

## **BAB II**

### **TINJAUAN PUSTAKA**

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

Berdasarkan Penelitian mengenai penelitian terdahulu merupakan referensi untuk melakukan penelitian ini. Salah satunya penelitian tentang “Mengukur Kepuasan Customer Terhadap Layanan Sistem Informasi Perpustakaan STMIK AKBA Menggunakan Metode Kano”. Penelitian pada tahun 2016 yang dilakukan oleh First Wanita, Ashari dan Nurpaída ini berkesimpulan bahwa sistem memiliki persentase nilai lebih besar dari 75 %, dan sistem juga dapat mengetahui mengenai kebutuhan yang diinginkan oleh pengguna.

Pada tahun 2017, Ellbert Hutabri melakukan penelitian berjudul “Penerapan Metode Kano dalam Analisis Tingkat Kepuasan Mahasiswa Terhadap Penerapan Sistem Informasi Akademik Berbasis Web”. Dari penelitian tersebut disimpulkan bahwa analisis dilakukan untuk menentukan kepuasan mahasiswa terhadap sistem informasi akademik berbasis *web*, mengevaluasi kekurangan sistem informasi sesuai dengan kebutuhan mahasiswa. Hasil dari penelitian tersebut, mahasiswa merasa paling puas pada fungsi pengambilan Kartu Rencana Studi (KRS) yaitu dengan tingkat kepuasan 0.90 dan pada fungsi melihat dan menghapus KRS pada halaman yang berbeda mahasiswa merasa kurang puas dengan tingkat kepuasan 0,52 sampai 0,41.

Penelitian berikutnya mengenai “Analisis Peningkatan Kualitas Layanan Pelanggan PT.POS Indonesia Persero Kantor Wilayah Surabaya Selatan dengan

Metode Servqual, Metode Kano, dan Quality Function Deployment (QFD)”. Penelitian pada tahun 2014 yang dilakukan oleh Lissa Rosdiana Noer, Udisubakti Ciptomulyo, dan Indung Sudarso ini menunjukkan bahwa hanya terdapat empat kategori untuk 25 variabel dari kualitas layanan yaitu *Must be*, *One dimensional*, *Attractive* dan *Indifferent*.

Penelitian selanjutnya “Analisis Layanan Biro Perjalanan Umrah Menggunakan Integrasi Servqual dan Model Kano di PT XYZ”. Penelitian pada tahun 2017 yang dilakukan oleh Alfiska Reva Supena, Agus Achmad Suhendra dan Adhya Rare Tiara ini berkesimpulan ada 8 atribut kebutuhan jemaah terhadap layanan umrah di PT.XYZ yang perlu di prioritaskan, yaitu kondisi ruang manasik, peralatan kondisi transport (bus) dari Bandung ke Bandara Soekarno Hatta, kesesuaian hotel selama tinggal di Mekkah, kualitas makanan yang disajikan selama tinggal di Mekkah, kecepatan karyawan menanggapi keluhan, kecepatan pembimbing ibadah dalam menanggapi jemaah, pembimbing dalam mengenali jemaah dan ketersediaan kompensasi untuk jemaah.

## **II.2. Sistem Informasi**

Sistem informasi dalam organisasi dapat dikaitkan sebagai sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima dengan menggunakan sistem informasi atau peralatan sistem lainnya. Sistem informasi akan bekerja pada

fungsi-fungsi organisasi sesuai dengan derajat kebutuhan informasi pada tingkat manajemen dalam organisasi.

Sistem Informasi adalah suatu sistem didalam satu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan untuk pihak luar tertentu. (Tata Sutabri ; 2012)

### **II.3. Kepuasan Pelanggan**

Kata kepuasan (*satisfaction*) berasal dari bahasa Latin “*satis*” (artinya cukup baik, memadai) dan *facio* (melakukan atau membuat). Kepuasan bisa diartikan sebagai “upaya pemenuhan sesuatu” atau “membuat sesuatu memadai”. Untuk mengukur kepuasan pelanggan ada beberapa cara yang biasanya dilakukan beberapa perusahaan yaitu, sistem keluhan dan saran, *ghost-shopping*, *lost customer analysis*, dan survei kepuasan pelanggan. ( Fandi Tjiptono ; 2015)

### **II.4. Metode Kano**

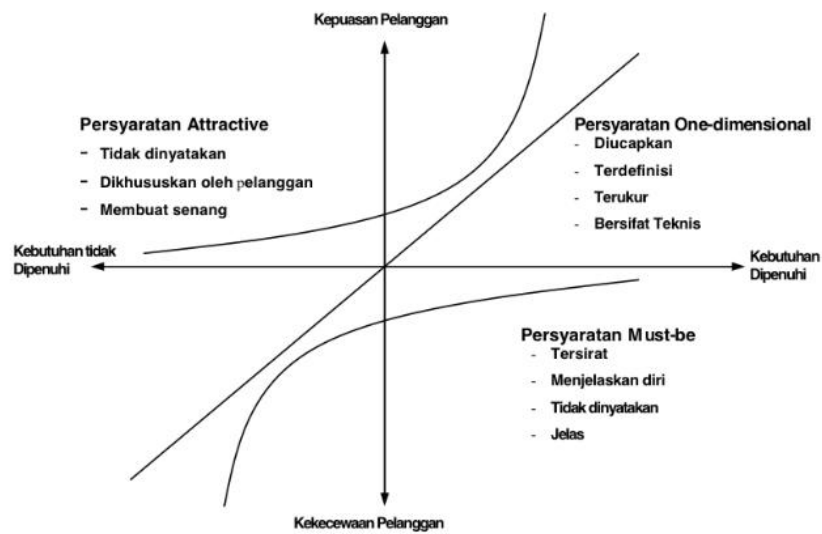
Dikembangkan oleh Dr.Noriaki Kano, Ia mengatakan bahwa kepuasan pelanggan merupakan fungsi non-linear dari *service functionality* dan terdiri atas 3 komponen, yaitu *attractive component*, *proportional (one-dimensional) component*, dan *must-be component* (Kano, et al., 1984). Kano menegaskan

bahwa layanan berkelas dunia harus memenuhi ketiga tipe kebutuhan dan bukan sekedar apa yang dikatakan pelanggan. Penyedia jasa harus mampu mengidentifikasi ekspektasi pelanggan saat ini dan ekspektasi potensial yang berkemungkinan untuk menyenangkan pelanggan. Ketiga komponen Metode Kano meliputi :

1. *Must-be (basic) needs*. Tipe ini merupakan kebutuhan yang diharapkan atau bahkan dituntut pelanggan untuk dipenuhi oleh sebuah produk atau jasa yang dibeli. Jika kebutuhan ini tidak terpenuhi, pelanggan akan amat tidak puas, namun jika terpenuhi tidak akan menimbulkan kepuasan pelanggan.
2. *One-dimensional (performance) needs*. Cakupannya adalah fitur layanan yang berpotensi meningkatkan kepuasan pelanggan dan bukan sekedar *basic product*. Semakin tinggi tingkat pemenuhannya, maka semakin tinggi pula tingkat kepuasan pelanggan.
3. *Attractive (exciting) needs*. Faktor ini mencerminkan kebutuhan yang melampaui ekspektasi pelanggan. Tidak diungkapkan secara eksplisit oleh pelanggan. Jika tidak terpenuhi tidak akan menurunkan kepuasan pelanggan namun jika terpenuhi akan jauh meningkatkan kepuasan pelanggan karena mencapai “*wow factor*” dan “*surprising quality*”. (Fandi Tjiptono ; 2015)

#### **II.4.1. Langkah-Langkah Metode Kano**

Pada penelitian ini penulis menggunakan metode kano untuk mengukur kepuasan layanan, berikut adalah penjabaran dari metode penelitian :



**Gambar II.1. Model Kano**

Langkah-langkah penelitian menggunakan Metode Kano adalah :

1. Identifikasi ide/permintaan pengguna yang akan diukur
2. Membuat kuisisioner Kano.

Dalam pembuatan kuisisioner, setiap satu pertanyaan memiliki dua bagian yaitu *functional* dan *disfunctional*. Memproses hasil jawaban kuisisioner dengan menggunakan *Tabulation of Surveys* untuk memproses hasil jawaban Tabel Evaluasi Kano sebagai berikut.

**Tabel II.1. Tabel Evaluasi Kano**

| Kebutuhan pelanggan<br>↓<br>→         |              | Pertanyaan Difungsional (Negatif) |         |          |         |              |
|---------------------------------------|--------------|-----------------------------------|---------|----------|---------|--------------|
|                                       |              | 1.Suka                            | 2.Harus | 3.Netral | 4.Boleh | 5.Tidak Suka |
| Pertanyaan<br>Fungsional<br>(Positif) | 1.Suka       | Q                                 | A       | A        | A       | O            |
|                                       | 2.Harus      | R                                 | I       | I        | I       | M            |
|                                       | 3.Netral     | R                                 | I       | I        | I       | M            |
|                                       | 4.Boleh      | R                                 | I       | I        | I       | M            |
|                                       | 5.Tidak suka | R                                 | R       | R        | R       | Q            |

Keterangan :

O : *One dimensional requirement* (satu ukuran)

A : *Attractive requirement* (menarik)

R : *Reversal requirement* (kemunduran)

Q : *Questionable requirement* (netral)

M : *Must be* (harus ada)

3. Menghitung jumlah masing-masing kategori kano dalam tiap atribut untuk menentukan *requirement* tertinggi dari masing-masing item.

4. Menentukan nilai Kepuasan

$$\frac{A + O}{A + O + M + I}$$

5. Menentukan nilai Ketidakpuasan

$$\frac{O + M}{(A + O + M + I) \times (-1)}$$

## II.5. Microsoft Visual Basic 2010

Visual basic merupakan bahasa pemrograman *event-driven* generasi ketiga dan *Integrated Development Environment* (IDE) dari Microsoft yang mulai diperkenalkan pada tahun 1991. Visual basic merupakan pengembangan dari BASIC yang dibuat sebagai bahasa pemrograman yang mudah dipelajari dan digunakan. Visual basic memungkinkan proses *Rapid Application Development* (RAD) dari aplikasi antarmuka, mengakses database, dan membuat kontrol objek. (Jubilee ; 2018)

Visual Basic adalah sebuah bahasa pemrograman komputer. Bahasa Pemrograman adalah perintah-perintah atau instruktur yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. Visual Basic kini seakan-akan menjadi kiblat bagi para software developer dan menjadi salah satu bahasa yang wajib dipelajari oleh berbagai kalangan, jika mereka ingin sukses di dunia komputer.

## II.6. SQL Server

Microsoft *SQL Server* merupakan salah satu produk RDBMS (*Relation Database Management System*). Salah satu produk andalan yang dibuat oleh Microsoft yang berfungsi sebagai relation database. Microsoft *SQL Server* mendukung SQL sebagai bahasa pemroses *query*. SQL merupakan bahasa standart internasional untuk proses *query database* dan SQL ini sudah banyak

digunakan pada hampir semua aplikasi, baik itu *e-commerce*, pendidikan, organisasi, pemerintah, atau bahkan personal sekalipun. (Agus Saputra ; 2018)

Sebagai software, fungsi utama SQL Server adalah menampung dan menggunakan data yang terintegrasi dengan aplikasi, baik pada komputer yang sama ataupun dari komputer lain dalam jaringan. Dilengkapi dengan fitur *In-Memory OLTP* memungkinkan untuk memindahkan suatu tabel secara individual ke dalam struktur data *in-memory* khusus, *Managed Backup* ke dalam Azure *memback-up* secara otomatis berdasarkan interval *requery*, Azure VM untuk replika ketersediaan. (Jubilee ; 2018)

## **II.7. UML (*Unified Modelling Language*)**


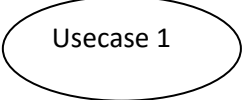

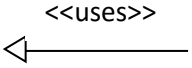
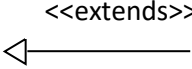
*Unified Modeling Language (UML)* merupakan kumpulan diagram-diagram yang sudah memiliki standart untuk pembangunan perangkat lunak berbasis objek. UML memiliki banyak diagram, empat diantaranya akan dibahas pada bab ini yaitu, *use case* diagram, *class* diagram, *sequence* diagram dan *state diagram/activity* diagram. (Feri Sulianta ; 2017)

### **II.7.1. Use Case Diagram**

*Use case* diagram merupakan diagram yang harus dibuat untuk pertama kali. Diagram *use case* akan menggambarkan apa yang dilakukan dan dikerjakan oleh aktor. Yang disebut aktor adalah pengguna aplikasi. Berikut adalah simbol-simbol yang digunakan untuk membuat *use case* diagram, antara lain :



Tabel II.2. Simbol *Use case Diagram*

| Gambar  | Nama     | Keterangan  |
|---|----------|---|
|    | Aktor    | Merupakan pengguna dari sistem. Penamaan aktor menggunakan kata benda.  |
|    | Use case | Merupakan pekerjaan yang dilakukan oleh aktor. Penamaan use case dengan kata kerja.   |
|    | Asosiasi | Hubungan antara aktor dengan use case.  |
|   | Include  | Hubungan antara use case dengan use case, include menyatakan bahwa sebelum pekerjaan dilakukan harus mengerjakan pekerjaan lain terlebih dahulu.                          |
|  | Extends  | Hubungan antara use case dengan use case, extends menyatakan bahwa, jika pekerjaan yang dilakukan tidak sesuai atau terdapat kondisi khusus maka dilakukan pekerjaan itu. |

(Sumber :Feri Sulianta ; 2017)

### II.7.2. Class Diagram

*Class diagram* dibuat setelah *use case diagram*. Pada pembuatan diagram ini harus menjelaskan hubungan apa saja yang terjadi antara suatu objek dengan objek yang lain sehingga terbentuklah suatu sistem aplikasi.

Pembuatan class diagram dibagi menjadi dua bagian, yaitu kelas itu sendiri dan relasi antarkelas. Kelas dibagi menjadi tiga bagian, yakni nama kelas, atribut kelas, serta operasi kelas (*methods*). Nama kelas adalah nama dari kelas itu

sendiri. Atribut adalah data yang dimiliki oleh kelas tersebut, dan operasi kelas adalah menunjukkan apa yang kelas tersebut bisa lakukan.

Setelah membuat kelas, selanjutnya harus membuat relasi antarkelas. Relasi memiliki *multiplicity* yang berarti banyak ke banyak. Berikut adalah indikator relasi dalam *class diagram*, antara lain :

**Tabel II.3. Simbol Class Diagram**

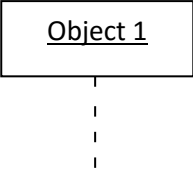

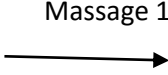

| Indikator | Arti             | Penjelasan  |
|-----------|------------------|---|
| 0..1      | Kosong ke satu   | Dalam hal ini objek yang berinteraksi dengan objek dari kelas lainnya hanya 1 atau tidak sama sekali. |
| 1         | Hanya satu       | Objek yang berinteraksi dengan objek dari kelas lainnya hanya 1                                       |
| 0..*      | Kosong ke banyak | Objek yang berinteraksi dengan objek dari kelas lainnya minimal 1 atau tidak sama sekali              |
| 1..*      | Satu ke banyak   | Objek yang berinteraksi dengan objek dari kelas lainnya minimal 1 atau lebih.                         |

(Sumber :Feri Sulianta ; 2017)

### II.7.3. Sequence Diagram

*Sequence diagram* dibuat untuk mengetahui alur dan interaksi antarobjek. Isi dari *sequence diagram* harus sama dengan *use case* dan *class diagram*. Satu *use case* tunggal akan digambarkan satu *sequence diagram*-nya. Berikut adalah simbol-simbol dalam *sequence diagram*, antara lain :

Tabel II.4. Simbol *Sequence Diagram*



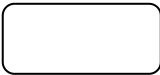
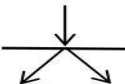
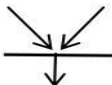


| Simbol  | Nama         | Keterangan  |
|---|--------------|---|
|    | Objek/ aktor | Sebuah objek yang berasal dari kelas. Atau dapat dinamai dengan kelasnya saja. Aktor termasuk objek. Garis putus - putus menunjukkan garis hidup suatu objek.           |
|    | Aktivasi     | Menunjukkan masa hidup dari objek.  |
|    | Pesan        | Interaksi antara satu objek dengan objek lainnya. Objek dapat mengirimkan pesan ke objek lain. Interaksi antarobjek ditunjukkan pada bagian operasi pada diagram kelas. |
|  | Return       | Pesan kembali dari komunikasi antarobjek.   |

(Sumber :Feri Sulianta ; 2017)

#### II.7.4. Activity Diagram

*Activity Diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah system atau proses bisnis. Simbol-simbol yang digunakan dalam *activity diagram*, yaitu :

Tabel II.5. Activity Diagram

| Gambar  | Keterangan   |
|---|--|
|    | <i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas.   |
|    | <i>End Point</i> , akhir aktifitas   |
|    | <i>Activities</i> menggambarkan suatu proses atau kegiatan bisnis.   |
|   | <i>Fork</i> (Percabangan) digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan parallel menjadi satu. |
|  | <i>Join</i> (Penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.  |
|  | <i>Decision points</i> menggambarkan pilihan untuk pengambilan keputusan, true, false  |
|  | <i>Swimlane</i> pembagian <i>activity</i> diagram untuk menunjukkan siapa melakukan apa  |

(Sumber :Feri Sulianta ; 2017)