BAB II
TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Untuk mendukung keberhasilan penelitian ini, penulis melakukan pendekatan teoritis melalui beberapa literatur yang berhubungan dengan penelitian yang dilakukan. Beberapa uraian penelitian terdahulu yang menjadi acuan dalam penelitian ini yaitu:

Andrian Dwi Putra, 2018, dengan penelitian "Analisa Kepuasan Konsumen Event Organizer XYZ menggunakan metode *Service Quality*". Penelitian ini menghasilkan Penelitian ini menghasilkan persentase hasil analisa tingkat kepuasan konsumen sebesar 69% terhadap Event Organizer XYZ dengan menggunakan sampel sekitar 120 konsumen sehingga didapat bahwa konsumen yang ada di Event Organizer XYZ sangat puas dengan pelayanan yang diberikan oleh perusahaan.

Zahkiyah Rahmiati, 2019, dengan penelitian "Sistem analisa tingkat kepuasan mahasiswa terhadap kegiatan belajar mengajar pada PSTI UNRAM dengan menggunakan metode *Service Quality*". Penelitian ini menghasilkan persentase hasil analisa tingkat kepuasan mahasiswa terhadap KBM pada PSTI UNRAM menggunakan metode *service quality* sebesar 75,51 % dengan sampel sebanyak 40 orang mahasiswa.

Nanny Raras Setyoningrum, 2018, dengan judul penelitian "Analisis tingkat kepuasan pengguna sistem informasi ujian akhir (SIUAS) menggunakan metode

service quality (servqual)". Penelitian ini menghasilkan tingkat kepuasan yang dirasakan pengguna sebesar 84,12% dengan gap/selisish sebesar 15,88% artinya pengguna berada pada rentang kategori sangat puas. Dari kelima dimensi tersebut variable tangible memiliki gap/selisih terkecil yaitu 12,74% sedangkan gap/selisih terbesar ada pada variable reliability, yaitu sebesar 18,06,%.

Adhy Purnama, 2017, dengan judul penelitian "Peningkatan tingkat kepuasan dosen terhadap kualitas pelayanan pelatihan jurnal berkualitas melalui metode servqual*"*. Penelitian ini menghasilkan dapat disimpulkan bahwa kualitas pelayanan di Kopertis Wilayah III belum sesuai dengan harapan peserta. Hal ini ditunjukkan dengan nilai kepuasan peserta pada lima dimensi (Tangible, Relibility, Responsivness, Assurance, Empathy) bernilai negatif yang berarti peserta tidak puas dengan pelayanan yang telah diberikan, sehingga kelima dimensi tersebut perlu dilakukan perbaikan. Berdasarkan ratarata nilai gap dari kelima dimensi diperoleh tingkat kepuasan sebesar -1.908 yang berarti peserta merasa tidak atau belum puas. Atribut yang menjadi prioritas utama untuk dilakukan perbaikan dalam kualitas pelayanan di Kopertis Wilayah III yaitu pada atribut keramahan, perhatian dan kesopanan staf instruktur beserta panitia (6), informasi yang jelas mengenai fasilitas dan peraturan bagi peserta (7), fasilitas ruang kelas menunjang kegiatan belajar mengajar (9), snack dan makanan berkualitas(11) dan rangkaian materi yang diberikan bermutu sesuai dengan sasaran program (14). Atribut-atribut yang menjadi prioritas utama untuk dilakukan perbaikan ini merupakan atribut yang berada di kuadran 1 pada diagram kartesius yang harus segera di perbaiki guna perbaikan kualitas Kopertis Wilayah III

sebagai peyelenggara kegiatan.

Felix Wuryo Handono, 2017, dengan judul penelitian "Tingkat kepuasan konsumen terhadap kualitas pelayanan pada CV Jia Pradana dengan metode Servqual". Penelitian ini menghasilkan Dimensi Tangibles berpengaruh negatif dan belum berpengaruh signifikan terhadap kepuasan konsumen, hal ini menunjukkan bahwa CV Jia Pradana perlu meningkatkan mutu atau kualitas secara langsung pada dimensi ini agar dapat menambah kepuasan konsumen. Dimensi Reliability juga berpengaruh negatif dan belum berpengaruh secara signifikan terhadap kepuasan konsumen. Perlu adanya peningkatan mutu pada setiap indikator yang ada pada dimensi ini agar dapat meningkatkan kepuasan konsumen dari CV Jia Pradana. Sama dengan dimensi Tangibles dan Reliabiliy, dimensi Responsiveness juga memiliki pengaruh negatif dan belum berpengaruh secara signifikan terhadap kepuasan konsumen. Perlu adanya peningkatan mutu pada setiap indikator yang ada pada dimensi Responsiveness agar dapat meningkatkan kepuasan kepada para konsumen.

II.2. Landasan Teori

II.2.1. Kepuasan Pelanggan

Kepuasan merupakan fungsi dari perbedaan antara kinerja yang dirasakan dengan harapan. Apabila kinerja di bawah harapan, maka pelanggan akan kecewa. Bila kinerja sesuai dengan harapan, pelanggan akan puas. Sedangkan bila kinerja melebihi harapan, pelanggan akan sangat puas. Harapan pelanggan dapat dibentuk

oleh pengalaman masa lampau, komentar dari kerabatnya serta janji dan informasi pemasaran dan saingannya. Pelanggan yang puas akan setia lebih lama, kurang sensitive terhadap harga dan memberi komentar yang baik tentang perusahaan. Untuk menciptakan kepuasan pelanggan, perusahaan harus menciptakan dan mengelola suatu sistem untuk memperoleh pelanggan yang lebih banyak dan kemampuan untuk mempertahankan pelanggannya

II.2.2. Metode Service Quality (Servqual)

Metode servqual adalah suatu kuesioner yang digunakan untuk mengukur kualitas jasa. Cara ini mulai dikembangkan pada tahun 1980-an oleh Zeithaml, Parasuraman & Berry, dan telah digunakan dalam mengukur berbagai kualitas jasa. Dengan kuesioner ini, kita bisa mengetahui seberapa besar celah (gap) yang ada di antara persepsi pelanggan dan ekspektasi pelanggan terhadap suatu perusahaan jasa. Kuesioner servqual dapat diubah-ubah (disesuaikan) agar cocok dengan industri jasa yang berbeda-beda pula (misalnya bank, restoran, atau perusahaan telekomunikasi)Metode servqual memiliki dua perspektif yaitu *perspektif internal* dan *perspektif eksternal*. Perspektif eksternal digunakan untuk memahami apa yang diharapkan konsumen, dirasakan konsumen,

Rumus dalam perhitungan metode *service quality* (servqual) :

Nilai *Service Quality* (Servqual) = Nilai Persepsi – Nilai Harapan

II.2.3. PHP (*Hypertext Preprocessor*)

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

Menurut Jubille Enterprise, (2017:1) PHPmerupakan bahasa pemrograman berjenis *service-side*. Dengan demikian, PHPakan diproses oleh *server* yang hasil olahannya akan dikirim kembali*browser.*Olehkarenaitu,salahsatu*tool*yang harustersediasebelum memulai pemrograman PHPadalah *server.*

II.2.4. *Data Base Management System*

Data Base Management System adalah Perangkat lunak komputer khusus yang disediakan oleh produsen komputer yang digunakan untuk membuat, mengakses, mengontrol, dan mengelola database (Yunus et al., 2017).

Menurut (Yunus et al., 2017) Keuntungan menggunakan *DBMS* yaitu:

1. Penggunaan Data Bersama (*The Data Can Be Shared*).
2. Mengurangi Kerangkapan Data (*Redudancy Can Be Reduced*).
3. Menghindari Ketidakkonsistenan Data (*Inconsistency Can Be Avoided*).
4. Integritas Data Terpelihara (*Integrity Can Be Maintained*).
5. Keamanan Terjamin (*Security Can Be Enforced*).

6. Kebutuhan User Yang Kompleks Dapat Teratasi (*Balanced conflicting requirements*).
7. Pelaksanaan Standarisasi (*Standards Can Be Enforced*).
8. Meningkatkan Produktivitas (*Increased productivity*).
9. Layanan Back up dan Recovery Semakin Baik (*Improved backup and recovery services*).

II.2.5. UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) digunakan sebagai suatu cara untuk mengkomunikasikan idenya kepada para pemrogram serta calon pengguna sistem/perangkat lunak. Dengan adanyan bahasa yang bersifat standar, komunikasi perancang dengan pemrogram (komunikasi antar anggota kelompok pengembang) serta calon pengguna diharapkan menjadi mulus, adapun pengertian UML menurut para ahli dapat dipaparkan sebagai berikut:

Menurut Shofwan Hanief & Dian Pramana (2018 : 166) menyatakan bahwa: "*Unified Modelling Language* (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisas, merancang dan mendokumentasikan sistem piranti lunak"

Menurut Eng RH. Sianipar (2017 :75) menyatakan bahwa:"UML merupakan singkatan *Unifed Modelling Language*, yang telah menjadi notasi popular untuk mempresentasikan perancangan atas sebuah program berorientasi objek".

Alat bantu yang digunakan dalam perancangan berorientasi objek berbasiskan

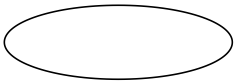
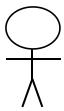
UMLadalah sebagai berikut:

1. *Use case* Diagram

Menurut Sri Mulyani (2018 : 245) menyatakan bahwa *Use Case Diagram* yaitu *diagram* yang menggambarkan dan merepresentasikan aktor, *use cas*e dan *dependencies* suatu proyek dimana tujuan dari diagram ini adalah menjelaskan konsep hubungan antara sistem dengan dunia luar

Jadi, dapat disimpulkan *Use case* adalah langkah – langkah atau urutan kegiatan yang dilakukan actor dan sistem informasi yang akan dibuat.Secara singkat, *Use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan.

Tabel II.1. Simbol *Use Case*

Gambar	Keterangan
	*Use case* menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama *use case*.
	Aktor adalah *abstraction* dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasikan 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 *use case*, tetapi tidak memiliki control terhadap *use case*.

	Asosiasi antara aktor dan *use case*, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengidikasikan aliran data.
	Asosiasi antara aktor dan *use case* yang menggunakan panah terbuka untuk mengidinkasikan bila aktor berinteraksi secara pasif dengan sistem.
	Include, merupakan di dalam *use case* lain (*required*) atau pemanggilan *use case* oleh *use case* lain, contohnya adalah pemanggilan sebuah fungsi program.
	Extend, merupakan perluasan dari *use case* lain jika kondisi atau syarat terpenuhi.

(Sumber : Sri Mulyani : 2018: 245)

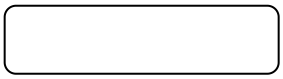
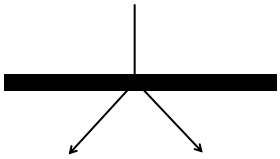
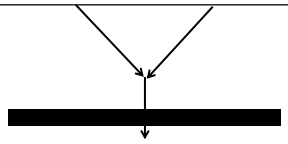
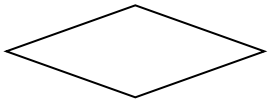
2. Diagram Aktivitas (*Activity Diagram*)

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 (2018 : 249) : "Activty diagram adalah diagram UML yang digunakan untuk menggambarkan alur aktivitas dari suatu proses" Menurut Muhammad Muslihudin dan Oktafianto (2018 :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
	Start point, diletakkan pada pojok kiri atas dan merupakan awal aktifitas.
	End point, akhir aktifitas.
	*Activites*, menggambarkan suatu proses/kegiatan bisnis.
	*Fork* (Percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan pararel menjadi satu.
	*Join* (penggabungan) atau rake, digunakan untuk menunjukkan adanya dekomposisi.
	*Decision Points,* menggambarkan pilihan untuk pengambilan keputusan, *true, false.*
	Swimlane, pembagian *activity* diagram untuk menunjukkan siapa melakukan apa.

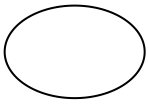
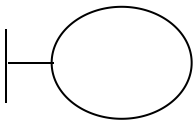
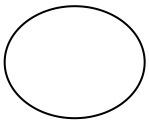
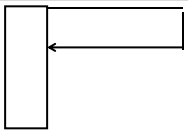
(Sumber : Sri Mulyani : 2018: 249)

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
	*EntityClass,* merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	*Boundary Class,* berisi kumpulan kelas yang menjadi *interface* atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan *form* cetak.
	*Control class,* suatu objek yang berisi logika sistem yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	Message, simbol mengirim pesan antar *class*.
	*Recursive,* menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	Activation, activation mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	Lifeline, garis titik-titik yang terhubung dengan objek,

	sepanjang *lifeline* terdapat *activation*.

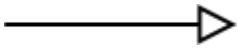
(Sumber : Irmayani & Susyati, 2017)

4. Diagram Kelas (*Class Diagram)*

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 (2018 : 247) menyatakan bahwa : "Class Diagram adalah diagram yang digunakan untuk mempresentasikan kelas, komponen-komponen kelas dan hubungan antara masing-masing kelas" Simbol *class diagram* dan *multiplicity class diagram* dapat dilihat pada Tabel II.4 dan TabelII.5.:

Tabel II.4. Simbol *Class Diagram*

Gambar	Keterangan
	*Generalization,* 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.
	Class, 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.
	Interface, untuk simbol *interface* atau dalam bahasa indonesianya antar muka. Konsep yang digunakan pun sama dengan pemrograman berorientasi object (OOP).

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

(Sumber : Sri Mulyani : 2018 : 247)

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

Multiplicity	Penjelasan
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 : 2018 : 247)