

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

A. Defenisi Sistem

Menurut kutipan dari Sutabri[1] “sebuah sistem terdiri atas bagian-bagian atau komponen yang terpadu untuk satu tujuan”. Model dasar dari suatu sistem adalah adanya masukan, pengolahan dan keluaran. Akan tetapi, sistem ini dapat dikembangkan hingga menyetakan media penyimpanan. Sistem dapat terbuka dan tertutup akan tetapi sistem informasi biasanya adalah sistem terbuka. Artinya sistem tersebut dapat menerima beberapa dari lingkungan luarnya.

B. Karakteristik Sistem

Sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik menurut Sutabri[1] adalah:

1. Komponen/elemen (*Component*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

2. Batas Sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan.

3. Lingkungan Sistem (*Environment*)

Apapun yang diluar batas sistem yang mempengaruhi operasional sistem.

4. **Penghubung Sistem (*Interface*)**

Merupakan media penghubung antara satu sub sistem dengan sub sistem lainnya.

5. **Masukan Sistem (*Input*)**

Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintance input*) dan sinyal (*signal input*).

6. **Keluaran Sistem (*Output*)**

Hasil energi yang diolah dan diklasifikasi menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai pengambilan keputusan atau hal-hal lain yang menjadi input bagi subsistem lain.

7. **Pengolahan sistem (*Process*)**

Suatu sistem dapat mempunyai proses yang akan mengubah masukan menjadi keluaran.

8. **Sasaran sistem**

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Kalau suatu sistem memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

C. Klasifikasi Sistem

Sistem merupakan suatu bentuk integritas antara satu komponen dengan komponen lain, karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam sistem tersebut. Oleh karena itu menurut Sutabri[1] “ sistem dapat diklasifikasikan dari beberapa sudut pandang”.

a. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem teologia, yaitu sistem yang berupa pemikiran hubungan antara manusia dengan Tuhan, sedangkan sistem fisik merupakan sistem yang ada secara fisik, misalnya sistem komputer, sistem produksi, sistem penjualan, sistem administrasi personalia dan lain sebagainya.

b. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang dan malam, pergantian musim. Sedangkan sistem buatan manusia adalah sistem yang melibatkan interaksi manusia dengan mesin yang disebut *human machine system*. Sistem informasi berbasis komputer merupakan contoh *human machine system* karena menyangkut penggunaan komputer yang berinteraksi dengan manusia.

c. Sistem determinasi dan sistem probabilistik

d. Sistem yang beroperasi dengan tingkah laku yang dapat diprediksi disebut sistem *deterministic*. Sistem komputer adalah contoh dari sistem yang tingkahlakunya dapat dipastikan berdasarkan program-program komputer yang dijalankan, sedangkan sistem yang bersifat probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur *probalistic*.

e. Sistem terbuka dan sistem tertutup

Sistem tertutup merupakan sistem yang berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa campur tangan pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.2 Konsep Dasar Informasi

A. Definisi Informasi

Menurut Sutanta[2] informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimaannya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang.

Menurut Sutabri[1] mengatakan bahwa kriteria kualitas informasi adalah :

a. Akurat

yang berarti informasi harus tidak bias atau menyesatkan dan bebas dari kesalahan

b. Tepat waktu

yang berarti informasi yang sampai kepada penerima tidak boleh terlambat. Mahalnya nilai informasi saat ini adalah karena harus cepatnya informasi tersebut didapatkan, sehingga diperlukan teknologi muktahir untuk mendapatkan, mengolah dan mengirimkannya.

c. Relevan

yang berarti informasi harus mempunyai manfaat bagi pihak yang menerimanya.

Sedangkan menurut Sutabri[1] dalam bukunya yang berjudul "Analisis Sistem Informasi", mengungkapkan "Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajeral dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar tertentu".

B. Komponen dan Jenis Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok penggunaan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data, blok kendali. Sebagai suatu system, keenam blok tersebut masing-masing saling berinteraksi data dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran. Berikut ini adalah komponen-komponen menurut Sutabri[1]

a. Blok Masukan (*Input Block*)

Input mewakili data yang masuk kedalam sistem informasi. Input yang dimaksud adalah metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

b. Blok Model (*Model Block*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan dibasis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

c. Blok Keluaran (*Out Block*)

Produk dari system informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai system.

d. Blok teknologi (*Tecnology Block*)

Teknologi merupakan "*Tool Box*" dalam system informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari system secara keseluruhan. Teknologi terdiri dari 3 (tiga) bagian utama, yaitu teknisi (*Brainware*), perangkat lunak (*Software*) dan perangkat keras (*Hardware*).

e. Blok Basis Data (Database block)

Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu sama lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

f. Blok Kendali (Control Block)

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan, kegagalan sistem itu sendiri, ketidak efisien, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

Secara teori komputer-komputer tidak harus digunakan didalam sistem tetapi tidaklah mungkin suatu sistem yang kompleks dapat berfungsi melibatkan elemen non komputer dan elemen komputer, berikut tipe sistem informasi menurut Sutabri[1] diantaranya:

- 1) Sistem informasi akuntansi
- 2) Sistem informasi pemasaran
- 3) Sistem informasi manajemen persediaan
- 4) Sistem informasi personalia
- 5) Sistem informasi distribusi

2.3 Kependudukan

No	Kata Kunci	Definisi
1	Penduduk	Menurut Undang-Undang Republik Indonesia Nomor 23 Tahun 2006 Tentang Administrasi Kependudukan menyatakan, Penduduk adalah warga Negara Indonesia dan orang asing yang bertempat tinggal di Indonesia.
2	Kelahiran	Menurut Peraturan Daerah Tingkat II Nomor 1 Tahun 1978 Kelahiran adalah proses penambahan jumlah penduduk yang diakibatkan oleh hal alami dari suatu perkawinan antara ayah dan ibu.
3	Kematian	Menurut Peraturan Daerah Tingkat II Nomor 1 Tahun 1978 Kematian adalah proses pengurangan jumlah penduduk yang diakibatkan oleh meninggalnya penduduk.
4	Pindah/ Datang	Menurut Peraturan Daerah Tingkat II Nomor 1 Tahun 1978 Penduduk pindah/datang adalah proses penambahan penduduk dari satu tempat ke tempat lain.

2.4 Pengertian Analisa Berorientasi Objek

Menurut Sutopo[3] dalam bukunya yang berjudul “Analisis Desain Berorientasi Object”, mendefinisikan Analisa sitem adalah proses menentukan kebutuhan sitem apa yang harus dilakukan sistem untuk memenuhi kebutuhan klien, bukanlah bagaimana sistem tersebut diimplementasikan.

Menurut Indrajani[4] dalam bukunya yang berjudul “Perancangan Basis Data dalam All in One”, mengatakan bahwa analisa berorientasi obyek terdiri dari:

A. UML (*Unified Modelling Language*)

UML (*Unifed Modelling Language*) adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadikma (berorientasi objek).

Pemodelan atau modelling sesungguhnya digunakan untuk penyederhanaan permasalahan – permasalahan yang konplek sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

Berdasarkan pendapat yang dikemukakan diatas dapat ditarik kesimpulan bahwa (*Unified Modelling Language*) atau (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasikan, menspesifikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis objek (*Object Oriented Programming*). Pendekatan – pendekatan yang dipakai dalam analisa berorientasi objek antara lain :

a. Pendekatan *top down*

Pendekatan *tip down* yaitu memecahkan masalah kedalam bagian – bagian terkecil atau perlevel sehingga mudah untuk diselesaikan.

b. Pendekatan modul

Pendekatan modul yaitu membagi sistem ke dalam modul – modul yang dapat beroperasi tanpa ketergantungan.

c. Penggunaan alat – alat bantu

Yaitu dalam bentuk grafik dan teks sehingga mudah untuk dimengerti serta dikoreksi apabila terjadi perubahan.

Pendekatan dalam analisa berorientasi objek dilengkapi dengan alat – alat dan teknik – teknik yang dibutuhkan dalam pengembangan sistem, sehingga hasil akhir dari sistem yang dikembangkan akan didapatkan sistem yang terdefinisi dengan baik dan jelas. Maka analisa berorientasi objek akan dilengkapi dengan alat dan teknik didalam mengembangkan sistem.

Alat bantu yang digunakan dalam analisa berorientasi objek antara lain:

B. Activity Diagram

Menurut Munawar[5] *Activity* diagram digunakan untuk menganalisis behavior dengan *use case* yang lebih kompleks dan menunjukan interaksi-interaksi di antara mereka satu sama lain. *Activity* diagram sebenarnya memiliki kesamaan dengan *statechart* diagram dalam hal menggambarkan aliran data pada model bisnis, tetapi *activity* diagram biasanya digunakan untuk menggambarkan aktivitas bisnis yang lebih kompleks, dimana digambarkan hubungan antar satu *use case* dengan *use case* lainnya.

Simbol – simbol yang sering digunakan pada saat pembuatan *activity diagram* :

- 1) Start Point (initial mode), diletakan pada pojok kiri atas dan merupakan awal aktivitas
- 2) End Point (*Activity Final Node*), akhir aktivitas.
- 3) *Activities*, menggambarkan proses bisnis dan di kenal sebagai *activity state*.

Jenis-jenis activity :

- a) *Black Hole Activities*, ada masukan dan ada keluaran.
- b) *Miracle Activities*, tidak ada masukan dan keluaran dan di pakai waktu start point.

- c) *Parallel Activities*, *activity* yang berjalan secara bersamaan terdiri dari:
- a) *Fork* (percabangan)

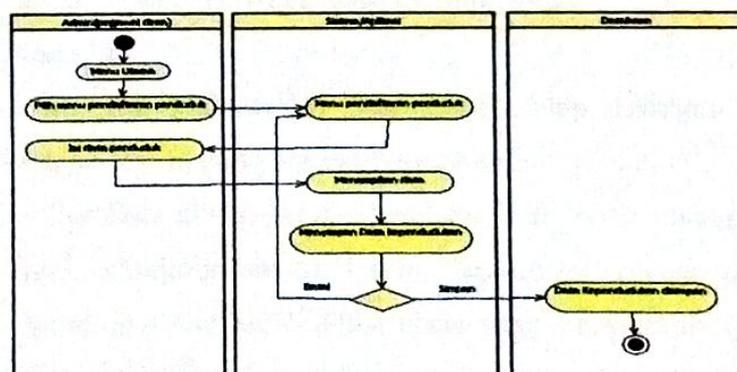
Mempunyai 1 transisi masuk dan 2 atau lebih transisi keluar.
 - b) *Join* (penggabungan)

Mempunyai 2 atau lebih transisi masuk dan hanya 1 transisi keluar.
- 4) *Transition* menggambarkan aliran perpindahan kontrol antara *state*.
- 5) *Decission Point*

Menggambarkan pilihan untuk pengambilan keputusan, true atau false.
- 6) *Swimlane*

Sebuah cara untuk mengelompok *activity* berdasarkan actor (mengelompokkan *activity* berdasarkan actor (mengelompokkan *activity* dalam sebuah urutan yang sama). *Actor* bisa ditulis nama *actor* ataupun sekaligus dalam lambang *actor* (*stik figure* pada *use case diagram*). *Swimlane* digambarkan secara vertical, walaupun kadang-kadang digambarkan *horizontal*.
- 7) *Swimarea*

Ketika sebuah *activity diagram* mempunyai banyak *swimlane*, perlu dipikirkan dengan pendekatan *swimarea*. *Swimarea* mengelompokkan *activity* berdasarkan kegiatan didalam *use case*.



Gambar 2.1

Gambar Contoh Activity Diagram

C. Analisa Dokumen Keluaran

Analisa keluaran adalah analisa mengenai dokumen – dokumen keluaran yang dihasilkan dari sebuah sistem”.

D. Analisa Dokumen Masukan

Analisa masukan adalah bagian dari pengumpulan informasi tentang sistem yang sedang berjalan “. Tujuan analisa masukan adalah memahami prosedur berjalan.

E. *Use case Diagram*

Use Case Diagram menggambarkan sebuah fungsionalitas yang diharapkan dari sebuah sistem dan bagaimana sistem berinteraksi dengan dunia luar. Yang ditekankan dalam *Use Case Diagram* adalah “apa” yang diperbuat sistem, dan bukan “bagaimana” sistem itu melakukannya. Sebuah *Use Case* merepresentasikan sebuah interaksi antar *actor* dengan sistem. *Use Case Diagram* juga menjelaskan manfaat sistem jika dilihat. *Use Case Diagram* terdiri dari:

1) *Actor*

Menurut Mulyanto[6] “*Actor* adalah *abstraction* dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem”.

Untuk mengidentifikasi *actor* harus ditentukan pembagian kerja dan tugas – tugas yang berkaitan dengan peran pada konteks target sistem. *Actor* dilukiskan dengan peran yang mereka mainkan dalam *use case*, seperti staff penjualan, pelanggan, dll.

2) *Use Case*

Menurut Mulyanto[6] “*Use Case* adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna”.

Use Case dibuat berdasarkan keperluan *actor*, merupakan “apa” yang dikerjakan sistem bukan “bagaimana” sistem mengerjakan. Setiap *use case* harus diberi nama yang menyatakan apa hal yang dicapai dari hasil interaksinya dengan *actor* . Nama *use case* boleh

terdiri dari beberapa kata dan tidak boleh ada dua *use case* yang memiliki nama yang sama.

3) *Relationship* (Relasi) / *Association* (Asosiasi)

Menurut Whitten[7] "asosiasi adalah sebuah relasi antara *actor* dengan *use case* dimana sebuah interaksi terjadi diantara mereka".

Relasi (*Relationship*) digambarkan sebagai bentuk garis antar dua simbol dalam *use case diagram*.

Ada empat jenis relasi/ asosiasi yang dapat timbul pada *use case diagram*, yaitu:

(a) Asosiasi antara *Actor* dan *use case*

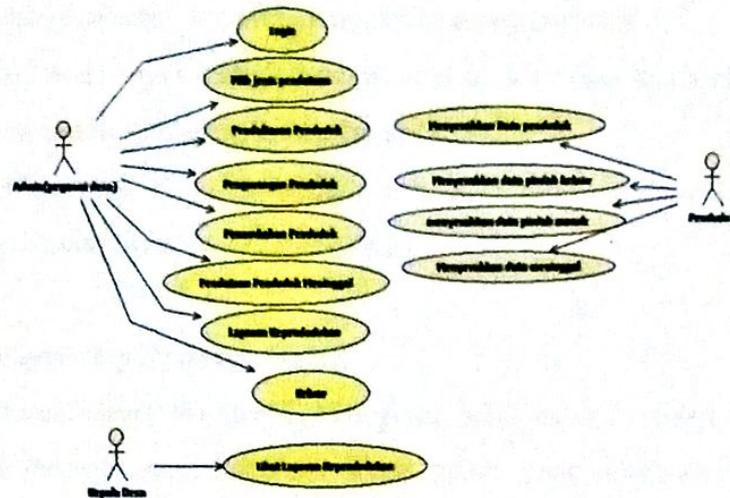
Ujung panah pada *association* antara *actor* dan *use case* mengindikasikan siapa / apa yang meminta interaksi dan bukannya mengindikasikan aliran data. Sebaliknya gunakan garis tanpa panah untuk *association* antara *actor* dan *use case*. *Association* antara *actor* dan *use case* yang menggunakan panah terbuka untuk mengindikasikan bila *actor* berinteraksi secara pasif dengan sistem.

(b) Asosiasi antara Use Case

Relasi antara *use case* dengan *use case* :

- (1) *Include*, menggambarkan suatu *use case* termasuk didalam *use case* lain (diharuskan). Contohnya adalah pemanggilan sebuah fungsi program. Digambarkan dengan garis lurus berpanah dengan tulisan <<include>>.
- (2) *Extend*, digunakan ketika hendak menggambarkan variasi pada kondisi perilaku normal dan menggunakan lebih banyak *control from* dan mendeklarasikan ekstension pada *use case* utama atau dengan kata lain adalah penulusuran dari *use case* lain jika syarat atau kondisi terpenuhi. Digambarkan dengan garis lurus berpanah dengan tulisan <<extend>>.

- (3) *Generalization / Inheritance* antar *Use Case*, dipakai ketika ada sebuah perlakuan khusus (*single condition*) dan merupakan pola hubungan *base – parent use case*
- (4) *Generalization / Inheritance* antar *Actos*, digambarkan *generalization* antar *actor* dibawah *base / parent usecase*.



Gambar 2.2

Gambar Contoh Use Case Diagram

F. Deskripsi Use Case (*Use Case Description*)

Deskripsi *use case* adalah resume langkah atau tahapan dalam *use case*. Kegunaannya untuk mendeskripsikan secara rinci mengenai *use case diagram*. Deskripsi *use case diagram* memiliki tiga komponen yang utama, yaitu:

1. *Use Case*, berisi nama *use case*.
2. *Actor*, berisi nama *actor* yang menjalankan sistem.
3. Deskripsi, menjelaskan bagaimana sistem berjalan.

2.5 Perancangan Berorientasi Obyek

Menurut Whitten[7] "Perancangan sistem berorientasi obyek (*Object – Oriented Design*) adalah suatu pendekatan yang digunakan untuk menspesifikasikan kebutuhan – kebutuhan sistem dengan

mengkolaborasikan obyek – obyek, atribut – atribut dan metode – metode yang ada”.

Perancangan berorientasi obyek bertujuan untuk :

- a. Sistematika proses pendesainan
- b. Menghasilkan pendesainan model program
- c. Memberikan gambaran pemecahan masukan dengan efektif

Fokus dari desain obyek adalah perencanaan struktur data dan algoritma yang diperlukan untuk implementasi setiap kelas.

Adapun diagram – diagram UML yang digunakan penulisan dalam merancang sistem berorientasi objek adalah:

a. ***Entity Relationship Diagram (ERD)***

Menurut Jeffery Whitten[7] “Diagram - ER adalah sebuah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam hal entitas dan relasi yang digambarkan oleh data tersebut”.

Yang pertama kali mendeskripsikan diagram- ER adalah Peter Chan yang dibuat sebagai bagian dari perangkat lunak case. Model ERD adalah suatu penyajian data dengan menggunakan antity dan relationship. Diagram ER menggambarkan hubungan antara data yang ada dan tidak menggambarkan proses – proses yang terjadi.

Simbol – simbol/notasi yang biasa digunakan dalam ERD, antara lain :

- 1) Menurut Whitten[7] “*Entity*”(entitas) adalah sebuah kelas dari orang, tempat objek, kejadian dan sebagainya atau sebuah konnsep yang mana kita perlukan untuk menangkap dan menyimpan data”.

Pada *entity* terdapat dua jenis, yaitu:

- a) *Strong Entity* adalah *entity* yang memiliki *primary key*
- b) *Weak Entity* adalah suatu *entity* yang tidak memiliki *primary key* dan keberadaan *entity* tersebut tergantung dari keberadaan *entity* lain. *Entity* yang merupakan induknya disebut

identifying owner dan relasinya disebut *identifying relationship*.

2) *Relationship* (Hubungan/relasi), menunjukkan adanya hubungan diantara sejumlah entitas yang berbeda.

3) *Cardinality/Kardinalitas*, *Cardinality* adalah tingkat hubungan atau derajat relasi. Tingkat *Cardinality* yang terjadi pada sebuah ERD adalah sebagai berikut:

a) *One to one* (1:1)

Hubungan relasi *one to one* yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B.

b) *One to Many* (1:M)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A.

c) *Many to One*(M:1)

Setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A, tetapi setiap entitas pada entitas A dapat berhubungan dengan satu entitas pada himpunan entitas B

d) *Many to many*(M:N)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas himpunan entitas B, begitu juga sebaliknya.

4) Menurut Whitten[7] "Atribut adalah suatu deskripsi karekteristik dari entitas".

Nilai atribut merupakan suatu data aktual atau informasi yang disimpan pada suatu atribut dalam satu *entity* atau *relationship*.

Terdapat dua jenis atribut, yaitu :

- a) *Identifier (key)*, menentukan *entity* secara unik (beda antara satu sama lain dan tidak mungkin sama).
- b) *Descriptor (non key atribut)*, menentukan *entity* yang tidak unik.

Atribut *relationship* sangat ditentukan oleh *cardinality*, yaitu :

- (1) Jika *cardinality* 1 : 1 dan 1 : M, atribut *relationship* diambil dari *identifer* dari entitas di kiri dan kanan.
 - (2) Jika *cardinality* M : N, atribut *relationship* diambil dari *identifer* dari entitas di kiri dan kanan ditambah dengan atribut lain yang bukan milik entitas di kiri ataupun di kanan.
- 5) *Line*, sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.

b. Logical Record Structure (LRS)

Diagram ER (ERD) harus di konversi ke bentuk *structure* (struktur *record* secara logik). Sebuah model sistem yang digambarkan dengan sebuah Diagram ER akan mengikuti pola/aturan pemodelan tertentu. Dalam kaitanya dengan konversi ke LRS, maka perubahan yang terjadi adalah mengikuti aturan – aturan berikut ini :

- 1) Setiap entitas akan diubah ke bentuk kotak.
- 2) Sebuah *relationship* kadang disatukan dalam sebuah kotak bersama entitas jika hubungan yang terjadi pada Diagram ER 1 : M (relasi bersatu dengan *cardinality* M) atau tingkat hubungan 1 : 1 (relasi bersatu dengan *cardinality* yang paling membutuhkan referensi), sebuah *relationship* dipisah dalam bentuk kotak tersendiri jika tingkat hubungannya M : N (*Many to many*).

c. Tabel /Relasi

Tabel adalah koleksi objek yang terdiri dari sekumpulan elemen yang diorganisasi secara kontinyu, artinya memori yang dialokasi antara satu elemen dengan elemen yang lainnya mempunyai *address* yang berurutan.

Untuk mentransformasi LRS ke tabel/relasi, berpedoman pada hal – hal sebagai berikut :

- 1) Tiap satu LRS akan menjadi satu tabel.
- 2) Nama LRS menjadi satu tabel.
- 3) Tiap 1 (satu) atribut akan menjadi 1 (satu kolom).
- 4) Nama atribut akan menjadi nama kolom.

d. Spesifikasi Basis Data

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lain dan tersimpan di luar komputer serta digunakan perangkat lunak (*software*) tertentu untuk memanipulasinya.

Sedangkan sistem berbasis data adalah suatu sistem penyusunan dan pengolahan *record – record* dengan menggunakan komputer dengan tujuan untuk menyimpan atau merekam serta melihat data operasional lengkap pada sebuah organisasi.

2.6 Pengertian Web

Menurut Supriyanto[8] *Web* atau lengkapnya *www (world wide web)* adalah sebuah koleksi keterhubungan dokumen-dokumen yang disimpan di internet dan diakses menggunakan protocol (*Hyper Text Transfer Protocol*). Intinya bahwa pengguna internet bisa memanfaatkan berbagai macam fasilitas informasi dengan biaya murah tanpa harus datang secara langsung ketempatnya. Informasi atau dokumen yang dapat diakses dapat berupa data teks, gambar atau image, animasi, video, suara, atau kombinasi diantaranya dan bahkan komunikasi bisa dilakukan secara langsung dengan suara dan video sekaligus. Website dikategorikan kedalam dua jenis diantaranya:

1. Web Statis

Merupakan web yang isi kontennya tidak dapat dirubah secara langsung oleh *end user*. Isi dari web hanya bisa dirubah oleh orang yang mengerti mengenai bahasa pemrograman untuk merancang web tersebut. Web jenis ini tidak mempunyai basis data , jadi informasi yang ada dituliskan langsung kedalam bahasa pemrograman.

2. Web Dinamis

Merupakan web yang isi kontennya sudah teratur dengan baik artinya sudah menerapkan konten manajemen sistem. *End user* bisa sewaktu-waktu memperbarui konten pada halaman web tanpa perlu paham bahasa pemrograman.

2.7 Perangkat Lunak Pendukung

Adapun perangkat lunak yang digunakan untuk membangun perancangan sistem informasi ini, yaitu :

a. Macromedia Dreamweaver 8

Merupakan alat desain web komprehensif yang disukai oleh para profesional web, tapi cukup mudah untuk web designer pemula untuk menggunakannya karena menggunakan WYSIWYG yg artinya apa yang dilihat adalah apa yang akan didapatkan dalam merancang antarmuka halaman web semudah mengetik dokumen ke dalam pengolah kata. Program ini juga menawarkan kemampuan untuk bekerja secara langsung dengan kode halaman web bagi mereka yang suka dengan HTML. Intuitif menu, panel, dan toolbar memungkinkan fitur-fitur canggih ke situs web dengan mudah.

b. MySQL

MySQL (*My Structure Query Language*) adalah sebuah program pembuat database yang bersifat *opensource*. MySQL sebenarnya produk yang

berjalan pada platform Linux karena sifatnya open source MySQL dapat dijalankan pada semua platform baik Windows maupun Linux. Database MySQL adalah database yang sangat powerful, stabil, mudah. MySQL sangat banyak dipakai dalam sistem database web dengan menggunakan PHP. PHP Triad juga memberikan fasilitas database yang. Karena PHP Triad dilengkapi dengan database MySQL maka terdapat tempat untuk menyimpan data (store), dan untuk mengambil kembali data anda. Seperti sistem database SQL (*Structured Query Language*) yang lain, MySQL juga dilengkapi dengan perintah-perintah dan sintaks-sintaks SQL, dengan keunggulan sebagai berikut.

- a. Konsep database MySQL berkecepatan tinggi tentang sistem penyajian data.
- b. Harga yang relatif murah, karena ada yang dapat diperoleh secara gratis.
- c. Sintaks bahasanya menggunakan perintah yang sederhana.
- d. Dapat bekerja dalam beberapa Sistem operasi seperti *Windows, Linux, MacOS, Unix (Solaris, AIX, dan DEC Unix), FreeBSD, OS/2, Irix.*
- e. Dukungan penggunaan banyak tersedia.

c. Xampp

Merupakan perangkat lunak bebas yang mendukung banyak sistem operasi, merupakan campuran dari beberapa program. Yang mempunyai fungsi sebagai server yang berdiri sendiri (localhost) yang terdiri dari program MySQL database, Apache HTTP server, dan penerjemah yang ditulis ke dalam bahasa pemrograman PHP dan perl. Beberapa komponen xampp diantaranya sebagai berikut :

1. Htdoc adalah folder dimana tempat meletakkan file yang akan di jalankan seperti file *PHP, HTML* dan script lainnya
2. PHPMyAdmin adalah bagian untuk mengelola database *MySQL* yang dikomputer
3. Control Panel adalah berfungsi untuk mengelola layanan xampp seperti *stop service* atau memulai.

2.8 Rangkuman Penelitian

No	Judul	Metode Yang Digunakan	Kesimpulan
1	<p>Sistem informasi administrasi kependudukan di kantor desa mandalawangi kab. Bandung barat[Puri, Ayu,Sundari.,2014, Sistem informasi administrasi kependudukan di kantor desa mandalawangi kab. Bandung barat. Tugas Akhir,Program Studi Manajemen Informatika, Univ. Komputer Indonesia, Bandung.]</p>	<p>Metode penelitian yang digunakan adalah metode pengumpulan data dan metode pengembangan sistem yang digunakan adalah prototype model.</p>	<p>Dengan adanya penyimpanan data yang sudah berbentuk database, maka kemungkinan tidak akan terjadi duplikasi data.</p> <p>Dengan sistem informasi kependudukan ini, diharapkan dapat menjadi sebuah sistem pelayanan kependudukan yang dapat meningkatkan pelayanan terhadap masyarakat</p>
2	<p>Rancangan sistem informasi akademik berbasis web pada SMK SORE pangkalpinang[Devyta, Anggrayni.,2014, Rancangan sistem informasi akademik berbasis web pada SMK SORE</p>	<p>Metode penelitian yang digunakan adalah metode survei lapangan,wawancara,dan analisa</p>	<p>Dengan sistem informasi akademik berbasis web ini dapat lebih efisien dalam pendistribusian informasi sekolah seperti data siswa dll.</p>

	<p>pangkalpinang, Skripsi,Program Studi Sistem Infomasi,Sekolah Tinggi Manajemen Informatika, PangkalPinang.]</p>		
3	<p>Penerapan sistem informasi administrasi kependudukan pada kantor Lurah Parit Padang dengan menggunakan metodologi berorientasi obyek[Gama, Yudishtira.,2014, Penerapan sistem informasi administrasi kependudukan pada kantor Lurah Parit Padang dengan menggunakan</p>	<p>Metode peelitian yang digunakan adalah obeservasi,wawancara,jaringan internet,dan studi kepustakaan.</p>	<p>Dengan sistem informasi admintras kependudukan pada kantor lurah parit padang dengan meotodlogi berorientasi objek ini maka akan membantu dalam melakukan pelayanan adminstrasi kependudukan</p>

	<p>metodologi berorientasi obyek, Skripsi, Program Studi Sistem Infomasi, Sekolah Tinggi Manajemen Informatika, PangkalPinang.]</p>		
4	<p>Analisa dan perancangan sistem infromasi adminitrasi kepegawaian pada kelurahan pasir garam pangkalpinang[Rahayu, Nengsih.,2014, Analisa dan perancangan sistem infromasi adminitrasi kepegawaian pada kelurahan pasir garam pangkalpinang, Skripsi, Program Studi Sistem Infomasi, Sekolah Tinggi Manajemen Informatika, PangkalPinang.]</p>	<p>Metode penelitian yang digunakan adalah pengamatan, wawancara, dan metode kepustakaan</p>	<p>Dengan sistem informasi kepegawaian pada kelurahan pasir garam pangkalpinang maka sistem komputerisasi mampu meminimalisir tingkat kesalahan yang terjadi pada sistem yang ada.</p>

5	<p>Analisa dan perancangan sistem informasi administrasi kependudukan di kantor kelurahan opas indah pangkalpinang[Angga, Wijaya.,2014, Sistem Analisa dan perancangan sistem informasi administrasi kependudukan di kantor kelurahan opas indah pangkalpinang, Skripsi,Program Studi Sistem Infomasi,Sekolah Tinggi Manajemen Informatika, PangkalPinang.]</p>	<p>Metode peelitian yang digunakan adalah obeservasi,wawancara,jaringan internet,dan studi kepustakaan.</p>	<p>Dengan adanya sistem informasi administrasi kependudukan ini makasangat membantu dalam melakukan pelayanan surat menyurat pada instansi yang ada.</p>
---	---	---	--