

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Dalam pembuatan tugas akhir ini ada beberapa tinjauan pustaka sebagai perbandingan untuk menganalisis sistem yang ada, antara lain :

2.1.1 Kajian Pustaka

Eldiabo (2017) dalam tugas akhir yang berjudul “Portal Sistem Informasi Pengelolaan Karang Taruna Kabupaten Kudus Berbasis Web” telah dilaksanakan dengan menganalisa permasalahan yang ada diantaranya mengenai pengelolaan pendataan yang kurang efektif dan efisien. Tujuan dari skripsi ini adalah menghasilkan aplikasi perangkat lunak untuk merancang suatu sistem Informasi Karang Taruna yang dapat membantu menunjang kelancaran kegiatan di Karang Taruna Kabupaten Kudus. Sistem ini dirancang menggunakan bahasa pemrograman yang digunakan adalah *PHP* dan *Database MySQL*. Hasil dari rancang bangun ini adalah sebuah aplikasi berbasis *web* untuk Karang Taruna Kabupaten Kudus.

Lutfie (2014) dalam tugas akhir yang berjudul “Rancang Bangun Sistem Informasi Spasial Sasana Krida Karang Taruna berbasis *web*” di Jakarta selatan. Karang Taruna adalah Organisasi kepemudaan yang diakui keberadaanya oleh Negara Republik Indonesia. Karang Taruna didirikan dengan tujuan memberikan pembinaan dan pemberdayaan kepada para remaja, misalnya dalam bidang keorganisasian, ekonomi, olahraga, advokasi, keagamaan dan kesenian. Untuk memaksimalkan kegiatan Karang Taruna maka perlu adanya wadah sebagai tempat pengembangan kegiatan. Oleh karena itu, perlu adanya sistem yang dapat membantu Suku Dinas Sosial Jakarta Selatan untuk memudahkan Pengurus Karang Taruna dalam melaporkankondisi SKKT yang perlu di renovasi. Sehingga dalam pelayanannya Suku Dinas Sosial dapat memberikan bantuan yang diharapkan. Sistem ini dirancang dengan menggunakan bahasa pemrograman *PHP* dan *Database menggunakan MySQL*.

Iskandar, Novi (2013), dalam tugas akhir yang berjudul “Analisis dan Media Informasi dan Komunikasi” Krida Taruna Maguwoharjo diperlukan media untuk memberikan fasilitas dalam penyampaian informasi kepada masyarakat umum. Selain itu juga diperlukan untuk menyediakan akses komunikasi antar anggota Krida Taruna Maguwoharjo, Sistem pemrosesan data yang kurang terstruktur dan dikerjakan secara manual menjadi predikat buruk kinerja Krida Taruna. Tujuan tugas akhir ini adalah untuk memberikan akses informasi kepada public tentang program kinerja dan segala sesuatu yang berkaitan dengan pemuda, serta menyediakan kenyamanan untuk para anggota Krida Taruna Maguwoharjo dalam penyimpanan data sebagai sarana dokumentasi dan peningkatan komunikasi. Sistem ini dirancang menggunakan bahasa pemrograman *PHP* dan *database* menggunakan *Mysql*.

Adha (2019), dalam tugas akhir yang berjudul “Pembuatan *Website* IKM KOTIM di Solo Raya menggunakan *Framework Codigniter*” dijelaskan bahwa permasalahan yang dihadapi IKM KOTIM dalam penyampaian informasi hanya dilakukan dengan media sosial seperti *Facebook*, *Whatsapp*, *Instagram* saja. Dan untuk penambahan anggota baru masih bersifat manual, dengan cara penyampaiannya informasi yang masih menggunakan media sosial maka yang dapat mengetahui informasi tersebut hanya mahasiswa yang sudah berteman dengan media sosial yang digunakan oleh anggota IKM KOTIM. Tujuan dari tugas akhir ini adalah membangun sebuah sistem *website* IKM KOTIM yang diharapkan dapat mempermudah penyampaian informasi kepada masyarakat dan bagi mahasiswa dari KOTIM yang belum terdaftar didalam IKM KOTIM dapat mendaftarkan diri melalui *website* IKM KOTIM Solo Raya. Sistem ini dirancang menggunakan *Framework Codeigniter*.

2.2 Kerangka Pemikiran

Kerangka pemikiran adalah merupakan gambaran dari awal sampai akhir yang di gunakan dalam perancangan implementasi *Website* karang taruna pada desa Balongcabe. Adapaun kerangka pemikiran yang dijalankan adalah sebagai berikut:

a. Latar belakang masalah

Tahapan paling awal, yakni menelusuri latar belakang masalah yang akan diangkat dalam tugas akhir.

b. Perumusan masalah

Penyimpulan latar belakang masalah menjadi satu perumusan masalah untuk menjadi bahan permasalahan yang menjadi pokok pembahasan dalam tugas akhir ini.

c. Penguasaan dasar *PHP* dan *MySQL*

Tahap untuk mempelajari dasar-dasar *PHP* dan *MySQL* yang akan digunakan untuk merancang bangun sistem.

d. Pengumpulan data tertulis dan tidak tertulis

Pengumpulan data dilakukan baik dengan tanya jawab (*interview*), observasi, maupun studi pustaka.

e. Observasi data dan informasi

Merupakan tahap pengamatan dan pengumpulan sampel-sampel data dan informasi yang dibutuhkan untuk menunjang tugas akhir.

f. Perancangan dan implementasi aplikasi terstruktur

Tahap ini melakukan analisis yang berkaitan dengan proses data yang diperlukan oleh sistem dengan menggunakan *tool Data Flow Diagram (DFD)*, perancangan sistem meliputi perancangan data, perancangan arsitektural, perancangan antar muka dan perancangan prosedur.

g. Implementasi *Website Karang Taruna*

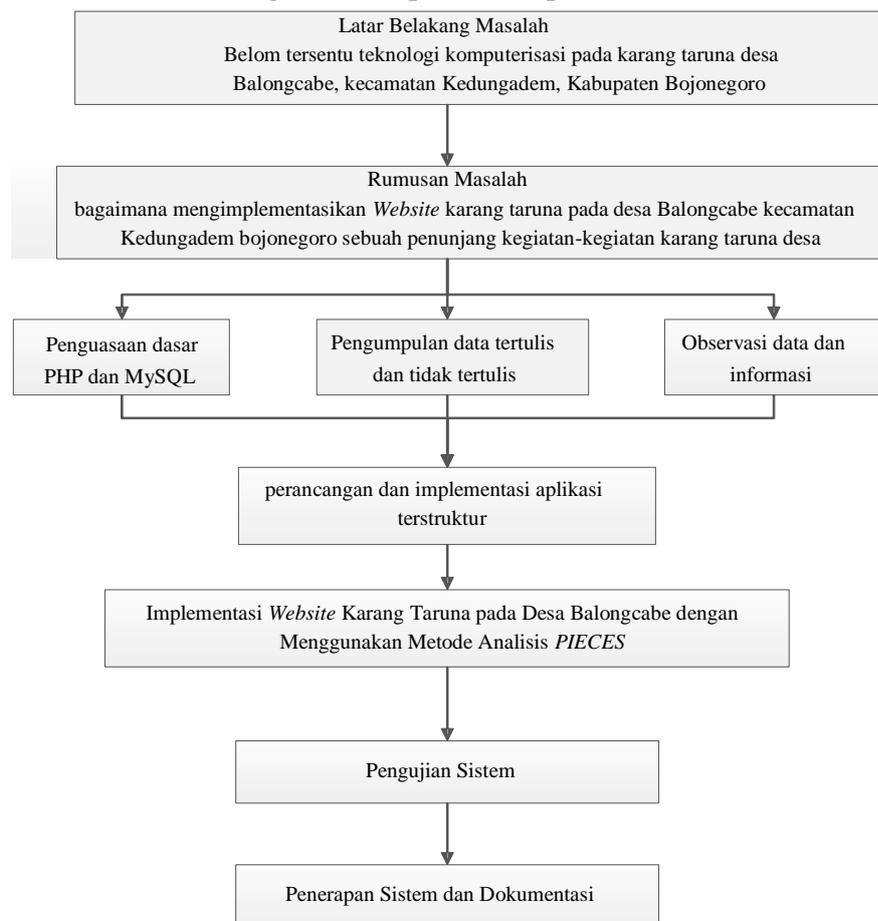
Implementasi merupakan tahap menerjemahkan perancangan berdasarkan hasil analisis, tujuan implementasi adalah untuk mengkonfirmasi modul program yang sudah dirancang bangun dapat mudah dimengerti (*friendly*) untuk pengguna atau Pengunjung atau belum, sehingga dapat memberi masukan kepada pengembang sistem.

h. Pengujian Sistem

Pengujian sistem akan dilakukan pada beberapa komputer untuk mengetahui jika ada kesalahan pada sistem.

i. Penerapan Sistem dan Dokumentasi

Tahap terakhir dari seluruh proses perancangan dan pembangunan sistem hingga dapat diimplementasikan, di gunakan oleh Karang Taruna Desa Balongcabe Kecamatan Kedungadem Kabupaten Bojonegoro dibuat dokumentasi dari keseluruhan kegiatan tersebut kedalam tugas akhir. Kerangka pemikiran Tugas Akhir *Website* karang taruna dapat dilihat pada Gambar 2.1.



Gambar 2.1 Kerangka Pemikiran

2.3 Landasan Teori Pendukung

Beberapa istilah yang berhubungan dengan pembuatan *Website* karang taruna pada Desa Balongcabe Kecamatan Kedungadem Kabupaten Bojonegoro.

2.3.1 Pengertian Karang Taruna

Menurut Warastuti (2016). Pengertian lain menyatakan organisasi kepemudaan adalah organisasi sosial wadah pengembangan generasi muda yang tumbuh dan berkembang atas dasar kesadaran dan tanggung jawab sosial dari, oleh, dan untuk masyarakat terutama generasi muda di desa/kelurahan atau komunitas adat sederajat yang bergerak dibidang usaha kesejahteraan sosial

Dinas Sosial berdasarkan Permensos RI Nomor 77/HUK/2010 Bahwa Karang Taruna merupakan salah satu Organisasi sosial kemasyarakatan yang diakui keberadaanya oleh negara dalam penyelenggaraan kesejahteraan sosial sebagaimana tercantum dalam pasal 38 ayat (2) Nomor 11 Tahun 2009 tentang kesejahteraan sosial.

2.3.2 Pengertian Implementasi

Konsep implementasi semakin marak dibicarakan seiring dengan banyaknya pakar yang memberikan kontribusi pemikiran tentang implementasi sebagai salah satu tahap dari proses. beberapa penulis menempatkan tahap implementasi pada posisi yang berbeda, namun pada prinsipnya implementasi memiliki makna yang sama.

Menurut Purwanto dan Sulistyastuti (2013). “implementasi intinya adalah kegiatan untuk mendistribusikan keluaran kebijakan (*to deliver policy output*) yang dilakukan oleh para implementor kepada kelompok sasaran (*target group*) sebagai upaya untuk mewujudkan kebijakan”.

Menurut Agustino (2012). “implementasi merupakan suatu proses yang dinamis, dimana pelaksana kebijakan melakukan suatu aktivitas atau kegiatan, sehingga pada akhirnya akan mendapatkan suatu hasil yang sesuai dengan tujuan atau sasaran kebijakan itu sendiri”

Dari berbagai defenisi diatas maka dapat disimpulkan bahwa implementasi adalah serangkaian tindakan yang dilakukan dalam berbagai kondisi dengan sarana-sarana pendukung berdasarkan aturan-aturan yang telah ditentukan untuk mencapai tujuan yang telah ditetapkan.

2.3.3 Pengertian *Website*

Website adalah suatu halaman *web* yang saling berhubungan yang umumnya berada pada peladen yang sama berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi. Sebuah situs *web* biasanya ditempatkan setidaknya pada sebuah *server web* yang dapat diakses melalui jaringan *internet*.

Adulloh, Rohi (2016) *Website* atau disingkat *web*, dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa *text*, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur *internet*. Lebih jelasnya, *Website* merupakan halaman-halaman yang berisi informasi yang ditampilkan oleh *browser* seperti *Mozilla Firefox*, *Google Chrome* atau yang lainnya.

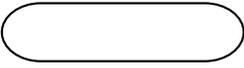
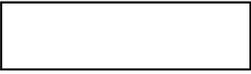
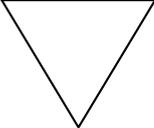
Website merupakan fasilitas *internet* yang menghubungkan dokumen dalam lingkup lokal maupun jarak jauh. Dokumen pada *Website* disebut dengan *web page* dan *link* dalam *Website* memungkinkan pengguna bisa berpindah dari satu halaman ke halaman yang lain, baik diantara halaman yang disimpan dalam *server* maupun *server* diseluruh dunia, (Hakim, Lukmanul, 2004).

2.3.4 *Flowchart*

Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi atau penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program atau bisa diartikan sebuah jenis diagram yang mewakili algoritme, alir kerja atau proses, yang menampilkan langkah-langkah dalam bentuk simbol-simbol grafis, dan urutannya dihubungkan dengan panah. Diagram ini mewakili ilustrasi atau penggambaran penyelesaian masalah. Diagram alir digunakan untuk menganalisa, mendesain, mendokumentasi atau memajemen sebuah proses atau program di berbagai bidang (Annafi, 2014).

Berikut adalah simbol yang digunakan dalam menyusun *flowchart*, kegiatan yang diawali dengan aturan yang diterapkan dalam penggunaan simbol disajikan pada Tabel 2.1.

Tabel 2.1 Simbol *Flowchart* (Sumber : Annafi, 2014)

NO	SIMBOL	NAMA	FUNGSI
1		TERMINATOR	Permulaan / akhir program
2		GARIS ALUR (FLOW LINE)	Arah aliran program
3		MANUAL	Menunjukkan suatu proses yang tidak di lakukan komputer
4		PROSES	Proses Perhitungan /proses pengolahan data
5		Offline Storage	Menunjukkan data dalam symbol ini akan disimpan ke suatu media tertentu
6		DECISION	Perbandingan pernyataan, penyelesaian data yang memberikan pilihan untuk langkah selanjutnya
7		Disk Storage	Menyatakan input dari disk atau output disimpan ke disk
8		Document	Mencetak keluaran dalam bentuk dokumen

2.3.5 *Context Diagram*

Context Diagram adalah bagian dari tujuan *Data Flow Diagram* yang berfungsi untuk memetakan model data untuk dipresentasikan dengan lingkaran tunggal yang mewakili seluruh sistem. *Context Diagram* dimulai dengan penggambaran terminator, aliran data, aliran control, penyimpanan dan proses tunggal yang mempresentasikan keseluruhan sistem (Pohan, 1997 : 3).

Context diagram menyoroti sejumlah karakteristik penting sistem, yaitu :

1. Kelompok Pemakai.

Organisasi atau sistem lain dimana sistem melakukan komunikasi yang disebut sebagai terminator.

2. Data Masuk.

Data Masuk merupakan data yang diterima sistem dari lingkungan dan harus diproses dengan cara tertentu.

3. Data Keluar.

Data Keluar merupakan data yang dihasilkan sistem dan diberikan ke pihak l.

4. Penyimpanan data (*Data Storage*).

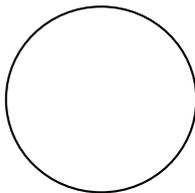
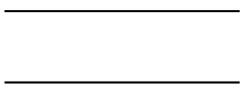
Penyimpanan data (*Data Storage*) digunakan secara bersama-sama antara sistem dengan terminator. Data ini dapat dibuat oleh sistem dan digunakan oleh lingkungan atau sebaliknya, dibuat oleh lingkungan dan digunakan oleh sistem.

2.3.6 **Data Flow Diagram (DFD)**

Menurut Irawan (2015:700) *Data Flow Diagram* merupakan gambaran arus data di dalam sistem secara terstruktur dan jelas. *Data Flow Diagram* dapat merupakan dokumentasi dari sistem yang baik. Simbol-simbol yang digunakan disajikan pada Tabel 2.2.

Tabel 2.2 Simbol-simbol *Data Flow Diagram*

(Sumber : Irawan 2015:700)

NO	SIMBOL	NAMA	FUNGSI
1.		EXTERNAL ENTITY	<i>External Entity</i> (Kesatuan Luar) atau <i>Boundary</i> (Batas Sistem) yang memisahkan suatu sistem dengan lingkungan luarnya. <i>External Entity</i> merupakan kesatuan (<i>entity</i>) dilingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada dilingkungan luarnya akan memberikan input atau menerima output dari sistem.
2.		DATA PROSES	Data proses adalah kegiatan yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masih dalam proses untuk menghasilkan arus data yang akan keluar dari proses.
3.		DATA FLOW (ARUS DATA)	Arus data ini menunjukkan arus data dari yang dapat berupa masukan untuk sistem atau hasil dari proses sistem. Arus data dalam DFD diberi simbol panah.
4.		DATA STORE	Tempat penyimpanan data dalam <i>data flow diagram</i> untuk data yang telah diproses.

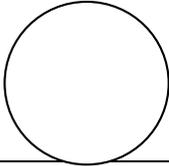
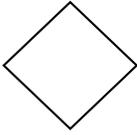
2.3.7 Diagram Hubungan Entitas (ER-Diagram)

Menurut Irawan (2015:700) ER-Diagram adalah diagram yang secara grafis menggambarkan relasi antara masing-masing himpunan entitas dilengkapi dengan atribut-atributnya. Entitas itu sendiri merupakan suatu objek yang mewakili suatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Sedangkan atribut dari suatu entitas berfungsi mendeskripsikan karakteristik dari suatu entitas. Dari beberapa atribut yang ada, maka dipilih satu atribut sebagai kunci

utama (*primary key*). Atribut kunci utama ini harus unik dan tidak sama dengan atribut lain. Berikut ini adalah Tabel 2.1 komponen yang digunakan dalam Diagram Hubungan Entitas atau ERD.

Tabel 2.3 Simbol-Simbol *Entity Relationship Diagram*

(Sumber : Irawan 2015:700)

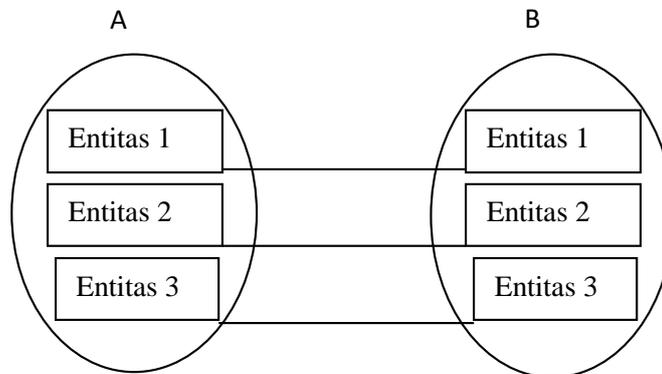
NO	SIMBOL	NAMA	FUNGSI
1.		ENTITAS	Persegi panjang, menyatakan himpunan entitas. Entitas/entity adalah orang, kejadian atau konsep yang informasinya direkam
2.		ATRIBUT	Lingkaran/Elips menyatakan atribut (atribut yang berfungsi sebagai key digaris bawah). Atribut adalah data elemen, data field atau data item.
3.		GARIS	Garis sebagai penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.
4.		RELASI	Belah ketupat, menyatakan himpunan relasi yaitu hubungan antara satu atribut dengan atribut yang lain.

Sedangkan relasi pada model data relasional menggambarkan hubungan antara satu entitas dengan entitas lainnya (misalnya entitas A dan entitas B), dan biasanya hubungan ini terjadi antara dua atribut yang sama (paling tidak tipe dan ukurannya) pada entitas yang memiliki relasi tersebut. Sebuah entitas dapat memiliki relasi yang lebih dari satu entitas lainnya selama syarat di atas terpenuhi. Relasi ini digambarkan dengan garis (baik lurus maupun patah) dengan notasi yang melambangkan jenis relasi di kedua ujungnya. Sedangkan jenis relasi yang terjadi antara dua entitas itu sendiri memiliki beberapa kemungkinan yaitu :

1. Relasi satu ke satu (*One to One*)

Relasi satu ke satu (*One to One*) yang berarti setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu pula sebaliknya setiap entitas pada himpunan B berhubungan paling

banyak dengan satu entitas pada himpunan entitas A. Kordinator Relasi *One To One* dapat di relasikan pada Gambar 2.2.

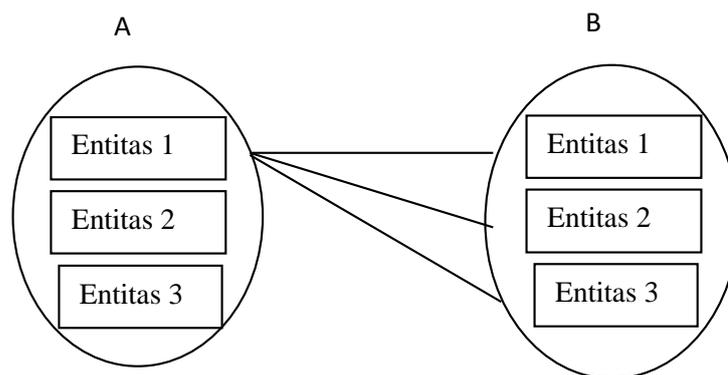


Gambar 2.2 Kardinalitas Relasi *One To One*

Sumber. Fathansyah (2014:77)

2. Relasi satu ke banyak (*One To Many*)

Relasi satu ke banyak (*One To Many*) yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan B, tetapi tidak sebaliknya setiap entitas pada himpunan B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A. Kordinator Relasi *One To Many* dapat di relasikan pada Gambar 2.3.



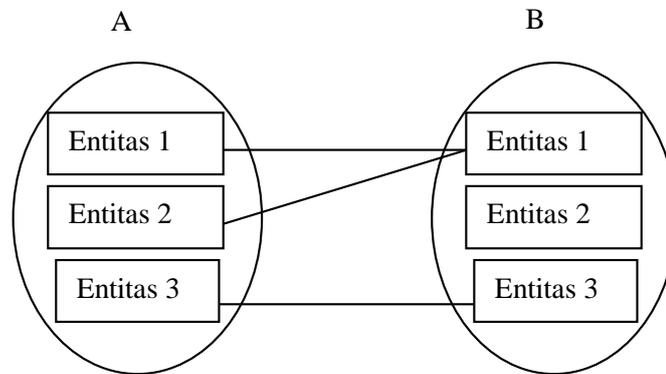
Gambar 2.3 Kardinalitas Relasi *One To Many*

(Sumber. Fathansyah (2014:78))

3. Relasi banyak ke satu (*Many To One*)

Relasi banyak ke satu (*Many To One*) yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan paling banyak satu entitas pada

himpunan entitas B, tetapi tidak sebaliknya dimana setiap entitas pada himpunan B berhubungan dengan paling banyak satu entitas pada himpunan entitas A. Kordinator Relasi *Many To One* dapat di relasikan pada Gambar 2.4.

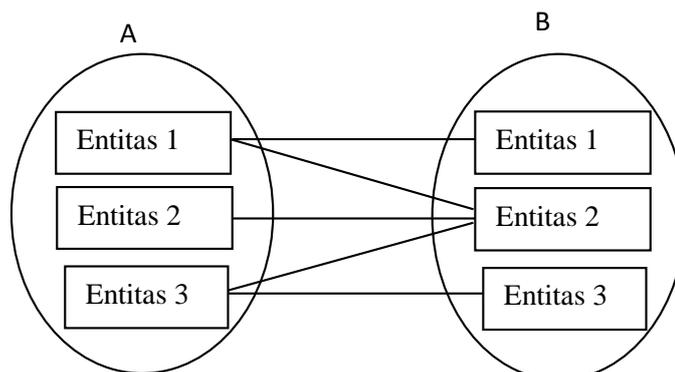


Gambar 2.4 Kardinalitas relasi *Many To One*

(Sumber. Fathansyah (2014:78))

4. Relasi banyak ke banyak (*Many To Many*)

Relasi banyak ke banyak (*Many To Many*) yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan B, dan demikian juga sebaliknya dimana setiap entitas pada himpunan entitas B berhubungan dengan banyak entitas pada himpunan entitas. Kordinator Relasi *Many To Many* dapat di relasikan pada Gambar 2.5.



Gambar 2.5 Kardinalitas Relasi *Many To Many*

(Sumber. Fathansyah (2014:78))

2.3.8 Database

Definisi *Database* adalah objek yang kompleks untuk menyimpan informasi yang terstruktur, yang diorganisir dan disimpan dalam suatu cara yang memungkinkan pemakainya dapat mengambil informasi dengan cepat dan efisien.

Sumber lain mengatakan bahwa “*Database* merupakan komponen terpenting dalam pembangunan sistem informasi, karena *Database* menjadi tempat untuk menampung dan mengorganisasikan seluruh data yang ada dalam sistem, sehingga dapat di eksploitasi untuk menyusun informasi dalam berbagai bentuk”.

Sumber lain menyatakan bahwa pengertian *Database* adalah “*Database* adalah sekumpulan data atau informasi yang teratur berdasarkan kriteria tertentu yang saling berhubungan” (Yuswanto, 2013). *Database* terdiri dari beberapa tabel, yang dimaksud adalah tabel yang terdapat pada pemrograman : “Tabel sebagai tempat penampung data yang mempunyai dua bagian utama, yaitu nama dan tipe data” (Ridwan, 2013). Sedangkan tujuan *Database* adalah untuk menyimpan informasi dan mengambilnya dengan cepat. Untuk dapat menggunakan informasi dalam *Database* perlu memperhatikan struktur *Database*.

2.3.9 *MySQL (Structured Query Language)*

MySQL merupakan salah satu *software Database* (basis data) *open source* dengan tujuan membantu Pengunjung untuk meyimpan data dalam tabel-tabel. Tabel terdiri atas *field* (kolom) yang mengelompokkan data-data berdasarkan kategori tertentu, misalnya nama, alamat, nomor telepon, dan sebagainya. Bagian lain dari tabel adalah *record* (baris) yang mencantumkan data yang sebenarnya. *MySQL* sebagaimana *software Database* lainnya, dapat menampung banyak schemata, dimana masing-masing schemata ini dapat digunakan oleh aplikasi-aplikasi yang berbeda, baik dari sisi tujuan maupun dari sisi bahasa pemrograman yang digunakan oleh masing-masing aplikasi yang bersangkutan. Terdapat empat instruksi dasar yang digunakan dalam *sql* (*structured query language*), yaitu *select* (menampilkan data), *Insert* (menginput atau menambah data), *Update* (mengubah data) dan *Delete* (menghapus data) dalam *Database*.

2.3.10 XAMPP

XAMPP adalah perangkat *software* gratis yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsi XAMPP adalah sebagai *server offline* yang berdiri sendiri (*localhost*). XAMPP terdiri dari

beberapa program yaitu *Apache (web server)*, *MySQL (Database)*, *PHP (server side scripting)*, *Perl*, *FTP Server*, *PHPMyAdmin* dan berbagai pustaka bantu lainnya. XAMPP terkenal sebagai program *web server (localhost)* yang mudah digunakan (*Pengunjung friendly*).

XAMPP ini adalah sebuah singkatan yang masing-masing hurufnya memiliki penjelasan, yaitu sebagai berikut :

1. X

XAMPP dapat dijalankan (kompatibel) di berbagai sistem operasi seperti Windows, Linux, Mac OS dan Solaris.

2. A

A adalah kepanjangan dari program *Apache*. *Apache* merupakan sebuah aplikasi *web server*. Tugas utama *Apache* adalah menghasilkan halaman web yang benar kepada Pengunjung (menerjemahkan) berdasarkan kode *PHP* dan *MySQL* yang ditulis oleh *web programmer*.

3. M

M adalah singkatan dari program *MySQL*. *MySQL* adalah sebuah aplikasi *Database server*, *MySQL* dapat digunakan untuk membuat dan mengelola *Database* beserta isinya. *MySQL* dapat digunakan untuk menambahkan, mengubah dan menghapus data yang berada dalam *Database*.

4. P

P pertama adalah singkatan dari program *PHP*. *PHP* adalah sebuah aplikasi bahasa pemrograman untuk membuat sebuah web yang sifatnya adalah *server-side scripting*.

5. P

P terakhir adalah singkatan dari program *Perl*. *Perl* adalah sebuah aplikasi Bahasa pemrograman.

2.4 Pengujian Sistem

Pada proses perancangan perangkat lunak, perancang pertama-tama berusaha membangun perangkat lunak dari konsep abstrak ke implementasi yang dapat dilihat, baru kemudian dilakukan pengujian.

Terdapat sejumlah aturan yang berfungsi sebagai sasaran pengujian.

- a. Pengujian adalah proses eksekusi suatu program dengan maksud menemukan kesalahan.
- b. Test case yang baik adalah test case yang memiliki probabilitas tinggi untuk menemukan kesalahan yang belum pernah ditemukan sebelumnya.
- c. Pengujian yang sukses adalah pengujian yang mengungkap semua kesalahan yang belum pernah ditemukan sebelumnya.

2.4.1 Black Box Testing

Menurut Shalahuddin dan Rosa (2015), **black box testing** adalah 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 **black box testing** harus dibuat dengan kasus benar dan kasus salah.

1. Kelebihan *Black Box Testing*

a. Kesalahan Logika

Digunakan pada sintaks “If “ dan pengulangan. Dimana *black box testing* akan mendeteksi kondisi-kondisi yang tidak sesuai dan mendeteksi kapan proses pengulangan akan berhenti.

b. Ketidak Sesuaian Asumsi

Menampilkan asumsi yang tidak sesuai dengan kenyataan, untuk di analisa dan diperbaiki.

c. Kesalahan Ketik

Mendeteksi bahasa pemrograman yang bersifat case sensitive