

BAB II

TINJAUAN PUSTAKA

Sistem informasi jasa percetakan berbasis *web* merupakan salah satu jenis aplikasi teknologi informasi yang menghubungkan industri percetakan dengan platform digital berupa *website*. Berikut landasan teori yang relevan:

2.1 Pengertian Sistem

Suatu sistem dapat didefinisikan sebagai kumpulan komponen atau elemen saling berhubungan dan bekerja sama untuk mencapai tujuan tertentu. Sistem tersebut dapat berupa sistem teknologi informasi, sistem lalu lintas, sistem pendidikan, sistem kontrol dan banyak sistem lain yang digunakan dalam berbagai bidang kehidupan. Setiap sistem memiliki karakteristik dan elemennya masing-masing yang unik, namun pada dasarnya tujuan dari sistem ini adalah untuk membantu pengguna untuk mencapai tujuan yang diinginkan dengan lebih efektif dan efisien. Jadi pengembangan dan manajemen sistem sangat baik penting untuk sukses di banyak bidang kehidupan.

Pemahaman pakar terhadap sistem adalah sebagai berikut: Menurut Raymond Mc. Leo Jr. Kutipan [1] menyatakan bahwa “sistem adalah seperangkat elemen terintegrasi yang memiliki tujuan yang sama untuk mencapai tujuan. Organisasi seperti perusahaan atau area fungsional cocok dengan definisi ini”.

2.2 Karakteristik Sistem

Sifat-sifat suatu sistem adalah sifat-sifat atau sifat-sifat yang dimiliki sistem. Secara umum suatu sistem dapat mempunyai beberapa fungsi digunakan sebagai panduan untuk pemahaman dan analisis sistem yang efektif, seperti tujuan, batasan, lingkungan, struktur dan perilaku sistem. Setiap fitur sistem memegang peranan penting dalam menentukan

efisiensi sistem, cara memperbaikinya dan cara mengoptimalkan kinerja sistem ukuran. Oleh karena itu, memahami sifat-sifat sistem sangat penting bagi siswa pakar sistem dan pengembang teknologi untuk menciptakan sistem yang optimal dan efektif.

Berikut fungsi sistem atau fungsi khusus yaitu [1]:

1. Bagian-bagian sistem

Sistem terdiri dari beberapa komponen yang saling berhubungan saling berkaitan, yang berarti bersama-sama membentuk satu kesatuan.

2. Sistem perbatasan (*border*)

Adalah suatu wilayah yang membatasi sistem di antaranya dengan sistem lain atau lingkungan eksternal. Batasan sistem menunjukkan ruang lingkup sistem.

3. Lingkungan luar (*environment*)

Adalah segala sesuatu yang berada di luar batas-batas sistem mempengaruhi pengoperasian sistem.

4. Sistem komunikasi (*interface*)

Merupakan media penghubung antar subsistem dengan subsistem lainnya. Sumber diaktifkan dengan metode koneksi ini aliran daya yang tersedia dari satu subsistem ke subsistem lainnya.

5. Masukan Sistem

Masukan merupakan hasil energi yang diberikan didalam sistem. masukannya dapat berupa masukan pemeliharaan (*maintenance input*) dan masukan sinyal (*input sinyal*). *Input* pemeliharaan adalah energi input.

6. Keluaran (*output*)

Sistem merupakan hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan keluaran gas buang yang tersisa dapat masukan ke subsistem lain. Contohnya adalah panas yang dihasilkan oleh sistem komputer menjadi keluaran yang

tidak berguna saat mencetak data diperlukan.

7. Prosesor sistem

Suatu sistem mungkin memiliki bagian prosesor yang sedang berjalan mengubah masukan menjadi keluaran.

8. Sasaran sistem

Sistem harus mempunyai tujuan atau sasaran (*goal*) yang sebenarnya menentukan masukan dan keluaran yang dibutuhkan sistem sistem yang dibuat.

2.3 Kriteria Sistem

Kriteria sistem yang dibutuhkan oleh pengguna dapat berbeda-beda tergantung penggunanya jenis sistem yang digunakan. Namun secara umum, kriteria sistem pengguna membutuhkan kinerja sistem yang cepat dan efisien, keandalan sistem keamanan informasi, kemudahan penggunaan dan aksesibilitas sistem, serta kemampuan sistem untuk berintegrasi dengan sistem lain. Saat mengembangkan suatu sistem, pakar sistem harus memperhatikan kebutuhan pengguna dan menentukan kriteria sistem yang jelas dan spesifik memastikan bahwa sistem yang dibangun memenuhi kebutuhan pengguna dan berkendara secara optimal. Kriteria sistem yang terpenuhi meningkatkan kepuasan pengguna dan menciptakan pengalaman pengguna yang lebih baik menggunakan sistem. Sehingga memenuhi kriteria sistem Apa yang dibutuhkan pengguna sangat penting untuk keberhasilan sistem.

Selain itu, kriteria sistem membantu menentukan prioritas internal pengembangan sistem, perbaikan dan peningkatan kinerja sistem memberikan panduan untuk membandingkan sistem yang berbeda. Karena, Pemahaman kriteria kebutuhan sistem sangat penting bagi para ahli pengembang sistem dan teknologi untuk membangun sistem yang berkualitas dan dapat memenuhi kebutuhan pengguna.

Kriteria sistem yang baik adalah sebagai berikut [1]:

1. Penggunaan sistem harus mampu menghasilkan informasi yang tepat waktu dan relevan untuk proses pengambilan keputusan.
2. Sistem perekonomian minimal harus mampu menghasilkan nilai tambah pada harga.
3. Keandalan *Output* dari sistem harus memiliki akurasi yang tinggi dan dapat beroperasi secara efektif dan efisien.
4. Kekuatan sistem harus cukup sederhana sehingga struktur dan pengoperasiannya dapat berjalan dengan baik prosedur yang mudah dimengerti dan mudah diikuti.
5. Fleksibilitas Sistem harus cukup fleksibel untuk beradaptasi terhadap perubahan.

2.4 Pengertian Dasar Informasi

Di dunia modern saat ini, pengetahuan sangatlah penting dan mempunyai peranan sebagai kunci untuk membuka pintu kemajuan dan pembangunan. Misalnya, dalam bisnis, informasi dapat membantu para pebisnis dalam mengambil keputusan strategi yang tepat dan efektif. Pada saat yang sama dalam pendidikan, informasi untuk membantu siswa memperoleh pengetahuan dan keterampilan yang diperlukan semoga sukses di masa depan.

Menurut sebagian orang, pengetahuan itu sendiri memiliki beberapa arti.

Menurut Raymond Mc. Kutipan dari Leo Jr. [1] menyatakan bahwa “Pengetahuan adalah informasi yang diolah atau informasi yang mempunyai makna, informasi sebenarnya berasal dari data, yang kemudian diolah sehingga menjadi data ini penting bagi pengguna”.

2.5 Sistem Informasi

Sistem adalah sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Sedangkan informasi adalah data yang sudah diolah, dibentuk atau dimanipulasi sesuai dengan keperluan tertentu. Pekerjaan informasi adalah

pekerjaan yang meliputi pengumpulan data, penyebaran data dengan meneruskannya ke unit lain atau langsung diolah menjadi informasi, kemudian informasi tersebut diteruskan ke unit lain. Semua pekerjaan data dan informasi dewasa ini sudah dikerjakan dengan bantuan komputer[2].

2.6 Domain Dan Hosting

Secara sederhana, *domain* ibarat sebuah alamat kantor, sementara *hosting* adalah kantornya. Jadi, jika mau menuju kantor itu harus lewat alamat *domain*. Tentu karena di *internet* diberikan ketentuan dalam membuat alamat, misalnya www.surahman.co, www.yahoo.com, www.wordpress-theme.asia [3].

Hosting adalah penyewaan tempat untuk menampung data-data yang diperlukan oleh sebuah *website* sehingga dapat di-akses lewat *Internet*. Data yang dimaksud dapat berupa berkas gambar, surat elektronik, aplikasi/programysekrip dan basis data. *Dedicated Hosting* adalah sebuah layanan *hosting* yang dikelola oleh penyedia layanan *hosting*, Klien menyewa seluruh server dan tidak membagikan server yang disewa dengan siapa saja. Keuntungannya adalah performa dan kerahasiaan yang lebih baik, karena server tidak dipakai oleh pihak ketiga. [4].

2.7 Website

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 dari semuanya [5]. *Website* merupakan sarana untuk menyajikan informasi kepada konsumen, menerima pemesanan, dan memberikan akses yang mudah kepada layanan yang ditawarkan oleh bisnis percetakan.

2.8 Sistem Informasi Berbasis Web

Sistem informasi *online* merupakan salah satu jenis sistem informasi dirancang untuk bekerja melalui *internet* atau *intranet*. Sistem ini bisa dapat diakses melalui *browser web* dari perangkat apa pun yang terhubung ke *internet*, seperti komputer, laptop, ponsel pintar, atau *tablet*. Sistem informasi berbasis web memungkinkan pengguna untuk mengakses dan memproses data secara *real time* dari mana saja selama Anda memiliki koneksi *internet*. Situs *web* adalah “lingkungan dengan banyak halaman satu sama lain (*hyperlink*) dimana situs mempunyai tugas informasi berupa teks, gambar, *video*, *audio* dan animasi atau kombinasinya semuanya”. [6]

Sistem jenis ini banyak digunakan oleh perusahaan, instansi pemerintah, atau organisasi lain untuk proses bisnis, manajemen informasi atau pelayanan publik Beberapa contoh sistem informasi berbasis *web* adalah sistem manajemen inventaris, sistem manajemen proyek, sistem manajemen konten, sistem manajemen akademik dan sistem manajemen personalia.

Keuntungan menggunakan sistem informasi *online* adalah kemudahan penggunaan dan pengoperasian, fleksibilitas dan kemungkinan skalabilitas banyak pengguna menganggap sistem ini mudah digunakan. Namun, Keamanan sistem juga perlu diperhatikan agar data dapat diakses dan data yang disimpan dalam sistem terlindungi dengan baik.

2.9 HTML

HTML adalah singkatan dari *Hypertext Markup Language*. HTML memungkinkan seorang user untuk membuat dan menyusun bagian paragraf, heading, link atau tautan, dan *blockquote* untuk halaman *web* dan aplikasi HTML adalah sebuah bahasa markup yang digunakan untuk membuat sebuah halaman *web* [7]. HTML inilah yang menyusun sebuah halaman *web* menjadi sebagaimana yang kita lihat melalui *browser* (penjelajah *internet*).

Jadi, belajar HTML dan mengerti bagaimana cara HTML bekerja akan sangat penting jika kamu ingin terjun ke dunia *web development*.

2.10 CSS

CSS adalah singkatan dari *Cascading Style Sheets* yaitu kumpulan kode program yang digunakan untuk mendesain atau mempercantik tampilan halaman HTML. Menjelaskan bagaimana elemen-elemen HTML ditampilkan di layar. Dengan CSS kita bisa mengubah desain dari text, warna, gambar dan latar belakang dari (hampir) semua kode tag HTML sehingga dapat mengontrol tata letak beberapa halaman *web* sekaligus. CSS biasanya selalu dikaitkan dengan HTML, karena keduanya memang saling melengkapi. HTML ditujukan untuk membuat struktur halaman *web*. Sedangkan CSS digunakan untuk tampilan dari halaman *web* tersebut [8].

Beberapa kegunaan CSS diantaranya :

1. Mempersingkat penulisan tag HTML Penulisan tag dengan *property* dan lain yang sama tidak perlu dituliskan pada setiap tag HTML.
2. Mempercepat proses *rendering* atau pembacaan HTML karena tidak terdapat pengulangan penulisan,
3. Mudah untuk memelihara skrip : karena CSS dapat dibuat terpisah maka tidak perlu merombak semua elemen atau *property* dalam HTML.
4. CSS dapat melakukan apa yang tidak bisa dilakukan oleh HTML seperti memberikan warna pada *input box* atau scrollbar.

2.11 JavaScript

Pengertian Java Script, JavaScript adalah bahasa pemrograman berbasis teks yang digunakan baik di sisi klien dan sisi server yang memungkinkan seseorang membuat

halaman *web* menjadi interaktif. Jika HTML dan CSS adalah bahasa yang memberikan struktur dan gaya ke halaman *web*, JavaScript adalah yang memberikan elemen interaktif halaman *web* yang melibatkan pengguna. Contoh umum JavaScript yang mungkin sering dilihat atau digunakan setiap hari adalah seperti kotak pencarian di *e-commerce*, video rekap berita yang disematkan di media *online*, atau me-refresh timeline Twitter [9].

JavaScript pertama kali diciptakan oleh Brendan Eich, seorang karyawan Netscape, pada tahun 1995. Netscape kala itu merupakan perusahaan software ternama yang dikenal dengan *web browser* miliknya, Netscape Navigator. Brendan Eich pada awalnya diminta untuk membuat bahasa scripting seperti Java namun dapat diterapkan untuk *browser*. Ia pun mendesain bahasa pemrograman baru dengan menggunakan fitur-fitur yang terinspirasi dari Java, Scheme, dan Self.

2.12 Structured Query Language

Structured Query Language atau yang disingkat SQL adalah sebuah bahasa yang digunakan untuk mengakses data dalam basis data rela Bahasa ini secara *de facto* merupakan bahasa standar yang digunakan dalam manajemen basis data relasional. Saat ini hampir semua server basis data yang ada mendukung bahasa ini untuk melakukan manajemen datanya [10]. SQL merupakan bahasa pemrograman khusus yang digunakan untuk memanajemen data dalam RDBMS. SQL biasanya berupa perintah sederhana yang berisi instruksi-instruksi untuk manipulasi dan pengambilan data pada *relational database* atau database yang terstruktur. Perintah SQL ini sering juga disingkat dengan sebutan *query*.

2.13 XAMPP

XAMPP adalah sebuah aplikasi *open source* terkait pengelolaan server yang dikembangkan oleh *Apache Friends*. Karena bersifat *open source*, aplikasi ini bisa Anda

digunakan secara gratis. Sel ai namanya, X pada XAMPP berarti *cross* platform. Artinya, mendukung berbagai platform seperti Windows, macOS dan Linux. XAMPP sendiri terdiri dari Apache, MariaDB (yang dikembangkan dari MySQL), PHP dan Perl. XAMPP juga memberikan solusi sederhana dan cukup ringan dijalankan, memungkinkan membuat *web* server lokal untuk melakukan pengetesan *website*. XAMPP dapat dijalankan pada Mac dan Linux. Dalam buku ajar ini penggunaan aplikasi XAMPP diimplementasikan pada sistem operasi Windows [10].

2.14 PHP

PHP (*Hypertext Preprocessor*) adalah salah satu Bahasa pemrograman *procedural* (memilih fungsi dan modul yang dapat dipanggil dari program utama) untuk membuat *website* dinamis yang sangat populer saat ini. PHP mendukung pemrograman berbasis objek (pemrograman yang menggunakan kelas dan objek) yang mudah dikembangkan. PHP hanya dapat dieksekusi pada sisi *server* saja, atau disebut sebagai server *side programming*. Berbeda dengan Bahasa pemrograman seperti HTML dan *javascript* yang dapat dieksekusi di *browser (client)* tanpa harus menginstall *web server (server)* [11].

2.15 Codeigniter MVC

Codeigniter MVC adalah lingkungan pengembangan aplikasi PHP berbasis arsitektur terstruktur. *Codeigniter MVC* merupakan salah satu *frame* yang di buat dan di kembangkan oleh Ellislab dan diluncurkan pada tanggal 28 Februari 2006. Pengeritain *Framework* berarti kerangka kerja. Dalam kaitannya dengan Bahasa pemrograman, kerangka kerja yang di maksud adalah kumpulan *function* , *class*, *method* dan aturan strip yan terorganisir sedemikan rupa sehingga memiliki keseragaman penulisan kode dan penempatan folder dalam membangun sebuah aplikasi [12].

Model-View-Controller (MVC) adalah desain arsitektur yang digunakan dalam

perangkat lunak, dan *CodeIgniter* adalah salah satu *framework* PHP yang mengimplementasikan desain arsitektur ini. Dalam konteks *CodeIgniter*, MVC membagi aplikasi *web* menjadi tiga komponen utama: Model, yang bertanggung jawab untuk mengelola logika bisnis dan mengakses *database*, Tampilan yang mengontrol bagaimana data ditampilkan dan disajikan kepada pengguna dan pengontrol, yang bertindak sebagai perantara antara model dan tampilan, mengendalikan aliran data dalam aplikasi.

Dengan mengimplementasikan MVC di *CodeIgniter*, pengembang dapat mengatur kode mereka secara terstruktur, meningkatkan pemisahan logika bisnis dan pandangan, serta memfasilitasi pemeliharaan dan pengembangan aplikasi web yang skalabel dan mudah dipelajari.

Berikut adalah contoh sederhana penerapan *Model-View-Controller* (MVC) dalam kerangka kerja *CodeIgniter*:

1. **Model (*models/Student_model.php*):** Ini adalah bagian yang mengelola logika bisnis dan akses ke basis data. Dalam contoh ini, kita akan memiliki model untuk entitas "Mahasiswa" yang berinteraksi dengan basis data dan mengambil data mahasiswa.

```
<?php
class Student_model extends CI_Model {
    public function get_students() {
        // Mengambil data mahasiswa dari basis data
        $query = $this->db->get('students');
        return $query->result();
    }
}
?>
```

2. **Controller (controllers/Student.php):** Controller berfungsi sebagai perantara antara Model dan View. Ini mengatur aliran informasi dalam aplikasi.

```
<?php  
  
class Student extends CI_Controller {  
  
    public function __construct() {  
  
        parent::__construct();  
  
        $this->load->model('Student_model');  
  
    }  
  
    public function index() {  
  
        $data['students'] = $this->Student_model->get_students();  
  
        $this->load->view('student_view', $data);  
  
    }  
  
}  
  
?>
```

3. **View (views/student_view.php):** View mengatur tampilan dan presentasi data kepada pengguna.

```
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
    <title>Daftar Mahasiswa</title>  
  
</head>  
  
<body>
```

```
<h1>Daftar Mahasiswa</h1>

<ul>

  <?php foreach ($students as $student): ?>

    <li><?= $student->name; ?></li>

  <?php endforeach; ?>

</ul>

</body>

</html>
```

Dalam contoh di atas, *Controller Student* menerima permintaan dan memanggil model *Student_model* untuk mengambil data mahasiswa dari basis data. Kemudian, data ini dikirim ke *View student_view*, yang bertanggung jawab untuk menampilkan data tersebut kepada pengguna.

Dengan pendekatan ini, aplikasi Anda terstruktur dengan baik sesuai dengan pola MVC, yang memungkinkan pemeliharaan, pengembangan, dan perubahan komponen aplikasi secara terpisah, membuat kode lebih terorganisir dan mudah dipahami.

2.16 MySQL

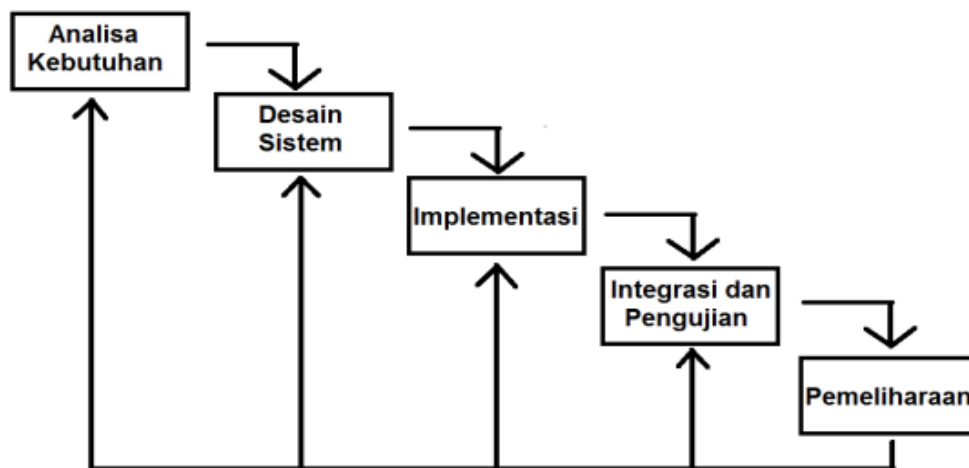
MySQL adalah DBMS yang *open source* dengan dua bentuk lisensi, yaitu *Free Software* (perangkat lunak bebas) dan *Shareware* (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah *database server* yang gratis dengan lisensi *GNU General Public License (GPL)* sehingga dapat Anda 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 dan 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. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user* [13].

2.17 Metode Waterfall

Model *Waterfall* yang sering juga dikenal sebagai model air terjun adalah model proses pertama yang diperkenalkan. Ia sangat mudah dimengerti dan digunakan. Dalam model *Waterfall*, setiap fase harus diselesaikan sebelum fase berikutnya dapat dimulai dan tidak ada fase yang tumpang tindih. Model *Waterfall* adalah pendekatan SDLC paling awal yang digunakan untuk pengembangan perangkat lunak [13]. Metode *Waterfall* adalah metode pengembangan perangkat lunak yang sistematis dan berurutan yang dimulai pada tingkat sistem dan berlanjut melalui analisis, desain, pengkodean, pengujian dan pemeliharaan.

Berikut adalah contoh tahapan dalam metode Waterfall:



Gambar 2.1 Tahapan Metode *Waterfall*

1. Analisis: Pada fase ini, tim pengembangan memeriksa kebutuhan konsumen secara menyeluruh. Mereka berinteraksi dengan pemangku kepentingan untuk mendapatkan pemahaman yang baik tentang persyaratan sistem. Hasil analisis ini berupa dokumen persyaratan yang lengkap dan komprehensif.

2. Desain: Setelah menyelesaikan analisis, tim desain merancang arsitektur dan desain sistem. Ini termasuk merancang struktur database, antarmuka pengguna dan elemen perangkat lunak lainnya. Dokumen desain sistem diperoleh pada akhir fase ini.
3. Implementasi: Pada fase ini, tim pengembang mulai menulis kode berdasarkan desain yang dibuat. Mereka mengimplementasikan semua fitur dan fungsi yang direncanakan selama tahap desain. Implementasi ini berfokus pada pembuatan perangkat lunak yang berfungsi.
4. Pengujian: Setelah kode siap, tim pengujian secara menyeluruh memeriksa kesalahan sistem dan memastikan bahwa semua persyaratan terpenuhi. Pengujian meliputi pengujian fungsional, integrasi, dan kinerja.
5. Pengiriman: Ketika pengujian berhasil dan perangkat lunak dianggap siap, perangkat lunak dikirimkan ke pelanggan atau pengguna akhir.
6. Pemeliharaan: Langkah terakhir melibatkan pemeliharaan perangkat lunak. Hal ini mencakup pemecahan masalah, *debugging*, dan penyempurnaan sesuai kebutuhan sepanjang siklus hidup produk.

Metode air terjun cocok untuk proyek dengan persyaratan yang sangat jelas yang tidak berubah selama proyek berlangsung. Namun, jika diperlukan perubahan atau koreksi setelah dimulainya tahap analisis, metode ini bisa menjadi kurang fleksibel. Oleh karena itu, pendekatan ini sering digunakan untuk proyek skala kecil dengan persyaratan yang stabil.

2.18 Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan salah satu standart bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek [14].

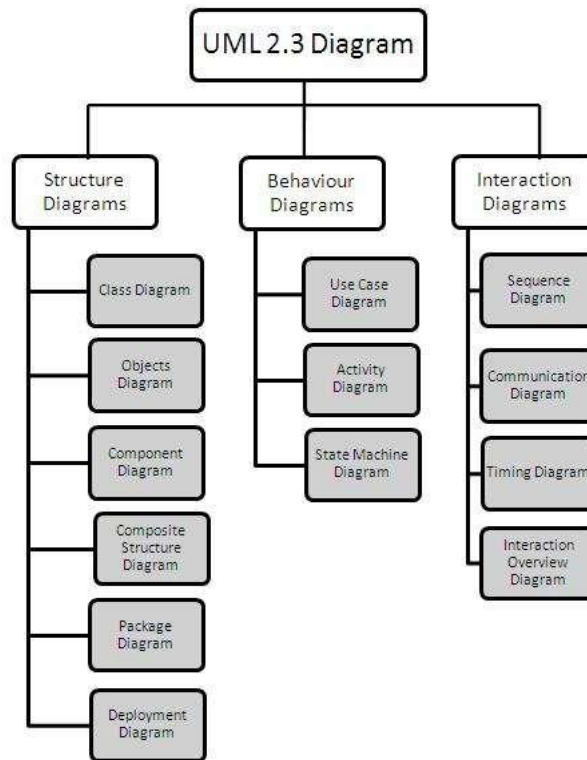
UML adalah bahasa dan notasi standar yang digunakan dalam rekayasa perangkat lunak

untuk mendeskripsikan, merancang, dan mendokumentasikan desain sistem perangkat lunak.

UML menyediakan kerangka kerja yang kaya dan terstruktur untuk menggambarkan berbagai aspek sistem perangkat lunak, termasuk struktur komponen, perilakunya, dan interaksi antar komponen tersebut. UML memungkinkan pengembang untuk mengkomunikasikan ide dan konsep mereka dengan lebih jelas kepada anggota tim dan pemangku kepentingan lainnya, memungkinkan pemahaman yang lebih baik dan pelacakan proyek yang lebih efisien.

UML memiliki berbagai diagram yang digunakan untuk menggambarkan berbagai aspek dari suatu sistem, seperti diagram kelas untuk menggambarkan struktur kelas dan hubungan antar kelas, diagram urutan untuk menggambarkan interaksi objek dari waktu ke waktu, dan diagram aktivitas untuk menggambarkan aliran aktivitas dalam suatu sistem.

Selain itu, UML juga mendukung pemodelan objek, yang memungkinkan pengembang menentukan properti dan perilaku objek dalam sistem perangkat lunak. Hal ini menjadikan UML alat yang sangat berguna sepanjang siklus pengembangan perangkat lunak, mulai dari desain hingga penerapan dan pemeliharaan. UML versi 2.3 terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam- macam diagram dapat dilihat pada Gambar 2.2 :



Gambar 2.2 Macam-Macam UML Diagram

Berdasarkan Gambar 2.2 berikut penjelasan singkat dari pembagian kategori tersebut:

- a. *Structure diagram*, merupakan kumpulan diagram yang digunakan untuk menggambarkan struktur statis dari sistem yang dimodelkan.
- b. *Behavior diagram*, merupakan kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada suatu sistem.
- c. *Interaction diagram*, merupakan kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sistem.

2.19 Entity Relationship Diagram (ERD)

Pemodelan awal basis data yang paling banyak digunakan adalah *Entity Relationship Diagram* (ERD). ERD digunakan untuk pemodelan basis data relational. Diagram relasi

entitas atau ERD merupakan suatu diagram dalam bentuk gambar atau simbol yang mengidentifikasi tipe dari entitas di dalam suatu sistem yang diuraikan dalam data dengan atributnya, dan menjelaskan hubungan atau relasi diantara entitas tersebut. Atau dapat dikatakan bahwa ERD adalah model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. ERD menekankan pada struktur dan *relationship* data [13].

Entity-Relationship Diagram (ERD) adalah alat visual yang digunakan dalam pengembangan basis data untuk menggambarkan hubungan antara entitas atau objek dalam suatu sistem informasi. ERD menggambarkan entitas sebagai tabel dalam basis data dan menggambarkan hubungan antara entitas-entitas tersebut, seperti hubungan *one-to-one*, *one-to-many*, atau *many-to-many*. ERD membantu dalam merancang struktur basis data dengan jelas dan memahami bagaimana data akan disimpan, diakses, dan dihubungkan dalam suatu sistem. Diagram ini sangat berguna dalam analisis kebutuhan bisnis dan pengembangan aplikasi, serta membantu dalam menjaga integritas data dan mengoptimalkan efisiensi basis data.


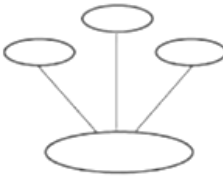









ERD memiliki fungsi yang dapat dikembangkan setelah berdiskusi dalam perancangan sistem sebagai berikut :

1. Memungkinkan Anda memfasilitasi analisis basis data dengan cara yang cepat dan mudah.
2. Dapat membuat hubungan antara data yang mempunyai hubungan dengan objek yang diasosiasikan dengan hubungan tersebut.
3. Mendokumentasikan informasi yang terkandung dalam *database* dengan menganalisis dan mengidentifikasi setiap objek atau entitas dan hubungannya.
4. Pengujian dapat dilakukan pada model yang diproduksi.

Secara umum, ada 3 model data dalam ERD; konseptual, logis dan fisik. Dalam pembahasan sebagai berikut:

1. Model data konseptual adalah model ER tertinggi yang berisi informasi rinci. Templat ini juga dapat digunakan untuk menentukan kumpulan data referensi utama yang digunakan oleh organisasi.
2. Model data logis, atau model ER, berisi informasi lebih rinci daripada model konseptual.
3. Model data fisik, yaitu model ER yang dikembangkan dari model data logis, yang bertindak sebagai *database*.

Memahami notasi, komponen, dan simbol ERD sangat penting untuk merancang ERD yang konsisten dan benar. Berikut ini adalah pembahasannya.

Simbol	Keterangan	Simbol	Keterangan
	= Entity		= Atribut Komposit
	= Weak Entity		= Atribut Derivatif
	= Relationship		= Total Participation of E2 in R
	= Identifying Relationship		= Cardinality Ratio 1:N For E1;E2 in R
	= Atribut		
	= Atribut Kunci		
	= Atribut Multivalued		

Gambar 2.3 Simbol ERD



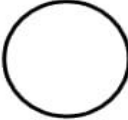



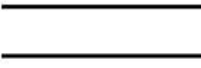

Berdasarkan simbol ERD dan keterangan dari notasi diatas, dapat dijelaskan sebagai berikut :

1. Simbol persegi panjang menggambarkan suatu entitas.
2. Bentuk belah ketupat menggambarkan suatu relasi atau proses.
3. Bentuk lingkaran atau elips menggambarkan suatu atribut.
4. Garis dapat menggambarkan suatu hubungan atau penghubung.

2.20 Diagram Alir Data (DFD)

Analisis sistem menggunakan banyak teknik grafis untuk menggambarkan sistem informasi. Salah satu metode yang populer adalah menggambar satu set diagram aliran data. *Data flow diagram* (DFD) menggunakan berbagai simbol untuk menunjukkan bagaimana sistem mengubah data input menjadi informasi yang berguna [15]. *Data Flow Diagram* (DFD) adalah bagian dari metode analisis dan desain terstruktur. Ini adalah alat visual untuk menggambarkan model logis dan mengekspresikan transformasi data dalam suatu sistem. DFD mencakup mekanisme untuk memodelkan aliran data. DFD mendukung dekomposisi untuk menggambarkan aliran data dan operasi secara rinci. DFD tidak dapat memberikan informasi mengenai urutan kegiatan. Oleh karena itu, DFD bukanlah suatu metode pemodelan proses atau prosedur.

Pemodelan DFD diawali dengan pembuatan *context diagram*. Secara simbol, DFD dan *context diagram* menggunakan jenis dan bentuk simbol yang sama, namun secara aturan terdapat perbedaan antara pemodelan DFD dan *context diagram*, dimana pada *context diagram* hanya diizinkan sebuah simbol proses saja sedangkan pada DFD dapat lebih dari satu simbol proses. Selain itu *context diagram* ditujukan untuk menggambarkan interaksi sistem dengan lingkungan luar, sedangkan pada DFD juga digambarkan hubungan antar proses didalam sistem.

Notasi Yourdon DeMarco	Notasi Gane & Sarson	Deskripsi
		Simbol Entitas Eksternal / Terminator menggambarkan asal atau tujuan data di luar system
		Simbol lingkaran menggambarkan entitas atau proses dimana aliran data masuk ditransformasikan ke aliran data keluar
		Simbol aliran data menggambarkan aliran data
		Simbol file menggambarkan tempat data disimpan

Gambar 2.4 Simbol DFD

2.21 Referensi Terkait

Terdapat beberapa penelitian yang berkaitan dengan penelitian dalam laporan ini, penelitian tersebut terlampir pada tabel 1 di bawah ini.

Tabel 2.1 Penelitian Terkait

No	Penulis Author	Judul Artike	Hasil
1.	Muslim Ramli (2022)	IMPLEMENTASI MODEL SPIRAL UNTUK RANCANG BANGUN SISTEM INFORMASI PENGELOLAAN PASIEN LABORATORIUM PATOLOGI ANATOMI UNIVERSITAS SUMATERA UTARA	<p>Kesimpulan :</p> <ol style="list-style-type: none"> 1. Memudahkan admin dan dokter untuk input hasil pemeriksaan pada media desktop. 2. Informasi admin dapat Memantau berkas pemeriksaan yang sudah selesai maupun yang masih pending melalui sistem notifikasi. 3. Data yang tersimpan dalam database terdokumentasi dengan baik. 4. Hasil pemeriksaan dan tagihan disajikan dalam bentuk digital maupun

			<p>cetakan.</p> <p>5. Laporan yang disajikan dapat di-generate secara real time dan historical.</p>
2.	Adi Supriyatna (2017)	SISTEM INFORMASI PEMESANAN JASA PERCETAKAN	<ol style="list-style-type: none"> 1. Memudahkan konsumen mendapatkan informasi barang. 2. Memudahkan interaksi dengan konsumen. 3. Sebagai penunjang keputusan oleh manajemen untuk proses pengambilan keputusan strategic
3.	Laser Narindro l , Hendra Maya Topani (2017)	PERENCANAAN-BISNIS-SISTEM-INFORMASI PERCETAKAN ONLINE BERBASIS APLIKASI WEB	<ol style="list-style-type: none"> 1. Segmentasi pelanggan yang terukur dan terarah menggunakan media online. 2. Mempermudah pengembangan bisnis percetakan dengan menggunakan media online berbasis aplikasi web sebagai sarana promosi dan penjualan
4.	Agna Rahmah Harsinta, Sri Rezeki Candra Nursari (2020)	PERANCANGAN SISTEM INFORMASI PENGELOLAAN JASA PERCETAKAN AGNA ADVERTISING BERBASIS WEB	<ol style="list-style-type: none"> 1. Sumber data untuk customer untuk mendapatkan informasi ,memperluas jangkauan pemasaran. 2. Memudahkan interaksi dengan konsumen. 3. Memudahkan administrasi serta pengelolaan data

Dengan memanfaatkan landasan teori ini, pengembangan Sistem Informasi Jasa Percetakan Berbasis *Website* dapat lebih terarah, relevan, dan mendukung tujuan serta manfaat yang diinginkan.

2.22 Hipotesis

Penerapan sistem informasi layanan cetak berbasis *web* dengan metode *Waterfall* adalah pendekatan ini memberikan kerangka kerja yang terstruktur dan berurutan untuk pengembangan sistem yang menghasilkan informasi yang akurat dan rinci. Dengan metode *Waterfall*, penelitian dapat melalui langkah-langkah yang jelas mulai dari perencanaan hingga pelaksanaan dan pemeliharaan, sehingga memudahkan pengumpulan informasi yang diperlukan untuk penelitian ini. Hal ini membantu penelitian untuk mengevaluasi dampak sistem informasi berbasis *web* terhadap efisiensi keseluruhan dan layanan layanan pencetakan.

Selain itu, metode *Waterfall* juga memungkinkan partisipasi aktif semua pihak baik itu individu, komunitas atau kelompok yang memiliki hubungan kepentingan terkait dalam seluruh tahapan pembangunan. Penting untuk mengumpulkan perspektif yang berbeda dan memahami bagaimana sistem ini mempengaruhi semua pihak terhadap kepentingan yang berbeda, seperti manajemen bisnis percetakan dan konsumen. Kesimpulan awal ini menunjukkan bahwa metode *Waterfall* dapat menjadi cara yang efektif untuk mengembangkan sistem informasi layanan pencetakan *online* yang dapat digunakan untuk mengevaluasi implementasi dan dampaknya secara komprehensif pada berbagai bidang bisnis.