

BAB II

STUDI PUSTAKA

2.1 Penelitian Relevan

Dalam melakukan penelitian ini menggunakan beberapa rujukan dari hasil penelitian sebelumnya yang berkaitan dengan informasi pengelolaan inventaris. Adapun penelitian terkait yang dimaksud adalah :

Penelitian yang dilakukan oleh (Oktaviani, Widiarta, & Nurlaily, 2019) yang berjudul “Sistem informasi Inventaris Barang Berbasis Web Pada SMP Negeri 1 Buer”. Dalam jurnal tersebut menjelaskan bahwa di SMP Negeri 1 Buer membutuhkan sebuah sistem untuk pengolahan inventaris sekolah, dikarenakan proses pencatatan data barang, peminjaman serta pengembalian masih bersifat konvensional yaitu dengan mencatat ke dalam buku sehingga membutuhkan waktu yang cukup lama dan kurang efektif serta sering terjadinya duplikat data. Maka diperlukan sebuah sistem berbasis web untuk membantu dan mempermudah petugas dalam pengolahan inventaris agar data terorganisir dengan baik dan lebih teratur. Penelitian ini menggunakan metode pengumpulan data dengan cara observasi, wawancara, dan studi pustaka dan metode pengembangan sistem yang digunakan adalah metode *Waterfall* menggunakan bahasa pemrograman PHP dan *database* MySQL. Hasil dari penelitian ini adalah sebuah sistem informasi inventaris berbasis web yang dapat membantu petugas dalam pengelolaan data inventaris serta memudahkan transaksi peminjaman barang di sekolah tersebut.

Penelitian yang dilakukan oleh (Mondev, Asparizal, & Adrianto, 2017) yang berjudul “Komputerisasi Barang Inventaris Pada Kantor Lingkungan Hidup Kota Dumai”. Permasalahan yang terdapat pada penelitian tersebut yaitu di Kantor Lingkungan Hidup (KLH) kota Dumai proses pengelolaan barang inventaris dilakukan secara terkomputerisasi dengan menginputkan data ke dalam *Microsoft Excel*, setelah data diinput maka petugas akan mencetak *hardcopy*nya dan menyimpannya di dalam rak arsip. Hal ini menyebabkan petugas kesulitan dalam melakukan pencarian data yang dibutuhkan dikarenakan harus mengecek rak arsip satu per satu, maka dilakukan penelitian untuk membangun sistem inventaris berbasis database yang dapat memudahkan petugas dalam proses membuat

salinan data serta mempersingkat waktu saat pembuatan laporan inventaris. Dalam penelitian ini menggunakan pengembangan sistem dengan model *Waterfall*, sedangkan untuk studi permasalahan, wawancara, dan observasi sebagai metode pengumpulan datanya. Penelitian ini menghasilkan sebuah sistem inventaris berbasis database menggunakan *Visual Basic.Net* sebagai bahasa pemrograman dan database MySQL.

Penelitian ini dilakukan oleh (Khalim, Asnawi, Hidayat, & Mardiyantoro, 2020) yang berjudul “Aplikasi Inventaris Barang Berbasis Web Pada Laboratorium Komputer Fastikom”. Menyatakan bahwa di Laboratorium Fastikom proses inventarisasi barang masih dilakukan dengan menggunakan *Microsoft Excel* sebagai pengolah data sehingga rentan terhadap kesalahan dan kehilangan data. Selain itu sering terjadi keterlambatan saat proses pembuatan laporan inventaris. Teknik pengumpulan data dilakukan dengan cara observasi, wawancara, studi pustaka, dan dokumentasi. Sedangkan, metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode *Waterfal* menggunakan bahasa pemrograman PHP dan MySQL untuk basis data. Penelitian ini menghasilkan sebuah sistem inventaris laboratorium (Simaslab) yang mempermudah petugas dalam monitoring barang yang masuk dan keluar serta mempercepat dalam membuat laporan karena ada fitur cetak laporan sesuai waktu yang dipilih.

Perbedaan dari ketiga penelitian diatas terletak pada penggunaan bahasa pemrograman. Pada penelitian pertama dan ketiga menggunakan bahasa pemrograman PHP dan untuk penelitian kedua menggunakan bahasa pemrograman Visual Basic .Net. sedangkan pada penelitian ini menggunakan bahasa pemrograman Python. Dari beberapa penelitian yang pernah dilakukan sebelumnya, penelitian ini memiliki beberapa keunggulan yaitu framework yang digunakan adalah *Framework Django*. Dalam aplikasi ini terdapat fitur seperti cetak QR Code untuk masing-masing barang yang telah diinput pada sistem. Setiap barang akan memiliki QR Code dengan pola yang berbeda yang berfungsi sebagai identitas dari barang tersebut. Oleh karena itu, sistem ini diharapkan dapat memudahkan petugas dalam proses sinkronisasi data inventaris yang tersedia serta mempermudah proses pencarian barang di kemudian hari.

2.2 Landasan teori

2.2.1 Inventaris

Inventarisasi merupakan serangkaian proses pencatatan keluar masuk barang dan menyusunnya secara tepat sesuai dengan peraturan yang ada. Inventarisasi dilakukan untuk memudahkan dalam melakukan pengawasan dan pengendalian terhadap barang yang dimiliki oleh suatu sekolah. (Khambali & Siswanto, 2018).

Inventaris berperan penting dalam pengelolaan data barang serta seluruh pekerjaan yang berkaitan dengan manajemen inventaris di suatu perusahaan atau organisasi. Kegiatan inventaris meliputi pendataan pengadaan barang, peletakan barang, pemindahan serta pemeliharaan barang. (Susandi & Sukisno, 2018).

Berdasarkan penjelasan yang telah dipaparkan di atas dapat disimpulkan bahwa inventaris merupakan kegiatan pencatatan atau pendataan barang-barang yang bertujuan untuk mempermudah dalam penyusunan serta pengawasan barang yang tersedia di suatu organisasi.

2.2.2 Framework Django

Django merupakan salah satu web *framework* dengan bahasa pemrograman Python yang digunakan untuk mengembangkan website yang kaya fitur, aman digunakan dan bersifat dinamis. Django dikelola oleh Django *Software Foundation* dan menjadi pilihan utama bagi seorang pengembang untuk melakukan pengembangan sistem/aplikasi karena fitur yang disediakan cukup lengkap. Dengan adanya *framework* dapat memudahkan pengembang untuk membuat aplikasi agar terstruktur dengan baik serta lebih cepat. (Saputra & Aji, 2018).

Framework Django menggunakan pola MVT, yaitu Model, View, dan Template. Django memiliki banyak fitur salah satunya pada keamanan. Fitur yang terdapat dalam Django ini dapat menjaga agar tidak ada proses eksekusi program pada lapisan Template. Beberapa perusahaan yang menggunakan *framework* Django, seperti Instagram, Quora, dan Disqus. (Fajri, Hambali, & Rahayu, 2019).

Menurut (Hourieh, 2008) Django memiliki kelebihan dibandingkan dengan *framework* lainnya yaitu sebagai berikut :

- a. *Object-Relational Mapping (ORM)* sebagai komponen database django yang menghubungkan antara model data dan *database engine*.
- b. *Automatic Administration Interface*, Django menyediakan desain *interface* yang siap dipakai oleh pengguna.
- c. Bersifat *Multi-Lingual Support*, karena Django mendukung lebih dari satu bahasa situs web dengan sistem built-in internasionalisasi.
- d. *Advanced Development Environment*, Django menyediakan web server yang ringan untuk memudahkan dalam pengembangan keamanan aplikasi dan pengujian.
- e. *Clean URL Design*, dimana sistem URL dalam Django bersifat fleksibel dan powerful. Dalam Django mendeksripsikan model untuk URL dalam sistem serta menetapkan tugas dari Python dalam menangani setiap model.
- f. *Tight Integration between Components*, Django memiliki komponen yang terintegrasi.

Berdasarkan hasil paparan di atas dapat disimpulkan bahwa django merupakan *web framework* yang mendukung pola MVT (Model, View, Template) menggunakan bahasa pemrograman Python untuk membuat aplikasi web yang dinamis, dimana django memiliki banyak fitur yang dibutuhkan untuk membuat sebuah aplikasi.

2.2.3 Python

Python merupakan salah satu bahasa pemrograman tingkat tinggi yang dapat dipahami dengan mudah serta termasuk ke dalam bahasa pemrograman yang mempunyai source code yang mudah dimengerti dan sangat lengkap. Python mendukung hampir semua sistem operasi, seperti Windows, Macintosh, dan Linux. Bahasa pemrograman Python bersifat *open source* dan *free* serta dapat digabungkan ke dalam bahasa pemrograman lainnya seperti *Visual Basic Net*.

Python memiliki beberapa kelebihan seperti sintaks yang mudah diingat dan dipahami serta memiliki dukungan library yang menyediakan beberapa modul-modul dan ekstensi program yang siap digunakan. Kekurangan dari bahasa pemrograman Python yaitu keterbatasan dalam mengakses basis data, kurang cocok untuk pengembangan platform mobile serta eksekusi program yang cukup lambat. (Wardana, 2019).

Berdasarkan hasil paparan di atas dapat disimpulkan bahwa Python merupakan bahasa pemrograman tingkat tinggi yang bersifat *open source*, sehingga siapa saja dapat mengembangkan sebuah program menggunakan bahasa pemrograman Python. Python dapat berjalan di berbagai platform sistem operasi serta struktur *syntax* yang mudah dipahami bagi pemula.

2.2.4 Visual Studio Code

Visual Studio Code merupakan salah satu teks editor ringan buatan *Microsoft* yang digunakan untuk berbagai sistem operasi mulai dari Linux, Windows, dan Mac. Bahasa pemrograman Node.js, JavaScript, dan TypeScript yang didukung secara langsung oleh *Visual Studio Code*. Teks editor ini juga mendukung bahasa pemrograman lainnya seperti C, C++, Python, Java, dan sebagainya yang dipasang di marketplace *Visual Studio Code*.

Beberapa fitur yang tersedia dalam teks editor ini yaitu *Intellisense*, *Debugging*, *Git Integration* serta fitur ekstensi yang akan terus bertambah seiring perkembangan *Visual Studio Code*. Teks editor ini bersifat *open source* karena kode sumbernya mampu diketahui di link *GitHub* dan pengembang dapat ikut andil dalam proses pengembangan *Visual Studio Code*. (Habibi & Suryansah, 2020).

Menurut (Setyawan & Prawiro, 2020) *Visual Studio Code* memiliki beberapa kelebihan sebagai berikut :

- a. Teks editor gratis.
- b. Bersifat multiplatform, karena dapat diakses di semua sistem operasi mulai dari Windows, Linux, dan MacOS.
- c. Terintegrasi dengan git.
- d. Memudahkan pekerjaan pengembang dalam pembuatan kode karena menyediakan banyak ekstensi.

Berdasarkan penjelasan di atas dapat disimpulkan bahwa *Visual Studio Code* merupakan *editor source code* buatan *Microsoft* yang bersifat *open source* dan mendukung *Debugging*, *Git Integration* serta fitur ekstensi lainnya.

2.2.5 Website

Website merupakan sekumpulan informasi dalam bentuk halaman web yang menyediakan berbagai macam informasi berbentuk teks, gambar, video, audio ataupun kombinasi dari semuanya dan dapat diakses melalui internet di seluruh dunia. (Saputra & Aji, 2018).

Menurut (Abdulloh, 2018) secara umum website terbagi menjadi tiga jenis, yaitu sebagai berikut.

1. Website statis merupakan jenis website yang tidak dapat diubah isinya melainkan harus mengubah script atau syntax yang ada di dalamnya. Jenis website ini biasanya hanya digunakan oleh *website company profile* untuk menampilkan informasi penting dari suatu perusahaan atau organisasi.
2. Website dinamis merupakan jenis website dimana informasi yang terdapat di dalamnya dapat diubah oleh pemilik website langsung dari websitenya. Jenis website ini banyak digunakan dalam aktifitas bisnis yang berkaitan dengan internet, seperti website berita dan web blog.
3. Website interaktif merupakan pengembangan dari website dinamis, dimana jenis website ini dapat berkomunikasi dari dua arah antara pemilik website dengan pengguna website itu sendiri. Di dalam website ini pengguna dapat berinteraksi, berkomentar atau mengajukan topik mengenai apa yang ada di dalam website tersebut, contoh dari jenis website ini yaitu blog, forum serta website *marketplace* seperti tokopedia, bukalapak dan lainnya.

Berdasarkan paparan di atas dapat disimpulkan bahwa website merupakan halaman web yang memuat berbagai informasi dalam bentuk teks, gambar, video atau audio yang dapat diakses oleh pengguna melalui internet. Website dibedakan menjadi 2 yaitu website statis dan website dinamis.

2.2.6 QR Code

Menurut M.P. Nugraha dalam jurnal (Wijaya & Gunawan, 2016) *QR Code* (*Quick Response Code*) adalah kode matriks dua dimensi yang mempunyai kegunaan dalam menyimpan banyak data di dalamnya. *Barcode* (kode batang) mengalami perkembangan yang disebut dengan istilah QR Code.

Menurut (Prathivi, 2018) dalam pembuatan sebuah QR Code memiliki

beberapa cara yang wajib dikerjakan, karena QR Code mempunyai banyak macam jenis inputan. Hal yang harus diperhatikan yaitu inputan yang dikonversi ke dalam bentuk biner agar dapat diubah menjadi titik hitam dan putih. QR Code mempunyai beberapa kelebihan daripada jenis-jenis barcode yang lain, yaitu :

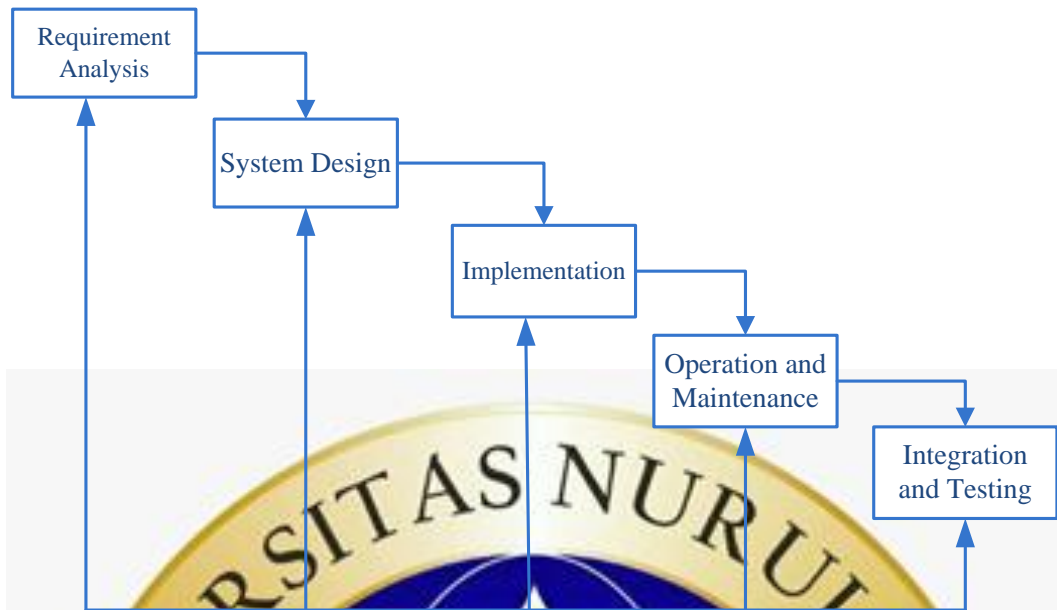
- a. Memiliki ukuran penyimpanan yang besar.
- b. Dapat dipahami dengan mudah.
- c. Ukuran yang kecil.
- d. Kemampuan dalam menyimpan data berupa huruf dan angka.
- e. Mampu dibaca dari segala arah.
- f. Tidak mudah rusak.

Berdasarkan hasil paparan di atas dapat disimpulkan bahwa QR Code merupakan kode matriks dua dimensi yang di dalamnya mampu menyimpan berbagai jenis informasi. Dalam QR Code terdapat beberapa komponen pada kotak kode, dimana tiga kotak besar di setiap sudut merupakan pembatas kode.

2.2.7 Model Pengembangan Sistem SDLC

SDLC (*System Development Life Cycle*) merupakan siklus hidup yang digunakan dalam pengembangan sistem informasi yang bertujuan untuk menyelesaikan suatu permasalahan secara efektif. SDLC memiliki banyak jenis model pengembangan sistem diantaranya seperti model *waterfall*, *prototype*, *Rapid Application Development (RAD)*, *iterative*, spiral dan lain sebagainya. (Hidayat, 2020). Implementasi dalam pengembangan sistem ini menggunakan model *waterfall*.

Waterfall merupakan salah satu model dari SDLC (*System Development Life Cycle*), dimana dalam pembuatan sistemnya dilakukan secara sistematis dan setiap tahap di dalamnya harus dikerjakan terlebih dahulu sebelum mengerjakan ke tahap selanjutnya. Dalam pengerjaannya harus dilakukan secara berurutan dimulai dari tahap pertama yaitu *requirement* atau analisis kebutuhan, *system design* atau desain sistem, *implementation* atau penulisan kode, *integration & testing* atau pengujian, dan *operation & maintenance* atau pemeliharaan. (Yurindra, 2017).



Gambar 2. 1 Tahapan Waterfall (Yurindra, 2017)

Menurut (Yurindra, 2017) Metode waterfall memiliki beberapa tahapan yaitu sebagai berikut :

1. Requirement Analysis

Tahap pertama akan dilakukan analisa terhadap kebutuhan perangkat dan kebutuhan data dari informasi yang didapat dari hasil wawancara, observasi dan studi literatur. Tujuannya ialah untuk mengetahui kebutuhan pengguna dalam sistem yang akan dibangun.

2. System Design

Tahap berikutnya adalah tahap pembuatan dari desain sistem yang akan dibuat secara keseluruhan. Tahapan ini memberikan gambaran apa yang harus dilakukan oleh pengembang untuk membuat suatu sistem serta dapat membantu dalam penentuan perangkat keras yang dibutuhkan.

3. Implementation

Pada tahap ini akan dilakukan pengembangan dengan melakukan pemecahan modul-modul menjadi beberapa bagian dan nantinya akan disatukan di langkah selanjutnya. Selain itu, akan dilakukan pengamatan pada sistem tersebut, apakah kebutuhan sistem sudah terpenuhi atau tidak.

4. Integration & Testing

Pada tahap ini, unit-unit yang telah dikembangkan di tahapan sebelumnya akan digabungkan kemudian akan diuji untuk mencegah terjadinya kesalahan pada sistem yang telah dibuat.

5. Operation & Maintenance

Tahapan model *waterfall* yang terakhir adalah pemeliharaan. Sistem yang telah selesai akan dilakukan pemeliharaan tujuannya yaitu untuk memastikan apakah sistem mampu berjalan dengan lancar serta memperbaiki kesalahan-kesalahan yang tidak terdeteksi di tahapan sebelumnya.

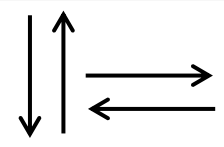

2.2.8 Flowchart

Menurut Bambang Hartono dalam jurnal (Dewi & Malfiany, 2018) *Flowchart* merupakan gambaran grafik dari tahapan dan proses suatu program serta hubungan dengan satu proses dan proses yang lain. Penggunaan *flowchart* bertujuan menyelesaikan suatu masalah secara sederhana dengan menggunakan simbol-simbol tertentu yang dapat dipahami oleh pengembang dimana tahapan penyelesaian masalah harus disajikan secara tepat, sederhana, dan jelas.


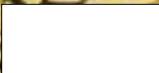




Menurut (Ranatarisza & Noor, 2013) ada beberapa jenis *Flowchart*, di antaranya :

1. Bagan Alir Sistem (*system flowchart*) yaitu bagan yang memperlihatkan alur proses suatu sistem secara keseluruhan.
2. Bagan Alir Dokumen (*document flowchart*) atau bagan alur formulir adalah grafik alur yang menggambarkan aliran dari laporan dan formulir termasuk salinannya.

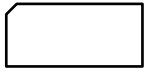

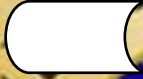


Tabel 2. 1. Simbol-simbol *flowchart*

No.	Simbol	Nama Simbol	Keterangan
1.		<i>Flow</i>	Simbol yang menunjukkan arah aliran dari proses.
2.		<i>Communication Link</i>	Simbol yang menunjukkan perpindahan data dari satu proses ke proses lainnya.

Tabel 2. 2. Simbol-simbol *flowchart* (Lanjutan)

No	Simbol	Nama Simbol	Keterangan
3.		<i>Connector</i>	Sebagai penghubung dari satu proses ke proses yang lain dari lembar yang sama.
4.		<i>Offline Connector</i>	Sebagai penghubung dari satu proses ke proses yang lain dari lembar yang berbeda.
5.		<i>Process</i>	Simbol yang menyatakan pengolahan yang dilakukan oleh komputer.
6.		<i>Manual</i>	Simbol yang menjelaskan pengolahan yang tidak dikerjakan oleh komputer.
7.		<i>Decision</i>	Digunakan sebagai pemilihan proses berdasarkan kondisi yang ada.
8.		<i>Predefined Process</i>	Digunakan sebagai tempat penyimpanan suatu proses.
9.		<i>Terminal</i>	Menyatakan sebagai suatu proses dari awal sampai akhir.
10.		<i>Input/Output</i>	Menjelaskan input dan output suatu proses.

Tabel 2. 3. Simbol-simbol *flowchart* (Lanjutan)



No	Simbol	Nama Simbol	Keterangan
11		<i>Punched Card</i>	Menjelaskan input yang berdasarkan kartu / output di tulis ke kartu.
12		<i>Magnetic-tape</i>	Menjelaskan input berdasarkan kartu / output yang berasal dari pita magnetis.
13		<i>Disk Storage</i>	Menyatakan input berasal dari disk/output di simpan ke disk.
14		<i>Document</i>	Menyatakan sebagai input/output yang berasal dari dokumen.
15		<i>Display</i>	Menyatakan sebagai hasil output melalui layar monitor.

Sumber : (Jalinus & Ambiyar, 2016)

2.2.9 Data Flow Diagram

Data Flow Diagram adalah suatu diagram arus data yang menggunakan simbol-simbol untuk menyediakan entitas, proses, arus data dan penyimpanan dari suatu sistem yang bertujuan untuk menjelaskan arus data serta perubahan yang diimplementasikan pada saat data berpindah dari masukan (input) menjadi keluaran (output). (Kurniawan, 2020). Arti simbol *Data Flow Diagram* (DFD) dijelaskan pada tabel berikut :

Tabel 2. 4. Simbol *Data Flow Diagram* (DFD)

No.	Simbol		Nama	Keterangan
	DeMarco and Yourdan	Gane and Sarson		
1.			<i>Source</i> (Kesatuan Luar)	Menjelaskan asal atau tujuan data di luar sistem.
2.			Proses	Menjelaskan proses dimana arus data masuk ditransformasikan ke arus data keluar
3.			<i>Data Flow</i> (Arus Data)	Menjelaskan proses arus data.
4.			<i>Data Store</i> (Penyimpanan Data)	Menggambarkan tempat data disimpan.

Sumber : (Indrajani, 2015)

Menurut (Mukhtar, 2018) dalam pembuatan data flow diagram terdapat beberapa aturan yang harus diperhatikan, yaitu :

- Tidak diperbolehkan untuk menghubungkan antara entity luar dengan entity luar yang lain secara langsung.
- Tidak diperbolehkan untuk menghubungkan data store yang satu dengan yang lainnya secara langsung.
- Tidak diperbolehkan untuk menghubungkan data store dengan entity luar secara langsung.
- Tiap proses harus memiliki arus data yang masuk dan arus data yang keluar.
- Tidak diperbolehkan ada proses dan arus data tidak memiliki nama, hal ini

menyebabkan arus data yang tidak mempunya hubungan bersatu.

- f. Tidak diperbolehkan ada proses yang tidak mempunyai nomor.

2.2.10 Entity Relationship Diagram

Entity Relationship Diagram merupakan suatu model yang menjelaskan susunan data yang tersimpan dari sistem secara abstrak. ERD memiliki tiga komponen utama, yaitu entitas (*entity*), atribut (*attribute*), dan relasi atau hubungan (*Relationship*). Pemodelan sistem ERD menjelaskan relasi antara entitas di dalam sistem. Tiap entitas memiliki karakteristik tersendiri, dimana salah satunya digunakan sebagai kunci utama (*primary key*) atau kunci asing (*foreign key*). (Saputra R. , 2015). Berikut merupakan arti dari simbol-simbol pada *Entity Relationship Diagram* sebagai berikut :

Tabel 2. 5. Simbol ERD (*Entity Relationship Diagram*)

No.	Simbol	Nama	Keterangan
1.		Entitas	Mendeksripsikan sebagai himpunan entitas.
2.		Relasi	Mendeksripsikan sebagai himpunan atau relasi.
3.		Atribut	Mendeksripsikan sebagai atribut yang digunakan sebagai key.
4.		Garis	Menghubungkan antara relasi dan entitas atau relasi dan entitas dengan atribut.

Sumber : (Rohayati & Irwandi, 2016)

Menurut (Muslihudin, Fauzi, & Abadi, 2021) kardinalitas atau derajat relasi merupakan relasi yang menyatakan jumlah maksimum entitas yang dapat berhubungan dengan entitas pada himpunan entitas yang lain. Macam-macam kardinalitas atau derajat relasi adalah sebagai berikut :

- a. Relasi satu ke satu (*one to one*), yaitu setiap himpunan entitas A hanya dapat dihubungkan dengan satu himpunan entitas B dan himpunan entitas B dapat dihubungkan dengan satu himpunan entitas A.
- b. Relasi satu ke banyak (*one to many*), yaitu setiap himpunan entitas A dapat dihubungkan dengan lebih dari satu himpunan entitas B tetapi tidak sebaliknya.
- c. Relasi banyak ke banyak (*many to many*), yaitu setiap himpunan entitas A dapat dihubungkan dengan banyak entitas himpunan entitas B dan juga sebaliknya.

2.2.11 Black Box Testing

Black Box Testing yaitu pengujian desain dan kode program, dimana pengujian yang dilakukan berfokus pada spesifikasi fungsional dari program tersebut. Pada pengujian *black box testing* hanya memeriksa tampilan luar saja serta fungsionalnya, dan tidak melihat apa yang terjadi pada proses di dalamnya. (Jan, Shah, Johar, Shah, & Khan, 2016)

Menurut (Yani, Setiawan, Egi, Subagja, & Desyani, 2020) *blackbox testing* memiliki beberapa kelebihan dan kekurangan yaitu :

1. Kelebihan
 - a. Penguji tidak perlu memiliki pengetahuan bahasa pemrograman yang digunakan.
 - b. Antara pengembang dan penguji saling berkaitan satu sama lain.
 - c. Pengujian dilakukan menggunakan sudut pandang pengguna saat melakukan pengujian sistem.
2. Kekurangan
 - a. Sering terjadi pengulangan pengujian yang dilakukan oleh penguji.
 - b. Spesifikasi sistem yang tidak jelas.

Berdasarkan pemaparan yang telah dijelaskan di atas dapat disimpulkan bahwa *black box testing* merupakan pengujian program terhadap fungsionalitas dari sistem yang sedang dikembangkan tanpa melihat struktur internal kode di dalamnya.