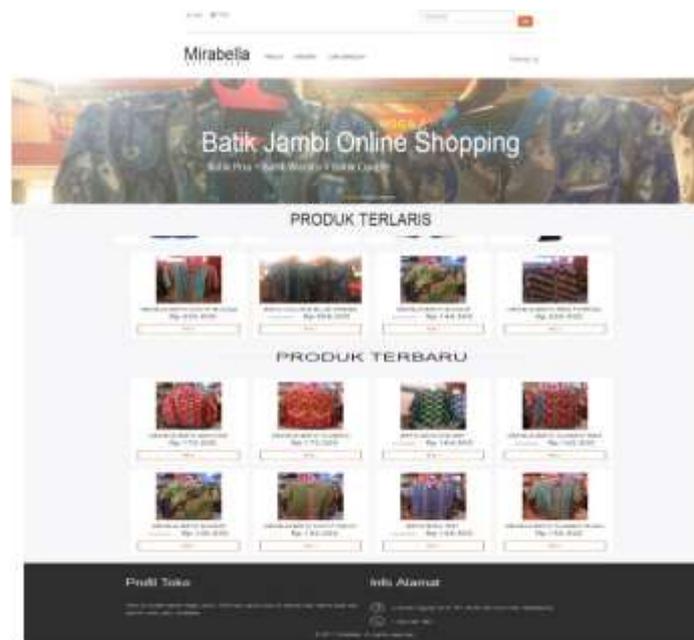


BAB II

LANDASAN TEORI

2.1 Kajian Pustaka

Penelitian yang dilakukan oleh Andi Ridho Rachman dan Erick Fernando (2017) dalam jurnal yang berjudul “E-Commerce Berbasis *Website* Pada Toko Mirabella Batik Jambi” dalam penelitiannya penulis akan membuat sebuah aplikasi Mirabella Batik Jambi berbasis *website*. Aplikasi yang di buat ini memiliki beberapa konten seperti: informasi batik, transaksi penjualan, *user account*, *order detail*, *cart*, *checkout*. Tampilan aplikasi penjualan batik disajikan pada Gambar 2.1.



Gambar 2.1 Tampilan Aplikasi Mirabella Batik Jambi.

Menurut Rulia Puji Hastanti dan Indah Uly Wardati (2018) dalam jurnal yang berjudul “Sistem Penjualan Berbasis *Web (E-commerce)* Pada Tata Distro Kabupaten Pacitan “ berpendapat bahwa *dengan* meluasnya perdagangan global, tidak cukup bagi perusahaan hanya mengandalkan selebaran dan iklan dalam media pemasaran . Mengaplikasikan situs *e-commerce* pada toko tersebut maka konsumen akan mengetahui barang atau buku yang ditawarkan toko tersebut. Konten yang terdapat

dalam aplikasi meliputi: product, pencarian, checkout, profile, informasi barang. Tampilan aplikasi pemesanan barang disajikan pada Gambar 2.2.

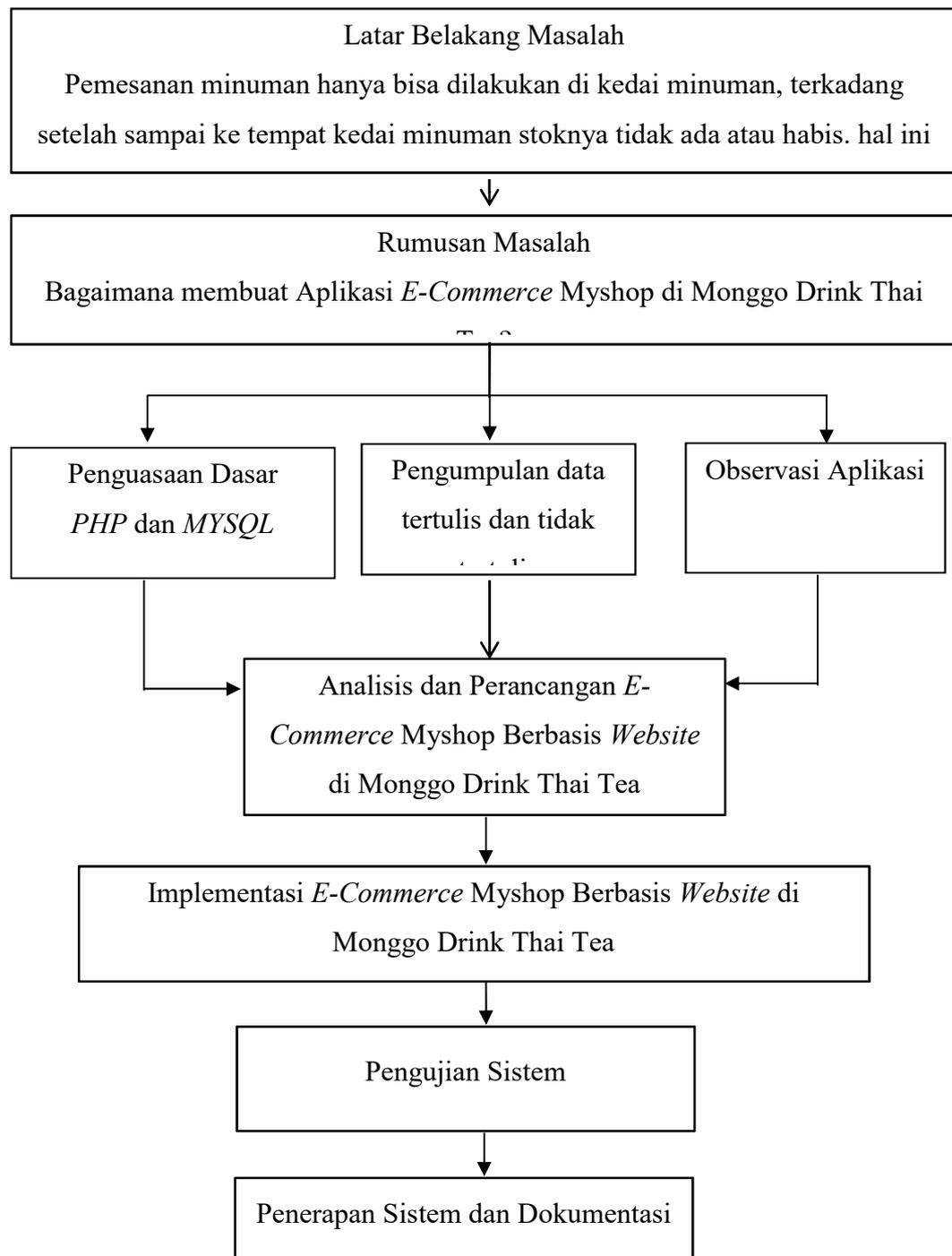


Gambar 2.2 Tampilan Aplikasi Penjualan Pada Tata Distro.

Penelitian yang dilakukan oleh Dian Nur Astiati (2016) dalam skripsi yang berjudul “Sistem Pengolahan Data bengkel Dan Sparepart Pada Distro Ban” dalam penelitiannya penulis akan membuat sebuah Sistem pengolahan data bengkel dan sperpart pada distro ban berbasis *web*, sistem pengolahan data ini buat untuk mempermudah pembuatan laporan,cek stok dan meminimalisir kesalahan saat terjadi transaksi. *Sistem pengolahan data ini memiliki beberapa konten seperti: penjualan, stok, laporan penjualan, retur barang. Perbedaan dengan system yang akan di buat, pembayaran bisa di lakukan dengan scan barcode QRIS maupun tunai.*

2.2 Kerangka Pemikiran

Berikut ini adalah tahapan kerangka pemikiran dalam membuat *E-Commerce* Myshop Berbasis *Website* di *Monggo Drink Thai Tea*. Kerangka pemikiran ditunjukkan pada Gambar 2.3.



Gambar.2.3 Kerangka Pikiran

1. Latar Belakang Masalah

Sistem transaksi dan pembuatan laporan masih manual serta pemesanan minuman hanya bisa dilakukan di kedai minuman, sehingga *costumer* datang secara langsung ke kedai minuman untuk memesan minuman.

2. Rumusan Masalah

Bagaimana membuat Aplikasi *E-Commerce* Myshop di Monggo Drink Thai Tea?

3. Penguasaan *PHP* dan *MYSQL*

Tahap untuk mempelajari dasar-dasar *PHP* dan *MYSQL* agar lebih menguasai program-program yang akan digunakan untuk membangun aplikasi.

4. Pengumpulan data

Tahap pengumpulan data pada penelitian ini melalui observasi, wawancara dan studi literatur. Pengumpulan data bertujuan untuk mengetahui permasalahan dan kebutuhan informasi mengenai *e-commerce* myshop berbasis *website* di Monggo Drink Thai Tea.

5. Observasi Aplikasi *E-Commerce* Myshop Berbasis *Website*

Merupakan tahap pengamatan contoh aplikasi yang telah ada, jurnal, buku, maupun karya ilmiah untuk kajian yang dapat dijadikan referensi untuk pembangunan aplikasi.

6. Analisis dan Perancangan *e-commerce* myshop berbasis *website* di Monggo Drink Thai Tea.

Merupakan tahap analisis kebutuhan, perancangan sistem dan desain tampilan untuk membuat aplikasi *e-commerce* myshop berbasis *website*.

7. Implementasi *e-commerce* myshop berbasis *website* di Monggo Drink Thai Tea.

Implementasi sistem adalah langkah-langkah atau prosedur-prosedur yang dilakukan dalam menyelesaikan desain aplikasi yang telah disetujui, untuk menginstal, menguji dan memulai sistem baru atau sistem yang diperbaiki.

8. Pengujian Sistem

Pengujian sistem adalah prosedur yang dilakukan untuk menyelesaikan rancangan sistem yang telah disetujui, menguji sistem, menginstal serta memulai

penggunaan sistem baru atau sistem yang telah diperbaiki. Dalam pengujian sistem ini menggunakan metode *webqual*.

2.3 Landasan Teori

2.3.1 Pengertian *E-Commerce*

Menurut Tata Sutabri (2014:108) Perdagangan elektronik atau pedagang (bahasa Inggris: *Electronic Commerce* atau *E-Commerce*) adalah penyebaran, pembelian, penjualan, pemasaran barang dan jasa melalui sistem elektronik seperti internet atau televisi, *www (World Wide Web)*, atau jaringan komputer lainnya. *E-commerce* dapat melibatkan transfer dana elektronik, pertukaran data elektronik, sistem manajemen inventori otomatis, dan sistem pengumpulan data otomatis.'

2.3.2 Konsep Dasar *Web*

Menurut Ardhana (2012), *web* adalah salah satu layanan sajian informasi yang menggunakan konsep *hyperlink* (tautan), yang memudahkan *surfer* (sebutan pemakai komputer yang melakukan *browsing* atau penelusuran informasi melalui (internet).

Menurut Risnandar, dkk (2013), *web* adalah salah satu fitur dari internet di samping fitur-fitur lain seperti *e-mail*, *remote (SSH)*, dan *FTP*, berdasarkan uraian di atas dapat disimpulkan *web* sebagai salah satu fitur dari internet dan layanan sajian informasi yang menggunakan konsep *hyperlink* (tautan), yang memudahkan *surfer* (sebutan pemakai komputer yang melakukan *browsing* atau penelusuran informasi melalui (internet).

2.3.3 *Website*

Menurut Yuhefizar (2013), *website* adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Hubungan antara satu halaman *web* dengan halaman *web* yang lainnya disebut dengan *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext*.

Menurut Hikmah, dkk (2015), *website* atau situs dapat diartikan sebagai kumpulan halaman yang digunakan untuk menampilkan informasi teks, gambar

diam atau gerak, animasi, suara, dan/atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

Berdasarkan pengertian para ahli di atas dapat disimpulkan bahwa *website* adalah kumpulan dari halaman-halaman *web* yang berisi sebuah data atau informasi baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman.

2.3.4 Bahasa Pemrograman

1. PHP (*Hypertext Preprocessor*)

Menurut Hidayatullah dan Kawistara (2017), PHP atau disingkat *Hypertext Preprocessor* ini adalah sebuah Bahasa *scripting* khususnya digunakan untuk *web development*. Karena sifatnya yang *server side scripting*, maka untuk menjalankan PHP harus menggunakan *web server*.

PHP juga dapat diintegrasikan dengan HTML, JavaScript, JQuery, Ajax. Namun, pada umumnya PHP lebih banyak digunakan bersamaan dengan file bertipe HTML. Dengan menggunakan PHP bisa membuat *website powerful* yang dinamis dengan disertai manajemen *database*-nya. Selain itu juga penggunaan PHP yang sebagian besar dapat jalan di banyak *platform*.

2. HTML (*Hyperlink Text Markup Language*)

Menurut Sidik dan Husni (2017), HTML kependekan dari *Hyperlink Text Markup Language*. Dokumen HTML adalah file teks murni yang dapat dibuat dengan editor teks sembarang. Dokumen ini dikenal sebagai *web page*. Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*. Dokumen ini umumnya berisi informasi atau *interface* aplikasi di dalam internet.

Menurut Sibero (2013), *Hyperlink Text Markup Language* (HTML) adalah bahasa yang digunakan pada dokumen *web* sebagai bahasa untuk pertukaran dokumen *web*.

Berdasarkan uraian di atas maka kesimpulannya *Hyperlink Text Markup Language* (HTML) adalah bahasa yang digunakan untuk menulis halaman *web* sebagai pertukaran dokumen *web* dan dokumen ini umumnya berisi informasi atau *interface* aplikasi di dalam internet.

2.3.5 Web Server

Menurut Supono dan Putratama (2016), *web-server* adalah perangkat lunak *server* yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui HTTP atau HTTPS dari klien itu, yang dikenal sebagai *web browser* dan mengirimkan kembali (reaksi) hasil dalam bentuk situs yang biasanya merupakan dokumen HTML.

Menurut Sibero (2013), *web server* adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak.

Berdasarkan pendapat di atas dapat disimpulkan bahwa *web server* adalah sebuah komputer yang terdiri dari perangkat keras dan perangkat lunak server yang berfungsi untuk menerima permintaan dalam bentuk situs *web* melalui *web browser* dan mengirimkan kembali (reaksi) hasil dalam bentuk situs yang biasanya merupakan dokumen HTML.

2.3.6 Web Browser

Menurut Sibero (2012), *web browser* adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi *web*.

Menurut Winarno (2007), *web browser* merupakan suatu program yang dirancang untuk mengambil informasi-informasi dari suatu *server* komputer pada jaringan internet. Jadi untuk mengakses *web* diperlukan suatu program yaitu *web browser* atau disebut *browser* saja.

Berdasarkan pengertian di atas dapat disimpulkan bahwa *browser* adalah sebuah perangkat lunak yang dirancang untuk menampilkan dan mengakses informasi atau halaman-halaman yang tersedia di *web server*.

2.3.7 Internet

Menurut Ahmadi dan Hermawan (2013), internet adalah komunikasi global yang menghubungkan seluruh komputer di dunia meskipun berbeda sistem operasi dan mesin.

Sedangkan Menurut Sibero (2013), internet (*interconnected network*) adalah jaringan komputer yang menghubungkan antar jaringan secara global, internet dapat juga disebut jaringan dalam suatu jaringan yang luas.

Berdasarkan kesimpulan di atas internet adalah sistem jaringan komunikasi secara global yang menghubungkan seluruh komputer di dunia dengan waktu dan wilayah tak terbatas sehingga semua orang dapat selalu terhubung di waktu siang maupun malam dengan platform apapun.

2.3.8 Definisi Basis Data

Definisi menurut Hidayatullah dan Kawistara (2017), basis data dapat didefinisikan sebagai himpunan kelompok data yang saling berhubungan yang diorganisasikan sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah. Prinsip utamanya adalah pengaturan data. Tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data.

Secara lebih lengkap pemanfaatan data dilakukan untuk memenuhi berikut ini :

1. Kecepatan dan Kemudahan (*Speed*)
2. Efisiensi ruang penyimpanan (*Space*)
3. Keakuratan (*Accuracy*)
4. Ketersediaan (*Availability*)
5. Kelengkapan (*Completeness*)
6. Keamanan (*Security*)
7. Pemakaian Bersama (*Sharability*)

2.3.9 MySQL

Menurut Sibero (2013:97), MySQL atau dibaca “My Sekuel” dengan suatu RDBMS (*Relational Database Management System*) yaitu aplikasi sistem yang menjalankan fungsi pengolahan data.

Menurut Hidayatullah dan Jauhari (2015:180), MySQL adalah salah satu aplikasi DBMS yang sudah banyak oleh para pemogram aplikasi *web*. Contoh DBMS lainnya adalah : PostgreSQL (freeware), SQL Server, MS Access dari Microsoft, DB2 dari IBM, Oracle dan Oracle Corp, Dbase, FoxPro, dsb.

Berdasarkan penjelasan di atas dapat disimpulkan bahwa MySQL adalah aplikasi DBMS yang menjalankan fungsi pengelolaan data untuk membangun sebuah aplikasi berbasis *web* maupun desktop.

2.3.10 Pengertian OOP (Objek Oriented Program)

Menurut Sukamto dan Shalahuddin (2014:100) mendefinisikan bahwa, ” *Object Oriented Programming* (OOP) adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya”.

Berdasarkan pengertian yang ada dapat disimpulkan bahwa Object Oriented Programing (OOP) merupakan suatu strategi atau cara baru untuk membuat program atau merancang sistem dengan memperhatikan objek .

Pada saat ini, metode berorientasi objek banyak dipilih karena metodologi lama banyak menimbulkan masalah seperti adanya kesulitan pada saat mentransformasi hasil dari satu tahap pengembangan ke tahap berikutnya, misalnya pada metode pendekatan terstruktur, jenis aplikasi yang dikembangkan saat ini berbeda dengan masa lalu. Aplikasi yang dikembangkan pada saat ini sangat beragam (aplikasi bisnis, real-time, utility, dan sebgaiannya) dengan platform yang berbeda-beda, sehingga menimbulkan tuntutan kebutuhan metodologi pengembangan yang dapat mengakomodasi ke semua jenis aplikasi tersebut (Sukamto dan Shalahuddin 2014:100).

2.3.11 Pengertian UML

Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman

berorientasi objek. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek (Sukamto dan Shalahudin, 2014).

Sukamto dan Shalahudin (2014), pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Berikut ini penjelasan singkat dari pembagian kategori tersebut.

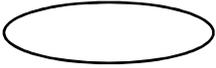
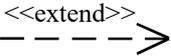
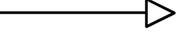
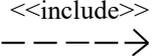
1. *Structure diagram*, yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan. Structure diagram terdiri dari *class diagram*, *object diagram*, *component diagram*, *composite structure diagram*, *package diagram* dan *deployment diagram*.
2. *Behavior diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem. *Behavior diagram* terdiri dari *use case diagram*, *activity diagram*, *state machine system*.
3. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun interaksi antar sub sistem pada suatu sistem. *Interaction diagram* terdiri dari *sequence diagram*, *communication diagram*, *timing diagram*, *interaction overview diagram*.

2.3.12 Pengertian Use Case Diagram

Sukamto dan Shalahudin (2014), *use case* atau *diagram use case* 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. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang

berhak menggunakan fungsi-fungsi itu. Simbol *use case diagram* disajikan pada Tabel 2.1.

Tabel 2.1. Simbol *Use Case Diagram* (Sukamto dan Shalahuddin, 2013).

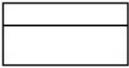
NO	SIMBOL	NAMA	KETERANGAN
1		<i>Use case</i>	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan kata kerja di awal di awal frase nama <i>use case</i> .
2		<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3		<i>Association</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
4		<i>Extend</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5		<i>Generalization</i>	Hubungan generalisasi dan spesialisasi (umum – khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6		<i>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.

2.3.13 Pengertian *Class Diagram*

Sukamto dan Shalahudin (2014), diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan method atau operasi. Berikut penjelasan atribut dan method :

1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau method adalah fungsi-fungsi yang dimiliki oleh suatu kelas. Simbol *class diagram* disajikan pada Tabel 2.2.

Tabel 4. Simbol *Class Diagram* (Sukamto dan Shalahuddin, 2013).

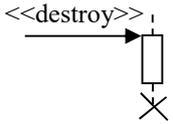
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.3.14 Pengertian *Activity Diagram*

Sukamto dan Shalahudin (2014), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu di

perhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Simbol *activity diagram* disajikan pada Tabel 2.3.

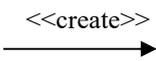
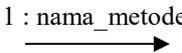
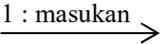
Tabel 2.3 Simbol *Activity Diagram* (Sukamto dan Shalahuddin, 2013).

No	Gambar	Nama	Keterangan
1		<i>Start Point</i>	Diletakkan pada objek kiri atas dan merupakan awal aktivitas
2		<i>End Point</i>	Akhir aktivitas
3		<i>Activities</i>	Menggambarkan suatu proses atau kegiatan bisnis
4		<i>Fork</i>	Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau digunakan untuk menggabungkan dua kegiatan menjadi satu
5		<i>Join</i>	Digunakan untuk menunjukkan dekomposisi
6		<i>Decision Points</i>	Menggambarkan pilihan untuk pengambilan keputusan true atau false
7		<i>Swimlane</i>	Pembagian activity diagram dalam sebuah urutan yang sama
8		Pesan tipe <i>destroy</i>	Menyatakan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

2.3.15 Pengertian *Sequence Diagram*

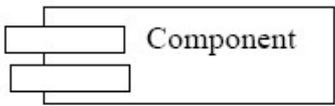
Sukamto dan Shalahudin (2014), diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dengan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. *Sequence diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.4.

Tabel 2.4. Simbol *Sequence Diagram* (Sukamto dan Shalahuddin, 2016).

NO	SIMBOL	NAMA	KETERANGAN
1		Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat itu sendiri.
2		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
3	<div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama objek : nama kelas</div>	Objek	Menyatakan objek yang berinteraksi pesan.
4		Waktu aktif	Menyatakan objek dalam keadaan aktif dan berinteraksi.
5		Pesan tipe create	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat
6		Pesan tipe call	Menyatakan suatu objek memanggil operasi / metode yang ada pada objek lain atau dirinya sendiri.
7		Pesan tipe send	Menyatakan bahwa suatu objek mengirimkan data / masukan / informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.

2.3.16 Pengertian *Component Diagram*

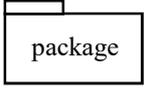
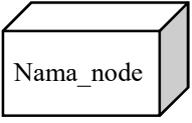
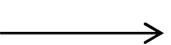
Component diagram menurut Sukamto dan Shalahudin (2014), diagram komponen atau *component diagram* dibuat untuk menunjukkan organisasi dan ketergantungan diantara kumpulan komponen dalam sebuah sistem. diagram komponen fokus pada komponen sistem yang dibutuhkan dan ada didalam sistem. Hubungan antara *component* dan *class*, *Component* adalah implementasi *software* dan sebuah *class*. *Class* mewakili abstraksi dari serangkaian *attribute* dan *operation*. Hal terpenting yang perlu diingat tentang *class* dan *component* adalah sebuah *component* bisa jadi merupakan implementasi dari lebih dari sebuah *class*. *Component diagram* mempunyai simbol-simbol yang ditunjukkan pada Tabel 2.5. Tabel 2.5. Simbol *Component Diagram* (Sukamto dan Shalahudin, 2014).

SIMBOL	NAMA	KETERANGAN
	Komponen	Sebuah komponen melambangkan sebuah entitas software dalam sebuah sistem.
	Dependency	Sebuah Dependency digunakan untuk menotasikan relasi antara dua komponen. Notasinya adalah tanda panah putus-putus yang diarahkan kepada komponen tempat sebuah komponen itu bergantung.

2.3.17 Pengertian *Deployment Diagram*

Menurut (Rosa & Shalahuddin, 2016) *deployment diagram* menunjukkan konfigurasi komponen dalam proses eksekusi aplikasi. Diagram *deployment* juga dapat digunakan untuk memodelkan sistem tambahan yang menggambarkan rancangan *device*, *node*, dan *hardware*. Sistem *client/server*, sistem terdistribusi murni, rekayasa ulang aplikasi. Berikut merupakan tabel mengenai simbol *deployment diagram* yang ditunjukkan pada Tabel 2.6.

Tabel 2.6 Simbol-simbol *Deployment*

NO	SIMBOL	NAMA	KETERANGAN
1.		<i>Package</i>	<i>Package</i> merupakan sebuah bungkusuan dari satu atau lebih <i>node</i> .
2.		<i>Node</i>	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak yang tidak dibuat sendiri (<i>software</i>), jika didalam <i>node</i> disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelum pada diagram komponen.
3.		<i>Dependency</i>	Kebergantungan antar <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
4.		<i>Link</i>	Relasi antar <i>node</i> .

2.3.18 Metode Webqual

WebQual merupakan salah satu metode atau teknik pengukuran kualitas *website* berdasarkan persepsi pengguna. WebQual sudah mulai dikembangkan sejak tahun 1998 dan telah mengalami beberapa interaksi dalam penyusunan dimensi dan butir pertanyaan (Pratama, 2015). Webqual merupakan alat untuk mengukur usability (kegunaan), information quality (kualitas informasi), dan interactionquality (kualitas interaksi). Masing-masing dimensi terdiri dari beberapa pertanyaan, yang akan dijadikan dasar dalam menyusun kuesioner seperti yang disajikan pada tabel berikut:

Tabel 2.7. Kualitas Penggunaan (*Usability Quality*)

No	Deskripsi Indikator
1	<i>Website</i> mudah untuk dioperasikan.
2	Interaksi dengan <i>website</i> sangat mudah dimengerti dan tidak membingungkan.
3	Pengguna merasa mudah untuk bernavigasi dalam <i>website</i> .
4	<i>Website</i> mudah untuk digunakan.
5	<i>Website</i> memiliki tampilan yang menarik.
6	Desain <i>website</i> sesuai dengan tipe <i>website</i> .
7	<i>Website</i> menunjukkan kemampuannya.
8	<i>Website</i> dapat memberikan pengaruh / pengalaman positif bagi pengguna.

Tabel 2.8. Kualitas Informasi (*Information Quality*)

No	Deskripsi Indikator
1	<i>Website</i> menyajikan informasi yang akurat.
2	Informasi yang disajikan <i>website</i> dapat dipercaya.
3	Informasi yang disajikan tepat waktu atau <i>up to date</i> .
4	Informasi yang disajikan relevan.
5	Informasi yang disajikan mudah dipahami.
6	Informasi yang disajikan sangat detail.
7	Informasi disajikan dalam format yang sesuai.

Tabel 2.9. Kualitas Interaksi (*Interaction Quality*)

No	Deskripsi Instansi
1	<i>Website</i> memiliki reputasi yang baik.
2	Pengguna merasa aman untuk melakukan transaksi atau interaksi dengan <i>website</i> ..
3	<i>Website</i> menjaga informasi pribadi pengguna.
4	<i>Website</i> memberi ruang untuk personalisasi.
5	<i>Website</i> memberi ruang untuk komunitas.
6	<i>Website</i> memudahkan pengguna untuk berkomunikasi dengan organisasi.
7	Pengguna merasa yakin dengan layanan/informasi yang disediakan karena sesuai dengan yang dijanjikan.

Tabel 2.10 Dimensi Keseluruhan

No	Deskripsi Indikator
1	Pendapat secara umum tentang <i>website</i> ini.

Beberapa tahap yang harus dilakukan untuk melakukan pengujian webqual antara lain:

a. Uji Instrumen

Menurut (Prakosad, 2017), uji instrument digunakan untuk mengetahui deskripsi mengenai variable-variabel dalam penelitian, uji instrument terdiri dari uji validitas dan uji reliabilitas.

b. Uji Validitas

Validitas mengandung dua bagian yaitu bahwa instrumen pengukuran adalah mengukur secara aktual konsep dalam pertanyaan dan bukan beberapa konsep yang lain; dan bahwa konsep dapat diukur secara akurat. Oleh karena itu, suatu instrumen pengukur bisa dikatakan valid jika mengukur apa yang hendak diukur dan mampu mengungkap data tentang karakteristik gejala yang diteliti secara tepat (Bailey, dalam Silalahi, 2013).

c. Reliabilitas

Reliabilitas adalah derajat sejauh mana ukuran menciptakan respon yang sama sepanjang waktu dan lintas situasi. Suatu alat ukur dikatakan reliabel jika hasil pengukuran dari alat ukur tersebut stabil dan konsisten (Silalahi, 2012). Dengan demikian reliabel adalah suatu keadaan di mana instrumen penelitian tersebut akan tetap menghasilkan data yang sama meskipun disebarkan pada sampel yang berbeda dan pada waktu yang berbeda. Uji reliabilitas akan dilakukan dengan menggunakan uji statistik *cronbach's alpha* (α) dengan ketentuan bahwa variabel yang diteliti dinyatakan *reliabel* apabila nilai *cronbach's alpha* (α) adalah di atas 0,6 (Ghozali, 2014).

d. Uji Asumsi Klasik

Menurut (Yoedo, Susilowati, & Julianto, 2016), uji asumsi klasik digunakan sebelum uji regresi linier berganda, ada data yang harus terpenuhi agar kesimpulan dari regresi bisa menjadi kuat, antara lain uji normalitas, uji multikolinearitas, uji heteroskedastisitas dan uji autokorelasi.

e. Uji Normalitas

Pengujian ini bertujuan untuk mengetahui apakah data terdistribusi secara normal/tidak. Pengujian normalitas normalitas dapat dilihat dari titik-titik yang menyebar mengikuti garis diagonal pada *Normal PP-Plot Regression*, di mana jika titik-titik tersebut mengikuti garis diagonal maka dikatakan bahwa data yang digunakan adalah berdistribusi normal (Prakosad, 2017).

f. Uji Multikolinearitas

Pengujian ini bertujuan untuk menguji apakah dalam model regresi ditemukan adanya korelasi variabel independen. Model regresi yang baik seharusnya tidak terjadi korelasi variabel independen. Pendeteksian dilakukan dengan menggunakan *tolerance value* dan *Variance Inflation Factor* (VIF). Jika

nilai *tolerance value* > 0.10 dan $VIF < 10$, maka tidak terjadi multikolinearitas, sebaliknya jika nilai *tolerance value* < 0.10 dan $VIF > 10$ maka terjadi multikolinearitas (Prakosad, 2017).

g. Uji Heterokedastisitas

Pengujian ini bertujuan untuk menguji apakah dalam model regresi terjadi ketidaksamaan *variance* dari *residual* satu pengamatan ke pengamatan yang lain. Jika *variance* dari residual satu pengamatan ke pengamatan yang lain tetap maka disebut Homokedastisitas dan jika berbeda maka disebut Heterokedastisitas. Model yang baik adalah yang homokedastisitas atau tidak terjadi heterokedastisitas. Pendeteksiannya dilakukan dengan melihat grafik plot antara nilai prediksi variabel terikat yaitu ZPRED (Z predictor) dengan residualnya SRESID (*standardized residual*). Deteksi terjadinya heterokedastisitas dapat dilakukan dengan melihat penyebaran titik-titik pada sumbu Y. Jika data tersebut baik di atas maupun di bawah sumbu Y, maka dapat disimpulkan bahwa tidak terjadi gejala heterokedastisitas (Prakosad, 2017).

h. Uji Autokorelasi

Uji Autokorelasi merupakan pengujian yang dilakukan untuk menguji ada tidaknya pengaruh antara variabel pengganggu dalam masing-masing variabel bebas. Uji autokorelasi bisa menggunakan tes Durbin Watson dengan ketentuan sebagai berikut :

$dW < dL$: berarti ada autokorelasi positif (+).

$dL < dW < 4-dL$: tidak dapat disimpulkan.

$dU < dW < 4-dU$: berarti tidak terjadi autokorelasi.

$4-dU < dW < 4-dL$: tidak dapat disimpulkan.

$dW > 4-dL$: berarti ada autokorelasi negatif (-).

Menurut Prakosad (2017) menyatakan bahwa uji autokorelasi merupakan pengujian asumsi dalam regresi dimana variabel dependen tidak berkorelasi dengan dirinya sendiri. Maksud korelasi dengan diri sendiri adalah bahwa nilai dari variabel dependen tidak berhubungan dengan nilai variabel itu sendiri, baik

nilai variabel sebelumnya atau nilai periode sesudahnya. Dasar pengambilan keputusan adalah sebagai berikut :

Angka D-W di bawah -2 : berarti ada autokorelasi.
 positif Angka D-W diantara -2 sampai +2 : berarti tidak ada.
 autokorelasi Angka D-W di atas +2 : berarti ada autokorelasi negatif.

i. Analisis Regresi Linier Berganda

Menurut Prakosad (2017) mengemukakan analisis regresi linier berganda digunakan untuk melakukan prediksi, bagaimana perubahan nilai variable dependen bila nilai variabel independen dinaikan atau diturunkan nilainya. Analisis ini digunakan dengan melibatkan dua atau lebih variabel bebas antara variabel dependen (Y) dan variabel independen (X1, X2, dan X3), cara ini digunakan untuk mengetahui kuatnya hubungan antara beberapa variabel bebas secara serentak terhadap variabel terkait dan dinyatakan dengan rumus.

Teknik analisis data yang digunakan dalam penelitian ini adalah teknik analisis kuantitatif dengan menggunakan model Regresi Linier Berganda (*Multiple Linier Regresion Method*) yang akan diolah dengan program *SPSS for Windows* versi 20. Metode ini digunakan untuk mengetahui pengaruh variabel bebas terhadap variabel tergantung. Rumuskan analisis regresi linier berganda sebagai berikut :

$$Y' = a + b_1X_1 + b_2X_2 + b_3X_3$$

Keterangan :

Y' = Kepuasan Pengguna, yang merupakan variabel dependen.

a = Konstanta, merupakan nilai terkait yang dalam hal ini adalah Y' pada saat variabel independennya adalah 0 (X1, X2, X3=0).

b1 = Koefisien regresi berganda variabel independen X1 terhadap variabel Y', bila variabel X2 dan X3 dianggap konstan.

b2 = Koefisien regresi berganda variabel independen X2 terhadap variabel Y', bila variabel X1 dan X3 dianggap konstan.

β_3 = Koefisien regresi berganda variabel independen X3 terhadap variabel Y', bila variabel X1 dan X2 dianggap konstan.

X1 = Kualitas Penggunaan yang merupakan variabel independen ke-1.

X2 = Kualitas Informasi yang merupakan variabel independen ke-2.

X3 = Kualitas Interaksi yang merupakan variabel independen ke-3.

j. Pengujian Hipotesis

Menurut Prakosad (2017) uji hipotesis dilakukan untuk menganalisis dan menarik kesimpulan terhadap permasalahan yang diteliti. Pengujian ini dimaksudkan untuk mengetahui apakah terdapat pengaruh variabel bebas terhadap variabel tidak bebas. Uji hipotesis yang digunakan adalah uji F dan uji t. Berdasarkan program SPSS, jika probabilitas hasil (*p value*) dari nilai $F < 5\%$ maka hipotesis yang diajukan diterima; dan jika probabilitas hasil (*p value*) dari nilai $F > 5\%$ maka hipotesis yang diajukan ditolak. Demikian juga apabila probabilitas hasil (*p value*) dari nilai $t < 5\%$ maka hipotesis yang diajukan diterima; dan jika probabilitas hasil (*p value*) dari nilai $t > 5\%$ maka hipotesis yang diajukan ditolak.

k. Uji F (Uji Simultan)

Menurut Prakosad (2017) uji F dimaksud untuk menguji hipotesis apakah variabel-variabel bebas secara bersama-sama mempunyai pengaruh terhadap variabel tergantung. Tahap-tahap uji F adalah :

1. Merumuskan hipotesis

$$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$$

Artinya variabel bebas secara bersama-sama tidak mempunyai pengaruh terhadap variabel tergantung.

$$H_a : \beta_1 = \beta_2 = \beta_3 \neq 0$$

Artinya variabel bebas secara bersama-sama mempunyai pengaruh terhadap variabel tergantung.

2. Menentukan tingkat signifikan

Tingkat signifikan yang diharapkan (α) adalah 5% dengan *degree of freedom* (df) = (k-1) dan (n-k) guna menetapkan nilai F tabel, di mana n= jumlah responden dan k = jumlah variabel bebas

3. Menentukan nilai F hitung, dengan rumus :

$$F_{hit} = F(k; n - k)$$

4. Membandingkan nilai F hitung dengan F tabel, bila:

$F_{hit} < F_{tab}$ = berarti H_0 diterima dan H_1 ditolak

$F_{hit} > F_{tab}$ = berarti H_1 diterima dan H_0 ditolak.

1. Uji T (Uji Parsial)

Menurut Prakosad (2017) uji t dimaksudkan untuk menguji hipotes apakah variabel bebas secara parsial mempunyai pengaruh terhadap variabel tergantung.

Tahap-tahap uji t adalah :

Merumuskan hipotesis

$H_0: b_1 = 0$

Artinya variabel bebas secara parsial tidak mempunyai pengaruh terhadap variabel tergantung.

$H_a: b_1 \neq 0$

Artinya variabel bebas secara parsial mempunyai pengaruh terhadap variabel tergantung.

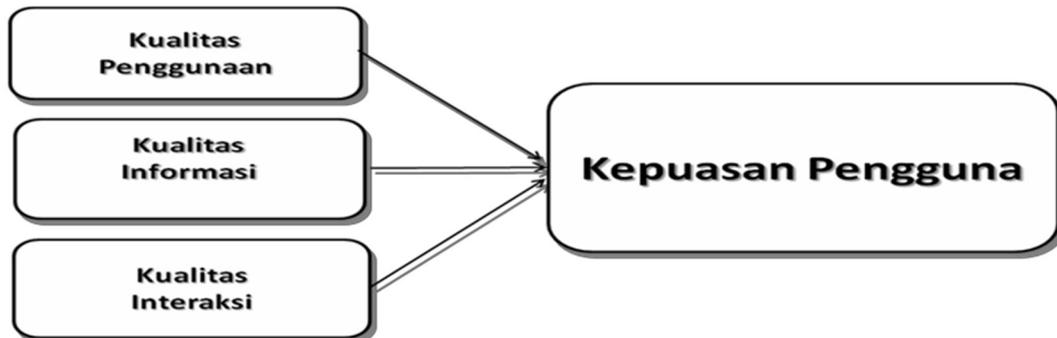
m. Uji Koefisien Determinasi Berganda (R^2)

Koefisien determinasi merupakan koefisien yang nilainya dimaksudkan untuk mengetahui seberapa besar variasi perubahan dalam satu variabel independen. Koefisien korelasi digunakan untuk menentukan koefisien determinasi. Dalam konteks ini, koefisien determinasi merupakan kuadrat dari koefisien korelasi yang dinotasi dengan R^2 (Prakosad, 2017). Oleh karena itu, semakin kuat korelasi diantara variabel yang diamati maka semakin besar pula koefisien determinasi yang dihasilkan. Koefisien determinasi dinyatakan dalam persen (%) sehingga harus dikalikan dengan 100%. Artinya adalah bahwa persentase dari variasi

perubahan dalam variabel Y adalah disebabkan oleh adanya variasi perubahan dalam variabel X.

n. Kerangka Konseptual

Kerangka Konseptual disajikan pada Gambar 2.4.



Gambar 2.4 Kerangka Konseptual

Berdasarkan kerangka konseptual tersebut maka akan dilakukan analisis data untuk mengetahui pengaruh variabel kualitas penggunaan, kualitas informasi, dan kualitas interaksi terhadap kepuasan pengguna.