

BAB II

LANDASAN TEORI

2.1 Animasi

Animasi adalah sebuah proses merekam dan memainkan kembali serangkaian gambar statis untuk mendapatkan sebuah ilusi pergerakan (*Ibiz Fernandes*, 2002). Secara harfiah animasi dapat diartikan menghidupkan. Yaitu menggerakkan objek yang tidak bisa bergerak sendiri.

Animasi juga merupakan salah satu media yang bisa digunakan untuk membuat proses pembelajaran menjadi menarik. Dengan animasi yang terdapat unsur edukasi tersebut siswa bisa tertarik untuk belajar. Fungsi animasi dapat direalisasikan dalam berbagai cara. Salah satunya adalah dengan menggunakan data (data-driven) pada setiap satuan waktu. Cara ini merupakan yang paling sederhana karena untuk menampilkan perubahan cukup menggunakan *data* yang sudah ada sebelumnya. Umumnya lebih dikenal dengan sebutan *frame-by-frame animation*. Cara ini pada umumnya digunakan pada gerakan yang cukup kompleks untuk dimodelkan seperti gerakan manusia berjalan.

2.1.1 Jenis-jenis Animasi

Animasi yang dahulunya mempunyai prinsip yang sederhana, sekarang telah berkembang menjadi beberapa jenis akibat dari semakin canggih teknologi yang ada, yaitu :

a. Animasi 2D

Animasi ini yang paling akrab dengan keseharian oleh masyarakat yang biasa disebut film kartun.

b. Animasi 3D

Merupakan pengembangan dari animasi 2D. Apabila dalam animasi 2D hanya menggunakan koordinat x dan y, maka dalam animasi 3D terdapat penambahan koordinat yang disebut dengan koordinat z. Sehingga objek yang diperlihatkan semakin hidup dan nyata, mendekati wujud objek aslinya. Animasi ini biasa disebut dengan *CGI (Computer Generated Imagery)*

c. Animasi Tanah Liat (clay animation)

Animasi ini mulai ada pada tahun 1897an yang bahan pembuatan animasinya menggunakan bahan dasar pasticin yang menutupi kerangka yang dibuat khusus sesuai bentuk tokoh yang ingin dibuat.

2.1.2 Software Pembuat Animasi

Di pasaran sekarang ini sudah banyak beredar *software* pembuat animasi, baik itu 2D maupun 3D.

a. Software animasi 2D

Software yang umum digunakan dalam pembuatan animasi 2D adalah flash, CoRETAS, corel RAVE, after effect, dan lain sebagainya.

b. Software animasi 3D

Software yang digunakan dalam pembuatan animasi 3D contohnya adalah maya, 3DSMax, motion builder, wing 3D, 3D Blender, dan lain sebagainya.

2.2 System Development Life Cycle (SDLC)

Menurut Turban (2003, p464), SDLC adalah untuk membangun sebuah sistem dengan terstruktur dan metodis yang mengharuskan tahap life cycle dari mulai ide awal hingga pada tahap akhir sistem. Salah satu tipe awal SDLC yang paling banyak digunakan adalah *metode waterfall*.

Pembangunan dengan metode waterfall memiliki tujuan yang berbeda untuk setiap fasenya. Setelah satu fase selesai, hasil dari fase tersebut akan dilanjutkan ke tahap selanjutnya dan tidak ada jalan kembali.

Tahapan-tahapan metode waterfall:

1. *Requirement Analysis*

Seluruh kebutuhan dalam pembuatan game harus bisa didapatkan pada fase berikut, termasuk didalamnya kegunaan yang diharapkan oleh pengguna dan batasan game yang akan dibuat.

2. *System Design*

Sebelum game mulai dibangun, tahap ini dilakukan terlebih dahulu untuk memberikan gambaran apa yang harus dikerjakan dan bagaimana tampilannya.

3. *Implementation*

Pada tahap ini mulai dilakukan coding. Pembuatan game dipecah menjadi modul-modul kecil yang nantinya digabungkan dalam tahap selanjutnya. Selain itu dalam tahap ini juga dilakukan pemeriksaan apakah sudah memnuhi fungsi yang diinginkan atau belum.

4. *Integration and Testing*

Di tahap berikut dilakukan pengujian untuk mengetahui apakah game yang dibuat telah sesuai dengan desainnya dan masih terdapat kesalahan atau

tidak.

5. *Operation and Maintenance*

Tahap ini adalah tahap terakhir. Game yang sudah jadi dijalankan serta dilakukan pemeliharaan. Termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.

2.3 3D Blender

3D Blender merupakan *software* animasi 3D yang sifatnya *open source* yang dibuat oleh tim Blender Foundation. *Software* Blender sangatlah ringan karena hanya sebesar 30MB hingga 50MB yang dapat diunduh secara gratis di www.blender.org.

Dalam pembuatan tugas akhir ini, penulis memakai blender versi 2.75 yang merupakan versi terbaru pada tanggal pembuatan proyek tugas akhir ini.



Gambar 2.1 Blender versi 2.75

Karena software ini bersifat *opensource*, maka perkembangannya pun sangat cepat. Sehingga apabila ada update oleh blender *foundation*, maka dalam waktu yang cukup singkat akan keluar blender terbaru.

2.4 Unity 3D

Unity 3D adalah sebuah game engine yang berbasis *cross-platform*. Unity dapat digunakan untuk membuat sebuah game yang bisa digunakan pada perangkat komputer, ponsel pintar android, iPhone, PS3, dan bahkan X-BOX. Unity adalah sebuah *tool* yang terintegrasi untuk membuat game, arsitektur bangunan dan simulasi. Unity bisa untuk *games PC* dan *games Online*. Untuk *games Online* diperlukan sebuah *plugin*, yaitu Unity Web Player, sama halnya dengan Flash Player pada *Browser*.

Unity tidak dirancang untuk proses desain atau *modelling*, dikarenakan unity bukan *tool* untuk mendesain. Jika ingin mendesain, pergunakan 3D editor lain seperti 3dsmax atau Blender. Banyak hal yang bisa dilakukan dengan unity, ada fitur *audio reverb zone*, *particle effect*, dan *Sky Box* untuk menambahkan langit. Fitur *scripting* yang disediakan, mendukung 3 bahasa pemrograman, JavaScript, C#, dan Boo. *Flexible and EasyMoving*, *rotating*, dan *scaling objects* hanya perlu sebaris kode. Begitu juga dengan *Duplicating*, *removing*, dan *changing properties*. *Visual Properties Variables* yang di definisikan dengan *scripts* ditampilkan pada *Editor*. Bisa digeser, di *drag and drop*, bisa memilih

warna dengan color picker. Berbasis .NET. Artinya perjalanan program dilakukan dengan *Open Source .NET platform, Mono*.

2.5 Game

Game secara bahasa berasal dari bahasa Inggris yang berarti permainan. Namun dalam proyek tugas akhir ini yang dimaksud dengan *game* adalah *game computer*. Permainan dengan alat satu set *computer* di mana user dalam arti pemain mempengaruhi benda atau object yang terlihat di layar computer untuk tujuan menghibur. Video game pada dasarnya adalah bentuk hiburan yang sama namun tidak hanya mengacu ke permainan yang berjalan di computer probadi, tetapi juga pada konsol dan mesin arcade. (*Wandah wibawanto, 2006*).

2.5.1 Jenis Game

Jenis – jenis game yang beredar saat ini adalah side scrolling game, shooting, RPG, RTS, simulation, racing, dan fighting (*Fauzi. A.R, hajar. S.P., Diginovac, 2009*).

2.5.1.1 Side Scrolling Game

Game jenis ini hanya dapat mennggerakkan karakter ke kanan, kiri, atas, atau bawah sesuai dengan gerakkan background. Game ini sangat terkenal di antaranya adalah contra, super Mario bros, metal slug, dan sebagainya.

2.5.1.2 Shooting

Game yang mewajibkan user untuk mencari dan menembak musuh dalam pencapaian tujuan. Contoh game jenis ini adalah counter strike, virtual cop, crisis, call of duty, dan sebagainya.

2.5.1.3 RPG (Role Playing Game)

Pada game jenis berikut *user* diminta untuk memerankan suatu karakter untuk mengemban misi khusus. Game ini sangat kompleks. Karena *user* akan dibawa ke dalam frame baru yang bisa mendapatkan sesuatu atau malah melakukan pertarungan. Contoh *game* jenis ini sangatlah banyak. Yaitu ragnarok, suikoden, dan yang cukup terkenal yang menjadi *icon* square enix adalah final fantasy, serta masih banyak lagi yang lainnya

2.5.1.4 RTS (Real Time Strategy)

Pada umumnya, game yang termasuk dalam kategori berikut merupakan game –game yang bertemakan peperangan. Ketika *user* memainkan game jenis berikut, *user* akan dibebani misi dengan dibekali pasukan seadanya. *User* dapat menggerakkan, memperbanyak, dan melengkapi persenjataan pasukan tersebut sambil merancang strategi untuk mempertahankan dan menguasai teritorial. Contoh game RTS yang terkenal adalah armyman, age of empire, war craft, dota, dan lain sebagainya.

2.5.1.5 Simulation

Sesuai dengan namanya, game jenis berikut merupakan game yang mensimulasikan sebuah keadaan sebenarnya. Sebagai contoh adalah simulasi pengendalian pesawat terbang dengan Microsoft flight simulator, simulasi kehidupan manusia pada kesehariannya pada game The Sims, atau simulasi menjadi manajer tim sepak bola pada game champhionsip manager, dan masih banyak lagi lainnya.

2.5.1.6 Racing

Dari namanya saja sudah tentu bisa diduga bahwa game jenis ini adalah game balapan. Contoh dari game ini cukup banyak, antara lain need for speed, nascar, crash tag team racing, grand turismo, dan masih banyak lagi lainnya.

2.5.1.7 Fighting

Ini adalah kategori game pertarungan. Contoh dari game ini adalah, street fighter, tekken, mortal kombat, marvel vs Capcom, dan sebagainya.

2.5.1.8 Casual

Sesuai namanya, game yang casual itu tidak kompleks, mainnya rileks dan sangat mudah untuk dipelajari (bahkan cenderung langsung bisa dimainkan). Jenis ini biasanya memerlukan spesifikasi komputer yang standar pada jamannya dan ukurannya tidak lebih dari 100 MB karena biasanya dapat di download versi demo-nya di website resminya. Genre permainannya biasanya puzzle atau action sederhana dan umumnya dapat dimainkan hanya menggunakan mouse (biasanya game lain menggunakan banyak tombol tergantung game-nya). Contoh: Diner Dash, Sally Salon, Bejeweled, Zuma, Feeding Frenzy, Insaniquarium.

2.5.1.9 Arcade

Game Arcade adalah genre game komputer yang mengandalkan ketangkasan pemainnya dalam memegang kontrol. Ciri-ciri *game arcade* adalah mempunyai level yang singkat, kontrol yang mudah, serta tingkat kesulitan yang bertambah dengan cepat. Jenis game ini dirancang untuk memancing adrenalin pemain serta tidak membutuhkan jalan cerita yang bagus. Game seperti ini

membutuhkan waktu belajar yang relatif singkat. Sifat dari game arcade adalah real-time.

2.6 Fungsi Animasi Game sebagai Media Pembelajaran (edukasi)

Dalam praktek belajar mengajar, pendidikan selalu diasosiasikan dengan hal-hal yang serius, sementara hiburan identik dengan sikap santai yang sarat akan permainan. Keduanya seolah merupakan 2 hal terpisah yang bahkan sangat bertolak belakang. Dalam teori pembelajaran modern, kini 2 hal tersebut ternyata dapat dipadukan. Hasil dari perpaduan keduanya ternyata malah memiliki dampak yang begitu dasyat. Suatu pembelajaran akan membuahkan hasil yang lebih memuaskan jika dilakukan dengan menyenangkan. Pembelajaran yang menyenangkan disebut dengan edutainment, perpaduan antara education dan entertainment. Sebuah proses belajar mengajar yang didesain sedemikian rupa sehingga muatan pendidikan dan hiburan dapat dikombinasikan dengan harmonis. Oleh karenanya pembelajaran terasa lebih menyenangkan karena dikombinasikan dengan humor, permainan, bermain peran, kuis, berselancar dengan internet, dan masih banyak lagi. Proses belajar yang interaktif justru lebih memberikan ruang kepada peserta didik untuk dapat mengalami, mencoba, dan merasakan dan menemukan sendiri. Dengan ini kelas menjadi arena bermain yang begitu menyenangkan bagi anak. Pelajaran dikemas dalam suasana bermain dan bereksperimen. Suasana kelas yang menyenangkan tersebut tidak hanya menjadi

tempat belajar yang bermanfaat bagi anak namun juga bisa menjadi tempat penghilang stress, meningkatkan keterampilan interpersonal dan kreatifitas siswa. (Rosenberg, 2001)

2.6.1 Dampak Animasi pada Proses Belajar Mengajar

Selama ini animasi digunakan dalam media pembelajaran untuk dua alasan.

Pertama, sebagai media untuk menarik perhatian *audience* dan memperkuat motivasi. Animasi jenis berikut biasanya berupa tulisan atau gambar bergerak yang lucu maupun aneh yang sekiranya dapat menarik perhatian peserta didik. Kedua, sebagai sarana untuk memberikan pemahaman kepada siswa/siswi atas materi yang diajarkan.

2.6.2 Fungsi Game sebagai Media Pembelajaran

Anak yang gemar bermain video *game* adalah anak yang sangat menyukai tantangan. Anak-anak ini cenderung tidak menyukai hal – hal yang ransangan daya tariknya lemah, *monoton*, tidak ada tantangan, dan lamban. Hal ini setidaknya berakibat pada proses belajar akademis. Suasana kelas seolah sebuah penjara bagi mereka. Tubuh di kelas namun pikiran, rasa penasaran, dan keinginannya ada di dalam video game. Malah tidak jarang pernah ditemukan seorang murid membolos sekolah hanya ingin bermain game. Namun uniknya adalah anak yang fanatic bermain game biasanya merupakan individu yang berintelijensi tinggi, bermotivasi, dan berorientasi pada prestasi.

Fungsi game dalam porsinya yang masih belum disadari adalah reflek saraf otak yang sebenarnya sedikit banyak terasah dalam sebuah game. Itulah sebabnya banyak dikembangkan game edukasi untuk anak. Karena dengan belajar melalui visualisasi yang menarik diharapkan semangat anak untuk belajar akan lebih terpacu.

2.7 Punakawan

SUNAN Kalijaga juga dipandang sebagai tokoh yang menghasilkan kreasi baru yaitu dengan adanya wayang kulit dengan segala seperangkat gamelannya. Wayang kulit ini merupakan pengembangan baru dari wayang beber yang memang sudah ada sejak zaman Erlangga. Diantara wayang ciptaan Sunan Kalijaga beserta Sunan Bonang dan Sunan Giri adalah wayang Punakawan pandawa yang terdiri dari semar, petruk, gareng dan bagong. (*Yoki Murantyo, 2013*). Adapun falsafah dari arti nama keempat punakawan pandawa itu adalah sebagai berikut:

a. Semar

Berasal dari bahasa Arab *Ismar* yang artinya paku berfungsi sebagai pengokoh yang goyah. Ibarat ajaran agama Islam yang didakwahkan para walisongo diseluruh kerajaan Majapahit, yang pada waktu itu sedang dalam pergolakan dengan awal didirikannya kerajaan Demak oleh Raden Patah. Hal senada sesuai dengan hadist *Al Islami Ismaraddunya* yang berarti Islam adalah pengokoh (paku pengokoh) keselamatan dunia.

b. Gareng

Dari bahasa Arab *Naal Qariin* oleh orang Jawa menjadi *naala gareng* yang berarti memperoleh banyak teman, dan tugas konsepsional para walisongo sebagai juru *dakwah* (da'i) ialah untuk memperoleh sebanyak-banyaknya kawan untuk kembali ke jalan Tuhan dengan sikap arif dan harapan yang baik.

c. Petruk

Berasal dari bahasa Arab *Fatruk* oleh pengucapan lidah Jawa menjadi petruk. Kata tersebut merupakan kata pangkal kalimat pendek dari sebuah *wejangan tasawuf* tinggi yang berbunyi. *Fat-ruk kulla man siwallahi*, yang artinya tinggalkan apapun selain Allah. *Wejangan* tersebut kemudian menjadi watak pribadi para wali dan muballigh pendidikan pada waktu itu.

d. Bagong

Dari bahasa Arab *Baghaa* oleh orang Jawa menjadi bagong yang berarti berontak, yaitu berontak terhadap *kebatilan* atau kemungkaran kesalahan. Dalam versi lain berasal dari kata *baqa'* (Arab) yang berarti kekal. Menurut versi lain lagi bagong berasal dari kata *Bahar* (Arab) yang berarti bumbu. Betapa gayengnya ki dalang mementaskan tokoh bagong sebagai bumbu penyedap lakon. Dia dikenal sebagai punakawan yang kritis, blokosuto, dan tidak segan-segan mengkritik dan menyindir keadaan yang dipandang tidak pas.

2.8 Object Oriented Analisis and Design

OOAD (*Object Oriented Analisis and Design*) adalah suatu pendekatan rekayasa perangkat lunak dari sebuah sistem yang terdiri dari sekelompok objek yang saling berinteraksi, dan setiap objek itu mewakili beberapa entitas. Yang ditandai dengan adanya sebuah kelas, elemen data dan perilaku dari objek tersebut.

2.8.1 Pengertian OOAD

OOAD adalah metode analisis yang memeriksa kebutuhan dari sudut pandang kelas-kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas.

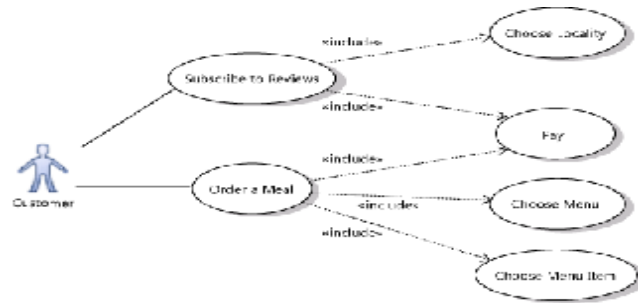
2.8.2 Unified Modeling Language

UML (*Unified Modeling Language*) adalah bahasa standar yang digunakan untuk menjelaskan dan memvisualisasikan artefak dari proses analisis dan desain berorientasi objek. UML menyediakan standar pada notasi dan diagram yang bisa digunakan untuk memodelkan suatu sistem. UML menyediakan 10 macam diagram untuk memodelkan suatu sistem atau aplikasi berorientasi objek diantaranya :

- Use Case Diagram : untuk memodelkan proses bisnis.
- Conceptual Diagram : untuk memodelkan konsep-konsep yang ada di dalam aplikasi.
- Sequence Diagram : untuk memodelkan pengiriman pesan (*message*) antar objek.
- Collaboration Diagram : untuk memodelkan interaksi antar objek.
- State Diagram : untuk memodelkan perilaku objek di dalam sistem.
- Activity Diagram : untuk memodelkan perilaku Use Case dan objek di dalam sistem.
- Class Diagram : untuk memodelkan struktur kelas.
- Object Diagram : untuk memodelkan struktur objek.
- Component Diagram : untuk memodelkan komponen objek.
- Deployment Diagram : untuk memodelkan distribusi aplikasi.

2.8.3 Use Case Diagram

Use case adalah abstraksi dari interaksi antara sistem dan actor. *Use case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.








Gambar 2.2 Contoh use case diagram

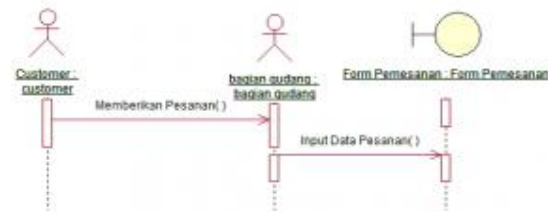
Diagram *Use Case* berguna dalam tiga hal :

1. Menjelaskan fasilitas yang ada (*requirement*).
2. Komunikasi dengan klien.
3. Membuat tes dari kasus-kasus secara umum.

Tabel 2.1 Simbol - simbol Use Case

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
3		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
5		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

2.8.4 Sequence Diagram



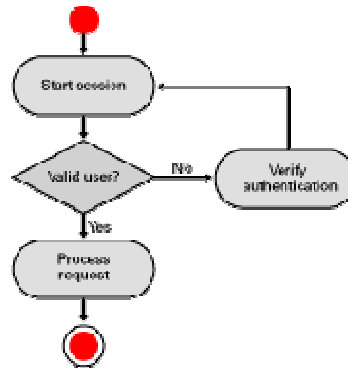
Gambar 2.3 Contoh sequence diagram

Sequence diagram menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram*.

Tabel 2.2 Penjelasan simbol sequence diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek entity, antarmuka yang saling berinteraksi.
		<i>Actor</i>	Digunakan untuk menggambarkan user / pemgguna.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		<i>Boundary</i>	Digunakan untuk menggambarkan sebuah form.
4		<i>Control Class</i>	Digunakan untuk menghubungkan <i>boundary</i> dengan tabel.
5		<i>Entity Clas</i>	Digunakan untuk menggambarkan hubungan kegiatan yang akan dilakukan.

2.8.5 Activity Diagram



Gambar 2.4 Contoh activity diagram

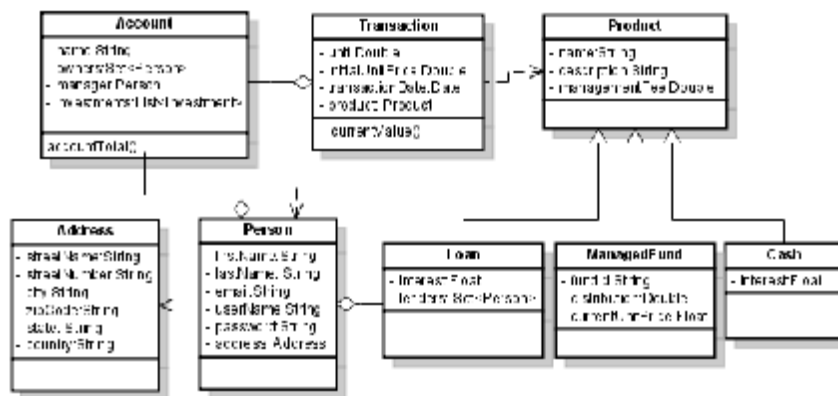
Activity diagram menyediakan analisis dengan kemampuan untuk memodelkan proses dalam suatu sistem informasi. *Activity diagram* dapat digunakan untuk alur kerja model, *use case individual*, atau logika keputusan yang terkandung dalam metode individual. *Activity diagram* juga menyediakan pendekatan untuk proses pemodelan paralel.

Tabel 2.3 Penjelasan tentang activity diagram

NO	GAMBAR	NAMA	KETERANGAN
1		Activity	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		Initial Node	Bagaimana objek dibentuk atau diawali.
3		Activity Final Node	Bagaimana objek dibentuk dan diakhiri
4		Decision	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu
5		Line Connector	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

2.8.6 Class Diagram

Class adalah dekripsi kelompok objek-objek dengan properti, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *class* diagram dapat memberikan pandangan global atas sebuah sistem. Hal tersebut tercermin dari *class- class* yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa class diagram. *Class diagram* sangat membantu dalam visualisasi struktur kelas dari suatu sistem.



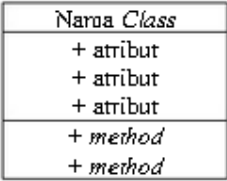


Gambar 2.5 Contoh class diagram

Kelas merupakan sebuah objek dimana dalam kelas diagram, kelas diwakilkan dengan kotak yang berisi tiga bagian diantaranya yaitu :

- Nama Kelas
- Atribut Kelas : merupakan informasi mengenai objek atau kelas yang disimpan atau setidaknya dipertahankan sementara.
- Method : merupakan hal – hal atau operasi yang dapat dilakukan oleh suatu objek atau kelas

- Visibilitas, merupakan *visibility* dari tiap atribut atau operator pada suatu kelas.

Tabel 2.4 Penjelasan tentang class diagram

Nama Komponen	Keterangan	Simbol
<i>Class</i>	<p><i>Class</i> adalah blok - blok perbangun pada pemrograman berorientasi obyek. Sebuah <i>class</i> digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari <i>class</i>. Bagian tengah mendefinisikan property/atribut <i>class</i>. Bagian akhir mendefinisikan <i>method-method</i> dari sebuah <i>class</i>.</p>	
<i>Association</i>	<p>Sebuah asosiasi merupakan sebuah relationship paling umum antara 2 <i>class</i> dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 <i>class</i>. Garis ini bisa melambangkan tipe-tipe <i>relationship</i> dan juga dapat menerapkan hura-hura multiplisitas pada sebuah <i>relationship</i>. (Contoh: <i>One-to-one</i>, <i>one-to-many</i>, <i>many-to-many</i>).</p>	
<i>Composition</i>	<p>Jika sebuah <i>class</i> tidak bisa berdiri sendiri dan harus merupakan bagian dari <i>class</i> yang lain, maka <i>class</i> tersebut memiliki relasi <i>Composition</i> terhadap <i>class</i> tempat dia bergantung tersebut. Sebuah <i>relationship composition</i> digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi/solid.</p>	
<i>Dependency</i>	<p>Kadangkala sebuah <i>class</i> menggunakan <i>class</i> yang lain. Hal ini disebut <i>dependency</i>. Urutannya penggunaan <i>dependency</i> digunakan untuk menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> yang lain. Sebuah <i>dependency</i> dilambangkan sebagai sebuah panah bertitik-titik.</p>	