

## **BAB II**

### **LANDASAN TEORI**

#### **II.1 Penelitian Terkait**

Penelitian terkait bertujuan untuk mengambil beberapa referensi jurnal terkait yang digunakan dalam mendukung penelitian publikasi ilmiah dalam jurnal lokal.

Penelitian yang dilakukan oleh Niken Paramitasari, 2016, dengan judul penelitian “Analisis Kualitas Pelayanan Jasa Menggunakan Metode Servqual Di Bagian Penerimaan Mahasiswa Baru Institut Informatika Dan Bisnis Darmajaya Bandar Lampung”. Penelitian ini bertujuan untuk mengetahui penilaian kualitas pelayanan jasa Bagian Penerimaan Mahasiswa Baru (PMB) Institut Informatika dan Bisnis Darmajaya didasarkan pada hasil analisa lima dimensi kualitas jasa dalam model SERVQUAL (service quality). Jumlah populasi subjek penelitian ini adalah mahasiswa baru Tahun Ajaran 2014/2015 sebanyak 946 mahasiswa dengan sampel terdiri dari 90 orang mahasiswa yang dipilih dengan menggunakan teknik sampling Probability dan teknik Snow Ball.

Penelitian yang dilakukan oleh Marlindawati, 2013, dengan judul penelitian “Analisis Kualitas Layanan Sistem Informasi Menggunakan Servqual Method”. Penelitian ini bertujuan untuk mengetahui hubungan antara variabel kualitas layanan sistem Informasi berdasarkan ServQual Method terhadap variabel Kepuasan Pengguna Sistem Informasi yang dilakukan secara parsial dan simultan serta mengetahui faktor  $\pm$  faktor dominan yang mempengaruhi kualitas layanan Sistem Informasi terhadap Kepuasan pengguna Sistem Informasi. Penelitian ini

bermanfaat dalam memberikan bukti empiris hubungan antara variable kualitas layanan Sistem Informasi terhadap Kepuasan pengguna Sistem Informasi dan faktor faktor dominan yang mempengaruhinya sehingga dapat menjadi bahan pertimbangan dalam meningkatkan kualitas layanan Sistem informasi yang dapat meningkatkan kepuasan pengguna Sistem Informasi tersebut.

Penelitian yang dilakukan oleh Yandra Prananda, dkk, 2019, dengan judul penelitian “Penerapan Metode Service Quality (Servqual) Untuk Peningkatan Kualitas Pelayanan Pelanggan”. Penelitian ini berfokus pada peningkatan layanan dengan menggunakan metode Servqual, dimana di dalam metode tersebut membahas tentang dimensi-dimensi kualitas yakni: Tangible, Reliability, Responsiveness, Assurance dan Empathy.

## **II.2 Uraian Teoritis**

Penelitian yang dilakukan dengan cara langsung pada Sistem Informasi Kepuasan Pelanggan Pada PT.Bess Finance Tebing-Tinggi dengan Metode Service Quality adalah menggunakan beberapa aplikasi atau bahasa program serta penjelasan, yaitu :

### **II.2.1 Kepuasan Pelanggan**

Kepuasan konsumen dihasilkan dari proses perbandingan antara kinerja yang dirasakan dengan harapannya. Menurut Lovelock dan Wirtz yang dikutip oleh Fandy Tjiptono (2016:74) Kepuasan konsumen adalah suatu sikap yang diputuskan berdasarkan pengalaman yang didapatkan. Sangat dibutuhkan penelitian untuk membuktikan ada atau tidaknya harapan sebelumnya yang

merupakan bagian terpenting dalam kepuasan. Seorang konsumen puas atau tidak, sangat tergantung pada kinerja pelayanan (*Perceived Performance*) dibandingkan ekspektasi konsumen bersangkutan dan apakah sang konsumen menginterpretasikan adanya deviasi atau gap di antara kinerja dan ekspektasi tersebut. Apabila kinerja lebih rendah dibandingkan ekspektasi, maka konsumen bersangkutan akan merasa tidak puas. Apabila kinerja sama dengan ekspektasi, maka ia akan puas. Sedangkan jika kinerja melampaui ekspektasi, maka konsumen itu akan merasa sangat puas atau bahkan bahagia (*Delighted*). Windasuri & Hyacintha (2016:64) menambahkan kepuasan konsumen adalah suatu tanggapan emosional pada evaluasi terhadap pengalaman konsumsi suatu pelayanan atau jasa.

### **II.2.2 Kualitas Jasa**

Menurut (Kusmara.J, 2018) kualitas jasa adalah sebagai ukuran seberapa bagus tingkat pelayanan yang diberikan mampu sesuai dengan ekspektasi pelanggan. Berdasarkan definisi ini, kualitas jasa dapat diwujudkan melalui pemenuhan kebutuhan dan keinginan pelanggan serta ketepatan penyampaiannya untuk mengimbangi harapan pelanggan

### **II.2.3 Dimensi Pelayanan**

Zeithaml, Berry dan Parasuraman, (1985) telah melakukan berbagai penelitian terhadap beberapa jenis jasa, dan berhasil mengidentifikasi lima dimensi karakteristik yang digunakan oleh para pelanggan dalam mengevaluasi kualitas pelayanan. Kelima dimensi karakteristik kualitas pelayanan tersebut adalah (Kusmara.J, 2018) :

1. Tangibles (Nyata) Yaitu meliputi penampilan fisik, perlengkapan, pegawai, dan sarana komunikasi.
2. Reliability (Keandalan) Yaitu kemampuan memberikan pelayanan yang dijanjikan dengan segera, akurat, dan memuaskan.
3. Responsiveness (Daya Tanggap) Yaitu keinginan para staf untuk membentuk para pelanggan dan memberikan pelayanan dengan tanggap.
4. Assurance (Jaminan) Mencakup pengetahuan, kemampuan, kesopanan, dan sifat dapat dipercaya yang dimiliki para staf bebas dari bahaya, resiko atau keragu-raguan.
5. Emphaty (Empati) Meliputi kemudahan dalam melakukan hubungan, komunikasi yang baik, perhatian pribadi, dan memahami kebutuhan pelanggan.

#### **II.2.4 Metode Service Quality**

Menurut (Maya Sari, Dkk : 2020) Service Quality merupakan topik yang sering diteliti dalam bidang pemasaran karena hubungannya dengan biaya, kepuasan, retensi dan loyalitas. Service Quality diartikan sebagai kemampuan situs web dalam memfasilitasi kegiatan belanja, pembelian, dan distribusi secara efektif dan efisien. E-service quality diartikan sebagai sejauh mana sebuah situs web memfasilitasi belanja, pembelian, dan pengiriman pelayanan dan layanan. E-service quality merupakan evaluasi dan penilaian keseluruhan konsumen atas keunggulan dan kualitas penawaran e-service di pasar virtual. Kualitas pelayanan yang tepat dan efektif diperlukan untuk meningkatkan kepuasan pelanggan. Service quality merupakan metode yang unggul dalam menganalisis

perbedaan persepsi dan ekspektasi dari suatu layanan yang dirasakan pengguna. Service quality menguji lima dimensi kualitas layanan, yaitu daya tanggap (responsiveness), jaminan (assurance), empati (empathy), berwujud (tangible), dan keandalan (reliability). Setiap dimensi merupakan ukuran kualitas layanan. Pada bidang kesehatan, service quality digunakan untuk mengevaluasi kepuasan pasien terhadap kualitas layanan kesehatan rumah sakit. Hasil penelitiannya dapat menunjukkan atribut dari service quality yang mempunyai level kepuasan yang tinggi dari konsumen dan atribut yang mempunyai level kepuasan yang rendah. (Asto Buditjahjanto : 2020).

### **II.2.5 Pengukuran *Service Quality***

Pengukuran kualitas jasa dalam model Servqual ini didasarkan pada skala multiitem yang dirancang untuk mengukur harapan dan persepsi pelanggan. Serta Gap diantara keduanya pada lima dimensi kualitas jasa yaitu (Reliability, daya tanggap, jaminan, Empaty dan Bukti fisik), kelima dimensi kualitas tersebut dijabarkan dalam beberapa butir pertanyaan untuk atribut harapan dan variabel persepsi berdasarkan skala likert. Skor Servqual untuk tiap pasang pertanyaan bagi masing-masing pelanggan dapat dihitung berdasarkan rumus berikut (Kusmara.J, 2018) :

$$\underline{SKOR\ SERVQUAL = SKOR\ PERSEPSI - SKOR\ HARAPAN}$$

*(sumber : Kusmara.J, 2018)*

Skor Gap kualitas jasa pada berbagai level secara rinci dapat dihitung berdasarkan:

- a. Item-by-item analysis, misal  $P1 - H1$ ,  $P2 - H2$ , dst. Dimana  $P$  = Persepsi dan  $H$  = Harapan
- b. Dimensi-by-dimensi analysis, contoh:  $(P1 + P2 + P3 + P4 / 4) - (H1 + H2 + H3 + H4 / 4)$  dimana  $P1$  sampai  $P4$  dan  $H1$  sampai  $H4$  mencerminkan 4 pernyataan persepsi dan harapan berkaitan dengan dimensi tertentu.
- c. Perhitungan ukuran tunggal kualitas jasa/gap servqual yaitu  $(P1 + P2 + P3 + \dots + P22 / 22) - (H1 + H2 + H3 + \dots + H22 / 22)$
- d. Untuk menganalisis kualitas akan jasa pelayanan yang telah diberikan, maka digunakan rumus  $Kualitas(Q) = \frac{Persepsi(P)}{Harapan(H)} \dots\dots\dots(1)$
- Jika  $Kualitas(Q) \geq 1$ , maka kualitas pelayanan dikatakan baik.
- e. Penghitungan harapan pelanggan mengenai pelayanan dapat menggunakan rumus  $\sum y_i = (\sum STT \times 1) + (\sum TT \times 2) + (\sum CT \times 3) + (\sum T \times 4) + (\sum ST \times 5) \dots\dots\dots(2)$

Keterangan :

$\sum y_i$  = jumlah bobot jawaban pernyataan harapan variable ke - i

$\sum STT$  = jumlah orang yang memilih jawaban sangat tidak terpenuhi

$\sum TT$  = jumlah orang yang memilih jawaban tidak terpenuhi

$\sum CT$  = jumlah orang yang memilih jawaban cukup terpenuhi

$\sum T$  = jumlah orang yang memilih jawaban terpenuhi

$\sum ST$  = jumlah orang yang memilih jawaban sangat terpenuhi

1,2,3,4,5 = skor untuk skala Likert (Kusmara.J, 2018).

## II.2.6 Pengertian PHP

Menurut Jubille Enterprise (2017:1) PHP adalah bahasa pemrograman yang digunakan untuk membuat sistem berbasis *website*. Sebagai sebuah sistem, *website* tersebut hendaknya memiliki sifat dinamis dan interaktif. Memiliki sifat dinamis, artinya *website* tersebut bisa berubah tampilan kontennya sesuai kondisi tertentu (misalnya, menampilkan pelayanan yang berbeda-beda untuk setiap Konsumen). Artinya interaktif *website* tersebut dapat memberi *feedback* bagi *user* (misalnya, menampilkan hasil pencarian pelayanan).

Menurut Jubille Enterprise, (2017:1) PHP merupakan bahasa pemrograman berjenis *service-side*. Dengan demikian, PHP akan diproses oleh *server* yang hasil olahannya akan dikirim kembali ke *browser*. Oleh karena itu, salah satu *tool* yang harus tersedia sebelum memulai pemrograman PHP adalah *server*

## II.2.7 Pengertian MySQL

Menurut Solichin (2016:138), MySQL merupakan salah satu perangkat lunak basis data yang sangat populer. Saat ini tersedia versi MySQL yang berbayar (MySQL *Enterprise Edition*), namun tetap tersedia versi MySQL yang gratis (MySQL *Community Edition*). MySQL *Community Edition* dapat diunduh secara gratis, dan bebas digunakan dalam berbagai keperluan. Walaupun demikian, secara fitur dan kemampuan, MySQL versi tidak terlalu jauh berbeda dengan versi berbayar.

### Kelebihan MySQL

1. Bersifat *open source*, yang memiliki kemampuan untuk dapat dikembangkan lagi

2. Menggunakan bahasa SQL (*Structure Query Language*), yang merupakan standar bahasa dunia dalam pengolahan data
3. *Super performance* dan *realible*, tidak bisa diragukan, pemrosesan *database*-nya sangat cepat dan stabil
4. Sangat muda dipelajari (*easy of use*)
5. Memiliki dukungan support (group) penggunaan MySQL
6. *Multiuser*, dimana MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami konflik

### **II.2.8 Basis Data**

Secara umum untuk menjelaskan tentang pengertian basis data dapat ditinjau dari dua sisi, pengertian secara kharfiah dan pengertian secara istilah. Menurut pengertian secara kharfiah, basis data terdiri dari dua kata yaitu basis dan data. Basis dapat diartikan sebagai suatu markas atau gudang, tempat bersarang atau tempat berkumpul. Data dapat diartikan merupakan representasi dari fakta dunia yang mewakili suatu obyek (manusia, barang, peristiwa, keadaan dsb) yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya. Adapun menurut pengertian secara istilah, terdapat beberapa definisi yaitu sebagai berikut:[6]

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundancy) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan tertentu.
4. Kumpulan data, yang dapat digambarkan sebagai aktifitas dari satu atau lebih organisasi yang berelasi.

### **II.2.9 Normalisasi**

Normalisasi diartikan sebagai suatu teknik yang menstrukturkan atau mendekomposisi atau memecah data menggunakan cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data. Permasalahan yang dimaksud adalah berkaitan dengan penyimpangan-penyimpangan (anomalies) yang terjadi akibat adanya kerangkapan data dalam relasi dan inefisiensi pengolahan. Proses normalisasi akan menghasilkan relasi yang optimal, yaitu:

1. Memiliki struktur record yang mudah untuk dimengerti.
2. Memiliki struktur record yang sederhana dalam pemeliharaan.
3. Memiliki struktur record yang mudah untuk ditampilkan kembali untuk memenuhi kebutuhan pemakai.
4. Minimalisasi kerangkapan data guna meningkatkan kinerja sistem.

### **II.2.10 UML (*Unified Modelling Language*)**

Hasil pemodelan pada OOAD terdokumentasikan dalam bentuk *Unified Modeling Language* (UML). UML adalah bahasa spesifikasi standar yang

dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak.

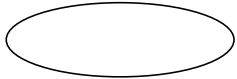
UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. UML saat ini sangat banyak dipergunakan dalam dunia industri yang merupakan standar bahasa pemodelan umum dalam industri perangkat lunak dan pengembangan sistem.

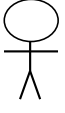

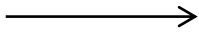
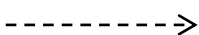
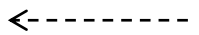
Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut :

### 1. *Use case Diagram*

Use case diagram merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dapat dikatakan use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut. Simbol-simbol yang digunakan dalam use case diagram, yaitu :

**Tabel II.1. Simbol *Use Case***

Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan</p>

	menggunakan kata kerja di awal nama <i>use case</i> .
	Aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang atau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain ( <i>required</i> ) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

(Sumber : Sri Mulyani : 2016: 245)



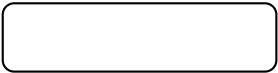
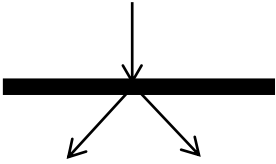
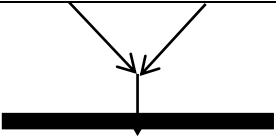
## 2. Diagram Aktivitas (*Activity Diagram*)

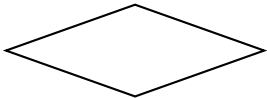

Activity diagram menggambarkan rangkaian aliran dari aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti use case atau interaksi.

Menurut Sri Mulyani (2016 : 249) : “Activity diagram adalah diagram UML yang digunakan untuk menggambarkan alur aktivitas dari suatu proses” Menurut Muhammad Muslihudin dan Oktafianto (2016 :63) mengungkapkan: “Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem”.

Simbol-simbol yang digunakan dalam *activity diagram* dapat dilihat pada tabel II.2:

**Tabel II.2. Simbol *Activity Diagram***

Gambar	Keterangan
	<i>Start point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	<i>End point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.

	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true, false</i> .
	<i>Swimlane</i> , pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa.

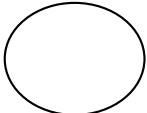
(Sumber : Sri Mulyani : 2016: 249

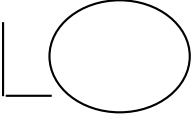
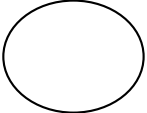

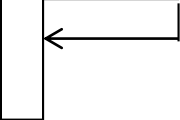


### 3. Diagram Urutan (*Sequence Diagram*)

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek.

Menurut (Irmayani & Susyatih, 2017), *Sequence Diagram* menggambarkan bagaimana sistem merespon kegiatan user. *Sequence Diagram* yang dibuat yaitu berhubungan langsung dengan kegiatan utama dari sistem anggaran pendapatan dan belanja desa berbasis objek. Simbol-simbol yang digunakan dalam *sequence diagram* dapat dilihat pada tabel II.3:

**Tabel II.3. Simbol *Sequence Diagram***

Gambar	Keterangan
	<i>EntityClass</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.

	<p><i>Boundary Class</i>, berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan <i>form</i> cetak.</p>
	<p><i>Control class</i>, suatu objek yang berisi logika sistem yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.</p>
	<p><i>Message</i>, simbol mengirim pesan antar <i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Irmayani & Susyati, 2017)

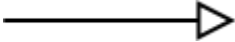

#### 4. *Class Diagram* (Diagram Kelas)

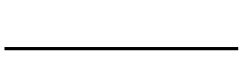
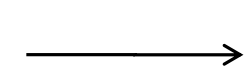
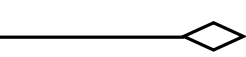

Diagram kelas (*Class Diagram*) adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem atau

perangkat lunak yang sedang kita kembangkan. Dan berikut ini merupakan penjelasan mengenai class diagram:

Menurut Sri Mulyani (2016 : 247) menyatakan bahwa : “Class Diagram adalah diagram yang digunakan untuk mempresentasikan kelas, komponenkomponen kelas dan hubungan antara masing-masing kelas” Simbol *class diagram* dan *multiplicity class diagram* dapat dilihat pada Tabel II.4 dan Tabel III.5.:

**Tabel II.4. Simbol Class Diagram**

Gambar	Keterangan			
	<p><i>Generalization</i>, untuk menghubungkan antar kelas dengan arti umum-khusus. Jadi jika ada kelas dengan arti umum-khusus. Jadi jika ada kelas bermakna umum dan kelas bermakna khusus dapat menggunakan simbol ini.</p>			
<table border="1" data-bbox="427 1507 643 1727"> <tr> <td data-bbox="435 1507 635 1581">Nama_kelas</td> </tr> <tr> <td data-bbox="435 1588 635 1662">+atribut</td> </tr> <tr> <td data-bbox="435 1668 635 1727">+operasi</td> </tr> </table>	Nama_kelas	+atribut	+operasi	<p><i>Class</i>, untuk sebuah kelas pada struktur sistem. Penulisan tidak boleh menggunakan spasi. Simbol ini memiliki 3 susunan, yaitu kotak pertama adalah nama kelas, kedua atribut dan ketiga operasi.</p>
Nama_kelas				
+atribut				
+operasi				
	<p><i>Interface</i>, untuk simbol <i>interface</i> atau dalam bahasa indonesianya antar muka. Konsep yang digunakan pun sama dengan pemrograman berorientasi object (OOP).</p>			

	<i>Association</i> , digunakan untuk menghubungkan atau merelasikan kelas satu dengan kelas yang lainnya dengan makna umum.
	<i>Directed Association</i> , adalah relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain.
	<i>Aggregation</i> , adalah relasi antar kelas dengan makna semua bagian.
	<i>Dependency</i> , adalah relasi antar kelas dengan makna kebergantungan antar kelas.

(Sumber : Sri Mulyani : 2016 : 247)

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

<b>Multiplicity</b>	<b>Penjelasan</b>
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimum 4

(Sumber : Sri Mulyani : 2016 : 247)