



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Sujatmiko (2012:156), “Komputer adalah mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program. Alat serbaguna ini memegang peran penting dalam teknologi komunikasi”.

“Computer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer” (Asropudin, 2013:19).

Berdasarkan pendapat diatas penulis menyimpulkan bahwa komputer adalah sebuah alat elektronik yang dapat memproses data dengan cepat sesuai dengan program atau perintah untuk menghasilkan informasi.

2.1.2 Pengertian Data

Asropudin (2013:22) mengemukakan bahwa, “Data merupakan kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi”.

McLeod (dikutip Yakub, 2012:5) menyatakan bahwa, “Data adalah kenyataan yang menggambarkan adanya suatu kejadian (*event*), data terdiri dari fakta (*fact*) dan angka yang secara relatif tidak berarti bagi pemakai”.

Berdasarkan pendapat diatas penulis menyimpulkan bahwa data adalah kumpulan fakta-fakta yang berupa angka, huruf, dan karakter lainnya yang memiliki nilai dan diperoleh dari suatu kejadian dan dapat diolah sehingga menghasilkan informasi.

2.1.3 Pengertian Sistem

Sutabri (2012:6) menyatakan, “Sistem adalah sekelompok unsur yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu”.



McLeod (dikutip Yakub, 2012:01) menyatakan bahwa “Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan”.

Berdasarkan pernyataan diatas penulis menyimpulkan sistem merupakan sekelompok unsur yang saling berhubungan, saling berinteraksi dan saling bergantung antara satu dengan yang lainnya untuk mencapai suatu tujuan.

2.1.4 Pengertian Informasi

Sujatmiko (2012:135), “Informasi adalah serangkaian simbol yang jika disusun menurut aturan tertentu dapat menyampaikan suatu pesan”.

Sutabri (2012:22), “Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan”.

Dari pernyataan diatas penulis dapat menyimpulkan bahwa informasi adalah kumpulan data yang telah diolah menjadi bentuk yang memiliki arti sehingga dapat digunakan dalam proses pengambilan keputusan oleh penggunanya.

2.1.5 Pengertian Sistem Informasi

O’Brian (dikutip Yakub, 2012:8), “Sistem informasi (*information system*) merupakan kombinasi teratur dari orang-orang, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi”.

Sujatmiko (2012:251) menyatakan, “Sistem Informasi adalah sistem pengiriman data melalui fasilitas telekomunikasi dari satu lokasi ke pusat pengolahan data tetapi data yang dikirimkan tidak langsung diproses oleh CPU”.

Berdasarkan pendapat diatas penulis menyimpulkan sistem informasi merupakan kumpulan komponen yang saling berhubungan dan mengolahnya menjadi informasi.



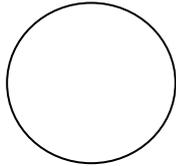
2.2 Teori Khusus

2.2.1 Data Flow Diagram (DFD)

Sukamto dan M. Shalahuddin (2016:70) mengemukakan bahwa “*Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

Sukamto dan M. Shalahuddin (2016:71), menjelaskan notasi yang digunakan dalam membuat DFD ada empat buah, yaitu sebagai berikut:

Tabel 3.1 Simbol-simbol *Data Flow Diagram*

Notasi	Keterangan
	<p>Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. catatan: nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>
	<p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM)). catatan: nama yang diberikan pada sebuah penyimpanan biasanya berupa kata benda.</p>

Lanjutan Tabel 2.1 Simbol-simbol *Data Flow Diagram*

Notasi	Keterangan
	<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>catatan: nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda.</p>
	<p>Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).</p> <p>catatan: nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”.</p>

(Sumber : Rosa A. S, dan M. Shalahuddin, 2016:71–72)

Sukanto dan M. Shalahuddin (2016:72), berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

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

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

2. Membuat DFD Level 1

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



3. Membuat DFD Level 2

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

4. Membuat DFD Level 3 dan seterusnya

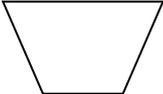
DFD Level 3, 4, 5, dan seterusnya merupakan *breakdown* dari modul pada DFD Level di-atasnya. *Breakdown* pada level 3, 4, 5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.2.2 Block Chart

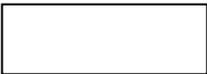
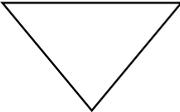
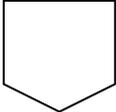
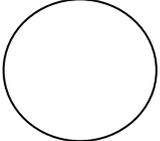
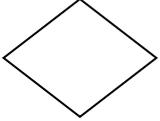
Kristanto (2008:68) mengemukakan bahwa, “*Block Chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *block chart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

Adapun simbol-simbol *Block Chart* adalah sebagai berikut:

Tabel 2.2 Simbol-Simbol *Block Chart*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
2.		Multi dokumen
3.		Proses manual

Lanjutan Tabel 2.2 Simbol-Simbol *Block Chart*

No.	Simbol	Keterangan
4.		Proses yang dilakukan oleh komputer
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (<i>data storage</i>)
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (<i>decision</i>)

Lanjutan Tabel 2.2 Simbol-Simbol *Block Chart*

No.	Simbol	Keterangan
12.		Layar peraga (<i>monitor</i>)
13.		Pemasukan data secara manual

(Sumber: Kristanto, 2008:68)

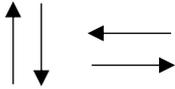
2.2.3 Flowchart

Ladjamudin (2013:211) mengemukakan bahwa, “*Flowchart* adalah bagan – bagan yang mempunyai arus yang menggambarkan langkah – langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma”.

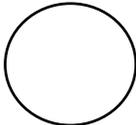
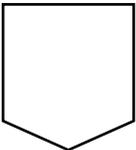
Ladjamudin (2013:266) mengemukakan bahwa, Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yaitu sebagai berikut :

1. *Flow Direction Symbols* (Simbol penghubung/alur)
2. *Processing Symbols* (Simbol Proses)
3. *Input-output Symbols* (Simbol input-output)

Tabel 2.3. Simbol-Simbol *Flow Direction*

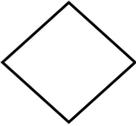
No.	Simbol	Arti
1.		Simbol arus / flow untuk menyatakan jalannya arus suatu proses
2.		Simbol Communication Link untuk menyatakan bahwa adanya transisi suatu data/informasi dari satu lokasi ke lokasi lainnya

Lanjutan Tabel 2.3. Simbol-Simbol *Flow Direction*

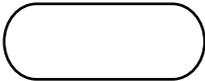
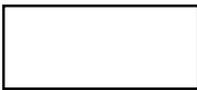
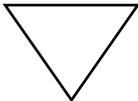
No.	Simbol	Arti
3.		Simbol Connector untuk menyatakan sambungan dari satu proses ke proses yang lainnya dalam halaman/lembar yang sama
4.		Simbol Offline Connector untuk menyatakan sambungan dari satu proses ke proses yang lainnya dalam halaman/lembar yang berbeda.

(Sumber: Ladjamudin, 2013:266)

Tabel 2.4. Simbol-Simbol *Flow Direction*

No.	Simbol	Arti
1.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.
2.		Simbol Manual Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.
3.		Simbol Decision/logika Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/tidak.

Lanjutan Tabel 2.4. Simbol-Simbol *Flow Direction*

No.	Simbol	Arti
4.		Simbol Predefined Process Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5.		Simbol Terminal Untuk menyatakan permulaan atau akhir suatu program.
6.		Simbol Keying Operation Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
7.		Simbol off-line storage Untuk menunjukkan bahwa data dalam symbol ini akan disimpan kedalam suatu media tertentu.
8.		Simbol Manual input Untuk memasukkan data secara manual dengan menggunakan online keyboard.

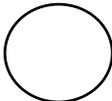
(Sumber: Ladjamudin, 2013:267)

Tabel 2.5 Simbol-Simbol Input-output

No.	Simbol	Arti
1.		Simbol Input/Output Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.



Lanjutan Tabel 2.5 Simbol-Simbol Input-output

No.	Simbol	Arti
2.		Simbol Punched Card Untuk menyatakan input berasal dari kartu atau output ditulis ke kartu.
3.		Simbol Magnetic-tape unit Untuk menyatakan input berasal dari pita magnetic atau output disimpan ke pita magnetic.
4.		Simbol Disk Storage Untuk menyatakan input berasal dari disk atau output disimpan ke disk.
5.		Simbol Document Untuk mencetak laporan ke printer
6.		Simbol Display Untuk menyatakan peralatan output yang digunakan berupa layar (video, komputer).

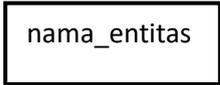
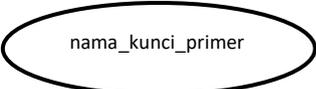
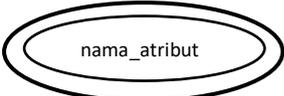
(Sumber: Ladjamudin, 2013:268)

2.2.4 Entity Relationship Diagram (ERD)

Sukanto dan M. Shalahuddin (2016:50), “Pemodelan awal basis data yang paling banyak digunakan adalah *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.”

Sukanto dan M. Shalahuddin (2016:50), menjelaskan simbol-simbol yang digunakan dalam ERD, yaitu:


Tabel 2.6 Simbol-simbol *Entity Relationship Diagram* (ERD)

No.	Simbol	Deskripsi
1.	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
4.	Atribut multivalai/ <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.

Lanjutan Tabel 2.6 Simbol-simbol *Entity Relationship Diagram* (ERD)

No.	Simbol	Deskripsi
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.
6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan one to many menghubungkan entitas A dan entitas B maka

(Sumber : Rosa A. S, dan M. Shalahuddin, 2016:50–51)

2.2.5 Kamus Data (*Data Dictionary*)

Sukamto dan M. Shalahuddin (2016:73), “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan)”.

Sukamto dan Shalahuddin (2016:74), menjelaskan simbol-simbol yang digunakan dalam kamus data, yaitu:

**Tabel 2.7** Simbol-simbol dalam Kamus Data

No	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik...atau...
4.	{ } ⁿ	n kali diulang/ bernilai banyak
5.	()	Data opsional
6.	*...*	Batas komentar

(Sumber : Rosa A. S, dan M. Shalahuddin, 2016:74)

2.2.6 Daftar Kejadian (*Event List*)

Kristanto (2008:64), Daftar kejadian digambarkan dalam bentuk kalimat sederhana dan berfungsi untuk memodelkan kejadian yang terjadi dalam lingkungan sehari-hari dan membutuhkan tanggapan atau respon dari sistem. Suatu kejadian mewakili satu aliran data atau proses dalam diagram konteks serta deskripsi penyimpanan yang digunakan untuk memodelkan data harus diperhatikan dalam kaitannya dengan daftar kejadian.

Adapun cara-cara mendeskripsikan daftar kejadian adalah sebagai berikut:

1. Pelaku adalah entiti luar, jadi bukan sistem.
2. Menguji setiap entiti luar dan mencoba mengevaluasi setiap entiti luar yang terjadi pada sistem.
3. Hati-hati dengan kejadian yang spesifik, yang tak sengaja menyatu dalam paket yang sama.
4. Harus diingat bahwa kejadian yang dimodelkan bukan hanya interaksi normal antara sistem dengan entiti luar, karena itu harus dievaluasi kebutuhan sistem untuk menanggapi kejadian yang gagal.
5. Setiap aliran keluaran sebaiknya merupakan respondari kejadian.



6. Setiap kejadian yang tidak berorientasi pada waktu dalam daftar kejadian sebaiknya mempunyai masukan sehingga sistem dapat mendeteksi kejadian yang berlangsung.
7. Setiap kejadian sebaiknya menghasilkan keluaran langsung sebagai respon atau disimpan dalam berkas untuk bahan masukan.

2.2.7 Metode Pengembangan Sistem

Sukanto dan M. Shalahuddin (2016:28), menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut mulai dari analisis, disain, pengodean, pengujian, dan tahap pendukung (*support*).

- a. Analisis kebutuhan perangkat lunak

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

- b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada disain pembuatan program sistem termasuk struktur data, arsitektur sistem, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

- c. Pembuatan kode program

Pada tahap pengodean, desain harus ditranlasikan ke dalam program sistem. Hasil dari tahap ini adalah program komputer sesuai dengan disain yang telah dibuat pada tahap desain.

- d. Pengujian

Tahap pengujian fokus pada sistem dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.



e. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah sistem mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau sistem harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.

2.2.8 Pengujian *Black-Box Testing* (Pengujian Kotak Hitam)

Sukanto dan M. Shalahuddin (2016:275), “Pengujian *Black-Box Testing* yaitu menguji perangkat lunak dari 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”.

2.3 Pengertian Judul

2.3.1 Pengertian Aplikasi

Sujatmiko (2012:23) menyatakan, “Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya Ms-Word, Ms-Excel”.

Sutabri (2012:147) menyatakan bahwa, “Aplikasi adalah alat terapan yang di fungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya”.

Berdasarkan pernyataan diatas penulis menyimpulkan aplikasi adalah alat untuk membantu manusia memudahkan suatu pekerjaan.

2.3.2 Pengertian Pelayanan

Mounir (dikutip Majid, 2009:34), “Pelayanan merupakan rangkaian kegiatan untuk memenuhi kebutuhan pelanggan, atas jasa yang mereka dapatkan dari suatu perusahaan”.



2.3.3 Pengertian Administrasi

Syafri (2012:3) mengemukakan bahwa “Secara *etimologis*, administrasi berasal dari bahasa Latin *ad* dan *ministrare*, yang berarti membantu, melayani, atau memenuhi, serta *administratio* yang berarti pemberian bantuan, pemeliharaan, pelaksanaan, pimpinan dan pemerintahan, pengelolaan”.

Sondang P. Siagian (dikutip Syafri, 2012:9), menyatakan bahwa “Administrasi didefinisikan sebagai keseluruhan proses kerja sama antara dua orang manusia atau lebih yang didasarkan atas rasionalitas tertentu untuk mencapai tujuan yang telah ditentukan sebelumnya”.

2.3.4 Pengertian Terpadu

Kamus Bahasa Indonesia (2008: 1101) mengemukakan bahwa “terpadu merupakan sudah dipadu (disatukan, dileburkan menjadi satu dsb)”.

2.3.5 Pengertian Kecamatan

Bastian (2015:51) mengemukakan bahwa “Kecamatan merupakan perangkat daerah kabupaten/kota, yaitu pelaksana teknis kewilayahan yang mempunyai wilayah kerja tertentu dan dipimpin oleh Camat. Camat berkedudukan di bawah dan bertanggung jawab kepada bupati/walikota”.

2.3.6 Pengertian Pelayanan Administrasi Terpadu Kecamatan (PATEN)

Pelayanan Administrasi Terpadu Kecamatan (PATEN) merupakan suatu kegiatan penyelenggaraan pelayanan publik di kecamatan yang mulai dari tahap permohonan sampai ke tahap terbitnya dokumen dilakukan dalam satu tempat (meja atau loket pelayanan).

2.3.6.1 Tujuan PATEN

Tujuan PATEN yakni dalam rangka meningkatkan kualitas pelayanan dengan cara mendekatkan pelayanan kepada masyarakat, sehingga dilihat dari aspek biaya lebih efisien dan jangka waktu penyelesaian akan lebih cepat dan terukur sesuai standar pelayanan.



2.3.6.2 Maksud Penyelenggaraan PATEN

Berdasarkan Peraturan Menteri Dalam Negeri Nomor 4 Tahun 2010 Tentang Pedoman Pelayanan Administrasi Terpadu Kecamatan, maksud penyelenggaraan PATEN adalah mewujudkan Kecamatan sebagai pusat pelayanan masyarakat dan menjadi simpul pelayanan bagi kantor/badan pelayanan terpadu di kabupaten/kota.

2.3.6.3 Pengertian Retribusi Daerah

Berdasarkan Peraturan Daerah Kabupaten Banyuasin Nomor 20 Tahun 2011 Tentang Retribusi Perizinan Tertentu, mengemukakan bahwa Retribusi Daerah, yang selanjutnya disebut Retribusi, adalah pungutan Daerah sebagai pembayaran atas jasa atau pemberian izin tertentu yang khusus disediakan dan/atau diberikan oleh Pemerintah Daerah untuk kepentingan orang pribadi atau Badan.

2.3.7 Pengertian Aplikasi Pelayanan Administrasi Terpadu Kecamatan (PATEN) pada Kantor Camat Gandus Kota Palembang

Aplikasi Pelayanan Administrasi Terpadu Kecamatan (PATEN) pada Kantor Camat Gandus Kota Palembang merupakan suatu aplikasi yang memberikan kemudahan bagi masyarakat Kecamatan Gandus untuk mengetahui semua jenis Pelayanan Administrasi Terpadu Kecamatan beserta persyaratannya. Aplikasi ini memudahkan masyarakat dalam mengajukan permohonan Pelayanan Administrasi Terpadu Kecamatan Gandus, proses kepengurusan pelayanan bisa dilakukan secara online dan masyarakat tidak perlu berulang-kali datang ke Kantor Camat Gandus karena dengan adanya aplikasi ini masyarakat dapat melihat status konfirmasi pelayanan yang mereka ajukan dan masyarakat dapat mendatangi Kantor Camat Gandus jika sudah diberikan konfirmasi selesainya pelayanan.



2.3.8 Pengertian Web

Sujatmiko (2012:317) menyatakan, “*Web* adalah salah satu aplikasi *internet* yang terdiri dari perangkat lunak, kumpulan protokol, dan seperangkat aturan yang memungkinkan untuk mengakses informasi di *internet*”.

Risnandar (2013:12) menyatakan, “*Web* adalah salah satu fitur dari *internet* di samping fitur-fitur lain seperti *e-mail*, *remote (SSH)*, dan *ftp*”.

Berdasarkan pernyataan diatas penulis menyimpulkan *Web* adalah salah satu aplikasi internet.

2.4 Teori Program

2.4.1 Pengertian Bahasa Pemrograman PHP (*Pre Hypertext Processor*)

Hidayat (2011:115) mengemukakan, “*PHP* adalah bahasa pemrograman *OpenSource* yang sangat banyak digunakan oleh situs *web(server site scripting)*, untuk memberikan tampilan yang menarik dan proses *scripting* lain pada situs *web*”.

Hidayatullah dan Jauhari Khairul Kawistara (2014:231) mengemukakan, “*PHP Hypertext Preprocessor* atau disingkat dengan *PHP* ini adalah suatu bahasa *scripting* khususnya digunakan untuk *web development*”.

Dari pendapat diatas penulis menyimpulkan bahwa *PHP* adalah bahasa pemrograman yang menerjemahkan baris kode sumber menjadi kode mesin sehingga dapat dibaca oleh komputer dan digunakan untuk membuat aplikasi berbasis *web*.

2.4.2 Pengertian HTML (*Hypertext Markup Language*)

Hidayatullah dan Jauhari Khairul Kawistara (2014:13) mengemukakan, “*Hypertext Markup Language (HTML)* adalah bahasa *standard* yang digunakan untuk menampilkan halaman *web*”.

2.4.3 Tag dan Elemen HTML

Hidayatullah dan Jauhari Khairul Kawistara (2014:13), “Penanda perintah di dalam *HTML* biasanya disebut *TAG*”. *TAG* digunakan untuk menentukan tampilan dari dokumen *HTML*.



Tag ada yang berpasangan dengan format berikut :

```
<BEGIN TAG>isi dokumen </END TAG>
```

Tag ada juga yang tidak berpasangan dengan format berikut :

```
<TAG>isi dokumen
```

Contoh : Setiap dokumen HTML diawali dan diakhiri dengan tag HTML.

```
<html>
  <head>
    . . . . .
  </head>
  <body>
    . . . . .
  </body>
</html>
```

2.4.4 Pengertian XAMPP

Hidayat (2011:20), “XAMPP merupakan tool yang menyediakan paket perangkat lunak (*software*) ke dalam satu buah paket”. Berikut ini adalah cara untuk menguji xampp:

- Bukalah program *browser* yang anda miliki (Mozilla Firefox, Opera, Google Chrome, Internet Explorer dan lainnya).
- Dalam jendela *browser*, ketikkan *localhost* pada bagian *address* dalam *browser* kemudian tekan *Enter*.
- Anda akan ditempatkan pada *splash page* untuk memilih opsi bahas yang akan dipakai dalam XAMPP. Pilih *English*.
- Selanjutnya cek instalasi PHP apakah berhasil atau belum dengan cara klik menu *phpinfo()* pada halaman awal XAMPP.
- Jika ingin melihat *database* MySQL dalam XAMPP, maka Anda bisa mengetikkan *localhost/phpmyadmin* pada browser Anda, atau klik *phpmyAdmin* dalam halaman awal XAMPP.



2.4.5 Pengertian Notepad++

MADCOMS (2016:15-16), Notepad++ adalah sebuah *text editor* yang sangat berguna bagi setiap orang dan khususnya bagi para developer dalam membuat program. Notepad++ menggunakan komponen Scintilla untuk dapat menampilkan dan menyuntingkan teks dan berkas kode sumber berbagai bahasa pemrograman yang berjalan di atas sistem operasi Microsoft Windows.

2.4.6 Pengertian CSS (*Cascading Style Sheet*)

Bekti (2015:47) menyatakan, “CSS (*Cascading Style Sheet*) merupakan salah satu bahasa pemrograman web yang digunakan untuk mempercantik halaman web dan mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam.”

2.4.7 Pengertian jQuery

Bekti (2015:59) menyatakan, “jQuery merupakan suatu librari JavaScript yang memungkinkan Anda untuk membuat program web pada suatu halaman web tanpa harus menambahkan *event* atau pun *property* pada halaman web tersebut .”

2.4.8 Pengertian *PhpMyAdmin*

Hidayatullah dan Jauhari Khairul Kawistara (2014:184) mengemukakan bahwa “ *phpMyAdmin* adalah *tool open source* yang ditulis dalam bahasa PHP untuk menangani administrasi MySQL berbasis *World Wide Web*[40].”

Cara membuka *phpMyAdmin* adalah sebagai berikut:

1. Bukalah XAMPP Control Panel v3.0.2 dengan cara klik kanan *xampp-control-3 beta* di C:\xampp, kemudian Run As Administrator.
2. Setelah itu tekan **Start** pada baris Apache, tekan **Start** pada baris MySQL, kemudian tekan **Admin** pada baris MySQL.