

BAB 2

LANDASAN TEORI

2.1. Kebudayaan

Budaya menurut Kamus Besar Bahasa Indonesia(KBBI) adalah sebuah hasil pikiran, akal budi. Kebudayaan adalah hasil kegiatan dan penciptaan batin (akal budi) manusia (seperti kepercayaan, kesenian, dan adat istiadat)[1]. Kebudayaan dapat pula didefinisikan sebagai keseluruhan pengetahuan manusia yang digunakannya untuk memahami dan menginterpretasi lingkungan dan pengalamannya, serta menjadi pedoman bagi tingkah lakunya. Kebudayaan merupakan milik bersama anggota suatu masyarakat atau golongan sosial tertentu yang penyebarannya kepada anggota-anggotanya dan pewarisannya kepada generasi berikutnya dilakukan melalui proses belajar dan dengan menggunakan simbol-simbol yang terwujud dalam bentuk yang terucapkan ataupun yang tidak termasuk berbagai peralatan yang dibuat oleh manusia[6].

2.2. Android

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri dan digunakan oleh berbagai macam perangkat *mobile*. Awalnya Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk telepon seluler. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorium dari 34 perusahaan *hardware*, *software*, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. Pada saat perilisan perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler[7].

2.3. Multimedia

Arti multimedia berbeda dari sudut pandang orang yang berbeda. Secara umum, multimedia berhubungan dengan penggunaan lebih dari satu macam cara untuk menyajikan informasi. Rekaman musik hanya menggunakan suara (mungkin disebut “unimedia”), *music video* adalah bentuk multimedia karena informasi menggunakan suara dan video.

Dalam dunia komputer, multimedia berhubungan dengan perangkat lunak yang digunakan dalam pengembangan dengan lebih dari satu cara untuk menyampaikan informasi kepada pengguna, seperti teks dan suara. Secara teknis, multimedia dalam komputer bukan merupakan hal baru, karena integrasi antara teks dan gambar merupakan komponen utama dalam banyak perangkat lunak aplikasi sudah lama dilakukan. Informasi yang menyajikan teks dengan diagram merupakan salah satu contoh yang telah banyak dibuat [8].

2.3.1. Komponen Multimedia

Multimedia terdiri dari beberapa komponen yaitu [8]:

1. Teks

Teks merupakan dasar dari pengolahan kata dan informasi berbasis multimedia. Dalam kenyataannya, multimedia menyajikan informasi kepada audiens dengan cepat, karena tidak diperlukan membaca secara rinci dan teliti.

2. Grafik

Secara umum grafik berarti gambar garis (*line drawing*). Manusia sangat berorientasi pada visual (*visual oriented*), dan gambar merupakan sarana yang sangat baik untuk menyajikan informasi. Grafik merupakan komponen penting dalam multimedia.

3. Gambar

Secara umum gambar berarti gambar raster (*halftone drawing*), seperti foto. Basis data karyawan dengan atribut seperti nama, alamat dan lainnya akan lebih efektif bila foto karyawan yang bersangkutan dapat ditampilkan. Demikian juga fotofoto seperti gedung dan lainlain sangat memerlukan penyimpanan yang besar.

Hal ini menyebabkan aplikasi multimedia disimpan dalam media penyimpanan yang cukup besar kapasitasnya seperti CD-ROM.

4. Animasi

Animasi menggambarkan objek yang bergerak agar terlihat hidup. Membuat animasi berarti menggerakkan gambar seperti kartun, lukisan, tulisan dan lain-lain. Animasi mulai dikenal sejak media televisi mulai menyajikan gambar-gambar bergerak yang berasal dari rekaman kamera maupun hasil seorang animator. Animasi sangat baik untuk presentasi, permodelan, dokumentasi dan lain-lain.

5. Suara

Suara dapat lebih memperjelas pengertian yang ditampilkan dengan cara lain. Contohnya, narasi merupakan kelengkapan dari penjelasan yang dilihat melalui video. Suara dapat memberi lebih penjelasan karakteristik suatu gambar, misalnya musik dan suara efek.

2.4.Game

Game merupakan permainan video yang pada awalnya dimainkan pada komputer dan salah satu media hiburan yang banyak digemari oleh semua kalangan usia. Permainan komputer telah berevolusi dari sistem grafis sederhana sampai sistem grafis yang kompleks dan mutakhir. *Game* 3 dimensi (3D) merupakan salah satu permainan yang memiliki sistem grafis kompleks [9]. *Game* terdiri atas sekumpulan peraturan yang membangun situasi bersaing dari dua sampai beberapa orang atau kelompok dengan memiliki strategi yang dibangun untuk memaksimalkan kemenangan sendiri atau pun untuk meminimalkan kemenangan lawan. Peraturan-peraturan menentukan kemungkinan tindakan untuk setiap pemain, sejumlah keterangan diterima setiap pemain sebagai kemajuan bermain, dan sejumlah kemenangan atau kekalahan dalam berbagai situasi [10]. *Game* pertama kali muncul pada tahun 1952. Dimulai dari seorang mahasiswa Universitas Cambridge bernama A.S Gouglas membuat game pertama yang berupa permainan OXO dalam bentuk grafik. Permainan ini dibuat dan dikembangkan untuk

mendemonstrasikan tesisnya tentang interaksi antara manusia dan komputer. *Game* mempunyai perkembangan pesat pada beberapa tahun terakhir ini. [9]

2.4.1 Jenis Jenis Game

Berikut ini adalah jenis-jenis dari game:

Berdasarkan *platform* atau alat yang digunakan[11]:

1. *Arcade Game*

Arcade Game yaitu game dengan mesin yang memang khusus didesain untuk jenis game tertentu dan memiliki fitur seperti pistol, kursi khusus sensor gerakan, sensor injak dan stir mobil, dan lain-lain.

2. *PC Game*

PC Game yaitu *game* yang dimainkan menggunakan *Personal Computer*.

3. *Console Game*

Console Game yaitu game yang dimainkan menggunakan *console* tertentu seperti Playstation 2, Playstation 3, XBOX 360, dan Nintendo Wii.

4. *Handheld Game*

Handheld Game yaitu *game* yang dimainkan di *console* khusus *game* yang dapat dibawa kemana-mana, contoh Nintendo DS dan Sony PSP.

5. *Mobile Game*

Mobile Game yaitu *game* yang dapat dimainkan khusus untuk *mobile phone*, *smartphone* atau PDA.

Berdasarkan genre permainannya:

1. *Shooting*

Shooting merupakan salah satu *genre* dalam *game* yang biasanya mensimulasikan tembak-tembakan, atau hajar-hajaran bisa juga tusuk-tusukan, tergantung cerita dan tokog di dalamnya, *game* jenis ini sangat memerlukan kecepatan refleks, koordinasi mata-tangan, juga *timing*, inti dari *game* jenis ini adalah tembak. Contoh : Medal of Honor, Counter Strike, dan Fallout.



Gambar 2.1. Contoh Game Shooting Counter Strike Global Offensive

2. *Fighting*

Fighting adalah jenis game yang memerlukan kecepatan refleks dan koordinasi mata-tangan, tetapi inti dari *game* ini adalah penguasaan *skill* karakter yang dimainkan, pengenalan karakter dan *timing* sangatlah penting, dan berbeda seperti *game* aksi pada umumnya yang hanya melawan *Artificial Intelligence* atau melawan komputer saja, pemain jenis *Fighting Game* ini baru teruji kemampuan sesungguhnya dengan melawan pemain lainnya. Contoh : Mortal Kombat, dan Tekken.



Gambar 2.2. Contoh Game Fighting Mortal Kombat X

3. *Adventure*

Adventure yaitu *game* yang menekankan pada jalan cerita dan kemampuan berpikir pemain dalam menganalisa tempat secara visual, memecahkan teka-teki maupun menyimpulkan rangkaian peristiwa dan percakapan karakter hingga penggunaan benda-benda tepat pada tempat yang tepat. Termasuk didalamnya petualangan dengan teks atau sistem *point and click*. Contoh : *Earthbound*, dan *Kings Quest*.



Gambar 2.3. Contoh Game Adventure Earthbound

4. *Simulation*

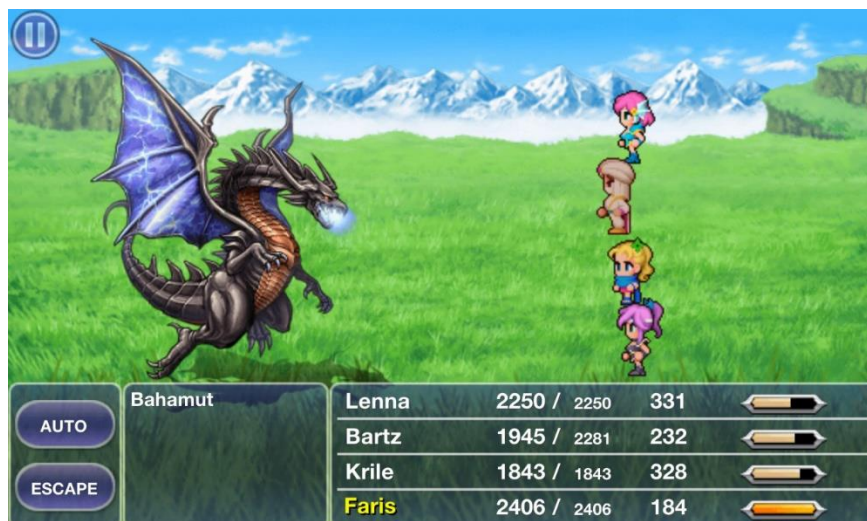
Simulation yaitu *game* yang menggambarkan dunia didalamnya sedekat mungkin dengan dunia nyata dan memperhatikan detail berbagai faktor. Dari mencari jodoh dan pekerjaan, membangun rumah, gedung hingga kora, mengatur pajak dan dana kota hingga keputusan memecat atau menambah karyawan. Dunia kehidupan rumah tangga sampai bisnis membangun konglomerasi. *Game* jenis ini membuat pemain harus berpikir untuk mendirikan, membangun dan mengatasi masalah dengan menggunakan dana yang terbatas. Contoh : *The Sims*, *Flight Simulator*, dan *Sim City*



Gambar 2.4. Contoh Game Simulation The Sims 4

5. *Role Playing*

Role Playing yaitu *game* yang memiliki penekanan pada tokoh/peran perwakilan pemain di dalam permainan, yang biasanya adalah tokoh utamanya, dimana seiring kita memainkannya, karakter tersebut dapat berubah dan berkembang ke arah yang diinginkan pemain dalam berbagai parameter yang biasanya ditentukan dengan naiknya level, baik dari status kepintara, kecepatan, dan kekuatan karakter dan lain-lain. Contoh : Final Fantasy, Dragon Age, dan The Witchers.



Gambar 2.5. Contoh Game Role Playing Final Fantasy FFV

6. Strategy

Strategy yaitu *game* yang memerlukan keahlian berpikir dan memutuskan setiap gerakan secara hati-hati dan terencana. *Game* strategi biasanya memberikan pemain atas kendali tidak hanya satu orang tapi minimal sekelompok orang dengan berbagai jenis tipe kemampuan. Pemain *game* strategi melihat dari sudut pandang lebih meluas dan lebih kedepan dengan waktu permainan yang lebih lama dan santai. Contoh : Warcraft, dan Age of Empire.



Gambar 2.6. Contoh Game Strategy Warcraft III

7. Sports

Sports yaitu *game* yang dibuat semirip mungkin dengan olahraga yang ada di dunia nyata walau kadang ada yang menambah unsur fiksi dalam *game* olahraga tersebut. Contoh : Fifa, dan NBA



Gambar 2.7. Contoh Game Sports FIFA 19

8. *Racing*

Racing yaitu *game* yang memungkinkan kita untuk mengendalikan sebuah kendaraan untuk memenangkan sebuah kompetisi balap. Contoh : Need for Speed, Moto GP, dan Gran Turismo



Gambar 2.8. Contoh Game Racing Need for Speed Most Wanted

9. *Musical*

Musical yaitu *game* yang memainkan suatu alat musik. Contoh : Gitar Hero, dan Synthesia



Gambar 2.9. Contoh Game Musical

2.4.2 Game Edukasi

Game edukasi adalah *game* digital yang dirancang untuk pengayaan pendidikan (mendukung pengajaran dan pembelajaran), menggunakan teknologi multimedia interaktif. Perancangan yang baik haruslah memenuhi kriteria dari *education game* itu sendiri. Berikut ini adalah beberapa kriteria dari sebuah *education game* menurut Hurd dan Jenuings, yaitu : [10]

1. Nilai Keseluruhan (*Overall Value*)

Nilai keseluruhan dari suatu *game* terpusat pada desain dan panjang durasi *game*. Aplikasi yang dibangun haruslah menarik dan interaktif, salah satunya dengan desain yang baik. Untuk penentuan panjang durasi dapat menggunakan fitur *timer*.

2. Dapat Digunakan (*Usability*)

Mudah digunakan dan diakses adalah poin penting bagi pembuat *game*. Aplikasi ini merancang sistem dengan *interface* yang *user friendly* sehingga *user* dapat dengan mudah mengakses aplikasi.

3. Keakuratan (*Accuracy*)

Keakuratan diartikan sebagai bagaimana kesuksesan model/gambaran sebuah *game* dapat dituangkan ke dalam percobaan atau perancangannya. Perancangan aplikasi harus sesuai dengan model *game* pada tahap perencanaan.

4. Kesesuaian (*Appropriateness*)

Kesesuaian dapat diartikan bagaimana isi dan desain *game* dapat diadaptasikan terhadap keperluan *user* dengan baik. Dengan menyediakan menu dan fitur yang diperlukan oleh *user* untuk membantu pemahaman *user* dalam menggunakan aplikasi.

5. Relevan (*Relevance*)

Relevan artinya dapat mengaplikasikan isi *game* ke target *user*. Agar dapat relevan terhadap *user*, sistem harus membimbing mereka dalam pencapaian tujuan pembelajaran. Desain yang digunakan harus sesuai dengan tujuan *user*.

6. Objektivitas (*Objectives*)

Objektivitas menentukan tujuan *user* dan kriteria dari kesuksesan atau kegagalan. Dalam aplikasi ini objektivitas adalah usaha untuk mempelajari hasil dari permainan.

7. Umpan Balik (*Feedback*)

Untuk membantu pemahaman *user* bahwa permainan (*performance*) mereka sesuai dengan objek *game* atau tidak, *feedback* harus disediakan.

2.5. UML (Unified Modeling Language)

UML (*Unified Modeling Language*) adalah bahasa grafis untuk mendokumentasi, menspesifikasikan, dan membangun sistem perangkat lunak. UML berorientasi objek, menerapkan banyak level abstraksi, tidak bergantung proses pembangunan, tidak bergantung bahasa dan teknologi, pemaduan beberapa notasi di beragam metodologi, usaha bersama dari banyak pihak, didukung oleh kakas-kakas yang diintegrasikan lewat XML. Standar UML dikelola oleh OMG (*Object Management Group*).

UML menyediakan sejumlah diagram untuk mengekspresikan pemodelan berorientasi objek yang dilakukan. Kegunaan diagram pada pemodelan adalah untuk formalisasi ekspresi model objek secara koheren, presisi dan mudah dirumuskan. UML yang dibahas pada bab ini adalah *Use Case Diagram*, *Class Diagram* dan *Sequence Diagram* [12].

2.5.1. Use Case Diagram

Use Case Diagram merupakan salah satu diagram untuk memodelkan aspek perilaku sistem. Masing-masing diagram *use case* menunjukkan sekumpulan *use case*, *actor* dan hubungannya. Diagram ini sangat penting untuk memvisualisasikan, menspesifikasikan dan mendokumentasikan kebutuhan perilaku sistem. *Use case* adalah interaksi antara *actor* eksternal dan sistem, hasil yang dapat diamati oleh *actor*, berorientasi pada tujuan, dideskripsikan di diagram *use case* dan teks. Diagram ini melibatkan :

1. Sistem yaitu sesuatu yang hendak kita bangun
2. *Actor*, entitas-entitas luar yang berkomunikasi dengan sistem
3. *Use case* adalah fungsionalitas yang dipersepsi oleh *actor*
4. Relasi adalah relasi antara *actor use case*

Elemen pada *use case diagram* adalah :

1. *Actor*

Actor adalah pemakai sistem, dapat berupa manusia atau sistem terotomatisasi lain. *Actor* adalah sesuatu atau seseorang yang berinteraksi dengan sistem, yaitu siapa atau apa yang menggunakan sistem. Yang dimaksud berinteraksi adalah *actor* mengirim atau menerima pesan ke atau dari sistem, atau mempertukarkan informasi dengan sistem.

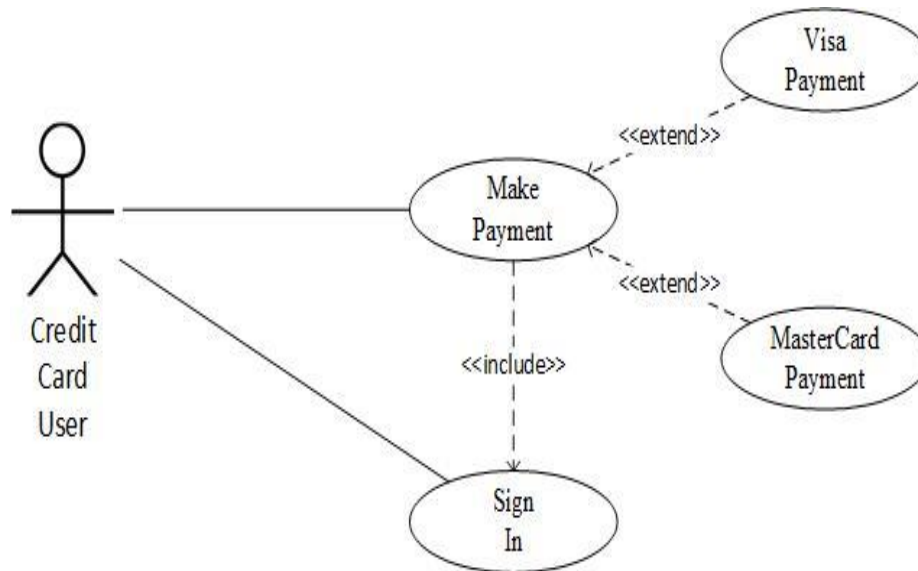
2. *Use Case*

Use case adalah cara spesifik penggunaan sistem oleh *actor*. *Use case* menspesifikasikan perilaku sistem atau bagian sistem dan merupakan deskripsi sekumpulan sekuen aksi termasuk varian-varian yang dilakukan oleh sistem untuk memproduksi hasil atau nilai ke *actor*. *Use case* berguna untuk menangkap kebutuhan sistem, berkomunikasi dengan pemakai akhir dan pakar domain masalah, serta pengujian sistem.

Relasi dalam *Use Case* diantaranya adalah [12]:

1. *Association*, menghubungkan link antar *element*.
2. *Generalization*, biasa disebut *inheritence*, sebuah *element* dapat merupakan spesialisasi dari *element* lainnya.
3. *Depedency*, sebuah *element* bergantung dalam beberapa cara ke *element* lainnya.

4. *Aggregation*, bentuk *association* dimana sebuah *element* berisi *element* lainnya.



Gambar 2.1. Contoh Use Case[12]

2.5.2. Activity Diagram

Diagram aktivitas atau *Activity Diagram* menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis. Dapat juga untuk menggambarkan aliran kejadian dalam *Use Case*. Komponen pada diagram ini antara lain [12]:

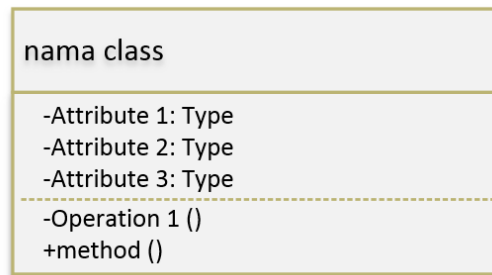
1. *Activity* atau *State* : menunjukkan aktivitas yang dilakukan
2. *Initial activity* atau *initial state* : Menunjukkan awal aktivitas dimulai
3. *Final activity* atau *final state* : Menunjukkan akhir dari aktivitas
4. *Decission* : Digunakan untuk menggambarkan *test* kondisi untuk memastikan bahwa *control flow* atau *object flow* mengalir ke satu jalur.
5. *Merge* : Berfungsi menggabungkan *flow* yang dipecah oleh *decission*
6. *Synchronization* : Dibagi menjadi dua yaitu *join* dan *fork*. *Fork* digunakan untuk memecah *behavior* menjadi *activity* atau *action* yang paralel. Sedangkan *join* untuk menggabungkan kembali aktivitas atau *action* yang paralel
7. *Swimlanes* : Memecah aktivitas diagram menjadi baris dan kolom untuk membagi tanggung jawab obyek-obyek yang melakukan aktivitas

8. *Transition* : Menunjukkan aktivitas selanjutnya setelah aktivitas sebelumnya.

2.5.3. Class Diagram

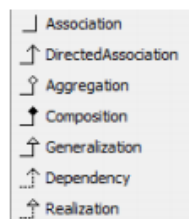
Class diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/perangkat lunak yang sedang kita kembangkan dimana diagram ini memberi gambaran tentang sistem/perangkat lunak dan relasi-relasi yang ada di dalamnya.

Atribut merupakan properti dari sebuah kelas yang melambangkan nilai-nilai yang mungkin ada pada kelas tersebut. *Operation* atau *method* merupakan *behavior* (tingkah laku) atau fungsi yang dapat dilakukan oleh kelas tersebut.



Gambar 2.2. Class Diagram[12]

Kolom paling atas merupakan representasi dari nama kelas, kolom tengah berisi atribut-atribut yang dimiliki kelas tersebut, dan kolom terakhir berisi operasi atau *method*. Ada pula relasi antar *class*, UML mendefinisikan beberapa cara untuk merepresentasikan relasi (hubungan) antara kelas, diantaranya adalah sebagai berikut :

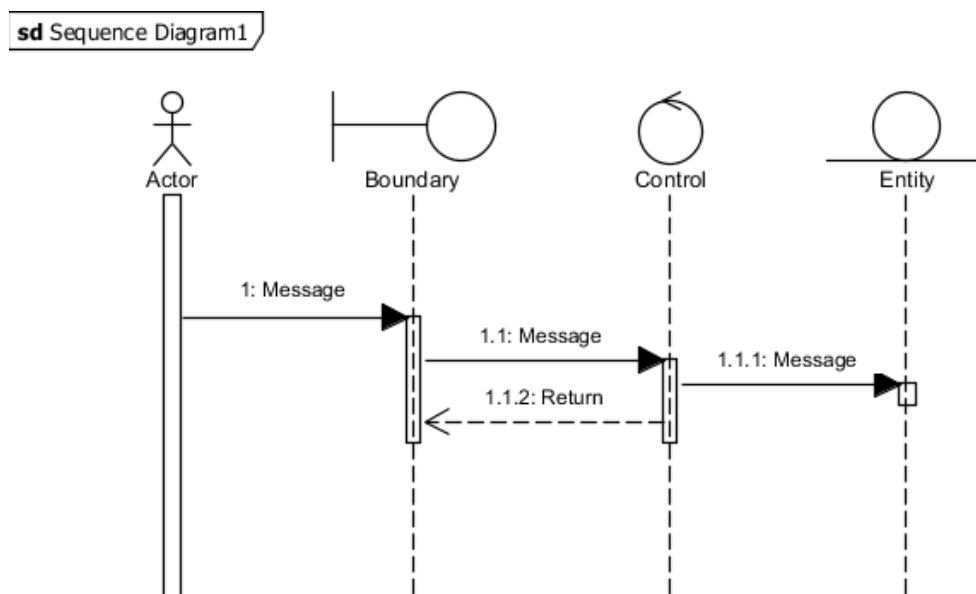


Gambar 2.3. Jenis Jenis Relasi[12]

2.5.4. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram jenis ini memberikan kejelasan sejumlah objek dan pesan-pesan yang diletakkan diantaranya didalam sebuah *use case*. Komponen utamanya adalah objek yang digambarkan dengan kotak segi empat, *message* yang digambarkan dengan garis penuh, dan waktu yang ditunjukkan dengan *progress vertical*.

Sequence diagram digunakan untuk memberikan gambaran detail dari setiap *use case* diagram yang dibuat sebelumnya. Setiap objek yang terlibat dalam sebuah *use case* digambarkan dengan garis putus-putus vertikal, kemudian *message* yang dikirim oleh objek digambarkan dengan garis horizontal secara kronologis dari atas ke bawah.



Gambar 2.4. Sequence Diagram[12]

2.6. Software yang Digunakan

Adapun beberapa perangkat lunak yang digunakan dalam pembuatan *game* edukasi fisika ini diantaranya Unity, Adobe Photoshop, Blender, dan Fuse.

1. Unity

Unity merupakan suatu aplikasi yang digunakan untuk mengembangkan *game multi platform* yang didesain untuk mudah digunakan. *Unity* itu bagus dan penuh

perpaduan dengan aplikasi yang profesional. *Editor* pada *Unity* dibuat dengan *user interface* yang sederhana. *Editor* ini dibuat setelah ribuan jam yang mana telah dihabiskan untuk membuatnya menjadi nomor satu dalam urutan teratas untuk *editor game*. *Grafis* pada *unity* dibuat dengan *grafis* tingkat tinggi untuk *OpenGL* dan *directX* [18].

2. Adobe Photoshop

Aplikasi *adobe photoshop* adalah program komputer yang merupakan perangkat lunak atau software yang di gunakan untuk pengolahan gambar/foto, dapat di gunakan untuk memanipulasi foto sehingga hasil foto lebih menarik. *Aplikasi Adobe Photoshop* pada dasarnya merupakan aplikasi pengolah gambar, namun seringkali digunakan untuk mengubah tampilan suatu objek, misalnya teks atau tulisan. *Adobe Photoshop* bukan pengolah kata, tapi dapat juga membuat beragam efek menarik untuk mempercantik tampilan gambar dan teks [19].

3. Blender

Blender adalah sebuah aplikasi gratis untuk pembuatan model 3D. *Blender* mendukung hampir keseluruhan Pemodelan 3D, Rigging, Animasi, Simulasi, Rendering, penggabungan, menentukan gerakan objek, editing video serta pembuatan game [20].

4. Fuse

Fuse adalah aplikasi gratis yang dapat digunakan membuat karakter untuk *game* 3D yang bisa digunakan dengan mudah, karena ketersediaan menunya lengkap. Setelah karakter selesai dibuat, tinggal menggunakan layanan *Mixamo* untuk membuat Rigging serta animasi dari karakter yang telah dibuat [20].

2.7. Pengujian Black Box

Pengujian *Blackbox* digunakan untuk merepresentasikan sistem yang cara kerja di dalamnya tidak tersedia untuk diinspeksi. Di dalam *blackbox*, item-item yang diuji dianggap “gelap” karena logikanya tidak diketahui, yang diketahui hanya apa yang masuk dan apa yang keluar dari *blackbox*.

Pada pengujian *blackbox* berdasarkan pada spesifikasi sistem. Rencana pengujian dapat dimulai sedini mungkin di proses pengembangan perangkat lunak.

Teknik pengujian konvensional yang termasuk pengujian *blackbox* adalah sebagai berikut :

1. *Graph-based testing*
2. *Equivalence partitioning*
3. *Comparison testing*
4. *Orthogonal array testing*

Pada pengujian *blackbox*, kita mencoba beragam masukan dan memeriksa keluaran yang dihasilkan. Kita dapat mempelajari apa yang dilakukan kotak, tapi tidak mengetahui sama sekali mengenai cara konversi dilakukan.

Teknik pengujian *blackbox* juga dapat digunakan untuk pengujian berbasis skenario, dimana isi dalam sistem mungkin tidak tersedia untuk diinspeksi tapi masukan dan keluaran yang didefinisikan dengan *use-case* dan informasi analisis yang lain.[12].

2.8. Pengujian Hipotesis

Pengujian Hipotesis adalah pengujian dengan asumsi atau dugaan mengenai sesuatu hal yang dibuat untuk menjelaskan hal yang sering dituntut untuk melakukan pengecekannya. Jika asumsi atau dugaan itu dikhususkan mengenai populasi, umumnya mengenai nilai-nilai parameter populasi, maka hipotesis itu disebut hipotesis statistik. Kecuali dinyatakan lain, disini dengan hipotesis dimaksudkan hipotesis statistik. Demikianlah misalnya, yang berikut dapat dianggap sebagai hipotesis:

1. Peluang lahirnya bayi berjenis laki-laki = 0,5.
2. 30% masyarakat termasuk golongan A.
3. Rata-rata pendapatan keluarga di suatu daerah Rp.35.000,00 tiap bulan.

Setiap hipotesis bisa benar atau tidak benar dan karenanya perlu diadakan penelitian sebelum hipotesis itu diterima atau ditolak. Langkah atau prosedur untuk menentukan apakah menerima atau menolak hipotesis dinamakan pengujian hipotesis[21].

Pengujian Hipotesis uji 2 sample t test adalah pengujian hipotesis untuk membandingkan 2 sampel data dengan asumsi kedua sampel memiliki *variance* yang sama. Langkah langkahnya adalah sebagai berikut :

1. Tentukan Hipotesis Nol (H_0) dan Hipotesis alternatif (H_1).
2. Tentukan taraf nyata atau *Significance level*.
3. Tentukan Nilai Kritis (nilai tabel) dan Statistik uji hipotesis-nya.
4. Hitung nilai statistik uji hipotesis, dengan rumus :

$$s_p^2 = \frac{(n_1 - 1)s_1^2 + (n_2 - 1)s_2^2}{n_1 + n_2 - 2}$$

$$t = \frac{(\bar{x}_1 - \bar{x}_2) - d_0}{s_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

$$df = n_1 + n_2 - 2$$

5. Pengambilan keputusan, tolak H_0 apabila t hitung $>$ t tabel (nilai kritis).