

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

Dalam penelitian Tugas Akhir ini digunakan 3 tinjauan studi yang nantinya dapat mendukung penelitian yang akan dilakukan, dimana tinjauan studi yang diambil dari jurnal berikut :

Tinjauan yang pertama yakni dari Charolina dkk (2015), dengan judul Analisis dan Perancangan Pengolahan Data Pada Bengkel Distro Ban. Masalah yang dihadapi Bengkel Distro Ban yakni dalam pengolahan data bengkel service dan sparepart masih menggunakan cara yang manual. Semua data barang masuk dan keluar masih dicatat dalam buku, sehingga kemungkinan terjadi kesalahan dalam penghitungan maupun ketersediaan stok barang yang ada menjadi tidak akurat atau tidak terkontrol.

Tujuan penulisan pada penelitian ini adalah membuat perancangan sistem pengolahan data pada Bengkel Distro Ban. Metode pengumpulan data dilakukan dengan cara observasi, wawancara, dan pustaka. Metode analisis sistem menggunakan Analisa PIECES, dan perancangan sistem secara terstruktur. Hasil dari penelitian ini yakni menghasilkan analisis dan perancangan sistem pengolahan data bengkel dan sparepart pada Distro Ban. Analisis sistem menggunakan metode PIECES dan perancangan sistem menggunakan perancangan sistem secara terstruktur. Selanjutnya dapat diimplementasikan menggunakan bahasa pemrograman.

Tinjauan yang kedua yakni dari Hasanudin (2018), dengan judul “Rancang dan Bangun Sistem Informasi *Inventory* Barang Berbasis *Web* (Studi Kasus PT. Nusantara Sejahtera Raya)”. Penelitian ini mengangkat masalah kekurangan yang ada pada sistem *order* barang yang sedang berjalan sekarang yaitu dalam pencatatan stok barang masih dihitung manual membuat celah lupa mencatat sehingga tidak seimbang antara jumlah stok barang di catatan dengan fisik yang ada, sehingga

tidak mengetahui ketersediaan barang yang ada di gudang, untuk luar kota *order* barang masih menggunakan sistem fax ke kantor pusat terkadang fax tersebut hilang dan *order* tersebut tidak di proses, tidak adanya pencatatan data *order* barang.

Tujuan penelitian ini adalah membuat sistem informasi *inventory* barang, yang dapat mengelola barang masuk dan keluar, stok barang, transaksi pemesanan barang dengan menggunakan *Data Flow Diagram* (DFD) untuk pemodelan terstruktur. Metode penelitian yang digunakan mengacu pada konsep pengembangan sistem *waterfall*. Model *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun *software*. Sistem Informasi yang telah dihasilkan pada perancangan ini dapat mengelola data barang masuk dan keluar, data stok barang, pemesanan barang dalam satu aplikasi berbasis *web*.

Tinjauan Pustaka yang ketiga dari Fahrival, dkk. (2019) dengan judul “Perancangan Sistem *Inventory* Barang Pada UD. Minang Dewi Berbasis *Website*”. Penelitian ini mengangkat masalah perkembangan perusahaan di berbagai bidang saat ini semakin pesat, maka sebab itu perusahaan diharuskan memperluas usahanya dan mempersiapkan gudang. Gudang merupakan salah satu bagian penting dari sebuah usaha distributor yang berfungsi sebagai tempat penyimpanan barang masuk. Di era modern ini masih banyak perusahaan distributor yang menggunakan sistem lama dalam penjualan barang, sehingga data stok barang tidak dapat diketahui dengan pasti.

Tujuan dari Penelitian ini agar keberadaan barang dalam gudang dapat terkelola dengan baik. Perancangan sistem dilakukan dengan metode *waterfall* yang merupakan model pengembangan sistem informasi yang sistematis dan sekuensial. Metode penggalan data yang digunakan adalah wawancara, observasi, dan studi kepustakaan. Dalam pembuatan sistem alat yang digunakan dalam membuat perancangan dan desain sistem *inventory* yaitu dengan menggunakan *Unified Modeling Language* (UML). Adapun pemrograman yang digunakan adalah pemrograman *website* dengan menggunakan *database* MySQL dan PHP. Hasil dari penelitian ini yakni aplikasi sistem *inventory* barang pada UD. Minang Dewi berbasis *website*. Dengan adanya perangkat lunak sistem *inventory* di UD.

MINANG DEWI ini, maka proses pengolahan data persediaan barang pada UD. MINANG DEWI yang akan datang dapat dilakukan dengan cepat dan akurat, mudah dan lebih baik dari masa sekarang ini.

Tinjauan yang ke-empat dari Apriani, dkk. (2019) dengan judul “Rancang Bangun Sistem Informasi *Inventory* Peralatan Komputer Berbasis *Website* Pada PT Indonesia Toray Synthetic”. Penelitian ini mengungkap masalah Kekurangan dalam hal pendataan khususnya dibidang persediaan peralatan komputer pada bagian informasi sistem PT Indonesia Toray Synthetic dimana untuk melakukan input data hasil pengecekan persediaan barang peralatan komputer masih menggunakan sistem yang konvensional dan lambat karena proses pengecekan persediaan barang masih dicatat pada sebuah *form* yang harus diisi oleh *admin* yang sebelumnya *admin* atau PIC tersebut harus datang ke tempat meja karyawan atau gudang peralatan komputer guna melakukan pengecekan.

Penelitian ini bertujuan agar dapat memberikan manfaat yang positif pada proses kerja penginputan persediaan peralatan komputer serta memberikan hasil atau laporan yang baik dan akurat pada PT Indonesia Toray Synthetic. Pada penelitian ini digunakanlah metode analisa SWOT yang terbagi menjadi 4 (empat) kategori yaitu diantaranya *Strength* (kekuatan), *Weakness* (kelemahan), *Opportunity* (peluang), dan *Threat* (ancaman) yang terdapat pada sistem persediaan peralatan komputer PT. Indonesia Toray Synthetic. Dan metode perancangan sistemnya dilakukan dengan menggunakan *Unified Modeling Language* (UML) serta alat bantu (*tools*) yang digunakan yaitu berupa aplikasi *Visual Paradigm UML* Hasil dari penelitian ini yakni Aplikasi Sistem *Inventory* Peralatan Komputer Berbasis *Website* pada PT Indonesia Torey Synthetic..

Berdasarkan beberapa penelitian lain yang telah dilakukan tersebut, peneliti dapat menyimpulkan bahwa permasalahan yang sering muncul dalam proses bisnis yang terjadi dalam sebuah toko adalah pada bagian penjualan barang, pembuatan laporan, dan permasalahan pada bagian *inventory* atau persediaan.

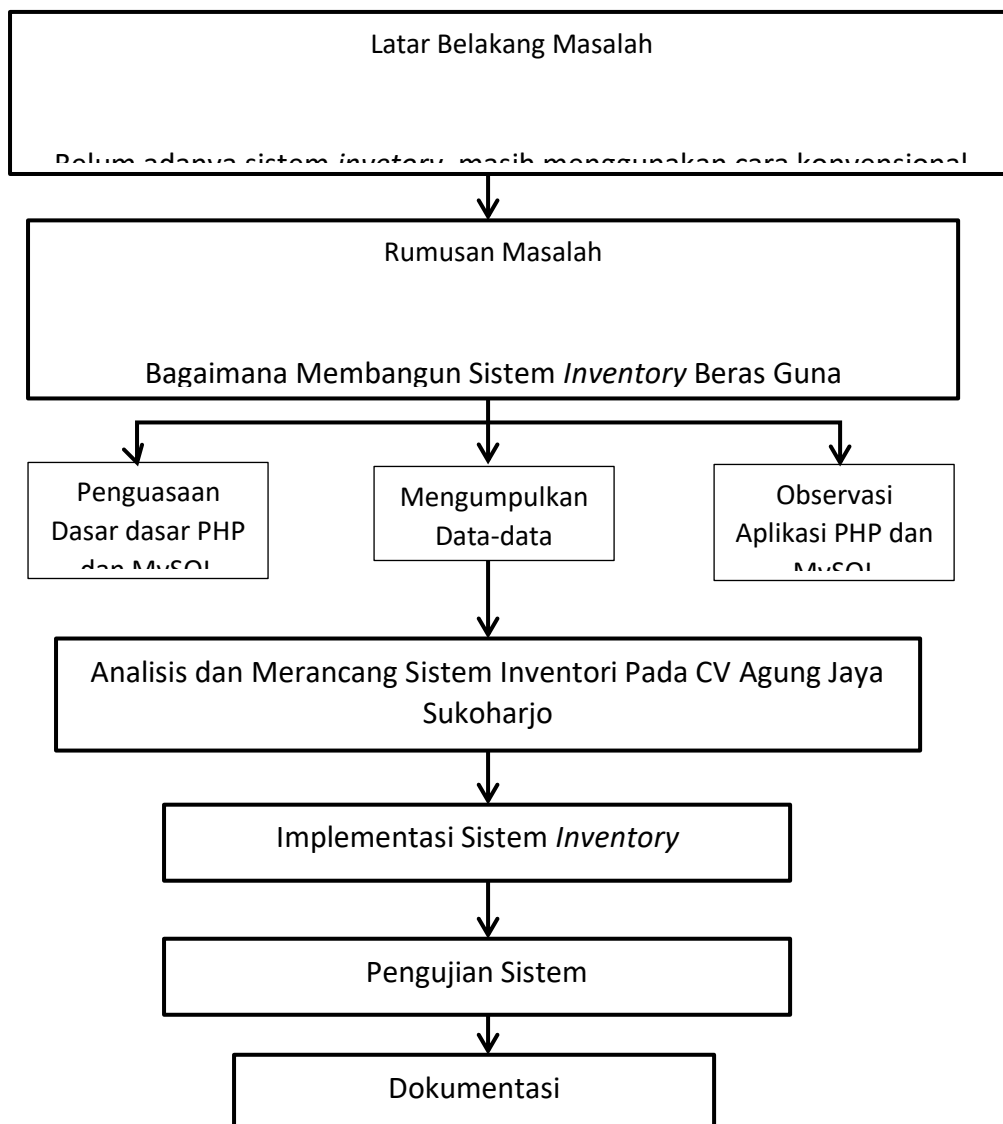
Dari kajian pustaka yang diambil dari beberapa jurnal diatas memiliki persamaan dengan penelitian yang penulis lakukan yaitu mengenai tema yang diteliti yakni mengenai sistem *inventory* barang. Sedangkan perbedaanya yakni

mengenai objek dan tempat yang diteliti serta metode yang digunakan juga ada perbedaan walaupun ada banyak persamaan. Penelitian yang akan dilakukan penulis lebih terfokus pada media manajemen sistem *inventory* beras.

Dengan demikian meskipun kajian pustaka diatas ada beberapa kesamaan dengan penelitian yang penulis akan buat, akan tetapi mengingat subjek, obyek dan tempat penelitian yang dilakukan jauh berbeda, maka peneliti tertarik untuk membuat aplikasi tentang sistem *inventory* beras di CV AGUNG JAYA Sukoharjo.

2.2. Kerangka Pemikiran

Berikut ini adalah tahapan kerangka pemikiran dalam melakukan pembuatan sistem *inventory* pada CV AGUNG JAYA Sukoharjo berbasis *web*. Kerangka pemikiran dapat dilihat pada Gambar 2.2



Gambar 2.1 Kerangka Pemikiran

Berdasarkan kerangka pemikiran penelitian yang telah digambarkan di atas, maka dapat diuraikan pembahasan masing-masing tahap dalam penelitian adalah sebagai berikut:

1. Identifikasi Masalah

Tahap ini merupakan langkah awal penelitian yang menggambarkan masalah apa yang terjadi pada objek penelitian di CV AGUNG JAYA Sukoharjo.

2. Merumuskan Masalah

Setelah melakukan identifikasi masalah yang terjadi di CV Agung Jaya Sukoharjo langkah selanjutnya yakni merumuskan masalah masalah yang terjadi.

3. Pengumpulan Data

Pada tahap ini dilakukan proses pengumpulan data dengan metode wawancara dan observasi untuk melakukan pengamatan dan analisa terhadap inventaris Beras pada CV AGUNG JAYA, sehingga mendapatkan data dan informasi yang dibutuhkan untuk penelitian.

4. Analisa dan Perancangan

Pada tahap ini yakni menganalisa proses yang berjalan saat ini lalu menggambarkan tentang tampilan sistem yang dirancang dan pemodelan sistem yang dilakukan untuk membantu proses perancangan sistem yang baru.

5. Implementasi Sistem

Dalam tahap ini akan dilakukan implementasi sistem yang dilakukan dengan membangun aplikasi *System Inventory* beras di CV AGUNG JAYA Sukoharjo.

6. Pengujian Sistem

Setelah melakukan implementasi sistem langkah selanjutnya yaitu pengujian sistem dengan cara menguji kelayakanya, apakah berjalan sesuai perancangan yang telah dilakukan sebelumnya?

7. Dokumentasi

Tahap ini dilakukan guna melengkapi laporan yang dilakukan selama penelitian

mulai dari awal hingga akhir.

2.3.Landasan Teori

2.3.1 Pengertian Sistem

Menurut Tohari (2017), “Sistem adalah kumpulan atau himpunan dari unsur atau variabel-variabel yang saling berkait, saling berintraksi, dan saling tergantung satu sama lain untuk mencapai tujuan. Selain itu, sistem juga dapat didefinisikan sebagai sekumpulan objek-objek yang saling berelasi dan berinteraksi, serta hubungan antara objek bisa di lihat sebagai satu kesatuan yang dirancang untuk mencapai satu tujuan yang telah di tetapkan.

2.3.2 Elemen Sistem

Menurut Hasanudin (2018), Sistem memiliki kombinasi elemen-elemen yang sama, tetapi susunan dasarnya sama. Elemen-elemen yang terdapat dalam sistem ditandai dengan adanya :

1. Tujuan ini menjadi motivasi yang mengarahkan pada sistem, karena tanpa tujuan yang jelas sistem menjadi tak terarah dan tak terkendali
2. Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan dapat berupa hal-hal berwujud maupun yang tidak berwujud. Masukan berwujud adalah bahan mentah, sedangkan yang tidak berwujud adalah informasi. Proses-proses merupakan elemen yang bertugas melakukan perubahan atau transformasi dari masukan / data menjadi keluaran / informasi yang berguna dan lebih bernilai.
3. Keluaran (*output*) merupakan hasil dari input yang sudah dilakukan pemrosesan sistem dan keluaran dapat menjadi masukan untuk sub sistem lain.
4. Batasan (*boundary*) sistem adalah pemisah antara sistem dan daerah diluar sistem. Selain itu juga sebagai batasan–batasan dari tujuan yang akan dicapai oleh sistem. Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.
5. Umpan balik ini digunakan untuk mengendalikan masukan maupun proses. Umpan balik juga bertugas mengevaluasi bagian dari output yang dikeluarkan.

Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan.

6. Lingkungan Lingkungan adalah segala sesuatu yang berada diluar sistem.

2.3.3 Pengertian *Inventory*

Secara teknis, *inventory* adalah suatu teknik yang berkaitan dengan penetapan terhadap besarnya persediaan bahan yang harus diadakan untuk menjamin kelancaran dalam kegiatan operasi produksi, serta menetapkan jadwal pengadaan dan jumlah pemesanan barang yang seharusnya dilakukan oleh perusahaan. Penetapan jadwal dan jumlah pemesanan yang harus dipesan merupakan pernyataan dasar yang harus terjawab dalam pengendalian persediaan (Putra, Anwariningsih, dkk., 2013)

2.3.4 Pengertian Sistem *Inventory*

Assauri (2016), Sistem *inventory* adalah sekumpulan kebijakan dan pengendalian, yang memonitor tingkat *inventory*, dan menentukan tingkat mana yang harus dijaga, bila stok harus diisi kembali dan berapa banyak yang harus dipesan.

Sistem *inventory* akan memberikan kemungkinan struktur organisasi dan kebijakan operasi produksi, untuk menjaga dan mengawasi barang-barang untuk distok. Dengan sistem *inventory* ini, diharapkan manajemen dapat bertanggung jawab terhadap pemesanan dan penerimaan barang yang dipesan. Hal ini dapat dilakukan dengan mengawasi waktu penempatan pesanan, dan menjaga atau mengawasi jalannya jalur dari apa yang dipesan, serta berapa banyak barang yang dipesan dan dari siapa *vendor*-nya. (Assauri, 2016)

2.3.5 Sistem Persediaan

Persediaan dapat diartikan sebagai sumber daya mengganggu yang menunggu proses lebih lanjut berupa kegiatan produksi pada sistem manufaktur, kegiatan pemasaran pada sistem distribusi ataupun kegiatan konsumsi pada setiap rumah tangga. (Nugraha, dkk. 2018)

2.3.6 Gudang

Gudang adalah tempat penyimpanan sementara dan pengambilan persediaan untuk mendukung kegiatan operasi bagi proses operasi berikutnya, ke lokasi distribusi, atau kepada konsumen akhir. Gudang berfungsi untuk :

1. Menyimpan barang untuk sementara waktu sambil menunggu giliran untuk diproses.
2. Memantau pergerakan dan status barang.
3. Meminimumkan biaya pergerakan barang, peralatan, dan karyawan.
4. Menyediakan media komunikasi dengan konsumen mengenai barang.
5. Titik penyeimbang aliran *inventory* dan barang.

Jika *inventory* berlebihan, diletakkan di gudang. Setelah itu, bagian gudang harus memastikan bahwa *inventory* / barang disimpan baik, terjaga kondisinya, dan tercatat statusnya sehingga tidak ada modal perusahaan yang hilang akibat kesalahan pengawasan di gudang. (Martono, 2015)

2.3.7 Website

Menurut Ermita (2013), *World Wide Web* atau lebih sering dikenal sebagai *web* adalah layanan internet yang paling banyak memiliki tampilan grafis dan kemampuan *link* yang sangat bagus. Keistimewaan inilah yang telah menjadikan *Web* sebagai *service* yang paling cepat pertumbuhannya. *Web* mengizinkan pemberian *highlight* (penyorotan atau penggaris bawahan) pada kata-kata atau gambar dalam sebuah dokumen untuk menghubungkan atau menunjuk ke media lain seperti dokumen, *frase*, *movie 17 clip*, atau *file* suara. *Web* dapat menghubungkan dari sebarang tempat dalam sebuah dokumen atau gambar ke sebarang tempat di dokumen lain. Dengan sebuah *browser* yang memiliki *Graphical User Interface* (GUI), *link-link* dapat dihubungkan ke tujuannya dengan menunjuk *link* tersebut dengan *mouse* dan menekannya.

2.3.8 Database

Database adalah sekumpulan data atau informasi yang teratur berdasarkan kriteria tertentu yang saling berhubungan. *Database* merupakan salah satu

komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi, menentukan kualitas informasi (akurat, tepat pada waktunya dan relevan). Informasi dapat di katakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya serta *database* mampu mengurangi pemborosan tempat simpanan luar. (Fahrissal, dkk. 2019)

2.3.9 MySQL

Menurut Ermita (2013), MySQL merupakan salah satu perangkat lunak sistem pengelola basis data DBMS (*Data Base Management System*). MySQL merupakan sebuah hubungan *Data Base Management System* (DBMS) yang membantu sebuah model data yang terdiri atas kumpulan hubungan nama (*named relation*). *Database* MySQL adalah salah satu *database* yang *open source*. *Database* ini banyak dipasangkan dengan *script* PHP. Penyebab utama MySQL begitu populer di kalangan *Web* karena cocok bekerja di lingkungan tersebut, selain itu karena :

1. MySQL tersedia di berbagai platform dan kompatibel dalam berbagai sistem operasi.
2. Fitur-fitur yang dimiliki MySQL merupakan fitur-fitur yang biasanya banyak dibutuhkan dalam aplikasi *web*.
3. MySQL memiliki *overhead* koneksi yang rendah, sehingga kecepatan koneksi relatif tinggi.

2.3.10 Pengertian *Personal Home Page* (PHP)

PHP adalah bahasa *scripting* yang menyatu dengan HTML dan dijalankan pada *serverside*, Artinya semua sintaks yang diberikan akan sepenuhnya dijalankan pada *server* sedang yang dikirim ke *browser* hanya hasilnya saja. Ketika seorang pengguna *internet* membuka situs yang menggunakan fasilitas *serverside scripting* PHP, maka terlebih dahulu *server* yang bersangkutan akan memproses semua perintah PHP diserver lalu mengirimkan hasilnya dalam format TML ke *web server* pengguna internet tadi. Sehingga kode asli yang ditulis dengan PHP tidak terlihat di browser pengguna. (Fauzi, dkk. 2015)

2.3.11 *Webqual*

Ada empat definisi yang ada pada webqual menurut (Elmayati & Wulandari, 2018) yaitu sebagai berikut :

1. Kualitas kegunaan (*usability*) meliputi kemudahan website untuk dipelajari, kemudahan untuk dimengerti, kemudahan untuk ditelusuri, kemudahan untuk digunakan, kemenarikan website, interface yang menyenangkan, memiliki kompetensi yang baik dan memberikan pengalaman baru yang menyenangkan.
2. Kualitas informasi (*information quality*) meliputi informasi yang akurat, informasi yang dapat dipercaya, informasi yang up to date, informasi sesuai dengan topik bahasan, kemudahan informasi untuk dimengerti, kedetailan informasi dan informasi yang disajikan dalam format desain yang sesuai.
3. Kualitas interaksi pelayanan (*service interaction quality*) meliputi kemampuan memberikan rasa aman saat transaksi, memiliki reputasi yang bagus, memudahkan komunikasi, menciptakan perasaan emosional yang lebih personal, memiliki kepercayaan dalam memberikan informasi pribadi, mampu menciptakan komunitas yang spesifik, memberikan keyakinan bahwa janji yang disampaikan akan ditepati.
4. Keseluruhan (*Overall*) adalah variabel yang menyatakan keseluruhan atau hal yang belum terdapat pada tiga variabel yang sudah ada.

Dari pengertian tersebut dapat disimpulkan bahwa pengertian dari webqual adalah suatu metode yang digunakan untuk mengukur suatu kualitas website, dengan menggunakan 4 variabel penilaian yaitu *usability*, *information quality*, *service interaction quality*, *overall*.

2.3.12 *Unified Modelling Language (UML)*

“*Unified Modeling Language*” (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”. Windu & Grace, (2013)

Unified Modeling Language (UML) adalah sebuah bahasa yang berdasarkan


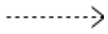




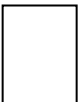



grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem *software*. Suendri (2018)

UML (*Unified Modelling Language*) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi objek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain. (Munawar, 2018)

2.3.13 Use Case Diagram

Use case adalah deskripsi fungsi dari sebuah sistem dan *perspektif* pengguna. *Use case* bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. (Munawar, 2018)

Tabel 2.1 *Use Case Diagram*





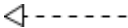
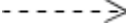

GAMBAR	NAMA	KETERANGAN
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
	<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
	<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
	<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2.3.14 *Class Diagram*

Class diagram adalah *diagram* statis. Ini mewakili pandangan statis dari suatu aplikasi. *Class diagram* tidak hanya digunakan untuk memvisualisasikan, menggambarkan, dan mendokumentasikan berbagai aspek sistem tetapi juga untuk membangun kode eksekusi (*executable code*) dari aplikasi perangkat lunak.

(Munawar, 2018)

Tabel 2.2 *Class Diagram*


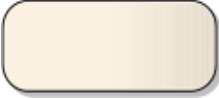



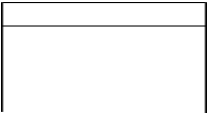
GAMBAR	NAMA	KETERANGAN
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor
	<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
	<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
	<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

2.3.15 *Activity Diagram*

Activity Diagram adalah bagian penting dari UML, yang menggambarkan aspek dinamis dari sistem. Logika prosedural, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah dideskripsikan dalam *activity diagram*. *Activity diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. (Munawar, 2018)

Tabel 2.3 *Activity Diagram*

SIMBOL	NAMA	KETERANGAN
--------	------	------------

	Status Awal/ <i>Initial</i>	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas/ <i>Activity</i>	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan/ <i>Decision</i>	Asosiasi percabangan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	Penggabungan/ <i>Join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas lebih dari satu.
	Status Akhir/ <i>Final</i>	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status satu.
	<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.


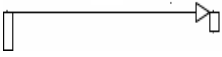

2.3.16 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah *scenario*. Diagram ini menunjukkan sejumlah contoh obyek dan *message* (pesan) yang diletakan diantara obyek-obyek ini di dalam *use case*

Komponen utama *sequence diagram* terdiri atas obyek yang dituliskan dengan kotak segiempat bernama. *Message* diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan *progress vertical*. (Munawar 2018)

Tabel 2.4 *Sequence Diagram*

GAMBAR	NAMA	KETERANGAN
--------	------	------------

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

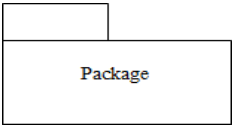
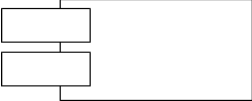
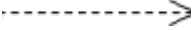


2.3.17 Component Diagram

Component software adalah bagian fisik dari sebuah sistem, karena menetap di komputer, bukan di benak para analis. Komponen bisa berupa tabel, *file data*, *file exe*, dll (*dynamic link library*), dokumen dan lain sebagainya.

Component adalah implementasi *software* dari sebuah *class*. *Class* mewakili abstraksi dari serangkaian *attribute* dan *operaton*. Hal terpenting yang perlu diingat tentang *class* dan *component* adalah sebuah *component* bisa jadi merupakan implementasi dari lebih dari sebuah *class*. (Munawar, 2018)

Tabel 2.5 *Component Diagram*

SIMBOL	NAMA	KETERANGAN
--------	------	------------

	<i>Package</i>	<i>Package</i> merupakan sebuah bungkus dari satu atau lebih komponen.
	Komponen/ <i>Component</i>	Komponen merupakan komponen sistem.
	Ketergantungan/ <i>Dependency</i>	Ketergantungan atau <i>dependency</i> atau kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai.
	Anatarmuka/ <i>Interface</i>	Antar muka atau <i>interface</i> merupakan antarmuka sama dengan interface pada pemrograman berorientasi objek, yaitu sebagai antar muka komponen agar tidak mengakses langsung komponen.
	<i>Link</i>	Menggambarkan relasi antar komponen.

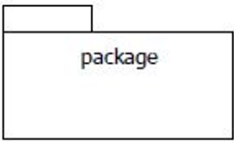
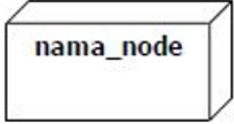


2.3.18 Deployment Diagram

Deployment Diagram menunjukkan tata letak sebuah sistem secara fisik, menampakan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*.

Bagian utama hardware atau perangkat keras adalah *node*; yaitu nama umum untuk semua jenis sumber komputasi. Ada dua tipe *node* yang mungkin, *Processor* adalah *node* yang bisa mengeksekusi sebuah *component*, sedangkan *device* tidak. *Device* adalah perangkat keras (seperti printer atau monitor) tipikalnya menjadi interface dengan dunia luar. ((Munawar, 2018)

Tabel 2.6 *Deployment Diagram*

SIMBOL	NAMA	KETERANGAN
--------	------	------------




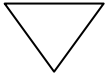
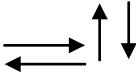
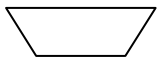
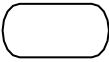
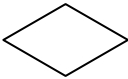


	<i>Package</i>	Package merupakan sebuah bungkus dari satu atau lebih node
	<i>Node</i>	Biasanya mengacu pada perangkat keras (hardware) perangkat lunak yang tidak dibuat sendiri (software), jika di dalam node disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen
	<i>Dependency</i>	Kebergantungan antar node, arah panah mengarah pada node yang dipakai
	<i>Link</i>	Relasi anatar node

2.3.19 Flow Chart

Flowchart adalah bagan -bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma. (Ladjamudin, 2013)

Tabel 2.7 *Flow Chart*

SIMBOL	NAMA	KETERANGAN
--------	------	------------

	Penghubung	Simbol untuk keluar/masuk atau proses dalam lembar atau halaman lain.
	<i>Input dan Output</i>	Simbol yang menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.
	Dokumen	Simbol yang menyatakan <i>input</i> berasal dari dokumen dalam bentuk kertas atau output dicetak di kertas.
	<i>On Line Storage</i>	Simbol yang menunjukkan bahwa data di dalam simbol ini akan di simpan.
	Garis Alir	Digunakan untuk menunjukkan arah selanjutnya yang akan dituju dari simbol-simbol dan <i>flowchart</i> .
	Manual	Simbol yang menunjukkan pengolahan yang tidak dilakukan oleh komputer.
	Terminal	Simbol yang menunjukkan untuk permulaan atau akhir suatu sistem.
	Kondisi	Simbol keputusan yang menunjukkan kondisi.
	Proses	Simbol yang menunjukkan pengolahan dilakukan oleh komputer.
	Penghubung	Penghubung Simbol untuk keluar/masuk prosedur atau proses dalam lembar atau halaman yang masih sama.