

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.1.1 Sistem

Sistem pada dasarnya merupakan elemen elemen yang berkumpul dan saling terintegrasi, untuk mencapai suatu tujuan tertentu. Menurut Hutahean (2014:2) sistem merupakan suatu jaringan kerja dari prosedur yang ada dan saling berhubungan, berkumpul bersama – sama untuk melakukan kegiatan atau untuk mencapai target tertentu.

Sistem sendiri memiliki suatu landasan atau karakteristik yang harus di anut atau di ikuti untuk menjadikan itu sistem yang suturenya. Menurut Sutabri (2016) terdapat beberapa karakteristik dari suatu sistem agar dapat dikatakan sebagai sebuah sistem yang seutuhnya, yaitu :

a. Komponen Sistem

Suatu sistem memiliki sejumlah komponen yang saling berinteraksi yang artinya komponen tersebut saling bekerja sama membentuk suatu kesatuan. Komponen komponen sistem tersebut dapat berupa suatu bentuk subsistem dan dari subsistem tersebut memiliki fungsi dan sifat untuk menjalankan sistem yang lebih besar yang biasa disebut “supra sistem”.

b. Batasan Sistem

Suatu sistem memiliki batasan tertentu dari satu sistem dan sistem lainya. Batasan sistem ini bertujuan menyatakan suatu sistem dipandang sebagai suatu kesatuan yang tidak terpisahkan.

c. Lingkungan Luar Sistem

Bentuk atau hal apapun yang terdapat diluar lingkup atau batasan sistem yang dapat mempengaruhi operasi sistem dapat disebut lingkup luar sistem.

Lingkungan luar sistem ini dapat bersifat menguntungkan dan merugikan sistem yang ada, dengan hal tersebut lingkungan luar sistem ini harus tetap dijaga dan dipelihara agar tidak mengganggu kelangsungan hidup sistem yang ada.

d. Penghubung Sistem

Penghubung sistem yang relevan adalah media yang digunakan sebagai penghubung antara suatu subsistem dengan subsistem lainnya, seperti antarmuka.

e. Masukan Sistem

Input atau masukan kedalam sebuah sistem disebut sebagai masukan sistem. Masukan sistem dapat berupa pemeliharaan dan *signal input* hal tersebut dilakukan agar sistem bekerja dengan baik dan sebagaimana semestinya.

f. Keluaran Sistem

Hasil atau proses yang telah di olah akan diklasifikasikan sebagai keluaran yang dihasilkan dari proses inputan dan menjadikan suatu informasi yang berguna.

g. Pengolahan Sistem

Pengolahan sistem ini merupakan proses perubahan sistem dari sebuah intputan untuk menghasilkan output sesuai dengan tujuan dari sebuah sistem.

h. Sasaran Sistem

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministic. Kalau suatu sistem tidak memiliki sasaran atau tujuan.

2.1.2 Informasi

Informasi merupakan kumpulan data yang sudah diolah untuk dijadikan sebuah informasi yang data dengan kualitas yang akurat dan berkualitas.

Menurut Kusrini (2002:22) Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berarti bagi pengguna, yang bermanfaat untuk mendukung pengambilan keputusan. Informasi memiliki beberapa manfaat antara lain sebagai berikut :

- a. Berfungsi untuk meningkatkan pengetahuan bagi pengguna/pembaca.
- b. Berfungsi untuk mengurangi ketidakpastian dalam proses pengambilan keputusan pemakai
- c. Berfungsi dalam menggambarkan keadaan yang sebenarnya dari suatu hal.

Kualitas Informasi bias yang diakibatkan kesalahan pada cara pengukuran dan pengumpulan, kegagalan mengikuti proses prosedur, kehilangan atau data tidak terproses, kesalahan perekaman dan koreksi data, dan ketidak fungsian sistem. Informasi harus memiliki sifat yang akurat, tepat, dan relevan, yaitu :

- a. Akurat berarti informasi harus bebas dari kesalahan atau bisa dan menyesatkan.
- b. Tepat berarti tepat waktu informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang akan mengalami penurunan nilai atau sama sekali tidak bernilai, karena informasi merupakan landasan dalam pengambilan keputusan.
- c. Relevan berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang berbeda-beda nilai Informasi ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya. Suatu informasi dapat dikatakan bernilai bila manfaat yang diberikan atau manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.

2.1.3 Sistem Informasi

Sistem informasi secara umum yaitu suatu yang menyediakan informasi untuk manajemen pengambilan keputusan atau kebijakan dan menjalankan operasional dari kombinasi dari orang, teknologi dan prosedur yang

terorganisir. Menurut Sidh (2013) Sistem informasi merupakan satuan komponen yang saling terhubung mengumpulkan, memproses, menyimpan, dan mendistribusikan informasi untuk mendukung keputusan dan kendali dalam suatu organisasi.

2.1.4 Jasa

Jasa merupakan suatu aktivitas atau kegiatan ekonomi dengan output yang tidak menghasilkan sebuah produk dalam pengertian fisik. Menurut Philip Kotler (2017), Jasa memiliki ciri ciri tertentu untuk dapat dikatakan sebagai jasa yaitu :

- a. Suatu hal yang tidak memiliki wujud, Tetapi dapat memenuhi kebutuhan konsumen.
- b. Segala proses produksi jasa dapat menggunakan atau tidak menggunakan bantuan suatu produk fisik.
- c. Penggunaan jasa tidak akan mengakibatkan peralihan hak atau kepemilikan.
- d. Terjadi interaksi antara penyedia jasa dan pengguna jasa.

2.1.5 Jasa Penyewaan

Penyewaan adalah kontrak di mana pembayaran dilakukan sebagai imbalan atas penggunaan sementara barang, lahan atau properti oleh pihak lain. Hal-hal yang bisa disewa datang dalam berbagai harga, lama sewa, dan jenis. Menurut Subekti (2005) penyewaan adalah pengalihan hak penyewa untuk menggunakan suatu barang, jasa, atau objek dari pemilik barang, jasa, atau objek kepada penyewa dalam jangka waktu tertentu dengan imbalan penyewa membayar uang kepada pemilik barang, tempat, jasa, atau objek tersebut sesuai dengan pihak perjanjian kedua.

2.1.6 Lahan/Properti

Lahan merupakan kesatuan berbagai sumberdaya daratan yang saling berinteraksi membentuk suatu sistem struktural dan fungsional. Sifat dan perilaku lahan ditentukan oleh macam sumberdaya yang merajai dan macam

serta intensitas interaksi yang berlangsung antar sumberdaya. Menurut Asfiati & Zurkiyah (2021) tanah/lahan terbuka berkaitan dengan arti atau tujuan pembangunan sosial ekonomi bagi masyarakat, yang dapat berupa ruang terbuka, tanah subur, atau tanah yang tidak diolah. Definisi ini menunjukkan bagaimana tanah adalah lanskap karena merupakan tempat di mana semua makhluk hidup ada dan memanfaatkan tanah.

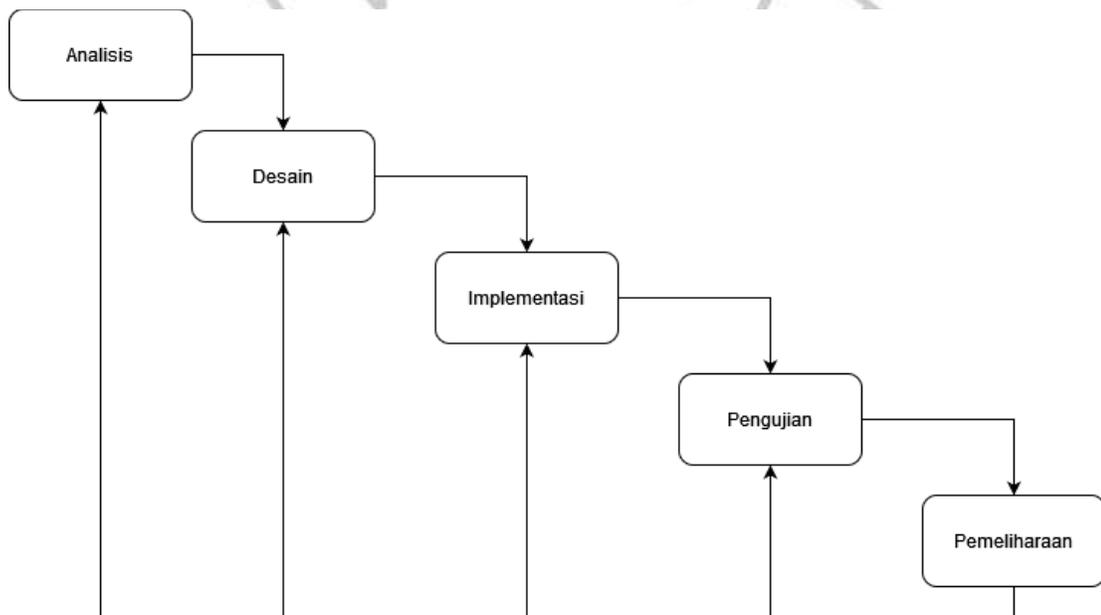
Properti bila merujuk pada KBBI bisa di artikan sebagai harta berupa tanah dan bangunan, sarana dan prasarana yang merupakan bagian yang tidak terpisahkan. Menurut Lestari, Cahyono, & Susanto (2019) properti adalah konsep hukum yang mencakup kepentingan, hak dan manfaat yang berkaitan dengan suatu kepemilikan. Properti terdiri atas hak kepemilikan, yang memberika hak kepada pemilik untuk suatu kepentingan tertentu (*specific interest*) atau sejumlah kepentingan atas apa yang dimilikinya.

2.1.7 SDLC (*System Development Life-Cycle*)

Menurut Permana & Romadlon (2019), mengembangkan suatu sistem informasi banyak orang menggunakan suatu metode dalam tahapan pengembangan sistem. Metode pengembangan sistem informasi memiliki daur hidup, daur hidup tersebut yang dinamakan SDLC (*Systems Development Life Cycle*) atau daur hidup pengembangan sistem. Metode SDLC merupakan salah satu metodologi klasik yang sering digunakan untuk mengembagkan, memelihara, dan menggunakan sistem informasi.

Tahapan yang harus diselesaikan untuk membuat aplikasi perangkat lunak dijelaskan dengan model SDLC yang dipakai dalam proses pengembangan software. Fase ini biasanya meliputi analisis kebutuhan perangkat lunak, desain, implementasi, pengujian, serta pemeliharaan. Dengan menggunakan SDLC, pengembangan perangkat lunak dapat memastikan kelancaran proses pengembangan perangkat lunak dan menghasilkan produk akhir yang sesuai dengan kebutuhan pengguna. Model *waterfall* merupakan salah satu dari penerapan metode SDLC. Metode air terjun atau yang sering

disebut metode *waterfall* sering dinamakan siklus hidup klasik (*classic life cycle*), nama model ini sebenarnya adalah “*Linear Sequential Model*” dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan analisis kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan desain, implementasi, pengujian, dan yang diakhiri dengan pemeliharaan dukungan pada perangkat lunak lengkap yang dihasilkan. Dapat dilihat pada **Gambar 2.1** dibawah :



Gambar 2. 1 SDLC waterfall

Sumber : (Sukamto, 2022)

Berikut adalah penjelasan dari tahapan **Gambar 2.1** diatas mengenai fase – fase *waterfall* menurut Sukamto (2022) yaitu :

1. Analisis Kebutuhan Perangkat Lunak

Menurut Sukamto (2022), tahap ini merupakan proses pengumpulan kebutuhan yang dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak

seperti apa yang cocok dan dibutuhkan oleh pengguna. Informasi dapat diperoleh melalui wawancara, diskusi atau survei langsung.

2. Desain

Menurut Sukamto (2022), pada tahap ini, terjadi proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranlasikan kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

3. Implementasi

Menurut Sukamto (2022), pada tahap ini, merupakan proses pengodean dari desain. Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Menurut Sukamto (2022), pada tahap ini, sistem dilakukan pengujian apakah sistem sepenuhnya atau sebagian memenuhi persyaratan sistem. Tahap ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan dan dibutuhkan.

5. Pemeliharaan

Menurut Sukamto (2022), ini adalah tahap akhir dari metode waterfall. Perangkat lunak yang sudah jadi dijalankan suatu saat dapat mengalami perubahan, perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

2.1.8 *Structured Analysis and Design (SSAD)*

Menurut Falaah Abdussalaam (2021) metode *structured analysis and design* merupakan hasil dari pemograman terstruktur, melalui pendekatan terstruktur permasalahan yang kompleks di organisasi dapat dipecahkan dan hasil dari sistem akan mudah untuk di pelihara dan dapat memenuhi kebutuhan penggunanya.

Berikut merupakan beberapa kelebihan dan kekurangan dari penerapan metode *structured analysis and design* dalam perancangan sebuah sistem antara lain yaitu Zainul (2015) :

Kelebihan *structured analysis and design* :

- Merupakan metode yang diketahui secara umum pada berbagai industry.
- Merupakan metode yang matang dikarenakan sudah digunakan dan diterapkan begitu lama.
- Memungkinkan untuk melakukan validasi antara berbagai kebutuhan.

Kekurangan *structured analysis and design* :

- Metode ini tidak selalu memenuhi kebutuhan pengguna.
- Sedikit sekali manajemen langung terkait dengan *structured analysis and design*.
- Interaksi antara analisis dan pengguna tidak komperhesif, karena sistem telah di definisikan dari awal, sehingga tidak adaptif pada perubahan.

2.1.9 **Basis Data**

Menurut Andaru (2018), konsep dasar basis data merupakan suatu atau banyak kumpulan dari catatan atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di

dalamnya. Basis data mengacu pada koleksi dari data-data yang saling berhubungan, dan perangkat lunaknya seharusnya mengacu sebagai sistem manajemen basis data (*database management system/DBMS*).

2.1.10 Alat Pendukung (Tool System)

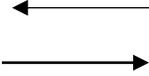
Alat pendukung untuk menggambarkan suatu sistem adalah Teori Pendukung. Tujuan dari peralatan pendukung adalah untuk menunjukkan kepada pengguna bagaimana sistem informasi dapat berfungsi menggunakan model logis (sistem yang sedang berkembang) dan model fisik (sistem operasi).

1. *Data Flow Diagram* (DFD)

Menurut Hasbiyalloh dan Jakaria (2018) Sebuah model yang dikenal sebagai diagram aliran data (DFD) digunakan untuk menggambarkan sistem yang ada atau sistem yang akan dikembangkan tanpa memperhitungkan lingkungan fisik di mana data tersebut berada, mengalir, atau disimpan. Dapat dilihat pada **Tabel 2.1** dibawah merupakan berbagai simbol yang mewakilkan DFD :

Tabel 2. 1 Simbol DFD

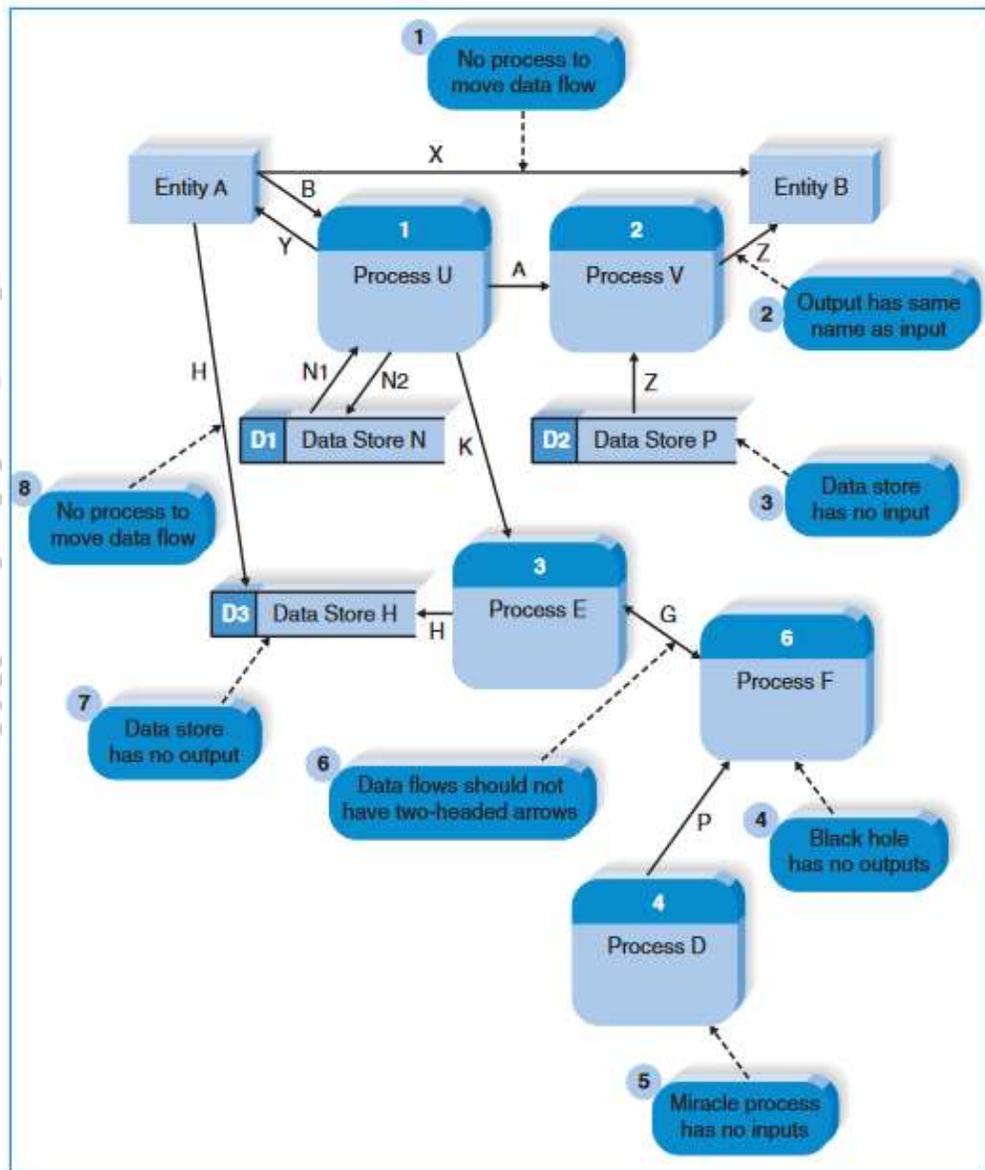
Simbol	Nama	Keterangan
	Entitas	Exernalentity/Enitasluar. Simbol ini menunjukkan orang, organisasi atau sistem yang berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.

	<p>Arus Data</p>	<p>Aliran data merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan(<i>input</i>) atau (<i>output</i>)</p>
	<p>Proses</p>	<p>Notasi proses atau fungsi atau prosedur, pada pemodelan perangkat lunak yang di implementasikan dengan pemograman terstruktur. Pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program.</p>
	<p>Simpan Data</p>	<p>File atau basis data atau <i>data store</i>, pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel – tabel basis data yang dibutuhkan sesuai dengan perancangan pada basis data (ERD).</p>

Sumber : (Sukamto, 2022)

DFD berguna sebagai alat untuk memverifikasi apakah sistem akan dibangun memenuhi kriteria diinginkan oleh pengguna atau belum. Data mengalir diagram dapat dikembangkan dari tingkat itu terendah ke tingkat yang lebih tinggi. DFD tingkat 0 adalah pengembangan dari diagram konteks, DFD level 1 adalah pengembangan dari DFD level 0. Setiap proses dari DFD bisa

dikembangkan lagi lebih rinci sampai proses - proses tidak dapat dikembangkan lagi (Sukamto, 2022). Dalam pembuatan *data flow diagram* ada indikator yang harus di pahami dan diterapkan dalam pembuatan *data flow diagram*, dapat dilihat pada **gambar 2.2** dibawah merupakan contoh pembuatan diagram DFD yang baik (Dennis, Wixom, & Roth, 2012).



Gambar 2. 2 Contoh Gambar DFD

Sumber : (Dennis, Wixom, & Roth, 2012)

Dilihat dari **gambar 2.2** diatas ada hal yang harus diperhatikan dalam menggambar atau merancang *data flow diagram* yaitu :

- 1) Data yang berpindah harus melalui suatu proses dan data yang keluar harus merupakan hasil dari suatu proses dengan hasil yang berbeda dari masukan.
- 2) *Data Store* harus memiliki masukan dan keluaran.
- 3) *Data flow* harus mengarah pada satu arah tidak dapat memiliki dua panah arah.
- 4) Suatu proses harus memiliki masukan dan keluaran.

Indikasi dari **gambar 2.2** diatas dan penjelasannya merupakan acuan dari menggambar atau merancang sebuah *data flow diagram* untuk memastikan sistem yang dibuat tidak mengalami kerusakan atau eror dan memastikan sistem berjalan dengan baik dan benar (Dennis, Wixom, & Roth, 2012).

2. Kamus Data

Kamus data atau *data dictionary* dan dapat juga disebut dengan istilah *system data dictionary* menurut Saleh (2020) suatu atau kumpulan daftar elemen yang terorganisir yang memiliki arti tetap dan sesuai dengan sistem, sehingga analisis sistem mempunyai pengertian yang sama dengan input, output dan kumpulan data *store*. Alur data pada DFD bersifat global yang hanya menunjukkan struktur dari alur data, untuk menunjukkan struktur dari alur data secara terinci maka terbentuk dan dibuatlah kamus data. Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur yang biasanya berisikan nama – nama dari data, digunakan pada proses yang terkait pada data, deskripsi data, dan informasi tambahan (tipe data, nilai data, batas nilai data) (Sukamto, 2022). Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut.

Tabel 2. 2 Simbol kamus data

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

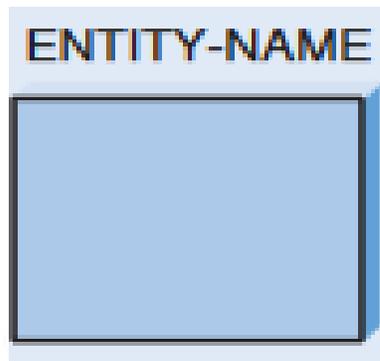
Sumber : (Sukamto, 2022)

3. Entity Relationship Diagram (ERD)

Entity Relationship Diagram atau biasa di sebut ERD merupakan pemodelan awal basis data yang akan dikembangkan. Menurut Hasbiyalloh & Jakaria (2018), ERD adalah diagram yang menggambarkan data dari suatu sistem informasi, didalam diagram tersebut menguraikan suatu susunan data yang disimpan pada sistem secara abstrak, seperti entitas, interaksi antar objek, dan atribut. Menurut Afifah, Azzahra, & Anggoro (2022), Terdapat 3 elemen dasar ERD, yaitu entitas, atribut, dan relasi.

1) Entitas

Entitas merupakan suatu atau banyak objek yang akan menjadi perhatian dalam basis data. Entitas sendiri dapat berupa manusia, tempat, benda, dan kondisi mengenai data yang dibutuhkan. Simbol dari entitas berbentuk persegi panjang.



Gambar 2. 3 Entitas Simbol

(Dennis, Wixom, & Roth, Systems Analysis And Design Fifth Edition, 2012)

2) Atribut

Atribut merupakan informasi yang terdapat dalam entitas. Sebuah entitas harus memiliki *primary key* sebagai acuan atau ciri khas entitas dan atribut deskriptif. Atribut biasanya terletak dalam tabel entitas atau dapat juga terpisah dari tabel. Simbol dari atribut berbentuk *elips*.



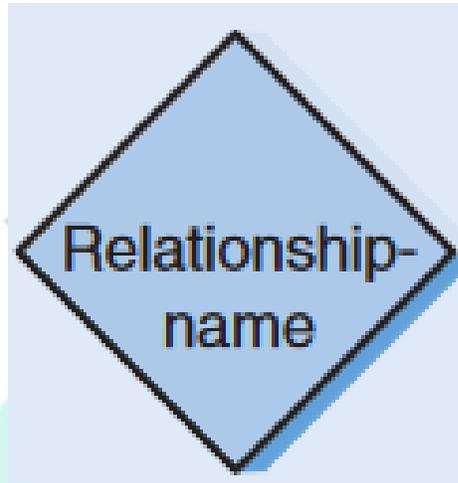
Gambar 2. 4 Atribut Simbol

Sumber : (Dennis, Wixom, & Roth, Systems Analysis And Design Fifth Edition, 2012)

3) Relasi

Menurut (Dennis, Wixom, & Roth, Systems Analysis And Design Fifth Edition, 2012) Relasi pada ERD adalah asosiasi antara entitas, dan mereka ditampilkan dengan garis yang menghubungkan entitas bersama. Setiap

hubungan memiliki entitas induk dan entitas anak, induk menjadi entitas pertama dalam relasi, dan entitas anak.



Gambar 2. 5 Relasi Simbol ERD

Sumber : (Dennis, Wixom, & Roth, Systems Analysis And Design Fifth Edition, 2012)

Cardinality merupakan suatu penggambaran jumlah entity yang dapat dihubungkan dengan entity lain melalui relasi. Berikut merupakan beberapa macam tingkatan hubungan *cardinality*, yaitu :

a. *One to One*

Menurut Dennis, Wixom, & Roth (2012), relasi 1:1 (dibaca sebagai “satu ke satu”) berarti bahwa satu instance entitas induk diasosiasikan dengan satu instance entitas turunan.

b. *One to Many*

Menurut Dennis, Wixom, & Roth (2012), hubungan adalah 1:N (dibaca sebagai “satu ke banyak”). Dalam hubungan semacam ini, satu instance dari entitas induk diasosiasikan dengan banyak instance dari entitas anak namun, instance entitas anak terkait hanya dengan satu instance dari induk..

c. *Many to Many*

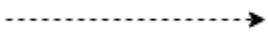
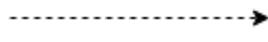
Menurut Dennis, Wixom, & Roth (2012), hubungan M:N (dibaca sebagai "banyak ke banyak"). Dalam hal ini, banyak instance entitas induk dapat berhubungan dengan banyak instance anak kesatuan.

4. *Usecase Diagram*

Diagram yang menggambarkan interaksi aktor dan use case yang diantara keduanya menjelaskan bagaimana pengguna atau aktor berinteraksi dengan sistem untuk memenuhi persyaratan fungsional. *Use Case* sendiri divisualisasikan sebagai elips *horizontal* dalam diagram *Use Case* yang mewakili suatu fitur atau proses yang akan dijalankan dalam sistem untuk memenuhi kebutuhan pengguna atau aktor (Dennis, Wixom, & Roth, *Systems Analysis And Design Fifth Edition*, 2012). Berikut adalah daftar simbol yang digunakan pada **Tabel 2.3** dibawah.

Tabel 2. 3 Simbol Usecase Diagram

Simbol	Nama	Deskripsi
	Use Case	Representasi dari fungsionalitas atau kebutuhan sistem yang dapat di implementasikan kedalam sistem
	Aktor	Representasi orang atau sistem yang berinteraksi dengan Use Case dan bertindak sebagai objek eksternal yang mengonsumsi data serta melakukan tindakan
	<i>Association</i>	Komunikasi dua arah untuk setiap Use Case yang diinisiasi oleh aktor utama
	<i>Generalization</i>	Hubungan antar aktor atau <i>usecase</i> yang dihubungkan melalui relasi generalisasi dan tidak memiliki keterkaitan dalam hal komunikasi antara keduanya

<pre><< include >></pre> 	<i>Include</i>	Hubungan include menggambarkan bahwa satu Use Case membutuhkan fungsionalitas dari Use Case lain untuk menyelesaikan tugasnya. Use Case yang disertakan tidak dapat berdiri sendiri dan harus menjadi bagian dari Use Case yang memasukkannya Relasi Use Case yang memperluas dari Use Case lain dalam situasi tertentu
<pre><< extend >></pre> 	<i>Extend</i>	Use Case yang memperluas dapat menambahkan langkah-langkah eksekusi tambahan ke Use Case yang diperluas

Sumber : (Dennis, Wixom, & Tegarden, 2015)

Contoh *usecase description* dapat mencakup deskripsi yang sepenuhnya lengkap, di mana semua langkah dari semua skenario use case yang mungkin dijelaskan secara detail. Satu use case bisa menghasilkan beberapa skenario yang berbeda yang mungkin terjadi selama pelaksanaan *usecase* oleh seorang aktor. Setiap skenario menyajikan alur peristiwa yang berbeda. Berikut adalah contoh penulisan format use case description (Laplante, 2017).

Tabel 2. 4 Contoh Spesifikasi Usecase

<i>Name</i>	Membaca Buku
<i>Actor</i>	Pembaca
<i>Description</i>	Proses membaca buku untuk mendapat pengetahuan atau hiburan
<i>Precondition</i>	<ul style="list-style-type: none"> - Pembaca memiliki buku yang ingin dibaca - Pembaca memiliki waktu yang cukup untuk membaca
<i>Basic Scenario</i>	<ol style="list-style-type: none"> 1. Pembaca memilih buku yang ingin dibaca 2. Pembaca membuka buku halaman pertama 3. Pembaca membaca setiap buku sesuai halaman 4. Pembaca memahami isi buku dan menyerap informasi yang disajikan 5. Pembaca melanjutkan membaca buku hingga halaman terakhir
<i>Alternative</i>	<ul style="list-style-type: none"> - Pembaca dapat membuat catatan atau menggarisbawahi bagian penting yang ditemukan dalam buku - Pembaca dapat berhenti sejenak untuk merenung atau memikirkan konsep yang baru dipelajari
<i>Failure</i>	<ul style="list-style-type: none"> - Jika pembaca tidak memiliki buku yang ingin dibaca, proses membaca tidak dapat dilakukan

<i>Failure Condition</i>	- Jika pembaca tidak menemukan buku yang diminati, proses membaca tidak dapat dilanjutkan
<i>Postcondition</i>	- Pembaca mendapatkan pengetahuan atau hiburan dari buku yang dibaca

Sumber : (Philip, 2017)

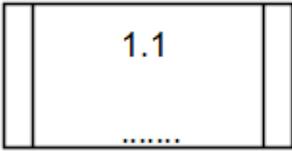
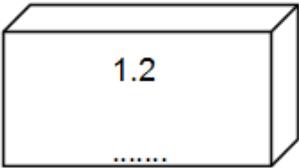
Setelah contoh yang dapat dilihat pada **Tabel 2.4** diatas, berikut merupakan pengertian menurut Philip (2017) dari contoh yang ada diatas.

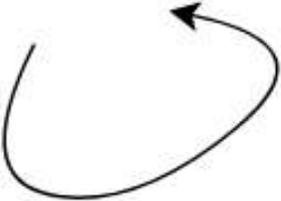
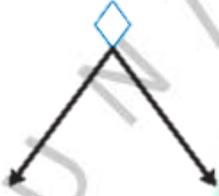
- *Name* : *Name* merupakan nama yang di ambil dari *usecase* yang digunakan untuk menamai spesifikasi *usecase* dengan proses dan nama yang sama.
- *Actor* : *Actor* mempresentasikan seseorang atau sesuatu seperti perangkat lain dan sistem, yang berinteraksi dengan sistem.
- *Description* : *Description* merupakan sebuah deskripsi singkat yang menjelaskan tentang spesifikasi *usecase*.
- *Precondition* : *Precondition* merupakan sebuah proses yang sudah dilakukan sebelum dapat menjalankan proses yang ingin dilakukan.
- *Basic Scenario* : *Basic Scenario* menjelaskan urutan utama tindakan dan interaksi di mana kasus penggunaan diselesaikan dengan benar dan mudah.
- *Alternative* : *Alternative* skenario menggunakan ekstensi kasus, dan menggambarannya sebagai cabang dari keberhasilan utama skenario.
- *Failure* : *Failure scenario Skenario* merupakan kegagalan menangani jalur melalui usecase yang mengarah pada kegagalan tujuan use case secara keseluruhan.
- *Post Condition* : *Post Condition* merupakan kasus berhasil dari proses yang dijalani.

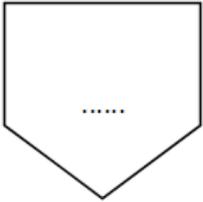
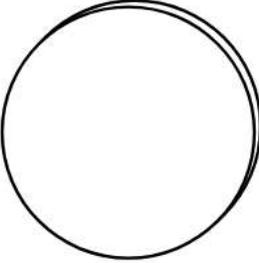
5. Structured chart

Bagan struktur adalah teknik penting yang membantu analis merancang program untuk sistem baru. Bagan struktur menunjukkan semua komponen kode itu harus dimasukkan dalam program, disusun dalam format hirarkis yang menyiratkan urutan (dalam urutan apa komponen dipanggil), pemilihan (di bawah apa kondisi modul dipanggil), dan iterasi (seberapa sering komponen diulang). Komponen dibaca dari atas ke bawah, kiri ke kanan, dan di berikan nomor dengan skema nomor hierarkis di mana *level* yang lebih rendah memiliki tambahan tingkat nasional penomoran (Dennis, Wixom, & Roth, 2012). Berikut merupakan simbol dan penjelasan dari *structured chart* yang dapat dilihat pada **tabel 2.5** dibawah.

Tabel 2. 5 Simbol Structured Chart

Simbol	Nama	Fungsi	Deskripsi
	<i>Module</i>	Menunjukkan bagian logis dari program yang diulang dalam diagram struktur	<ol style="list-style-type: none"> 1. Memiliki nomor. 2. Memiliki nama.
	<i>Library module</i>	Menunjukkan bagian logis dari program	<ol style="list-style-type: none"> 1. Memiliki nomor. 2. Memiliki nama. 3. Merupakan modul control jika dipanggil modul lain di bawahnya. 4. Merupakan modul bawahan jika dikendalikan oleh modul yang memiliki tingkatan lebih tinggi.

	<p><i>Loop</i></p>	<p>Mengkomunikasikan bahwa sebuah modul(s) diulang</p>	<ol style="list-style-type: none"> 1. Di gambarkan dengan anak panah yang melengkung. 2. Di tempatkan di sekitar garis dari satu atau lebih modul yang terdapat pengulangan.
	<p><i>conditional line</i></p>	<p>Mengkomunikasikan modul bawahan dipanggil oleh modul control berdasarkan beberapa kondisi</p>	<ol style="list-style-type: none"> 1. Digambar dengan berlian. 2. Mermasukn modul yang dipanggil berdasarkan beberapa kondisi.
	<p><i>Data couple</i></p>	<p>Mengkomunikasikan bahwa data sedang diteruskan dari satu modul ke modul lainnya</p>	<ol style="list-style-type: none"> 1. Memiliki garis panah dengan lingkaran. 2. Mempunyai nama data yang sedang dikirim. 3. bisa di arahkan ke atas dan bawah.
	<p><i>control couple</i></p>	<p>Mengkomunikasikan bahwa pesan atau <i>flag</i> sistem sedang diteruskan dari satu modul ke modul lainnya</p>	<ol style="list-style-type: none"> 1. Memiliki garis panah dengan bulatan. 2. Mempunyai nama pesan atau <i>flag</i> sistem yang diteruskan. 3. Hanya dapat mengarah ke atas

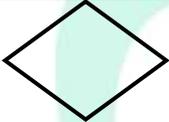
	<i>off-page connector</i>	Mengidentifikasi kapan bagian diagram dilanjutkan ke halaman lain dari bagan struktur	<ol style="list-style-type: none"> 1. Dilambangkan dengan segi enam 2. Memiliki nama 3. digunakan ketika diagram terlalu besar untuk memuat semuanya pada halaman yang sama
	<i>on-page connector</i>	Mengidentifikasi kapan bagian diagram dilanjutkan di tempat lain pada halaman yang sama dari bagan struktur	<ol style="list-style-type: none"> 1. Dilambangkan dengan lingkaran 2. Memiliki nama 3. Digunakan saat diagram terlalu besar untuk memuat semuanya di tempat yang sama pada halaman.

Sumber : (Dennis, Wixom, & Tegarden, 2015)

6. Flow Map

Menurut (Hasbiyalloh & Jakaria, 2018), *flow map* merupakan bagan-bagan yang mempunyai arus untuk penyelesaian dari suatu permasalahan yang biasanya digunakan untuk menggambarkan keseluruhan aliran dokumen, ini mendefinisikan hubungan antara bagian (aktor proses), proses (manual/bantuan komputer), dan aliran data (format dokumen keluaran dan masukan). Dapat dilihat dibawah pada **Tabel 2.6** merupakan beberapa simbol yang digunakan pada *flow map*.

Tabel 2. 6 Simbol flow map

Simbol	Keterangan
	Proses Komputer
	Terminal Point awal/akhir
	Flow, menunjukkan arah proses
	Keputusan dalam program

Sumber : (Bagir & Putro, 2018)

2.1.11 Black Box Testing

Menurut Setiyani (2019), Pengujian black box testing bertujuan untuk menemukan fungsi yang tidak benar, kesalahan antarmuka, kesalahan pada struktur data, kesalahan perfomansi, kesalahan inisialisasi dan terminasi. Black box testing merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak.

2.2. Tinjauan Studi

Tinjauan Studi dilakukan penulis untuk mendukung penulisan riset penelitian, tinjauan studi ini berkaitan dengan topik yang penulis bahas. Berikut adalah beberapa referensi:

1. Jurnal penelitian yang dilakukan oleh Risky Pratama Putra¹⁾ Ahmad Riyadi 2) Setia Wardani dengan judul “RANCANG BANGUN SISTEM INFORMASI PERSEWAAN DAN PENJUALAN PROPERTI DI DIY BERBASIS WEB”. Penelitian tersebut bertujuan dan membahas mengenai Sistem informasi persewaan dan penjualan properti di DIY berbasis web, dan membuat aplikasi yang dapat menampilkan gambaran detail properti kepada calon pembeli. Obyek pada penelitian ini adalah persewaan dan penjualan properti di Daerah Istimewa Yogyakarta. Pada penelitian tersebut juga memiliki latar belakang masalah yang serupa yaitu membahas kelemahan dari situs *marketplace* yang ada saat ini. Dengan kata lain situs - situs *property marketplace* tersebut tidak fokus pada bisnis *property*, Masih jarang memiliki fitur pemetaan lokasi properti yang dijual atau disewakan sehingga konsumen kesulitan dalam memperoleh gambaran letak geografis, dan jarang terdapat fitur untuk menghitung simulasi kredit penjualan rumah, sehingga konsumen tidak memiliki gambaran mengenai berapa angsuran perbulan bila properti yang dijual dikreditkan. Rancang bangun sistem informasi persewaan dan penjualan properti ini dibangun dengan bahasa pemrograman PHP dan database MySQL dengan Metode perancangan menggunakan diagram konteks, DAD Sistem, relasi antar tabel dan tampilan antar muka dengan sistem yang diusulkan. Tahap penyembangan aplikasi meliputi analisis, perancangan sistem, implementasi, dan pengujian. Pengujian sistem dilakukan dengan black box test dan alpha test.

2. Jurnal penelitian yang dilakukan oleh Siswidiyanto¹⁾ Ahmad munif²⁾ Diah Wijayanti³⁾ Eko Haryadi pada april tahun 2020 yang berjudul "Sistem Informasi Penyewaan Rumah Kontrakan Berbasis Web Dengan Menggunakan Metode Prototype" Jurnal tersebut membahas tentang Penyewaan rumah kontrakan membutuhkan sekali adanya suatu sistem informasi, karena proses penyewaan kontrakan saat ini masih menggunakan cara manual, yaitu dengan mencari kontrakan secara langsung dan pemilik kontrakan hanya memasang selebaran kertas di jalan terdekat dengan kontrakan yang dimiliki, dengan cara tersebut kurang efektif dan efisien. Selain itu jurnal tersebut melakukan analisa kebutuhan penyewaan property berbasis web, kesimpulan dari jurnal tersebut yaitu aplikasi web dapat mempermudah pengguna atau pengunjung dalam menyalurkan dan mendapatkan informasi seputar property. Kebutuhan dan bisnis properti akan terus tumbuh dengan seiring meningkatnya populasi disuatu wilayah tersebut. Dilansir dari situs Kontan.co.id dari survei yang dilakukan oleh Rumah.com bertajuk Consumer Sentiment Study di paruh kedua tahun ini, Mencatat ada 34% responden yang mencari hunian sendiri. Minat masyarakat untuk memiliki properti atau hunian tetap tinggi walau pandemi Covid-19 masih menghantui. Dalam survei tersebut juga terlihat, 25% responden berminat untuk melakukan merenovasi atau menggunakan kembali area tertentu di rumah mereka. Paulus Totok Lusida, Ketua Umum DPP REI menjelaskan, hasil survei ini menandai bahwa industri properti tetap diminati masyarakat saat pandemi terjadi. Dari berita tersebut diharapkan dengan adanya rancangan aplikasi ini dapat mempermudah masyarakat dalam mencari dan melakukan transaksi bisnis property.

3. Jurnal penelitian yang dilakukan oleh Adrian Marshall Lay1) Yaulie Deo Y. Rindengan2) Xaverius B. N. Najoan yang berjudul “Rancang Bangun Aplikasi Penyewaan Fasilitas di Kota Manado Berbasis Web” Jurnal tersebut membahas tentang Kota Manado yang merupakan Ibu kota Provinsi Sulawesi Utara, Kota ini memiliki 11 Kecamatan serta 87 Kelurahan dan desa. Manado terletak di Teluk Manado dan dikelilingi oleh daerah pegunungan. Memiliki berbagai macam fasilitas yang disewakan mulai dari Apartemen, Gedung Serbaguna, Studio dan lain-lain. Namun meski memiliki banyak fasilitas yang disewakan, Informasi yang disediakan untuk fasilitas-fasilitas tersebut masih sulit dicari dan diketahui oleh masyarakat karena kurangnya publikasi dan promosi tentang fasilitas yang disewakan tersebut. Selain itu jurnal ini juga bertujuan merancang aplikasi penyewaan berbasis web yang ada di Manado demi memudahkan masyarakat yang ada disana dalam penyewaan properti atau fasilitas. Metode pengumpulan data yang digunakan dalam penelitian ini menggunakan metode observasi, wawancara serta studi literatur. Dan untuk perancangan aplikasi menggunakan Unified Modeling Language (UML) . Tujuan penelitian ini adalah untuk membangun Aplikasi Penyewaan Fasilitas di Kota Manado Berbasis Web. Hasil yang diperoleh dari penelitian ini adalah Aplikasi penyewaan fasilitas di kota Manado berbasis web ini telah selesai dibuat dan diharapkan dapat mempermudah masyarakat khususnya di Kota Manado untuk mencari informasi tentang fasilitas dan untuk melakukan reservasi atau booking fasilitas berupa gedung, studio, ruangan dan sebagainya.

4. Jurnal penelitian yang dilakukan oleh Alfan Faizal Arsidin¹⁾ Taufiq Abidin²⁾ M. Muzaqi yang berjudul “RANCANG BANGUN APLIKASI PENYEWAAN DAN PENJUALAN RUKO BERBASIS WEBSITE” Jurnal tersebut membahas industri yang berkembang di Kota Tegal salah satunya adalah industri di bidang property khususnya di komoditi sewa dan jual ruko, dibuktikan dengan adanya bangunan-bangunan ruko baru yang ada di setiap sudut Kota Tegal, membuktikan bahwa perkembangan industri di bidang properti di komoditi sewa dan jual ruko banyak diminati. Akan tetapi perkembangan industri di Kota Tegal tidak dibarengi dengan adanya pemanfaatan teknologi, hal ini menyebabkan penyebaran informasi tidak tersampaikan kepada masyarakat secara luas. Sehingga masyarakat sulit untuk mencari ruko dan menentukan ruko yang mereka inginkan. Kesulitan yang dialami oleh masyarakat adalah kurangnya media informasi yang mudah dan efisien dalam mencari informasi tentang ruko, harga ruko, serta fasilitas dan spesifikasi yang ada pada ruko tersebut, sehingga masyarakat kesulitan dalam mencari informasi. Sistem ini merupakan aplikasi berbasis Website yang dibuat untuk membantu pemilik ruko untuk mempromosikan rukonya dan juga membantu masyarakat yang ingin mencari ruko. Aplikasi ini dibangun dengan menggunakan model waterfall dan dengan menggunakan Framework Laravel, dengan bahasa pemrograman PHP dan sebagai database dalam pembuatan aplikasi ini menggunakan MySql. Pada penelitian tersebut penulis berhasil membuat rancangan aplikasi penyewaan dan penjualan ruko berbasis web, selain itu penulis juga sudah menguji menggunakan *black box* menunjukkan aplikasi yang telah dibuat berjalan dengan fungsinya sebesar 85% dan 15% untuk sistem yang tidak berjalan sesuai fungsinya.

5. Jurnal penelitian yang dilakukan oleh Sundari Agustami¹⁾ Ratna Mutu Manikam²⁾ yang berjudul “PERANCANGAN APLIKASI SISTEM INFORMASI PENYEWAAN GEDUNG SERBAGUNA WILAYAH JAKARTA BARAT” Jurnal tersebut membahas membuat desain rancangan web usulan perancangan sistem informasi penyewaan gedung serbaguna wilayah Jakarta barat yang dapat memberikan informasi tentang gedung - gedung dan mengenai sewa menyewa gedung yang berada di Jakarta barat, Berhasil merancang desain usulan perancangan sistem informasi penyewaan gedung serbaguna wilayah Jakarta barat dapat dilakukan pemesanan melalui online dan membayar via bank dengan mudah dan cepat. Perancangan aplikasi ini dapat memberikan informasi yang detail untuk para penyewa gedung serbaguna ini secara online. Rancangan aplikasi sistem informasi penyewaan gedung dilakukan dengan metode SDLC dan permodelan *waterfall* dalam mulai perancangan hingga produk jadi. Pada penelitian penulis tersebut telah berhasil membuat desain rancangan web usulan perancangan sistem informasi penyewaan gedung serbaguna pada wilayah Jakarta barat yang dapat memberikan informasi mengenai gedung – gedung dan mengenai sewa menyewa gedung yang berada pada di lokasi tersebut. Penulis juga berhasil merancang desain usulan perancangan sistem informasi penyewaan gedung serbaguna wilayah Jakarta barat yang dapat di akses dan dipesan melalui *online* dan membayar via bank dengan mudah dan cepat. Penulis juga berhasil dalam merancang aplikasi sistem informasi penyewaan gedung yang detail untuk menginformasikan kepada para calon penyewa gedung serbaguna yang dapat di akses secara *online* tersebut.