BAB 2
TINJAUAN PUSTAKA

2.1 Game

Game adalah sebuah hiburan berbentuk permainan multimedia yang dibuat semenarik mungkin yang menggunakan media elektonik sebagai media untuk memainkannya. *Game* juga dapat diimplementasikan sebagai media pembelajaran dan edukasi yang memiliki tujuan spesifik sebagai alat pendidikan, sebagai contoh *game* dapat dimainkan untuk belajar mengenal warna, bentuk, huruf, angka dan menghitung. Dalam perkembangannya *game* juga dapat dimainkan untuk belajar bahasa asing.

Dikutip dari buku Cerdas Dengan *Game* yang ditulis oleh Samuel Henry yang mengutip dari Sharon Helmer Poggenpohl tahun 2002 "*Video game* adalah alat yang efektif untuk belajar karena game mampu menawarkan lingkungan hipotesis untuk siswa, dimana mereka dapat mengeksplorasi berbagai keputusan alternatif tanpa resiko kegagalan. Pemikiran dan tindakan digabungkan menjadi perilaku yang bertujuan mencapai suatu tujuan. *Video game* mengajari kita cara menyusun strategi, mempertimbangkan alternatif, dan berpikir fleksibel" [5].

2.2 Pengertian game Game

Beberapa ahli mengutarakan definisi dari *game* [6] adapun definisi dari *game* menurut beberapa ahli adalah sebagai berikut :

a. Albert Einstein
   "*Game* adalah bentuk investigasi paling tinggi" [6].
b. Agustunus Nilwan
   "*Game* merupakan permainan komputer yang dibuat dengan teknik dan metode animasi*"* [6].
c. Clark C. ABT
   "*Game* adalah kegiatan yang melibatkan keputusan pemain, berupaya mencapai tujuan dengan dibatasi oleh konteks tertentu" [6].

d. Eric Zimmerman & Katie Salen

"Sebuah permainan adalah sebuah sistem dimana pemain terlibar dalam konflik buatan, ditentukan oleh aturan, yang menghasilkan hasil yang terukur" [6].

e. Mitchell Wade

"*Game* adalah lingkungan pelatihan yang baik bagi dunia nyata dalam organisasi yang menuntut pemecahan masalah secara kolaborasi" [6].

f. John Naisbitt

"*Game* merupakan sistem partisipatoris dinamis karena *game* memiliki tingkat penceritaan yang tidak dimiliki film" [6].

2.3 Sejarah Game

Pada tahun 1996 *game* pertama kali ditemukan oleh Ralph Baer bersama 500 orang *team*-nya yang terdiri dari teknisi dan para insinyur, pada *game*nya Baer dapat menampilkan dua buah titik yang saling berkejar-kejaran dilayar. Tahun 1972 nama baru muncul pada dunia *game*, Nolan Bushel mendirikan sebuah perusahaan Atari dan membuat *game Arcade Pong*. Pada tahun 1988 era baru didunia *game* berkembang yang didominasi oleh perusahan Jepang. Nintendo memproduksi *video game* yang mencapai omset miliaran dolar. Pada *fase* ini lah perkembangan *video game* terus berkembang dan melewati beberapa *fase* dan merambah keberbagai perangkat seperti *game* PC, *game console*, dan *game online*.

Indonesia juga memiliki perkembangan sejarah. *Game* pertama di Indonesia adalah Divinekids yang diakui dan mendapatkan penghargaan resmi dari MURI (Museum Rekor Indonesia) sebagai *game* pertama Indonesia pada tahun 2004. Namun menurut sebagian besar pengamat *game* di Indonesia menyatakan bahwa *game* pertama di Indonesia adalah *game* devid [7].

2.4 Jenis-Jenis Game

Game memiliki beberapa jenis *genre* dan dapat dikelompokkan menjadi beberapa kelompok *genre* [8]. Berikut ini adalah beberapa jenis *genre game* dan *sub genre* pada *game.*

a. Action

Action Game adalah salah satu *genre* yang paling banyak diminati dan ditemukan. Pada *Action Game* biasanya dibutuhkan ketangkasan dan kelincahan pemain untuk bisa menyelesaikan setiap bagian didalam *game* tersebut. Adapun *sub game* pada *Action Game* adalah sebagai berikut :

1. *Platform Games*

   Platform Games atau biasa juga disebut dengan *Platformer* adalah sebuah *games* yang pemainnya diharuskan mencegah karakter jatuh dari *platform* atau pijakan pada *games* tersebut [8].

2. *Shooter Games*

   Shooter Games atau permainan tembak-tembakan adalah *sub genre* dari *action game* walaupun banyak yang mengatakan bahwa *shooter games* merupakan jenis game tersendiri. Pada *shooter game* biasanya pemain akan memaikan sebuah karakter yang menggunakan senjata. Tujuan dari *game* yang ber*genre shooter games* biasanya memiliki tujuan untuk menembak atau mengalahkan lawan untuk melanjutkan misi. Dalan *shooter games* juga memiliki beberapa tipe permainan yaitu : FPS (*First-Person Shooter),* TPS/3PS (*Third-Person Shooter*), MMOFPS (*Massively Multiplayer Online First-Person Shooter*) [8].

3. *Fighting Games*

   Fighting Games adalah permainan pertarungan satu lawan satu antara dua karakter. Jenis permainan ini biasanya akan dimainkan dengan konsol dengan mengkombinasikan beberapa tombol untuk menggerakan karakter saat bertarung [8].

4. *Stealth Games*

   Stealth Games adalah sebuah permainan dengan *genre action games* dimana pemain diharuskan menyelesaikan misi secara diam-diam tanpa diketahui oleh musuh pada *games* tersebut [8].

5. *Survival Games*

   Survival Games adalah sebuah permainan dimana para pemain diberikan sedikit peralatan dan diharuskan bertahan hidup dengan

beberapa skenario yang telah disediakan. Pemain pula dapat membuat peralatan sendiri dari berbagai alat yang ditemukan pada saat bermain [8].

b. Adventure

Game adventure merupakan sebuah *genre* yang menonjolkan alur cerita yang harus diselesaikan, dan biasanya ketika pemain telah menyelesaikan sebuah *event* pada *game adventure* pemain tidak dapat kembali ke *event* sebelumnya. Pada *genre adventure* pula biasanya memiliki sistem *Butterfly Effects* yang artinya adalah saat pemain menentukan pilihan maka pilihan tersebut akan mempengaruhi alur cerita dikemudian waktu [8].

c. Action-Adventure

Action Adventure adalah sebuah *genre games* yang menggabungkan *genre action* dan *genre adventure* dimana pemain diharuskan memiliki ketangkasan dan kecermatan dalam memainkannya dan pemain juga akan memilih sendiri alur cerita dari *game* tersebut sesuai pilihan dan tidakan saat bermain *game* tersebut [8].

d. RPG (*Role-Playing Game*)

Game RPG adalah sebuah permainan yang pemainnya akan memainkan tokoh atau karakter dan akan membuat sebuah cerita tersendiri. Para pemain juga akan memilih aksi dari karakternya masing-masing, dalam permainan RPG jarang sekali ada istilah menang atau kalah, tingkat keberhasilan dalam *game* RPG bergantung pada peraturan sistem yang diberlakukan [8].

e. Simulation

Game simulation adalah sebuah permainan yang mensimulasikan sebuah kejadian atau kehidupan nyata dari seorang karakter yang ada didalam *game* tersebuh. Akan tetapi terkadang terdapat beberapa *game simulation* yang mensimulasikan kejadian yang tidak terjadi pada dunia nyata [8].

f. Strategy

Game strategy adalah sebuah permainan yang mengharuskan para pemainnya berfikir cepat dan cekatan dalam penyelesaikan sebuah permasalahan atau permainan. Dalam *game strategy* biasanya para pemain

akan mengatur terlebih dahulu strategi sebelum permainan dimulai, dan ada pula permainan yang pemainnya harus menyusun strategi secara *real-time* atau pada saat permainan berjalan [8].

g. Sports

Game sports adalah sebuah permainan olah raga yang para pemainnya akan memainkan banyak karakter dalam sekali main, didalam *game sports* terkadang pula pemain hanya akan memainkan seorang karakter saja dan biasanya *game* yang memainkan seorang pemain saja bersifat balapan atau *racing* [8].

h. Idle Gaming

Idle gamming adalah sebuah *genre game* yang baru muncul dan terkenal dikalangan para *gamers* sebutan untuk para pemain *game*, *game* ini menciptakan sensasi baru karena para pemainnya tidak perlu bersusah payah untuk memainkannya cukup dengan menyentuh karakter pada *game* tersebut untuk memainkannya [8].

2.5 Komponen Game

Menurut Teresa Dillon dalam sebuah *game* terdapat komponen-komponen yang menjadi penyusun sebuah *game* bisa dimainkan [9], adapun komponen-komponen yang menjadi penyusunya adalah sebagai berikut :

a. *Game Rule*

Aturan dalam sebuah permainan dalam menjalankan karakter ataupun obyek apapun yang ada didalam *game* tersebut.

b. *Plot*

Berisi tentang informasi dari hal-hal yang akan dilakukan oleh pemain dalam sebuah *game* secara rinci dengan tujuan yang harus dicapai.

c. *Theme*

Berisikan sebuah pesan moral yang akan disampaikan oleh karakter didalam game kepada pemain.

d. *Object*

Sebuah hal penting yang biasanya digunakan pemain dalam memecahkan masalah.

e. *Text, Graphic, Sound*

Merupakan beberapa elemen yang akan ditampilkan oleh *game* dengan mengkombinasikan beberapa elemen tersebut.

f. *Animation*

Sebuah tampilan gambar bergerak yang akan menjadikan *game* lebih hidup dan dapat pula menyampaikan pesan pada *game* tersebut.

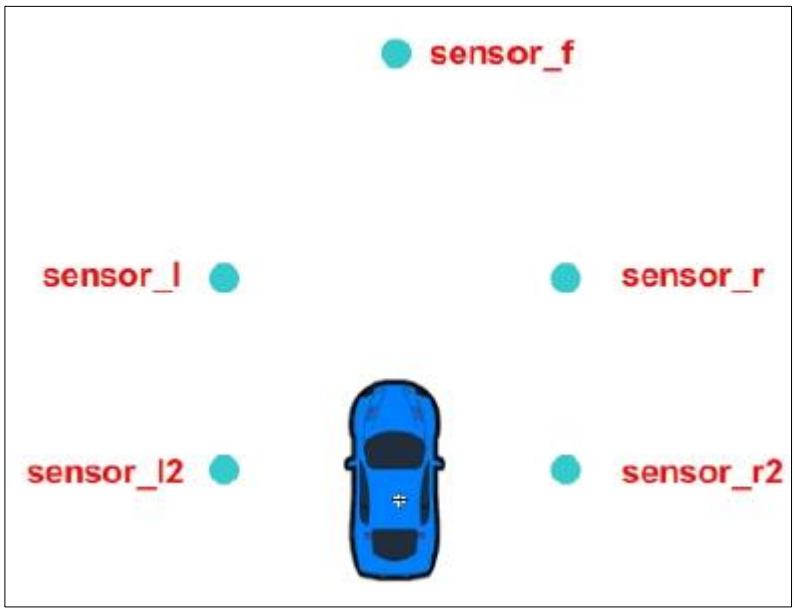
g. *User Interface*

Tampilan game yang berfungsi untuk menghubungkan user dengan *game*.

2.6 Metode Trigger Detection

Metode *Trigger Detection* biasa digunakan untuk menggerakan NPC kendaraan pada permainan *virtual* lalu lintas. Metode ini memanfaatkan sensor pemicu (*trigger*) untuk mendeteksi keberadaan obyek disekitarnya. Mekanisme yang mendukung *trigger* ini adalah ditambahkan disekitar NPC (*Non-Player Character*).

Sensor yang ditambahkan akan dihitung secara matematis untuk menentukan apakah sensor mengenai obyek lain seperti tembok, kendaraan lain atau obyek lainnya yang berada di lingkungan *virtual*. Data yang didapatkan akan menentukan gerakan dari NPC selanjutnya [10].



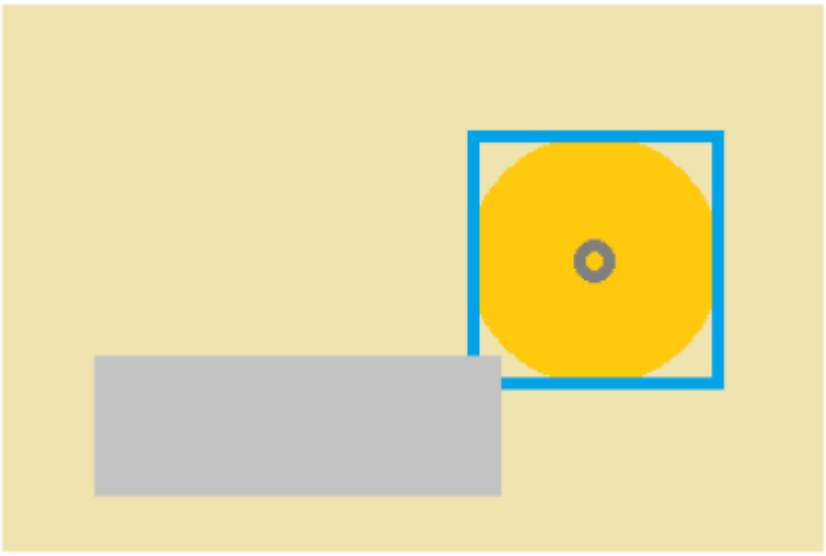
Gambar 2.1 Peletakan Posisi Sensor

2.7 Metode Collision Detection

Collision detection adalah proses pendeteksian tabrakan antara dua obyek. Dalam *game* sendiri tabrakan tidak hanya terjadi antara dua obyek, tetapi bisa terjadi antara satu obyek dengan banyak obyek. *Collision detection* yang akurat dibutuhkan dalam *game* karena setelah menentukan terjadinya tabrakan maka pembuat *game* harus menentukan apa yang terjadi pada benda yang ditabrak dan yang menabrak.

Manfaat dari *collision detection* adalah menentukan dari suatu obyek dengan obyek lain sehingga tidak ada obyek yang saling menembus, maka game yang akan dibuat memiliki kesamaan realita yang ada. Penggunaan *collision detection* ini dibedakan menajdi dua jenis metode, yaitu menggunakan *hitTestObject* dan *hitTestPoint* [11]. Perbedaannya adalah :

a. Metode *hitTestObject* digunakan untuk memberi perintah ketika *movieclip* terjadi tabrakan dengan *movieclip* lain.
b. Metode *hitTestPoint* digunakan untuk mendeteksi obyek dalam ruang suatu gambar, metode ini sangat diperlukan dalam pembuatan *game* dengan contoh karakter musuh terkena tembakan oleh karakter utama dan nyawanya berkurang.



Gambar 2.2 Contoh *Collision Detection*

2.8 Regulasi Aturan Pemerintah Mengenai Satwa Endemik

Pemerintah Republik Indonesia telah mengeluarkan aturan dan hukum mengenai satwa liar dan tumbuhan yang dilindungi oleh pemerintah Republik Indonesia. Aturan tersebut dituliskan pada Undang-Undang Republik Indonesia Nomor 5 Tahun 1990 Tentang Konservasi Sumber Daya Alam Hayati dan Ekosistemnya pada Pasal 21 ayat (1) dan (2) [12]dan juga Peraturan Pemerintah Republik Indonesia Nomor 7 Tahun 1999 [13] Tentang Pengawetan Jenis Tumbuhan Dan Satwa pada Pasal 5 ayat (1).

Adapun hukuman yang akan dikenakan kepada para pelanggar Undang-Undang tersebut diatur pada Undang-Undang Republik Indonesia Nomor 5 tahun 1990, Pasal 40 Ayat (1), (2), (3) dan (4) yang berbunyi sebagai berikut :

(1) "Barang siapa dengan sengaja melakukan pelanggaran terhadap ketentuan sebagaimana dimaksud dalam Pasal 19 ayat (1) dan Pasal 33 ayat (1) dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan denda paling banyak Rp. 200.000.000,00 (dua ratus juta rupiah)."

(2) "Barang siapa dengan sengaja melakukan pelanggaran terhadap ketentuan sebagaimana dimaksud dalam Pasal 21 ayat (1) dan ayat (2) serta Pasal 33 ayat (3) dipidana dengan pidana penjara paling lama 5

(lima) tahun dan denda paling banyak Rp. 100.000.000,00 (seratus juta rupiah).”

(3) “Barang siapa karena kelalaiannya melakukan pelanggaran terhadap ketentuan sebagaimana dimaksud dalam Pasal 19 ayat (1) dan Pasal 33 ayat (1) dipidana dengan pidana kurungan paling lama 1 (satu) tahun dan denda paling banyak Rp. 100.000.000,00 (seratus juta rupiah).”

(4) “Barang siapa karena kelalaiannya melakukan pelanggaran terhadap ketentuan sebagaimana dimaksud dalam Pasal 21 ayat (1) dan ayat (2) serta Pasal 33 ayat (3) dipidana dengan pidana kurungan paling lama 1 (satu) tahun dan denda paling banyak Rp. 50.000.000,00 (lima puluh juta rupiah).”

2.9 UML (Unified Modeling Language)

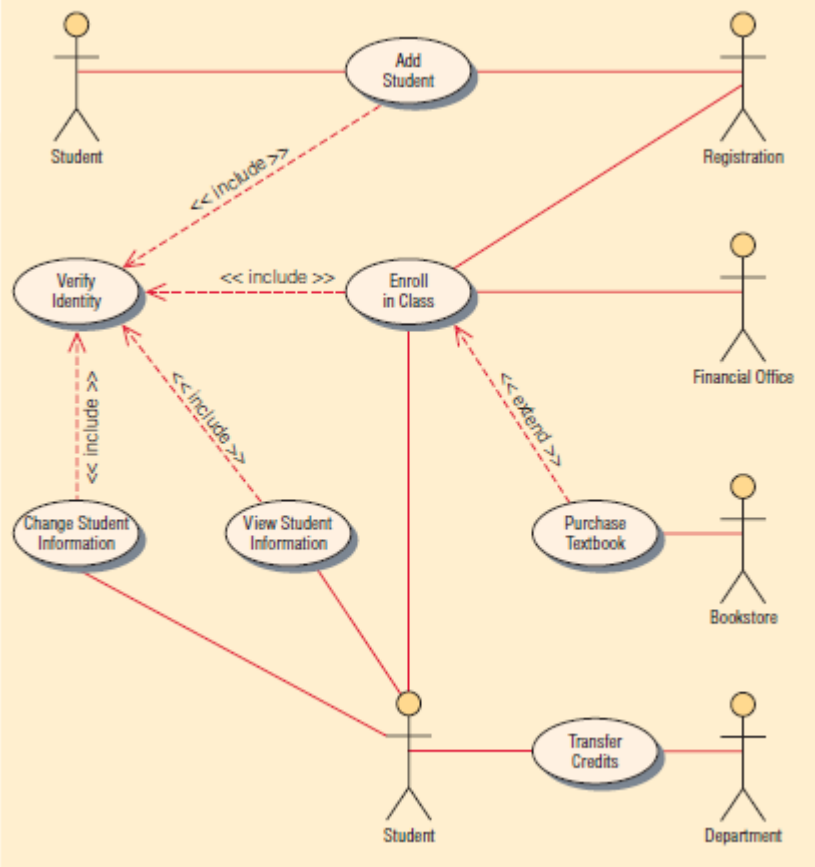
Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi *standart* dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem .

Unified Modelling Language (UML) adalah alat yang dapat meningkatkan kualitas analisis dan desain sistem, dengan demikan dapat membantu menciptakan sistem yang berkualitas tinggi. Alat bantu pada UML mencakup diagram yang memungkinkan orang untuk memvisualisasikan konstruksi dari sistem yang berorientasi *Object*.

Terdapat dua diagram utama dalam UML yaitu diagram struktural dan diagram perilaku. Diagram structural digunakan untuk menggambarkan hubungan antar kelas meliputi diagram kelas, diagram obyek, diagram komponen dan diagram penyebaran. Sedangkan diagram perilaku digunakan untuk menggambarkan *Use Case* yang menjelaskan bagaimana aktor menggunakan sistem, diagram perilaku meliputi *Use Case*, *Sequence Diagram*, *Communication Diagram*, *Statechart Diagram* dan *Activity Diagram* [14].

2.9.1 Use Case Diagram

Use Case Diagram adalah sebuah model yang menunjukkan tampilan sistem dari perspektif pengguna, sehingga dapat menggambarkan apa yang dilakukan sistem tanpa menjelaskan bagaimana sistem melakukannya. Dalam *use case* selalu terdapat tiga komponen penting yaitu *actor*, *event* dan *use case* [14].

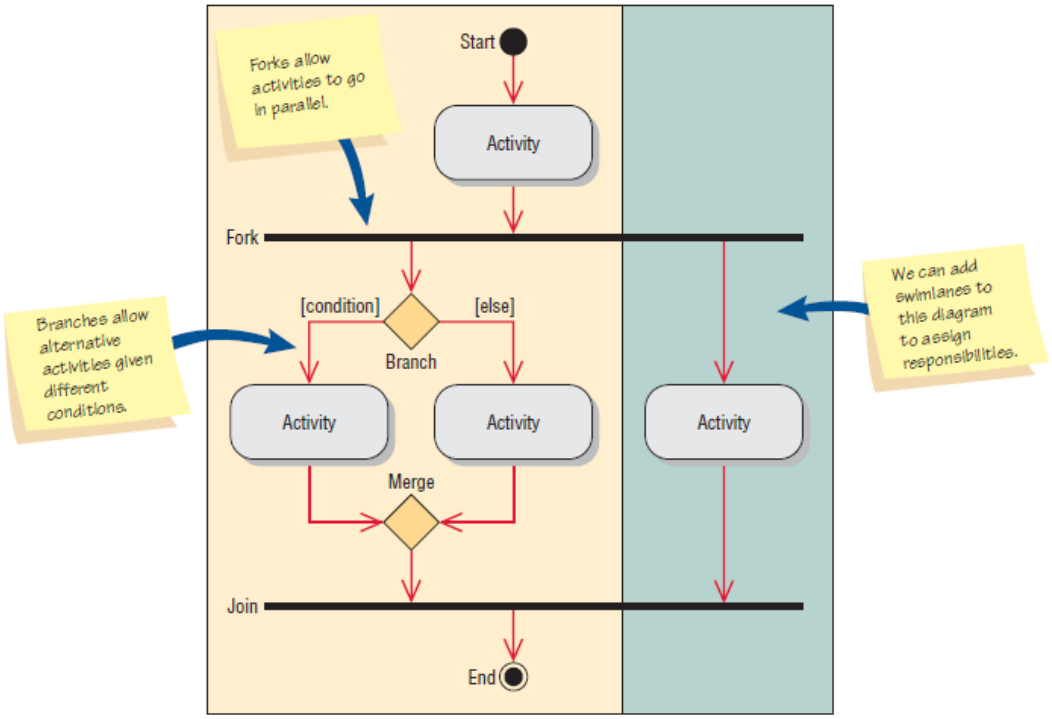


Gambar 2.3 Contoh *Use Case Diagram*

2.9.2 Activity Diagram

Activity Diagram adalah diagram yang menunjukkan urutan aktivitas dari suatu proses, baik berurutan maupun parallel serta keputusan yang dibuat. *Activity diagram* biasanya digunakan hanyak untuk satu *use case* dan dapat menunjukkan scenario yang berbeda-beda.

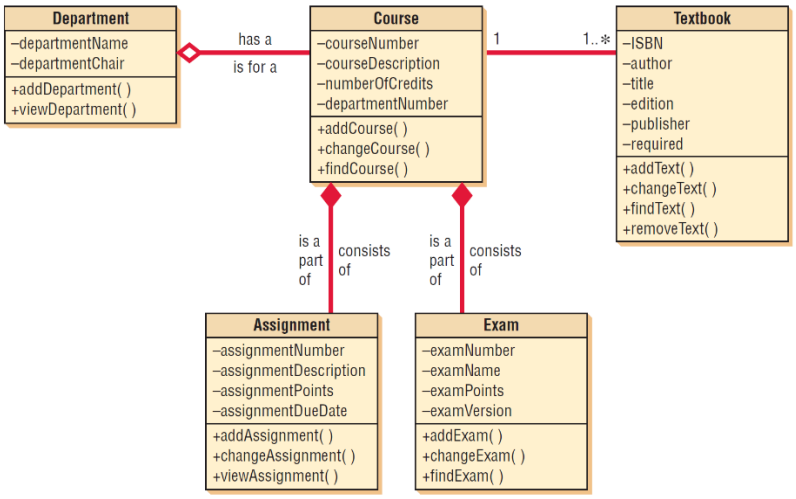
Penggunaan *activity diagram* dilakukan untuk membantu memahami *use case*, membaca aliran kontrol yang kompleks, kebutuhan untuk memodelkan alur kerja dan perlunya menunjukkan semua skenario [14].



Gambar 2.4 Contoh *Activity Diagram*

2.9.3 Class Diagram

Dalam UML *class diagram* berfungsi untuk menunjukkan fitur statis dari sistem dan tidak mewakili suatu proses tertentu, *class diagram* juga menunjukkan sifat dari hubungan antar kelas selain itu *class diagram* juga menunjukkan persyaratan penyimpanan data serta persyaratan pengolahan. Kelas memiliki tiga area pokok yaitu nama kelas, atribut dan *method* atau operasi [14].

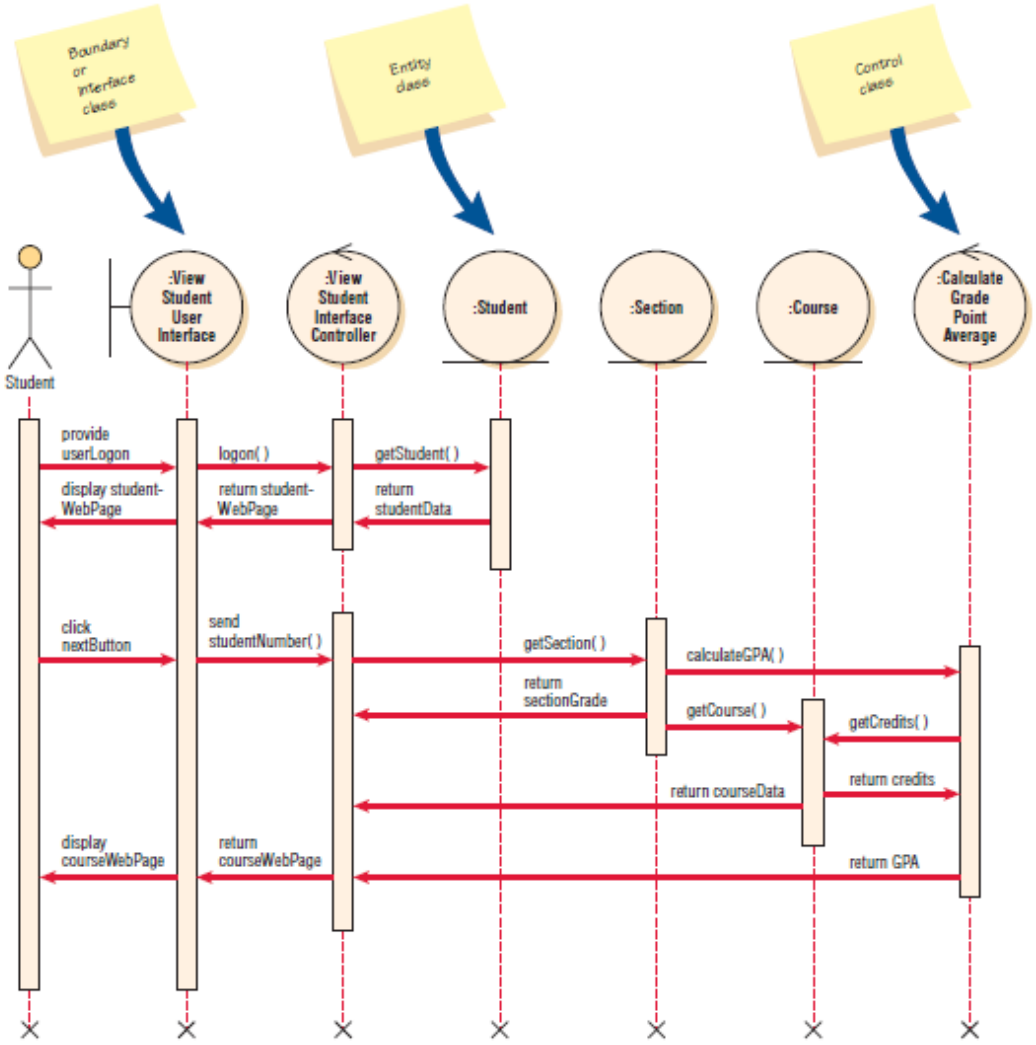


Gambar 2.5. Contoh *Class Diagram*

2.9.4 Sequence Diagram

Sequence diagram dapat menggambarkan perilaku obyek pada *use case* dengen mendeskripsikan waktu hidup obyek dan pesan yang dikirimkan dan diterima antar obyek. *Sequence diagram* dapat digunakan untuk menerjemahkan skenario *use case* ke dalam alat cisual untuk menganalisa sistem.

Dalam *sequence diagram* biasanya diawali dengan menunjukkan aktor dan kelas yang berinteraksi untuk proses tertentu. Dalam *sequence diagram* terdapat beberapa simbol yaitu *actor* atau proses lain yang berinteraksi dengan sistem yang dibuat diluar sistem tersebut, *lifeline* atau garis hidup yang menyatakan kehidupan suatu obyek, *object* menyatakan obyek yang berinteraksi, waktu aktif menyatakan obyek dalam keadaan aktif dan berinteraksi pesan, pesan menyatakan spesifikasi dari komunikasi antar obyek yang memuat informasi-informasi tentang aktifitas yang terjadi [14].



Gambar 2.6 Contoh Sequence Diagram

2.10 Aplikasi Pembangunan Perangkat Lunak

Aplikasi atau *tools* yang akan digunakan dalam pembangunan perangkat lunak pada penelitian ini akan menggunakan beberapa aplikasi. Adapun aplikasi yang akan digunakan adalah sebagai berikut :

2.10.1 Construct 2

Construct 2 adalah sebuah *tools* yang berguna untuk membuat game yang dikhususkan pada tampilan dua dimensi dengan *game* berbasis HTML 5. *Construct 2* tidak menggunakan bahasa pemograman khusus karena dalam pembuatan game pada *Construct 2* menggunakan *Event and Action*. *Construct 2* dapat diimplementasikan kepada beberapa *platform* seperti *IOS* dan *Android* [15].

Construct 2 juga memeliliki beberapa kelebihan dibandingkan dengan *tools* pembuatan *game* lain yaitu :



Gambar 2.7 *Construct 2 by Scirra*

a. *Quick and Easy*

Dalam segi tampilan *Construct 2* menampilkan dengan beragam kelebihan seperti antarmuka *Ribbon* yang cepat dan mudah untuk dipahami. Pada tampilan *layout editor* pula menampilkan antarmuka yang sangat efisien karena menampilkan apapun yang dilakukan oleh pembuat game dengan istilah *what you see is what you get* dengan begitu segala perubahan akan ditampilkan.

b. *Powerfull Event System*

Construct 2 dibuat untuk mempermudah para pemakainya dalam membuat game karena tidak menggunakan bahasa pemograman dan hanya menggunakan *Event and Action* atau *layout* pada aplikasi *Construct 2* adalah *EventSheet*. Pada *EventSheet* sendiri berisikan pernyataan kondisi. Dimana jika kondisi terpenusi maka tindakan atau aksi dapat dilakukan

c. *Flexible Behaviors*

Pada *Construct 2* menyediakan *Behaviors* yang memudahkan untuk menampah kemampuan obyek, meningkatkan produktivitas dan mempercepat pembangunan.

d. *Instant Preview*

Pada *Construct 2* mempunyai kelebihan yaitu mempercepat tahap kompilasi atau menampilkan hasil pembuatan *game*. Dalam hal ini *Construct 2* akan menampilkan hasil kompilasi pada *web browser.*

e. *Stunning Visual Effects*

Pada *Construct 2* juga memiliki efek *visual* yang bagus dikarenakan terdapat lebih dari 70 efek yang berbasis WebGL untuk melakukan *warp, distort, blend, blur, mask, re-color* dan lain-lain. Dengan hal ini diharapkan para pemain *game* yang dibangun menggunakan *Construct 2* akan mendapatkan pengalaman terbaik dari sebuah *game.*

f. *Multiplatform Export*

Kelebihan *Construct 2* yang sangat menonjol adalah *Construct 2* dapat dipublikasi dengan berbagai *platform* hanya dengan satu proyek yang telah dibuat seperti *Web, Desktop PC, Mac, Linux, Windows Phone, Windows 8 Store, IOS dan Android.*

g. *Easy Extensibility*

Construct 2 memiliki lebih dari 20 *built-in plugin,*lebih dari 20 *Behaviors* dan lebih dari 70 *Visual Effects* yang menjadikan sangat spesial.

2.10.2 Photoshop CS 6

Photoshop CS 6 adalah salah satu produk yang dibuat oleh *Adobe Systems* yang dibuat untuk proses pengolahan citra, pengeditan gambar dan pembuatan efek-efek. CS sendiri adalah singkatan dari *Creative Suite* dan *Photoshop CS 6* adalah versi Photoshop generasi ke tiga belas yang dirilis pada 21 maret 2012. *Adobe Systems* sendiri adalah sebuah *vendor* internasional yang bergerak pada bidang pengembangan perangkat lunak grafis dan multimedia.

Photoshop sendiri adalah sebuah program yang banyak digunakan untuk beberapa sistem operasi seperti *Windows* dan *Mac OS*. *Photoshop* juga merupakan program pengolah gambar yang memiliki standart internasional maka dari itu

Photoshop banyak digunakan diberbagai Negara sebagai program pengolahan gambar dan citra [16].



Gambar 2.8 Logo *Adobe Photoshop CS6*

2.10.3 Adobe Illustrator

Adobe Illustrator merupakan aplikasi yang hampir sama fungsinya dengan Photoshop dan dikembangkan pula oleh *Adobe Sistems* sebagai perusahaan pengembang perangkat lunak untuk desain grafis. Sama seperti *Photoshop, Adobe illustrator* juga merupakan aplikasi yang digunakan untuk meng*edit* dan membuat desain-desain lain hanya saja perbedaannya adalah *Adobe Illustrator* digunakan untuk membuat desain dengan format *vector*.

Vector sendiri adalah format gambar yang disusun berdasarkan disusun dari beberapa garis dan melalui perhitungan matematis yang akan menghasilkan gambar dengan kualitas yang bagus, dan ketika gambar diperbesar ataupun diperkecil maka gambar tidak akan rusak atau pecah [17].



Gambar 2.9 Logo *Adobe Illustrator*

2.10.4 Intel XDK

Intel XDK adalah *development kit* keluaran dari Intel untuk membuat aplikasi *native* untuk perangkat *mobile* menggunakan teknologi *web* seperti HTML5, CSS, dan JavaScript. Aplikasi *web* dikompilasi dengan menggunakan *platform Cordova* dan *PhoneGap deserver online* untuk membuat aplikasi *hybrid* yang *cross-platform*. Aplikasi *Intel XDK* sendiri adalah aplikasi *cross-platform* yang tersedia untuk sistem operasi *Windows*, *Linux* dan *OSX*. Dengan teknologi seperti ini, *developer* dapat membuat satu aplikasi yang dapat berjalan dibanyak *platform mobile* hanya menggunakan satu basis kode saja [18].



Gambar 2.10 Logo *Intel XDK*

2.10.5 PhoneGap

PhoneGap adalah sebuah project *OpenSource* yang digunakan untuk membuat sebuah aplikasi *mobile application*. Memanfaatkan teknologi *PhoneGap* membuat kita dapat menggunakan teknologi web seperti HTML, CSS dan JavaScript menjadi sebuat aplikasi *mobile*.

Phonegap pertama kali dikembangkan oleh *iPhoneDevCamp* pada kegiatan hackathon pada tahun 2008. Banyak sekali yang ikut berkontribusi mengembangkan phonegap sebab setiap *iphone developer* yang masih *newbie* pasti menggunakan *Objective-C*. Sementara tidak semua *web developer* memahami *Objective-C*. Satu pertanyaan *universal* yang mendorong teciptanya phonegap adalah apakah mungkin seseorang menciptakan sebuah *framework* yang bisa membantu web developer untuk menggunakan kemampuan HTML, CSS dan Javascript agar bisa berinteraksi dengan *native device* seperti *iPhone, Blackberry, Android etc* untuk mengakses *native features* yang dimiliki setiap *native device* seperti *camera, accelerometer, gyroscope* dst.

Seiring waktu berlalu akhirnya *PhoneGap* dikembangkan oleh perusahaan Nitobi dan pada akhirnya di akuisisi oleh *adobe*, sebagai bagian dari suatu diakuisisi *source code* dari *phonegap* akhirnya didonasikan kepada (ASF) *Apache Software Foundation* yang membuat *PhoneGap* menjadi *Open Source* [19].



Gambar 2.11 Logo *PhoneGap*

2.10.6 GameBench

GameBench adalah sebuah perangkat lunak yang dibangun untuk mengetahui performa sebuah aplikasi *game*, dengan menguji penggunaan RAM, CPU, GPU, dan menghitung FPS. *GameBench* bekerja pada *background* pada saat aplikasi *game* yang akan dinilai [20].



Gambar 2.12 Logo *GameBech*