



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

Pada Teori Umum ini akan dibahas sebagai berikut:

2.1.1 Pengertian Komputer

Menurut Rianto (2021:3), “Komputer merupakan suatu perangkat elektronika yang memiliki kemampuan untuk menerima dan mengolah data menjadi informasi, menjalankan program yang tersimpan dalam memori, serta dapat bekerja secara otomatis berdasarkan perangkat aturan tertentu.”

Menurut Krisbiantoro (2018:1), “Komputer merupakan alat yang dipakai untuk mengolah dan memproses data menurut perintah yang telah dirumuskan”.

Jadi, dapat disimpulkan bahwa Komputer adalah sebuah alat elektronik yang dapat mengelola data menjadi informasi yang berguna dalam mempermudah pekerjaan manusia.

2.1.2 Pengertian Perangkat Lunak

Menurut Romindo dkk (2023:1), “Perangkat lunak (*software*) merupakan semua perintah yang digunakan untuk mengolah data dan informasi.”

Menurut Rianto (2021:5), “*Software* (Perangkat Lunak), merupakan suatu data yang diprogram sedemikian rupa dan disimpan dalam bentuk digital yang tidak terlihat secara fisik tetapi tersimpan dalam media penyimpanan komputer.”

Jadi, dapat disimpulkan bahwa Perangkat Lunak adalah sebuah program komputer yang dibuat untuk membantu mempermudah para pengguna atau *user* dalam mengerjakan tugas-tugas tertentu.

2.1.3 Pengertian Data

Menurut Arhami dan Nasir (2020:15), “Data merupakan fakta dan statistik yang telah dikumpulkan secara bersama-sama untuk digunakan dalam berbagai



macam analisis atau dijadikan sebagai referensi-referensi dalam mendukung berbagai macam penelitian atau pendapat-pendapat”.

Sedangkan menurut Kadir (2020:2), “Data adalah suatu bahan mentah, yang kelak dapat diolah lebih lanjut menjadi sesuatu yang lebih bermakna”.

Jadi, dapat disimpulkan bahwa Data adalah sekumpulan fakta yang dapat dijadikan bahan untuk diolah sehingga menjadi suatu informasi yang berguna.

2.1.4 Metode Pengembangan Aplikasi

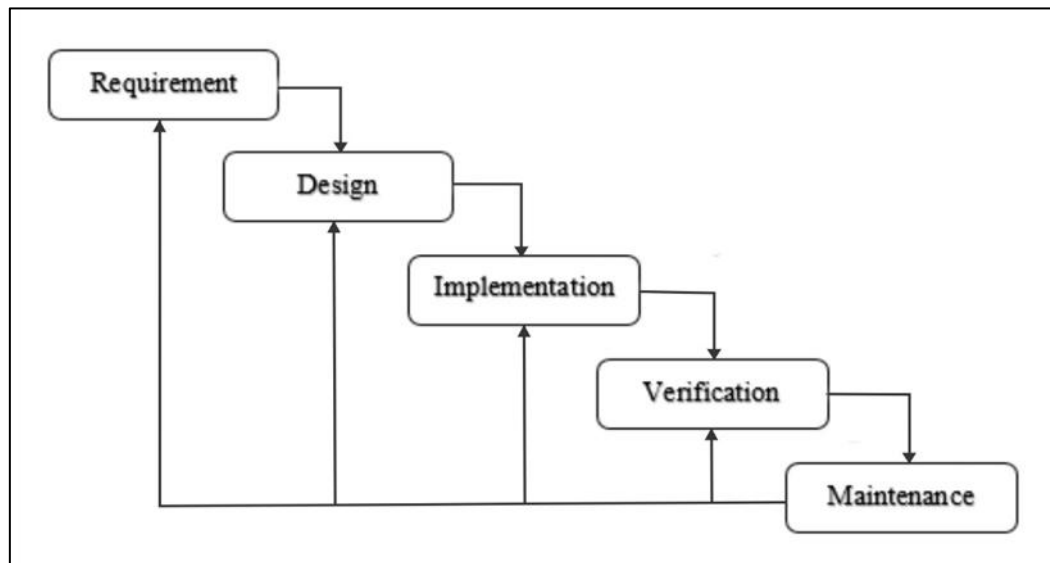
Metode yang penulis pilih sebagai dasar untuk menentukan pembuatan aplikasi berdasarkan model pengembangan perangkat lunak yaitu model air terjun (*waterfall*).

Menurut Wahid (2020:1), “Model *Waterfall* merupakan salah satu model *SDLC* (*System Development Life Cycle*) yang sering digunakan dalam pengembangan sistem informasi atau perangkat lunak. Model ini menggunakan pendekatan sistematis dan berurutan. Tahapan dalam model ini dimulai dari tahap perencanaan hingga tahap pengelolaan (*maintenance*) dan dilakukan secara bertahap. Pengembang perlu mengetahui lebih lanjut tentang bagaimana proses pengembangan sistem jika menggunakan model *waterfall* dan juga karakteristik dari model *waterfall* tersebut.

Menurut Rosa dan Shalahuddin (2018:28), “Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain pengodean, pengujian dan tahap pendukung (*support*)”.

Jadi, dapat disimpulkan bahwa Metode *Waterfall* adalah suatu proses atau metode pengembangan sistem secara berurutan yang mengalir ke bawah (seperti air terjun) melalui tahapan-tahapan *Requirement Analysis, Design System, Implementation, Verification & Testing, dan Maintenance*.

Tahapan metode *waterfall* dapat dilihat pada gambar di bawah ini:



Gambar 2.1 Tahapan-Tahapan Metode *Waterfall*

Berdasarkan gambar di atas, pengembangan sistem dengan menggunakan metode waterfall memiliki beberapa tahapan yang berurutan yaitu: *Requirement* (analisis kebutuhan), *Design System* (desain sistem), *Coding* (pengkodean) & *Testing* (pengujian), Penerapan Program, Pemeliharaan. Tahapan-tahapan dari metode *waterfall* adalah sebagai berikut:

a) *Requirement Analisis*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

b) *System Design*

Spesifikasi kebutuhan tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem dipersiapkan. Desain Sistem membantu dalam menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.



c) *Implementation*

Tahap ini dimana sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit *testing*.

d) *Integration & Testing*

Seluruh unit yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing unit. Setelah integrasi seluruh sistem diuji untuk mengecek setiap kegagalan maupun kesalahan.

e) *Operation & Maintenance*

Operation & Maintenance merupakan tahap akhir dalam model *waterfall*. Perangkat lunak yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaiki implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

2.1.5 Pengujian Aplikasi *Black Box Testing*

Menurut Ambarsari, dkk (2021:9642), "*Black Box Testing* adalah hanya menguji fungsionalitas dan interface (antarmuka) tanpa mengetahui proses yang detail dan hanya dapat mengetahui input dan outputnya saja".

Menurut Mustaqbal (dalam Setiyani, 2019:21), *Black Box Testing* merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak. Pengujian *black box testing* bertujuan untuk menemukan fungsi yang tidak benar, kesalahan antarmuka, kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi".

Jadi, dapat disimpulkan bahwa Black Box Testing adalah pengujian yang dilakukan bertujuan untuk mengetahui dan mengamati hasil *input* dan *output* dari perangkat lunak tanpa mengetahui prosesnya.



2.2 Teori Khusus

Pada Teori Khusus akan dibahas sebagai berikut:

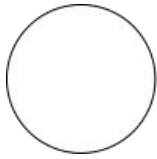
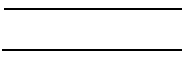
2.2.1 Data Flow Diagram (DFD)

Menurut Rosa dan M. Shalahudin (2018:70), menjelaskan tentang *Data Flow Diagram (DFD)* sebagai berikut :


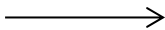
Data Flow Diagram (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).diaplikasikan

Rosa dan M. Shalahudin (2018:71) menjelaskan notasi pada DFD adalah sebagai berikut:

Tabel 2.1 Simbol-simbol *Data Flow Diagram (DFD)*

Notasi	Nama Simbol	Keterangan
	Proses	Proses/fungsi/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.
	Data Storage	File 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 (ERD)</i> , <i>Conceptual Data Model (CDM)</i> , <i>Physical Data Model (PDM)</i>).
	Entitas Luar	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.
	Aliran Data	Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau (<i>output</i>).

(Sumber: Rosa dan M. Shalahudin (2018:71))

Rosa dan M. Shalahudin (2018:72) menjelaskan tentang tahapan-tahapan perancangan dengan menggunakan *Data Flow Diagram (DFD)* yaitu :

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. Modul mana saja yang harus di-breakdown lebih detail tergantung pada kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu 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

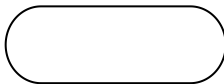
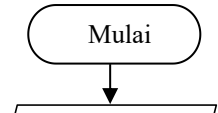

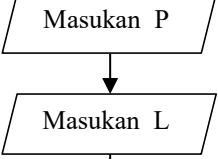
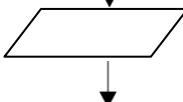
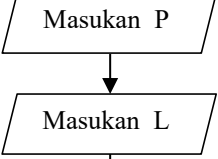
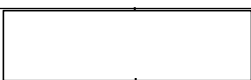
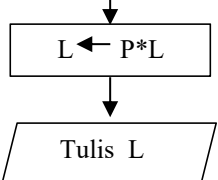
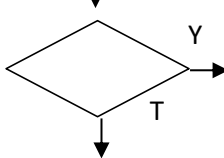
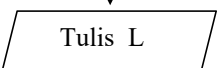
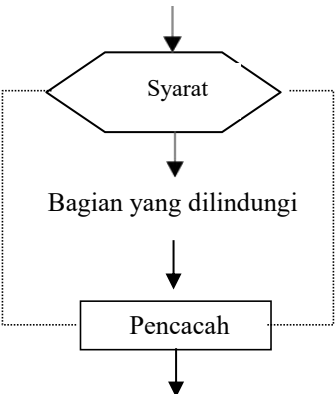
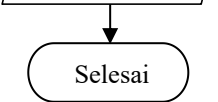
DFD Level 3,4,5, dan seterusnya merupakan breakdown dari modul pada DFD Level di-atasnya. Breakdown pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.2.2 *Flowchart*

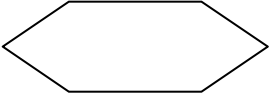

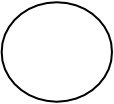
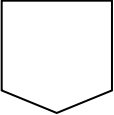
Pratiwi (2020:14) menjelaskan, “*Flowchart* adalah bentuk gambar/diagram yang mempunyai aliran satu atau dua arah secara sekuensial. *Flowchart* digunakan untuk merepresentasikan maupun desain program. Oleh karena itu *Flowchart* harus bisa merepresentasikan komponen-komponen dalam bahasa pemrograman”.

Tabel berikut memberikan penjelasan singkat tentang simbol-simbol *Flowchart* :

Tabel 2.2 Simbol-simbol *Flowchart*

No.	Keterangan	Lambang	Contoh
1.	Mulai/Selesai (Terminator)		
2.	Aliran Data		
3.	Input/Output		
4.	Proses		
5.	Percabangan		
6.	Perulangan		



7.	Preparation (Pemberian nilai awal suatu variabel)		
8.	Call (Memanggil suatu prosedur/fungsi)		
9.	Titik <i>connector</i> yang berada di halaman yang sama		
10.	Titik konektor yang berada di halaman lain		


(Sumber: Pratiwi (2020:15))

2.2.3 Blockchart

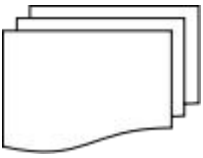


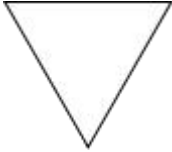
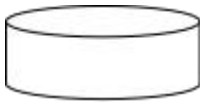

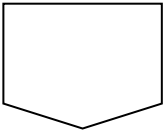
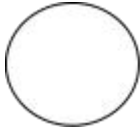
Menurut Kristanto (2018:75) “*Block Chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.”

Adapun simbol-simbol yang sering digunakan dalam *blockchart*, dapat dilihat pada tabel berikut ini:

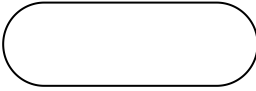
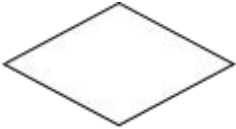


Tabel 2.3 Simbol-simbol *Blockchart*

No.	Simbol	Arti
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel, berkas atau cetakan.



2.		Multi dokumen.
3.		Proses secara manual.
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 terdefiniskan termasuk aktivitas fisik.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman 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>).
12.		Layar peraga (monitor).
13.		Pemasukan data secara manual.

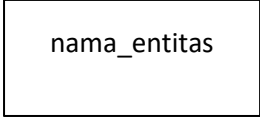
(Sumber : Uus Rusmawan (2019:75))

2.2.4 Entity Relationship Diagram (ERD)

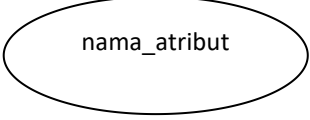
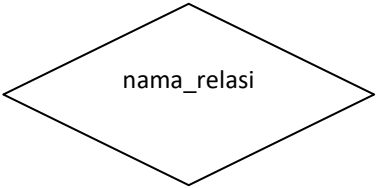

Rosa, Shalahudin (2018:50), ERD adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional.

Simbol-simbol yang digunakan dalam ERD, yaitu :

Tabel 2.4 Simbol-simbol *Entity Relationship Diagram (ERD)*

No.	Simbol	Deskripsi
1.	Entitas/ entity 	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.	<p>Atribut</p> 	Field atau kolom data yang butuh disimpan dalam suatu entitas.
3.	<p>Relasi</p> 	Relasi yang menghubungkan antar entitas ; biasanya diawali dengan kata kerja.
4.	<p>Asosiasi / association</p> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian.

(Sumber: Rosa, Shalahuddin (2018:50))

2.2.5 Kamus Data (*Data Dictionary*)

Rosa dan Shalahuddin (2018:73), 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).

Simbol-simbol yang digunakan dalam kamus data, yaitu:



Tabel 2.5 Simbol-simbol Kamus Data

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik...atau.....
{ ⁿ }	n kali diulang/ bernilai banyak
()	Data opsional
...	Batas komentar

(Sumber: Rosa dan Shalahuddin (2018:74))

2.3 Teori Judul

Pada Teori Judul akan dibahas sebagai berikut:

2.3.1 Pengertian Aplikasi

Menurut Kinaswara, dkk (2019:72), “Aplikasi adalah perangkat lunak yang diciptakan dengan berbagai komponen atribut yang sesuai dengan pengguna agar dapat membantu pengguna dalam mengolah setiap data agar menghasilkan input dan output”.

Menurut Huda dan Priyatna (2019:82), “Aplikasi adalah suatu program berbentuk perangkat lunak yang berjalan pada suatu sistem tertentu yang berguna untuk membantu berbagai kegiatan yang dilakukan oleh manusia”.

Jadi, dapat disimpulkan bahwa aplikasi adalah program komputer yang beroperasi pada sistem dan memiliki beberapa fitur tertentu yang dapat diakses oleh pengguna.

2.3.2 Pengertian Pendataan

Menurut Susena, dkk (2019:14), “Pendataan dapat diartikan sebagai proses pembuktian dari hasil penelitian dan lebih ditekankan dengan pengumpulan secara empiris”.

Menurut Ratnaningsih (dalam Julianti dkk, 2019:93), “Pendataan (pengolahan data) adalah suatu proses merubah *event* tanpa nilai apapun kemudian dilakukan



suatu kegiatan sehingga *event* tersebut dapat memberikan suatu keputusan yang berarti”.

Jadi, dapat disimpulkan bahwa pendataan merupakan suatu kegiatan memproses data dengan mengumpulkan data lalu mengolahnya sehingga menjadi suatu informasi.

2.3.3 Pengertian Inventaris

Menurut Huda dan Amalia (2020:14), “Inventaris adalah item atau material yang dipakai oleh suatu organisasi atau perusahaan untuk menjalankan bisnisnya”.

Menurut Suana dan Nilawati (dalam Ramita dkk, 2020:81), “Inventaris adalah daftar lengkap barang-barang yang ada di sebuah perusahaan atau instansi baik yang keluar maupun yang masuk, baik yang berwujud ataupun tidak”.

Jadi, dapat disimpulkan bahwa Inventarisasi merupakan kegiatan pencatatan atau penyusunan data barang yang dimiliki oleh suatu institusi. Inventarisasi yang baik akan mendukung kelancaran operasional atau kegiatan yang telah dirancang sebelumnya.

2.3.4 Barang Milik Negara (BMN)

Berdasarkan Peraturan Pemerintah Nomor 28 Tahun 2020 tentang pengelolaan barang milik negara / daerah, bahwa Barang Milik Negara (BMN) adalah semua barang yang dibeli atau diperoleh atas beban Anggaran Pendapatan dan Belanja Negara atau berasal dari perolehan lainnya yang sah. Disebutkan juga bahwa pengelolaan BMN adalah kegiatan yang dilakukan atas BMN mulai dari perencanaan sampai dengan penghapusan yang meliputi 10 (sepuluh) kegiatan sebagai berikut :

- a. Perencanaan Kebutuhan dan Penganggaran adalah kegiatan merumuskan rincian kebutuhan BMN untuk menghubungkan pengadaan barang yang telah lalu dengan yang sedang berjalan sebagai dasar dalam melakukan tindakan yang akan datang.
 - b. Pengadaan Perencanaan Anggaran yang mencerminkan kebutuhan riil BMN pada kementerian/ lembaga/ satuan kerja perangkat daerah selanjutnya
-



menentukan pencapaian tujuan pengadaan barang yang diperlukan dalam rangka penyelenggaraan tugas pokok, dan fungsi pemerintahan. Pengadaan BMN dilaksanakan berdasarkan prinsip-prinsip efisien, transparan dan terbuka, bersaing, adil/tidak diskriminatif dan akuntabel.

- c. Penggunaan adalah kegiatan yang dilakukan oleh pengguna barang dalam mengelola dan menatausahakan BMN yang sesuai dengan tugas pokok dan fungsi instansi yang bersangkutan.
- d. Pemanfaatan adalah pendayagunaan BMN yang tidak dipergunakan sesuai dengan tugas pokok dan fungsi kementerian / lembaga/satuan kerja perangkat daerah, dalam bentuk sewa, pinjam pakai, kerjasama pemanfaatan, dan bangun serah guna/bangun guna serah dengan tidak mengubah status kepemilikan.
- e. Pengamanan dan Pemeliharaan yang mencakup pengamanan administratif yang ditunjang oleh pengamanan fisik dan pengamanan hukum atas BMN merupakan bagian penting dari pengelolaan BMN. Kuasa pengguna barang, pengguna barang dan pengelola barang memiliki wewenang dan tanggung jawab dalam menjamin keamanan BMN yang berada dibawah penguasaannya dalam rangka menjamin pelaksanaan tugas pokok dan fungsi pemerintah. Sedangkan yang dimaksud dengan pemeliharaan adalah suatu rangkaian untuk menjaga kondisi dan memperbaiki semua BMN agar selalu dalam keadaan baik, dan siap untuk digunakan secara berdaya guna dan berhasil guna.
- f. Penilaian adalah suatu proses kegiatan penelitian yang selektif didasarkan pada data/fakta yang objektif dan relevan dengan menggunakan metode/ teknik tertentu untuk memperoleh nilai barang milik Negara.
- g. Penghapusan adalah tindakan menghapus BMN dari daftar barang dengan menerbitkan surat keputusan dari pejabat yang berwenang untuk membebaskan pengguna dan /kuasa pengguna barang dan / atau pengelola barang dari tanggung jawab administrasi dan fisik atas barang yang berada dalam penguasaannya.



- h. Pemindahtanganan adalah pengalihan kepemilikan barang milik Negara sebagai tindak lanjut dari penghapusan dengan cara dijual, dipertukarkan, dihibahkan atau disertakan sebagai modal pemerintah.
- i. Penatausahaan adalah rangkaian kegiatan yang meliputi pembukuan, inventarisasi, dan pelaporan BMN sesuai dengan ketentuan yang berlaku.
- j. Pembinaan, pengawasan dan pengendalian Menteri Keuangan menetapkan kebijakan umum dan kebijakan teknis pengelolaan BMN. Sedangkan Menteri Dalam Negeri menetapkan kebijakan teknis dan melakukan pembinaan pengelolaan BMN sesuai dengan kebijakan sebagaimana yang telah ditetapkan dalam kebijakan umum

2.3.5 Pengertian *Web*

Menurut Salamah, dkk (2020:3), “*Website* adalah halaman informasi yang disediakan jalur internet sehingga bisa diakses dimana saja, selama terkoneksi dengan jaringan”.

Menurut Rohi Abdullah (2018:1), “*Website* adalah kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang diseluruh dunia”.

Jadi, dapat disimpulkan bahwa *Website* adalah kumpulan halaman berisi tampilan berupa teks, gambar, animasi, audio, video, atau penggabungannya yang diakses oleh seluruh pengguna dengan menggunakan internet.

2.4 Teori Program

Pada Teori Program akan dibahas sebagai berikut:

2.4.1 Basis Data (*Database*)

Menurut Rozaq (2020:87), “*Database* adalah komponen kunci dari sebagian besar aplikasi perangkat lunak yang dirancang untuk menyimpan informasi tentang beberapa entitas bisnis kata nyata seperti *database* karyawan, *database* pelanggan, *database* siswa dan sebagainya”.



Menurut Rosa A. S dan M. Shalahuddin (2018:43), “Basis data (*database*) adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan”.

Jadi, dapat disimpulkan bahwa Basis Data adalah sebuah tempat penyimpanan informasi di dalam komputer yang sistematis sehingga dapat diakses dan dikelola dengan mudah untuk mendapatkan suatu informasi.

2.4.2 HTML (*HyperText Markup Language*)

Menurut Abdurahman, dkk (2019:43), “HTML merupakan bahasa standar yang digunakan dokumen yang ada dalam website. Bahasa pemrograman HTML menggunakan tag (akhiran) yang menandakan cara suatu keyword, kebanyakan browser mengenali akhiran HTML, biasanya tag berpasangan dan setiap tag ditandai dengan simbol $\langle \rangle$ ”.

Menurut Abdullah (2018:7), “HTML merupakan singkatan dari Hypertext Markup Language yaitu bahasa standar web yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari *website*. HTML berperan sebagai penyusun struktur halaman *website* yang menempatkan setiap elemen website sesuai layout yang diinginkan”.

Jadi, dapat disimpulkan bahwa *HTML* adalah suatu bahasa pemrograman yang digunakan untuk membuat halaman website dengan menggunakan tanda-tanda tertentu (tag).

2.4.3 PHP (*Hypertext Preprocessor*)

Menurut Abdurahman, dkk (2019:43), “*PHP* merupakan bahasa *server-side* yang menyatu dengan html, untuk membuat halaman *web* yang *dinamis*, salah satu fungsinya adalah untuk menerima dan mengolah dan menampilkan data ke sebuah situs, data yang diterima akan diolah di sebuah program databases *server*, untuk kemudian hasilnya ditampilkan kembali ke layar browser sebuah situs”.

Menurut Yudhanto dan Prasetyo (2018:9), “*PHP* atau *Hypertext Preprocessor* adalah bahasa pemrograman *script server side* yang sengaja



dirancang lebih cenderung untuk membuat dan mengembangkan *web*. Bahasa pemrograman ini memang dirancang untuk para pengembang *web* agar dapat menciptakan suatu halaman *web* yang bersifat dinamis”.

Jadi, dapat disimpulkan bahwa PHP adalah sebuah bahasa pemrograman server-side yang dipakai dalam pembuatan dan pengembangan suatu web dan biasanya digunakan bersama HTML.

2.4.4 *PhpMyAdmin*

Menurut Haqi (2019:9), “*PhpMyAdmin* adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menangani administrasi *MySQL* melalui jagat jembar (*World Wide Web*)”.

Menurut Yudanto dan Prasetyo (2018:15), “*PhpMyAdmin* adalah aplikasi *web* untuk mengelola database *MySQL* dan database *MariaDB* dengan mudah melalui antarmuka (*interface*) grafis”.

Jadi, dapat disimpulkan bahwa *PhpMyAdmin* adalah *software* yang ditulis dalam bahasa pemrograman PHP yang biasanya digunakan untuk mengelola database *MySQL*.

2.4.5 *MySQL*

Menurut Abdurahman, dkk (2019:44), “*MySQL* adalah salah satu jenis database yang banyak digunakan untuk membuat aplikasi berbasis web yang dinamis. *MySQL* termasuk jenis RDBMS (*Relational Database Management Sistem*). *MySQL* ini mendukung Bahasa pemrograman PHP. *MySQL* juga mempunyai query atau bahasa SQL (*Structured Query Language*) yang simple dan menggunakan escape character yang sama dengan PHP”.

Menurut Zahir (2019:4), “*MySQL* adalah sebuah software yang bersifat open source dan digunakan untuk manajemen basis data serta memiliki beberapa fitur pendukung dan ragam tipe data serta bersifat multiuser”.

Jadi, dapat disimpulkan bahwa *MySQL* adalah sebuah sistem manajemen database yang digunakan untuk mengelola basis data di dalam website yang multiuser.



2.4.6 CSS (*Cascading Style Sheets*)

Menurut Khozami (2020:55), “CSS merupakan singkatan dari *Cascading Style Sheets* yang memiliki sifat “*Style Sheets Language*” yang berarti bahasa pemrograman yang digunakan untuk mendesign halaman *website* dengan menggunakan penanda berupa *id* dan *class*”.

Menurut Lewenusa (2020:1), “CSS (*Cascading Style Sheet*) secara sederhana adalah sebuah metode yang digunakan untuk mempersingkat penulisan tag HTML, seperti *font*, *color*, *text*, dan tabel menjadi lebih ringkas sehingga tidak terjadi pengulangan penulisan”.

Jadi, dapat disimpulkan bahwa CSS adalah bahasa pemrograman yang digunakan untuk membangun sebuah *website* yang berfokus dalam mendesain tampilan halaman web.

2.4.7 *CodeIgniter*

Menurut Hustinawati, dkk (dalam Irawan dkk, 2020:159), “*CodeIgniter* adalah *framework* aplikasi web yang bersifat *open source* yang digunakan untuk membangun aplikasi PHP yang dinamis. Tujuan utamanya adalah untuk membantu pengembangan pengembang. *CodeIgniter* untuk mengerjakan aplikasi lebih cepat daripada menulis semua kode dari nol”.

Menurut Destiningrum dan Adrian (dalam Sulistiani dkk, 2018:90), “*CodeIgniter* merupakan sebuah *framework* yang akan dibangun berbasis web yang menggunakan konsep MVC (*Model, View, Controller*). *Framework* PHP ini dapat menjadi *tools* bagi seorang web *developer* untuk mengembangkan suatu situs dengan lebih mudah karena menyediakan *resource* yang lengkap”.

Jadi, dapat disimpulkan bahwa *CodeIgniter* merupakan *framework* PHP yang digunakan untuk membantu dalam pengembangan web dengan menggunakan konsep MVC (*Model, View, Kontrol*).



2.4.8 *Bootstrap*

Menurut Noviantoro, dkk (2022:91), “*Bootstrap* adalah platform CSS (*Cascading Style Sheet*) yang digunakan untuk merancang *website*. *Bootstrap* merupakan tool yang sangat baik untuk digunakan programmer saat membuat tampilan sebuah *website*. CSS, misalnya, dalam *bootstrap* menyediakan jenis, tombol, navigasi, dan komponen lainnya, serta JavaScript, yang membuat antarmuka perkembangan menjadi lebih mudah dan stabil”.

Menurut Nugroho dan Setyawati (dalam Satria dkk, 2022:115), “*Bootstrap* adalah *framework* CSS untuk membuat tampilan web yang menyediakan *class* dan komponen yang sudah siap dipakai dan digunakan”.

Jadi, dapat disimpulkan bahwa *Bootstrap* adalah *framework* CSS yang menyediakan berbagai komponen untuk merancang tampilan web.

2.4.9 *XAMPP*

Menurut Noviantoro, dkk (2022:90), “*XAMPP* merupakan perangkat lunak berbasis web server yang bersifat *open source* (bebas), serta mendukung diberbagai sistem operasi, baik Windows, Linux, atau Mac OS. *XAMPP* digunakan sebagai *standalone* server atau biasa disebut dengan localhost. Hal tersebut memudahkan dalam proses pengeditan, desain, dan pengembangan aplikasi”.

Menurut Mawaddah dan Fauzi (2018:2), “*XAMPP software web server apache* yang di dalamnya tertanam server *MySQL* yang didukung dengan bahasa pemrograman PHP untuk membuat *website* yang dinamis. *XAMPP* sendiri dapat berjalan pada berbagai macam *platform* seperti *Windows*, *Linux*, *Max OS X* dan *Solaris*”.

Jadi, dapat disimpulkan bahwa *XAMPP* adalah perangkat lunak web server yang digunakan untuk mengembangkan dan merancang situs *website* pada server lokal dan bersifat *open source*.



2.4.10 *Visual Studio Code*

Menurut Ramadhani dan Adriansyah (2022:36), “*Visual Studio Code (Vs Code)* adalah sebuah teks editor ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux*, *Mac*, dan *Windows*. Teks editor ini secara langsung mendukung Bahasa pemrograman *JavaScript*, *Typescript*, dan *Node js*, serta Bahasa pemrograman lainnya dengan bantuan plugin yang dapat dipasang via *marketplace Visual Studio Code* (seperti *C++*, *C#*, *Python*, *Java*, dst).”

Menurut Agustini dan Kurniawan (2019:155), “*Visual Studio Code* adalah kode editor sumber yang dikembangkan oleh *Micorsoft* untuk *Windows*, *Linux*, dan *macOS*”.

Jadi, dapat disimpulkan bahwa *Visual Studio Code* adalah sebuah software kode editor untuk membantu pengembangan sebuah aplikasi yang dapat dijalankan diberbagai *platform* sistem operasi.