
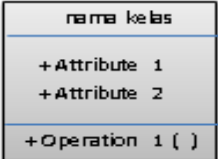

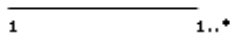











## DAFTAR SIMBOL




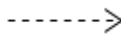
### 1. Package Diagram

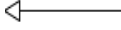





Simbol	Deskripsi
Package 	Package merupakan sebuah bungkus dari satu atau lebih kelas
Operasi 	Kelas pada struktur sistem
Antarmuka / interface 	sama dengan konsep interface dalam pemrograman berorientasi objek
Asosiasi 	relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
Asosiasi berarah/directed asosiasi 	relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan multiplicity
Generalisasi 	relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
Ketergantungan / dependency 	relasi antar kelas dengan makna ketergantungan antar kelas
Agregasi 	relasi antar kelas dengan makna semua-bagian (whole-part)

## 2. Simbol Activity Diagram


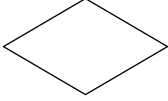
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran


## 3. Simbol Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<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> ).
3		<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> ).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .

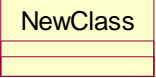

5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

#### 4. Simbol Diagram Hubungan Entitas (ERD)

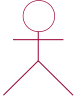
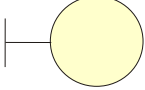
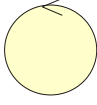
	<b>Entitas</b> Menggambarkan kumpulan objek yang anggota-anggotanya berperan dalam sistem atau menggambarkan/menyatakan suatu himpunan entitas.
	<b>Relasi</b> Menggambarkan sehubungan hubungan antara objek yang dibangun (relationship) / menggambarkan himpunan hubungan yang ada diantara himpunan entitas.




	<p><b>Garis Penghubung</b></p> <p>Merupakan penghubung antara entitas dengan relationship ataupun sebaliknya relationship ke entitas.</p>
---	---

### 5. Simbol Class Diagram

	<p><b>Class</b></p> <p>Memiliki tiga area pokok, yaitu : nama, atribut, <i>method</i>. Bagian atas adalah bagian nama dari <i>class</i>. Bagian tengah mengidentifikasi atribut <i>class</i>. Bagian akhir mendefinisikan method dari sebuah <i>clas</i></p>
	<p><b>Association</b></p> <p>Menggambarkan mekanisme komunikasi suatu objek dengan objek lainnya.</p>

### 6. Simbol Sequence Diagram

	<p><b>Actor</b></p> <p>Menggambarkan seseorang / sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.</p>
	<p><b>Boundary</b></p> <p>Menggambarkan interaksi antara satu / lebih actor dengan sistem, memodalkan bagian dari sistem yang bergantung pada pihak lain disekitarnya dan merupakan pembatas sistem dengan dunia luar.</p>
	<p><b>Control</b></p> <p>Menggambarkan “perilaku mengatur” mengkoordinasikan perilaku sistem dan dinamika dari suatu sistem, mengenai tugas utama dan mengontrol alur kerja suatu sistem.</p>

	<p><b>Entity</b> Menggambarkan informasi yang harus disimpan oleh sistem (struktur data dari sebuah sistem).</p>
	<p><b>Object Message</b> Menggambarkan pesan/hubungan antar obyek yang menunjukkan urutan kejadian yang terjadi.</p>
	<p><b>Message to Self</b> Menggambarkan pesan/hubungan obyek itu sendiri, yang menunjukkan urutan kejadian yang terjadi.</p>