

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Penelitian Terdahulu**

Berdasarkan penelitian yang dilakukan oleh Mas Ayoe Elhias Nst (2018) mengenai Perancangan Kualitas Pelayanan Jasa Transportasi Pengiriman Barang Menggunakan Metode *Service Quality* Pada CV.Karya Agung Medan, Mas Ayoe Elhias Nst menyimpulkan bahwa dimensi kualitas pelayanan Jasa Transportasi pengangkutan barang meliputi Kehandalan, jaminan, daya tanggap, bukti langsung, dan empati. Nilai Dependensi merupakan harapan yang diinginkan perusahaan terhadap nilai pelayanan yang diberikan kepada konsumen.

Berdasarkan penelitian yang dilakukan oleh Jazuli, dkk (2020) mengenai Analisis Kualitas Pelayanan Dengan SERVQUAL Dan Importance Performance Analysis Di PT. XYZ, Jazuli, dkk menyimpulkan bahwa Layanan Wifi Id belum dapat memenuhi harapan pengguna, terbukti dari adanya gap atau selisih antara kenyataan dan harapan yang bernilai negatif pada seluruh dimensi.

Berdasarkan penelitian yang dilakukan oleh Shofa, dkk (2019) mengenai Aplikasi Pengukuran Tingkat Kepuasan Mahasiswa Terhadap Sarana Dan Prasarana Dengan Metode *Service Quality* Di Laboratorium Informatika Teori Dan Pemrograman Dasar Universitas Siliwangi, Shofa, dkk menyimpulkan bahwa berdasarkan perhitungan indeks tingkat kepuasan pelanggan pengunjung cafe menunjukkan nilai 84,17% dikatakan bahwa pelanggan merasa sangat puas akan pelayanan yang diberikan cafe XYZ di Surabaya. Berdasarkan hasil pengukuran

kualitas jasa yang dilakukan terdapat 1 atribut yang bernilai negatif (-) yaitu ketersediaan biji kopi yang lengkap, yang menunjukkan masih adanya kesenjangan antara harapan pelanggan dengan kondisi nyata yang ada pada cafe XYZ di Surabaya, sehingga 1 atribut ini membutuhkan adanya suatu perbaikan kualitas pelayanan. diperoleh usulan perbaikan untuk masalah tersebut yaitu: 10 (Preliminary Anti-Action) yaitu: sebaiknya cafe melakukan pengendalian persediaan biji kopi yang lengkap agar ketersediaan jumlah maupun bermacam1macam biji kopi di cafe XYZ selalu ada saat pelanggan membutuhkannya.

Berdasarkan penelitian yang dilakukan oleh Prananda, dkk (2019) mengenai Penerapan Metode *Service Quality* (SERVQUAL) Untuk Peningkatan Kualitas Pelayanan Pelanggan, Prananda, dkk disimpulkan bahwa nilai kualitas pelayanan (Q) Hotel Vidi 2 semuanya masih di bawah 1, namun dalam hal ini dikatakan cukup baik. Tetapi masih perlu dilakukan usaha-usaha peningkatan yang lebih terarah dari pihak Hotel Vidi 2 supaya kualitas pelayanannya semakin meningkat. Dimensi bukti fisik (tangible) merupakan dimensi yang memiliki nilai kualitas terkecil yaitu sebesar 0,806, penilaian tersebut. Nilai kualitas terbesar terdapat pada dimensi jaminan (Assurance) yaitu sebesar 0,964. Hal ini dapat dilihat dari sifat dapat dipercaya yang dimiliki para pegawai, bebas dari bahaya, resiko, atau keraguan.

Berdasarkan penelitian yang dilakukan oleh Nagara, dkk (2020) mengenai Analisis Peningkatan Kualitas Pelayanan Dengan Metode SERVQUAL Dan TRIZ (Studi Kasus Pada Cafe XYZ), Nagara, dkk menyimpulkan bahwa dalam penelitian

ini telah dilakukan penerapan metode *service quality* dalam rancang bangun kuesioner online untuk mengetahui tingkat kepuasan mahasiswa terhadap sarana dan prasarana di Laboratorium Informatika Teori dan Pemrograman Dasar (INTRO).

Berdasarkan penelitian yang dilakukan oleh Suliantoro dan Aisy (2020) mengenai Analisis Kepuasan Pelanggan Wifi ID Corner Dengan Metode Service Quality Dan Quality Function Deployment Pada Wilayah Usaha Telekomunikasi PT. Telekomunikasi Indonesia, Tbk. Tangerang, Suliantoro dan Aisy menyimpulkan bahwa kualitas pelayanan yang diberikan PT. XYZ Gresik berdasarkan kelima dimensi *service quality* diperoleh nilai persepsi sebesar 4,50 dan nilai harapan sebesar 3,91 dengan nilai gap positif (0,59). Dari nilai tersebut dapat dilihat bahwa nilai persepsi melampaui nilai harapan sehingga pelayanan yang diberikan sebagai kualitas ideal.

## **II.2. Landasan Teori**

### **II.2.1. Penerapan**

Penerapan merupakan suatu proses penerapan ide, konsep, kebijakan atau inovasi dalam suatu tindakan praktis sehingga memberikan dampak baik berupa perubahan pengetahuan, keterampilan maupun nilai dan sikap. Pengertian implementasi secara sederhana implementasi diartikan sebagai pelaksanaan atau penerapan. Implementasi merupakan aktivitas yang saling menyesuaikan. Implementasi merupakan sistem rekayasa. Pengertian-pengertian tersebut memperlihatkan bahwa kata implementasi bermuara pada aktivitas, adanya aksi,

tindakan atau mekanisme suatu sistem. Ungkapan mekanisme mengandung arti bahwa implementasi bukan sekedar aktivitas tetapi suatu kegiatan yang terencana dan dilakukan secara sungguh-sungguh berdasarkan acuan norma tertentu untuk mencapai tujuan kegiatan. (Magdalena, dkk, 2021 : 120).

### **II.2.2. Service Quality (ServQual)**

*Service Quality* (ServQual) merupakan metode pengendalian kualitas yang dikembangkan oleh Parasuraman, Zeithaml, dan Berry pada tahun 1990. ServQual dibuat karena adanya dua faktor utama mengenai kualitas dari jasa, yaitu persepsi yang dimiliki pelanggan, dan layanan yang diterima pelanggan (*perceived service*) dari layanan yang diharapkan pelanggan untuk mereka terima (*expected service*). ServQual sebagai penilaian konsumen mengenai keseluruhan layanan yang baik, atau dapat diartikan sebagai seberapa jauh perbedaan antara kenyataan dan harapan pelanggan atas pelayanan yang mereka terima/peroleh. (Suliantoro dan Aisy, 2020 : 3).

Metode ServQual adalah metode yang sering digunakan untuk mengukur kualitas pelayanan. Pengukuran mutu untuk produk fisik tidak sama dengan industri jasa. Analisis diawali dengan sebuah kuisioner yang disebarkan kepada konsumen, setiap item pertanyaan memiliki dua jawaban dalam skala (*Likert*), yaitu apakah menurut nasabah hal tersebut penting untuk dilakukan atau dilaksanakan dan bagaimana kinerjanya, baik atau tidak baik. Misalnya contoh pertanyaan sebagai berikut :

Karyawan ramah terhadap tamu hotel vidi :

Apakah menurut anda : Tidak penting (1→5) Penting.

Bagaimana kinerjanya : Tidak penting (1→5) Baik. (Prananda, dkk, 2019 : 2).

Model kualitas jasa yang paling populer dan hingga kini banyak dijadikan acuan dalam riset manajemen dan pemasaran jasa adalah model Service Quality yang dikembangkan oleh Zeithaml, dkk dalam serangkaian penelitian mereka terhadap sektor jasa. Model yang dikenal pula dengan istilah Gap Analysis Model ini berkaitan erat dengan model kepuasan pelanggan yang didasarkan pada ancangan diskonfirmasi. Metode ini mengukur kualitas secara kuantitatif dalam bentuk kuisisioner yang mengandung dimensi-dimensi kualitas jasa, yaitu *Tangibles*, *Reliability*, *Responsiveness*, *Assurance*, *Emphaty*. *Service Quality* dibangun atas adanya perbandingan dua faktor utama, yaitu persepi pelanggan atas layanan yang nyata pelanggan terima (*perceived service*) dengan layanan yang sesungguhnya diharapkan (*expected service*). *Perceived Service* dan *Expected Service* ini digunakan dalam konsep servqual untuk menghitung *gap* yang ada. Berikut adalah persamaannya :

$$G = P - E \dots\dots\dots (1)$$

Keterangan :

G = *Gap* (Kesenjangan)

P = *Perceived Service* (persepsi layanan)

E = *Expected Service* (harapan layanan). (Nagara, dkk, 2020 : 79).

Lima gap kualitas layanan yang berpotensi menjadi sumber masalah kualitas layanan, yaitu :

1. *Gap* antara harapan pelanggan dan persepsi manajemen (*knowledge gap*).

2. *Gap* antara persepsi manajemen terhadap konsumen dan spesifikasi kualitas jasa (*standards gap*).
  3. *Gap* antara spesifikasi kualitas jasa dan penyampaian jasa (*delivery gap*).
  4. *Gap* antara penyampaian jasa dan komunikasi eksternal (*communication gap*).
  5. *Gap* antara jasa yang dipersepsikan dan jasa yang diharapkan (*service gap*).
- (Suliantoro dan Aisy, 2020 : 3).

### **II.2.3. Kepuasan**

Kepuasan adalah perasaan senang atau kecewa yang muncul setelah membandingkan kinerja (hasil) produk yang dipikirkan terhadap kinerja (atau hasil) yang diharapkan. Sikap itu sifatnya sangat kompleks karena setiap individu memiliki ribuan sikap yang berbeda dengan individu lainnya. Karyawan terhadap aspek-aspek yang menyenangkan atau tidak menyenangkan mengenai pekerjaan yang sesuai dengan penilaian masing-masing pekerja. (Ramdani, dkk, 2019 : 26).

### **II.2.4. Kepuasan Pelanggan**

Kepuasan pelanggan adalah perasaan yang timbul sebagai hasil evaluasi terhadap pengalaman pemakaian produk atau jasa, Kepuasan Konsumen adalah perasaan senang atau kecewa seseorang yang muncul setelah membandingkan kinerja (hasil) produk yang dipikirkan terhadap kinerja yang diharapkan. Umumnya jika para konsumen tidak puas maka akan terjadi pertukaran komunikasi terhadap lingkungannya tentang ketidakpuasan tentang produsen dan akan berpengaruh brand image perusahaan yang akan berdampak sebagai kerugian dan hilangnya

potensi konsumen baru. Kepuasan pelanggan merupakan faktor yang menghasilkan kepercayaan dalam perusahaan yang menawarkan produk atau layanan, Sehingga semakin banyak kepuasan yang dirasakan oleh pelanggan terhadap suatu produk, maka kepercayaan pelanggan terhadap suatu produk yang dimilikinya semakin kuat. (Sebastian dan Pramono, 2021 : 700).

#### **II.2.5. Menyewa**

Pada prinsipnya, sewa menyewa merupakan suatu perjanjian yang melahirkan perikatan antara para pihak yang membuatnya. Perikatan dapat diartikan sebagai suatu hubungan hukum antara 2 (dua) orang atau lebih dalam lapangan harta kekayaan, dimana pihak yang satu berkewajiban untuk memenuhi prestasinya kepada pihak yang lain. Dari pengertian di atas, maka perikatan hanya terbatas pada suatu hubungan hukum, yaitu hubungan yang menimbulkan akibat hukum. 12 Perjanjian sewa menyewa merupakan perjanjian timbal-balik, yang artinya masing-masing pihak mempunyai kewajiban satu terhadap lainnya. Pasal 1548 KUH Perdata memberikan pengertian sewa menyewa sebagai suatu perjanjian, dengan mana pihak yang satu mengikatkan dirinya untuk memberikan kepada pihak yang lainnya kenikmatan dari sesuatu barang, selama suatu waktu tertentu dan dengan pembayaran sesuatu harga, yang oleh pihak tersebut belakangan itu disanggupi pembayarannya. (Graciano, 2020 : 325).

### II.2.6. Truk Tronton

Truk merupakan salah satu wujud kendaraan niaga yang sering digunakan untuk mobilitas barang niaga. Kendaraan jenis ini digunakan sebagai kendaraan komersial, dimana semakin banyak barang yang mampu dimuat tentunya akan semakin banyak pula nilai ekonomi yang diperoleh. Pelaku usaha ekspedisi truk biasanya melakukan modifikasi pada bagian suspensi untuk meningkatkan kekuatan struktur understeel truk. (Supriyanto dan Hermawan, 2020 : 141).

### II.2.7. PHP

PHP merupakan singkatan dari “*Hypertext Preprocessor*”. PHP adalah sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaknya mirip dengan bahasa pemrograman C, Java, ASP dan Perl ditambah beberapa fungsi PHP yang spesifik dan mudah dimengerti. PHP digunakan untuk membuat tampilan *web* menjadi lebih dinamis, dengan PHP anda bisa menampilkan atau menjalankan beberapa *file* dalam 1 *file* dengan cara di *include* dan *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa *database* walaupun dengan kelengkapan yang berbeda yaitu seperti DBM, MySQL, Oracle. (Rahmasari, 2019 : 415).

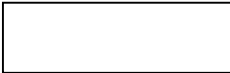
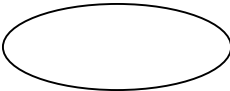
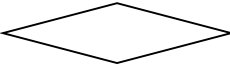
PHP adalah bahasa pemrograman yang sering disisipkan ke dalam HTML. PHP sendiri berasal dari kata *Hypertext Preprocessor*. Sejarah PHP pada awalnya merupakan kependekan dari *Personal Home Page* (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang

digunakan untuk mengolah data formulir dari *web*. Bahasa pemrograman ini menggunakan sistem *server-side*. *Server-side programming* adalah jenis bahasa pemrograman yang nantinya *script*/program tersebut akan dijalankan/diproses oleh *server*. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari. (Sugijanto, dkk, 2020 : 2).

#### II.2.8. *Entity Relationship Diagram* (ERD)

*Entity Relationship Diagram* (ERD) adalah suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. (Siagian, dkk, 2020). Simbol-simbol yang digunakan untuk menggambarkan *Entity Relationship Diagram* (ERD) dapat dilihat pada Tabel II.1.

**Tabel II.1. Simbol-Simbol *Entity Relationship Diagram* (ERD)**

| No. | Simbol                                                                              | Nama                | Keterangan                                                                                                           |
|-----|-------------------------------------------------------------------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------|
| 1.  |  | <i>Entitas</i>      | Suatu kumpulan objek atau sesuatu yang dapat dibedakan atau didefinisikan secara unik.                               |
| 2.  |  | <i>Attribut</i>     | Karakteristik dari entitas atau <i>relationship</i> yang menyediakan penjelasan detail relasi atau <i>relation</i> . |
| 3.  |  | <i>Relationship</i> | Hubungan yang terjadi antara satu relasi atau lebih.                                                                 |
| 4.  |  | <i>Link</i>         | Garis sebagai penghubung antara himpunan, relasi dan hubungan entitas dari atributnya.                               |

(Sumber : Siagian, dkk, 2020)

### II.2.9. MySQL

Definisi MySQL merupakan *software* RDMS (*Relational Database Management System*) yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak pengguna dan dapat melakukan suatu proses secara sinkron atau bersamaan. (Rahmasari, 2019 : 415).

MySQL merupakan *database engine* atau *server database* yang mendukung bahasa *database* pencarian SQL. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL. (Sugijanto, dkk, 2020 : 2).

### II.2.10. Unified Modeling Language (UML)

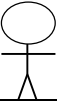
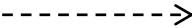
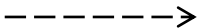
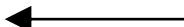
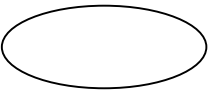
UML yaitu satu kumpulan konvensi permodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem perangkat lunak yang terkait dengan objek. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan system yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. (Andikos, 2019 : 39).

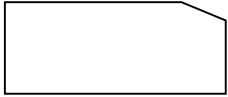
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

## 1. Use Case Diagram

*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat digambarkan dengan sumber-sumber pada Tabel II.2.

**Tabel II.2. Simbol Use Case**

| Gambar                                                                              | Nama                  | Keterangan                                                                                                                                                                                 |
|-------------------------------------------------------------------------------------|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | <i>Actor</i>          | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .                                                                                        |
|   | <i>Depedency</i>      | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>independent</i> ). |
|  | <i>Generalization</i> | Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).                                        |
|  | <i>Include</i>        | Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.                                                                                                                           |
|  | <i>Extend</i>         | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.                                                            |
|  | <i>Association</i>    | Apa yang mnghubungkan antara objek satu dengan objek lainnya.                                                                                                                              |
|  | <i>System</i>         | Menspesifikasikan paket yang menampilkan sistem secara terbatas.                                                                                                                           |
|  | <i>Use Case</i>       | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .                                                               |
|  | <i>Collaboration</i>  | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk                                                                                                                            |

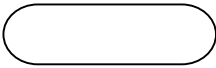
|                                                                                   |             |                                                                                               |
|-----------------------------------------------------------------------------------|-------------|-----------------------------------------------------------------------------------------------|
|                                                                                   |             | menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).             |
|  | <i>Note</i> | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi |

(Sumber : Andikos, 2019 : 39)

## 2. Diagram Aktivitas (*Activity Diagram*)

*Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.3.

**Tabel II.3. Simbol *Activity Diagram***

| Gambar                                                                              | Nama                  | Keterangan                                                                                 |
|-------------------------------------------------------------------------------------|-----------------------|--------------------------------------------------------------------------------------------|
|  | <i>Activity</i>       | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain. |
|  | <i>Action</i>         | <i>State</i> dari sistem yang mencerminkan eksekusi dari suatu aksi.                       |
|  | <i>Initial Node</i>   | Bagaimana objek dibentuk atau diawali.                                                     |
|  | <i>Activity Final</i> | Bagaimana objek dibentuk dan dihancurkan                                                   |
|  | <i>Fork Node</i>      | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran                       |

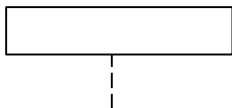
(Sumber : Andikos, 2019 : 39)

## 3. Diagram Urutan (*Sequence Diagram*)

*Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang

digambarkan terhadap waktu. *Sequence Diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.4.

**Tabel II.4. Simbol *Sequence Diagram***

| Gambar                                                                            | Nama            | Keterangan                                                                                              |
|-----------------------------------------------------------------------------------|-----------------|---------------------------------------------------------------------------------------------------------|
|  | <i>Lifeline</i> | Objek <i>entity</i> , antarmuka yang saling berinteraksi.                                               |
|  | <i>Message</i>  | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi. |
|  | <i>Message</i>  | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi. |

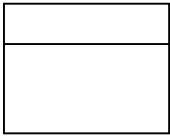
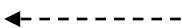
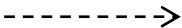
(Sumber : Andikos, 2019 : 39)

#### 4. *Class Diagram* (Diagram Kelas)

*Class* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class diagram* dapat digambarkan dengan simbol-simbol seperti pada Tabel II.5.

**Tabel II.5. *Class Diagram***

| Gambar                                                                              | Nama                    | Keterangan                                                                                                                                          |
|-------------------------------------------------------------------------------------|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
|  | <i>Generalization</i>   | Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ). |
|  | <i>Nary Association</i> | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.                                                                                         |

|                                                                                     |                      |                                                                                                                                                                   |
|-------------------------------------------------------------------------------------|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    | <i>Class</i>         | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.                                                                                           |
|    | <i>Collaboration</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>                                        |
|    | <i>Realization</i>   | Operasi yang benar-benar dilakukan oleh suatu objek.                                                                                                              |
|    | <i>Depedency</i>     | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri |
|  | <i>Assocation</i>    | Apa yang menghubungkan antara objek satu dengan objek lainnya                                                                                                     |

(Sumber : Andikos, 2019 : 39)