BAB 2
TINJAUAN PUSTAKA

2.1 Tinjauan Sekolah

Peninjauan terhadap sekolah yang akan dilakukan meliputi sejarah singkat sekolah, visi dan misi sekolah serta struktur organisasi sekolah.

2.1.1 Sejarah Singkat Sekolah

Pada tahun 1949, KH. Wahid Hasyim (waktu itu menteri Agama RI), R.Jaya Rahmat dan KH. E.Z Muttaqien bersama – sama membentuk sebuah organisasi guru – guru Islam yang diberi nama Persatuan Guru – guru Islam Indonesia (PGII). Organisasi ini bertujuan untuk meningkatkan kualitas guru Islam dan didukung oleh 13 organisasi Islam lainnya.

Bersama dengan kembalinya Indonesia menjadi negara kesatuan pada tanggal 17 Agustus 1950, didirikan sebuah SMA atas usaha KH. E.Z Muttaqien dan Affandi Ridwan Sultan Abdul Gani, waktu itu SMA ini diberi nama SMA Muslimin, karena sasaran operasional dikelola oleh Lembaga Pendidikan Muslimin (LPM). Seahun kemudian LPM menyerahkan kembali kepada majelis pendidikan PGII, sehingga sekolah ini menjadi cikal bakal SMA PGII.

Bersamaan dengan perubahan – perubahan politik yang tak menentu kemudian PGII memperjelas diri menjadi sebuah yayasan yang diberi nama Yayasan Pendidikan PGII dan SMA PGII pun akhirnya semakin hari semakin berkembang bahkan dari tahun ke tahun.

Perluasan maupun peningkatan mutu terus menerus dilakkan oleh Yayasan Pendidikan PGII. Animo masyarakat semakin tinggi, tuntutan – tuntutan pun semakin besar sehingga pengembangan dari segi kualitas melahirkan ide untuk mendirikan SMA dan SMP PGII 2 Bandung yang berlokasi di Jalan Pahlawan Blk No. 17 Bandung.

SMP PGII 2 pun dari hari ke hari membenahi diri untuk dapat bersaing dengan SMP – SMP yang ada, khususnya SMP swasta. Stelah turunnya Undang – Undang Sistem Pendidikan Nasional nomor 2 tahun 1989 yang mengubah nama

Sekolah Menengah Pertama (SMP) menjadi Sekolah Lanjutan Tingkat Pertama (SLTP) sehingga SMP PGII 2 pun menyesuaikan diri menjadi SLTP PGII 2 Bandung.

Peningkatan kualitas dan kuantitas diupayakan terus menerus sehingga mampu mendongkrak prestasi siswa dari tahun ke tahun, bahkan pada tahun pelajaran 1996/1997 Nilai Ebtanas Murni (NEM) mencapai angka 49,26 dibidang lain (kreatifitas) alhadulillah beberapa kali menjuarai kegiatan perlombaan baik dalam seni, olahraga, dan sebagainya.

2.1.2 Visi, Misi, dan Strategi Sekolah

Adapun Sekolah Menengah Pertama PGII 2 Bandung memiliki visi yaitu menjadi sekolah islami yang unggul. Adapun misi yang ingin dicapai yaitu :

1. Menghasilkan lulusan yang beriman dan bertaqwa kepada Alloh Swt. Berakhlak mulia, cerdas, kreatif, mandiri, bertanggung jawab adaftif serta peduli terahadap lingkungan.
2. Menumbuh kembangkan budaya belajar, kompetitif dan Islami
3. Menyelenggarakan kegiatan persekolahan yang efektif, efisien, produktif, berkualitas dan Islami.
4. Melaksanakan pembelajaran yang berkualitas, profesional,dan manusiawi yang dijiwai oleh nilai-nilai keimanan dan ketaqwaan.
5. Menerapkan manajemen yang transparan, amanah, profesional, partisipatif, kompetitif dan Islami.
6. Memberikan pelayanan yang prima (excellence) terhadap stake holder pendidikan yang edukatif administratif dan irformatif.

Adapun strategi yang ingin dicapai, yaitu sebagai berikut :

1. Menumbuhkan sikap dan rasa tanggung jawab yang tinggi terhadap pelaksanaan syariat Islam dalam lingkungan sehari-hari.
2. Mewujudkan siswa didik yang memiliki prestasi akademik yang tinggi.
3. Menumbuhkan siswa yang memiliki jiwa kepedulian terhadap lingkungannya.
4. Mewujudkan suasana belajar yang islami, kondusif, kompetitif, dan Refresentatif.

2.1.3 Program Unggulan

Adapun program unggulan SMP PGII 2 yang ingin dikembangkan adalah sebagai berikut :

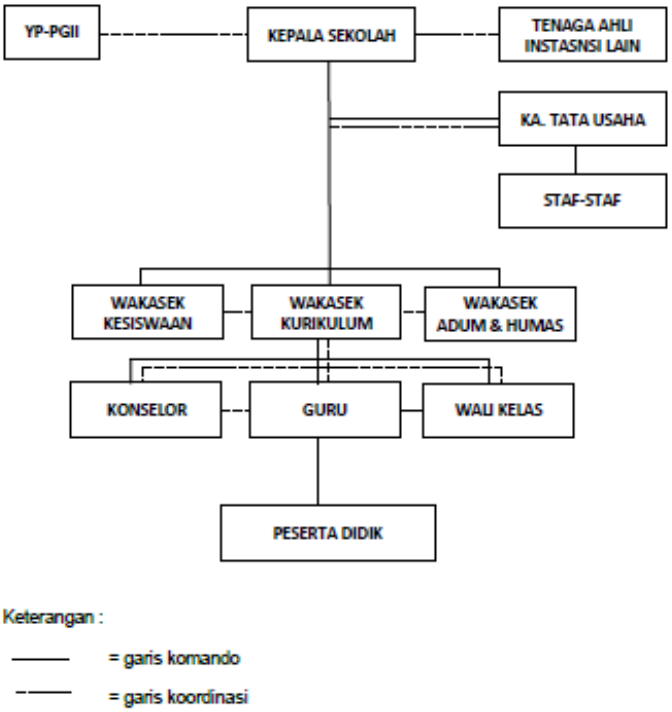
1. Nursyid (ilmuwan)
    a. Nilai ketuntasan permata pelajaran minimal 6.
    b. Membaca Al – Quran dengan baik dan hafal 30 surat dari juz amma.
    c. Menguasai 10 Hadist Arbain.
    d. Menguasai kecepatan membaca sesuai standar kecepatan baca
    e. Memilki sertifikat keahlian
2. Muhsin (berbudi pekerti luhur)
    a. Menjadi imam sholat keluarga
    b. Hormat dan santun terhadapa orang tua dan guru
    c. Memiliki kepekaan sosial yang tinggi
    d. Berbudaya tertib, indah, dan bersih
    e. Cerdas dalam menentukan pilihan
3. Mujadid (kreatif inovatif)
    a. Percaya diri
    b. Sungguh – sungguh untuk meraih mencapai cita – cita
    c. Bersemangat dalam belajar
    d. Pantang menyerah
    e. Berdisiplin tingi

2.1.4. Logo SMPPGII 2 Bandung



Gambar 2.1 Logo SMP PGII 2 Bandung

2.1.5. Struktur Organisasi SMP PGII 2 Bandung



Gambar 2.2 Struktur Organisasi SMP PGII 2 Bandung

2.1.6. Job Description

Memberikan informasi mengenai bagian – bagian dari struktur organisasi beserta tugas – tugas yang dikerjakan, sebagai berikut :

1. Kepala Sekolah

a. Menyusun perencanaan.
b. Mengorganisasikan kegiatan.
c. Mengarahkan kegiatan.
d. Melaksanakan pengawasan.
e. Melakukan evaluasi terhadap kegiatan.
f. Menentukan kebijaksanaan.
g. Mengadakan rapat koordinasi.
h. Mengambil keputusan.

2. Tata Usaha

a. Menyusun Program Tata Usaha Sekolah.
b. Mengurus administrasi ketenagaan.

c. Mengurus administrasi siswa seperti buku induk, buku mutasi dan buku klaper.
d. Membinadan mengembangkan karir pegawai tata usaha sekolah.
e. Mengurus pengarsipan surat menyurat.
f. Membuat usul kenaikan pangkat.
g. Menyusun laporan pelaksanaan kegiatan pengurusan ketatausahaan sekolah secara berkala.

3. Wakasek Kesiswaan

a. Melaksanakan penerimaan siswa baru.
b. Membuat penataran atau orientasi.
c. Melakukan bimbingan dan konseling.
d. Menyusun program pembinaan siswa.
e. Melakukan pembinaan terhadap kegiatan ekstrakulikuler.

4. Wakasek Kurikulum

a. Mengurus pembagina tugas mengajar atau penyusunan jadwal
b. Penyusunan perangkat program pengajaran.
c. Penyajian pelajaran atau pelaksanan KBM.
d. Melakukan evaluasi terhadapa ulangan harian, ulangan umum, dan ujian akhir.
e. Mengurus kenaikan kelas atau pemilihan program.
f. Meyusun laporan evaluasi.

5. Wakasek Humas

a. Membina hubungan dengan pengurus yayasan.
b. Mengatur dan menyelenggarakan rapat pleno.
c. Mengatur dan menyelenggarakan rapat pengurus.
d. Mengatur dan menyelenggarakan rapat orang tua siswa.
e. Melakukan konsultasi dengan instansi terkait.

6. Guru

I. Membuat Program Semester, meliputi :
   a. Tujuan Pembelajaran Umum.
   b. Tujuan Pembelajaran Khusus.

c. Tema/Konsep/Pokok Bahasan/Sub Pokok bahasan.
d. Alokasi Waktu.
e. Pelaksanaan.

II. Membuat Satuan Pelajaran/Silabus, meliputi
a. Sistematika.
b. Perumusan Tujuan Pembelajaran Khusus, Standard Kompetensi, Kompetensi dasar.
c. Perumusan KBM/Pengalaman Belajar Metode, dan Strategi.
d. Perumusan Materi Pembelajaran.
e. Menentukan alat/media pelajaran dan buku sumber.
f. Menentukan jenis dan alat penilaian/tagihan.

III. Buku Administrasi Nilai Prestasi Belajar Siswa, meliputi :
a. Buku Daftar Nilai.
b. Buku Absensi Siswa.
c. Buku Kumpulan Tugas/Kegiatan Ko Kurikuler.
d. Buku Catatan kesulitan belajar siswa.
e. Buku Catatan Daya Serap Siswa per Semester dan per Kelas.
f. Program Perbaikan /Remedial.

IV. Meneliti catatan tentang pencapaian target kurikulum, meliputi :
a. Penyesuaian materi.
b. Prosentase pencapaian materi.

V. Memperhatikan dan melaksanakan pengelolaan kelas, meliputi:
a. Ketepatan jam masuk dan keluar kelas.
b. Meneliti kehadiran siswa.
c. Modifikasi tingkah laku, iklim sosio emosional, disiplin siswa.

VI. Memperhatikan teknik pengelolaan kegiatan instruksiasi, meliputi :
a. Pre Test (apresiasi).
b. Kejelasan penyampaian materi.
c. Penerapan metode dan strategi.
d. Pemanfaatan alat bantu/media dan sumber lainnya.
e. Pelaksanaan evaluasi hasil belajar.

f. Membuat rangkuman hasil belajar.
g. Pemberian tugas / kokurikuler.
h. Penampilan.
i. Penguasaan materi.
j. Komunikasi/interaksi guru dan siswa.

VII. Memiliki dan mempersiapkan peralatan, meliputi :
a. Alat yang sesuai dengan pelajaran.
b. Buku Kurikulum /GBPP.
c. Buku pegangan Guru.
d. Buku Acuan.
e. Buku Nilai.
f. Buku Absensi.
g. Buku Masalah dan penanggulangannya.

7. Wali Kelas

a. Menyusun Program Kerja Wali Kelas
b. Pengelolaan dan Pembinaan kelas
c. Penyelenggaraan Administrasi Kelas, meliputi:
1. Denah tempat duduk Siswa
2. Papan Absensi Siswa
3. Jadwal Pelajaran Kelas
4. Jadwal Piket Kelas
5. Buku Absensi Siswa
6. Buku Agenda KBM
7. Tata tertib di kelas/siswa.
8. Struktur Organisasi kelas
d. Penyusunan/pembuatan statistik bulanan siswa
e. Pengisian daftar kumpulan nilai siswa
f. Pencatatan mutasi siswa
g. Pengisian buku laporan pendidikan
h. Pembagian buku laporan pendidikan

2.2 Landasan Teori

Landasan teori merupakan kumpulan dari teori – teori yang menjadi dasar dalam pembangunan aplikasi ini yang dikutip dari berbagai buku dan sumber.

2.2.1 Pengertian *Game*

Menurut Kimpraswil mengatakan bahwa definisi permainan adalah usaha olah diri (olah pikiran dan olah fisik) yang sangat bermanfaat bagi pengingkatan dan pengembangan motivasi, kinerja, dan prestasi dalam melaksanankan tugas dan kepentingan organisasi dengan lebih baik.

Menurut Joan Freeman dan Utami Munandar mendefinisikan permainan sebagai suatu aktifitas yang membantu anak mencapai perkembangan yang utuh, baik fisik, intelektual, sosial, moral, dan emosional.

Selama ini mungkin kita mengira bahwa tujuan game hanyalah untuk menghibur diri atau bersenang – senang. Namun sesungguhnya dengan game kita juga bisa belajar sambil bermain.

Adapun edugame menurut Karl Royle, akademisi yaitu *game* yang didesain sesuai tujuan kurikulum tertentu untuk dipelajari seringkali *edugame* dibuat untuk membantu pelajaran secara praktisnya.

Menurut Marc Prensky, pembicara dan penulis yaitu *game* yang didesain untuk belajar tapi tetap bisa menawarkan bermain dan bersenang – senang. Edugame adalah gabungan dari konten edukasi, prinsip pembelajaran, dan game komputer.

Menurut beberapa pendapat para ahli tersebut menyimpulkan definisi permainan adalah suatu aktifitas yang dilakukan oleh beberapa anak untuk mencari kesenangan yang dapat membentuk proses kepribadian anak dan membantu anak mencapai perkembangan fisik, intelektual, sosial, moral, dan emosional [6].

2.2.2 Jenis – Jenis Game

Dalam permainan ada beberapa jenis *game* yang bisa dimainkan diantaranya sebagai berikut [9].:

1. *Actiongame*

   Action game membutuhkan refleks yang cepat dan waktu yang tepat untuk menaklukan rintangan. Tujuannya adalah menguasai musuh tanpa harus menghancurkan diri sendiri dan merupakan *real-time game play*. Adapun subgenre *action game* yaitu platformers, *shooters* (FTP & TPS), *fighting*, dan *racing*. Contoh gamenya yaitu : Super Mario Bros, Tekken, Burnout Paradise, Crysis, dan lain – lain.

2. *AdventureGame*

   Adventure game bukanlah sebuah *game* persaingan atau simulasi serta tidak menawarkan sebuah proses yang harus diatur atau mengalahkan sebuah musuh. *Adventure game* merupakan sebuah *game* yang berinteraksi melalui cerita tetang suatu karakter yang dikendalikan oleh pemain. Contoh *game*nya yaitu : God of War, Uncharted 2, Legend Of Zelda, dan lain – lain.

3. *Role Playing Game* (RPG)

   RPG merupakan *game* yang mana karakter pemain mempunyai kemampuan yang dihadirkan melalui statistika. *Gameplay* melibatkan karakter yang mengeksplorisasi dan meyelesaikan pencarian yang dapat membangun sebuah statistika dan harta benda pemain. RPG biasanya mempunyai garis cerita yang sangat kuat. Contoh g*ame*nya yaitu : Elder Scrolls IV, Dragon Age, Star Wars, dan lain – lain.

4. *StrategyGame*

   Strategy game membutuhkan pemain untuk mengambil peran kepemimpinan dan melihat setiap detail yang disediakan skenario. *Gameplay* berfokus dalam strategi dan perencanaan yang matang serta mengatur sumber daya untuk menang. *Game strategy* dibagi 2 yaitu :

   a. *Real Time Strategy* (RTS)

      Pada game ini kita dapat mengendalikan pasukan secara langsung dari mencari sumber daya hingga menghancurkan musuh. Semua pertempuran itu dapat kita saksikan secara langsung. Contohnya *game* Warcraft.

b. *Turn Base Strategy* (TBS)

Pada game ini selain mengendalikan karakter utama, kita mengendalikan pasukan dan kota kita untuk memenangkan pertarungan. Biasanya kita memainkan *game* ini diatas peta. Contoh game Civilization.

5. Simulation game

Simulasi *game* adalah genre yang mementingkan *realism*. Segala faktor pada game ini sangat diperhatikan agar mirip di dunia nyata. Segala nilai, material, referensi, dan faktor lainnya adalah berdasarkan dunia nyata. Cara memainkannya juga berbeda karena biasanya kontrol yang dimiliki cukup rumit. Contohnya *game* Flight simulator, The Sims, SimCIty, dan lain – lain.

6. *Massively Multiplayer Online* (MMO)

Pemain bermain langsung dalam waktu yang sama dengan jumlah pemain mencapai seratus atau seribu pemain. MMO terkenal karena memiliki alasan seperti interaksi sosial dalam jumlah besar, kemampuan bermain baik menjadi komputer atau pemain lain, dan bermain sebagai tim. Adapun yang genre lain yang termasuk MMO diantaranya MMOFPS, MMORTS, MMO sport, dan lain – lain. Contohnya *game* World Of Warcraft, EverQuest, Lord Of The Rings, Eve, City Of Heroes, dan lain – lain.

7. Sport game

Sport game adalah permainan olahraga dengan sistem permainan akan berbeda – beda tergantung jenis olahraga yang menjadi tema game tersebut. Contohnya *game* NFL Blitz, PES, Winning Eleven, dan lain – lain.

8. *Shooting game*

Shooting game adalah permainan tembak menembak untuk menghancurkan beberapa objek atau musuh. *Shooting game* dibagi lagi menjadi dua jenis yaitu:

a. *First Person Shooting* (FPS)

FPS adalah *game* tembak menembak yang memiliki ciri utamanya adalah penggunaan sudut pandang orang pertama yang membuat kita dibelakang senjata. Contohnya *game* Counter Strike.

b. *Third Person Shooting* (TPS)

TPS adalah *game* yang mirip dengan FPS dengan *gameplay* tembak menembak hanya saja pada sudut pandang yang digunakan dalam *game* ini adalah orang ketiga. Contohnya *game* Lost Planet.

9. *Puzzle game*

Puzzle game adalah permainan dalam memecahkan suatu teka – teki. *Puzzle game* biasanya variasi pada satu tema. Sebuah *puzzlegame* membutuhkan tantangan tapi tidak terlalu susah, kelihatan menarik, dan menyenangkan untuk bermain. Contohnya *game* kartu Solitaire.

2.2.3 BoardGame

Board game adalah jenis permainan dimana alat – alat atau bagian permainan ditempatkan, dipindahkan, atau digerakkan pada permukaan yang telah ditandai atau dibagi – bagi menurut seperangkat aturan. Permainan mungkin didasarkan pada strategi murni, kesempatan atau campuran dari keduanya dan biasanya memiliki tujuan yang harus dicapai seseorang. *Board game* versi awal merepresentasikan pertempuran antara dua pasukan dan *board game* terbaru masih didasarkan pada tujuan untuk mengalahkan pemain lawan dalam hal jumlah, posisi menang atau poin tertinggi.

Ada berbagai jenis dan gaya*board game*. Representasi *boardgame* dari situasi kehidupan nyata bervariasi mulai dari tidak memiliki tema yang inheren, seperti halma, hingga yang memiliki tema dan narasi yang spesifik seperti Cluedo. Aturan permainannya berkisar dari yang sangat sederhana seperti permainan tic-tac-toe, hingga menggambarkan sebuah permainan alam semesta dengan sangat rinci misalnya permainan Dungeons & Dragon, meskipun sebagian besar permainan ini adalah berdasarkan alur cerita dan *board*nya adalah tambahan yang berfungsi untuk memvisualisasikan skenario. Lamanya waktu yang diperlukan untuk belajar bermain atau menguasai permainan dapat bervariasi dari *game* ke *game*. Waktu belajar tidak harus berkolerasi dengan jumlah atau kompleksitas

peraturan, beberapa permainan seperti catur dan go memiliki aturan yang masih bisa menimbulkan skenario yang kompleks [8].

2.2.4 Game Edukasi

Edukasi atau pendidikan menurut undang - undang no. 20 tahun 2003 tentang sistem pendidikan nasional, pasal1 adalah usaha sadar dan terencana untuk mewujudkan suasana belajar dan proses pembelajaran agar peserta didik secara aktif mengembangkan potensi dirinya untuk memiliki kekuatan spiritual keagamaan, pengendalian diri, kepribadian, kecerdasan, akhlak mulia serta ketrampilan yang diperlukan dirinya, masyarakat, bangsa, dan negara.

Game edukasi adalah *game* yang didesain agar pemainnya bisa belajar dengan cara yang lebih menyenangkan. Metode yang digunakan sebagai pendekatan pembelajaran dalam *game* edukasi bervariasi bisa menggunakan alat peraga, komputer atau hal lainnya yang menunjang dalam pembelajaran tersebut [3].

2.2.5 Multimedia

Multi artinya banyak atau bermacam – macam. Media adalah sesuatu yang dipakai untuk menyampaikan atau membawa sesuatu. Multimedia dapat diartikan sebagai penggunaan beberapa media yang berbeda untuk menggabungkan dan menyampaikan informasi dalam bentuk text, audio, grafik, animasi, dan video.

Multimedia dapat digunakan dalam bidang periklanan yang efektif dan interaktif. Bidang pendidikan dalam penyampaian bahan pengajaran secara interaktif dan dapat mempermudah pembelajaran karena didukung oleh berbagai aspek suara, video, animasi, teks, dan grafik. Dalam bidang jaringan dan internet yang membantu untuk pembuatan website yang menarik, informatif, dan interaktif.

Multimedia mampu mengubah sesuatu yang konvensional menjadi sesuatu yang modern seperti [15]:

1. Mengubah tempat kerja

Dengan adanya *teleworking* para pekerja dapat melakukan pekerjaannya tidak harus dari kantor. Contoh *software*yang medukung t*eleworking/telecommuting* yaitu n*etmeeting*.

2. Mengubah cara belanja

   Homeshopping/teleshopping dapat dilakukan dengan menggunakan internet kemudian barang datang dengan sendirinya.

3. Mengubah cara bisnis

   Banyak perusahaan menggunakan sistem jual beli *online*, bank menggunakan cara*online banking*.

4. Mengubah cara memperoleh informasi

   Orang – orang mulai menggunakan internet dan berbagai *software* untuk mencari informasi. Misalnya membaca koran*online*, detik.com, menggunakan *software* kesehatan, belajar gitar dari *software* dan masih banyak lagi.

5. Mengubah cara belajar

   Sekolah mulai menggunakan komputer multimedia, belajar *online*, menggunakan *e-book.*

6. Internet multimedia juga mulai bersaing dengan televise dan radio.

2.5.2.1 Objek – Objek Multimedia

Objek – objek multimedia merupakan penghubung agar bisa melakukan interaksi satu sama lain, diantaranya yaitu [15]:

1. Teks

   Bentuk data multimedia yang paling mudah tersimpan dan dikendalikan adalah teks. Teks dapat membentuk kata , surat, atau narasi dalam multimedia, yang menyajikan bahasa kita. Meskipun mungkin saja ada multimedia tanpa teks kebanyakan sistem multimedia menggunakan teks sebab teks sangat efektif untuk menyampaikan ide, serta memberikan panduan kepada pengguna.

2. Gambar

   Gambar merupakan suatu representasi spatial dari suatu objek dalam pandangan 2D atau 3D. Gambar memiliki peran untuk membantu teks agar lebih mudah dalam memberikan instruksi dan menghubungkan *user* dengan

perangkat antarmuka sebuah aplikasi. Menurut Hofstetter ada lima jenis gambar, yaitu :

a. *Bitmaps*

   Gambar yang disimpan berdasarkan kumpulan *pixel* yang sesuai dengan *grid* titik pada layar komputer.

b. *Vector Images*

   Gambar yang berdasarkan dengan kurva, garis, dan shapes yang telah ditentukan dan diatur oleh algoritma matematika.

c. *Clip Arts*

   Gambar yang sudah tersedia di dalam *clip library* dan dapat dipakai dalam produksi multimedia dengan tujuan untuk menghemat waktu.

d. *Digitized Picture*

   Gambar yang didapatkan dengan mengambil frame yang terdapat dalam video sehingga dapat digunakan dalam aplikasi multimedia.

e. *Hyperpictures*

   Gambar yang telah diberikan sebuah alamat link. Saat user melakukan klik pada *hyperpicture* maka user akan berpindah sesuai dengan alamat link yang terdapat pada *hyperpicture* tersebut.

3. Suara

   Suara adalah getaran suatu benda yang berupa sinyal analaog dengan amplitudo yang berubah secara kontinyu terhadap waktu. Diperlukan narasi atau *sound* yang menyertai dan menjelaskan informasi yang disampaikan agar lebih mudah dipahami. Format dasar audio yaitu wav dimana data tidak terkompres sehingga seluruh *sample* audio disimpan semuanya dihardisk sedangkan midi (*music instrument digital interface*) memberikan cara yang lebih efisien dalam merekam music dan kapasitas data yang dihasilkan juga jauh lebih kecil.

4. Animasi

   Animasi adalah "*illusion of motion*" yang dibuat dari *image* statis yang ditampilkan secara berurutan. Frame bisa dihasilkan dari komputer, dari fotografi atau dari gambar lukisan. Ketika frame – frame tersebut digabungkan maka terdapat ilusi perubahan gambar. Jenis – jenis animasi yaitu :

a. Animasi Cel

Kata cel berasal dari kata "*celluloid*" yang merupakan materi yang digunakan untuk membuat film gambar bergerak. Animasi cel biasanya merupakan lembaran – lembaran yang membentuk animasi tunggal.

b. Animasi Frame

Animasi frame merupakan bentuk animasi paling sederhana, misalnya ketika kita membuat gambar yang berbeda – beda kemudian kita putar secara cepat maka akan terlihat gambar bergerak.

c. Animasi Sprite

Pada animasi sprite, gambar digerakkan dengan latar belakang yang diam. Sprite adalah bagian dari animasi yang bergerak secara mandiri.

d. Animasi Path

Animasi path adalah animasi dari objek yang bergerak sepanjang garis kurva yang ditentukan dengan lintasan.

e. Animasi Spline

Spline adalah representasi matematis dari kurva sehingga gerakkan objek tidak hanya mengikuti garis lurus melainkan berbentuk kurva.

f. Animasi Vektor

Vektor adalah garis yang memiliki ujung pangkal, arah, dan panjang. Animasi vektor mirip dengan animasi sprite tapi animasi sprite menggunakan bitmap sedangkan animasi vektor menggunakan rumus matematika untuk menggambarkan spritenya.

g. Animasi Karakter

Animasi karekter biasanya terdapat difilm kartun, semua bagian dalam film kartun selalu bergerak bersamaan.

5. Video

Video adalah teknologi untuk menangkap, merekam, memproses, mentransmisikan, dan menata ulang gambar bergerak. Biasanya menggunakan

film seluloid, sinyal elektronik atau media digital. Video menyediakan sumber daya yang kaya dan membuat aplikasi multimedia lebih hidup.

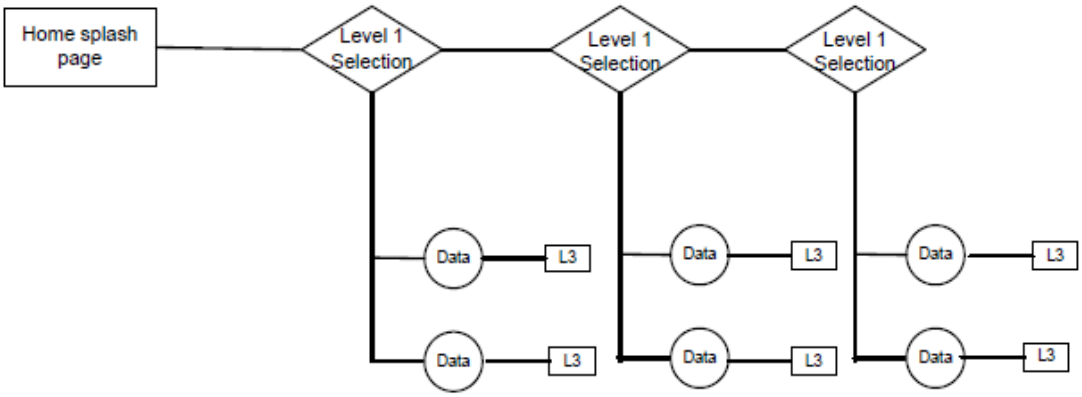
2.5.2.2 Struktur Informasi Multimedia

Struktur sistem informasi multimedia merupakan bagian yang sangat penting dari keseluruhan pembuatan aplikasi multimedia. Struktur ini berguna untuk memvisualisasikan seluruh relasional dan apliksai yang sedang dibangun. Struktur ini menjelaskan organisasi file dari macromedia sebagai perangkat lunak utama, grafik dan sumber daya lain, sehingga tidak hanya memudahkan dalam menemukan file tertentu, tetapi juga memudahkan untuk melakukan revisi padatiap komponen dalam aplikasi Multimedia ketika dibutuhkan.

Ada empat struktur sistem informasi multimedia menurut Laura Lemay di mana masing-masing struktur memiliki tujuan dan kegunaan yang berbeda-beda. Adapun struktur yang dimaksud adalah sebagai berikut [15]:

1. Struktur Linear

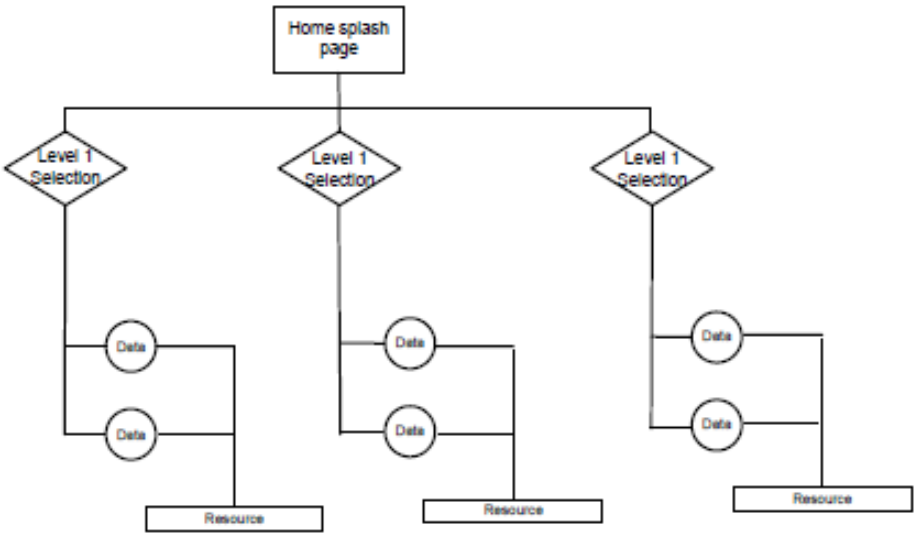
   Desain struktur linear cocok untuk menonjolkan arus antar level bila ingin menonjolkan topik tiap level. Berikut ini gambar dari struktur linear, yaitu :



Gambar 2.3 Struktur Linear

2. Struktur Hirarki

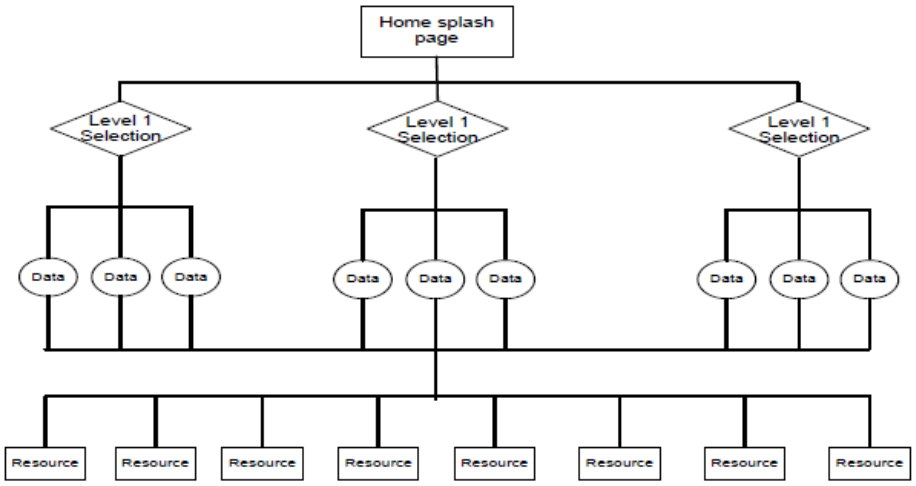
   Struktur ini lebih cocok untuk menonjolkan semua level secara bersamaan. Berikut ini gambar dari struktur hirarki, yaitu :



Gambar 2.4 Struktur Hirarki

3. Struktur Piramida

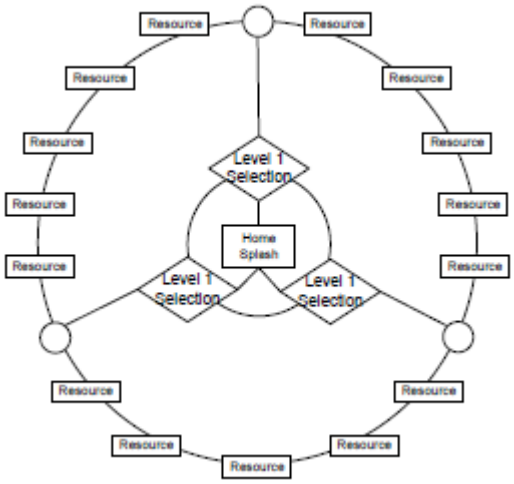
   Struktur ini akan lebih cocok apabila digunakan untuk menunjuk semua level yang memiliki *resource*yang sama. Berikut ini gambar dari struktur piramida, yaitu :



Gambar 2.5 Struktur Piramida

4. Struktur Polar

   Dalam struktur polar kita akan melihat bahwa semua level memiliki *resource* yang universal. Berikut ini gambar dari struktur polar, yaitu :



Gambar 2.6 Struktur Polar

2.2.6 Materi Pembelajaran

Bahasa bukan hanya sebagai suatu bidang kajian, melainkan sebagai faktor sentral dalam perkembangan intelektual, sosial, dan emosional peserta didik. Penguasaan Bahasa Arab menjadi persyaratan penting bagi keberhasilan individu dalam menjawab tantangan zaman di era globalisasi.

Melalui pembelajaran bahasa Arab dapat dikembangkan keterampilan peserta didik dalam berkomunikasi lisan dan tulisan untuk memahami dan menyampaikan informasi, pikiran, dan perasaan. Mata pelajaran Bahasa Arab merupakan suatu mata pelajaran yang diarahkan untuk mendorong, membimbing, mengembangkan, dan membina kemampuan serta menumbuhkan sikap positif terhadap Bahasa Arab baik reseptif maupun produktif.

Kemampuan reseptif yaitu kemampuan untuk memahami pembicaraan orang lain dan memahami bacaan. Kemampuan produktif yaitu kemampuan menggunakan bahasa sebagai alat komunikasi baik secara lisan maupun secara tertulis. Kemampuan berbahasa Arab serta sikap positif terhadap bahasa Arab tersebut sangat penting dalam membantu memahami sumber ajaran Islam yaitu al-Qur'an dan Hadits, serta kitab-kitab berbahasa Arab yang berkenaan dengan Islam bagi peserta didik [7].

2.2.6.1 Bahasa Arab Al-Qur'an

Bahasa Arab al-quran yaitu membahas mengenai ilmu tajwid, kosakata dan terjemahan yang ada pada juz ke-30. Materi yang dibahasa pada kelas VII berasal dari surat at-takatsur sampai surat an-nas. Dimana siswa diharapkan bisa meyimak, membaca, berbicara dan menulis dari materi yang dipelajari. Berikut ini materi surat yang diajarkan pada mata pelajaran bahasa arab al-qur'an [14]:

1. Surat At-Takatsur

Surat At Takaatsur terdiri atas 8 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah surat Al Kautsar. Dinamai *At Takaatsur* (bermegah-megahan) diambil dari perkataan *At Takaatsur* yang terdapat pada ayat pertama surat ini.

Surat ini menjelaskan mengenai keinginan manusia untuk bermegah-megahan dalam soal duniawi, sering melalaikan manusia dari tujuan hidupnya. Dia baru menyadari kesalahannya itu setelah maut mendatanginya; manusia akan ditanya di akhirat tentang nikmat yang dibangga-banggakannya.

2. Surat Al-Asr

Surat Al 'Ashr terdiri atas 3 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah surat Alam Nasyrah. Dinamai *Al 'Ashr* (masa) diambil dari perkataan *Al 'Ashr* yang terdapat pada ayat pertama surat ini.

Surat ini menjelaskan mengenai semua manusia berada dalam keadaan merugi apabila dia tidak mengisi waktunya dengan perbuatan-perbuatan baik.

3. Surat Al-Humazah

Surat Al Humazah terdiri atas 9 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah surat Al Qiyaamah. Dinamai *Al Humazah* (pengumpat) diambil dari perkataan *Humazah* yang terdapat pada ayat pertama surat ini.

Surat ini menjelaskan mengenai ancaman Allah terhadap orang-orang yang suka mencela orang lain, suka mengumpat dan suka mengumpulkan harta tetapi tidak menafkahkannya di jalan Allah.

4. Surat Al-Fiil

Surat ini terdiri atas 5 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah surat Al Kaafirun. Nama *Al Fiil* diambil dari kata *Al Fiil* yang terdapat pada ayat pertama surat ini, artinya *gajah*. Surat Al Fiil mengemukakan cerita

pasukan bergajah dari Yaman yang dipimpin oleh Abrahah yang ingin meruntuhkan Ka'bah di Mekah. Peristiwa ini terjadi pada tahun Nabi Muhammad s.a.w. dilahirkan.

5. Surat Quraisy

Surat ini terdiri atas 4 ayat, termasuk golongan surat-surat Makkiyyah dan diturunkan sesudah surat At Tiin. Nama *Quraisy* diambil dari kata *Quraisy* yang terdapat pada ayat pertama, artinya suku Quraisy. Suku Quraisy adalah suku yang mendapat kehormatan untuk memelihara Ka'bah.

Surat ini menjelaskan mengenai peringatan kepada orang Quraisy tentang nikmat-nikmat yang diberikan Allah kepada mereka karena itu mereka diperintahkan untuk menyembah Allah.

6. Surat Al-Ma'un

Surat ini terdiri atas 7 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah surat At Taakatsur. Nama *Al Maa'uun* artinya barang-barang yang berguna.

Surat ini menjelaskan mengenai beberapa sifat manusia yang dipandang sebagai mendustakan agama. Ancaman terhadap orang-orang yang melakukan shalat dengan lalai dan riya**.**

7. Surat Al-Kautsar

Surat al-kautsar terdiri dari 3 ayat, termasuk golongan surat – surat Makiyyah diturunkan sesudah surat Al-'Adiyat. Al-Kautsar yang mempunyai arti nikmat yang banyak. Surat ini sebagai penghibur hati Nabi Muhammad S.A.W.

Surat ini menjelaskan mengenai Allah telah melimpahkan nikmat yang banyak karena itu bersembahyang dan berkorbanlah; Nabi Muhammad s.a.w. akan mempunyai pengikut yang yang banyak sampai hari kiamat dan akan mempunyai nama yang baik di dunia dan di akhirat, tidak sebagai yang dituduhkan pembenci-pembencinya.

8. Surat Al-Kafirun

Surat Al Kaafiruun terdiri atas 6 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah surat Al Maa'uun. Dinamai *Al Kaafiruun* (orang-orang kafir). Surat ini menjelaskan mengenai pernyataan Tuhan yang disembah Nabi

Muhammad s.a.w. dan pengikut-pengikutnya bukanlah apa yang disembah oleh orang-orang kafir, dan Nabi Muhammad s.a.w. tidak akan menyembah apa yang disembah oleh orang-orang kafir.

9. Surat An-Nasr

Surat An Nashr terdiri atas 3 ayat, termasuk golongan surat-surat Madaniyyah yang diturunkan di Mekah sesudah surat At Taubah. Dinamai *An Nashr* (pertolongan). Surat ini menjelaskan mengenai janji bahwa pertolongan Allah akan datang dan Islam akan mendapat kemenangan; perintah dari Tuhan agar bertasbih memuji-Nya, dan minta ampun kepada-Nya di kala terjadi peristiwa yang menggembirakan.

10. Surat Al-Lahab

Surat ini terdiri atas 5 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah surat Al Fath. Nama *Al Lahab* diambil dari kata *Al Lahab* yang terdapat pada ayat ketiga surat ini yang artinya gejolak api. Surat ini juga dinamakan surat*Al Masad*.

Surat ini menjelaskan mengenai cerita Abu Lahab dan isterinya yang menentang Rasul s.a.w. Keduanya akan celaka dan masuk neraka. Harta Abu Lahab, tak berguna untuk keselamatannya demikian pula segala usaha-usahanya.

11. Surat Al-Ikhlas

Surat ini terdiri atas 4 ayat, termasuk golongan surat-surat Makkiyyah, diturunkan sesudah sesudah surat An Naas. Dinamakan *Al Ikhlas* karena surat ini sepenuhnya menegaskan kemurnian keesaan Allah S.W.T

Surat ini menjelaskan mengenai penegasan tentang kemurnian keesaan Allah s.w.t. dan menolak segala macam kemusyrikan dan menerangkan bahwa tidak ada sesuatu yang menyamai-Nya.

12. Surat Al-Falaq

Surat ini terdiri atas 5 ayat, termasuk golongan surat-surat Makkiyah, diturunkan sesudah surat Al Fiil. Nama *Al Falaq* diambil dari kata *Al Falaq* yang terdapat pada ayat pertama surat ini yang artinya waktu subuh. Surat ini menjelaskan mengenai perintah agar kita berlindung kepada Allah s.w.t. dari segala macam kejahatan.

13. Surat An-Nas

Surat ini terdiri atas 6 ayat, termasuk golongan surat-surat Makkiyah, diturunkan sesudah surat Al Falaq. Nama *An Naas* diambil dari *An Naas* yang berulang kali disebut dalam surat ini yang artinya manusia.

Surat ini menjelaskan mengenai perintah kepada manusia agar berlindung kepada Allah dari segala macam kejahatan yang datang ke dalam jiwa manusia dari jin dan manusia.

2.2.6.2 Ilmu Tajwid

Ilmu tajwid adalah ilmu yang dipergunakkan untuk mengetahui tempat keluarnya huruf (makhraj) dan sifat – sifatnya serta bacaan – bacaannya. Tujuan ialah agar orang dapat membaca ayat – ayat Al-Quran dengan fasih (terang dan jelas) dan cocok dengan ajaran – ajaran Nabi Muhammad S.A.W serta dapat menjaga lisannya dari kesalahan – kesalahan ketika membaca Al-Quran. Hukum belajar ilmu tajwid itu wajib kifayah, tetapi mengamalkan ilmu tajwid wajib ‘ain bagi orang Islam baik laki – laki maupun wanita. Berikut ini materi ilmu tajwid yang diajarkan pada kelas VII yaitu hukum nun mati dan hukum mim mati [13].

1. **Hukum Nun Mati (نْ)**

   a. Idzhar

   Idzhar yaitu membaca dengan terang atau jelas dimana makhrajnya dengan tidak bercampur ghunnah dan tasydid. Hukum idzhar berlaku apabila(نْ)(nun mati) atau(ً ٍ ٌ)(tanwin) bertemu dengan huruf idzhar(ء ع غ ح خ ه)maka dibacanya harus jelas. Berikut contoh dari hukum tajwid idzhar, yaitu :

Tabel 2.1 Hukum Tajwid Idzhar

عَنْ أَنفُسِكُمْ	ء
طَبَقًا عَن طَبَقٍ	ع

أَجْرٌ غَيْرُ مَمْنُونٍ	غ
وَإِنْ خِفْتُمْ	خ
سَبِيلٍ حَتَّىٰ	ح
قَلَّ مِنْهُ	ه

b. Idgham Bighunnah

Idgham bighunnah yaitu memasukkan huruf satu kedalam huruf yang lain. Hukum idgham bighunnah berlaku apabila (نْ) (nun mati) atau ً ٌ ٍ (tanwin) bertemu dengan huruf idgham bighunnah (ي و م ن) maka dibacanya berdengung. Berikut contoh dari hukum tajwid idgham bighunnah, yaitu :

Tabel 2.2 Hukum Tajwid Idgham Bighunnah

حِسَابًا يَسِيرًا	ي
وَفَضْلٍ وَأَنَّ اللَّهَ	و
نَصِيبًا مَّفْرُوضًا	م
عَن نَّفْسٍ	ن

c. Idgham Bilaghunnah

Idgham bilaghunnah yaitu memasukkan huruf satu kedalam huruf yang lain. Hukum idgham bilaghunnah berlaku apabila(نْ) (nun mati) atau ً ٌ ٍ (tanwin) bertemu dengan huruf idgham bilaghunnah(ل ر) maka dibacanya tidak dengung. Berikut contoh dari hukum tajwid idgham bilaghunnah, yaitu :

Tabel 2.3 Hukum Tajwid Idgham Bilaghunnah

بَيَانٌ لِّلنَّاسِ	ل
وَمِن رِّبَاطِ	ر

d. Ikhfa

Ikhfa yaitu samar. Hukum ikhfa berlaku apabila (نْ) (nun mati) atau ً ٍ ٌ (tanwin) bertemu dengan huruf ikhfa (ت ث ج د ذ ز س ش ص ض ط ظ ف ق ك) maka dibacanya mendengung. Berikut contoh dari hukum tajwid ikhfa, yaitu :

Tabel 2.4 Hukum Tajwid Ikhfa

إِن كُنتُمْ صَٰدِقِينَ	ك
مِّأْئَةٌ صَابِرَةٌ	ص
حَلَٰلًا طَيِّبًا	ط
عَن سَبِيلِ ٱللَّهِ	س
يَنظُرُونَ	ظ

e. Iqlab

Iqlab yaitu mengganti bacaan nun mati atau tanwin dengan bacaan mim yang disamarkan dengan mendengung. Hukum iqlab berlaku apabila (نْ)(nun mati) atau ً ٍ ٌ(tanwin) bertemu dengan huruf iqlab (ب)maka dibacanya mengganti bunyi nun mati atau tanwin dengan mim(م) dan mendengung. Berikut contoh dari hukum iqlab, yaitu :

Tabel 2.5 Hukum Tajwid Iqlab

خَبِيرُۢ بِمَا	ب
قَوۡمِۢ بَيۡنَكُمۡ	
تَكُنۢ بَيۡنَكُمۡ	

2. Hukum Mim Mati (مْ)

a. Idzhar Syafawi

Idzhar syafawi apabila mim mati (مْ) bertemu dengan huruf hijaiyah ا ت ث ج ح خ د ذ ر ز س ش ص ض ط ظ ع غ ف ق ك ل ن و ي ه selain mim dan ba (م ب) maka membacanya suara mim mati (مْ) dibaca jelas dan tidak mendengung. Berikut contoh dari hukum tajwid idzhar syafawi, yaitu :

Tabel 2.6 Hukum Tajwid Idzhar Syafawi

لَقَوۡكُمۡ قَالُوٓاْ	ق
أَن يُمِدَّكُمۡ رَبُّكُم	ر
لَهُمۡ تَعَالَوۡاْ	ت
أَيۡدِيهِمۡ ثُمَّ	ث
تَمۡسَسۡكُمۡ حَسَنَةٞ	ح

b. Ikhfa Syafawi

Ikhfa syafawi apabila mim mati (مْ) bertemu dengan huruf (ب) maka membacanya suara mim mati (مْ) dengan dengung. Berikut contoh dari hukum tajwid ikhfa syafawi, yaitu :

Tabel 2.7 Hukum Tajwid Ikhfa Syafawi

ذَٰلِكُم بَلَآءٌ	ب
يَعِظُكُم بِهِۦٓ	
رَبُّكُم بِثَلَٰثَةِ	

c. Idgham Mitslain

Idgham mitsalin apabila mim mati (مْ) bertemu dengan huruf mim (م)maka membacanya suara mim ditasydidkan dan dibaca dengan dengung. Berikut contoh dari hukum tajwid idgham mitslain, yaitu :

Tabel 2.8 Hukum Tajwid Idgham Mitslain

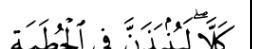
بِأَفْوَٰهِهِم مَّا لَيْسَ	م
قَدْ أَصَبْتُم مِّثْلَيْهَا	
أَنَّهُم مُّلَٰقُوا۟	

2.2.6.3 Kosakata Al-Quran

Kosakata Al-Quran diambil dari surat at-takatsur sampai surat an-nas dimana berisi kosakata yang nantinya akan diartikan per kata pada tiap surat – surat tersebut. Berikut contoh dari kosakata al-qur'an, yaitu :

Tabel 2.9 Kosakata Al-Quran

ٱلنَّاسِ "manusia"	بِرَبِّ "Tuhan Pemelihara "	أَعُوذُ "aku berlidung"	قُلْ "katakanlah"	قُلْ أَعُوذُ بِرَبِّ ٱلنَّاسِ
	ٱلْمِسْكِينِ "orang miskin"	عَلَىٰ طَعَامِ "untuk memberi makan"	وَلَا يَحُضُّ "dan tidak menganjurkan"	وَلَا يَحُضُّ عَلَىٰ طَعَامِ ٱلْمِسْكِينِ

	 “dalam Hutamah”	 “sungguh dia dilemparkan”	 “sekali-kali tidak”	

2.3 Kecerdasan Buatan (Artificial Intelegence)

Kecerdasan buatan merupana bagian dari ilmu komputer yang mempelajari bagaimana membuat mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia bahkan bisa lebih baik daripada yang dilakukan manusia.

Manusia cerdas dalam menyelesaikan permasalahan karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan diperoleh dari belajar, semakin banyak bekal pengetahuan yang dimiliki manusia juga diberi akal untuk melakukan penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang dimiliki. Tanpa memiliki kemampuan untuk menalar dengan baik, manusia dengan segudang pengalaman dan pengetahuan tidak akan dapat menyelesaikan masalah dengan baik. Demikian juga dengan kemampuan yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik [12].

Agar mesin bisa cerdas (bertindak seperti dan sebaik manusa) maka harus diberi bekal pengetahuan dan mempunyai kemampuan untuk menalar. Bagian utama yang dibutuhkan untuk aplikasi kecerdasan buatan ada 2 yaitu :

1. Basis pengetahuan (*knowledge base*) berisi fakta – fakta, teori, pemikiran dan hubungan antara satu dengan lainnya.
2. Motor inferensi (*inference engine*) kemampuan menarik kesimpulan berdasarkan pengalaman.

2.3.1 Perbedaan Kecerdasan Buatan dan Kecerdasan Alami

Perbedaan kecerdasan buatan dan kecerdasan alami dapat dilihat pada keterangan berikut. Kelebihan kecerdasan buatan, yaitu [12]:

1. Lebih bersifat permanen. Kecerdasan alami bisa berubah karena sifat manusia pelupa. Kecerdasan buatan tidak berubah selama sistem komputer dan program tidak mengubahnya.
2. Lebih mudah diduplikasi dan disebarkan. Mentransfer pengetahuan manusia dari satu orang ke orang lain membutuhkan proses yang sangat lama dan keahlian tidak akan pernah dapat didupilkasi dengan lengkap. Jadi jika pengetahuna terletak pada suatu sistem komputer, pengetahuan tersebut dapat disalin dari komputer tersebut dan dapat dipindahkan dengan mudah ke komputer yang lain.
3. Lebih murah. Menyediakan layanan computer akan lebih mudah dan murah dibandingkan mendatangkan seseorang untuk mengerjakan sejumlah pekerjaan dalam jangka waktu yang sangat lama.
4. Bersifat konsisten karena kecerdasan buatan adalah bagian dari teknologi komputer sedangkan kecerdasan alami senantiasa berubah – ubah.
5. Dapat didokumentasikan. Keputusan yang dibuat komputer dapat didokumentasikan dengan mudah dengan cara melacak setiap aktivitas dari sistem tersebut. Kecerdasan alami sangat sulit untuk direproduksi.
6. Lebih cepat dan baik.

Kelebihan Kecerdasan Alami :

1. Kreatif. Manusia memiliki kemampuan untuk menambah pengetahuan, sedangkan pada kecerdasan buatan untuk menambah pengetahuan harus dilakukan melalui sistem yang dibangun.
2. Memungkinkan orang untuk menggunakan pengalaman secara langsung. Sedangkan pada kecerdasan buatan harus bekerja dengan input – input simbolik.
3. Pemikiran manusia dapat digunakan secara luas, sedangkan kecerdasan buatan sangat terbatas.

2.3.2 Perbedaan Kecerdasan Buatan dan Program Konvensional

Perbedaan kecerdasan buatan dan program konvensional dapat dilihat pada tabel berikut dibawah ini [12]:

Tabel 2.10 Perbedaan Kecerdasan Buatan dan Program Konvensional

	kecerdasan buatan	**program konvensional**
fokus pemrosesan	konsep simbolik / numerik (pengetahuan)	data dan informasi
pencarian	Heuristic	algoritma
sifat input	bisa tidak lengkap	harus lengkap
keterangan	Disediakan	biasanya tidak disediakan
struktur	kontrol dipisahkan dari pengetahuan	kontrol terintegrasi dengan data
sifat output	Kuantitatif	kualitatif
kemampuan menalar	Ya	tidak

2.3.3 Kecerdasan Buatan Pada Aplikasi Komersial

Kecerdasan buatan pada aplikasi komersial mempunyai ruang lingkup utama yang dapat dijadikan sebagai kecerdasan buatan, yaitu [12] :

1. Sistem pakar (*expert system*)

   Komputer sebagai sarana untuk menyimpan pengetahuan para pakar sehingga komputer memiliki keahlian menyelesaikan permasalahan dengan meniru keahlian yang dimiliki pakar.
2. Pengolahan bahasa alami (*natural language processing*)

   User dapat berkomunikasi dengan komputer menggunakan bahasa sehari – hari, misalnya bahasa Inggris, bahasa Indonesia, bahasa Jawa dan lain – lain.
3. Pengenalan ucapan (*speech recognition*)

   Manusia dapat berkomunikasi dengan komputer menggunakan suara.
4. Robotika dan sistem sensor.
5. *Computer vision*

   Menginterpretasikan gambar atau objek – objek tampak melalui komputer.
6. *Intelligent computer-aided construction*

   Komputer dapat digunakan sebagai tutor yang dapat melatih dan mengajar.
7. *Game playing*

Soft computing merupakan inovasi baru dalam membangun sistem cerdas yaitu sistem yang memiliki keahlian seperti manusia pada domain tertentu, mampu beradapatasi dan belajar agar dapat bekerja lebih baik jika terjadi perubahan lingkunagan. *Soft computing* mengeksploitasi adanya toleransi terhadap ketidaktepatan, ketidakpastian, dan kebenaran parsial untuk dapat diselesaikan dan dikendalikan dengan mudah agar sesuai dengan realita. Metodologi – metodologi yang digunakan dalam *soft computing* adalah [12] :

1. Sistem fuzzy (mengakomodasi ketidaktepatan) : logika fuzzy.
2. Jaringan syaraf (menggunakan pembelajaran) : jaringan syaraf tiruan.
3. *Probabilistic reasoning* (mengakomodasi ketidakpastian).
4. *Evolutionary computing* (optimasi) : algoritma genetika.

2.4 Pemograman Berorientasi Objek

OOP (*Object Oriented Programming*) atau yang dikenal dengan Pemrograman Berorientasi Objek merupakan paradigma pemrograman yang berorientasikan kepada objek. Semua data dan fungsi di dalam paradigma ini dibungkus ke dalam kelas-kelas atau objek-objek. Model data berorientasi objek dikatakan dapat memberi fleksibilitas yang lebih, kemudahan mengubah program, dan digunakan luas dalam teknik piranti lunak skala besar. Lebih jauh lagi, pendukung OOP mengklaim bahwa OOP lebih mudah dipelajari bagi pemula dibanding dengan pendekatan sebelumnya, dan pendekatan OOP lebih mudah dikembangkan dan dirawat.

Dengan menggunakan OOP maka dalam melakukan pemecahan suatu masalah kita tidak melihat bagaimana cara menyelesaikan suatu masalah tersebut (terstruktur) tetapi objek-objek apa yang dapat melakukan pemecahan masalah tersebut. Sebagai contoh anggap kita memiliki sebuah departemen yang memiliki manager, sekretaris, petugas administrasi data dan lainnya. Misal manager tersebut ingin memperoleh data dari bag administrasi maka manager tersebut tidak harus mengambilnya langsung tetapi dapat menyuruh petugas bag administrasi untuk mengambilnya. Pada kasus tersebut seorang manager tidak harus mengetahui bagaimana cara mengambil data tersebut tetapi manager bisa mendapatkan data tersebut melalui objek petugas adminiistrasi. Jadi untuk

menyelesaikan suatu masalah dengan kolaborasi antar objek-objek yang ada karena setiap objek memiliki deskripsi tugasnya sendiri [10].

2.4.1 Objek (*object*)

Orientasi objek merupakan teknik dalam menyelesaikan masalah yang kerap muncul dalam pengembangan perangkat lunak. Objek adalah entitas yang memiliki atribut, karakter (*behaviour*), dan kadang kala disertai kondisi (*state*). Objek mempresentasikan sesuatu sistem real seperti siswa, sistem kontrol permukaan sayap pesawat, sensor atau mesin. Objek juga mempresentasikan sesuatu dalam bentuk konsep seperti nasabah bank, merek dagang, pernikahan atau sekedar *listing*. Bahkan bisa juga menyatakan visualisasi seperti bentuk huruf, histogram, polygon, garis atau lingkaran. Semuanya memiliki fitur atribut (untuk data), *behavior* (*operation* atau *method*), keadaan (memori), identitas, dan tanggung jawab. Proses menjabarkan sistem nyata menjadi objek dinamakan abstraksi (*abstraction*). Abstraksi mengeleminir aspek yang tidak perlu dalam suatu objek.

2.4.2 Kelas (*class*)

Kelas adalah penggambaran satu set objek yang memiliki atribut dan *behavior* yang sama. Kelas mirip tipe data pada pemograman non objek, tapi lebih komperhensif karena terdapat struktur sekaligus karakteristiknya. Kita dapat membentuk kelas baru yang lebih spesifik dari kelas *general*-nya. Kelas dan objek merupakan jantung dari pemograman berorientasi objek. Untuk menghasilkan program jenis ini sangat penting untuk selalu berpikir dalam konsep objek.

2.4.3 Pembungkusan (*encapsulation*)

Pembungkusan sebagai penggabungan potongan – potongan informasi dan perilaku – perilaku spesifik yang bekerja pada informasi tersebut, kemudian mengemasnya menjadi apa yang disebut sebagai objek. Dalam perbankan kita mengenal objek rekening yang memiliki perilaku misalnya buka, tutup, penarikan, penyimpanan, ubah nama, ubah alamat, dan sebagainya. Akibat perubahan – perubahan pada sistem perbankan yang berkaitan dengan rekening dapat secara

sederhana diimplementasikan satu kali saja pada objek rekening. Keuntungan lainnya adalah membatasi efek – efek perubahan pada sistem. Misalnya. Saat manajemen bank menentukan jika seseorang memiliki rekening pinjaman di bank yang bersangkutan, rekening pinjaman itu harus dapat juga digunakan sebagai saran bagi penarikan rekening.

2.4.4 Pewarisan (*inheritance*) dan Generalisasi / Spesialisasi

Pewarisan adalah konsep dimana metode dan atribut yang ditentukan dalam sebuah objek kelas dapat diwariskan atau digunakan lagi oleh objek kelas lainnya. Generalisasi/spesialisasi merupakan teknik dimana atribut dan perilaku yang umum pada beberapa tipe kelas objek (dinamakan *supertype*). Atribut dan metode kelas objek *supertype* kemudian diwariskan oleh kelas objek tersebut (dinamakan *subtype*).

2.4.5 *Polimorfisme*

Polimorfisme berarti suatu fungsionalitas yang diimplementasikan dengan berbagai cara yang berbeda. Pada terminologi berorientasi objek, ini berarti kita dapat memliki berbagai implementasi untuk sebagian fungsionalitas tertentu. Sebagai contoh, misalkan kita akan mengembangkan sistem berbasis grafis. Saat pengguna mau menggambar sesuatu, misalnya garis atau lingkaran, sistem akan memunculkan perintah gambar. Sistem akan mengenali berbagai bentuk gambar masing – masing dengan perilakunya sendiri – sendiri. Manfaat dari *polimorfisme* adalah kemudahan pemeliharaannya.

2.5 Pengertian dan Sejarah UML (*Unified Modeling Language*)

UML singkatan dari *Unified Modeling Language*yang berarti bahasa pemodelan standar. UML memiliki sintaks dan semantik. Ketika kita membuat model menggunakan konsep UML ada aturan - aturan yang harus diikuti. Bagaimana elemen pada model – model yang kita buat berhubungan satu dengan lainnya harus mengikuti standar yang ada. UML bukan hanya sekedar diagram, tetapi juga menceritakan konteksnya. UML dapat diaplikasikan untuk maksud tertentu, biasannya antara lain [11] :

1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses – proses dan organisasinya.

Blok pembangunan utama UML adalah diagram. Para pengembang sistem berorientasi objek menggunakan bahasa model untuk menggambarkan, membangun, dan mendokumentasikan sistem yang mereka rancang. UML merupakan alat komunikasi yang konsisten dalam mendukung para pengembang sistem saat ini.

Sebelum ada UML pengembang bahasa pemograman berorientasi objek kesulitan untuk menentukan suatu notasi yang cocok. Para pengguna notasi yang berlainan ini saling berebut pengaruh agar notasi yang digunakan menjadi notasi standar. Walaupun dijumpai beberapa notasi yang cukup jelas dan sangat cocok untuk menyelesaikan masalah pembuatan perangkat lunak akan tetapi belum diakui oleh pengembang sistem yang lain.

Pada bulan Oktober 1994, Jim Rumbaugh, penemu notasi *Object Modeling Techinque* (UMT) dan Grady Booch, penemu *Booch Method* bersama – sama menyamakan notasi mereka. Ditahun yang sama, Ivar Jacobson (penemu *Objectory Method*) ikut bergabung. Ketiga orang tersebut sering disebut "*three amigos*", bersama – sama membangun notasi standar oop untuk *software* Rational IBM. Jim Rumbaugh banyak memberikan masukan dalam hal pembuatan notasi dan analisa UML. Grady Booch merancang secara detil kapabilitas UML sedangkan Ivar Jacobson berusaha membuat UML cocok dengan model bisnis dan mencoba mengembangkan *usecasediagram* lebih lanjut.

Ketiga pengembang UML tersebut mengalami kesulitan akibat kompleksnya permasalahan oop yang ada namun mereka dibantu oleh *Object Management Group* (OMG). OMG adalah gabungan dari kurang lebih 800 perusahaan pengembang perangkat lunak berorietasi objek. Setelah perdebatan sengit cukup lama, konsensus tentang notasi berhasil dicapai dengan sukses pada bulan November 1997 berkat bantuan OMG.

Sejak tahun 1997, divisi *Revision Task Force* (RTF) milik OMG beberapa kali melakukan revisi UML. Revisi dimaksudkan untuk memperkuat konsistensi notasi, meningkatkan kekompakan antara *user* dan pengembang perangkat lunak. Akan tetapi UML terpaksa mengikuti perkembangan *software – software* berbasis objek yang ada dari sisi pendekatan komponen dan kemampuan *toolssoftware* tersebut. Setelah dilakukan perubahan secara sistematik akhirnya dihasilkan UML 2.0 pada tahun 2003.

2.5.1 Diagram – Diagram UML

Model diagram UML dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain :

1. Diagram kelas

   Diagram ini memperlihatkan himpunan kelas, antarmuka, koalborasi, serta relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas – kelas aktif.
2. Diagram paket

   Diagram ini memperlihatkan kumpulan kelas – kelas merupakan bagian dari diagram komponen dan bersifat statis.
3. Diagram *use case*

   Diagram ini memperlihatkan himpunan *use case* dan aktor – aktor. Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna. Diagram ini bersifat statis.
4. Diagram interaksi dan *sequence*(urutan)

   Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu. Diagram ini bersifat dinamis.
5. Diagram komunikasi

   Diagram ini sebagai pengganti diagram kolaborasi UML 1.4 yang menekankan organisasi struktural dari objek – objek yang menerima serta mengirim pesan. Diagram ini bersifat dinamis.

6. Diagram *statechart.*

   Diagram status memperlihatkan keadaan – keadaan pada sistem, memuat status, transisi, kejadian serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka, kelas, kolaborasi, dan terutama penting pada pemodelan sistem – sisem yang reaktif.

7. Diagram akitfitas

   Diagram aktifitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktifitas ke aktifitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi – fungsi suatu sistem dan member tekanan pada aliran kendali antar objek. Diagram ini bersifat dinamis.

8. Diagram komponen

   Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem atau perangkat lunak pada komponen – komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secata tipikal dipetakan kedalam satu atau lebih kelas, antarmuka serta kolaborasi. Diagram ini bersifat statis.

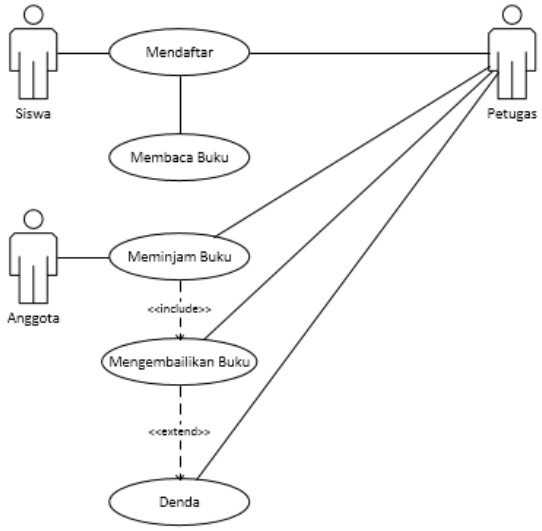
9. Diagram *deployment*

   Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan. Memuat simpul – simpul beserta komponen – komponen yang ada di dalamnya. Diagram deployment berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin.

2.5.2 Diagram *Use Case*

Diagram *use case* bersama dengan narasi *use case* dan skenario mendefinisikan tujuan suatu sistem atau pengklasifikasi lain seperti *enterprise*, subsistem atau komponen. Konsep ini diperkenalkan oleh Ivar Jacobson bersama organisasinya dalam bentuk metodologi yang mereka namakan *Object Oriented Software Engineering* (OOSE). Tujuan dibentuknya metode ini adalah agar dihasilkan fokus yang baik pada pengembangan dan tujuan utama tanpa terpengaruh oleh implementasi praktis.

Use case mengekspresikan apa yang *user* harapkan terhadapa sistem. Narasi *use case* menjelaskan secara detail bagaiman *user* berinteraksi dengan sistem saat mengakses *use case*. Skenario memecah penjelasan narasi untuk menyediakan penjelasan detail terhadap segala kemungkinan yang terjadi pada *use case* dan apa respon sistem. Gambar dibawah ini merupakan salah satu contoh bentuk diagram *use case*[10].



Gambar 2.7 Contoh *Use Case*

Komponen pembentuk diagram *use case* adalah :

1. Aktor

   Menggambarkan pihak – pihak yang berperan dalam sistem. Prioritaskan calon aktor berdasarkan pengaruhnya terhadap sistem yang kita rancang.



Gambar 2.8 Contoh Aktor

2. *Use case*

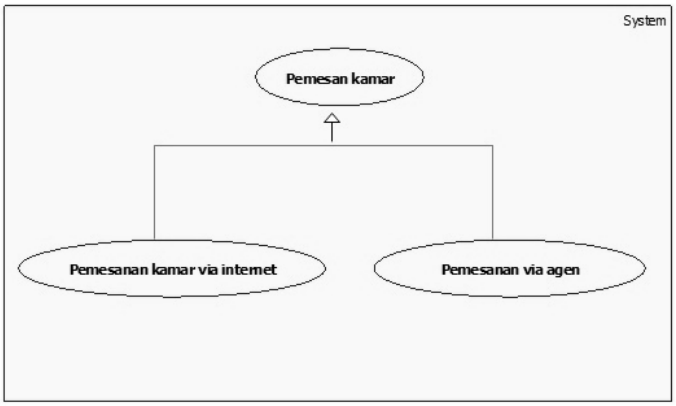
   Aktivitas yang disiapkan oleh sistem. *Usecase* menggambarkan fungsi tertentu dalam suatu sistem berupa komponen, kejadian atau kelas. *Usecase* sangat menentukan karakteristik sistem yang kita buat.

3. Hubungan (*relation*)

   Aktor mana saja yang terlibat dalam *usecase* ini. Relasi digambarkan sebagai sebuah garis antara dua simbol. Pemaknaan relasi berbeda – beda tergantung bagaimana garis tersebut digambar dan tipe simmbol apa yang digunakan untuk menghubungkan garis tersebut. Relasi yang digunakan adalah generalisasi, inklusi, dan ekstensi.

4. Generalisasi

   Generalisasi pada aktor dan *usecase* dimaksudkan untuk menyederhanakan model dengan cara menarik keluar sifat – sifat pada aktor maupun use case uang sejenis.



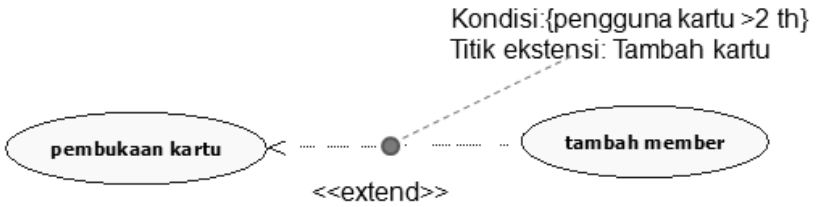
Gambar 2.9 Contoh Generalisasi

5. Ekstensi

   Ekstensi pada *use case* terdiri dari langkah yang diekstraksi dari *use case* yang lebih kompleks untuk menyerderhanakan masalah orisinal dan memperluas fungsinya. Hubungan antara ekstensi *use case* dan *use case* yang diperluas disebut *extend relationship* diberi simbol "<<*extend*>>" dan hubungannya berupa garis putus – putus berpanah terbuka.



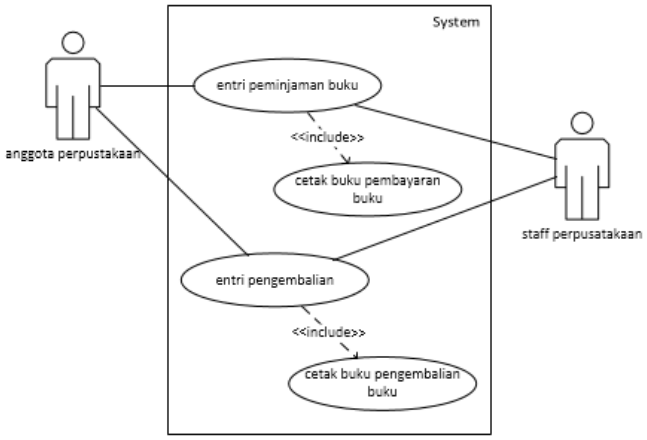
Gambar 2.10 Contoh Ekstensi



Gambar 2.11 Contoh Titik Ekstensi

6. Inklusi

   Use case dasar yang akan diinklusi tidak lengkap, berbeda dengan *usecase* dasar yang akan diekstensi. Sehingga *usecase* inklusi bukan merupakan *use case optional*dan tidak boleh tidak dijalankan. Simbol hubungan inklusi adalah garis putus – putus dengan anak panah terbuka dan diberi keterangan "<<*include*>>".



Gambar 2.12 Contoh Inklusi

Baik inklusi maupun ekstensi semuanya bermaksud memperluas perilaku *use case* dasarnya. Untuk lebih jelasnya berikut perbedaan antara inklusi dan ekstensi.

Tabel 2.11 Perbedaan Inklusi dan Ekstensi

Inklusi	***Ekstensi***
use case terpanggil (*include use case*) selalu diperlukan oleh *use case* dasar	*use case* ekstensi tidak selalu dibutuhkan oleh *use case* dasar
yang memutuskan kapan dipanggilnya *use case included* adalah *use case* dasar	yang memutuskan kapan dipanggilnya *use case* ekstensi adalah *use case* ekstensi itu sendiri

panah hubungan dari *use case* dasar ke *use case included*	panah hubungan dari *use case* ekstensi ke *use case* dasar

2.5.3 Diagram Kelas

Diagram kelas adalah inti dari proses pemodelan objek. Baik *forward engineering* maupun *reverse engineering* memanfaatkan diagram ini. *Forward engineering* adalah proses perubahan model menjadi kode program sedangkan reverse engineering sebaliknya kode program menjadi model. Atribut dan operasi yang menggambarkan perilaku suatu kelas seta perluasannya seperti *stereotypes, tagged values,* dan batasan merupakan fitur – fitur suatu kelas. Kemudian dibahas juga hubungan antara satu kelas dengan kelas lainnya beserta sifat – sifat hubungan tersebut seperti penampakan (*visibility*), *multiplicity*, kemudahan berubah (*change ability*), nilai pokok (*default values*), dan property (*properties*). Adapun pemberian nama untuk diagram kelas yaitu [10]:

1. Dimulai dengan huruf besar.
2. Ditengah pada kompartemen bagian atas.
3. Ditulis dengan cetak tebal.
4. Ditulis dengan cetak miring jika kelas tersebut abstrak.

Atribut atau rincian suatu kelas misalnya warna mobil, jumlah sisi suatu bentuk, dan sebagainya dinamakan atribut. Atribut bisa sederhana (*integer, floating point,* dan sebagainya) bisa juga kompleks. Atribut mendefinisikan tipe informasi suatu kelas dengan standar sintaks :

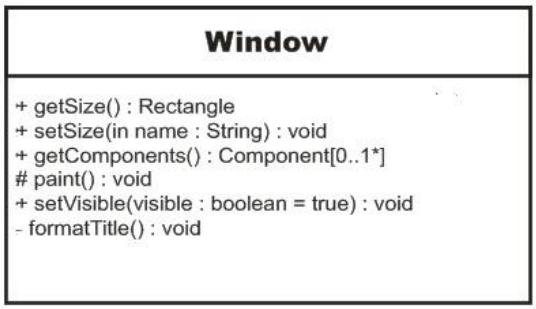
[visibility][/]name [:type] [multiplicity] [=default] [{property-string}]

Operasi sebagai fitur dari kelas yang menspesifikasikan bagaiman perilaku dijalankan. Misalnya menggambar segiempat, menghitung jumlah item suatu daftar dan sebagainya. UML membedakan dengan jelas antara operasi dan *method*. Jika operasi mendefinisikan suatu perilaku dijalankan, *method* adalah pelaksanaan dari operasi. *Method* berupa prosedur atau algoritma untuk menjalankan operasi. Diagram kelas hanya mencantumkan operasi saja sedangkan *method* tidak. Operasi diletakkan dalam kompartemen terpisah dengan sintaks :

visibility name (paramaters) : return-type {properties}

paremeter ditulis sebagai berikut :

direction parameter_name : type [multiplicity] = default_value {properties}



Gambar 2.13 Contoh Operasi

Tanpa adanya hubungan dengan kelas lain, kita tidak mampu melihat mekanisme suatu sistem. Tiap hubungan memiliki tipe yang berbeda sesuai dengan kebutuhan dengan harapan saat kita memodelkan suatu sistem, pembaca akan memahami hubungan yang kita buat. Ada beberapa relasi dalam diagram kelas ini yaitu :

1. Ketergantungan

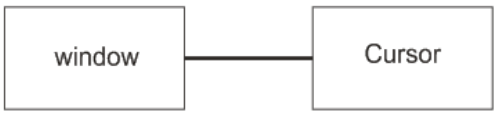
   Ketergantungan antar kelas bermakna suatu kelas menggunakan atau memiliki pengetahuan terhadap kelas lainnya. Hubungan ketergantungan hanya sesaat (*transient*) dan tidak ada batas waktu yang jelas.



Gambar 2.14 Contoh Ketergantung

2. Asosiasi

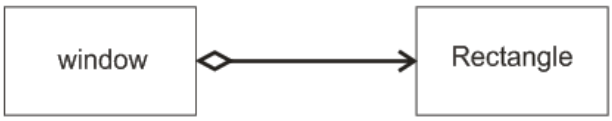
   Hubungan pada asosiasi lebih kuat dari hubungan ketergantungan dalam arti suatu kelas tetap berhubungan dengan kelas seterusnya. Hubungan dapat diputus tanpa mengganggu kelas lainnya.



Gambar 2.15 Contoh Asosiasi

3. Agregasi

   Agregasi adalah versi kuat dari asosiasi. Agregasi mengimplikasikan kempemilikan suatu kelas. Hubungan agregasi dinyatakan dengan simbol "*diamond*" pada kelas pemilik (*owning class*) dan garis utuh berpanah ke kelas yang dimilki.



Gambar 2.16 Contoh Agregasi

4. Komposisi

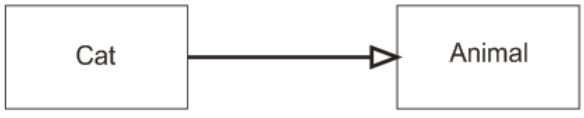
   Komposisi merupakan hubungan yang paling kuat antar kelas. Komposisi digunakan untuk mengambil seluruh bagian hubungan. Hanya boleh ada satu komposisi dalam satu waktu. Secara notasi, komposisi digambarkan dnegan belah ketupat tebal "*filled diamond*" dekat kelas pemilik dan garis tebal berpanah menuju kelas yang dimiliki.



Gambar 2.17 Contoh Komposisi

5. Generalisasi

   Generalisasi berarti target dari hubungannya ke kelas yang lebih general. Sebagai contoh jika kita memiliki kelas bernama *cat* dan *dog*, kita dapat membuat generalisasi dari keduanya dengan nama*animal*. Generalisasi dimulai dari kelas yang lebih spesifik menuju kelas yang lebih umum.

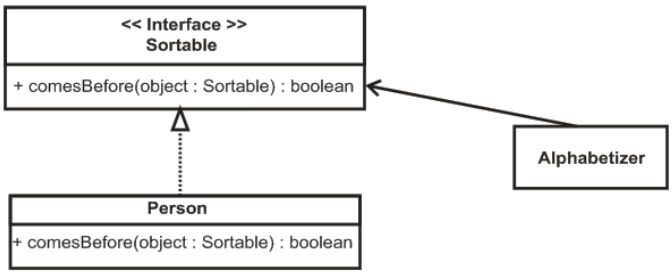


Gambar 2.18 Contoh Generalisasi

6. Antarmuka

   Suatu antarmuka mendefinisikan sekumpulan perilaku yang harus dimiliki oleh kelas implementator antamuka. Operasi – operasi pada kelas tidak memiliki

metode. Atribut dalam kelas antarmuka berbentuk abstrak. Jika kelas implementaor antarmuka bisa menyediakan akses dalam bentuk informasi yang sama dengan operasinya, maka tidak perlu mendeklarasikan atribut lagi.

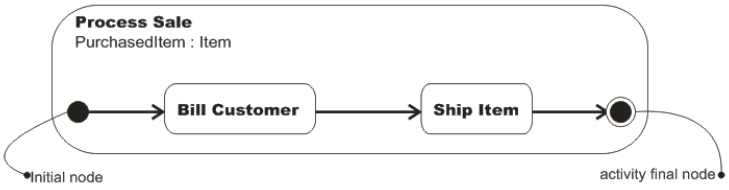


Gambar 2.19 Antarmuka

2.5.4 Diagram Aktivitas

Diagram aktivitas lebih memfokuskan diri pada eksekusi dan alur sistem daripada bagaimana sistem itu dirakit. Diagram ini tidak hanya memodelkan *software* melainkan memodelkan bisnis juga. Diagram aktivitas menunjukkan aktivitas sistem dalam bentuk kumpulan aksi – aksi.

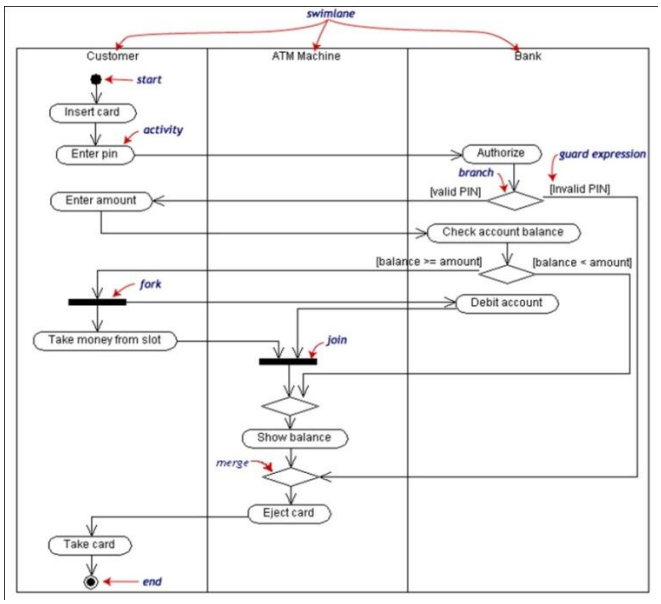
Aktivitas merupakan kumpulan aksi – aksi. Aksi – aksi melakukan langkah sekali saja tidak boleh dipecah menjadi beberapa langkah lagi. Contoh aksi yaitu fungsi matematika, pemanggilan perilaku, dan pemrosesan data. Aktivitas digambarkan dengan persegi panjang tumpul. Namanya ditulis dikiri atas. Parameter yang terlibat dalam aktivitas ditulis dibawahnya.



Gambar 2.20 Diagram aktivitas dengan titik awal dan titik akhir

Untuk menunjukkan aliran aktivitas dengan menggabungkan aksi – aksi bersama menggunakan tepian aktivitas. Tepian aktivitas terdiri dari tepian kontrol, aliran kontrol (*control flows*), dan aliran objek (*object flows*). Aliran kontrol secara *eksplisit* memodelkan kontrol dari satu aksi ke aksi berikutnya. UML menyediakan tepian aktivitas yang berupa data saja, dinamakan aliran objek. Secara konseptual, informasi yang bergerak sepanjang tepian disebut token.

Titik aktivitas terdiri dari titik parameter, titik objek, pin, dan titik kontrol. Titik objek digunakan untuk mempresentasikan data kompleks yang melewati diagram aktivitas. Guna mempesingkat notasi, UML mendefinisikan notasi khusus untuk titik objek yaitu pin sebagai masukan dan keluaran aksi. Sebagai tambahan terhadap suatu aksi , aktivitas mungkin memasukan titik lain unuk mempresentasikan pengambilan keputusan, persetujuna atau sinkoronisasi yang disebut titik kontrol. Titik control terdiri dari titik inisial, titik keputusan, dan penggabungan serta titik *fork-join*. Titik inisial sebagai inisialisasi, titik keputusan dan penggabungan berbeda dengan *fork-join* karena data keluar tidak dilakukan sinkronisasi [10].



Gambar 2.21 Diagram aktivitas dengan *swimlane*

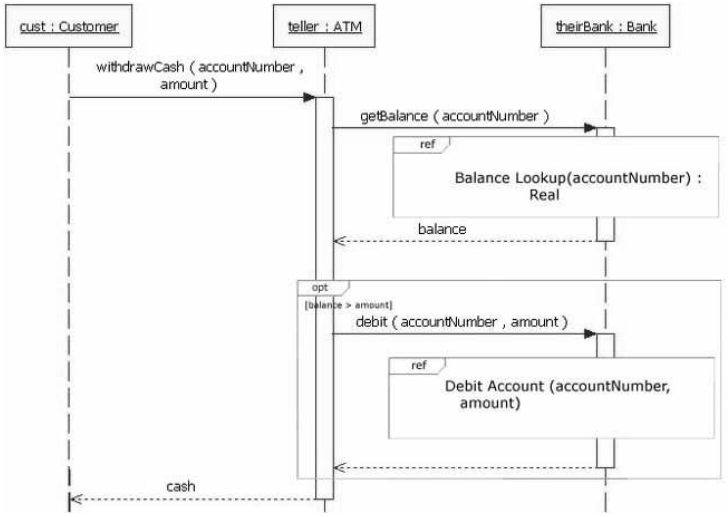
2.5.5 Diagram *Sequence*

Sequence diagram menggambarkan interaksi antar objek di dalam dan disekitar sistem (termasuk pengunna, *display,* dansebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri dari dimensi vertikal (waktu) dan dimensi horizontal (objek – objek yang terkait).

Sequence diagram biasa digunakan untuk menggambarkan scenario atau rangkaian langkah – langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan *output* tertentu. Diawali dari apa yang men-*trigger* aktivitas

tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.

Masing – masing objek, termasuk aktor memiliki *lifeline* vertikal. *Message* digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari *class. Activation bar* menunjukkan lamanya eksekusi sebuah proses biasanya diawali dengan diterimanya sebuah *message.* Untuk objek – objek yang memiliki sifat khusus, standar UML mendefinisikan *icon* khusus untuk objek *boundary*, *controller*, dan *persistent entity*[10].



Gambar 2.22 Diagram *Sequence*

2.6 Metode Pengujian Sistem

Metode pengujian sistem untuk mengetahui efektifitas dari *software* yang digunakan selain memberikan kesempatan kepada pengguna untuk mengoperasikan dan melakukan pengecekan terhadap laporan yang dihasilkan melalui *software*. Metode pengujian sistem terdiri dari Pengujian *White-box* dan Pengujian *Black-box* [1].

2.6.1 Pengujian *White Box*

Pengujian *white-box* dilakukan untuk menguji prosedur-prosedur yang ada. Lintasan lojik yang dilalui oleh setiap bagian prosedur diuji dengan memberikan kondisi/loop spesifik. Pengujian *white-box* menjamin pengujian terhadap semua lintasan yang tidak bergantungan minimal satu kali,

mencobasemua keputusan lojik dari sisi „*true*‟ dan „*false*‟, eksekusi semua *loop* dalam batasan kondisi dan batasan operasionalnya dan pengujian validasi data internal. Adapun konsep pengujian basis path sebagai berikut :

1. Merupakan bagian dari pengujian *whitebox* dalam hal pengujian prosedur – prosedur
2. Mempergunakan notasi aliran *graph* (*node*, *link* untuk merepresentasikan *sequence, if, while, until,* dan sebagainya).
3. Konsep kompleksitas *cyclomatic* antara lain cara perhitungan daerah tertutup pada *graph* planar dimana dapat manghubungkan batas atas jumlah pengujian yang harus direncanakan dan dieksekusi untuk menjamin pengujian seluruh statement program.
4. Memunculkan kasus – kasus yang akan diuji dengan membuat daftar lintasan kasus pengujian berdasarkan kompleksitas dan *cyclomatic* yagn didapat.
5. Membuat alat bantu*graph* matriks yang membantu pengawasan pengujian.

2.6.2 Pengujian *Black Box*

Pengujian yang dilakukan untuk antarmuka perangkat lunak, pengujian ini dilakukan untuk memperlihatkan bahwa fungsi – fungsi bekerja dengan baik dalam arti masukan yang diterima dengan benar dan keluaran yang dihasilkan benar – benar tepat, pengintegrasian dari eksternal data berjalan dengan baik.

Metode pengujian *black box* memfokuskan pada *requirement* fungsi dari perangkat lunak, pengujian ini merupakan komplenetari dari pengujian *white box.* Pengujian *white box* dilakukan terlebih dahulu pada proses pengujian, sedangkan pengujian *black box* dilakukan tahap akhir dari pengujian perangkat lunak. Proses yang terdapat dalam proses pengujian *black box* antara lain sebagai berikut :

1. Pembagian kelas data untuk pengujian setiap kasus yang muncul pada pengujian *white box.*
2. Analisis batasan nilai yang berlaku untuk setiap data.

2.7. *Tools* Yang Digunakan

Tools yang akan digunakan untuk membangun aplikasi game edukasi bahasa Arab (mencocokan kosakata) adalah Construct 2 dan Adobe Illustrator.

2.7.1. Construct 2

Construct 2 adalah aplikasi pembangun game berbasis HTML 5 yang dibangun oleh Scirra Ltd. Aplikasi ini dibuat khususnya untuk game 2 dimensi dan siapapun bisa membangun game tanpa koding yang dibutuhkan hanya pemahaman logika. Tidak perlu membutuhkan memori yang kompleks dan bahasa pemograman, cukup dengan fokus pada mendesain game yang diinginkan.

Construct 2 hanya membutuhkan action dan event untuk mengolah game tentunya dengan fungsi – fungsi yang sudah disediakan oleh aplikasi ini dan pengaturan pada properties bar. Aplikasi ini pun menyediakan behavior atau sifat object yang bisa kita atur apakah lari, melompat, tembak, dan lain-lain. Construct 2 membantu untuk berpikir secara logika dan mengerti konsep dari bahasa pemograman.

Pada construct 2 juga memberikan fitur untuk melakukan eksport ke beberapa platform tanpa harus mengatur lagi. Eksport bisa dilakukan pada desktop, Mac, dan linux menggunakan Node-Webkit wrapper dan juga pada aplikasi android dengan menggunakan coconJS, Mobi dan phonegap.

Construct 2 adalah aplikasi game engine yang menjadikan pembuatan sebuah game dengan mudah. Hanya dengan beberapa menit atau jam bisa membuat prototype game yang bisa dibagikan dengan teman untuk dimainkan. Bisa melakukan test untuk mengetahui apakah ide aplikasi game yang dibuat menyenangkan atau tidak. Membuat game seperti bagian dari suatu mahakarya hanya drag dan drop object, tambahkan pula behaviours pada obeject dan buat semuanya lebih hidup dengan events [16].

2.7.2. Adobe Illustrator

Adobe illustrator pertama kali dikembangkan oleh Adobe Inc pada bulan Desember 1986 sebagai komersialisasi di rumah adobe huruf pengembangan perangkat lunak dan PostScript format file. Adobe illustrator adalah produk pendamping dari adobe photoshop dimana photoshop diarahakan pada foto digital manipulasi dan fotorealistik gaya ilustrasi komputer, sementara illustrator memberikan hasil dalam typesetting dan logo daerah grafik desain.

Pertama kali dirilis Adobe Illustrator, mereka pasti keberhasilan program. Perusahaan tidak memiliki persentase besar dari pasar, dan alat-satunya perangkat lunak yang mampu menantang Adobe Illustrator adalah LaserWriter, alat yang diproduksi oleh Apple. Adobe Illustrator adalah alat yang ampuh yang memiliki kurva belajar rendah. Meskipun ini, terkenal dengan tingkat presisi. Banyak orang telah menemukan Adobe Illustrator untuk menjadi alternatif yang berguna untuk alat desain kompleks seperti AutoCad. Selain ini,Adobe Illustrator tempat penekanan pada lukisan, sesuatu yang hilang dari AutoCad ketika pertama kali diperkenalkan.

Alat utama yang digunakan di Illustrator adalah kurva Bezier.. Sebuah kurva dapat dibuat, dan lingkaran atau busur dapat dibuat dari itu. Illustrator juga mendukung file PostScript juga. Dukungan untuk PostScript diperbolehkan pengembang untuk menciptakan program yang secara khusus dirancang untuk membuat file Illustrator. Ketika Adobe Illustrator 1.1 diperkenalkan, itu menampilkan video yang menunjukkan John Warnock pengajaran penggunaan program. Satu-satunya fitur yang Illustrator kekurangan untuk jangka waktu yang panjang adalah campuran atau gradien mengisi. Bahkan, fitur ini tidak akan sering terlihat sampai rilis dari Adobe Illustrator 5. Sebagai sebuah perusahaan, Adobe dikenal karena mengambil risiko dengan rilis Illustrator, terutama ketika datang ke antarmuka.

Sementara Illustrator awalnya dirilis untuk Macintosh, ia dirancang untuk platform lainnya pada awal 1990-an.Beberapa platform yang didukung adalah Adobe Illustrator Sun Solaris, SGI IRIX, dan MS Windows. Namun, perangkat lunak tidak melakukan dengan baik pada banyak platform, dan dukungan bagi mereka dijatuhkan. . Program menggambar yang ideal untuk OS Windows CorelDraw. Beberapa perubahan besar dibuat ke versi 6 Illustrator yang diperkenalkan pada tahun 1996 [17].