BAB II

LANDASAN TEORI

2.1 Pengertian Doa

Menurut bahasa do'a berasal dari kata da'a artinya memanggil. Sedangkan menurut istilah syara do'a berarti Memohon sesuatu yang bermanfaat dan memohon terbebas atau tercegah dari sesuatu yang hal yang buruk, doa juga bisa berarti sebagai pujian dan permintaan. Berikut ini arti doa dalam AlQuran

1. Ibadah, seperti firman Allah: Dan janganlah kamu menyembah apa-apa yang tidak memberi manfaat dan tidak memberi madharat kepadamu selain Allah, sebab jika kamu berbuat demikian make, kamu termasuk orang-orang yang zhalim. (Yunus: 106).
2. Perkataan atau Keluhan. Seperti pada firman Allah: Maka tetaplah demikian keluhan mereka, sehingga kami jadikan mereka sebagai tanaman yang telah dituai, yang tidak dapat hidup lagi. (al Anbiya: 15).
3. Panggilan atau seruan. Allah berfirman: Maka kamu tidak akan sanggup menjadikan orang-orang yang mati itu dapat mendengar, dan menjadikan orang-orang yang tuli dapat mendengar seruan, apabila mereka itu berpaling ke belakang. (ar- Rum: 52)
4. Meminta pertolongan. Allah berfirman: Dan jika kamu (tetap) dalam keraguan tentang at Qur'an yang Kami wahyukan kepada hamba Kami (Muhammad) buatlah satu surat yang semisal at Qur'an itu dan ajaklah

penolong-penolongmu selain Allah, jika kamu orang-orang yang benar. (al Baqarah: 23).

5. Permohonan. Seperti firman Allah: Dan orang-orang yang berada dalam neraka berkata kepada penjagapenjaga jahannam: "Mohonkanlah kepada Tuhanmu supaya Dia meringankan azab dari kami barang sehari." (al Mukmin: 49).

2.2 Pengertian Game

Game adalah sesuatu aktifitas yang dimainkan dengan aturan tertentu sehingga ada pemenang dan yang kalah, yang bertujuan untuk menghibur dan memberi rasa senang kepada para pemainya menurut Brenda Brathwaite And Ian Schreiber mendefinisikan Game merupakan sebuah Permainan yang secara garis besar betujuan menghibur dan memberikan kesenangan kepada para penggunanya, Ada banyak definisi dari kata "*Game*" namun tidak ada yang telah diterima secara universal. Oleh karena itu ada banyak sekali definisi dari *game* tergantung dari tujuan dari *game* itu sendiri [4].

Video game merupakan sebuah permainan yang berbasis elektronik visual yang memanfaatkan teknologi audio visual, video game ini adalah permainan yang banyak digunakan dikarenakan proses visual dari permainan yang sangat interaktif.

2.1.1 Jenis *Game*

Ada banyak sekali jenis *game* yang beredar saat ini namun dari semua jenis *game* yang ada bisa di kategorikan sebagai berikut

A. *Shooting*

B. *Role Playing game*

C. *Casual Game*

D. *Education Game*

E. *Stategi*

F. *Simulation Game*

G. *Adventure Game*

2.3 Pengertian Edukasi

Edukasi dalam bahasa indonesia berarti pendidikan merupakan suatu kegiatan yang dilakukan untuk mendapatkan suatu ilmu tertentu, dalam pendidikan konvensional biasanya pembelajaran suatu ilmu tertentu interaksi dari pembelajaranya harus dilakukan secara langsung baik melalui media pengajar maupun media buku dan lainya,

2.4 Pengertian Game edukasi

Game edukasi adalah permainan yang telah dirancang untuk media pembelajaran tentang topik tertentu, *game* ini dibangun dengan memadukan antara permainan yang menyenagkan dan menghibur dan pembelajaran, sehingga *game* ini mempunyai dua manfaat sekaligus yaitu menghibur dan memberikan pembelajaran bagi para penggunaya. Karena game edukasi bersifat kognitif dimana kognitif bersifat pasif statis yang merupakan potensi atau daya untuk memahami sesuatu, dan itelegensi bersifat aktif yang merupakan aktualisasi dari aktifitas atau perilaku, kognitif merupakan suatu proses berfikir untuk

menghubungkan, menilai, memperimbangkan suatu peristiwa atau kejadian. Ada banyak sekali interaksi dalam *game edukasi* diantaranya dengan menggunakan control dari *joystick, keyboard, touchscreen*, suara dan *augmented reality*.

2.3.1 Pengertian Augmented Reality

Augmented reality (AR) sebuah tampilan real-time langsung atau tidak langsung dari sebuah fisik dari sebuah objek nyata ditambah dengan menambahkan objek pada dunia maya sehinggan menghasilkan informasi tambahan pada objek yang ada. *Augmented reality* ini menggabungkan benda-benda nyata dan virtual objek yang ada, virtual objek ini hanya bersifat menambahkan bukan menggantikan objek nyata, sedangkan tujuan dari *augmented reality* ini adalah menyederhanakan objek nyata dengan membawa objek maya sehingga informasi tidak hanya untuk pengguna secara langsung (*user interface*), tetapi juga untuk setiap pengguna yang tidak langsung berhubungan dengan *user interface* dari objek nyata, seperti *live-streaming video* [2]. Perangkat utama untuk *augmented reality* adalah *display*, perangkat *input*, *tracking*, dan komputer.

AR Interface Salah satu aspek paling penting dari *augmented reality* adalah untuk menciptakan yang sesuai teknik untuk interaksi intuitif antara pengguna dan konten virtual AR aplikasi. Ada empat cara utama interaksi dalam aplikasi AR: tangible AR interface, kolaboratif *AR interface*, hibrida *AR interface*, dan muncul multimodal interface.

a) *Tangible Interface*

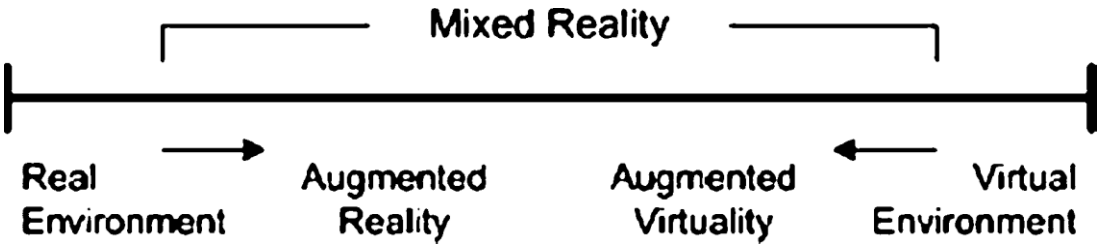
Berwujud interface yang mendukung interaksi secara langsung dengan dunia nyata dengan memanfaatkan objek nyata, salah satu contoh dari tangible interface ini adalah pada aplikasai virtual fittng room dan *game augmented reality pringleys* dimana tabung dari kemasan pringleys menjadi marker dan control dari *game* tersebut.

b) Kolaborasi *AR Interface*

Kolaborasi AR interface menampilkan beberapa tampilan untuk mendukung suatu kegiatan dalam berbagi *interface* 3D untuk meningkatkan interaksi kolaboatif dengan banyak perangkat dan banyak user. Interface ini dapat diintegrasikan dengan aplikasi medis untuk melakukan diagnosa operasi.

c) *Interface Hybrid*

menggabungkan berbagai macam device berbeda, tetapi saling melengkapi interface serta memungkinan untuk berinteraksi melalui berbagai perangkat interaksi, contoh pada aplikasi *Augmented Reality* yang menggunakan sarung tangan dan kacamata seperti pada gambar 2.1 Cara Kerja Augmented Reality.



Gambar 2. 1 Cara Kerja Augmented Reality

2.3.2 Marker

Marker adalah real enviroment berbentuk objek nyata yang akan menghasilkan virtual reality, marker ini digunakan sebagai tempat augmented reality muncul, berikut ini beberapa jenis marker yang digunakan pada aplikasi augmented reality:

1. *Quick Response* (QR) Kode dua dimensi kode yang terdiri dari banyak kotak diatur dalam pola persegi, Biasanya QR ini berwarna hitam dan putih, kode QR diciptakan di Jepang pada awal 1990-an dan digunakan untuk melacak berbagai bagian dalam manufaktur kendaraan. Dan saat ini QR digunakan sebagai link cepat ke website, dial cepat untuk nomor telepon, atau bahkan dengan cepat mengirim pesan SMS seperti pada gambar 2.2 QR (quick response) Code.



Gambar 2. 2 QR (quick response) Code

2. Fiducial Marker adalah bentuk paling sering digunakan oleh teknologi AR karena marker ini digunakan untuk melacak benda-benda di virtual reality tersebut. kotak hitam dan putih digunakan sebagai titik referensi atau untuk memberikan skala dan orientasi ke aplikasi. Bila penanda tersebut deteksi dan dikenali maka augmented reality akan keluar dari marker ini seperti pada gambar 2.3 Fiducial Marker.



Gambar 2.3 Fiducial Marker

3. Markerless Marker berfungi sama seperti fiducial marker yang namun bentuk markerless marker tidak harus kotak hitam putih, markerless ini bisa

berbentuk gambar yang mempunyai banyak warna seperti pada gambar 2.4 Markerless marker.



Gambar 2.4 Markerless marker

2.3.3 Sejarah Augmented reality

Penampilan pertama dari *Augmented reality* 1950-an ketika Morton Heilig, sinematografer, berfikir bioskop adalah kegiatan yang akan memiliki kemampuan untuk menarik penonton. Pada tahun 1962, Heilig membangun prototipe dari visinya, yang digambarkan pada tahun 1955 dalam "Future Cinema," bernama Sensorama. Selanjutnya, Ivan Sutherland adalah yang pertama untuk menciptakansebuah realitas ditambah sistem menggunakan optik tembus kepala-mount layar. Pada tahun 1975, Myron Krueger menciptakan Videoplace, ruang yang memungkinkan pengguna untuk berinteraksi dengan objek virtual untuk pertama kalinya. Kemudian, Tom Caudell dan David Mizell dari Boeing menciptakan *Augmented reality* sementara pekerja membantu merakit kabel-kabel untuk pesawat terbang. Mereka juga mulai membahas keuntungan dari *Augmented reality* vs *Virtual Reality* (VR), *augmented reality* (AR) adalah penggabungan antara obejek virtual dengan objek nyata. Sebagai contoh, adalah

saat stasiun televisi, menyiarkan pertandingan sepak bola, terdapat objek virtual, tentang skor pertandingan yang sedang berlangsung. Menurut Ronald Azuma pada tahun 1997, *augmented reality* adalah menggabungakan dunia nyata dan virtual, bersifat interaktif secara real time, dan merupakan animasai 3D[2]. Sejarah tentang *augmented reality* dimulai dari tahun 1957-1962, ketika seorang penemu yang bernama Morton Heilig, seorang sinematografer, menciptakan dan memapatenkan sebuah simulator yang disebut Sensorama dengan visual, getaran dan bau. Pada tahun 1966, Ivan Sutherland menemukan head-mounted display yang dia claim adalah, jendela ke dunia virtual. Tahun 1975 seorang ilmuwan bernama Myron Krueger menemukan Videoplace yang memungkinkan pengguna, dapat berinteraksi dengan objek virtual untuk pertama kalinya. Tahun 1989, Jaron Lanier, memeperkenalkan Virtual Reality dan menciptakan bisnis komersial pertama kali di dunia maya, Tahun 1992 mengembangkan *Augmented reality* untuk melakukan perbaikan pada pesawat boeing, dan pada tahun yang sama, LB Rosenberg mengembangkan salah satu fungsi sistem AR, yang disebut *Virtual Fixtures*, yang digunakan di Angkatan Udara AS Armstrong Labs, dan menunjukan manfaatnya pada manusia, dan pada tahun 1992 juga, Steven Feiner, Blair Maclntyre dan dorée Seligmann, memperkenalkan untuk pertama kalinya Major Paper untuk perkembangan *Prototype AR*. Pada tahun 1999, Hirokazu Kato, mengembangkan *ArToolkit* di HITLab dan didemonstrasikan di SIGGRAPH, pada tahun 2000 Bruce.H.Thomas mengembangkan *ARQuake*, sebuah Mobile *Game* AR yang ditunjukan di *International Symposium* on Wearable Computers. Pada tahun 2008, Wikitude AR *Travel Guide*,

memperkenalkan *Android G1 Telephone* yang berteknologi AR, tahun 2009, Saqoosha memperkenalkan *FLARToolkit* yang merupakan perkembangan dari *ArToolkit*. *FLARToolkit* memungkinkan kita memasang teknologi AR di sebuah website, karena output yang dihasilkan *FLARToolkit* berbentuk Flash. Ditahun yang sama, Wikitude Drive meluncurkan sistem navigasi berteknologi AR di *Platform Android*. Tahun 2010, Acrossair menggunakan teknologi AR pada *IPhone 3GS*[3].

2.3.4 Penerapan *Augmented Reality* pada berbagai bidang

A. Medis

Beberapa proyek yang mengeksplorasi daerah ini aplikasi. Di UNC Chapel Hill, kelompok riset telah melakukan percobaan berjalan pemindaian Rahim wanita hamil dengan sensor *ultrasound*, menghasilkan representasi 3-D dari janindalam rahim dan menampilkan bahwa dalam tembus HMD. Tujuannya adalah untuk memberkati dokter dengan kemampuan untuk melihat janin, bergerak menendang berbaring di dalam Rahim [5] seperti pada gambar 2.5 *Virtual fetus inside womb of pregnant patient*. (Courtesy UNC ChapelnHill Dept. of Computer Science.).



Gambar 2. 5 Virtual fetus inside womb of pregnant patient. (Courtesy UNC ChapelnHill Dept. of Computer Science.)

B. *Manufactur*

Beberapa proyek penelitian telah menunjukkan prototipe. Steve Kelompok Feiner di Columbia membangun sebuah aplikasi pemeliharaan laser printer, menunjukkan objek maya eksternal. Sebuah kelompok di Boeing sedang mengembangkan teknologi AR untuk memandu teknisi di membangun memanfaatkan kabel yang merupakan bagian dari sistem listrik pesawat itu. Menyimpan petunjuk ini dalam bentuk elektronik akan menghemat ruang dan mengurangi biaya. saat ini,teknisi menggunakan papan besar layout fisik untuk membangun harness tersebut, dan Boeingmembutuhkan beberapa gudang untuk menyimpan semua papan ini. Ruang tersebut dapat dikosongkanuntuk penggunaan lain jika aplikasi ini terbukti sukses Boeing menggunakan Teknologi Program Reinvestasi (TRP)[5] seperti pada gambar 2.6 *Augmented reality* pada proses *manufactur Game*.



Gambar 2.6 *Augmented reality* pada proses *manufacturGame*

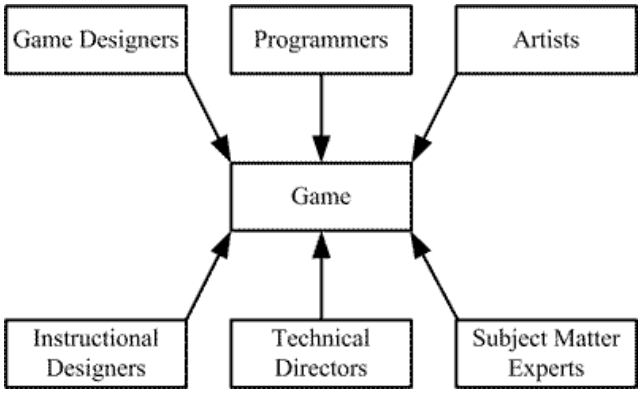
2.3.5 Game Edukasi Dengan Teknik Interaksi *Augmented Reality*

Game edukasi yang ada saat ini umunya harus menggunakan konsol atau piranti dalam memainkanya, *game* edukasi dengan teknik interaksi

augmented reality adalah sebuah *game* yang memadukan antara *game* edukasi dan teknologi *augmented reality* dimana interaksi dari *game* ini menggunakan objek nyata dari *augmented reality* dan permainan edukasinya terdapat dalam realita tambahan dari objek nyata ini. Sehingga dalam penggunaanya *game* ini dapat lebih interaktif dan dapat menghibur.

2.5 Proses Pengembangan Game

game biasanya dikembangkan oleh sebuah tim terdiri dari programmer, seniman, desainer game, direksi teknis, ahli subjek (UKM), dan desainer instruksional seperti pada gambar 2.7 Struktur Tim Pengembangan Game edukasi.



Gambar 2.7 Struktur Tim Pengembangan Game edukasi

1. *Subject Matter Experts (SME)*

   Bertanggung jawab untuk menyediakan konten pembelajaran untuk permainan. Ini adalah tanggung jawab utama mereka untuk mengidentifikasi tujuan pembelajaran untuk metrik pada pasokan permainan, untuk penilaian, dan untuk membantu mengidentifikasi strategi instruksional untuk permainan.

2. *Instructional Designers*

Bertanggung jawab untuk mengambil isi diidentifikasi oleh SME dan datang dengan strategi untuk menempatkannya dalam permainan dengan cara yang memaksimalkan efektivitas instruksional tersebut. Dalam skenario SME adalah seorang ahli didalam bidangnya tapi tidak ahli dalam teori pendidikan.

3. *Game Designers*

Terutama bertanggung jawab untuk mendefinisikan teka-teki, aturan, dan manfaat yang akan menghibur dan menantang untuk pemain dan kemudian memberikan tim produksi dengan peta jalan yang cukup spesifik bagi mereka untuk menciptakan permainan yang hidup sampai permainan dibayangkan oleh desainer. Sampai batas tertentu, kita dapat mengatakan bahwa desainer game melayani peran yang sama dengan penulis atau sutradara dalam sebuah film. Visi mereka mendorong sisa proses produksi. Seluruh aset yang diciptakan oleh tim seni dan kegiatan diprogram oleh para insinyur perangkat lunak yang didorong oleh keputusan yang dibuat oleh para desainer game pada tahap pra-produksi.

4. *Programmers*

Menulis *Source Code* untuk membuat permainan. Programer biasanya memiliki latar belakang di bidang Ilmu Komputer dan bekerja di bidang yang sangat khusus. Secara umum programmer bertanggung jawab atas kecerdasan buatan (AI), rendering, jaringan, I/O, dan *Source Code* bermain game. Tim yang yang memiliki seorang programmer tunggal mengurus semua bidang ini cenderung menghasilkan game rendah.

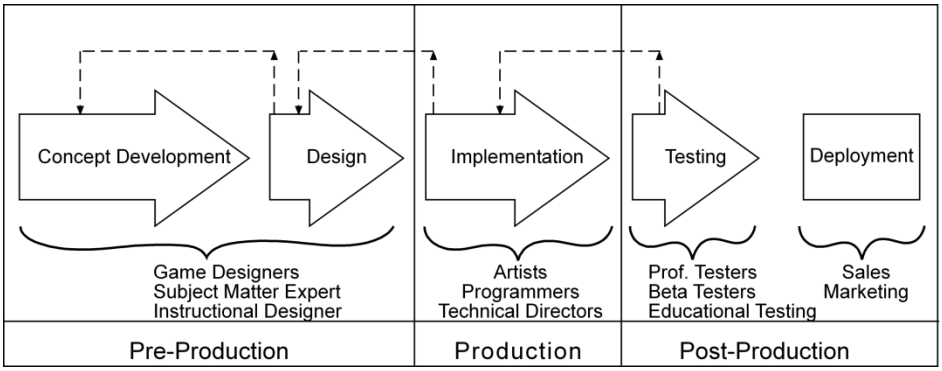
5. *Artists*

Memproduksi *Assets Game* untuk permainan. Mereka membuat model, tekstur, elemen 2D, dll. Proses produksi seni sangat padat karya dan juga khusus. Ini tidak jarang untuk menemukan peran pemodel, animator, envelopers, seniman tekstur, spesialis menangkap gerakan, dan seniman antarmuka pengguna diisi oleh orang yang berbeda.

6. *Technical Directors*

Dan secara teknis, seniman memiliki latar belakang pemrograman dan juga mengetahui arah produksi seni. Karena mereka memiliki pengetahuan yang baik tentang para programmer dan seniman, mereka memiliki kemampuan untuk menulis *custom tools* untuk para seniman yang mempercepat proses produksi. Mereka juga melayani sebagai penghubung teknis antara seniman dan programer.

Dari pespektif global, desainer permainan, SME, dan desainer instruksional melakukan sebagian besar produksi pra-produksi bagi tim. Ini adalah visi mereka dari bermain game dan aspek instruksional dari permainan yang mendorong proses produksi. Permainan ini benar-benar diterapkan oleh tim produksi, yang terdiri dari *Artists*, *Programmers*, dan *Technical Directors*. Efektivitas permainan diuji pada tahap pasca-produksi oleh kedua penguji permainan dan spesialis pendidikan (SME dan ID). Fokus *Instructional Designer* game pada penilaian aspek adalah untuk membuat game jadi menyenangkan. SME dan *Instructional Designer* instruksional pada bagian pendidikan dari permainan seperti pada gambar 2.8 Proses Pengembangan Game.



Gambar 2.8 Proses Pengembangan Game

2.4.1 UML

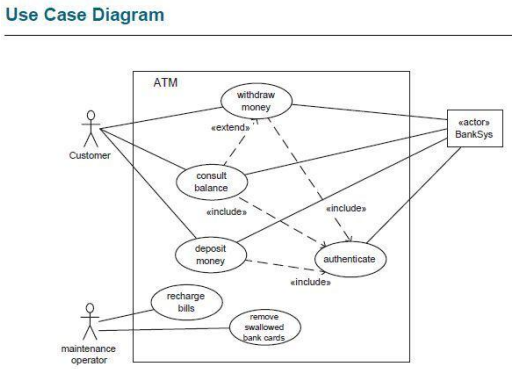
UML adalah bahasa untuk mengspesifikasi, memvisualisasi, membangun dan mendokumentasi *artefacts* (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak), seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya. UML merupakan bahasa standar untuk penulisan *blueprint software* yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasian alat-alat dari system perangkat lunak.

UML disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan untuk mendesain secara cepat[6].

Bahasa pemodelan merupakan bagian terpenting dari metode. Ini merupakan bagian kunci tertentu untuk komunikasi. Jika anda ingin berdiskusi tentang desain dengan seseorang, maka Anda hanya membutuhkan bahasa pemodelan bukan proses yang digunakan untuk mendapatkan desain.

2.4.1.1 Use Case Diagram

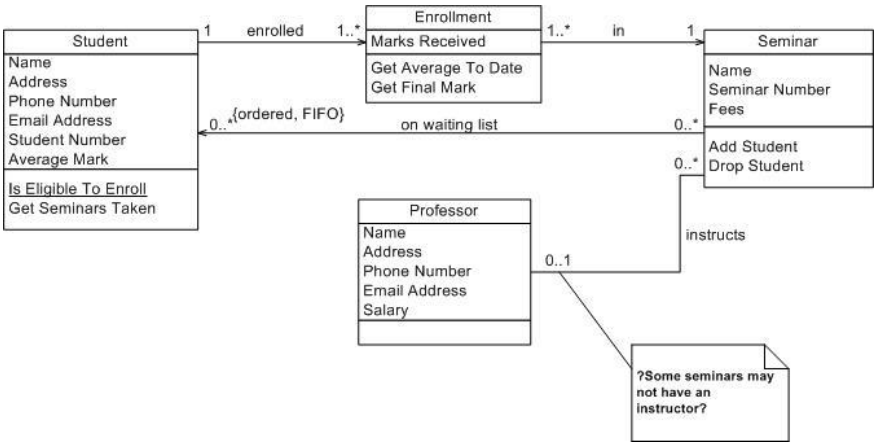
Menggambarkan sejumlah eksternal aktor dan hubungannya ke use case yang diberikan oleh sistem seperti pada gambar 2.9 Use Case Diagram.



Gambar 2.9 Use Case Diagram

2.4.1.2 Class Diagram

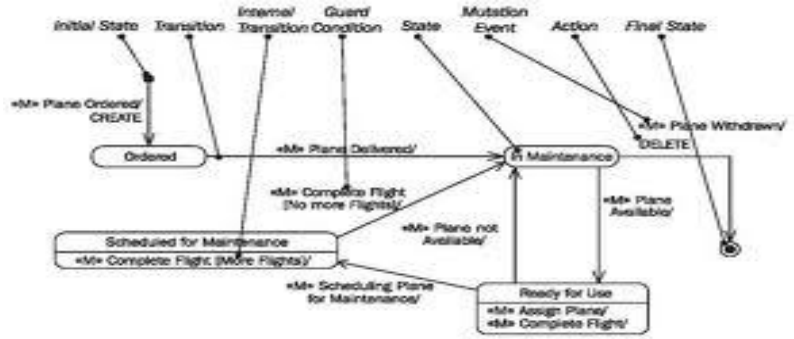
Menggambarkan struktur dan deskripsi class, *package* (paket) dan objek beserta hubungan satu sama lain seperti *containment* (penahanan), pewarisan, asosiasi dan lain-lain seperti pada gambar 2.10 Class Diagram.



Gambar 2.10 Class Diagram

2.4.1.3 Statechart Diagram

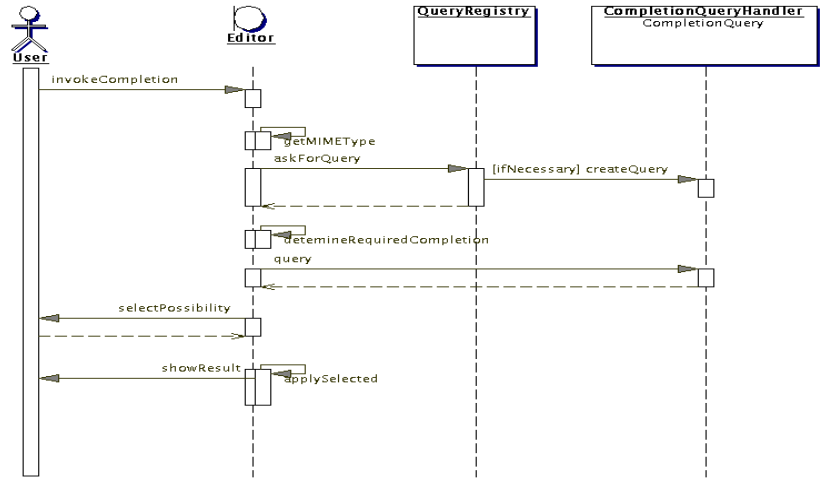
Menggambarkan semua state yang dimiliki oleh suatu objek dari suatu class dan keadaan yang menyebabkan state berubah seperti pada gambar 2.11 Statechart Diagram.



Gambar 2.11 Statechart Diagram

2.4.1.4 Sequance Diagram

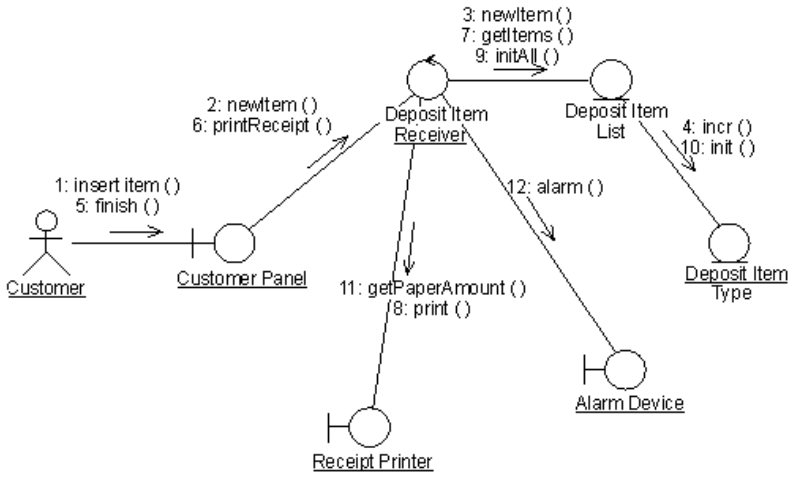
Menggambarkan kolaborasi dinamis antara sejumlah objek dan untuk menunjukan rangkaian pesan yang dikirim antara objek juga interaksi, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem seperti pada gambar 2.12 Sequance Diagram



Gambar 2.12 Sequance Diagram

2.4.1.5 Collaboration Diagram

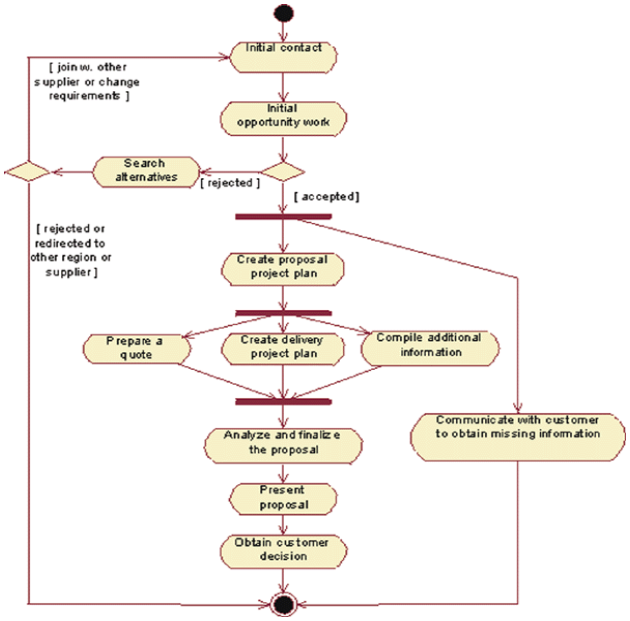
Menggambarkan kolaborasi dinamis seperti sequence diagram. Dalam menunjukan pertukaran pesan, collaboration diagram menggambarkan objek dan hubungannya seperti pada gambar 2.13 Collaboration Diagram.



Gambar 2.13 Collaboration Diagram

2.4.1.6 Activity Diagram

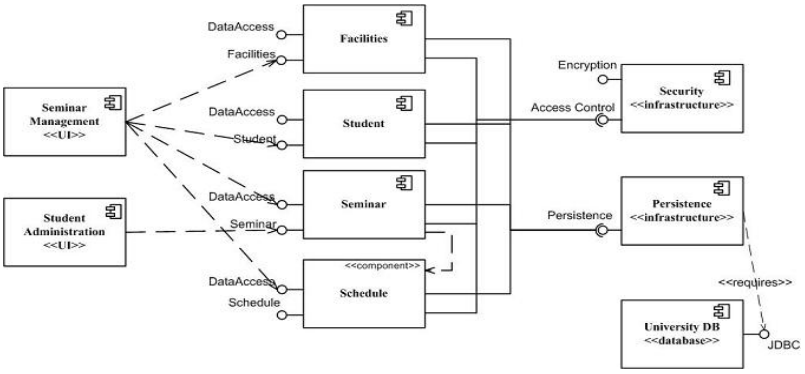
Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktivitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktivitas lainya seperti Use Case atau interaksi seperti pada gambar 2.14 Activity Diagram.



Gambar 2.14 Activity Diagram

2.4.1.7 Component Diagram

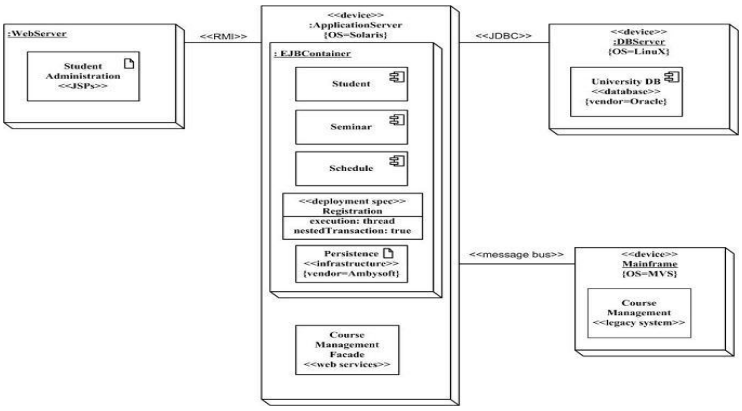
Mengambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan diantaranya. Komponen dapat juga berupa interface, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lainnya seperti pada gambar 2.15 Component Diagram.



Gambar 2.15 Component Diagram

2.4.1.8 Deployment Diagram

Menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem, menunjukan hubungan computer dengan perangkat satu sama lain dan jenis hubungannya seperti pada gambar 2.16 Deployment Diagram.



Gambar 2.16 Deployment Diagram

2.4.1.9 Tujuan Penggunaan UML

1. Memodelkan suatu sistem (bukan hanya perangkat lunak) yang menggunakan konsep berorientasi objek.
2. Menciptakan suatu bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin.

2.6 Proses Pengujian White Box

Merupakan metode perancangan test case yang menggunakan struktur kontrol dari perancangan prosedural maupun objek untuk mendapatkan test case. Seperti Diagram alir, *Cyclomatic Complexity, Graph Matrix* Dengan menggunakan metode white box, analis sistem akan dapat memperoleh test case :

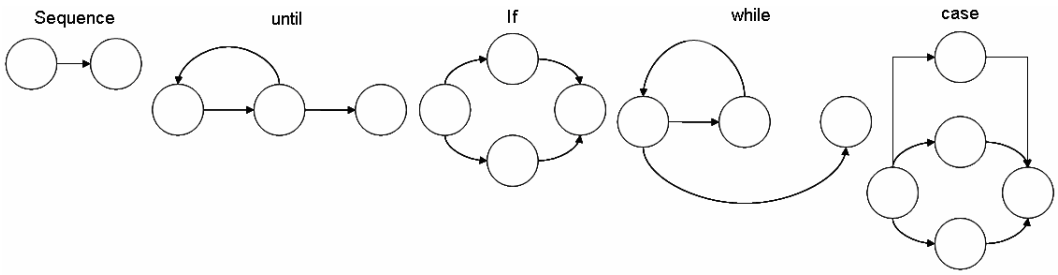
A. Menjamin seluruh independent path di dalam modul yang dikerjakan sekurang-kurangnya sekali
B. Mengerjakan seluruh keputusan logikal
C. Mengerjakan seluruh loop yang sesuai dengan batasannya
D. Mengerjakan seluruh struktur data internal yang menjamin validitas

1. Notasi Diagram alir

Lingkaran *(node)*, menggambarkan satu/lebih perintah prosedural. Urutan proses dan keputusan dapat dipetakan dalam satu node. Tanda panah *(edge)*, menggambarkan aliran kontrol. Setiap node harus mempunyai tujuan node. *Region* adalah daerah yang dibatasi oleh edge dan node. Termasuk daerah diluar grafik alir. Contoh menterjemahkan *pseudo code* ke grafik alir seperti terlihat pada gambar 2.17.

```
1 : do while record masih ada
baca record
2 : if record ke 1 = 0
3 : then proses record
simpan di buffer
naikan counter
4 : else if record ke 2 = 0
```

```
5 : then riser counter
6 : proses record
simpan pada file
7a: endif
endif
7b: enddo
8 : end
```

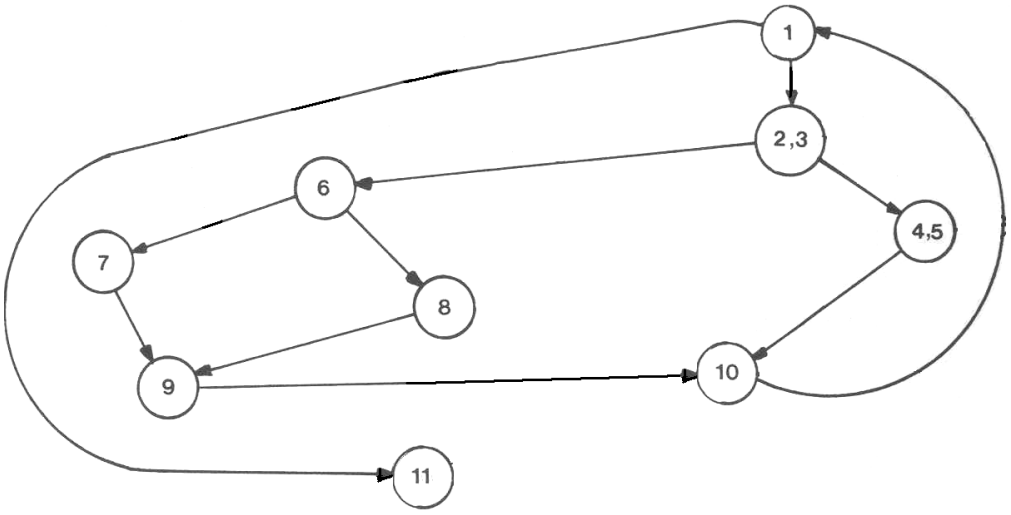


Gambar 2.17 *Flow Graph*

2. Cyclomatic Complexity

Cyclomatic complexity adalah metrik software yang menyediakan ukuran kuantitatif dari kekompleksan logikal program. Apabila digunakan dalam konteks metode uji coba basis *path*, nilai yang dihitung untuk *cyclomatic complexity* menentukan jumlah jalur independen dalam basis set suatu program dan memberi batas atas untuk jumlah uji coba yang harus dikerjakan untuk menjamin bahwa seluruh perintah sekurang-kurangnya telah dikerjakan sekali.

Jalur independent adalah jalur yang melintasi atau melalui program dimana sekurang-kurangnya terdapat proses perintah yang baru atau kondisi yang baru seperti terlihat pada gambar 2.18.



Gambar 2.18 *Path*

Dari gambar:

Path 1 = 1 - 11

Path 2 = 1 - 2 - 3 - 4 - 5 - 10 - 1 - 11

Path 3 = 1 - 2 - 3 - 6 - 8 - 9 ...: 10 - 1 - 11

Path 4 = 1 - 2 - 3 - 6 - 7 - 9 - 10 - 1 - 11

Path 1,2,3,4 yang telah didefinisikan diatas merupakan basis set untuk diagram alir.

Cyclomatic complexity digunakan untuk mencari jumlah path dalam satu flowgraph. Dapat dipergunakan rumusan sebagai berikut :

1. Jumlah region grafik alir sesuai dengan cyclomatic complexity.

2. Cyclomatix complexity V(G) untuk grafik alir dihitung dengan rumus:

V(G) = E - N + 2

Dimana:

E = jumlah edge pada grafik alir

N = jumlah node pada grafik alir

3. Cyclomatix complexity V(G) juga dapat dihitung dengan rumus:

V(G) = P + 1

Dimana P = jumlah predicate node pada grafik alir

Pada Gambar dapat dihitung cyclomatic complexity:

1. Flowgraph mempunyai 4 region

2. V(G) = 11 edge - 9 node + 2 = 4

3. V(G) = 3 predicate node + 1 = 4

Jadi cyclomatic complexity untuk flowgraph adalah 4

3. Graph Metrik

Graph metrik merupakan software yang dikembangkan untuk membantu uji coba basis path atau struktur data. Graph metrik adalah matrik empat persegi yang mempunyai ukuran yang sama dengan jumlah node pada flowgraph.Masing-masing baris dan kolom mempunyai hubungan dengan node yang telah ditentukan dan pemasukan data matrik berhubungan dengan hubungan (edge) antar node. Contoh sederhana pemakaian graph metrik seperti terlihat pada gambar 2.19.

	1	2	3	4	5	6	7	8	9	10	11	12	13	
1		1												0
2			1											0
3				1										0
4					1									0
5						1								0
6							1							0

7								1					1	1
8									1					0
9										1				0
10											1			0
11												1		0
12							1							0
13														0
SUM (E) + 1														2

Gambar 2.19 Graph Matriks

2.7 *Tools* yang digunakan

2.5.1 Openspace 3d

OpenSpace 3d adalah suatu perangkat lunak yang khusus digunakan untuk membuat aplikasi augmented reality, dimana aplikasi ini mempunyai beberapa keunggulan software ini telah mendukung metode tangible dan markerless selain itu perangkat lunak ini telah mendukung untuk aplikasi flash dan dekstop serta menggunakan bahasa pemograman html 5 dan keunggulan utama aplikasi ini tidak berbayar berikut gambar aplikasi openspace 3d seperti pada gambar 2.20.



Gambar 2. 20 Openspace 3d

2.5.2 Adobe Photoshop

Adobe Photoshop adalah software pengolah gambar yang sangat *powerfull* dengan segala fasilitasnya. Hasil gambar olah dengan Adobe Photoshop ini banyak dilihat di berbagai website, brosur, koran, majalah, dan media lainnya. seperti terlihat pada gambar 2.21 :



Gambar 2. 21 Adobe Photoshop CS3

Berikut ini adalah menu kerja yang tersedia pada photoshop:

1. Menu Bar

   Berisi perintah utama untuk membuka file, save, mengubah ukuran gambar, filter dan lain-lain.

2. Option

   Berisi pilihan dari tool yang Anda pilih. Misalnya dipilih kuas/brush, maka ukuran/diameter brush ada di sini.

3. Gambar

   Menampilkan gambar yang sedang di buat atau di edit.

4. Pallete Well

Cara cepat untuk mengakses palet brushes, tool resets dan Layer Comps. Juga dapat digunakan untuk meletakkan palet yang sering digunakan.

5. Toolbox

   Berisi tool untuk menyeleksi dan memodifikasi gambar.

6. Palette

   Berisi jendela-jendela kecil yang di dalamnya terdapat perintah dan pilihan untuk dokumen/gambar yang sedang dikerjakan.

2.5.3 Adobe Flash CS5

perangkat lunak komputer yang merupakan produk unggulan Adobe Systems. Adobe Flash digunakan untuk membuat gambar vektor maupun animasi gambar tersebut. Berkas yang dihasilkan dari perangkat lunak ini mempunyai *file extension* .swf dan dapat diputar di penjelajah web yang telah dipasangi Adobe Flash Player seperti pada gambar 2.22.



Gambar 2.22 Adobe Flash