



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

Teori Umum terdiri dari 3 (tiga) bagian yaitu Pengertian Komputer, Pengertian Perangkat Lunak, Pengertian Data. Berikut Penjelesannya.

2.1.1 Pengertian Komputer

Prawiro (2019:16) mengemukakan bahwa, “Komputer merupakan suatu perangkat elektronik yang dapat digunakan untuk mengolah data sesuai dengan prosedur yang telah dirumuskan sebelumnya sehingga menghasilkan informasi yang telah dirumuskan sebelumnya sehingga menghasilkan informasi yang bermanfaat bagi penggunanya”.

Selain pengertian di atas, adapun pengertian yang dikemukakan oleh Zainal Ansori dan Yulmaini (2019), “Komputer adalah sebuah mesin hitung elektronik yang secara cepat menerima informasi masukan digital dan mengolah informasi tersebut menurut seperangkat instruksi yang tersimpan dalam komputer tersebut dan menghasilkan keluaran informasi yang dihasilkan setelah diolah”.

Berdasarkan pendapat di atas, dapat disimpulkan bahwa definisi dari komputer adalah sebuah alat elektronik yang dapat memproses data digital dan informasi yang bermanfaat bagi *user* sehingga dapat menghasilkan keluaran informasi yang dihasilkan setelah diolah.

2.1.2 Pengertian Perangkat Lunak

Sukamto dan Shalahuddin (2018:2), “Perangkat Lunak (*Software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (*user manual*)”. Menurut Swara dan Febriadi (2018:8), “Perangkat lunak merupakan seluruh perintah yang digunakan untuk memproses informasi”.



Berdasarkan pendapat diatas, dapat disimpulkan bahwa perangkat lunak merupakan cara penggunaan yang ditujukan kepada komputer sesuai kebutuhan pemakai yang digunakan untuk memproses informasi.

2.1.3 Pengertian Data

Menurut Kristanto (2018:8) menyatakan bahwa, “Data merupakan bentuk yang belum dapat memberikan manfaat yang besar bagi penerimanya, sehingga perlu suatu model yang nantinya akan dikelompokkan dan diproses untuk menghasilkan informasi”. Selain itu menurut Nofriansyah dan Nur Cahyo (2019:1) mengatakan bahwa, “Data yaitu kumpulan fakta yang terekam atau sebuah entitas yang tidak memiliki arti dan selama ini terabaikan”.

Berdasarkan pendapat di atas, dapat disimpulkan bahwa definisi dari data adalah fakta yang belum dapat memberikan manfaat yang besar bagi penerimanya, sehingga perlu suatu model untuk menghasilkan informasi yang memiliki arti bagi suatu organisasi atau perusahaan.

2.2 Teori Khusus

Teori Khusus terdiri dari 3 (tiga) bagian yaitu Pengertian Unified Modeling Language (UML), Kategori Unified Modeling Language (UML), Jenis-Jenis Diagram (UML). Berikut ini Penjelasan.

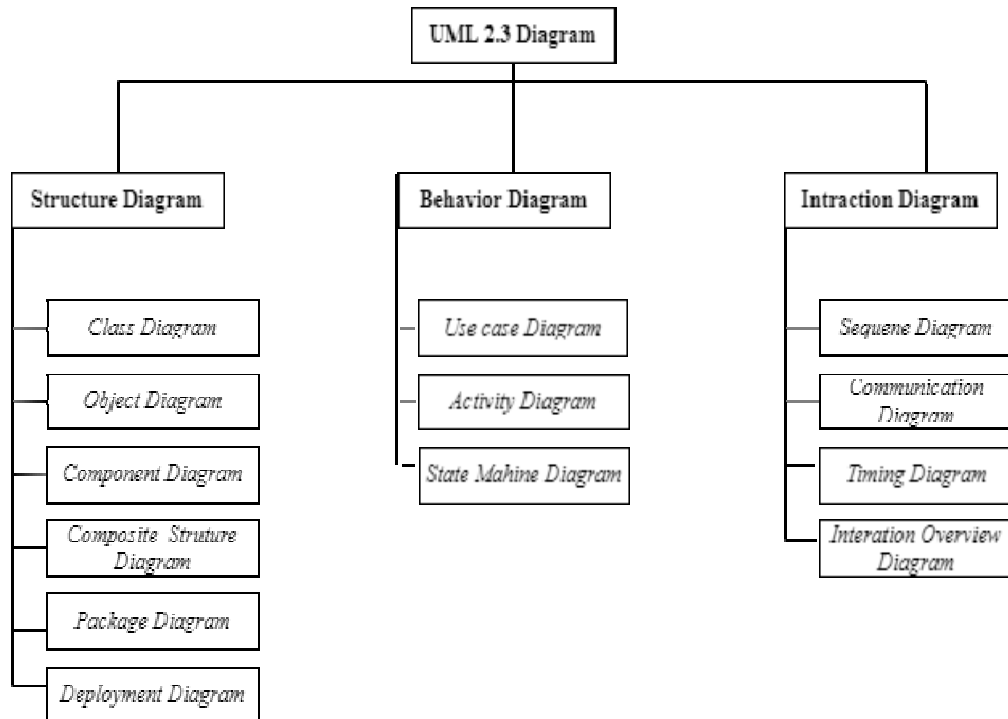
2.2.1 Pengertian *Unified Modeling Language* (UML)

Sukamto dan Shalahuddin (2018:133), menjelaskan tentang pengertian *Unified Modeling Language* sebagai berikut: “*Unified Modeling Language* (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.”

UML menyediakan serangkaian gambar dan diagram yang sangat baik. Beberapa diagram memfokuskan diri pada teori *object-oriented* dan sebagian lagi memfokuskan pada detail rancangan dan konstruksi. Semua dimaksudkan sebagai sarana komunikasi antar *team programmer* maupun pengguna.

2.2.2 Kategori *Unified Modeling Language* (UML)

Menurut Sukamto dan Shalahuddin (2018:140), “Pada UML 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 (tiga) kategori”. Pembagian kategori dan macam-macam diagram Menurut Sukamto dan Shalahuddin tersebut dapat dilihat pada gambar dibawah :



Gambar 2.1 Kategori dan Macam-Macam Diagram UML

Haviluddin dalam jurnal penelitiannya yang berjudul Memahami Penggunaan UML (*Unified Modelling Language*), halaman 3-5, menyebutkan bahwa pada UML versi 2 terdiri atas tiga kategori dan memiliki 13 jenis diagram, yaitu:

a. Struktur Diagram

1) *Class Diagram*

Class diagram menggambarkan struktur statis dari kelas dalam sistem anda dan menggambarkan atribut, operasi dan hubungan antara kelas.



2) *Object Diagram*

Object diagram menggambarkan kejelasan kelas dan warisan dan kadangkadang diambil ketika merencanakan kelas, atau untuk membantu pemangku kepentingan non-program yang mungkin menemukan diagram kelas terlalu abstrak.

3) *Component Diagram*

Component diagram menggambarkan struktur fisik dari kode, pemetaan pandangan logis dari kelas proyek untuk kode aktual di mana logika ini dilaksanakan.

4) *Deployment Diagram*

Deployment diagram memberikan gambaran dari arsitektur fisik perangkat lunak, perangkat keras, dan artefak dari sistem.

5) *Composite Structure Diagram*

Sebuah diagram struktur komposit mirip dengan diagram kelas, tetapi menggambarkan bagian individu, bukan seluruh kelas.

6) *Package Diagram*

Paket diagram biasanya digunakan untuk menggambarkan tingkat organisasi yang tinggi dari suatu proyek *software*.

b. Behavior Diagram

1) *Use Case Diagram*

Diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor.

2) *Activity Diagram*

Menggambarkan aktifitas-aktifitas, objek, state, transisi state dan event. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas.

3) *State Machine Diagram*

Menggambarkan *state*, transisi *state* dan *event*.



c. Interaction Diagram

1) Communication Diagram

Diagram yang bila dibandingkan dengan Sequence diagram, diagram komunikasi lebih terfokus pada menampilkan kolaborasi benda daripada urutan waktu.

2) Interaction Overview Diagram

Diagram yang berfokus pada gambaran aliran kendali interaksi dimana node adalah interaksi atau kejadian interaksi.

3) Sequence Diagram

Diagram yang menjelaskan interaksi objek yang disusun berdasarkan urutan waktu.

4) Timing Diagram

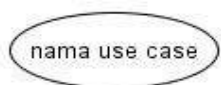
Timing diagram di UML didasarkan pada diagram waktu hardware awalnya dikembangkan oleh para insinyur listrik.

2.2.3 Jenis-Jenis Diagram Unified Modeling Language (UML)

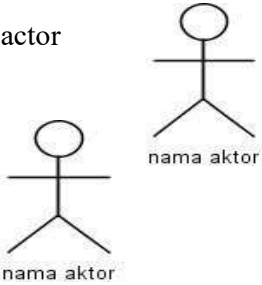

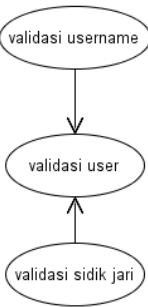
2.2.3.1 Pengertian Use Case Diagram

Menurut Sukanto dan Shalahuddin (2018:155), “*Use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem.” Adapun simbol-simbol yang digunakan dalam *use case* adalah sebagai berikut:

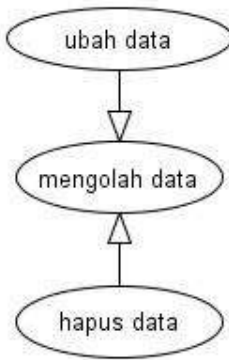
Tabel 2.1 Simbol-Simbol Use Case Diagram

No	Simbol	Deskripsi
1.	 <i>Use case</i>	fungsi-fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal-awal frase nama <i>use case</i> .

Lanjutan Tabel 2.1 Simbol-Simbol Use Case Diagram

2.	aktor / actor 	orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
3.	asosiasi / association	komunikasi antar aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan actor.
4.	ekstensi / extend  <<extend>>	relasi use case tambahan ke sebuah use case dimana use case yang di tambahkan dapat berdiri sendiri walau tanpa use case tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan, misalnya
	 <<extend>>	arah panah mengarah pada use case yang ditambahkan biasanya use case yang menjadi extend-nya merupakan jenis yang sama dengan use case yang menjadi induknya

Lanjutan Tabel 2.1 Simbol-Symbol Use Case Diagram

No	Simbol	Deskripsi
5.	Generalisasi <i>generalization</i>	<p>hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya,</p>  <pre>graph TD; A([hapus data]) --> B([mengolah data]); B --> C([ubah data]);</pre> <p>misalnya: arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum).</p>
6.	menggunakan / include /uses > <<include>>	<p>relasi tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini ada dua sudut pandang yang cukup besar mengenai include di <i>use case</i>:</p> <p><i>Include</i> berarti <i>use case</i> yang ditambahkan</p>

Sumber: Sukamto dan Shalahuddin (2017:156)



Ada dua hal utama pada *use case* yaitu:


1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit unit yang saling bertukar pesan antar unit atau aktor.

2.2.3.2 Pengertian Class Diagram

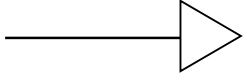
Menurut Sukamto dan Shalahuddin (2018:141), “*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.”

Adapun simbol-simbol yang digunakan dalam *class* diagram adalah sebagai berikut:

Tabel 2.2 Simbol-simbol *Class* Diagram

No	Simbol	Deskripsi
1.	Kelas nama_kelas +atribut +operasi()	Kelas pada struktur sistem
2	antarmuka / interface  nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
3.	asosiasi / association	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>

Lanjutan Tabel 2.2 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
4.	asosiasi berarah / <i>directed association</i>	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
5.	Generalisasi 	Relasi antarkelas dengan makna generalisasi – spesialisasi (umum khusus)
6.	kebergantungan / <i>dependency</i>	Relasi antarkelas dengan makna kebergantungan antar kelas
7.	agregasi / <i>aggregation</i>	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)


Sumber: Sukanto dan Shalahuddin

2.2.3.3 Pengertian Activity Diagram


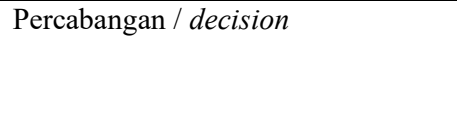
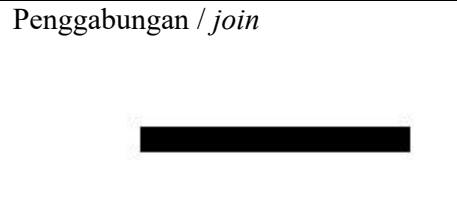
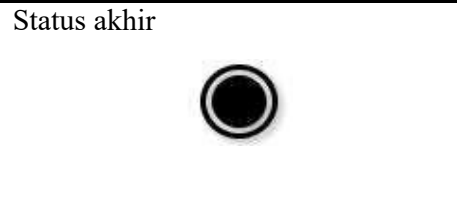
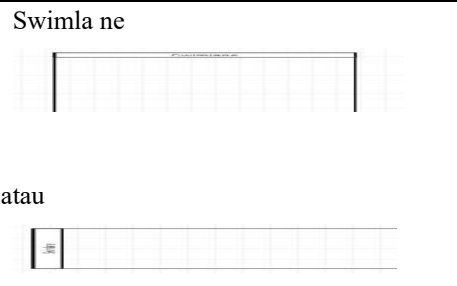
Menurut Sukanto dan Shalahuddin (2018:161), “*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.”

Adapun simbol-simbol yang digunakan dalam *activity diagram* adalah sebagai berikut:

Tabel 2.3 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal

Lanjutan Tabel 2.3 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3.	Percabangan / <i>decision</i> 	Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu
4.	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5.	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
6.	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber : Rosa dan Shalahuddin (2018:162)

2.2.3.4 Pengertian Sequence Diagram

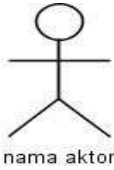
(Sukanto dan Shalahuddin, 2018:165), “Diagram sekuen menggambarkan kelakuan objek pada *Use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek -objek yang terlibat dalam sebuah beserta metode-metode yang dimiliki kelas yang diinstansikan

menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *Use case*.”

Banyaknya diagram sekuen yang harus digambarkan adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada diagram sekuen sehingga semakin banyak *use case* yang didefinisikan maka diagram sekuen yang harus dibuat juga semakin banyak.



Dapat penulis simpulkan bahwa *Sequence diagram* adalah penggambaran skenario dari sebuah objek yang ada pada *use case* yang meliputi rangkaian langkah- langkah aktivitas dari objek berdasarkan waktu hidup objek dan pesan-pesan yang diterima maupun yang dikirimkan objek kepada objek lainnya.

Tabel 2.4 Simbol-Simbol Pada Sequence Diagram

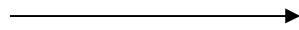
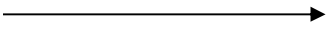
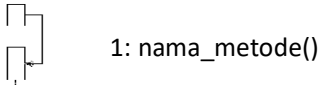
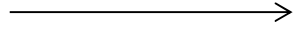
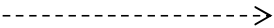
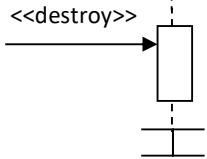
No	Simbol	Deskripsi
1.	Actor  atau Nama aktor tanpa waktu aktif	orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama actor
2.	Garis hidup / lifeline	menyatakan kehidupan suatu objek



Tabel 2.4 Simbol-Simbol Pada Sequence Diagram

No	Simbol	Deskripsi
3.	Objek  Nama objek : nama kelas	menyatakan objek yang berinteraksi pesan
4.	Waktu aktif 	menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya 2: cekStatusLogin() 1: login() 3: open() maka cek Status Login () dan open() dilakukan di dalam metode login() aktor tidak memiliki waktu aktif

Lanjutan Tabel 2.4 Simbol-Simbol Pada Sequence Diagram

No	Simbol	Deskripsi
5.	Pesan tipe create <<create>> 	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6.	Pesan tipe call 1: nama metode() 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,  arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas.
7.	Pesan tipe send 1: masukan 	Menyatakan suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
8.	Pesan tipe return 1: keluaran 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian
9.	Pesan tipe destroy <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i>



2.2.3.5 Metode Penelitian

Menurut (Sukamto & Shalahuddin, 2018:28) "Model SDLC air terjun (waterfall) sering juga disebut juga 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 dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (support). Berikut adalah gambar model air terjun :

1. Analisis Kebutuhan Perangkat Lunak Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasi kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user.
2. Desain
Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.
3. Pembuatan kode program
Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.
4. Pengujian
Pengujian fokus pada perangkat lunak secara dari segi logik 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.
5. Pendukung (*support*) atau pemeliharaan (*maintenance*)
Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan



lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.3 Teori Judul

Teori Judul terdiri dari 4 bagian yaitu Pengertian Aplikasi, Pengertian Penjualan, Pengertian Pembelian, dan Pengertian Pengelolaan, Pengertian Persediaan Berikut penjelasannya:

2.3.1 Pengertian Aplikasi

Menurut Indrajani (2018:3), “**Aplikasi** adalah program yang menentukan aktivitas pemrosesan informasi yang di butuhkan untuk penyelesaian tugas-tugas khusus dari pemakaian komputer”. (Syani & Werstantia, 2019:88) “**aplikasi** adalah sebuah perangkat sesuai dengan keinginan”.

Berdasarkan definisi diatas dapat disimpulkan bahwa pengertian aplikasi adalah suatu program yang dibuat untuk melakukan tugas-tugas tertentu.

2.3.2 Pengertian Penjualan

Menurut Karmila dan Rusda (2019:3), “Penjualan adalah suatu usaha yang terpadu untuk mengembangkan rencana – rencana strategis yang diarahkan pada usaha pemuasan kebutuhan dan keinginan pembeli guna mendapatkan penjualan yang menghasilkan laba. Srisadono (2018:172), “Penjualan adalah aktivitas usaha atau sebuah bisnis yang mana menyediakan dan menjual barang atau jasa”.

Dapat disimpulkan dari beberapa definisi diatas penjualan adalah aktivitas usaha dimana kebutuhan penjual dan pembeli terpenuhi melalui pertukaran barang ataupun jasa.



2.3.3 Pengertian Pembelian

Ilham et al., (2019:2), “Pembelian adalah usaha untuk memenuhi kebutuhan atas barang atau jasa yang diperlukan oleh perusahaan dan dapat diterima tepat waktunya dengan mutu yang sesuai serta harus yang menguntungkan.

Selain pengertian diatas adapun pengertian yang dikemukakan oleh Mulyadi (2016:244), “Pembelian adalah proses pengadaan barang yang meliputi permintaan pembelian, pemilihan pemasok, penempatan order pembelian, penerimaan barang, dan pencatatan transaksi pembelian

Dari pernyataan yang telah ditemukan oleh pengarang di atas maka penulis menarik kesimpulan bahwa, pembelian adalah suatu transaksi dimana perusahaan membutuhkan barang dan jasa pada suatu periode untuk dipakai maupun untuk persediaan yang akan dijual.

2.3.4 Pengertian Pengelolaan

Menurut Erwin (2018:11), “pengertian pengelolaan atau manajemen pada umumnya mengacu pada kegiatan-kegiatan yang meliputi perencanaan, pengorganisasian, pengarahan, pengoordinasian, pengawasan, dan penilaian. Hermanet et al., (2022:179), “Pengelolaan adalah proses penataan kegiatan yang akan dilaksanakan melalui fungsi-fungsi manajemen tentu gunanya sebagai tolak ukur untuk menentukan keberhasilan sebagai bentuk dari pencapaian tujuan bersama yang telah disepakati.

Dari beberapa pengertian diatas dapat disimpulkan bahwa pengelolaan adalah serangkaian kegiatan merencanakan, mengorganisasikan,, mengendalikan, dan mengembangkan segala upayadi dalam mengatur dan mendayagunakan sumber daya manusia, sarana dan prasarana untuk mencapai tujuan organisasi agar bisa tercapai hasil yang optimal.



2.3.5 Pengertian Persediaan

Resista et al., (2020:3), “Persediaan adalah suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha yang normal, atau persediaan barang-barang masih dalam pengerjaan/proses produksim, ataupun persediaan barang baku yang menunggu penggunaannya dalam suatu proses produksi. Menurut Anwar (2019:90). “Persediaan adalah barang yang disimpan dan disediakan oleh perusahaan untuk dapat digunakan dalam proses produksi, maupun untuk dijual ke konsumen.

Berdasarkan beerapa pengertian persediaan tersebut di atas, dapat diambil kesimpulan bahwa persediaan adalah bahan-bahan, bagian yang disediakan, dan bahan-bahan dalam proses yang terdapat dalam perusahaan untuk proses produksi, serta barang-barang jadi atau produk yang di sediakan untuk memenuhi permintaan pelanggan atau kosumen setiap waktu yang disimpan dan dirawat menurut aturan tertentu dalam tempat persediaan agar selalu dalam keadaan siap pakai.

2.3.6 Pengertian CV Daniel Putra

Perusahaan CV Daniel Putra merupakan perusahaan yang bergerak dibidang manufaktur yang aktifitas produksinya memproses barang mentah menjadi bahan jadi guna menyediakan produk yang dibutuhkan oleh pasar.

2.3.7 Pengertian Aplikasi Penjualan, Pembelian dan Pengelolaan Persediaan Kopi Pada CV Daniel Putra Berbasis Website

Aplikasi Penjualan,Pembelian dan Pengelolaan Persediaan Kopi Pada CV Daniel Putra Berbasis Website adalah sebuah Aplikasi yang dibuat untuk membantu dan mempermudah pihak CV Daniel Putra.



2.4 Teori Program

Teori Program terdiri dari 6 bagian yaitu Pengertian Basis Data, Pengertian MySQL, Pengertian Website, Pengertian XAMPP, Pengertian PHP, Pengertian HTML, Pengertian PHP MY ADMIN Berikut penjelasannya,

2.4.1 Pengertian Basis Data

Menurut Kristanto (2018:79) “Basis Data adalah kumpulan data, yang dapat digambarkan sebagai aktifitas dari satu atau lebih organisasi yang berelasi”. Menurut Aldo dkk (2020:104) "Basis data terdiri dari kata basis dan data. Basis dapat diartikan sebagai markas atau gudang. Sedangkan data adalah catatan atas kumpulan fakta dunia nyata yang mewakili objek seperti manusia, barang, hewan, konsep, peristiwa dan sebagainya yang diwujudkan dalam bentuk huruf, angka, simbol, gambar, teks, bunyi atau kombinasinya".

Menurut definisi diatas dapat disimpulkan pengertian basis data yaitu kumpulan data yang saling berhubungan yang disimpan dalam perangkat keras komputer dan akan diolah menggunakan perangkat lunak sehingga mempermudah aktivitas untuk memperoleh informasi

2.4.2 Pengertian MySQL

Menurut Rahayu (2020:14) “Database MYSQL merupakan database yang sangat kuat dan cukup stabil untuk digunakan sebagai media penyimpanan data sehingga database ini paling digemari di kalangan programmer web, dengan alasan bahwa bahwa program ini sebagai sebuah database server yang mampu untuk manajemen database dengan baik, MySQL terhitung merupakan data base yang paling digemari dan paling banyak digunakan dibanding database lainnya”.

Pengertian MySQL (My Structured Query Language) Menurut Sidik (2017:301) “MySQL merupakan software database yang termasuk paling populer dilingkungan Linux, kepopuleran ini karena ditunjang karenaperformasi query dari databasenya yang saat itu bisa dikatakan paling cepat dan jarang bermasalah. MySQL telah tersedia juga dilingkungan Windows”. Jadi, My SQL atau My



Structured Query Language salah satu database yang digunakan untuk menyimpan data.

2.4.3 Pengertian Website

Menurut Elgamar (2020:3) “Website merupakan sebuah media yang memiliki banyak halaman yang saling terhubung (*hyperlink*), dimana website memiliki fungsi dalam memberikan informasi berupa teks, gambar, video, suara dan animasi atau penggabungan darisemuanya.

Menurut Abdulloh (2018:1), *website* dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang di sediakan melalui jalur koneksi internet sehingga dapat di akses dan dilihat oleh semua orang di seluruh dunia. Halaman *website* dibuat menggunakan bahasa standar yaitu *HTML*. Skript *HTML* ini akan diterjemahkan oleh *web browser* sehingga dapat ditampilkan dalam bentuk informasi yang dapat di baca oleh semua orang.

Menurut pendapat diatas pengertian website adalah kumpulan *file* dalam bentuk digital baik berupa teks, gambar, video, audio, dan animasi lainnya yang terhubung ke internet.

2.4.4 Pengertian XAMPP

Menurut Enterprise (2018:3) XAMPP merupakan server yang paling banyak digunakan untuk keperluan belajar PHP secara mandiri, terutama bagi programmer pemula. Menurut Iqbal (2019:15), “Xampp merupakan sebuah software web server apache yang didalamnya sudah tersedia database server MySQL dan support PHP programming”. Dapat disimpulkan, XAMPP adalah software yang berisi web server local.

2.4.5 Pengertian PHP

Menurut Nurul Huda (2020:14) "PHP adalah sebuah bahasa pemrograman web berbasis server (server-side) yang mampu memarsing kode PHP dari kode web dengan ekstensi.php, sehingga menghasilkan tampilan website



yang dinamis di sisi client (browser). Menurut Supono & Putratama (2018:1) “**PHP** adalah suatu bahasa pemrograman yang digunakan untuk menterjemahkan basis kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat server- side yang ditambahkan ke HTML”.

PHP menurut Hidayatullah dan Kawistara **dalam** bukunya yang berjudul Pemrograman Web (2017:223) “**PHP** Hypertext Preprocessor adalah suatu bahasa scripting khususnya digunakan untuk web development”. **PHP** memiliki sifat server side scripting sehingga untuk menjalankan **PHP** harus menggunakan web server.

Jadi kesimpulannya, PHP atau Hypertext Preprocessor adalah bahasa script yang diproses di server yang kemudian hasilnya dikirimkan ke klien. Menurut Supono & Putratama (2018:1) “**PHP** adalah suatu bahasa pemrograman yang digunakan untuk menterjemahkan basis kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat server- side yang ditambahkan ke HTML”.

PHP menurut Hidayatullah dan Kawistara **dalam** bukunya yang berjudul Pemrograman Web (2017:223) “**PHP** Hypertext Preprocessor adalah suatu bahasa scripting khususnya digunakan untuk web development”. **PHP** memiliki sifat server side scripting sehingga untuk menjalankan **PHP** harus menggunakan web server. Jadi kesimpulannya, PHP atau Hypertext Preprocessor adalah bahasa script yang diproses di server yang kemudian hasilnya dikirimkan ke klien.

2.4.6 Pengertian HTML

Menurut Hidayatullah dan Kawistara (2017:15) HTML (Hypertext Markup Language) adalah “bahasa standard yang digunakan untuk menampilkan halaman web”. Menurut Adam (2019:2) “HTML atau *Hyper Text Markup Language* merupakan sebuah bahasa pemrograman terstruktur yang dikembangkan untuk membuat laman website yang dapat diakses atau ditampilkan menggunakan *web browser* (peramban web)”.



Jadi HTML atau Hypertext Markup Language adalah bahasa pemrograman yang digunakan untuk menampilkan halaman web pada *World Wide Web*.

2.4.7 Pengertian PHP My Admin

Menurut Nugroho (2019:8), *phpMyadmin* merupakan sebuah program bebas yang berbasis web yang dibuat menggunakan aplikasi *PHP*, tujuan dibuatnya program ini adalah untuk mengakses *database MySQL*, intinya adalah digunakan untuk menjadi administrator dari server *MySQL*. Dengan adanya program ini akan mempermudah dan mempersingkat kinerja kita, dengan kelebihan-kelebihan yang ada mengakibatkan para pengguna awam tidak harus mampu untuk mengetahui sintaks-sintaks *SQL* dalam pembuatan *database* dan *table*.

Menurut Menurut Yudhanto dan Prasetyo (2019:172), *phpMyadmin* merupakan sebuah program aplikasi yang dibuat menggunakan program *PHP* untuk mempermudah kita dalam mengakses *database MySQL*.

Dari pernyataan yang telah ditemukan oleh pengarang di atas maka penulis menarik kesimpulan bahwa *phpMyadmin* adalah administrator dari server *MySQL*, menggunakan program *PHP* dengan ini kita dipermudah untuk memanggil *database* yang telah kita buat.