

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Penelitian terdahulu**

Penelitian yang dilakukan oleh (Herlina, 2021) menggunakan metodologi *prototyping* yang meliputi pengumpulan data, analisis sistem, perancangan sistem, pengujian program, dan tahapan implementasi sistem untuk mengembangkan situs web penjualan. Penelitian ini menjelaskan tentang kebutuhan aplikasi untuk mengelola data supplier, produk, dan transaksi. Pada pengujian dengan metode *black box*, dibuatlah aplikasi penjualan perlengkapan kantor di toko dengan sistem pengeluaran uang tunai elektronik, dan sistem pengeluaran uang tunai elektronik digunakan untuk setiap barang yang dijual kepada pembeli. Setiap barang yang dijual akan discan *ID* barang sampai menampilkan harga barang. Aplikasi ini dikembangkan menggunakan aplikasi pemrograman *web* dengan menggunakan bahasa pemrograman *PHP, MySQL, HTML*.

Penelitian oleh (Suharya, 2021) membuat sebuah aplikasi penjualan tanaman. Mitra ini menggunakan transaksi hanya mengandalkan sebuah kalkulator tanpa dukungan komputerisasi. Hal tersebut kurang efisien dalam waktu, maka dari itu penelitian ini bertujuan untuk memudahkan pemilik toko dalam menjual tanaman. Penelitian ini menggunakan metode Waterfall serta menggunakan bahasa pemrograman *PHP, Database MySQL* dan *CMS wordpress*. Hasil yang di dapatkan dari penelitian ini adalah sebuah aplikasi *E-commerce* yang bisa diakses oleh 2 user yaitu admin dan pembeli dimana admin berperan sebagai pengelola aplikasi.

Penelitian oleh (Baihaqy, 2021) membuat Sistem Informasi Penjualan Berbasis Web. Metode pengumpulan data yang di gunakan penulis untuk memperoleh data yang dibutuhkan yaitu observasi dan literatur. Metode pengembangan Sistem yang digunakan pada penelitian ini menggunakan metode waterfall. Penelitian ini menghasilkan *website*. Dengan adanya sistem informasi penjualan, maka saat ini sistem pembayaran dan laporan penjualan menjadi lebih efisien , sehingga dapat melakukan pembayaran dengan menggunakan web yang sudah dibuat dan data otomatis langsung masuk ke dalam laporan. Sistem ini

diharapkan akan memberikan kemudahan bagi pemilik dalam melakukan pendataan dan bisa sebagai laporan bulanan yang rapi untuk instansi. Digunakannya sistem informasi penjualan, maka pegawai ataupun *admin* bisa memantau makanan atau minuman yang sudah terjual sama atau tidak dengan penghasilan yang di dapat.

Penelitian oleh (Priyanda, 2021) yaitu membuat Aplikasi Penjualan Dan Pemasaran Berbasis *Website*. Salah satu toko petshop untuk kebutuhan hewan peliharaan yang dimana sistem penjualan dan pemasaraanya secara konvensional. Pelanggan harus datang secara langsung ke lokasi, jauhnya jarak tempuh, waktu dan juga biaya yang dibutuhkan untuk datang secara langsung ke toko menyebabkan pelanggan tidak mau untuk berbelanja di petshop ini. Alasan inilah yang membuat penulis untuk membuat sebuah sistem pemasaran dan penjualan yang diperlukan oleh instansi yaitu Rancang Bangun Sistem Pemasaran dan Penjualan Berbasis *Website*. Dengan menggunakan framework CodeIgniter dan menggunakan metode waterfall dalam proses pembuatannya. Hasil yang didapat dari penelitian ini yaitu produk dari petshop dapat di pasarkan secara *online* melalui sistem. *Website* ini berhasil mengatasi permasalahan yang ada pada toko seperti memudahkan pembukuan data transaksi, memudahkan memberikan informasi kepada pelanggan secara *online* dan lengkap, memudahkan proses pembelian dengan jarak yang cukup jauh dan juga meningkatkan kepercayaan kepada pembeli terhadap kualitas toko. Oleh karena itu *website* ini sangat bermanfaat bagi toko.

Penelitian oleh (Izulhaq, 2023) yaitu membuat Aplikasi pemesanan produk menggunakan Metode Waterfall. Jurnal ini memuat aplikasi dengan kode program menggunakan *PHP* sebagai Bahasa pemrograman dengan *Database MySQL* dan framework CodeIgniter. Penelitian ini berhasil mengembangkan sebuah sistem yang memfasilitasi proses pemesanan produk cetak melalui situs web. Sistem ini dapat memberikan akses kemudahan untuk pelanggan dalam memesan jasa produk cetak tanpa perlu mengunjungi langsung lokasi. Untuk pengujian aplikasi menggunakan metode *blackbox testing*

Terdapat 5 jurnal penelitian terdahulu yang penulis jadikan referensi. Kelima jurnal tersebut membahas tentang pengembangan sistem informasi *website* dan *e-commerce* penjualan menggunakan teknologi yang sama dengan penelitian yang akan penulis lakukan, yaitu menggunakan bahasa pemrograman *PHP* dan *MySQL* sebagai *Database management system*, dan *framework Bootstrap* untuk tampilan. Penelitian ini bertujuan untuk mengembangkan serta memperbaiki sistem informasi *website* penjualan yang lebih efektif dan efisien. Metodologi penelitian yang akan digunakan adalah metode *waterfall*.

## **2.2 Landasan teori**

Landasan teori adalah seperangkat definisi, konsep, proposisi yang telah disusun rapi dan sistematis tentang variable-variabel dalam sebuah penelitian. Landasan teori merupakan bagian dari penelitian itu sendiri. Landasan teori merupakan bagian dari penelitian yang memuat teori-teori dan hasil-hasil penelitian yang berasal dari studi kepustakaan yang memiliki fungsi sebagai kerangka teori untuk menyelesaikan pekerjaan penelitian (Ramdhan, 2021).

### **2.2.1 Rancang Bangun**

Rancang Bangun adalah penggambaran, perencanaan, dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi. Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil analisa ke dalam bentuk paket perangkat lunak kemudian menciptakan sistem tersebut atau memperbaiki sistem yang sudah ada (Puimera, 2018).

### **2.2.2 E-commerce**

*E-commerce* adalah wadah bagaimana konsumen bisa membeli produk yang mereka inginkan secara *online*. Oleh karena itu, *E-commerce* adalah barang yang dijual langsung melalui internet kepada konsumen (bisnis-ke-konsumen) dan bisnis (bisnis-ke-bisnis). Menurut (Purnama, 2021) *E-commerce* adalah seperangkat media yang digunakan untuk membeli dan menjual jasa atau produk antara dua pihak melalui internet. Ibaratnya sebuah mekanisme bisnis yang fokus pada transaksi antar individu di internet sebagai media untuk melakukan penjualan.

### 2.2.3 Website

(Abdulloh, 2018) Website adalah sekumpulan halaman yang memuat informasi digital dalam bentuk teks, gambar, animasi, suara, dan video, atau kombinasi dari semuanya. Informasi ini disediakan melalui koneksi internet sehingga dapat diakses dan dilihat oleh siapa saja di seluruh dunia. Halaman website umumnya merupakan dokumen yang ditulis dalam format HyperText Markup Language (HTML), yang dapat diakses melalui HTTP. HTTP adalah protokol yang mengirimkan berbagai informasi dari server website untuk ditampilkan kepada pengguna melalui web browser (Nofyat, 2018).

Jenis *website* secara umum dibagi 3 yaitu jenis *website* statis, dinamis dan interaktif. *website* statis adalah jenis *website* yang didalamnya tidak diperbarui secara terus menerus sehingga dari waktu ke waktu isinya akan selalu tetap sama, biasanya *website* seperti ini profil perusahaan. *Website* dinamis yaitu isinya yang diperbarui secara berkala, jenis *website* ini biasanya diminati oleh perusahaan yang bisnisnya berhubungan dengan internet, misalnya *website* berita. Sedangkan *website* interaktif sama seperti *website* dinamis, tapi informasinya tidak hanya diubah oleh pengguna melainkan banyak dilakukan oleh pengunjung *website* atau pengguna, seperti sosial media dan *marketplace* (Abdulloh, 2018).

*Website* biasanya terdiri dari 3 komponen utama, yaitu *HTML*, *CSS* dan *JavaScript*. *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. Untuk mengetikkan *script HTML* dapat menggunakan *text editor* seperti *Notepad* sebagai paling bentuk sederhana atau *text editor* khusus yang dapat 22 mengenali setiap unsur *script HTML* dan menampilkannya dengan warna yang berbeda sehingga mudah dibaca, seperti *Notepad++*, *Sublime Text*, dan masih banyak lagi aplikasi lain yang sejenis (Abdulloh, 2018).

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Contoh Aplikasi Web</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <h1>Selamat Datang di Aplikasi Web Saya</h1>
  <div id="content">
    <p id="demo">Ini adalah contoh teks.</p>
    <button onclick="changeText()">Ubah Teks</button>
  </div>
  <?php include 'script.php'; ?>
  <script src="script.js"></script>
</body>
</html>

```

**Gambar 2. 1** *Script HTML*

CSS merupakan singkatan dari *Cascading Style Sheet* yaitu dokumen *web* yang berfungsi mengatur elemen *HTML* dengan berbagai properti yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan. Banyak orang menganggap CSS bukan salah satu bahasa pemrograman karena strukturnya sederhana, hanya berupa kumpulan-kumpulan aturan yang mengatur *style* elemen *HTML*. Cara kerja CSS dalam memodifikasi *HTML* dengan memilih elemen *HTML* yang akan diatur kemudian memberikan properti yang sesuai dengan tampilan yang diinginkan. Dalam memberikan aturan pada elemen *HTML*, *script* CSS terdiri dari atas 3 bagian yaitu *selector* untuk memilih elemen yang akan diberi aturan, *property* yang merupakan aturan yang diberikan, dan *value* sebagai nilai dari aturan yang diberikan. Cara penulisan CSS dibedakan menjadi tiga macam yaitu *inline*, *internal*, dan *external* (Abdulloh, 2018).

```

body {
  font-family: Arial, sans-serif;
  background-color: #f4f4f4;
  color: #333;
}

h1 {
  text-align: center;
}

#content {
  text-align: center;
  margin-top: 20px;
}

```

**Gambar 2. 2** *Script CSS*

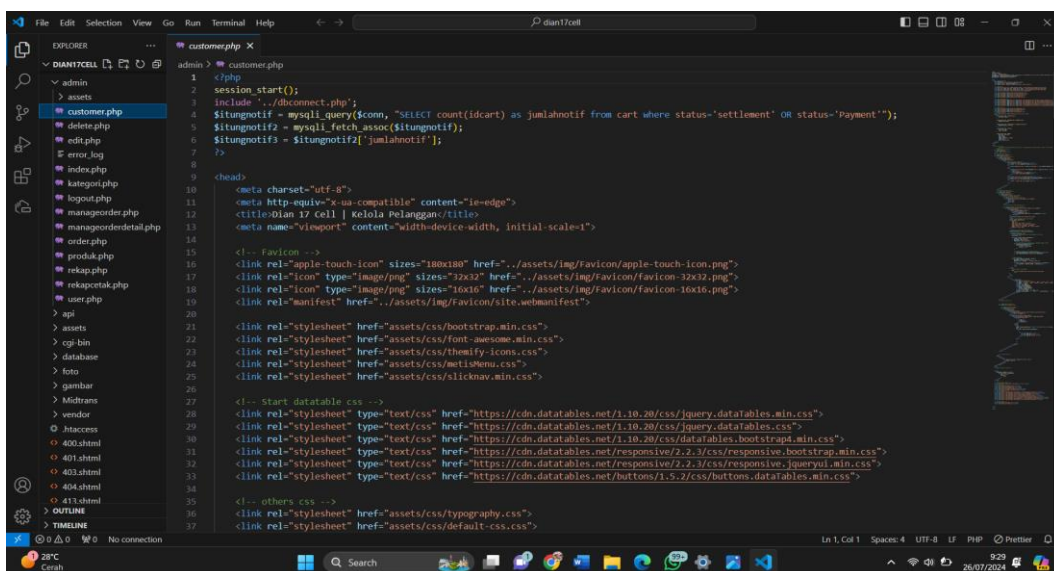
*JavaScript* adalah bahasa pemrograman *website* yang prosesnya dilakukan di sisi *client*. Karena berjalan di sisi *client*, *JavaScript* dapat dijalankan hanya dengan menggunakan *browser*. Berbeda dengan *PHP* yang bekerja di sisi *server*, untuk menjalankan *script JavaScript* tidak memerlukan *refresh* pada *browser*. *JavaScript* biasanya dijalankan ketika ada *event* tertentu yang terjadi pada *halam web*. Baik *event* yang dilakukan oleh *user*, maupun *event* yang terjadi karena adanya perubahan pada *halaman Website* (Abdulloh, 2018).

```
function changeText() {
    document.getElementById("demo").innerHTML = "Teks telah diubah!";
}
```

Gambar 2. 3 *JavaScript*

#### 2.2.4 *Visual Studio Code*

*Visual Studio Code* adalah aplikasi *editor code open source* yang dibesarkan oleh *Microsoft* untuk sistem operasi *Windows*, *Linux*, dan *MacOS*. *Visual Code* memudahkan dalam penulisan *code* yang mendukung beberapa jenis pemrograman, seperti *C++*, *C#*, *Java*, *Python*, *PHP*, *GO*. *Visual Code* memiliki kemampuan untuk mengidentifikasi jenis bahasa pemrograman yang digunakan dan memberi variasi warna sesuai dengan fungsi dalam rangkaian *code* tersebut. *Visual Studio Code* juga telah terintegrasi ke *Github* (Nufriana, 2019).



Gambar 2. 4 Tampilan *Visual Studio Code*

### 2.2.5 Bootstrap

Bootstrap menyediakan berbagai kelas dan komponen yang sudah siap digunakan. Framework ini juga menyediakan template desain berbasis *HTML* dan *CSS* yang digunakan untuk memperindah tampilan tipografi, bentuk, tombol, navigasi, dan berbagai komponen lainnya. Selain itu, terdapat pula ekstensi Javascript opsional yang dapat digunakan. Menurut (Prayana, 2019) Bootstrap merupakan sebuah framework yang memudahkan pengembang dalam mengembangkan situs web dengan cepat. Framework ini terdiri dari beberapa file yang berisi kumpulan kode *CSS* dan *JavaScript* dalam bentuk *class*.

### 2.2.6 Hypertext Preprocessor (PHP)

Menurut (Enterprise, 2018) *PHP* merupakan pemrograman yang digunakan untuk membuat *Website* dinamis dan interaktif, dinamis artinya *website* tersebut bisa berubah – ubah tampilan dan konten sesuai kondisi tertentu. Menurut (Abdulloh, 2018) “*PHP* merupakan kependekan dari *PHP Hypertext Preprocessor* yaitu bahasa pemrograman *web* yang dapat disisipkan dalam skrip *HTML* dan bekerja di sisi *server*”. *PhpMyAdmin* adalah aplikasi berbasis *web* yang dikembangkan menggunakan bahasa pemrograman *PHP*. Melalui *PhpMyAdmin*, pengunjung dapat melakukan perintah *Query* tanpa harus mengetikkan seperti pada *MS DOS*. Perintah tersebut misalnya administrasi *User* dan *privileges*, *export* dan *import Database*, manajemen *Database*, manajemen label dan struktur *table*, dan sebagainya. *PHPMYAdmin* sangat *user friendly*, sangat mudah untuk digunakan walau pengguna baru (*newbie*) (Kurniawan, 2018).

```
<?php
// Menampilkan pesan selamat datang
echo "Selamat datang di aplikasi PHP!";

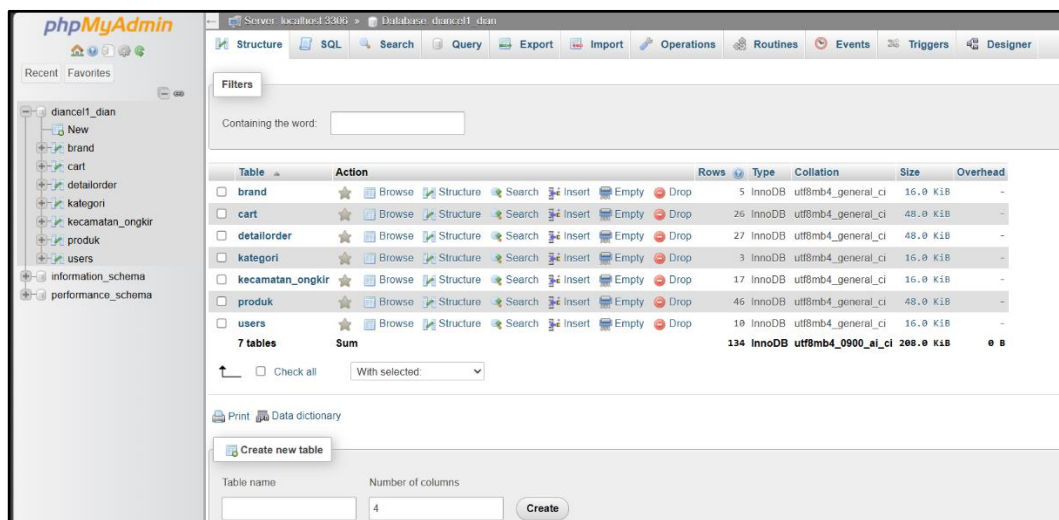
// Menghitung luas persegi panjang
$panjang = 10; // panjang dalam satuan
$lebar = 5;    // lebar dalam satuan
$luas = $panjang * $lebar;

// Menampilkan hasil perhitungan
echo "<br>Luas persegi panjang adalah: " . $luas . " satuan persegi";
?>
```

Gambar 2. 5 Script Php

### 2.2.7 Database

(Abdulloh, 2018) berpendapat bahwa *Database* atau Basis data adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi. Menurut (Kristanto, 2018) Basis data adalah kumpulan data, yang dapat digambarkan sebagai aktivitas dari satu atau lebih organisasi yang berelasi.



**Gambar 2. 6** Tampilan *Database*

Sebuah basis data memiliki jenjang sebagai berikut:

1. **Karakter:** Unit data terkecil yang dapat berupa angka, huruf, atau karakter khusus yang membentuk item atau field.
2. **Field:** Representasi dari suatu atribut dalam record atau tupel yang menunjukkan sebuah item data.
3. **Record:** Kumpulan field yang membentuk unit data individu atau rekaman, menggambarkan suatu entitas data spesifik.
4. **File:** Kumpulan record yang serupa, membentuk sebuah kesatuan data yang mewakili jenis data tertentu.
5. **Database:** Kumpulan file atau tabel yang membentuk sebuah basis data, menyimpan data secara terstruktur dan terorganisir.



Menurut (Enterprise, 2018) *MySQL* adalah sebuah server yang digunakan untuk mengelola basis data. Untuk membuat dan mengolah basis data, kita menggunakan bahasa pemrograman khusus yang disebut *SQL (Structured Query Language)*. Basis data diperlukan ketika ingin mengumpulkan data dari pengguna melalui formulir HTML, kemudian mengolahnya dengan PHP, dan menyimpannya ke dalam basis data *MySQL*. *MySQL* merupakan pengembangan dari proyek *UNIREG* yang awalnya dikembangkan oleh *Michael Monty Widenius* dan *TcX*, sebuah perusahaan perangkat lunak asal Swedia .

Menurut (Ramadhan, 2020) *MySQL* adalah DBMS yang open source dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi *MySQL* adalah *Database* server yang gratis dengan lisensi GNU General Public License (GPL) sehingga dapat Anda pakai untuk keperluan pribadi atau komersial tanpa harus membayar lisensi yang ada. Seperti yang sudah disebutkan sebelumnya, *MySQL* masuk ke dalam jenis RDBMS (*Relational Database Management System*). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada *MySQL*. Contohnya di dalam *MySQL* sebuah *Database* terdapat satu atau beberapa tabel. *MySQL* merupakan *database engine* atau server *Database* yang mendukung bahasa *Database SQL* sebagai bahasa interaktif dalam mengelola data.

### **2.2.8 Xampp**

Menurut (Enterprise, 2018) XAMPP adalah aplikasi web server *Apache* yang menyediakan integrasi dengan *MySQL* dan *PHPMyAdmin*. Nama XAMPP berasal dari singkatan X (kemampuan untuk diinstal di berbagai sistem operasi), *Apache Server*, *MySQL*, dan *PHP*. Aplikasi ini dapat diinstal pada sistem operasi *Windows*, *Linux*, *MacOS*, dan *Solaris*.

XAMPP adalah perangkat lunak *open source* yang mendukung berbagai sistem operasi, yang terdiri dari berbagai program. Fungsinya adalah sebagai server lokal (*localhost*) yang mandiri, yang mencakup *Apache HTTP Server*, basis data *MySQL*, serta interpreter untuk bahasa pemrograman PHP dan Perl (Rachmatsyah, 2021).

### 2.2.9 Apache

Apache adalah aplikasi *web* server dengan penerimaan terbanyak di dunia. Apache menjadi salah satu penggerak utama supaya *Website* dapat terhubung dengan pengunjung (*user*). Tetapi, apache memiliki beberapa kekurangan salah satunya apache menjadi lamban pada saat permintaan yang masuk berjumlah besar. Salah satu cara untuk melakukan peningkatan performa *web* server Apache adalah dengan *reverse proxy caching* (Luthfi, 2018).

Apache adalah nama web server yang digemari karena kompak, modular, mengikuti standar protokol HTTP, memiliki jumlah pengguna jauh lebih besar dibandingkan pesaingnya. Apache merupakan web server paling populer dan banyak digunakan di dunia. Keunggulan Apache sangat fleksibilitas dan kemampuannya dalam menangani berbagai jenis konten web, seperti halaman statis, dinamis, dan skrip (Ridho, 2023).

### 2.2.10 Hosting

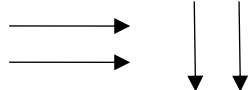
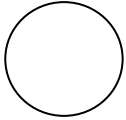
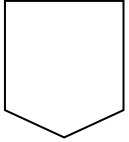
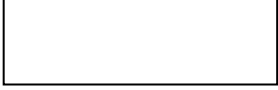
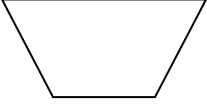
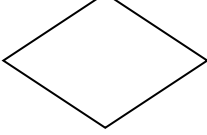
Menurut (Kurniansyah, 2020) Hosting adalah layanan internet yang menyediakan server atau sumber daya komputer untuk menyimpan dan mengelola halaman website sehingga dapat diakses secara online oleh orang lain. Hosting juga mencakup penyediaan sumber daya seperti *HTTP*, *FTP*, *email*, dan *DNS* untuk memungkinkan individu atau organisasi menempatkan informasi mereka di internet.


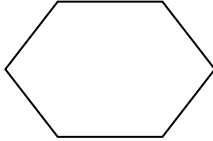

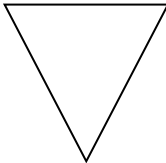
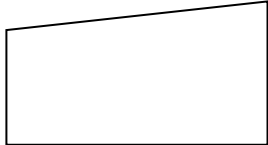

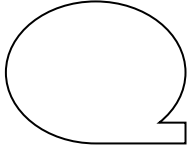
### 2.2.11 Flowchart

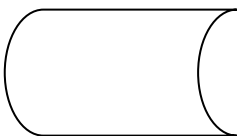



*Flowchart* adalah cara penulisan algoritma dengan menggunakan notasi grafis. *Flowchart* merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses beserta pernyataannya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan antara proses digambarkan dengan garis penghubung. Dengan menggunakan *flowchart* akan memudahkan untuk melakukan pengecekan bagian-bagian yang terlupakan 4 dalam analisis masalah.

Flowchart juga didefinisikan sebagai representasi grafis dari langkah-langkah program dan urutan proses. Flowchart membantu analisis dan pengembangan memecah masalah menjadi segmen yang lebih kecil dan menganalisis opsi tindakan lainnya. Flowchart biasanya mempermudah penyelesaian suatu masalah, terutama yang memerlukan penyelidikan dan evaluasi lebih lanjut (Zalukhu, 2023). Tabel 2.1 adalah simbol-simbol *flowchart* yang biasa digunakan (standar keluaran *ANSI* dan *ISO*) beserta penjelasannya :

**Table 2. 1** Simbol-Simbol Flowchart

NO	SIMBOL	KETERANGAN
1		Simbol arus/ <i>flow</i> , Untuk menghubungkan antara simbol yang satu dengan simbol yang lain atau menyatakan jalannya arus dalam suatu proses.
2		Simbol <i>connector</i> , Simbol suatu keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang sama.
3		Simbol <i>offline connector</i> , Simbol untuk keluaran atau masukan prosedur atau proses dalam lembar atau halaman yang berbeda.
4		Simbol <i>process</i> Untuk menunjukkan pengolahan yang dilakukan oleh komputer atau PC.
5		Simbol <i>manual</i> , Berfungsi untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer atau PC
6		Simbol <i>decision</i> , Menunjukkan suatu perbandingan yang harus dibuat bila hasilnya “ya”, maka alir data menunjukkan ke suatu

		tempat, bila “tidak” maka akan menuju ke tempat lain.
7		Simbol <i>terminal</i> , Simbol ini digunakan untuk menunjukkan awal kegiatan (start) atau akhir dari suatu kegiatan (stop)
8		Simbol <i>predefined process</i> , Berfungsi untuk mempersiapkan penyimpanan yang sedang/ akan digunakan sebagai tempat pengolahan didalam storage.
9		Simbol <i>keying operation</i> , berfungsi untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai <i>keyboard</i> .
10		Simbol <i>offline-storage</i> , berfungsi untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
11		Simbol <i>manual input</i> , berfungsi untuk memasukkan data secara manual dengan menggunakan <i>online keyboard</i> .
12		Simbol <i>input/output</i> , Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
13		Simbol <i>magnetic tape</i> , berfungsi untuk menyatakan <i>input</i> berasal dari pita magnetis atau <i>output</i> disimpan ke pita magnetis.

14		Simbol <i>disk storage</i> , berfungsi untuk menyatakan <i>input</i> berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
15		Simbol <i>document</i> , berfungsi untuk mencetak keluaran dalam bentuk dokumen (melalui <i>printer</i> ).
16		Simbol <i>punched card</i> , berfungsi untuk menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu.
17		Simbol predefined Untuk pelaksanaan suatu bagian (subprogram) / prosedur

### 2.2.12 Use Case Diagram


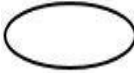


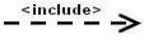
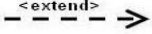
Use case merupakan deskripsi tekstual berupa skenario aplikasi untuk menggambarkan interaksi antara aktor dan sistem. Use case kemudian direpresentasikan secara visual dalam bentuk diagram use case yang menggambarkan konteks sistem yang sedang dikembangkan (Kurniawan, 2018). Use case menggambarkan fungsionalitas suatu sistem dari sudut pandang atau sudut pandang pengguna sistem. Kasus penggunaan menggunakan skenario. Ini adalah deskripsi proses atau langkah yang menjelaskan apa yang akan dilakukan pengguna terhadap sistem atau sebaliknya (Setiyani, 2021).

Ada beberapa komponen kunci yang terdapat pada *use case* adalah :

1. *Actor*, merupakan suatu entitas yang akan menggunakan atau digunakan oleh sistem, pada umumnya pengguna tetapi bisa juga sistem eksternal.
2. *Connections*, merupakan suatu penghubung yang menghubungkan antara actor ke use case.
3. *Relationships*, yaitu suatu hubungan antara aktor itu sendiri dengan use case.

Simbol-simbol *Use Case Diagram* terlihat seperti table 2.2


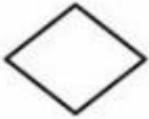
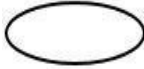

Table 2. 2 Simbol-Simbol Use Case Diagram

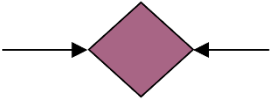

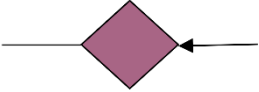

SIMBOL	NAMA	KETERANGAN
	Actor	Mewakili peran orang, sistem lain, atau alat ketika berkomunikasi
	Use Case	Abstraksi dan interaksi antara sistem dan actor
	Association	Abstraksi dari penghubung antara actor dan use case
	Generalization	Menunjukkan spesialisasi actor untuk berpartisipasi dengan use case
	Include	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
	Extend	Menunjukkan bahwa suatu use case merupakan tambahan fungsionalitas dari use case lainnya jika suatu kondisi terpenuhi.

### 2.2.13 Entity Relationship Diagram (ERD)

ERD adalah jaringan yang secara abstrak menyusun buah data yang disimpan dalam suatu sistem. ERD adalah alat desain untuk pemodelan *Database*. Tujuan pembuatan ERD dalam suatu organisasi adalah untuk mendokumentasikan data yang ada dengan membuat model yang menunjukkan hubungan antar data terkait dan dengan mendeskripsikan setiap bagian data serta hubungannya. Simbol-simbol ERD dan kardinalnya dapat dilihat pada Table 2.3

**Table 2. 3** Simbol-Simbol ERD Diagram

SIMBOL	NAMA	KETERANGAN
	Entitas	Kumpulan dari objek yang dapat <u>diidentifikasi</u> secara unik
	Relasi	Hubungan yang terjadi antara salah satu atau lebih entitas. Jenis hubungannya antara lain <i>one to one</i> , <i>one to many</i> , dan <i>many to many</i> .
	Atribut	Mendeskrispikan karakteristik dari entitas atau relasi yang merupakan penjelasan detail dari entitas.
	Penghubung	Hubungan antara entitas dan atributnya dan himpunan entitas dengan himpunan relasinya.


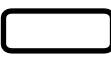




	One to one	Dua objek yang saling terhubung dengan garis tunggal yang menunjukkan hubungan satu-satu.
	Many to One	Beberapa objek di satu sisi terhubung ke satu objek di sisi lainnya.
	One to Many	Satu objek di satu sisi dengan beberapa objek di sisi lainnya.
	Many to Many	Matriks atau jaringan dengan banyak simpul yang saling terhubung.

#### 2.2.14 Activity Diagram

(Sukanto, 2018) menyatakan bahwa diagram aktivitas atau activity diagram menggambarkan alur kerja atau aktivitas dari suatu sistem atau proses bisnis, atau menu yang terdapat dalam perangkat lunak. Menurut Haviluddin dalam (Suendri, 2018), Diagram aktivitas adalah suatu visualisasi yang menampilkan aktivitas-aktivitas, objek, keadaan, transisi keadaan, dan peristiwa dalam sistem. Diagram alur kerja ini secara efektif mewakili perilaku sistem untuk setiap aktivitas yang ada. Berikut dibawah ini adalah simbol-simbol yang digunakan dalam activity diagram disertai dengan keterangan fungsinya sebagaimana dijelaskan pada Tabel 2.4.

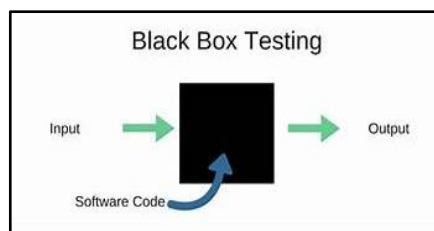


Tabel 2. 4 Simbol Simbol Activity diagram

Simbol	Nama	Keterangan
	Status Awal	Sebuah diagram aktivitas dimulai dengan status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, biasanya dimulai dengan kata kerja.
	Percabangan / <i>Decision</i>	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / <i>Join</i>	Penggabungan dimana yang mana lebih dari satu aktivitas lalu digabungkan jadi satu.
	Status Akhir	Status akhir dalam sebuah diagram aktivitas, menunjukkan bahwa semua aktivitas telah selesai.
	<i>Swimlane</i>	Swimlane memisahkan organisasi bisnis yang bertanggung jawab atas aktivitas yang terjadi.

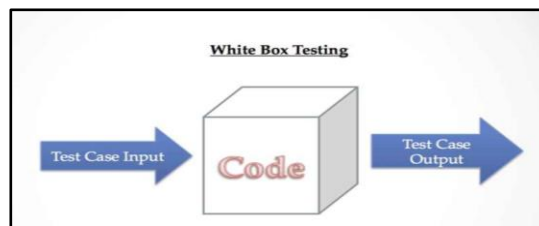
### 2.2.15 Black-Box dan White-Box Testing

Gambar 2.1 menunjukkan sistem alur pengujian *Black Box Testing*. *Black-Box Testing* merupakan Teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak. Blackbox testing bekerja dengan mengabaikan struktur kontrol sehingga perhatiannya difokuskan pada Informasi domain. Black box testing memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program (Jaya, 2018).



**Gambar 2. 7** Sistem pengujian *Black box Testing*

Pengujian black box hanya melibatkan database antara input dan output. Pengujian ini menangani kebutuhan pelanggan dari input yang valid maupun tidak valid. Pengujian black box memiliki peran penting dalam pengujian perangkat lunak yaitu untuk memvalidasi fungsi keseluruhan sistem apakah telah bekerja dengan baik (Parlika, 2020).



**Gambar 2. 8** Sistem pengujian *Whitebox testing*

*White box testing* atau yang dapat diartikan menjadi “pengujian kotak putih” adalah pengujian yang dilakukan untuk menguji perangkat lunak dengan cara menganalisa dan meneliti struktur *internal* dan kode dari perangkat lunak. Lain halnya dengan *black box testing* yang hanya melihat hasil *input* dan *output* dari perangkat lunak, pengujian white box testing berfokus pada aliran input dan output dari perangkat lunak (Setiawan, 2022).

