

BAB II

TINJAUAN PUSTAKA

I.1. Penelitian Terkait

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

Penelitian yang dilakukan oleh Afan Suriyana, Lukman Junaed (2020) yaitu menggunakan metode *Rapid Application Development* sebagai sarana dalam pengembangan sistem informasi penjualan. Dimana aplikasinya adalah *e-commerce* yang dapat dijalankan secara efisien. Data yang menjadi inputannya adalah data produk yang tersedia ditoko sehingga menghasilkan suatu sistem informasi mengenai produk yang diakses secara *online*. Dengan adanya sistem penjualan *online e-commerce* dapat membantu menyelesaikan masalah yang ada dan guna mencapai efektifitas dan efisiensi kinerja pada Toko Cindyah Collection.

Penelitian yang dilakukan oleh Eka Puspita Sari, et al (2020) yaitu menerapkan metode *Waterfall* dalam pengembangan sistem penjualan pakaian. Dimana aplikasi *website e-commerce* data yang menjadi inputannya adalah data baju yang tersedia ditoko. Dengan adanya sistem informasi penjualan pakaian ini, akan mempermudah karyawan dalam melakukan proses transaksi penjualan dan pengolahan data penjualan sehingga mengurangi resiko kesalahan dalam pencatatan penjualan. Mampu Meningkatkan penjualan yang ada karena dapat meningkatkan

kepuasan pelanggan dan meningkatkan laba penjualan karena sistem informasi penjualan ini mampu menghemat biaya dalam proses penjualan.

Penelitian yang dilakukan oleh Handini Widyastuti, et al (2020) yaitu menggunakan metode *Waterfall* dalam pengembangan sistem informasi penjualan baju. Dimana aplikasinya adalah *website e-commerce*, data yang menjadi inputannya adalah data produk baju yang tersedia ditoko. Perancangan *website* ini adalah sarana informasi berbasis *online* yang bisa di akses oleh siapapun juga. Dengan adanya informasi penjualan produk Baju berbasis *web* ini dapat memperluas daerah jangkauan bisnis.

Penelitian yang dilakukan oleh Mira (2020) yaitu pada toko RA-Collection yang bergerak dibidang sebuah bisnis penjualan pakaian. Dalam pengelolaan informasinya masih secara manual. Dengan menggunakan sistem berbasis *web* maka akan memudahkan dalam melakukan promosi dengan kapan saja, penyampaian informasi akan lebih mudah dan cepat. Sistem penjualan *e-commerce* ini menggunakan metode (*RAD*), dengan inputan data produk pakaian dan transaksi produk. Sehingga penelitian ini menghasilkan suatu sistem untuk mempermudah pelanggan dalam proses transaksi pembelian.

Penelitian yang dilakukan oleh Nanda Ancas Fitra Riyadi (2020) yaitu menggunakan metode *Rapid Application Development* sebagai sarana dalam pengembangan sistem informasi penjualan pada Toko Rindu se Windu Collection yang merupakan suatu usaha yang bergerak di bidang penjualan produk baju mulai dari baju anak-anak sampai baju dewasa. Masalah yang ditemui kesulitan dalam memperkenalkan dan memasarkan produknya ke masyarakat luas. Dengan adanya rancangan *web e-commerce* akan memberi kenyamanan pada konsumen.

Penelitian yang dilakukan oleh Rizky Adi Pratama (2019) yaitu menggunakan metode *Rapid Application Development* sebagai sarana dalam pengembangan sistem informasi dengan tahapan Pemodelan Bisnis, Pemodelan Data, Pemodelan Proses, Pemodelan Aplikasi, Pengujian dan Pergantian. Permasalahan yang dihadapi Labirin Adv adalah media promosi hanya menggunakan spanduk, penjualan masih sebatas *offline*, penyimpanan dan pencatatan laporan masih ditulis pada buku catatan. Aplikasi berbasis *web* dimana dapat diakses secara *online*. Dengan adanya sebuah *e-commerce* penjualan berbasis *web* yang telah dirancang akan membantu dalam memperkenalkan produk baru kepada pelanggan, memudahkan proses penjualan yang dilakukan secara *online* sehingga pelanggan tidak perlu selalu datang ke toko, serta memudahkan dalam mengolah data dan pencetakan laporan penjualan yang tidak lagi memerlukan buku catatan.

Penelitian yang dilakukan oleh Wahid, B. A (2019) yaitu menerapkan metode *Rapid Application Development* dalam menghasilkan sistem penjualan Fashion Distro secara *online*. Dalam penjualannya masih mengandalkan media interaksi dengan pengguna secara langsung dan juga masih berpusat pada wilayah tertentu. Sehingga aplikasi berbasis *online* ini dapat diakses kapan saja dan dapat memperluas pangsa pasarnya.

Menurut penelitian yang dilakukan oleh Anik Andriani, Esti Qurniati (2018) yaitu sistem informasi penjualan pada toko *online* berbasis *web* menggunakan metode *Rapid Application Development*, dimana aplikasi ini dapat menghasilkan suatu sistem informasi yang diperlukan untuk memenuhi kebutuhan konsumen dalam hal kemudahan, kecepatan, dan kenyamanan dalam bertransaksi. Sehingga dalam pemasaran, promosi, maupun dalam penjualannya menjadi luas jangkauan

pasar untuk penjualan. Penelitian ini menggunakan *RAD* untuk mengembangkan sebuah sistem dikarenakan tahapan tahapannya yang cepat dan sistematis dengan hasil yang sesuai seperti yang diharapkan.

Penelitian yang dilakukan oleh Moh Afdhol Rizal, Titik Misriati (2018) yaitu menggunakan metode *Waterfall* dalam pengembangan sistem informasi penjualan pakaian. Melalui *website* penjualan pada Toko Uj Outlet, pengunjung dapat mengetahui macam-macam produk pakaian bermerek terbaru dan dapat melakukan transaksi penjualan dengan konsumen tanpa harus mengunjungi toko secara langsung.

Penelitian yang dilakukan oleh Richard Lie (2017) yaitu Sistem Informasi Penjualan Pakaian Berbasis *Web* pada Toko Sweet Batam. Penelitian tersebut di latar belakang oleh banyaknya perusahaan dagang dan jasa yang mulai menggunakan *e-commerce* sebagai media penjualan *online*. Penelitian ini menggunakan aplikasi CMS PrestaShop sebagai media pembuatan *e-commerce*. Perancangan dimaksudkan untuk mempermudah pengelolaan, penjualan dan promosi juga mempermudah pembeli. Dalam pembuatannya menerapkan teknik pembangunan secara *waterfall*.

Sedangkan penelitian yang dilakukan oleh penulis yaitu Rancang Bangun *Web E-commerce* Produk Pakaian Dengan Metode *Rapid Application Development (RAD)* Pada Toko Idola Fashion menghasilkan aplikasi *e-commerce* berbasis *web* yang dapat diakses secara *online* untuk mempermudah toko Idola Fashion dalam mempromosikan produk pakaian yang diperjual belikan dan mempermudah pembeli dalam melihat produk pakaian yang ditawarkan serta dalam melakukan pesanan secara *online*. Sistem ini dilengkapi dengan validasi *update* status pesanan yang bisa membantu pembeli serta *admin* dalam informasi status produk yang

dipesan, serta memudahkan pihak toko Idola Fashion dalam mengolah data, pencetakan laporan penjualan dan laporan produk. Penelitian ini menggunakan metode *rapid application development (RAD)* dengan tujuan mempermudah pengembangan sistem yang akan dirancang dikarenakan tahapannya yang cepat dan sistematis dengan hasil yang sesuai seperti yang diharapkan.

II.2. Landasan Teori

Landasan teori peneliti kutip dari beberapa teori yang berkaitan dengan penelitian ini yaitu :

II.2.1. Rancang Bangun

Rancang Bangun adalah proses pembangunan sistem untuk menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun hanya sebagian (Yuntari Purba Sari, 2017).

II.2.2. Website

Website adalah sebuah komponen yang terdiri dari teks, gambar, suara, animasi sehingga menjadi media informasi yang menarik dikunjungi oleh orang lain (Zufria & Azhari, 2017:52).

II.2.4. E-commerce

E-commerce merupakan proses pembelian dan penjualan melalui Internet atau setiap transaksi bisnis yang melibatkan perpindahan kepemilikan atas barang atau jasa melalui jaringan komputer. Meskipun populer, definisi tersebut tidak cukup komperhensif untuk dapat mendeskripsikan perkembangan dalam fenomena

bisnis tersebut. Definisi yang lebih lengkap: *E-Commerce* merupakan penggunaan komunikasi elektronik dan teknologi pengolahan informasi digital dalam transaksi bisnis untuk menciptakan, mengubah dan mendefinisikan kembali hubungan baru diantara organisasi dan/atau antara organisasi dengan individu (Silvia Sofian, 53 ; 2017).

III.2.4. Produk

Produk adalah segala sesuatu yang dapat ditawarkan ke pasar untuk mendapatkan perhatian, akuisisi, penggunaan, atau konsumsi yang dapat memuaskan keinginan atau kebutuhan (Budi Prasetya Sjawal, et al 2020).

II.2.5. Pakaian

Pakaian merupakan salah satu kebutuhan pokok manusia dalam menjalani hidup. Kegunaan pakaian pada masa sekarang ini sudah mulai begeser dari fungsi utamanya. Dulu pakaian berfungsi sebagai penutup tubuh, saat ini fungsi pakaian lebih dari sekedar penutup tubuh, namun juga sebagai sarana bergaya dan mengikuti perkembangan jaman, penggunaan pakaian yang seperti itu dengan diikuti adanya perkembangan ragam jenis pakaian yang semakin banyak disebut sebagai *fashion* (Rinda Yuda Wardana & Argyo Demartoto, 2017).

II.2.6. Toko

Toko adalah bangunan yang digunakan sebagai tempat berusaha, tempat menjual barang dan jasa yang menghasilkan pendapatan bagi pemiliknya. Pengertian toko itu sendiri adalah salah satu *public space* yang dipergunakan

sebagai tempat berbisnis yang sifatnya sendiri adalah sebagai aktifitas memajang, menyimpan dan menjual, juga sebagai area pertemuan antara pengusaha dengan konsumen yang mampu membuat keuntungan bagi pengelola maupun pemiliknya (Rudi Haryanto, et al, 2022).

II.2.7. Basis Data (*Database*)

Basis data merupakan kumpulan dari data yang saling berhubungan antara satu dengan yang lain. Basis data atau *database* merupakan salah satu komponen yang penting dalam sistem informasi, karena berfungsi sebagai basis penyedia informasi bagi pemakainya, Sistem basis data adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan dengan yang lainnya dan untuk membuatnya tersedia beberapa aplikasi yang bermacam-macam dalam suatu sistem organisasi. Sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi atau perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan. (Priyo Sutopo, et al. 2018).

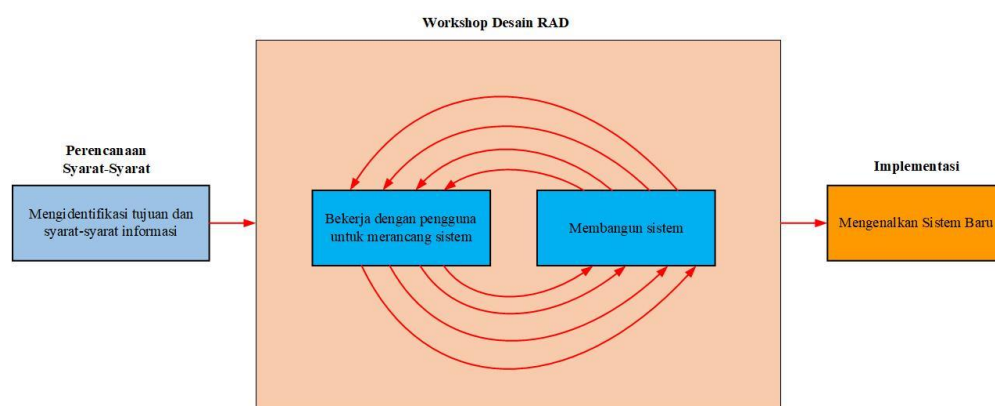
II.2.8. PHP (*Hypertext Preprocessor*)

PHP merupakan bahasa pemrograman yang digunakan untuk membuat *website* dinamis dan interaktif. Dinamis artinya *website* bisa berubah-ubah tampilan dan kontennya sesuai dengan 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) (Jubilee Enterprise, 2018).

II.2.9. RAD (*Rapid Application Development*)

RAD merupakan model proses perangkat lunak yang menekankan pada daur pengembangan hidup yang singkat. *RAD* merupakan versi adaptasi cepat dari model *waterfall*, dengan menggunakan pendekatan konstruksi komponen (Putri & Effendi ; 2018). Dimana metode *RAD* ini sangat cocok digunakan dalam pengembangan aplikasi yang memiliki waktu terbatas atau karena keperluan yang mendesak, sehingga penggunaan metode *RAD* dapat mengefektifkan dalam waktu pengerjaan pengembangan sebuah aplikasi. Tahapan dari metode *RAD* dapat dilihat pada Gambar II.1.



Gambar II.1. Siklus *RAD*

(Sumber : Putri & Effendi ; 2018)

Adapun tahapan dari *RAD (Rapid Application Development)* yaitu :

1. Perencanaan Syarat-syarat

Pada tahap ini diketahui apa saja yang menjadi kebutuhan sistem yaitu dengan mengidentifikasi kebutuhan informasi dan masalah yang dihadapi untuk menentukan tujuan, batasan-batasan sistem, kendala dan juga alternatif pemecahan masalah. Analisis digunakan untuk mengetahui perilaku sistem dan juga untuk mengetahui aktivitas apa saja yang ada dalam sistem tersebut.

2. *Workshop Desain RAD*

Pada tahap ini mengidentifikasi solusi alternatif dan memilih solusi yang terbaik. Kemudian membuat desain proses bisnis dan desain pemrograman untuk data-data yang telah didapatkan dan dimodelkan dalam arsitektur sistem informasi. *Tools* yang digunakan dalam pemodelan sistem biasanya menggunakan *Unified Modelling Language (UML)*.

3. *Implementasi*

Pada tahap ini merupakan tahapan pemasangan sistem yang dilanjutkan dengan tahap pengujian sistem. Untuk pengujian unit menggunakan *Black Box Testing* dikarenakan metode pengujian unit ini memfokuskan pada fungsionalitas dari sistem yang dibangun. Dan untuk mengetahui apakah sistem telah memenuhi kebutuhan pada analisa kebutuhan atau tidak maka *Black Box Testing* merupakan yang paling tepat untuk mengetahui hal tersebut.

II.2.10. *MySQL*

MySQL merupakan server yang melayani *Database*. Untuk membuat dan mengolah *Database*, kita dapat mempelajari pemrograman khusus yang disebut *query* (perintah) *SQL*. *Database* sendiri dibutuhkan jika kita ingin menginput data dari *user* menggunakan *form HTML* untuk kemudian diolah *PHP* agar bisa disimpan ke dalam *Database MySQL* (Jubilee Enterprise, 2018:2).

II.2.11. *UML (Unified Modeling language)*

Menurut Sukanto dan Shalahuddin (2018:13), “*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* hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan *UML* tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya *UML* paling banyak digunakan pada metodologi berorientasi objek. *UML* ini terdiri dari 13 macam *diagram* namun hanya beberapa *diagram* yang digunakan, diantaranya :

II.2.11.1. *Use Case Diagram*

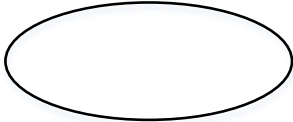
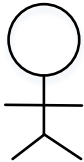


Sukanto dan Shalahuddin (2018:155), *Use Case* atau *diagram Use Case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk

mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Aktor adalah orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi meskipun simbol aktor adalah gambar dari orang, aktor tidak tentu seseorang. *Use case* adalah fungsionalitas yang disediakan oleh sistem sebagai unit yang bertukar pesan antar unit atau aktor.

Berikut adalah simbol-simbol yang ada pada *diagram* use case:

Tabel II.1. Use Case Diagram

Gambar	Keterangan
	Fungsionalitas yang disediakan oleh sistem sebagai unit-unit yang dapat saling bertukar pesan antar unit atau aktor, umumnya dapat dinyatakan dengan kata kerja di awal frase nama <i>use case</i> .
	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, meskipun simbol dari aktor merupakan gambar orang, namun aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan suatu kata benda di awal frase nama aktor.
	Komunikasi antara aktor dengan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu merupakan termasuk fungsi yang lebih umum dari yang lainnya.

<<include>>	<i>Include</i> berarti <i>use case</i> tambahan yang akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan dapat berjalan sebelum <i>use case</i> tambahan dijalankan.
<<extends>>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri meski tanpa <i>use case</i> tambahan tersebut, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan. Umumnya <i>use case</i> yang menjadi <i>extend-nya</i> yaitu jenis yang sama dengan <i>use case</i> yang menjadi induknya.

Sumber : Sukamto dan Shalahuddin (2018:156-158)

II.2.11.2. Activity Diagram



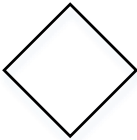
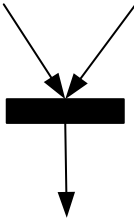

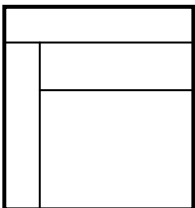
Sukamto dan Shalahuddin (2018:161), *Diagram* aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. *diagram* aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem”. *Diagram* aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut:

- Desain proses bisnis di mana setiap urutan tugas dijelaskan adalah proses bisnis sistem yang ditentukan.
- Tampilan sistem / urutan antarmuka pengguna atau pengelompokan di mana setiap tugas dianggap memiliki desain antarmuka.

- Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kamus ujinya.
- Desain menu yang ditampilkan dalam perangkat lunak.

Berikut adalah simbol-simbol yang ada pada *diagram* aktivitas:

Tabel II.2. Activity Diagram

Gambar	Keterangan
	Status awal aktivitas sistem ialah sebuah <i>diagram</i> aktivitas yang memiliki sebuah status awal
	Aktivitas yang dilakukan oleh sistem, aktivitas ini biasanya diawali dengan kata kerja.
	Asosiasi percabangan dimana jika ada sebuah pilihan aktivitas lebih dari pada satu.
	Sebuah Asosiasi penggabungan dimana lebih dari satu aktivitas yang dapat digabungkan menjadi satu.
	Status akhir yang mana dilakukan oleh sistem, merupakan <i>diagram</i> aktivitas yang memiliki sebuah status akhir.
Swimlane 	Memisahkan sebuah organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

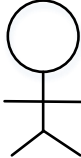

Sumber : Sukamto dan Shalahuddin (2018:156-158)



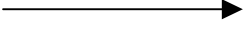
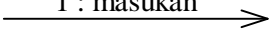
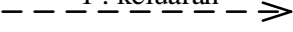
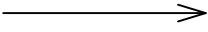
II.2.11.3. *Sequence Diagram*

Sukamto dan Shalahuddin (2018:165), *Diagram sequence* 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. Membuat *diagram sequence* juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya *diagram sequence* yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup dalam *diagram sequence* sehingga semakin banyak *use case* yang didefinisikan maka *diagram sequence* yang harus dibuat juga semakin banyak”.

Berikut adalah simbol-simbol yang ada pada *sequence diagram*:

Tabel II.3. *Sequence Diagram*

Gambar	Keterangan
<p>Aktor</p> 	<p>Menggambarkan orang, proses yang saling berinteraksi dengan sistem informasi yang berada di luar sistem informasi yang akan dibuat, walaupun simbol aktor ialah citra seseorang, aktor tersebut belum pasti orang, menggunakan nama.</p>
<p>Life Line</p> 	<p>Menyatakan kehidupan dari suatu objek</p>

<p>Objek</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama_objek:nama_kelas</div>	Menyatakan objek yang dapat saling berinteraksi pesan.
<p>Waktu Aktif</p> 	Yang menyatakan objek dalam keadaan aktif dan saling berinteraksi, semua yang terhubung dengan waktu aktif ini ialah sebuah tahapan yang dilakukan didalamnya.
<p>Create</p> 	Yang menyatakan suatu objek dan membuat objek yang lain, arah panah mengarah terhadap objek yang dibuat.
<p>Pesan tipe call 1 : nama_metode()</p> 	Yang menjelaskan suatu objek memanggil operasi/metode yang jelas ada pada objek lain atau dirinya sendiri. Arah panah menuju pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode sehingga operasi/metode yang dipanggil harus ada pada <i>diagram</i> kelas sesuai dengan kelas objek yang berinteraksi.
<p>Pesan tipe send 1 : masukan</p> 	Menjelaskan bahwa sebuah objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah terhadap objek yang dikirimin.
<p>Pesan tipe return</p> <p>1 : keluaran</p> 	Menggambarkan bahwa suatu objek yang telah menjalankan suatu operasi atau metode yang menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah kepada objek yang menerima kembalian tersebut.
<p>Pesan tipe destroy</p> 	Yang menjelaskan suatu objek mengakhiri hidup objek lain, arah panah mengarah pada objek yang akan diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .

Sumber : Sukamto dan Shalahuddin (2018:156-158)

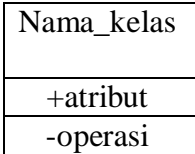
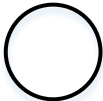


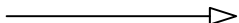


II.2.11.4. Class Diagram


Sukamto dan Shalahuddin (2018:141), *Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. *Class Diagram* memiliki apa yang disebut atribut model dan metode atau operasi;

- *Atribut* adalah *variabel* yang dimiliki oleh suatu kelas.
- Operasi atau metode adalah fungsi yang dimiliki oleh suatu kelas.

Berikut adalah simbol-simbol yang ada pada *diagram* kelas:

Tabel II.4. Class Diagram

Gambar	Keterangan
<p>Kelas</p> 	Menggambarkan kelas pada sebuah struktur <i>system</i> .
<p>Antarmuka / interfaces</p>  <p>Nama_interface</p>	Konsep <i>interface</i> sama dengan yang ada dalam pemrograman berorientasi objek.
<p>Asosiasi / association</p> 	Relasi antar kelas dengan makna umum, asosiasi umumnya juga disertai dengan suatu <i>multiplicity</i> .
<p>Asosiasi berarah /directed association</p> 	Merupakan relasi antar kelas dengan makna kelas yang satu digunakan sama kelas yang lain, asosiasi umumnya juga disertai dengan <i>multiplicity</i> .
<p>Generalisasi</p> 	Merupakan suatu relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).
<p>Kebergantungan/dependency</p> 	Merupakan suatu relasi antar kelas dengan makna kebergantungan antar kelas.
<p>Agregasi/aggregation</p> 	Sebuah relasi antar kelas dengan maksud makna

	semua-bagian (<i>whole part</i>).
---	-------------------------------------

Sumber : Sukamto dan Shalahuddin (2018:156-158)