

BAB II

TINJUAN PUSTAKA

2.1 Penelitian Terkait

Dalam mengembangkan Sistem Informasi Penjualan dan *Inventory* di Toko Kelontong Narjo berbasis web, sejumlah penelitian telah memberikan wawasan yang berharga. Anusha et al. (2023) menggunakan pendekatan pengembangan perangkat lunak yang tidak disebutkan secara spesifik untuk meningkatkan efisiensi dan kemudahan dalam manajemen inventaris. Toko kelontong menemukan bahwa implementasi sistem ini berhasil menghasilkan proses yang lebih otomatis dan meminimalkan keterlibatan manual dalam pengelolaan stok barang [7]. Sebaliknya, Soegoto dan Palalungan (2020) melakukan studi deskriptif dengan mengumpulkan data melalui literatur terkait, menyoroti bahwa sistem inventaris *online* berbasis web membawa kemudahan dalam pengelolaan barang dan meningkatkan efisiensi kerja di perusahaan [8].

Studi tinjauan sistematis literatur oleh Misahuaman, Daza, dan Zavaleta (2021) mengeksplorasi sistem kontrol inventaris berbasis web dan menemukan bahwa pendekatan ini dapat memberikan pemahaman yang lebih baik tentang pentingnya dan manfaat sistem berbasis web dalam kontrol inventaris, termasuk efisiensi dan penghematan biaya [9].

Saillaja et al. (2023) mengambil langkah lebih maju dengan mengembangkan sistem berbasis IoT untuk manajemen inventaris di toko

ritel, yang memanfaatkan sensor dan analitik berbasis *cloud* untuk menyediakan data *real-time* tentang inventaris, yang kemudian digunakan untuk optimasi level stok dan meningkatkan kinerja penjualan [10]. Sementara itu, Emmanuel, Wijaya, dan Riantono (2023) memperkenalkan integrasi teknologi AI dalam Sistem Informasi Akuntansi melalui studi kualitatif, menyimpulkan bahwa sistem yang dikembangkan memberikan rekomendasi kontrol internal dan meningkatkan efisiensi dalam pencatatan transaksi penjualan dan inventaris [11].

Dalam konteks pengembangan aplikasi berbasis web, Angellin, Oetama, dan Amri (2023) menggunakan metode *Rapid Application Development* dan menemukan bahwa sistem yang dikembangkan berhasil mengatasi masalah pencatatan manual, mengurangi *human error*, dan meningkatkan otomatisasi dalam pengelolaan inventaris dan penjualan [12].

Terakhir, Yulianto dan Fauzi (2020) meneliti penggunaan metode *waterfall* dalam pengembangan sistem informasi penjualan *online* dan menemukan bahwa sistem yang dikembangkan berhasil meningkatkan proses bisnis, menyediakan informasi stok *real-time*, dan mendukung pengambilan keputusan bisnis yang lebih baik [13].

Referensi-referensi ini memberikan wawasan yang luas tentang berbagai pendekatan dan teknologi yang dapat diterapkan dalam pengembangan sistem informasi penjualan dan *inventory* berbasis web untuk toko kelontong, dan akan menjadi dasar penting dalam merancang sistem yang efektif dan efisien untuk Toko Kelontong Narjo.

2.2 Landasan Teori

2.2.1 Sistem

Asal usul kata "Sistem" berasal dari bahasa Latin *systema* dan bahasa Yunani *sustema*. Definisi sistem merujuk pada suatu kesatuan yang terdiri dari komponen atau elemen yang saling terhubung untuk memfasilitasi aliran informasi, materi, atau energi guna mencapai suatu tujuan tertentu. Lebih lanjut, sistem dapat diartikan sebagai kumpulan unsur atau elemen yang saling terkait dan berinteraksi dalam melakukan aktivitas bersama untuk mencapai tujuan yang ditetapkan. Secara umum, sistem dapat dipahami sebagai susunan terorganisir dari pandangan, teori, atau prinsip-prinsip yang berkaitan dengan suatu hal tertentu [14].

Menurut Fajarianto et al., (2017), suatu sistem memiliki karakteristik yang dapat menandakan bahwa hal tersebut dapat dianggap sebagai suatu sistem. Karakteristik tersebut mencakup [15]:

1. **Komponen (*Components*):** Sistem terdiri dari sejumlah komponen yang berinteraksi dan bekerja sama untuk membentuk satu kesatuan. Komponen ini bisa berupa subsistem, masing-masing dengan fungsi tertentu yang memengaruhi proses sistem secara keseluruhan.
2. **Batasan Sistem (*boundary*):** Sistem memiliki batasan yang memisahkan antara sistem tersebut dengan sistem lain atau

dengan lingkungannya. Batasan ini memungkinkan sistem dianggap sebagai satu kesatuan yang terpadu.

3. Lingkungan Luar Sistem (*Environment*): Segala hal yang berada di luar batasan sistem dan memengaruhi operasinya disebut sebagai lingkungan luar sistem.

2.2.2 Sistem Informasi

Pengertian Informasi dapat diartikan sebagai sekumpulan data yang telah disusun atau diorganisasi dengan baik dan telah memiliki kegunaan dan manfaat [16]. Sehingga dapat juga dikatakan bahwa pengertian Informasi merupakan hasil dari sebuah pengolahan dari sekumpulan data berbagai sumber, yang kemudian dilakukan berbagai proses, sehingga memberikan nilai, arti dan manfaat. Sehingga dapat disimpulkan bahwa sebuah informasi merupakan hasil dalam memproses berbagai data hingga memiliki makna dan bermanfaat bagi pemakainya [17].

Sehingga pengertian dari sebuah sistem informasi dapat dikatakan sebagai sebuah cara atau proses yang diorganisasi dengan baik untuk mengelompokkan, mengumpulkan, memasukan, mengolah serta menyimpan data, dan cara yang dipakai untuk menyimpan, mengelola, mengendalikan dan melaporkan informasi dilakukan secara sedemikian rupa, sehingga kegiatan sebuah organisasi dapat mencapai tujuan yang telah ditetapkan [18].

Dapat disimpulkan bahwa sistem informasi adalah suatu kumpulan dari data-data yang terintegrasi, memiliki relasi dan saling melengkapi, sehingga menghasilkan *output* yang baik berguna untuk memecahkan masalah dan membantu dalam pengambilan keputusan.

2.2.3 Penjualan

Menurut Swastha Basu (2019:8-10), “Menjual adalah ilmu dan seni mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang jasa yang ditawarkan. Jadi, adanya penjualan dapat tercipta suatu proses pertukaran barang dan/atau jasa antara penjual dengan pembeli” [19]. Penjualan tatap muka merupakan komunikasi orang secara individual yang dapat dilakukan untuk mencapai tujuan seluruh usaha pemasaran pada umumnya, yaitu meningkatkan penjualan yang dapat menghasilkan laba dengan menawarkan kebutuhan yang memuaskan kepada pasar dalam jangka panjang.

Menurut Enni (2016), “Jualan (*sale*) artinya hasil proses menjual atau yang dijual atau hasil penjualan. Penjualan (*selling*) artinya proses penjualan. Sedangkan menjual (*sell*) artinya menyerahkan sesuatu kepada pembeli dengan harga tertentu” [20].

Dari beberapa pengertian di atas mengenai penjualan, maka dapat disimpulkan bahwa penjualan adalah suatu kegiatan dan cara untuk mempengaruhi pribadi agar terjadi pembelian (penyerahan) barang atau jasa yang ditawarkan, berdasarkan harga yang telah

disepakati oleh kedua belah pihak dalam kegiatan tersebut secara tunai maupun kredit

2.2.4 Sistem Informasi Penjualan

Sistem penjualan adalah solusi yang digunakan oleh perusahaan untuk meningkatkan penjualan dan manajemen persediaan secara bersamaan. Sistem informasi penjualan merupakan solusi untuk toko bangunan berkah karena dapat menyediakan stok barang tanpa batas, pencatatan barang yang akurat, kemudahan pengelolaan inventaris, transaksi penjualan terkomputerisasi, dan manajemen informasi pegawai. Penggunaan sistem informasi penjualan dengan perangkat keras dapat meningkatkan pertumbuhan dan pengembangan bisnis secara keseluruhan, serta menjadi pendekatan efektif dalam manajemen bisnis [21].

2.2.5 *Inventory*

Inventory adalah persediaan bahan baku, atau stok yang tersimpan untuk digunakan di masa depan bisa juga dalam jangka pendek. *Inventory* yaitu proses pengelolaan perolehan atau penyimpanan stok barang milik suatu tempat usaha atau perusahaan.

Oleh karena itu, penulis menyimpulkan bahwa persediaan yaitu proses perolehan dan pengiriman stok barang yang dimiliki oleh suatu tempat usaha atau perusahaan untuk melakukan kegiatan untuk waktu yang akan datang maupun untuk waktu tertentu [22].

2.2.6 *Website*

Menurut Muhyidin et al., (2020) menyatakan “*Website* merupakan suatu layanan sajian informasi yang menggunakan konsep *hyperlink*, yang memudahkan *surfer* (sebutan bagi pemakai komputer yang melakukan penelusuran informasi di internet)” [23]. Selanjutnya menurut Doni & Rahman, (2020) *Website* adalah sekumpulan dokumen yang berada pada server dan dapat dilihat oleh user dengan menggunakan *browser*. Dokumen itu bisa terdiri dari beberapa halaman. Tiap-tiap halamannya memberi informasi atau interaksi yang beraneka ragam. Informasi atau interaksi yang beraneka ragam. Informasi dan interaksi itu bisa berupa tulisan, gambar atau bahkan dapat ditampilkan dalam bentuk video, animasi, suara, dan lain-lain [24].

Dari kedua peneliti diatas dapat disimpulkan bahwa *website* adalah sebuah layanan informasi yang memanfaatkan konsep *hyperlink* untuk memudahkan pengguna internet dalam mencari dan menelusuri informasi. *Website* merupakan kumpulan dokumen yang tersimpan di server dan dapat diakses melalui *browser*. Dokumen-dokumen tersebut terdiri dari berbagai halaman yang menyediakan informasi atau interaksi yang beragam, seperti teks, gambar, video, animasi, suara, dan lainnya. Dengan demikian, *website* menjadi sarana yang penting dalam menyajikan informasi secara variatif kepada pengguna internet.

2.2.7 PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) adalah pemrograman interpreter yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan. PHP disebut sebagai pemrograman *Server Side Programming*, hal ini dikarenakan seluruh prosesnya dijalankan pada server tidak dijalankan pada client. PHP merupakan suatu bahasa dengan hak cipta terbuka atau yang juga dikenal dengan istilah *Open Source*, yaitu pengguna dapat mengembangkan kode fungsi PHP dengan kebutuhannya [25].

Cara kerja PHP melibatkan serangkaian langkah yang terjadi di sisi server web. Berikut adalah langkah-langkah utama tentang bagaimana PHP bekerja [26]:

1. Permintaan pengguna

Ketika pengguna mengakses situs *web*, *server* menerima permintaan tersebut.

2. Pemrosesan *server*

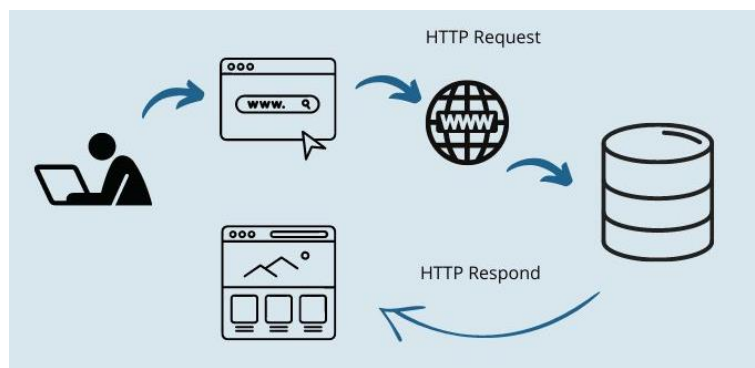
Kode PHP di *server* mengolah permintaan ini. Ini bisa mencakup mengambil data dari basis data, memproses formulir atau menghasilkan konten dinamis lainnya.

3. Penghasilan laman *web*

Setelah pemrosesan selesai, PHP menghasilkan halaman yang siap ditampilkan.

4. Pengiriman ke *browser*

Halaman *web* yang sudah jadi dikirimkan ke *browser* pengguna untuk ditampilkan.



Gambar 2.1 Alur Kerja PHP

Ada pun fungsi PHP umumnya adalah untuk mengubah halaman statis menjadi halaman dinamis. Memakai PHP, *website* bisa menyesuaikan tampilan konten berdasarkan situasi. Selain itu, PHP juga memiliki beberapa fungsi lainnya yaitu [27]:

1. Mempersingkat Tatanan HTML (*Hypertext Markup Language*) dan CSS (*Cascading Style Sheets*)

Fungsi yang pertama yakni mempersingkat tatanan penulisan HTML, karena tidak efektif apabila tatanan suatu web terlalu panjang. Maka dengan menggunakan PHP kamu dapat mengaturnya sesuai jumlah baris yang relatif.

2. *Input Data*

Berfungsi sebagai penyimpan dan penginput data pada sistem database, misalnya MySQL.

3. Manajemen *Cookie* dan *Session*

Fungsi ini berguna untuk menyimpan *username* dan *password* pengguna pada *browser*, sehingga tidak perlu mengetik ulang ketika membuka situs yang sama.

4. Kompres Teks

Fungsi terakhir yaitu mengompres teks menjadi lebih singkat dan jelas.

2.2.8 *Laravel*

Laravel adalah kerangka kerja PHP yang sangat berguna dan memungkinkan pembuatan aplikasi web dengan mudah. *Laravel* menggunakan pola desain MVC (*model-view-controller*) dan sistem *Symfony*, serta menyediakan sistem paket modular yang memungkinkan perluasan aplikasi dengan modul baru. *Laravel* juga mengadopsi banyak komponen dari kerangka kerja lain yang membantu menciptakan aplikasi yang aman dan operasional dengan cepat [2].



Gambar 2.2 Logo *Framework Laravel*

2.2.9 **Basis Data**

Basis data terdiri dari 2 kata yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempang



bersarang atau berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (Pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya. Yang diwujudkan dalam bentuk angka, huruf, symbol, teks, gambar, bunyi, atau kombinasinya [28].

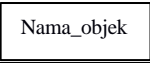


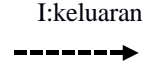

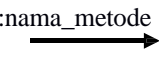
2.2.10 UML (*Unified Modeling Language*)

UML (*Unified Modelling Language*) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. Awal mulanya, UML diciptakan oleh *Object Management Group* dengan versi awal 1.0 pada bulan Januari 1997. UML juga dapat didefinisikan sebagai suatu bahasa standar visualisasi, perancangan, dan pendokumentasian sistem, atau dikenal juga sebagai bahasa standar penulisan *blueprint* sebuah *software*.

UML diharapkan mampu mempermudah Pengembangan Piranti Lunak (RPL) serta memenuhi semua kebutuhan pengguna dengan efektif, lengkap, dan tepat. Hal itu termasuk faktor-faktor *scalability*, *robustness*, *security*, dan sebagainya [29].

Tabel 2.1 Simbol-Simbol UML (*Unified Modeling Language*)

No.	Gambar	Nama	Keterangan
1		Aktor	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi itu sendiri.
2		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.

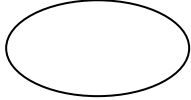
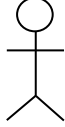
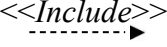
No.	Gambar	Nama	Keterangan
3		<i>Objek</i>	Menyatakan objek yang berinteraksi oleh pesan.
4		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
5		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
6		<i>Pesan tipe return</i>	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.
7		<i>Pesan tipe send</i>	Menyatakan bahwa suatu objek mengirim data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
8		<i>Pesan tipe call</i>	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.

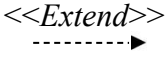
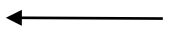
UML memberikan kemampuan yang bermanfaat dalam memodelkan sistem dari berbagai perspektif. Konsep dasar abstraksi UML terdiri dari klasifikasi struktural, perilaku dinamis, dan manajemen model. Berikut adalah beberapa diagram yang umum digunakan dalam UML :

1. Use Case Diagram

Use Case Diagram merupakan diagram yang menggambarkan hubungan antara aktor dengan sistem. *Use case diagram* dapat mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat. *Use case diagram* dapat digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan dapat mempresentasikan sebuah interaksi aktor dengan sistem [30]. Simbol – simbol *Use Case Diagram* dapat dilihat pada Tabel 2.2.

Tabel 2.2 Simbol-Simbol *Use Case Diagram*

No.	Gambar	Nama	Keterangan
1.		<i>Use Case</i>	Fungsional yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2.		<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar 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>Include</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use</i>

No.	Gambar	Nama	Keterangan
			<i>case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya
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; 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 misal Arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i> -nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.
5.		Generalisasi atau <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, misalnya: arah panah mengarah pada <i>use case</i> yang menjadi




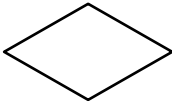

No.	Gambar	Nama	Keterangan
			generalisasinya (umum)

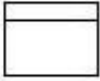
2. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak [31].

Simbol – simbol *Activity Diagram* dapat dilihat pada Tabel 2.3.

Tabel 2.3 Simbol-Simbol *Activity Diagram*

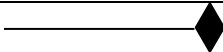
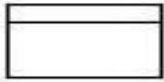
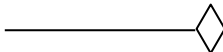


No.	Gambar	Nama	Keterangan
1.		Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Penggabungan atau <i>Join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
4.		Percabangan atau <i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari Satu.
5.		Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.

No.	Gambar	Nama	Keterangan
6.		<i>Swimlane</i>	<i>Swimlane</i> memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

3. *Class Diagram*

Class Diagram adalah gambar yang menjelaskan struktur dari program yang akan dibuat menggunakan konsep OOP (*Object Oriented Programming*). *Class diagram* menggambarkan bagaimana objek pada dunia nyata digambarkan pada struktur yang biasa memiliki atribut dan method [32]. Simbol – simbol *Class Diagram* dapat dilihat pada Tabel 2.4.

Tabel 2.4 Simbol-Simbol *Class Diagram*

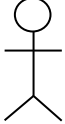
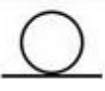
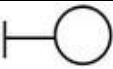
No.	Gambar	Nama	Keterangan
1.		<i>Composition</i>	Menggambarkan sebuah relasi dari sebuah <i>class</i> yang tidak bisa berdiri sendiri.
2.		<i>Class</i>	Kelas pada struktur sistem.
3.		<i>Aggregation</i>	Menunjukkan sebuah bagian relasi agregasi
4.		<i>Dependency</i>	Relasi antar kelas dengan makna kebergantungan antar kelas.
5.		<i>Association</i>	Garis yang menghubungkan antara dua kelas



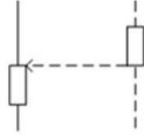
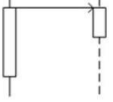
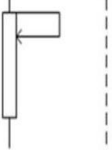
No.	Gambar	Nama	Keterangan
			atau lebih dan menunjukkan bahwa kelas yang terhubung menunjukkan sebuah relasi seperti : <i>one-to-one</i> (1 – 1), <i>one-to-many</i> (1 – M) , <i>many-to-many</i> (M – M).

4. *Sequence Diagram*

Sequence diagram adalah grafik dua dimensi dimana obyek ditunjukkan dalam dimensi horizontal, sedangkan *lifeline* ditunjukkan dalam dimensi vertikal [32]. Simbol – simbol *Sequence Diagram* dapat dilihat pada Tabel 2.5.

Tabel 2.5 Simbol-Simbol *Sequence Diagram*

No	Gambar	Nama	Keterangan
1.		<i>Actor</i>	Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.
2.		<i>Entity Class</i>	Menggambarkan informasi yang harus disimpan oleh sistem (struktur data dari sebuah sistem)
3.		<i>Boundary Class</i>	Menggambarkan interaksi antara satu atau lebih <i>actor</i> dengan sistem, memodelkan bagian dari yang bergantung pada pihak lain

No	Gambar	Nama	Keterangan
			disekitarnya dan merupakan pembatas dengan dunia luar.
4.		<i>Control Class</i>	Menggambarkan “perilaku mengatur”, mengkoordinasikan perilaku dan dinamika dari suatu sistem, menangani tugas utama dan mengontrol alur kerja suatu sistem.
5.		<i>Lifeline</i>	Eksekusi obyek selama <i>sequence</i> (<i>message</i> dikirim atau diterima dan aktifasinya).
6.		<i>Return Message</i>	Menggambarkan pesan/hubungan antar obyek, yang menunjukkan urutan kejadian yang terjadi.
7		<i>Object Message</i>	Menggambarkan pesan/hubungan antar obyek yang menunjukkan urutan kejadian yang terjadi.
8		<i>Generalization</i>	Menggambarkan pesan/hubungan obyek itu sendiri, yang menunjukkan urutan kejadian yang terjadi.