

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Penelitian Terdahulu**

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti judul penelitian berikut. Namun penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis.

Pada penelitian sebelumnya yang dilakukan oleh (Gat, 2015) dalam jurnal yang berjudul **“Perancangan Basis Data Perpustakaan Sekolah Dengan Menerapkan Model Data Relasional”**. Permasalahan pada penelitian ini adalah perancangan basis data yang menerapkan model data relasional yang menggunakan metode penelitian dan pengembangan atau lebih dikenal dengan *Research and Development (R&D)* yang tahapan penelitiannya meliputi perancangan basis data yang menggunakan metode DBLC (*Data Base Life Cycle*), membangun model data *Conceptual database design* yang digunakan di dalam suatu perusahaan yang bersifat independen dari semua pertimbangan fisikal, Perancangan *Physical Database Design* merupakan tahapan untuk mengimplementasikan hasil perancangan database secara logis menjadi tersimpan secara fisik pada media penyimpanan eksternal sesuai dengan *Database Management System (DBMS)* yang digunakan, sehingga menghasilkan database dengan model relasional diperlukan sebagai upaya untuk menjamin dan menjaga integritas data dari setiap tabel yang berhubungan. Sedangkan *software* yang digunakan adalah Microsoft Acces dan sekaligus digunakan sebagai *database* dari perpustakaan sekolah.

Pada penelitian sebelumnya yang dilakukan oleh (Yogantoro, 2016) dalam jurnal yang berjudul **“Implementasi Basis Data Terdistribusi Dengan Metode**

***Heterogenous Distributed Database System Pada Sistem Informasi Barang Redshit Distro***". Permasalahan pada penelitian ini adalah bagaimana mengimplementasikan basis data terdistribusi dengan menggunakan metode *Heterogenous Distributed Database System Pada Sistem Informasi Barang Redshit Distro*, *software* yang digunakan ada dua jenis yaitu Oracle XE 11g dan MySQL sedangkan *hardware* yang digunakan adalah router sehingga menghasilkan *prototype* sistem informasi barang Redshit Distro.

Pada penelitian sebelumnya yang dilakukan oleh (Suko, 2016) dalam jurnal yang berjudul "**Implementasi Basis Data Multimedia Dalam Website Pariwisata Kabupaten Purworejo**". Permasalahan pada penelitian ini adalah bagaimana cara membuat sebuah website yang digunakan untuk dinas pariwisata, Website ini dibangun dengan menggunakan metode *waterfall* yang tahapan penelitiannya meliputi analisis sistem yang merupakan penguraian dari suatu sistem yang utuh ke dalam kegiatan-kegiatan komponennya, perancangan desain sistem merupakan persiapan rancang bangun implementasi yang menggambarkan bagaimana suatu sistem dibentuk berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam satu kesatuan yang utuh dan berfungsi, penulisan program atau pengkodean kemudian ujicoba program yang merupakan elemen kritis dari SQA (*Software Quality Assurance*) dan mempresentasikan tinjauan ulang yang menyeluruh terhadap spesifikasi, desain dan pengkodean dilanjutkan dengan tahap implementasi yang merupakan tahap dimana semua elemen dan aktivitas sistem disatukan sehingga menghasilkan sebuah website yang efektif/tepat guna sebagai alat/sarana untuk memperoleh informasi pariwisata dan untuk Implementasi sistem yang digunakan adalah HTML, CSS, javascript serta basis data yang digunakan adalah oracle dan penyusunan koding menggunakan Bahasa pemrograman JSP.

Pada penelitian sebelumnya yang dilakukan oleh (Sucipto, 2017) dalam jurnal yang berjudul "**Perancangan Active Database System pada Sistem Informasi Pelayanan Harga Pasar**". Permasalahan pada penelitian ini adalah bagaimana membuat *Active Database System* pada Sistem Informasi pelayanan

harga pasar, *software* yang digunakan adalah MySQL dan PostgreSQL. Penelitian ini menggunakan metode *Action Research*, penelitian yang berfokus pada tindakan sosial.

Pada penelitian sebelumnya yang dilakukan oleh (Prihatin, 2017) dalam jurnal yang berjudul **“Optimasi Query Pada Sistem Informasi Pencatatan Aktifitas Perubahan Data Nilai Mahasiswa”**. Permasalahan pada penelitian ini adalah bagaimana membuat sistem pencatatan aktifitas dan bagaimana meminimalisir waktu pencarian data menggunakan *query optimization*, dengan menggunakan metode *waterfall* yang tahapan penelitiannya meliputi *Requirement Definition* yang merupakan tahap untuk mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh sistem yang akan dibangun, sistem dan *Software Design* yang merupakan proses desain sistem mengalokasikan persyaratan baik perangkat keras atau perangkat lunak sistem dengan membentuk arsitektur sistem secara keseluruhan, *Implementation and Unit Testing* merupakan tahapan menerjemahkan desain sistem ke dalam kode program dengan menggunakan bahasa pemrograman tertentu, *Integration and Sistem Testing* merupakan tahap penyatuan unit-unit program kemudian diuji secara keseluruhan (sistem testing), *Operation and Maintenance* merupakan tahap mengoperasikan program di lingkungannya dan melakukan pemeliharaan, seperti penyesuaian atau perubahan karena adaptasi dengan situasi sebenarnya. Sistem yang dibangun kemudian diuji secara unit. sedangkan *software* yang digunakan harus berspesifikasi sistem operasi windows 8 ultimate, aplikasi local server XAMPP 3.2.1, Notepad++, Browser Google Chrome, Microsoft Visio 2010, yang dikerjakan di laptop.

Berdasarkan dari lima hasil penelitian terdahulu, semuanya membahas database yang digunakan untuk menyimpan hasil yang diinput melalui sebuah *website* dan basis datanya adalah MySQL sedangkan penulis melakukan penelitian yang datanya diinputkan melalui sebuah alat absensi yang menggunakan sensor RFID dan Sidik Jari dan hanya berfokus pada *database* yang menggunakan MySQL.

## 2.2. Pengertian Basis Data

Basis Data (*Database*) dapat dibayangkan sebagai lemari arsip. Jika kita memiliki sebuah lemari arsip dan berwenang untuk mengelolanya, atau kumpulan informasi yang terorganisasi dan disajikan untuk tujuan khusus. Prinsip utama Basis Data adalah pengaturan data/arsip. Sedangkan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan data/arsip. Sedangkan sistem Basis Data merupakan perpaduan antara basis data dan sistem manajemen basis data (DBMS). DBMS adalah *software* yang menangani semua akses ke basis data (Didinhawari, 2009).

## 2.3. Web Browser

Web browser adalah suatu program, *software*, aplikasi atau perangkat yang digunakan untuk mencari atau menjelajahi informasi di dalam jaringan internet dari sebuah media seperti situs, blog, jejaring sosial, dan lain sebagainya yang tersimpan di dalam internet. Tentunya kamu tidak asing lagi dengan nama-nama web browser seperti Chrome, Firefox, Opera, Safari dan lain-lain. (Pamungkas, 2016).



**Gambar 2.1.** Macam-Macam Web Browser

( Sumber : <https://www.computerworld.com/article/3220411/how-to-get-a-cutting-edge-web-browser.html> )

## 2.4. PHP

PHP adalah bahasa pemrograman *script server-side* yang didesain untuk pengembangan web. Selain itu, PHP juga bisa digunakan sebagai bahasa pemrograman umum (wikipedia). PHP di kembangkan pada tahun 1995 oleh Rasmus Lerdorf, dan sekarang dikelola oleh *The PHP Group*. Situs resmi PHP beralamat di <http://www.php.net>.

PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer *server*. Hal ini berbeda dibandingkan dengan bahasa pemrograman *client-side* seperti *JavaScript* yang diproses pada web browser (*client*).

Pada awalnya PHP merupakan singkatan dari *Personal Home Page*. Sesuai dengan namanya, PHP digunakan untuk membuat *website* pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga *website* populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll.

Saat ini PHP adalah singkatan dari PHP: *Hypertext Preprocessor*, sebuah kepanjangan rekursif, yakni permainan kata dimana kepanjangannya terdiri dari singkatan itu sendiri: PHP: *Hypertext Preprocessor*.

PHP dapat digunakan dengan gratis (*free*) dan bersifat *Open Source*. PHP dirilis dalam lisensi PHP *License*, sedikit berbeda dengan lisensi GNU *General Public License* (GPL) yang biasa digunakan untuk proyek *Open Source* (Andre, 2014). Pemrograman PHP bersama HTML digunakan oleh penulis untuk membuat tampilan data *report* absensi dosen dan mahasiswa serta hasil perhitungan kompensasi khusus digunakan untuk absensi mahasiswa.

## 2.5. HTML

HTML adalah singkatan dari *Hypertext Markup Language*. Disebut *hypertext* karena di dalam HTML sebuah *text* biasa dapat berfungsi lain, kita dapat membuatnya menjadi *link* yang dapat berpindah dari satu halaman ke halaman lainnya hanya dengan meng-klik *text* tersebut. Kemampuan *text* inilah yang dinamakan *hypertext*, walaupun pada implementasinya nanti tidak hanya *text* yang dapat dijadikan *link*.

Disebut *Markup Language* karena bahasa HTML menggunakan tanda (*mark*), untuk menandai bagian-bagian dari *text*. Misalnya, *text* yang berada di antara tanda tertentu akan menjadi tebal, dan jika berada di antara tanda lainnya akan tampak besar. Tanda ini di kenal sebagai HTML tag.

Jika ingin melihat bagaimana sebenarnya HTML, silahkan klik kanan halaman *browser*, lalu pilih *View Page Source* (di Mozilla Firefox atau Google Chrome). Akan tampil sebuah halaman baru yang merupakan kode HTML dari halaman sebuah website.

HTML merupakan bahasa dasar pembuatan *web*. Disebut dasar karena dalam membuat *web*, jika hanya menggunakan HTML tampilan *web* terasa hambar. Terdapat banyak bahasa pemrograman *web* yang ditujukan untuk memanipulasi kode HTML, seperti *JavaScript* dan PHP (Andre, 2017). Pemrograman HTML bersama PHP digunakan oleh penulis untuk membuat tampilan data *report* absensi dosen dan mahasiswa serta hasil perhitungan kompensasi khusus digunakan untuk absensi mahasiswa.

## 2.6. MySQL

MySQL merupakan *software database open source* yang paling populer di dunia, dimana saat ini digunakan lebih dari 100 juta pengguna di seluruh dunia. Dengan kehandalan, kecepatan dan kemudahan penggunaannya, MySQL menjadi pilihan utama bagi banyak pengembang software dan aplikasi baik di platform *web* maupun *desktop*. Pengguna MySQL tidak hanya sebatas pengguna perseorangan maupun perusahaan kecil, namun perusahaan seperti Yahoo!, Alcatel-Lucent, Google, Nokia, Youtube, Wordpress dan Facebook juga merupakan pengguna Mysql (Solichin, 2010). *Software* MySQL digunakan oleh penulis untuk membangun basis data untuk data kehadiran dosen dan mahasiswa Jurusan Teknik Komputer Politeknik Negeri Sriwijaya.



**Gambar 2.2.** Logo MySQL

( **Sumber** : <https://waterfall-security.com/solutions-archive/databases/attachment/mysql-logo>)

## 2.7. SQL

SQL adalah singkatan dari *Structured Query Language*. Sedangkan pengertian SQL adalah suatu bahasa (*language*) yang digunakan untuk mengakses data di dalam sebuah database relasional. SQL sering juga disebut dengan istilah *query*, dan bahasa SQL secara praktiknya digunakan sebagai bahasa standar untuk manajemen database relasional. Hingga saat ini hampir seluruh server database atau *software database* mengenal dan mengerti bahasa SQL (Solichin, 2010).

## 2.8. Jenis Perintah SQL

Dalam penggunaan SQL terdapat beberapa perintah yang berguna untuk mengakses dan memanajemen data yang terdapat dalam database. Jenis perintah SQL secara umum dibagi kepada tiga sub perintah, yaitu DDL (*Data Definition Language*), DML (*Data Manipulation Language*), dan DCL (*Data Control Language*). Ketiga sub perintah tersebut sangat perlu untuk dipahami bagi anda yang ingin menguasai bahasa sql dan mahir dalam pembuatan *database* (Solichin, 2010).

### 2.8.1. *Data Definition Language* (DDL)

DDL adalah sub perintah dari bahasa SQL yang digunakan untuk membangun kerangka sebuah *database*, dalam hal ini *database* dan tabel. Terdapat tiga perintah penting dalam DDL, yaitu *CREATE*, *ALTER*, *DROP*.

- *CREATE*: perintah ini digunakan untuk membuat, termasuk di dalamnya membuat *database* baru, tabel baru *view* baru, dan kolom baru.

Contoh: *CREATE DATABASE* nama\_ *database*

- *ALTER*: perintah *ALTER* berfungsi untuk mengubah struktur tabel yang telah dibuat. Mencakup di dalamnya mengubah nama tabel, menambah kolom, mengubah kolom, menghapus kolom, dan memberikan atribut pada kolom.

Contoh: *ALTER TABLE* nama\_ *tabel* *ADD* nama\_ *kolom* *datatype*

*DROP*: perintah *DROP* berfungsi untuk menghapus *database* atau tabel.

Contoh: *DROP DATABASE* nama\_ *database*

### 2.8.2. *Data Manipulation Language (DML)*

DML adalah sub perintah dari bahasa SQL yang digunakan untuk memanipulasi data dalam database yang telah dibuat. Terdapat empat perintah penting dalam DML, yaitu *INSERT*, *SELECT*, *UPDATE*, dan *DELETE*.

- *INSERT*: perintah ini digunakan untuk memasukkan data baru ke dalam sebuah tabel. Perintah ini tentu saja bisa dijalankan ketika database dan tabel sudah dibuat.

Contoh: *INSERT INTO* nama\_tabel *VALUES* (data1, data2, dst...);

- *SELECT*: perintah ini digunakan untuk mengambil dan menampilkan data dari tabel atau bahkan dari beberapa tabel dengan penggunaan relasi.

Contoh: *SELECT* nama\_kolom1, nama\_kolom2 *FROM* nama\_tabel;

- *UPDATE*: perintah update digunakan untuk memperbaharui data pada sebuah tabel.

Contoh: *UPDATE* nama\_tabel *SET* kolom1=data1, kolom2=data2,... *WHERE* kolom=data;

- *DELETE*: perintah *delete* digunakan untuk menghapus data dari sebuah tabel.

Contoh: *DELETE FROM* nama\_tabel *WHERE* kolom=data;

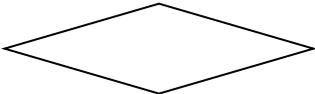
### 2.8.3. *Data Control Language (DCL)*

DCL adalah sub bahasa SQL yang berfungsi untuk melakukan pengontrolan data dan server databasenya, seperti manipulasi *user* dan hak akses (*priviledges*). Yang termasuk perintah dalam DCL ada dua, yaitu *GRANT* dan *REVOKE*.

- *GRANT*: perintah ini digunakan untuk memberikan hak akses oleh admin ke salah satu user atau pengguna. Hak akses tersebut bisa berupa hak membuat (*CREATE*), mengambil data (*SELECT*), menghapus data (*DELETE*), mengubah data (*UPDATE*), dan hak khusus lainnya yang berhubungan dengan sistem database.
- *REVOKE*: perintah ini digunakan untuk mencabut hak akses yang telah diberikan kepada user. Dalam ini merupakan kebalikan dari perintah *GRANT*.

### 2.9. ERD (*Entity Relationship Diagram*)

ERD adalah suatu model untuk menjelaskan mengenai hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang memiliki hubungan antar relasi. *Entity Relationship Diagram* digunakan untuk memodelkan struktur data serta hubungan antar data, untuk dapat menggambarkannya digunakan beberapa notasi serta simbol. Berikut simbol-simbol dari ERD (*Entity Relationship Diagram*) terlihat pada Gambar 2.3.

Simbol	Keterangan
	Entitas, yaitu kumpulan dari objek yang dapat diidentifikasi secara unik.
	Relasi, yaitu hubungan yang terjadi antara salah satu lebih entitas. Jenis hubungan antara lain. <i>One to one, One to many, dan many to many.</i>
	Atribut, yaitu karakteristik dari entitas atau relasi yang merupakan penjelasan detail tentang entitas.
	Hubungan antara entitas dengan atributnya dan himpunan entitas dengan himpunan relasinya

**Gambar 2.3.** Simbol ERD (*Entity Relationship Diagram*)

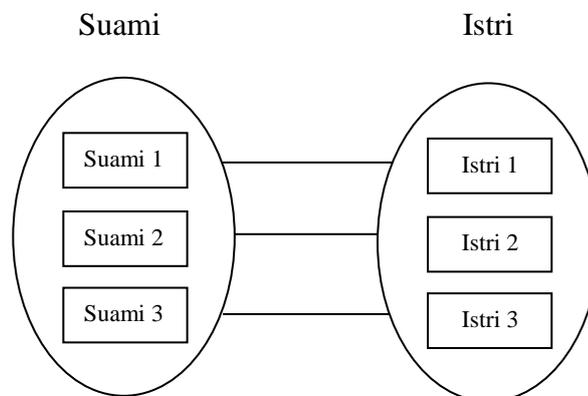
( Sumber : <http://mfikri.com/assets/images/00a1b09e500687b60f3c347aa67ab7fa.jpg>)

## 2.10.Kardinalitas

Kardinalitas merupakan objek dasar dan relasi antara entitas. Entitas dapat diartikan sebagai objek dan diidentifikasi secara unik, dan objeknya dapat berbentuk orang, barang, dan sebagainya. Kardinalitas relasi menunjukkan maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas lain. Kardinalitas relasi yang terjadi di antara dua himpunan entitas dapat berupa satu ke satu (*one to one*), satu ke banyak (*one to many*), dan banyak ke banyak (*many to many*).

### 2.10.1. Satu ke satu (*one to one*)

Hubungan satu ke satu (*one to one*) berarti setiap himpunan entitas hanya boleh berhubungan dengan satu himpunan entitas lainnya. Sebagai contoh himpunan suami dan istri berikut:



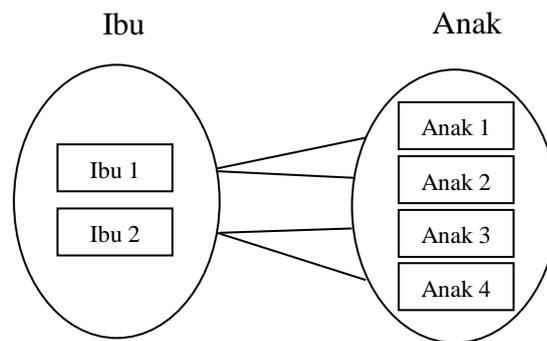
**Gambar 2.4.** Kardinalitas satu ke satu (*one to one*)

( Sumber : <http://mfikri.com/assets/images/8400d4f078e0c06d6f29f4b972cf4fe4.jpg>)

Pada gambar diatas dapat dilihat bahwa satu himpunan entitas suami hanya berhubungan tepat dengan satu himpunan entitas istri. Dalam arti kata suami atau istri tidak boleh selingkuh.

### 2.10.2. Satu ke banyak (*one to many*)

Hubungan satu ke banyak (*one to many*) berarti satu dari setiap himpunan entitas boleh berhubungan dengan banyak himpunan entitas lainnya. Sebagai contoh himpunan ibu dan anak berikut:



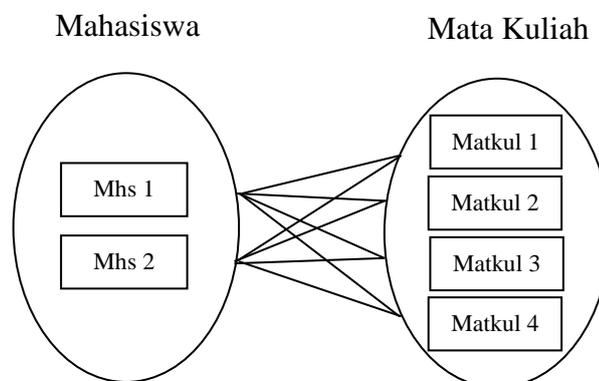
**Gambar 2.5.** Kardinalitas satu ke banyak (*one to many*)

( Sumber : <http://mfikri.com/assets/images/e59eb3c0dad705ab197735ece0bed9d9.jpg>)

Pada gambar diatas dapat dilihat bahwa satu himpunan ibu memiliki banyak hubungan ke himpunan entitas anak. Dalam arti kata satu ibu bisa memiliki banyak anak dan satu anak hanya dimiliki oleh satu ibu.

### 2.10.3. Banyak ke banyak (*many to many*)

Hubungan banyak ke banyak (*many to many*) berarti setiap himpunan entitas boleh berhubungan dengan banyak himpunan entitas lainnya dan sebaliknya. Sebagai contoh himpunan matakuliah dan mahasiswa berikut:



**Gambar 2.6.** Kardinalitas banyak ke banyak (*many to many*)

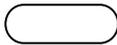
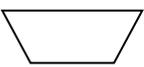
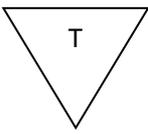
( Sumber : <http://mfikri.com/assets/images/08f801a4f0c7133cc6f6302f6c60abb7.jpg>)

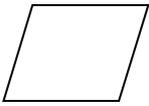
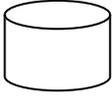
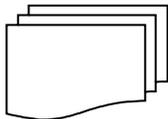
Pada gambar diatas dapat dilihat bahwa satu himpunan mahasiswa memiliki banyak hubungan ke himpunan entitas matakuliah dan satu dari himpunan matakuliah memiliki banyak hubungan ke himpunan entitas mahasiswa. Dalam arti kata satu mahasiswa bisa memiliki banyak matakuliah dan satu matakuliah bisa dimiliki oleh banyak mahasiswa.

### 2.11.Flowchart

*Flowchart* atau diagram alir merupakan sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran algoritma atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing-masing langkah tersebut menggunakan tanda panah. Diagram ini bisa memberi solusi langkah demi langkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut (Fadhillah, 2017).

**Tabel 2.2** Simbol-Simbol *Flowchart*

No	Simbol	Pengertian	Keterangan
1.		Mulai / berakhir ( <i>Terminal</i> )	Digunakan untuk memulai, mengakhiri, atau titik henti dalam sebuah proses atau program; juga digunakan untuk menunjukkan pihak eksternal.
2.		Dokumen	Sebuah dokumen atau laporan; dokumen dapat dibuat dengan tangan atau dicetak oleh komputer.
3.		Kegiatan Manual	Sebuah kegiatan pemrosesan yang dilaksanakan secara manual.
4.		Arsip	Arsip dokumen disimpan dan diambil secara manual. Huruf didalamnya menunjukkan cara pengurutan arsip: N = Urut Nomor; A = Urut Abjad; T = Urut Tanggal.

5.		Input / Output; Jurnal / Buku Besar	Digunakan untuk menggambarkan berbagai media input dan output dalam sebuah bagan alir program.
6.		Disk Bermagnet	Data disimpan secara permanen pada disk bermagnet.
7.		Penghubung Pada Halaman Berbeda	Menghubungkan bagan alir yang berada di halaman yang berbeda.
8.		Pemasukan Data On Line	Entri data alat oleh on line seperti terminal CRT dan komputer pribadi.
9.		Pemrosesan Komputer	Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer biasanya menghasilkan perubahan terhadap data atau informasi
10.		Arus Dokumen atau Pemrosesan	Arus dokumen atau pemrosesan; arus normal adalah ke kanan atau ke bawah.
11.		Keputusan	Sebuah tahap pembuatan keputusan
12.		Penghubung Dalam Sebuah Halaman	Menghubungkan bagan alir yang berada pada halaman yang sama.
13.		Dokumen Rangkap	Digambarkan dengan menupuk simbol dokumen dan pencetakan nomor dokumen dibagian depan dokumen pada bagian kiri atas.

( Sumber : Syafitri, 2011)