



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Ansori dan Yulmaini (2019:56), “Komputer adalah alat elektronik yang dapat menerima input, mengolah input, memberikan informasi, menggunakan suatu program yang tersimpan di memori komputer, dapat menyimpan program dan hasil pengolahan, serta bekerja secara otomatis”.

Menurut Hanafri, dkk (2019:88), “Komputer adalah sebuah alat elektronik otomatis yang digunakan untuk mengolah data sesuai perintah secara cepat dan tepat, serta menerima dan menyimpan data berdasarkan perintah yang telah tersimpan di dalam memorinya”.

2.1.2 Pengertian Perangkat Lunak (*Software*)

Menurut Ali dalam Sudarso (2022:5), “Perangkat lunak aplikasi merupakan program yang ditujukan untuk menyelesaikan suatu permasalahan dalam aplikasi yang tertentu yang sudah dibuat oleh pabrik pembuat perangkat lunak aplikasi. Program aplikasi dibuat dengan menggunakan perangkat lunak bahasa (*language software*)”.

Menurut Indra dalam Sudarso (2022:5), “*Software* adalah sekumpulan data elektronik yang disimpan dan diatur oleh komputer. Data elektronik yang disimpan dapat berupa program atau instruksi yang akan menjalankan perintah. *Software* dapat diartikan juga sebagai segala jenis program yang digunakan untuk pengoperasian komputer dan peralatannya”.



2.1.3 Pengertian Data

Menurut Gunadi dan Widiyanto dalam Wahono dan Ali (2021:227), “Data adalah bahan mentah yang perlu dilakukan pengolahan sehingga menghasilkan informasi atau keterangan, baik kualitatif maupun kuantitatif yang menunjukkan fakta sehingga dapat memberi manfaat bagi peneliti atau memberi gambaran kepada peneliti tentang kondisi atau suatu keadaan. Sedangkan informasi adalah sekumpulan data yang sudah diolah sehingga menghasilkan suatu analisa untuk digunakan oleh pihak yang membutuhkan”.

Menurut Dewi (2020:226), “Data adalah deskripsi dari suatu benda, peristiwa, aktivitas dan transaksi yang direkam, dikelompokkan, dan disimpan dalam jumlah yang besar dan belum diolah menjadi sebuah informasi”.

2.1.4 Pengertian Internet

Menurut Ahdan dan Susanto dalam Priyanga dan Febriyo dalam (2022:4), “Internet adalah organisasi dari berbagai jaringan komputer yang menghubungkan semua komputer, mulai dari super komputer berukuran besar di berbagai badan pemerintahan, server perusahaan, hingga computer personal di rumah”.

Menurut Putri (2020:93), “Internet merupakan jaringan komputer yang terdiri dari berbagai perangkat komputer yang terhubung melalui protokol untuk memberikan informasi dan internet menyediakan sumber-sumber yang dapat diakses oleh manusia di seluruh dunia”.

2.1.5 Pengertian Perangkat Keras (*Hardware*)

Menurut Ansori (2019:56), “Perangkat keras komputer (*hardware*) adalah semua bagian fisik komputer, dan dibedakan dengan data yang berada di dalamnya atau yang beroperasi di dalamnya, seperti: *Motherboard, Power Supply, Processor, RAM, Hardisk, CD Drive, Battery CMOS, VGA Card, Sound Card, dll*”.



Menurut Rizky Dhanta dalam Putri dalam Jatmiko (2022:187), “Hardware adalah perangkat komputer yang terdiri atas susunan komponen-komponen elektronik berbentuk fisik (berupa benda)”.

2.1.6 Pengertian Basis Data (Database)

Menurut Sudarso (2022:3), “Basis data adalah tempat berkumpulnya data yang saling berhubungan dalam suatu wadah (perusahaan/organisasi) bertujuan agar dapat mempermudah dan mempercepat untuk pemanggilan atau pemanfaatan kembali data tersebut”.

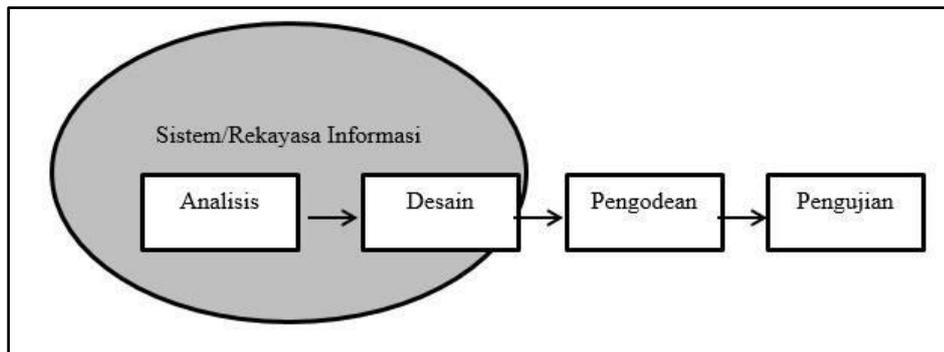
Menurut Hardiansyah dan Dewi (2020:223), “Basis data merupakan data yang dapat didesain dan berintegrasi sehingga dapat memenuhi kebutuhan user dalam perusahaan atau organisasi”.

Menurut Subandi et al (2018:3), “Basis Data merupakan data yang terintegrasi, yang diorganisasi untuk memenuhi kebutuhan para pemakai di dalam suatu organisasi”.

2.1.7 Metode Pengembangan Sistem *Waterfall*

Penelitian Laporan Akhir ini menggunakan metode pengembangan sistem *Waterfall*. Menurut Ridwan dan Fitri (2021:174), “Metodologi SDLC merupakan sebuah proses pembuatan dan perubahan pada system. Yang biasanya Sistem tersebut adalah Sistem Komputer atau Sistem Informasi. Sistem tersebut memiliki tahapan-tahapan yang terstruktur dari perencanaan, Analisa, Design, Implementasi, Testing & Maintenance”. 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:



Gambar 2.1 Ilustrasi Model *Waterfall*

a. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan proram 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. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. 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.



d. 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.

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

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. 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.

2.2 Teori Khusus

2.2.1 Kamus Data

Menurut Ma Chunga et al., (2022:23), “Kamus data adalah dokumen yang berisi gambaran terperinci tentang data yang digunakan dalam basis data. Menggambarkan karakteristik semua item dalam basis data, menjelaskan bagian dari metadata dan nilai dari seriap atribut, mendeskripsikan aliran data, mencakup informasi cara catatan diatur, dan diperlukan untuk membantu programmer dalam membuat kode untuk mengakses data”.

Menurut Puspita dan Anggadini (dikutip Prastianto dan Rostiani, 2020:26), “Kamus data (*Data Dictionary*) adalah suatu penjelasan tertulis mengenai data yang ada di dalam *database*. Kamus data pertama berbasis dokumen, kamus data itu tersimpan berbentuk *hard copy* dengan mencatat semua penjelasan data dalam bentuk tercetak”.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut:

**Tabel 2.1** Simbol-Simbol pada Kamus Data

No.	Simbol	Keterangan
1	=	disusun atau terdiri dari
2	+	dan
3	[]	baik ...atau...
4	{ }”	n kali diulang/ bernilai banyak
5	()	data opsional

(Sumber: Rosa dan Shalahuddin (2020:80))

Kamus data ini sangat membantu analis sistem dalam mendefinisikan data yang mengalir di dalam sistem, sehingga pendefinisian data itu dapat dilakukan dengan lengkap dan terstruktur.

2.2.2 Pengertian *Unified Modeling Language* (UML)

**Gambar 2.2** Logo UML

Sumber: Google

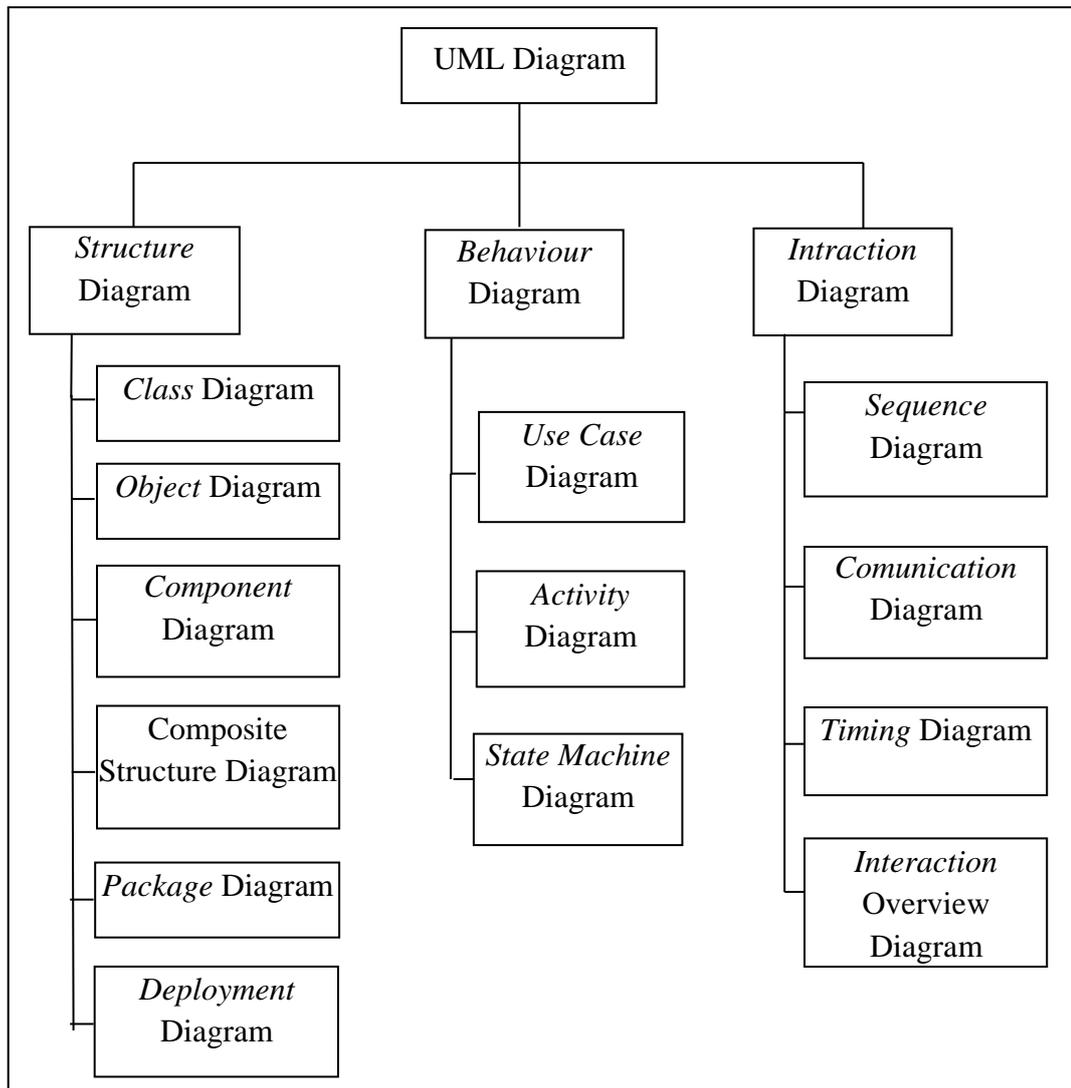
Menurut Rasid Ridho (2021:52),”Unified Modeling Language (UML) merupakan sebuah bahasa pemrograman visual standar untuk permodelan yang banyak dipakai didalam dunia kerja industri guna untuk mendefinisikan requirement, pembuat analisis design serta, menggambarkan arsitektur dalam pemrograman yang berorientasi pada objek”.



Menurut Feby, dkk (2022:15), “UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan untuk dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya”.

2.2.3 Klasifikasi Diagram UML (*Unified Modelling Language*)

Pada perkembangan teknik pemrograman berorientasi objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *Unified Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun dan dokumentasi dari system perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah system dengan menggunakan diagram dan teks-teks pendukung. Pembagian kategori dan macam-macam diagram tersebut dapat dilihat pada gambar dibawah ini:



(Sumber: Rosa dan Shalahuddin (2020:140))

Gambar 2.3 Klasifikasi Diagram UML (*Undefined Modelling Language*)

2.2.4 Jenis-Jenis Diagram *Unified Modeling Language* (UML)

2.2.4.1 Pengertian *Use Case Diagram*

Menurut Rasid Ridho (2021:51), “Use Case diagram adalah pemodelan terhadap kelakuan (behavior) pada sebuah sistem informasi yang akan dirancang. Secara umum use case diagram digunakan untuk memberikan gambaran secara detail akan fungsi dari setiap sistem dan juga untuk mengetahui hak akses dalam



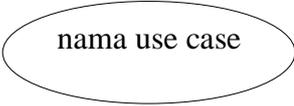
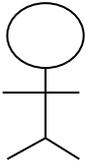
menggunakan sistem tersebut”. Menurut Tohari dalam Tabrani dan Aghniya (dikutip Apriliah., dkk, 2019:83), menyimpulkan bahwa “Use case adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor”.

Ada dua hal utama pada use case yaitu pendefinisian apa yang disebut aktor dan *use case*.

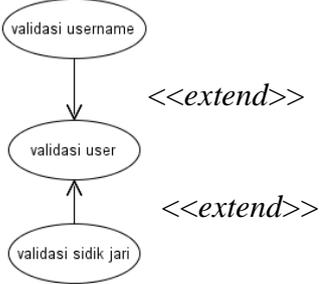
- a. **Aktor** merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
- b. **Use case** merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Simbol-simbol pada *Use Case Diagram* adalah sebagai berikut:

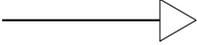
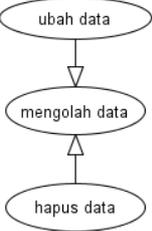
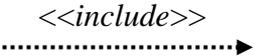
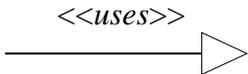
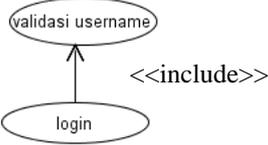
Tabel 2.2 Simbol-simbol *Use Case Diagram*

No	Simbol	Deskripsi
1.	<p><i>Use Case</i></p>  <p>nama use case</p>	<p>Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i>.</p>
2.	<p>Aktor / <i>actor</i></p>  <p>nama aktor nama_interface</p>	<p>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 -</p>

Lanjutan Tabel 2.2 Simbol-Simbol *Use Case* Diagram

No	Simbol	Deskripsi
		merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
3.	Asosiasi / <i>association</i> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> yang memiliki interaksi dengan actor.
4.	Ekstensi / <i>extend</i> 	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang di tambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misalnya</p>  <p>arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>

Lanjutan Tabel 2.2 Simbol-Simbol *Use Case* Diagram

No	Simbol	Deskripsi
5.	Generalisasi/ <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya:  arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum).
6.	Menggunakan <i>/include /uses</i>  	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> : a. <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu di panggil saat <i>use case</i> tambahan dijalankan, misalnya pada kasus berikut:  b. <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan -

Lanjutan Tabel 2.2 Simbol-Simbol *Use Case Diagram*

No	Simbol	Deskripsi
		<p>apakah <i>use case</i> yang di tambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misal pada kasus berikut:</p> <pre> graph BT A([ubah data]) -- "<<include>>" --> B([validasi user]) </pre> <p>kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>

(Sumber: Rosa dan Shalahuddin (2020:156-158))

2.2.4.2 Pengertian *Class Diagram*

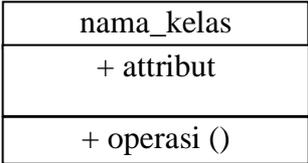
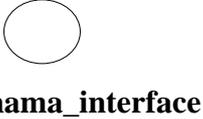
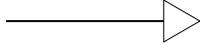
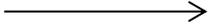
Menurut Rosa dan Shalahuddin (2020:141), “*Class Diagram* atau diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi”.

Menurut Tohari dalam Tabrani dan Aghniya (2019:43), mendefinisikan bahwa “Kelas (*class*) adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan perancangan berorientasi objek”.



Simbol-simbol pada *Class Diagram* adalah sebagai berikut:

Tabel 2.3 Simbol-Simbol *Class Diagram*

No	Simbol	Deskripsi
1.	<p>Kelas</p> 	Kelas pada struktur sistem.
2.	<p>Antarmuka / <i>interface</i></p> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	<p>Asosiasi / <i>association</i></p> 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.	<p>Asosiasi berarah / <i>directed association</i></p> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	<p>Generalisasi</p> 	Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).
6.	<p>Kebergantungan / <i>dependency</i></p> 	Relasi antarkelas dengan makna kebergantungan antarkelas.
7.	<p>Agregasi / <i>aggregation</i></p> 	Relasi antarkelas dengan makna semua- bagian (<i>whole-part</i>).

(Sumber: Rosa dan Shalahuddin (2020:146-147))



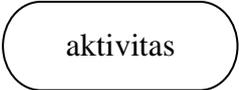
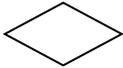
2.2.4.3 Pengertian Activity Diagram

Menurut Rasid Ridho (2021:53), “Activity Diagram merupakan sebuah gambaran aliran kerja atau sebuah aktivitas yang dilakukan pada sebuah sistem ataupun proses bisnis. Dan diagram aktivitas ini hanyalah menggambarkan kegiatan aktivitas sistem bukan apa yang dilakukan oleh actor.

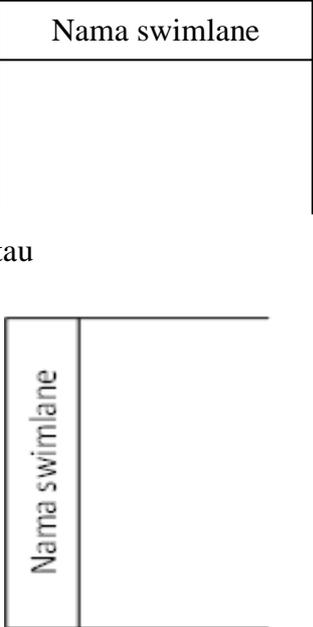
Menurut Tohari (dikutip Tabrani dan Aghniya, 2019:42), mendefinisikan bahwa “Activity diagram memodelkan *workflow* proses bisnis dan urutan aktifitas dalam sebuah proses. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari suatu aktifitas lainnya atau dari aktifitas ke status”.

Simbol-simbol pada Activity Diagram adalah sebagai berikut:

Tabel 2.4 Simbol-simbol Activity Diagram

No	Simbol	Deskripsi
1.	Status Awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.	Percabangan / <i>decision</i> 	Asosiasi percabangan dimana 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.

Lanjutan Tabel 2.4 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
6.	<p><i>Swimlane</i></p> 	<p><i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.</p>

(Sumber: Rosa dan Shalahuddin (2020:162-163))

2.2.4.4 Pengertian *Sequence Diagram*

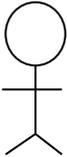
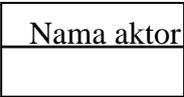
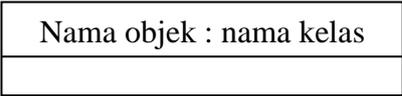
Menurut Rasid Ridho (2021:53), “*Sequence Diagram* adalah sebuah diagram yang menggambarkan kelakuan dari objek yang ada pada use case dengan cara mendeskripsikan waktu kejadian objek dan pesan yang akan dikirim dan diterima oleh antar objek”.

Menurut Tohari (dikutip Tabrani dan Aghniya, 2019:43), menyimpulkan bahwa “*Sequence diagram* menggambarkan interaksi antara sejumlah objek dalam urutan waktu”.

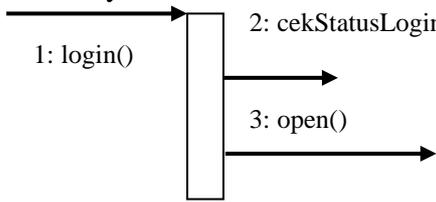
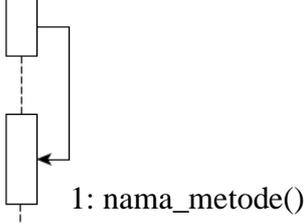


Simbol-simbol pada *Sequence Diagram* adalah sebagai berikut:

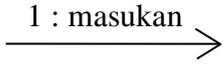
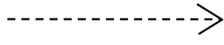
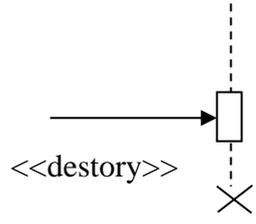
Tabel 2.5 Simbol-Simbol *Sequence Diagram*

No	Simbol	Deskripsi
1.	<p>Aktor</p>  <p>Atau</p>  <p>tanpa waktu aktif</p>	<p>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 <i>actor</i>.</p>
2.	<p>Garis hidup/ <i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek.</p>
3.	<p>Objek</p> 	<p>Menyatakan objek yang berinteraksi pesan.</p>

Lanjutan Tabel 2.5 Simbol-Simbol *Sequence Diagram*

No	Simbol	Deskripsi
4.	<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnva</p>  <p>maka cek Status Login () dan open() dilakukan di dalam metode login() Aktor tidak memiliki waktu aktif.</p>
5.	<p>pesan tipe <i>create</i></p> <p><code><<create>></code></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.</p>
6.	<p>Pesan tipe <i>call</i></p> <p><u>1 : nama_metode()</u></p> 	<p>Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,</p>  <p>arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus -</p>

Lanjutan Tabel 2.5 Simbol-Simbol *Sequence Diagram*

No	Simbol	Deskripsi
		ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi.
7.	Pesan tipe <i>send</i> 	Menyatakan bahwa suatu objek mengirimkan data/ masukan/ informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
8.	Pesan tipe <i>return</i> 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
9.	Pesan tipe <i>destrory</i> 	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> .

(Sumber: Rosa dan Shalahuddin (2020:165-167))



2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Menurut Harip Santoso (dikutip Syabania dan Rosmawarni, 2021:45), “Aplikasi adalah alat bantu untuk mempermudah dan mempercepat proses pekerjaan dan bukan merupakan beban bagi penggunanya”.

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

2.3.2 Pengertian E-commerce

Menurut Afrian (2018:158) “E-Commerce pada dasarnya merupakan suatu kontak transaksi perdagangan antara penjual dan pembeli dengan menggunakan media internet. Jadi proses pemesanan barang, pembayaran transaksi hingga pengiriman barang dikomunikasikan melalui internet”.

Menurut Alfiansyah et al (2023:3), “E-commerce adalah suatu kegiatan.membeli atau menjual secara elektronik yang dilakukan pada jaringan internet. E-commerce terlihat lebih nyata, dengan adanya kebutuhan penjualan dan pembeli untuk melakukan transaksi”.

Menurut Fauzi et al (2023:2), “E-commerce merupakan singkatan dari electronic commerce yang dapat diartikan bagaimana kita dapat melakukan proses komersil dengan memanfaatkan media eletkronik. E-commerce telah berkembang menjadi lebih modern dan canggih, di mana pelanggan bisa mengakses produk-produk yang dijual secara detail, melakukan pemesanan, memilih metode pengiriman yang diinginkan dan proses pembayaran dengan bermacam pilihan diantaranya seperti bank (mobile banking atau internet banking), kartu kredit ataupun melalui pihak ketiga lainnya seperti indomaret, alfamart dan sebagainya Dimana e-commerce dapat dibagi menjadi beberapa jenis berdasarkan karakteristiknya yaitu Business to Business (B2B), Business to Customer (B2C), Customer to Customer (C2C)”.



Menurut Melisa dan Anastasia Sitanggang (2022:21), “Electronic commerce (ecommerce) atau perdagangan lewat elektronik adalah perdagangan yang dilakukan dengan memanfaatkan jaringan telekomunikasi terutama internet. Internet memungkinkan orang atau organisasi yang berada pada jarak yang jauh dapat saling berkomunikasi dengan biaya yang murah. Hal ini kemudian dimanfaatkan untuk melakukan transaksi perdagangan”.

Menurut Musthofa & Adiguna (2022:199), “E-Commerce adalah salah satu implementasi dari bisnis online yang merupakan aktivitas dari penjualan dan pembelian produk serta jasa yang ditawarkan melalui jaringan. Adanya e-commerce yang tepat dan akurat dapat mengurangi terjadi kesalahan yang tidak diinginkan sehingga dapat meningkatkan penjualan yang efisien dalam penjualan suatu produk barang ataupun jasa”.

2.3.2 Pengertian Percetakan

Menurut Sihombing dan Siahaan (2019:107) “Percetakan adalah sebuah proses percetakan tulisan yang biasa di sebut dengan fotocopy atau percetakan gambar untuk pakaian yang biasa disebut dengan sablon dengan menggunakan sebuah mesin cetak yang terdiri dari bahan tinta dan minyak sehingga menghasilkan cetakan”.

Menurut Fauzi et.al (2018:5) “Percetakan menjadi salah satu penemuan yang penting dan cukup berpengaruh dalam sejarah kehidupan manusia. Percetakan merupakan sebuah komunikasi masa yang dapat digunakan sejak pertengahan tahun 1400-an hingga awal 1900-an. Percetakan dalam masa sekarang ini dikategorikan sebagai industri penting di berbagai negara, khususnya negara maju di dunia”.

Menurut Rizky Perdana & Zulkarnain (2022:162) “Percetakan merupakan suatu proses industri untuk membuat tulisan dan juga gambar dengan tinta diatas kertas dengan bantuan mesin cetak”.

Menurut Teisnajaya et al (2020:37), “Percetakan merupakan sebuah proses industri untuk memproduksi secara masal tulisan dan gambar baik terutama dengan



tinta dan kertas dengan menggunakan mesin cetak. Percetakan merupakan sebuah kegiatan penting di bidang penerbitan dan percetakan transaksi”.

2.3.2 Pengertian Digital Printing

Menurut Hikmawati (2021:459) “Digital printing adalah salah satu sub kategori dari printing komersial yang mempunyai keunggulan pada kecepatan pencetakan lembaran dokumen secara langsung melalui komputer tanpa melalui bantuan perantara seperti film atau pencetakan pelat seperti yang ada pada percetakan offset konvensional”.

Menurut Syamsudin dan Faisal (2019:273), “Digital printing adalah sebuah metode percetakan gambar berupa file, kemudian bisa langsung dicetak pada berbagai media secara instan dan cepat di era perkembangan teknologi informasi”.

2.3.3 Pengertian Website

Menurut Heryanto Hasan dan Muhammad (2020:47), “Pengertian web yaitu sebuah dokumen yang ditulis dalam *hype text markup language* (HTML) yang dapat diakses melalui *protocol hype text tranfer protocol* (HTTP) yang merupakan protokol untuk menyampaikan informasi dari sebuah pusat situs *web* untuk ditampilkan dihadapan pengguna program pembaca informasi yang ada pada situs *web*”.

Menurut Arafat et al (2022:7), “Website atau Web merupakan Sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk digital, baik berupa teks, gambar, video, audio dan animasi lainnya yang disediakan melalui jalur koneksi internet. Lebih jelasnya, website merupakan halaman–halaman yang berisi informasi yang ditampilkan oleh browser seperti Mozilla, Firefox, Google Chrome atau yang lainnya”.



2.3.4 Pengertian Aplikasi E-commerce Berbasis Website Menggunakan Framework Laravel Pada Jasa Percetakan Digital Printing Halma Grafika PT. Halma Delapan Indonesia Palembang

Aplikasi E-commerce Berbasis Website Menggunakan Framework Laravel Pada Jasa Percetakan Digital Printing Halma Grafika PT. Halma Delapan Indonesia Palembang adalah aplikasi berbasis website yang mempermudah pemesanan, penjualan, proses transaksi, dan rekapitulasi laporan keuangan pada percetakan.

2.4 Teori Program

2.4.1 Pengertian HTML



Gambar 2.4 Logo HTML

Sumber: Google

Menurut Yuda dan Faris (2023:43), “HTML (*Hypertext Markup Language*) yaitu sebuah intruksi pemrograman yang dituliskan dalam bentuk baris-baris kode, yang dikenal dengan istilah Tag. *Script* HTML dikenal hanya mampu menghasilkan sebuah *website* dengan *output* yang sederhana, baik dari segi tampilan maupun kapabilitas yang dimilikinya. Maintenance terhadap halaman web (seperti menambah dan meremajakan isi *website*), hanya dapat dilakukan dengan mengubah *script* HTML yang ada dalam program”.

Menurut Zainy, dkk (2022:336), “HTML adalah salah satu Bahasa pemrograman yang digunakan dalam pembuatan sebuah *website*”.



2.4.2 Pengertian JavaScript



Gambar 2.5 Logo JavaScript

Sumber: Google

Menurut Sahi (2020:122), “JavaScript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk *web*”.

Menurut Fandopa & Santoso (2022:5373), “JavaScript merupakan bahasa pemrograman web. Dimana sebagian besar situs website menggunakan javascript, dan semua browserweb modern di desktop, tablet, dan ponsel menyertakan bahasa javascript, menjadikan javascript yang merupakan s bahasa pemrograman yang paling banyak atau umum digunakan. Javascriptmerupakan bahasa pemrograman yangdinamis dan bahasa pemrograman dengan interpretasi yang sangat cocok untuk berorientasi objek dan programming fungsional

2.4.3 Pengertian CSS



Gambar 2.6 Logo CSS

Sumber: Google



Menurut Saputra dalam Handayani., dkk (2019:34), “Kepanjangan dari CSS adalah *Cascading Style Sheet* yang merupakan suatu bahasa pemrograman suatu bahasa pemrograman *web* yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam *web* sehingga tampilan *web* akan lebih rapi, terstruktur, dan seragam”.

Menurut Kurniawan dalam Arizha., dkk (2019:52), menjelaskan bahwa “CSS atau *Cascading Style Sheets* adalah sebuah dokumen yang berisi aturan yang digunakan untuk memisahkan isi dengan layout dalam halaman-halaman web yang dibuat”.

2.4.4 Pengertian PHP



Gambar 2.7 Logo PHP

Sumber: Google

Menurut Betha Sidik dalam Noviyanti., dkk (2021:70), “*PHP* merupakan secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen *HTML* secara *on the fly* yang dieksekusi di *server web*, dokumen *HTML* yang dihasilkan dari suatu aplikasi bukan dokumen *HTML* yang dibuat dengan menggunakan editor teks atau editor *HTML*”.

Menurut Simare dan Yana (2022:72), “*PHP* 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”.



2.4.5 Pengertian *Bootstrap*



Gambar 2.8 Logo *Bootstrap*

Sumber: Google

Menurut Mediana dan Nurhidayat (dikutip Purwanto dan Mubarok, 2020:154), “*Bootstrap* merupakan sebuah kerangka *framework* atau wadah untuk pengembangan aplikasi *web* yang lebih cepat dan mudah untuk suatu pengembangan sebuah sistem yang akan dibangun”.

Menurut Sopian, dkk (2020:50), “*Bootstrap* juga merupakan salah satu *framework* HTML, CSS dan *javascript* yang paling populer di kalangan *web* developer yang digunakan untuk mengembangkan sebuah *website* yang responsive. Sehingga halaman *website* nantinya dapat menyesuaikan sesuai dengan ukuran *monitor device* (desktop, tablet, ponsel) yang digunakan pengguna disaat mengakses *website-website* dari *browser*”.

2.4.6 Pengertian *Laravel*



Gambar 2.9 Logo *Laravel*

Sumber: Google

Menurut Ikhsan (2021:3), “*Laravel* merupakan *framework* aplikasi web yang ekspresif dan sintaks yang elegan. Kami membangun *framework* ini dengan menyenangkan dan dari pengalaman-pengalaman kreatif agar menjadi produk yang



memuaskan. Laravel berusaha mencoba menjadi framework yang mudah digunakan dengan mengurangi tugas-tugas umum yang sering digunakan dalam sebagian besar proyek-proyek web seperti otentikasi, routing, session, dan caching”.

Menurut Ngaibo dan Fikry (2021:9), “Laravel adalah sebuah framework PHP yang dirilis dibawah lisensi MIT, dibangun dengan konsep Model View Controller (MVC). Konsep MVC merupakan suatu konsep yang memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang menjadi pengontrol aplikasi”.

2.4.7 Pengertian XAMPP



Gambar 2.10 Logo XAMPP

Sumber: Google

Menurut Imam Mulhim (dalam Tumini dan Fitria (2021:14), “XAMPP adalah pake instalasi program yang terdiri atas program apache HTTP Server, MySQL, database dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan perl”.

Menurut Jubilee Enterprise dalam Tumini dan Fitria (2021:14), “XAMPP adalah server yang paling banyak digunakan untuk keperluan belajar PHP secara mandiri, terutama bagi programmer pemula”.



2.4.8 Pengertian MySQL



Gambar 2.11 Logo MySQL

Sumber: Google

Menurut Tebai dan Wijanarko (2022:119), “*MySql* merupakan pemrograman / sistem manajemen database (kumpulan data yang terstruktur) yang menggunakan basis Bahasa *SQL (Structured Query Language)*”.

Menurut Adi Nugroho (dikutip Renaldy dan Rustam, (2022:29), “MySQL (*My Structured Query Language*) adalah suatu sistem basis data relation atau *Relational Database Management System (RDBMS)* yang mampu bekerja secara cepat dan mudah digunakan oleh banyak user”.

2.4.9 Pengertian Visual Studio Code



Gambar 2.12 Logo Sublime Text

Sumber: Google

Menurut Nuraprian, dkk (2021:53), “*Visual Studio Code* adalah *editor source code* yang dikembangkan oleh Microsoft untuk Windows, Linux dan Mac OS. Ini termasuk dukungan untuk *debugging*, GIT Control yang disematkan, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, dan kode *refactoring*. Hal ini juga dapat diselesaikan, sehingga pengguna dapat mengubah tema *editor*,



shortcut keyboard, dan referensi. *Visual Studio Code* gratis dan *open-source*, meskipun unduhan resmi berbeda di bawah lisensi *proprietary*".

2.4.10 Pengertian Figma



Gambar 2.13 Logo Figma

Sumber: Google

Menurut Kurniawan dan Romzi (2022:3), "Figma adalah aplikasi desain UI dan UX berbasis *browser*, dengan desain yang sangat baik, *prototyping*, dan alat pembuatan kode. Saat ini (bisa dibilang) alat desain antarmuka terkemuka di industri, dengan fitur-fitur canggih yang mendukung tim yang bekerja pada setiap fase proses desain".

Menurut Febyla, dkk (2022:274), "Figma merupakan salah satu *design tool* berbasis *cloud* dan alat *prototyping* untuk produk digital yang biasanya digunakan untuk membuat *design* aplikasi *mobile*, *website*, *desktop* dan sebagainya. Figma dirancang untuk memungkinkan pengguna berkolaborasi dalam proyek dan bekerja sebagai tim di manapun secara sekaligus. Figma dapat digunakan pada sistem operasi *mac*, *linux* ataupun *windows* dengan menghubungkan perangkat ke internet".