bagian data yang terkecil, dapat berupa karakter numerik, huruf ataupun karakterkarakter khusus (*special characters*) yang membentuk suatu item data/field.

**b. Field**

merepresentasikan suatu atribut dari record yang menunjukkan suatu item dari data, seperti misalnya nama, alamat dan lain sebagainya. Kumpulan dari field membentuk suatu record.

- *field name* : harus diberi nama untuk membedakan field yang satu dengan lainnya.
- *field representation* : tipe field (karakter, teks, tanggal, angka, dsb), lebar field (ruang maksimum yang dapat diisi dengan karakterkarakter data).
- *field value* : isi dari field untuk masing-masing record.

c. Record

Kumpulan dari field membentuk suatu record. Record menggambarkan suatu unit data individu yang tertentu. Kumpulan dari record membentuk suatu file. Misalnya file personalia, tiap-tiap record dapat mewakili data tiap-tiap karyawan.

d. File

File terdiri dari record-record yang menggambarkan satu kesatuan data yang sejenis. Misalnya file mata pelajaran berisi data tentang semua mata pelajaran yang ada.

e. Database

Kumpulan dari file / table membentuk suatu database.

2.2.2 Data Flow Diagram

Sukamto et. al (2014:70), “*Data Flow Diagram (DFD)* atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

a. Proses (*Process*)

Aktivitas atau fungsi yang dilakukan untuk alasan bisnis yang spesifik, biasa berupa manual maupun terkomputerisasi.



b. Aliran Data (*Data Flow*)

Satu data tunggal atau kumpulan logis suatu data, selalu diawali atau berakhir pada suatu proses.

c. Penyimpanan Data (*Data Store*)

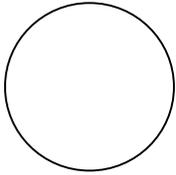
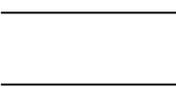
Kumpulan data yang disimpan dengan cara tertentu. Data yang mengalir disimpan dalam *data store*. Aliran data di-*update* atau ditambahkan ke *data store*.

d. Entitas (*Entity*)

Orang, organisasi, atau sistem yang berada di luar sistem tetapi berinteraksi dengan sistem.

Notasi yang digunakan dalam membuat DFD ada empat buah, yaitu sebagai berikut:

Tabel 2.1 Simbol-simbol Data Flow Diagram

| No. | Notasi | Keterangan |
|-----|---|---|
| 1 |  | Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama yang diberikan pada sebuah proses biasanya berupa kata kerja. |
| 2 |  | <i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya |



Lanjutan Tabel 2.1 Simbol-Simbol Data Flow Diagram

| | | |
|---|---|--|
| | | <p>dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel- tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (Entity Relationship Diagram (ERD), Conceptual Data Model (CDM), Physical Data Model (PDM)).</p> <p>Catatan : Nama yang diberikan pada sebuah penyimpanan biasanya berupa kata benda.</p> |
| 3 |  | <p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai atau berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>Catatan:</p> <p>Nama yang diberikan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda.</p> |
| 4 |  | <p>Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).</p> <p>Catatan:</p> <p>Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.</p> |

(Sumber: Sukamto et. al, 2014:71)



Menurut Sukamto et. al (2014:72), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*

DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil *breakdown* DFD Level 0 yang sebelumnya sudah dibuat.

3. Membuat DFD Level 2

Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-*breakdown*.

4. Membuat DFD Level 3 dan seterusnya

DFD Level 3, 4, 5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

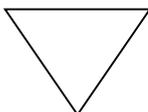
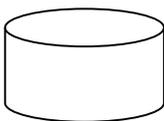
2.2.3 Block Chart

Kristanto (2011:68) menjelaskan bahwa *block chart* berfungsi untuk memodelkan masukan, keluaran, proses, dan transaksi dengan menggunakan simbol-simbol tertentu yang pembuatannya harus memudahkan bagi pemakai dalam memahami alur dari sistem.



Adapun simbol-simbol yang sering digunakan dalam *block chart* adalah sebagai berikut:

Tabel 2.2 Simbol-simbol Block Chart

| No. | Simbol | Arti |
|-----|--|---|
| 1 |  | Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan. |
| 2 |  | Proses yang dilakukan oleh komputer. |
| 3 |  | Menandakan dokumen yang diarsipkan (arsip manual). |
| 4 |  | Data penyimpanan (data storage). |

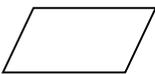
(Sumber: Kristanto, 2011:68)

2.2.4 Flowchart

Sitorus (2015:14), "Flowchart (diagram alir) menggambarkan urutan logika dari suatu prosedur pemecahan masalah, sehingga *flowchart* merupakan langkah-langkah penyelesaian masalah yang dituliskan dalam simbol-simbol tertentu."

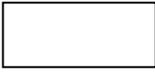
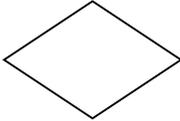
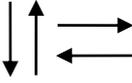
Simbol-simbol yang digunakan untuk menggambarkan *flowchart* dapat dilihat pada tabel berikut:

Tabel 2.3 Simbol-simbol Flowchart

| No. | Simbol | Nama | Fungsi |
|-----|---|----------------------|---|
| 1 |  | <i>Terminal</i> | Menyatakan permulaan atau akhir suatu program. |
| 2 |  | <i>Input/ Output</i> | Meyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya. |



Lanjutan Tabel 2.3 Simbol-Simbol Flowchart

| | | | |
|---|---|-----------------|---|
| 3 |  | <i>Process</i> | Menyatakan suatu tindakan (proses yang dilakukan oleh komputer). |
| 4 |  | <i>Decision</i> | Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya atau tidak. |
| 5 |  | <i>Document</i> | Mencetak keluaran dalam bentuk dokumen (melalui printer) |
| 6 |  | <i>Flow</i> | Menyatakan jalannya arus suatu proses. |

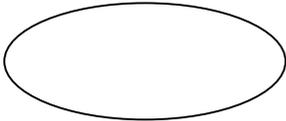
(Sumber: Sitorus, 2015:15)

2.2.5 Entity Relationship Diagram (ERD)

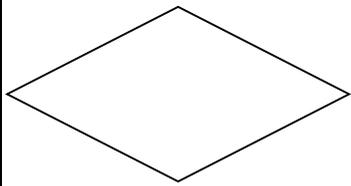
Sukamto et. al (2014:50), “ERD adalah permodelan awal basis data relasional yang dikembangkan berdasarkan teori himpunan dalam.”

Simbol-simbol yang digunakan untuk menggambarkan ERD dapat dilihat pada tabel berikut ini:

Tabel 2.4 Simbol-simbol *Entity Relationship Diagram* (ERD)

| No. | Simbol/Gambar | Keterangan |
|-----|---|---|
| 1 | <i>Entitas</i>  | Orang, tempat, atau benda; memiliki nama tunggal; ditulis dengan huruf besar; dan berisi lebih dari 1 <i>instance</i> . |
| 2 | <i>Attribute</i>  | Properti dari entitas, harus digunakan oleh minimal 1 proses bisnis, dan dipecah dalam <i>detail</i> . |

**Lanjutan Tabel 2.4** Simbol-simbol *Entity Relationship Diagram* (ERD)

| | | |
|---|--|--|
| 3 | Relationship  | Menunjukkan hubungan antar 2 entitas, dideskripsikan dengan kata kerja, memiliki modalitas (<i>null/not null</i>) memiliki kardinalitas (1:1, 1:N, atau M:N) |
|---|--|--|

(Sumber: Sukamto et. al, 2014:50)

2.2.6 Kamus Data (KD) atau Data Dictionary (DD)

Sukamto et. al (2014:73), “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).”

Simbol-simbol yang digunakan dalam kamus data, yaitu:

Tabel 2.5. Simbol-simbol dalam Kamus Data

| No. | Simbol | Keterangan |
|-----|------------------|-------------------------------------|
| 1 | = | disusun atau terdiri dari |
| 2 | + | dan |
| 3 | [] | baik...atau... |
| 4 | { } ⁿ | n kali diulang atau bernilai banyak |
| 5 | () | data opsional |
| 6 | *...* | batas komentar |

(Sumber: Sukamto et. al, 2014:74)

2.3 Teori Program

2.3.1 HTML

Suryana et. al (2014:29), “*HyperText Markup Language* (HTML) adalah bahasa yang digunakan untuk menulis halaman web.”

Sibero (2013:19), “*HyperText Markup Language* atau HTML adalah bahasa yang digunakan pada dokumen web sebagai bahasa untuk pertukaran dokumen web.”



Dari pengertian diatas, dapat disimpulkan bahwa HTML (*HyperText Markup Language*) merupakan bahasa yang sering digunakan untuk membuat halaman web.

Sibero (2013:19) menjelaskan bahwa struktur dokumen HTML sebagai berikut:

```
<html>
  <head>
    <title></title>
  </head>
  <body>
  </body>
</html>
```

2.3.2 PHP (*PHP Hypertext Preprocessor*)



Gambar 2.2. Logo PHP

Sibero (2013:49), “PHP adalah pemrograman *interpreter* yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan. PHP disebut sebagai pemrograman *Server Side Programming*, hal ini dikarenakan seluruh prosesnya dijalankan pada *server*.”

Wahana Komputer (2014:33), “PHP merupakan bahasa berbentuk *script* yang ditempatkan di dalam server baru kemudian diproses. Kemudian hasil

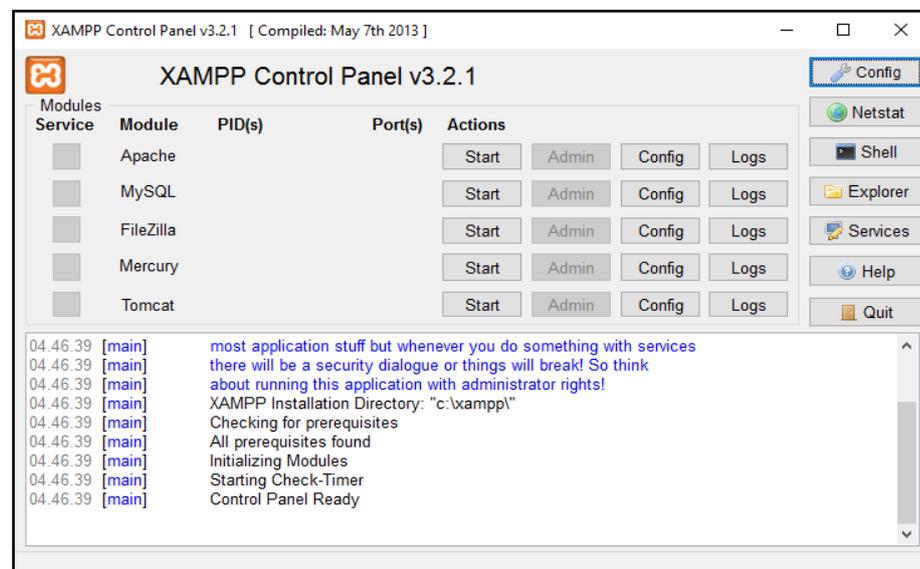


pemrosesan dikirimkan kepada *web browser* klien. Bahasa pemrograman ini dirancang khusus untuk membentuk web dinamis.”

Dari kedua pengertian di atas, dapat disimpulkan bahwa PHP adalah bahasa pemrograman web yang mampu menerjemahkan kode dan memproses sehingga menghasilkan tampilan website.

2.3.3 Pengertian XAMPP

Wahana Komputer (2014:72), “XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP, dan Perl. XAMPP adalah *tool* yang menyediakan paket perangkat lunak dalam satu buah paket.”



Gambar 2.3. Tampilan XAMPP Control Panel

2.3.4 MySQL

Sibero (2013:97) menjelaskan bahwa MySQL atau dibaca “*My Sekuel*” adalah suatu RDBMS (*Relational Database Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data. Berikut merupakan tabel beberapa pernyataan yang ada pada MySQL:

**Tabel 2.6.** Pernyataan MySQL

| | | | |
|---|--------|---------------------------------------|--|
| 1 | INSERT | Untuk memasukkan data ke dalam tabel. | INSERT INTO nama_tabel (kolom_1, kolom_2, ... kolom_n) VALUES ('nilai_kolom_1', 'nilai_kolom_2', ..., 'nilai_kolom_n'); |
| 2 | SELECT | Untuk melihat isi tabel. | SELECT nama_kolom FROM nama_tabel; |
| 3 | UPDATE | Untuk mengubah data dalam tabel. | UPDATE nama_tabel SET kolom_1 = nilai_baru_1, kolom_2 = nilai_baru_2, ... kolom_n = nilai_baru_n WHERE kondisi; |
| 4 | DELETE | Untuk menghapus data dalam tabel. | DELETE FROM nama_tabel WHERE kondisi; |

Sibero (2013:71) menjelaskan fungsi-fungsi PHP MySQL adalah sebagai berikut:

Tabel 2.7. Fungsi-fungsi PHP MySQL

| No. | Fungsi MySQL | Penggunaan |
|-----|--------------------------|---|
| 1 | <i>mysql_connect</i> | Membuka koneksi <i>database</i> . |
| 2 | <i>mysql_error</i> | Mengambil pesan kesalahan. |
| 3 | <i>mysql_fetch_array</i> | Mengambil hasil record dan mendefinisikan dalam bentuk array. |
| 4 | <i>mysql_num_rows</i> | Mengambil jumlah baris data hasil query. |
| 5 | <i>mysql_query</i> | Mengirimkan perintah query pada server. |
| 6 | <i>mysql_select_db</i> | Memilih dan mengaktifkan database. |



2.3.5 CSS (*Cascading Style Sheets*)

Sibero (2013:), “*Cascading Style Sheet (CSS)* dikembangkan untuk menata gaya pengaturan halaman web.”

Suryana (2014:101), “CSS (*Cascading Style Sheet*) adalah suatu bahasa *stylesheet* yang digunakan untuk mengatur tampilan suatu *website*, baik tata letaknya, jenis huruf, warna, dan semua yang berhubungan dengan tampilan.”

Wahana Komputer (2014:5) menjelaskan bahwa CSS merupakan bahasa pemrograman yang khusus menangani tampilan tiap elemen di dalam dokumen HTML. Dengan memanfaatkan CSS, struktur kode HTML kita akan terlihat rapi dan terstruktur.”

Dari ketiga kesimpulan diatas, maka CSS sendiri dapat diartikan sebagai suatu bahasa pemrograman yang khusus untuk menata atau menangani tampilan suatu web baik itu jenis huruf, warna, dan *style* web.

2.4 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *Agile Development*. *Agile Development* sendiri merupakan konsep yang memegang nilai-nilai berikut, sebagaimana disebutkan dalam Agile Manifesto (Moerira, dkk : 2010 : 7) :

1. Individu-individu dan interaksi-interaksi dibanding proses-proses dan alat-alat.
2. Software yang berjalan dibanding dokumentasi komprehensif
3. Kolaborasi dengan konsumen dibanding negosiasi kontrak.
4. Merespon perubahan dibanding mengikuti rencana.

Sedangkan tahapan *iterative* dari *agile development* ialah :

1. Perencanaan (*planning*), pada tahap ini pengembang dan klien membuat rencana tentang kebutuhan dari perangkat lunak yang akan dibuat, melakukan rencana untuk konsep yang dipakai dan kebutuhan apa saja yang dibutuhkan dalam perencanaan tersebut.
2. *Requirements Analysis*, pada tahap ini, dilakukan analisis kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian,



- wawancara, atau *study literature*. Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Dokumen inilah yang akan menjadi acuan sistem analis untuk menerjemahkan ke dalam bahasa pemrograman.
3. Desain, tahap dari proses desain akan menerjemahkan kebutuhan user ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum tahap coding. Proses ini berfokus pada struktur database berupa kamus data dan tabel data, perancangan sistem (DFD), dan desain tampilan (rancangan aplikasi).
 4. Teknik (*engineering*), pada tahap ini, dilakukan pengembangan prototipe yang mencakup seluruh kebutuhan aplikasi monitoring peralatan pada PT. Sucofindo cabang Palembang, Lahat dan Bangka.
 5. Evaluasi (*evaluation*), pada tahap ini, dilakukan penilaian dan pengujian untuk mengembangkan tujuan dari aplikasi monitoring peralatan pada PT. Sucofindo cabang Palembang, Lahat dan Bangka untuk proses iterasi selanjutnya.

Agile sendiri merupakan gabungan dari metode *incremental* (proses bertahap) dan *iterative* (proses berulang). Agile berproses iteratif dengan perputaran yang pendek. Kebutuhan direncanakan, diimplementasikan, diuji, dan dievaluasi secara berulang dalam hitungan minggu. Implikasi dari pemakaian *Agile Software Development* adalah kemampuan adaptasi yang tinggi terhadap perubahan selama proses pembangunan software. Sebagai sebuah konsep, Agile mempunyai bermacam model, termasuk didalamnya *scrum*, *agile modelling*, XP, dan lain-lain. Didalam penelitian ini, akan digunakan metode Scrum. Walaupun sebenarnya metode ini biasanya digunakan secara tim, tapi tidak ada larangan untuk menggunakannya secara perorangan. Metode Scrum membagi proses development menjadi beberapa *Sprint* (periode pembangunan sistem). *Sprint* ini akan bertindak sebagai *milestone* (acuan) dalam pembuatan software. Adapun tahapan yang terdapat didalam Scrum adalah :

1. Perancangan *Product Backlog*: merupakan proses merancang pekerjaan-pekerjaan yang akan dilakukan berdasarkan permintaan *user*.



2. Memindahkan item dari *Product Backlog* ke *Sprint-Sprint* : Memasukan pekerjaan-pekerjaan yang harus dilakukan dan menyusun prioritasnya ke dalam sprint-srint.

2.5 Penelitian Terdahulu

Tri (2013) melakukan penelitian tentang sistem monitoring data asset dan inventaris PT. Telkom Cianjur berbasis web, bahwa pengelolaan perangkat yang ada di PT Telkom Cianjur merupakan hal yang sangat penting untuk dilakukan, karena perangkat merupakan salah satu aset perusahaan, sehingga data inventaris atas aset tersebut harus selalu dijaga agar dapat memberikan informasi yang jelas dan memudahkan dalam melakukan perawatan. Maka dari itu dibutuhkan sebuah sistem yang bisa memonitoring data asset agar data dapat terjaga keakuratan dan kejelasan informasinya sehingga proses rekapitulasi data dan pelaporan data dapat berjalan dengan baik.

Lucky, Kamil, dan Setiawan (2016) melakukan penelitian tentang aplikasi monitoring perangkat keras menggunakan Windows Query language bahwa dokumentasi aset merupakan salah satu agenda rutin yang harus dilakukan sebagai bagian dari sistem manajemen mutu. Dokumen ini dapat memiliki fungsi sebagai alat komunikasi, alat bukti dan juga knowledge sharing. Dibutuhkan sebuah sistem monitoring yang dapat memudahkan pelaksana dokumentasi untuk mendapatkan informasi yang dibutuhkan secara cepat, akurat dan real-time menggunakan aplikasi yang terintegrasi dengan sistem yang dibangun di server.

Rosa dan Afriyan (2016) melakukan penelitian tentang Pengembangan perangkat lunak sistem informasi monitoring mahasiswa bidikmisi berbasis web menggunakan pendekatan DSDM, bahwa Pengembangan sistem monitoring dan evaluasi berbasis web diperlukan untuk mengatasi kendala dan kekurangan dalam monitoring dan evaluasi akademik dan non akademik mahasiswa Bidik Misi sehingga tugas dan tanggung jawab perguruan tinggi sebagai pengelola dapat menjadi lebih optimal.



Sya, Sudiro, dan Tri (2016) mengatakan dalam penelitiannya bertujuan untuk membuat aplikasi Pemantauan yang berfungsi sebagai perantara usulan perbaikan dari ruangan pelayanan kesehatan. Aplikasi ini akan membantu petugas dalam melakukan pemantauan terhadap alat yang rusak, dipinjam atau yang dihapus.

Suryanti dan Welda (2017) melakukan penelitian tentang sistem monitoring penanganan kerusakan peralatan elektronik di STMIK STIKOM Indonesia saat ini dilaksanakan dengan melakukan pencatatan di buku besar. Kendala yang dihadapi adalah sering terjadinya kesimpang siuran informasi peralatan yang sudah diperbaiki dan yang belum diperbaiki. Teknisi yang telah memperbaiki kadang tidak melaporkan ke Bagian Rumah Tangga hasil dari perbaikan tersebut. Hal ini menyebabkan Bagian Rumah Tangga, BAA, dan PK II tidak tahu informasi mengenai kondisi peralatan saat itu sehingga pelayanan STMIK STIKOM Indonesia terhadap dosen dan mahasiswa menjadi kurang efektif.