

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Penelitian Terkait**

Penelitian terkait bertujuan untuk mengambil beberapa referensi jurnal terkait yang digunakan dalam mendukung penelitian publikasi ilmiah dalam jurnal lokal.

1. Penelitian yang dilakukan oleh Maria Rosario Borroek, SE, M.S.I (2014) dengan judul Perancangan Sistem Informasi Manajemen Aset Pada Stikom Dinamika Bangsa Jambi (Studi Kasus: Penjualan dan Disposasi Aset Tetap). Pengelolaan aset pada STIKOM Dinamika Bangsa masih belum optimal, salah satu cara agar pengelolaan dapat lebih optimal adalah dengan penggunaan system informasi. Aset yang sudah tidak optimal dalam fungsinya biasanya akan dilakukan penjualan atau dihapus. Penelitian ini menghasilkan prototype system yang menggambarkan fungsional penjualan dan penghapusan aset pada STIKOM Dinamika Bangsa Jambi sedangkan model pengembangan system yang digunakan adalah Unified Modeling Language yakni use case diagram, class diagram dan activity diagram
2. Penelitian yang dilakukan oleh Hans Saputra (2016) dengan judul Perancangan Sistem Informasi Manajemen Aset Komputer Pada Perusahaan Jasa Konstruksi. Aset dalam catatan akuntansi dibagi menjadi dua bagian termasuk aset tetap dan bukan aset tetap. Dalam hal ini aset komputer termasuk aset tetap perusahaan. Yang memiliki masa pakai lebih dari satu

tahun, dan bisa dijual kembali. Dalam perancangan sistem informasi manajemen aset komputer menggunakan wawancara dengan departemen teknologi informasi yang mengelola aset komputer. Informasi yang diperoleh menggunakan analisis tradisional Unified Modeling Language. Hasil penelitian ini menyimpulkan bahwa dengan bantuan sistem informasi manajemen, data aset komputer terjaga lebih akurat dan mengurangi kesalahan manusia dalam pelaporan. Bagian teknologi informasi sisi lain dapat dengan cepat mencari dan menyajikan data yang dibutuhkan oleh manajemen dalam mengambil keputusan.

3. Penelitian yang dilakukan oleh Y Maryono (2010) dengan judul Analisis Dan Perancangan Sistem Informasi Manajemen Aset TIK Studi Kasus: Asmi Santa Maria Yogyakarta. Sistem informasi ini telah berhasil dirancang dan menyediakan fungsionalitas pengelolaan informasi aset TIK yang meliputi registrasi aset, penempatan aset, pemindahan aset, penghitungan depresiasi, penilaian aset, pencatatan maintenance, penghapusan aset, pelacakan aset, dan pembuatan laporan. Perancangan Sistem informasi manajemen aset TIK ini dilakukan dengan berbasis web (intranet) dan dengan pendekatan berorientasi objek (OOA).
4. Penelitian yang dilakukan oleh Irma Yunita (2017) dengan judul Analisis Dan Perancangan Sistem Informasi Manajemen Aset Pada Smk Negeri 4 Kota Jambi. Penelitian ini menghasilkan *prototype system* yang menggambarkan fungsional perencanaan, pengadaan, penerimaan dan penempatan aset pada SMK N 4 Kota Jambi. Metode yang digunakan adalah pendekatan berorientasi

objek menggunakan UML (unified modelling language) yakni use case diagram, class diagram dan activity diagram. Penelitian ini menghasilkan sebuah prototype yang dapat diimplementasikan lebih lanjut sehingga menghasilkan sistem informasi manajemen aset yang dapat diimplementasikan pada SMK N 4 Kota Jambi.

5. Penelitian yang dilakukan oleh Maria Rosario Borroek (2014) dengan judul Perancangan Sistem Informasi Manajemen Aset Pada Stikom Dinamika Bangsa Jambi (Studi Kasus: Penjualan dan Disposals Aset Tetap). Penelitian ini menghasilkan prototype system yang menggambarkan fungsional penjualan dan penghapusan asset pada STIKOM Dinamika Bangsa Jambi sedangkan model pengembangan system yang digunakan adalah Unified Modeling Language yakni use case diagram, class diagram dan activity diagram

## **II.2. Landasan Teori**

### **II.2.1. Sistem**

Sistem merupakan suatu kumpulan dari komponen-komponen yang membentuk satu kesatuan. Sebuah organisasi dan sistem informasi adalah sistem fisik dan sosial yang ditata sedemikian rupa untuk mencapai tujuan tertentu. Seorang manajer yang mengorganisasikan pekerjaan unitnya adalah orang yang menciptakan suatu sistem pengembangan personal *computer* (PC) menciptakan suatu sistem yang terdiri dari perangkat keras dan lunak PC ini menjadi sub sistem dari suatu perangkat konferensi elektronik. Sistem fisik dan sosial adalah sistem

yang abstrak (*abstract system*) dari konsep dan ide. Contoh sistem yang abstrak ialah pengembangan daur hidup atau pengembangan sistem perangkat lunak (*Tyoso ; 2016 : 1*).

### **II.2.2. Informasi**

Pengertian informasi masih bersifat kontradiktif dan belum mempunyai ciri-ciri khusus. Maka para pakar teori atau ilmu pengetahuan informasi belum dapat membuat satu definisi yang jelas dan lengkap mengenai apa yang disebut informasi.

Menurut para pakar informasi, informasi meliputi aspek abstrak dan khusus. Informasi dipandang sebagai ilmu pengetahuan tertulis atau yang disampaikan secara lisan dan sebagai hasil dari data yang diolah, biasanya diolah secara formal. Ada pula yang menyebut informasi sebagai sarana pengurang ketidakpastian. Pengertian ini sama dengan arti informasi dalam istilah ekonomi, yakni informasi adalah penghalau ketidakpastian, misalnya di dalam pasar, *consumer preference*, dan harga (*Tyoso ; 2016 : 31*).

### **II.2.3. Sistem Informasi Akuntansi**

Sistem informasi akuntansi (SIA) merupakan suatu kerangka pengkoordinasian sumber daya (data, meterial, *equipment, suppliers, personal, and funds*) untuk mengkonversi input berupa data ekonomik menjadi keluaran berupa informasi keuangan yang digunakan untuk melaksanakan kegiatan suatu entitas dan menyediakan informasi akuntansi bagi pihak-pihak yang berkepentingan.

Transaksi memungkinkan perusahaan melakukan operasi, menyelenggarakan arsip dan catatan yang *up to date*, dan mencerminkan aktivitas organisasi. Transaksi akuntansi merupakan transaksi pertukaran yang mempunyai nilai ekonomis. Tipe transaksi dasar adalah :

1. Penjualan produk atau jasa,
2. Pembelian bahan baku, barang dagangan, jasa, dan aset tetap dari supplier,
3. Penerimaan kas,
4. Pengeluaran kas kepada supplier,
5. Pengeluaran kas gaji karyawan. Sebagai pengolah transaksi, sistem informasi akuntansi berperan mengatur dan mengoperasionalkan semua aktivitas transaksi perusahaan (*Dwi Fitri Khasanah ; 2013 : 1*).

Sistem informasi akuntansi (SIA) merupakan gabungan dari tiga kata, yaitu sistem, informasi dan akuntansi. Sistem adalah serangkaian prosedur formal untuk mencapai suatu tujuan. Informasi adalah sekumpulan data yang telah diproses dan didistribusikan ke pemakai (*users*). Sistem informasi akuntansi terdiri dari tiga subsistem utama, yaitu :

- (1) Sistem pemrosesan transaksi (*Transaction processing systems*)
- (2) Sistem pelaporan buku besar/keuangan (*General ledger/financial reporting systems*) dan
- (3) Sistem pengendalian manajemen (*David Efendi ; 2013 : 73*)

#### II.2.4. Manajemen Aset

Manajemen merupakan suatu disiplin ilmu yang digunakan dalam berbagai bidang kegiatan. pengertian manajemen lainnya adalah Organisasi dan koordinasi kegiatan perusahaan sesuai dengan kebijakan tertentu dan dalam pencapaian tujuan yang ditetapkan. Manajemen sering dimasukkan sebagai faktor produksi bersama dengan mesin, bahan, dan uang (Saputra, 2016 : 2)

Aset adalah sumber daya yang dikuasai oleh perusahaan sebagai akibat dari peristiwa masa lalu dan dari manfaat ekonomi di masa depan diharapkan akan diperoleh perusahaan (Borroek, 2014 : 3).

Aset tetap adalah aset berwujud yang dimiliki untuk digunakan dalam produksi atau penyediaan barang dan jasa untuk direntalkan kepada pihak lain, atau untuk tujuan administrative dan diharapkan untuk digunakan selama lebih dari satu periode (Saputra, 2016 : 2).

#### II.2.5. PHP

PHP (*Personal Home Page*) adalah pemrograman (*interpreter*) adalah proses penerjemahan baris sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan. Definisi lain mengenai PHP adalah skrip bersifat *server-side* yang ditambahkan ke dalam HTML. PHP sendiri merupakan singkatan dari *Personal Home Page Tools*. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan ke dalam HTML sehingga suatu halaman web tidak lagi bersifat statis, namun menjadi bersifat dinamis. Berdasarkan beberapa pengertian PHP tersebut dapat disimpulkan bahwa PHP (*Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris

kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan ke dalam HTML (*Hypertext Markup Language*) (Supono : 2016 : 3)

### II.2.6. MySQL

MySQL merupakan salah satu sistem database yang sangat handal karena menggunakan sistem SQL (*Structured Query Language*). Pada awalnya SQL (*Structured Query Language*) berfungsi sebagai bahasa penghubung antara program *database* dengan bahasa pemrograman yang kita gunakan. Dengan adanya SQL (*Structured Query Language*) maka para pemrogram jaringan dan aplikasi tidak mengalami kesulitan sama sekali di dalam menghubungkan aplikasi yang mereka buat. Setelah itu SQL (*Structured Query Language*) dikembangkan lagi menjadi sistem database dengan munculnya MySQL. MySQL merupakan *database* yang sangat cepat, beberapa *user* dapat menggunakan secara bersamaan dan lebih lengkap dari SQL (*Structured Query Language*). MySQL merupakan salah satu *software* gratis yang dapat di-download melalui situsnya. MySQL merupakan sistem manajemen *database*, relasional sistem *database* dan *software open source* (Stendy B Sakur ; 2015 : 57).

### II.2.7. Pengertian ERD

ERD adalah suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. ERD juga merupakan gambaran

yang merelasikan antara objek yang satu dengan objek yang lain dari objek di dunia nyata yang sering dikenal dengan hubungan antar entitas. Sebagai contoh jika membuat ERD dari sistem perpustakaan maka bahan sebagai objek ERD bisa berupa anggota, buku, peminjam, pengembalian dan sebagainya. ERD terdiri dari 4 komponen utama, yaitu (Robi Yanto ; 2016 : 32) :

**Tabel II.1. Simbol ERD**

Notasi	Keterangan
	<b>Entitas</b> , adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.
	<b>Relasi</b> , menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.
	<b>Atribut</b> , berfungsi mendeskripsikan karakter entitas (atribut yg berfungsi sebagai key diberi garis bawah)

(Sumber : Robi Yanto ; 2016 : 32)

### II.2.8. Pengertian Kamus Data

Kamus data umumnya berguna pada pengembangan model sistem dan dapat digunakan untuk menangani semua informasi dari semua tipe model sistem. Kamus data sederhana adalah daftar alfabetis dari nama-nama termasuk pada berbagai model sistem. Seperti namanya, kamus harus mencakup deskripsi yang berhubungan dengan entitas bernama tersebut dan, jika nama itu merepresentasikan objek komposit, mungkin saja ada deskripsi mengenai komposisinya. Informasi lain seperti tanggal pembuatan, pembuatnya dan representasi entitas juga dapat dimasukkan, tergantung pada tipe model yang sedang dikembangkan. Keuntungan penggunaan kamus data adalah :

1. Kamus data merupakan mekanisme untuk manajemen nama. Banyak orang yang harus menciptakan nama untuk entitas dan relasi ketika mengembangkan model sistem yang besar. Nama-nama ini harus dipakai secara konsisten dan tidak boleh bentrok. Perangkat lunak kamus data dapat memeriksa keunikan nama dan memberitahu analisis persyaratan sekiranya terjadi duplikasi nama.
2. Kamus data berfungsi sebagai tempat penyimpanan informasi organisasional yang dapat menghubungkan analisis, desain, implementasi dan evolusi. Sementara sistem dikembangkan, informasi diambil untuk memberitahu perkembangan. Informasi baru ditambahkan pada sistem. Semua informasi mengenai entitas berada pada satu tempat (*Ian Sommerville ; 2012 : 151*).

**Tabel II.1 Entri Kamus Data**

Nama	Keterangan	Tipe	Tanggal
Has-labels	Relasi 1 N antara entitas bertipe node atau link dan entitas bertipe label	Relasi	5.10.1998
Label	Berisi informasi terstruktur atau tidak terstruktur mengenai node atau link. Label dipresentasikan oleh ikon (yang bisa berupa kotak transparan) dan teks yang berhubungan	Entitas	8.10.1998
Link	Relasi 1 : 1 antara entitas desain yang dipresentasikan sebagai node. Link diketikkan dan dapat diberi nama	Relasi	8.10.1998
Name (Label)	Setiap label memiliki nama yang mengidentifikasi tipe label. Nama harus unik dalam set tipe label yang dipakai pada desain	Atribut	8.10.1998
Name (Node)	Setiap node harus memiliki nama yang unik di dalam sebuah desain. Panjang nama bisa mencapai 64 karakter.	Atribut	15.10.1998

*(Sumber : Ian Sommerville ; 2012 : 151)*

## II.2.9. Pengertian Normalisasi

Normalisasi adalah suatu proses untuk membuat data yang tidak normal menjadi data yang normal. Bentuk data yang tidak normal / data mentah biasa disebut juga *unnormalized form*. Masing – masing level normalisasi mempunyai aturan tersendiri.

### 1. *First Normal Form*

Suatu tabel dikatakan dalam keadaan *first normal form* (1NF) jika :

- a. Tidak ada perulangan *record* data dalam tabel.
- b. Setiap sel memiliki satu nilai saja. Artinya tidak ada perulangan *group* dan *array*.
- c. Data yang diinputkan memiliki tipe data yang sama dengan tipe data kolom dalam tabel.

### 2. *Second Normal Form*

Suatu tabel dikatakan dalam keadaan *Second Normal Form* (2NF) jika tabel tersebut sudah dalam keadaan *First Normal Form* (1NF) dan jika semua atribut yang bukan kunci tabel, baik *primary key* maupun *foreign key* tergantung pada semua kunci dalam tabel.

### 3. *Third Normal Form*

Suatu tabel dikatakan dalam keadaan *third normal form* (3NF) jika tabel tersebut sudah dalam keadaan *second normal form* (2NF) dan jika tidak terdapat ketergantungan yang transitif. Artinya, data-data yang mungkin diisi berulang-ulang dapat dibuat sebuah tabel baru.

#### 4. *Boyce-Codd Normal Form (BCNF)*

Tabel dikatakan dalam keadaan *boyce-codd normal form* (BCNF) jika tabel tersebut dalam keadaan *third normal form* (3NF) dan setiap determinan adalah kunci kandidat.

#### 5. *Fourth Normal Form (4NF)*

Suatu tabel dikatakan dalam keadaan *fourth normal form* (4NF) jika tabel tersebut dalam keadaan *boyce-codd normal form* (BCNF) dan jika tidak terdapat ketergantungan nilai ganda.

#### 6. *Fiveth Normal Form (5NF)*

Tabel dikatakan dalam keadaan *Fiveth Normal Form* (5NF) jika tabel tersebut dalam keadaan *fourth normal form* (4NF) dan jika setiap ketergantungan dalam join ada pada tabel sudah konsekuen dengan kunci kandidat pada tabel tersebut (*Ema Utami ; 2012 : 73-76*).

### **II.2.9. UML (*Unified Modeling Language*)**

Menurut *Gata (2013 : 4)* Hasil pemodelan pada OOAD (*Object Oriented Analysis Design*) terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

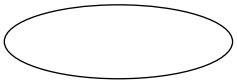
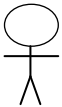
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML



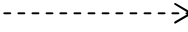
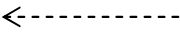
saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

### 1. *Use case* Diagram

*Use case* diagram 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. Dapat dikatakan *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam *use case* diagram, yaitu :

**Tabel II.2. Simbol *Use Case***

Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>use case</i>.</p>
	<p>Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i>, tetapi tidak memiliki control terhadap <i>use case</i>.</p>




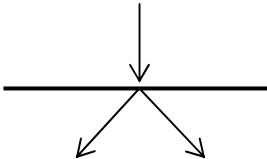
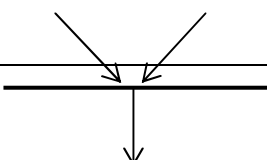
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain ( <i>required</i> ) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

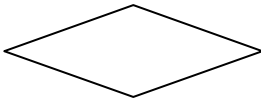

(Sumber : Gata, 2013 : 4)

## 2. Diagram Aktivitas (*Activity Diagram*)

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu :

**Tabel II.3. Simbol *Activity Diagram***

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.

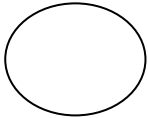
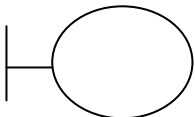
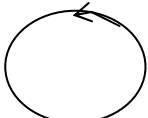
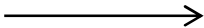
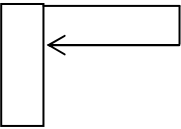
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> , <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.



(Sumber : Gata, 2013 : 6)

### 3. Diagram Urutan (*Sequence Diagram*)

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Simbol-simbol yang digunakan dalam *sequence diagram*, yaitu :

**Tabel II.4. Simbol *Sequence Diagram***

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.

	<p><i>Activation, activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Gata, 2013 : 7)

#### 4. *Class Diagram* (Diagram Kelas)

Merupakan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan dan tanggung jawab entitas yang menentukan perilaku sistem.

*Class diagram* juga menunjukkan atribut-atribut dan operasi-operasi dari sebuah kelas dan *constraint* yang berhubungan dengan objek yang dikoneksikan. *Class diagram* secara khas meliputi: Kelas (*Class*), Relasi, *Associations*, *Generalization* dan *Aggregation*, Atribut (*Attributes*), Operasi (*Operations/Method*), *Visibility*, tingkat akses objek eksternal kepada suatu operasi atau atribut. Hubungan antar kelas mempunyai keterangan yang disebut dengan *multiplicity* atau kardinaliti.

**Tabel II.5. *Multiplicity Class Diagram***

<b>Multiplicity</b>	<b>Penjelasan</b>
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Gata, 2013 : 9)