

BAB 2 LANDASAN TEORI

1.8 Penelitian Terdahulu

Dalam penyusunan, penulis mengambil referensi dari beberapa penelitian terdahulu untuk menggali informasi tentang teori yang berkaitan dengan judul yang digunakan. Adapun penelitian sebelumnya yang menjadi tinjauan pustaka dalam penulisan ini adalah sebagai berikut:

Penelitian yang dilakukan oleh (Revaldi Fuad Azhar, Novrini Hasti) dengan judul Sistem Informasi Studio Foto Pada AN Studio Berbasis Website. Penelitian ini membahas mengenai pembangunan sistem informasi berbasis website yang dapat menghitung secara otomatis data yang telah diinputkan, menyimpan histori laporan, dan mengecek ketersediaan tempat serta waktu yang dibutuhkan juga dapat memberikan reminder dan push notification. Hasil dari penelitian ini adalah dapat membantu SPV dalam melakukan monitoring karyawan dan mempermudah pemesanan dalam pembuatan laporan penjualan harian ketersediaan tempat serta waktu yang dibutuhkan.

Penelitian yang dilakukan oleh (Muhammad Tabrani, Suhardi, Hananda Priyandaru, 2021) dengan judul Sistem Informasi Manajemen Berbasis Website Pada UNL Studio Dengan Menggunakan Framework Codeigniter. Fotografi saat ini tidak lagi hanya menjadi seni, tetapi juga menjadi hobi yang dapat menghasilkan uang atau sebagai pekerjaan profesional. Kebutuhan akan fotografer profesional pun meningkat guna mengabadikan peristiwa penting yang dialami seperti momen pernikahan, wisuda. Melihat hal ini, penulis tertarik untuk membuat sebuah situs atau website dinamis yang terhubung ke basis data, menggunakan framework CodeIgniter berbasis bahasa PHP. Dalam website ini nantinya pengunjung dapat menyewa studio foto juga melakukan konfirmasi

pembayaran dan melihat jadwal sewa studio, sehingga pengunjung tidak perlu menakukan panggilan telepon untuk menanyakan secara langsung pada admin.

Penelitian yang dilakukan oleh (Muhammad, Al Hamid Afiza, 2023) dengan judul Pembangunan Sistem Informasi Pengelolaan Transaksi Berbasis Web Pada Mutiara Studio Photo Padang. Teknologi informasi sudah menjadi sarana utama dalam membantu manusia melakukan aktivitas sehari-hari termasuk usaha yang bergerak dibidang fotografi. Salah satunya adalah Mutiara Studio Photo. Saat ini proses pemesanan dan pengelolaan di Mutiara Studio Photo masih menggunakan cara konvensional. Semua pemesanan dicatat dalam sebuah buku secara manual sehingga membutuhkan waktu yang lama untuk rekapan transaksi dan menyebabkan besarnya peluang terjadi kesalahan pencatatan transaksi atau hilangnya bukti fisik transaksi. Cara pemesanan pun dilakukan dengan cara datang langsung ke studio foto atau pesan secara online melalui whatsapp bussines. Cara ini juga kurang efektif untuk merekap pesanan yang masuk karena masih dicatat secara manual dan belum ada sistem yang dapat menyimpan informasinya secara terstruktur. Untuk itu perlu adanya perancangan dan pembangunan sistem informasi untuk mempermudah pengelolaan transaksi dan pemesanan jasa foto. Metode pengembangan sistem yang digunakan adalah metode waterfall. Sistem ini dikembangkan berbasis website dan dibangun menggunakan bahasa pemrograman PHP dengan framework laravel serta database MYSQL. Pengujian sistem dilakukan dengan metode black box testing untuk menguji kesesuaian fungsional aplikasi dengan analisis sistem yang dirancang. Hasil yang diharapkan dari penelitian ini adalah dapat memudahkan user dalam mencatat setiap transaksi dan pengelolaan laporan penjualan, serta mempermudah proses pemesanan jasa foto di Mutiara Studio Photo.

Penelitian yang digunakan oleh (Maulida, Friska, 2018) dengan judul Sistem Informasi Pengelolaan Data



Transaksi Produk Dan Jasa Di Studio Foto Ciamis. Sistem informasi penjualan pada Cloud Studio Photo Ciamis berbasis web untuk memproses sistem penjualan yang terjadi, mengolah beberapa laporan ,memberikan informasi terkini mengenai studio photo dan produk yang disediakan. Jenis usaha ini adalah usaha yang bergerak di bidang media rekam, menerima jasa dibidang fotografi dan video recording. Sistem yang dikerjakan masih secara manual akan menghambat proses transaksi karena lamban dalam pencatatan serta penghitungan dan informasi yang kurang tepat ketika diadakannya pembukuan transaksi, maka perlu adanya sistem informasi pengelolaan data yang terkomputerisasi sebagai alat bantu dalam mengolah data. Dengan adanya sistem informasi penjualan produk dan jasa pada cloud studio photo ciamis diharapkan dapat mempermudah dan mempercepat proses pengelolaan data transaksi dan informasi, sistem ini dapat menghasilkan data dan informasi hasil kegiatan penjualan yang terjadi serta menghasilkan laporan pertanggung jawaban dengan lebih cepat dan praktis. Pembuatan sistem menggunakan metode Extreme Programming dengan metode pemogramannya menggunakan skala prioritas untuk mengatasi permasalahan dalam pengolahan data pemesanan transaksi jasa. Metode tersebut untuk menentukan dan membandingkan seberapa besar dan seberapa pentingnya hal tersebut yang harus dilakukan. Penerapan metode skala prioritas menjadi solusi bagi permasalahan pada pemesanan atau pembokingan jasa, dengan melakukan proses skala prioritas ini tidak terjadi pembentrokan jadwal. Sistem yang dibangun mudah dipelajari oleh pengguna(user) dan bahasa pemograman yang digunakan adalah PHP dan MySQL untuk membuat basis data.

Penelitian yang dilakukan oleh (Joko Heru Atmoko, 2011) dengan judul Pembuatan Sistem Informasi Penjualan Pada Studio Foto Puji Photo Di Kalibening Banjarnegara. Studio Foto Puji Photo adalah sebuah perusahaan yang berlokasi di Kalibening, Banjarnegara dan bergerak di bidang



Seni Media Rekam yaitu Fotografi dan Video Rekam yang juga menjual aksesori. Kondisi penggunaan sistem penjualan pada Studio Foto Puji Photo masih menggunakan sistem manual dan belum menggunakan teknologi yang lebih berkompetensi dan berkomputerisasi, dimana persaingan dagang selalu muncul di dunia usaha. Untuk itu penulis mengajukan pada Studio Foto Puji Photo perlu dibangunnya sebuah software sistem informasi penjualan berbasis data yang relevan dan akurat dalam mempermudah pengolahan data transaksi penjualan yang akan menghasilkan informasi laporan penjualan perusahaan secara benar serta dapat menghemat waktu dan tenaga dalam penyusunannya. Adapun beberapa penelitian-penelitian sebelumnya dengan masalah penelitian yang sedang diteliti terdapat pada tabel berikut:





Tabel 2. 1 Penelitian Terdahulu

N O	Nama Peneliti	Judul	Persamaan	Perbedaan
1	Evaldi Fuad Har, Novrini (2011)	Sistem Informasi Studio Foto Pada AN Studio Berbasis Website	Bertujuan untuk melakukan pengolahan data, dan memberikan informasi kepada pelanggan serta dapat melakukan proses transaksi dengan efektif.	Menggunakan Metode SDLC (Software development Life Cycle) Penyimpanan data menggunakan <i>firebase</i>
2	Muhammad Orani, Hardi, Nanda Wandaru, (2011)	Sistem Informasi Manajemen Berbasis Website Pada UNL Studio Dengan Menggunakan Framework Codeigniter	Bertujuan untuk memudahkan manager dalam menangani pemesanan tau booking studio foto,	Perancangan berbasis web



NO	Nama Peneliti	Judul	Persamaan	Perbedaan
3	Muhammad, Hamid (2023)	Pembangunan Sistem Informasi Pengelolaan Transaksi Berbasis Web Pada Mutiara Studio Photo Padang	Memiliki tujuan untuk merancang dan membangun sistem informasi untuk mempermudah pengelolaan transaksi dan pemesanan jasa foto.	Menggunakan metode waterfall. Sistem dibangun menggunakan bahasa pemrograman PHP dengan framework laravel, database MYSQL
4	Aulida, (2018)	Sistem Informasi Pengelolaan Data Transaksi Produk Dan Jasa Di Studio Photo Ciamis	Memiliki tujuan untuk mempermudah dan mempercepat proses pengelolaan data transaksi dan informasi	Menggunakan PHP dan MySQL untuk membuat basis data.
5	Heru (2011)	Pembuatan Sistem Informasi Penjualan Pada Studio Foto Puji Photo Di	Bertujuan untuk mempermudah pengolahan data transaksi secara benar serta dapat menghemat	Pembuatan berbasis web



NO	Nama Peneliti	Judul	Persamaan	Perbedaan
		Kalibening Banjarnegara	waktu dan tenaga dalam penyusunannya.	

Dengan beberapa penelitian terdahulu yang disebutkan diatas bahwa yang membedakan peneltian ini dengan penelitian sebelumnya adalah penambahan fitur transaksi (payment gateway) yang memudahkan pelanggan bisa melakukan pembayaran secara online sehingga pelanggan tidak perlu ke lokasi untuk melakukan transaksi.

1.9 Kajian Pustaka

1.9.1 Sistem Informasi Manajemen

Menurut (Gita Rosalit) menyebutkan bahwa Sistem Informasi Manajemen (SIM) adalah sebuah sistem yang dirancang untuk menyediakan informasi yang dibutuhkan untuk mendukung operasi, manajemen, dan pengambilan keputusan dalam suatu organisasi. Sistem ini mengintegrasikan berbagai komponen seperti perangkat keras, perangkat lunak, prosedur, manusia, dan data untuk menghasilkan informasi yang berguna dan relevan bagi manajer dan pihak terkait dalam menjalankan fungsi-fungsi manajemen mereka.

Sistem Informasi Manajemen memungkinkan organisasi untuk mengumpulkan, menyimpan, memproses, dan menyebarkan informasi dengan lebih efisien dan efektif. Tujuannya adalah untuk meningkatkan produktivitas, efisiensi operasional, dan kemampuan pengambilan keputusan dengan menyediakan data yang akurat dan terkini.

Fungsi utama dari Sistem Informasi Manajemen meliputi:

1. Pengumpulan Data: Mengumpulkan data dari berbagai sumber internal dan eksternal.
2. Pengolahan Data: Mengubah data mentah menjadi informasi yang bermakna.
3. Penyimpanan Data: Menyimpan data dan informasi untuk penggunaan di masa mendatang.



4. Distribusi Informasi: Menyampaikan informasi kepada pihak yang membutuhkan dalam format yang tepat waktu dan mudah dipahami.
5. Pendukung Keputusan: Memberikan informasi yang dibutuhkan untuk mendukung proses pengambilan keputusan di berbagai level manajemen.

Dengan demikian, Sistem Informasi Manajemen adalah alat yang sangat penting bagi organisasi dalam upaya meningkatkan kinerja dan mencapai tujuan strategisnya.

1.9.2 Metode Waterfall

Menurut **Roger S. Pressman**: Dalam bukunya "Software Engineering: A Practitioner's Approach," Pressman menggambarkan metode Waterfall sebagai pendekatan sistematis dan berurutan yang cocok untuk proyek-proyek di mana persyaratan sistem dapat sepenuhnya dipahami sebelum pengembangan dimulai. Pressman juga mencatat bahwa meskipun model Waterfall memiliki kelemahan dalam menangani perubahan kebutuhan, model ini masih memiliki tempat dalam proyek-proyek yang membutuhkan pendekatan yang terstruktur dan terdokumentasi dengan baik.

Tahapan dalam Metode Waterfall:

1. **Kebutuhan Sistem dan Perangkat Lunak (System and Software Requirements):**
 - Identifikasi kebutuhan sistem dan perangkat lunak secara lengkap dan terperinci.
2. **Analisis Sistem dan Perangkat Lunak (System and Software Design):**
 - Mendesain arsitektur sistem dan perangkat lunak berdasarkan kebutuhan yang telah diidentifikasi.



3. **Implementasi (Implementation):**
 - Mengubah desain sistem menjadi kode program yang dapat dieksekusi.
4. **Pengujian (Testing):**
 - Menguji perangkat lunak untuk memastikan bahwa semua fungsi bekerja sesuai dengan yang diharapkan dan tidak ada bug.
5. **Penerapan (Deployment):**
 - Menerapkan perangkat lunak ke lingkungan produksi dan memulai penggunaan oleh pengguna akhir.
 -
6. **Pemeliharaan (Maintenance):**
 - Memelihara perangkat lunak dengan memperbaiki bug yang ditemukan setelah penerapan, serta melakukan peningkatan dan pembaruan sesuai kebutuhan.

Metode Waterfall memiliki keuntungan berupa pendekatan yang terstruktur dan dokumentasi yang lengkap, namun juga memiliki kelemahan seperti kurang fleksibel terhadap perubahan kebutuhan selama proses pengembangan

Kelebihan menggunakan metode waterfall adalah memungkinkan untuk melakukan kontrol. Proses pengembangan model fase one by one, sehingga mengurangi kemungkinan kesalahan yang akan terjadi (Binus, 2019).

1.9.3 Unified Modeling Language (UML)





UML adalah bahasa pemodelan yang digunakan untuk menspesifikasikan, memvisualisasikan, membuat, dan mendokumentasikan artefak sistem perangkat lunak baik yang sedang dirancang ataupun dikembangkan (Riri Fitri Sari dan Ardiati, 2021 : 107). Ada 4 pemodelan UML yang terdiri dari:



1. Use Case Diagram

Use Case Diagram yang memberikan gambaran himpunan *use-case* serta aktor yang sesuai dengan jenis kelas. Diagram tersebut sangatlah berguna dalam pengorganisasian dan juga pemodelan perilaku pada sistem. Notasi *use case diagram* dijabarkan pada tabel berikut:



Tabel 2. 2 *Use Case Diagram*

Simbol	Notasi	Keterangan
	Aktor	Yang berinteraksi langsung dengan sistem
	Use Case	Fungsionalitas yang dimiliki sistem
	Association	Relasi yang terjadi antara aktor dan use case
	Generalisasi	Struktur pewaris

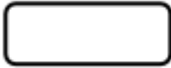

2. Activity Diagram

Diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem.

Tabel 2. 3 *Activity Diagram*

Simbol	Keterangan
	Simbol titik awal
	Simbol titik akhir



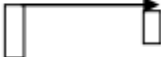




Simbol	Keterangan
	Simbol pekerjaan Activity yang dilakukan dalam aliran kerja
	Generalisasi untuk memperhatikan struktur pewaris yang terjadi

3. Sequence Diagram

Menggambarkan objek yang saling berhubungan dan menyampaikan kode/pesan tertentu. Apa yang ada dalam use case akan digambarkan lebih detail melalui pemodelan ini.

Tabel 2. 4 Sequence Diagram

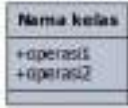




Simbol	Keterangan
	Aktor, sebagai pengguna sistem yang berinteraksi dengan sistem
	Lifeline, objek antar muka yang saling berinteraksi.
	Message Entry, menggambarkan pesan antar objek yang menunjukkan urutan kejadian yang terjadi.
	Message to self, menggambarkan pesan objek itu sendiri, yang menunjukkan urutan kejadian yang terjadi
	Self Message menggambarkan sebuah aktifitas pesan pada aktor.

4. Class Diagram

Hubungan antara class dan penjelasan detail dari setiap class yang ada pada desain suatu sistem,

serta menampilkan aturan-aturan dan tanggung jawab setiap entitas data.

Tabel 2. 5 Class Diagram

Simbol	Keterangan
	Terdiri dari nama, atribut, <i>method</i>
	Hubungan umum antara 2 kelas
	Relasi kelas tempat dia bergantung
	Sebuah kelas menggunakan kelas yang lain
	Relasi <i>Generalization</i> pada sebuah relasi <i>inheritance</i>

1.9.4 Framework Laravel

Menurut (Naista, 2017) Laravel merupakan salah satu framework berbasis PHP bersifat open source (terbuka), dan menggunakan konsep MVC (model - view - controller). Laravel juga menjadi salah satu framework yang dapat membantu developer untuk memaksimalkan penggunaan PHP didalam proses pengembangan website. Selain itu, Laravel juga memiliki beberapa fitur unggulan, seperti template engine, routing, dan modularity. Berikut ini merupakan keuntungan menggunakan *framework* menurut Daqiqil (2011:1) sebagai berikut :

1. Menghemat Waktu Pengembangan

Dengan struktur *library* yang telah disediakan oleh *framework* maka tidak perlu lagi memikirkan hal-hal tersebut.

2. *Refuse of Code*

Dengan menggunakan *framework* maka pekerjaan kita akan memiliki struktur yang baku, sehingga kita dapat menggunakan kembali dddi proyek-proyek lainnya.

3. Bantuan Komunitas

Ada komunitas-komunitas yang siap membantu jika ada permasalahan.

4. Kumpulan *Best Practice*

Sebuah *framework* merupakan kumpulan *best practice* yang sudah teruji. Jadi kita dapat meningkatkan kualitas kode kita.

1.9.5 PHP

Menurut (David Sklar), PHP adalah bahasa skrip yang kuat untuk pengembangan web, dan dia memberikan pandangan yang sangat positif terhadap PHP dalam bukunya "Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5" yang ditulis bersama Adam Trachtenberg.

2.1.1 Pandangan (David Sklar) tentang PHP:

1. **Kemudahan Pembelajaran:** Sklar menekankan bahwa PHP adalah bahasa yang mudah dipelajari bagi pemula dalam pengembangan web. Dia menulis buku tersebut dengan tujuan untuk memudahkan orang-orang untuk memulai dengan PHP, MySQL, dan teknologi web lainnya.
2. **Fleksibilitas:** PHP diketahui karena kemampuannya untuk bekerja dengan berbagai teknologi web, termasuk database seperti MySQL, dan frontend dengan HTML, CSS, dan JavaScript.
3. **Penggunaan Luas:** Sklar mengakui bahwa PHP digunakan secara luas di seluruh dunia untuk mengembangkan aplikasi web dari yang sederhana

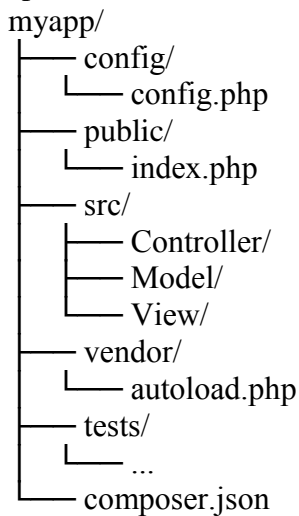


hingga yang kompleks. Ini termasuk situs web, aplikasi e-commerce, sistem manajemen konten (CMS), dan banyak lagi.

4. **Komunitas dan Dukungan:** Dia juga mengapresiasi komunitas PHP yang besar dan aktif yang menyediakan dukungan melalui forum, dokumentasi, tutorial, dan berbagai sumber daya lainnya.
5. **Perkembangan Terus-menerus:** David Sklar mencatat bahwa PHP terus berkembang, dengan versi terbaru (pada saat itu) menawarkan fitur-fitur baru dan peningkatan performa.

2.1.2 Contoh Struktur Proyek PHP

Berikut adalah contoh struktur proyek sederhana untuk aplikasi web PHP:



1.9.6 DBMS

DBMS (Database Management System) adalah suatu sistem yang dirancang khusus untuk mengelola suatu database dan menjalankan operasi terhadap data yang diminta oleh banyak pengguna.



Tujuan utama pengguna DBMS dalam jaringan komputer adalah untuk menghindari kekacauan dalam hal pengelolaan data yang jumlahnya besar.

Ada dua jenis bahasa komputer yang dapat digunakan dalam berinteraksi dengan DBMS yaitu MySQL merupakan sebuah perangkat lunak sistem manajemen basis data SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian basis data. Secara umum SQL terdiri dari dua bahasa yaitu *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML) . DDL digunakan untuk mendefinisikan, mengubah, serta menghapus basis data dan objek-objek yang diperlukan dalam basis data. DML digunakan untuk memanipulasi data yang ada dalam suatu tabel (Piyanti & Iriani, 2013).

1.9.7 Desain User Interface

Menurut Wilbert O. Galitz, *user interface* adalah bagian dari komputer dan perangkat lunak yang dapat dilihat, didengar, disentuh, diajak bicara, dan yang dapat dimengerti secara langsung oleh manusia.

User Interface yang baik harus memenuhi syarat sebagai berikut (Arifin et al., 2018) :

1. The Structure Principle
Desain User Interface harus disusun sesuai kegunaan berdasarkan model atau komponen yang dikenali oleh pengguna.
2. The Simplicity Principle
Desain User Interface harus dibuat secara sederhana dan seinformatif mungkin.
3. The Visibility Principle
Desain User Interface yang baik adalah desain yang tidak membebani pengguna dengan opsi yang tidak diperlukan.
4. The Feedback Principle



Desain User Interface harus memastikan pengguna mendapatkan notifikasi terhadap perubahan kondisi, aksi, ataupun kesalahan yang dilakukan pengguna.

5. The Tolerance Principle

Desain User Interface harus dirancang sefleksibel mungkin sehingga meminimalisir kemungkinan kesalahan yang dibuat oleh pengguna.

6. The Reuse Principle

Komponen desain user interface harus mampu digunakan ulang dengan tepat sesuai tujuan dan bukan hanya sekadar penggunaan ulang.

7. Conserve Attention

Desain User Interface yang baik adalah desain yang mampu menarik dan mempertahankan perhatian pengguna.

8. Simple Language Over Technical Terms

Pemilihan bahasa yang digunakan dalam Desain User Interface lebih baik jika menggunakan bahasa yang umum digunakan. Hindari menggunakan bahasa teknik karena pengguna bukan pengembang sistem.

1.9.8 Desain Basis Data

Menurut Michael Hernandez, seorang pakar desain basis data, desain basis data harus dilihat sebagai suatu proses yang komprehensif dan sistematis untuk menghasilkan basis data yang efisien, fleksibel, dan mudah dipahami. Berikut adalah beberapa prinsip dan pandangan utama Michael Hernandez tentang desain basis data:



1.9.9 Prinsip-prinsip Desain Basis Data Menurut Michael Hernandez:

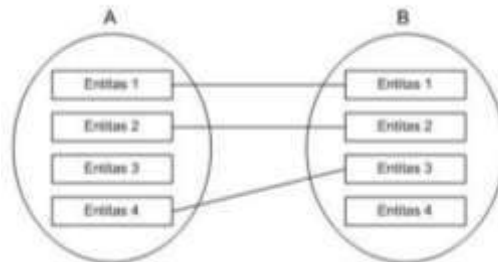
1. **Memahami Kebutuhan Bisnis:** Sebelum memulai desain basis data, penting untuk memahami kebutuhan bisnis yang spesifik dan memastikan bahwa desain tersebut memenuhi kebutuhan tersebut dengan baik.
2. **Normalisasi:** Hernandez menekankan pentingnya normalisasi dalam desain basis data. Normalisasi adalah proses untuk mengorganisir data dalam tabel relasional sehingga redundansi data diminimalkan dan integritas referensial terjaga. Hal ini membantu memastikan bahwa basis data lebih efisien dan konsisten.
3. **Hubungan Tabel yang Baik:** Desain yang baik mempertimbangkan hubungan antara tabel-tabel yang berbeda. Ini termasuk pemilihan kunci primer dan kunci asing yang tepat, serta memastikan bahwa relasi antar tabel didokumentasikan dengan baik.
4. **Efisiensi Kinerja:** Desain harus mempertimbangkan efisiensi kinerja, termasuk pengindeksan yang tepat dan pertimbangan atas ukuran dan volume data yang diharapkan.
5. **Fleksibilitas:** Basis data harus dirancang dengan mempertimbangkan kemungkinan perluasan di masa depan dan perubahan kebutuhan.
6. **Dokumentasi yang Baik:** Setiap aspek desain harus didokumentasikan dengan baik, termasuk skema basis data, relasi antar tabel, dan aturan integritas data.
7. **Kesederhanaan:** Desain harus sederhana dan mudah dipahami. Ini membantu dalam memelihara dan mengelola basis data secara efektif.

Berdasarkan di atas, dapat disimpulkan bahwa basis data merupakan data yang dapat didesain dan berintegrasi sehingga dapat memenuhi kebutuhan user dalam perusahaan atau organisasi.



1. Relasi one to one

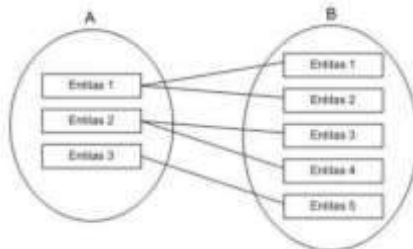
relasi one to one adalah relasi dimana setiap entitas pada himpunan A hanya berhubungan dengan satu entitas himpunan B, dan begitupun sebaliknya setiap entitas pada himpunan B hanya berhubungan dengan satu entitas himpunan A.



Gambar 2. 1 Relasi one to one

2. Relasi one to many

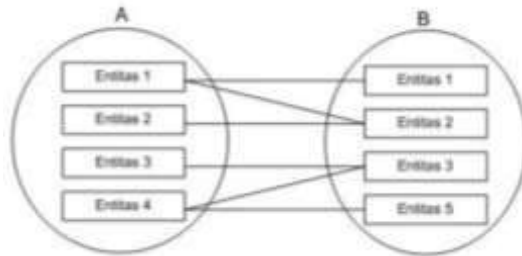
Relasi one to many adalah relasi dimana suatu entitas di himpunan A memiliki relasi di beberapa entitas di himpunan B.



Gambar 2. 2 Relasi one to many

3. Relasi many to many

Relasi many to many adalah relasi dimana setiap entitas pada himpunan A dapat berhubungan dengan banyak entitas himpunan B, dan juga sebaliknya dimana setiap entitas pada himpunan B dapat berhubungan dengan banyak entitas pada himpunan A.



Gambar 2. 3 Relasi many to many

2.2.9 Black Box

"Black box" adalah istilah yang sering digunakan dalam berbagai disiplin ilmu, seperti ilmu komputer, teknologi informasi, dan teknik. Berikut adalah beberapa penjelasan mengenai "black box" dalam konteks yang berbeda:

1. Ilmu Komputer dan Pemrograman:

- Black Box Testing: Dalam pengujian perangkat lunak, black box testing adalah metode pengujian di mana penguji tidak mengetahui struktur internal atau kode dari sistem yang diuji. Penguji hanya melihat output berdasarkan input yang diberikan dan memastikan bahwa fungsi perangkat lunak berjalan sesuai dengan spesifikasi yang diharapkan. Fokusnya adalah pada fungsionalitas perangkat lunak daripada cara kerjanya.

2. Kecerdasan Buatan (AI) dan Pembelajaran Mesin:

- Model Black Box: Dalam konteks AI dan pembelajaran mesin, model black box mengacu pada sistem atau algoritma yang menghasilkan output dari input tanpa memberikan pemahaman yang jelas tentang bagaimana keputusan tersebut dibuat. Misalnya, jaringan saraf dalam (deep neural networks) sering dianggap sebagai black box karena kompleksitas dan kesulitan dalam menginterpretasikan bagaimana mereka mencapai kesimpulan atau prediksi.

3. Elektronika dan Teknik:

- Black Box Systems: Dalam elektronika, black box menggambarkan sebuah sistem atau komponen yang fungsinya diketahui dan digunakan tanpa memerlukan pemahaman tentang mekanisme internalnya. Misalnya, sebuah modul elektronik yang digunakan dalam suatu rangkaian tanpa mengetahui detail komponen dalam modul tersebut.

4. Penerbangan:

- Black Box Recorder: Dalam konteks penerbangan, black box merujuk pada perekam penerbangan (flight recorder) yang menyimpan data penerbangan dan rekaman suara kokpit. Alat ini digunakan untuk menyelidiki kecelakaan atau insiden penerbangan dengan merekam data penerbangan dan percakapan di kokpit.

Secara umum, istilah "black box" digunakan untuk menggambarkan suatu sistem atau komponen yang penggunaannya tidak memerlukan pemahaman tentang cara kerjanya secara internal. Fokus utama adalah pada input dan output dari sistem tersebut.

