# BAB 2
# TINJAUAN PUSTAKA

## 2.1 Tinjauan Sekolah

Pada sub bab ini membahas peninjauan terhadap tempat penelitian yaitu SMK Negeri 1 Majalengka.

### 2.1.1 Sejarah Sekolah

Sekolah Menengah Kejuruan (SMK) Negeri 1 Majalengka yang berdiri dengan megah di atas tanah seluas 40.000 m2 dengan status tanah milik sendiri, terletak di Jalan Raya Tonjong-Pinangraja No.55, Majalengka. Dengan luas bangunan 6.565 m2, keberadaan SMK Negeri 1 Majalengka merupakan salah satu sekolah kejuruan yang mendapat perhatian dari Pemerintah Daerah sebagai sekolah kejuruan negeri yang ada di wilayah Kabupaten Majalengka dengan fasilitas belajar yang cukup lengkap, wajar kalau keberadaannya merupakan sekolah favorit bagi kalangan siswa di wilayah Kabupaten Majalengka.

Cikal bakal berdirinya SMK Negeri 1 Majalengka, tidak terlepas dari keberadaan sekolah sebelumnya. Pada tahun 1950 Pemerintah mendirikan STP Negeri, jenjang pendidikan ini setingkat SLTP dengan lamanya pendidikan 2 tahun. Selang beberapa tahun berikutnya, untuk kesejajaran SLTP dirubah menjadi STN (Sekolah Teknik Negeri), dengan Kepala Sekolah dijabat oleh Bapak R.Tamyid pada tahun 1960. Kemerosotan ekonomi pada saat itu sangat mempengaruhi keberadaan sekolah dengan banyaknya guru-guru STN yang pindah dan alih profesi. Dengan kebijakan Kepala Sekolah Inspeksi Pendidikan Teknik yang berkedudukan di Bandung, maka status STN berubah menjadi STM pada tahun 1965. Karena suasana politik dengan lahirnya orde baru, sekolah kejuruan terpaksa dibubarkan. Tahun 1968 mencoba kembali didirikannya STM dan berjalan dengan pesatnya yang berlindung dibawah Yayasan Pemda dan dikenal dengan nama STM Pemda. Pada tahun 1974 sesuai dengan pergerakan waktu, STM Pemda dijabat oleh Bapak H.Rusdi, B.Sc.

Pada saat kepemimpinan H.Rusdi, B.Sc, status STM Pemda Majalengka berubah menjadi STM Negeri Majalengka sesuai SK Menteri Pendidikan dan Kebudayaan RI No.0267/0/1980 Tanggal 1 Juli 1980. Pada tahun 1986 kepala STM Negeri Majalengka diganti oleh Bapak Amir Sugandi, BA. Pada masa pengelolaan Bapak Amir Sugandi, BA, mendapat pengembangan Sekolah Kejuruan melalui dana Loan ADB, sehingga memiliki kampus baru di Jalan Raya Tonjong-Pinangraja No.55 Kelurahan Cicenang Kecamatan Cigasong Kabupaten Majalengka seluas 4 Ha dengan jarak dari sekolah lama kurang lebih 2,5 km ke arah Jatiwangi. Sekolah baru ini pada tahun 1995/1996 dikembangkan menjadi 3 jurusan, yaitu Mekanik Umum, Listrik Pemakaian dan Bangunan Gedung

Pada tahun 2008/2009 sampai saat ini, SMK Negeri 1 Majalengka terus berkembang dengan bertambahnya pilihan jurusan seperti: Teknik gambar bangunan, Teknik instalasi tenaga listrik, Teknik pemesinan, Teknik kendaraan ringan, Teknik sepeda motor, Teknik komputer dan jaringan dan Rekayasa perangkat lunak.

### 2.1.2 Visi, Misi dan Motto

**1. Visi**

Visi SMK Negeri 1 Majalengka adalah Menjadikan Lembaga Pendidikan pelatihan kejuruan bertaraf internasional dan berwawasan lingkungan yang menghasilkan tamatan profesional, mampu berwirausaha, beriman dan bertaqwa.

**2. Misi**

Misi yang dimiliki SMK Negeri 1 Majalengka adalah sebagai berikut:

1. Melaksanakan sistem manajemen mutu (SMM) berbasis ICT dan berkelanjutan.
2. Meningkatkan kualitas tenaga pendidik dan kependidikan yang memenuhi kualifikasi dan kompetensi standar.
3. Meningkatkan fasilitas dan lingkungan belajar yang nyaman dan memenuhi standar kualitas dan kuantitas.

4. Mengembangkan kurikulum, metodologi pembelajaran dan sistem pernilaian berbasis kompetensi.
5. Menyelenggarakan pembelajaran sistem CBT (*Competency-Based Training*) dan PBE (*Production-Based Education*) menggunakan bilingual dengan pendekatan ICT.
6. Membangun kemitraan dengan lembaga yang relevan baik dalam maupun luar negeri.
7. Menyelenggarakan kegiatan ekstrakurikuler agar peserta didik mampu mengembangkan kecakapan hidup (*Life Skill*) dan berakhlak mulia.

**3. Motto**

Motto SMK Negeri 1 Majalengka yaitu Pelayanan prima, unggul dalam mutu, tinggi dalam prestasi.

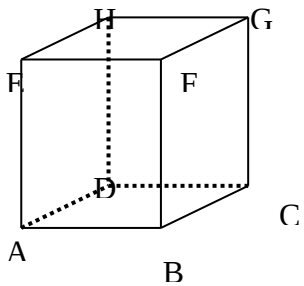
## 2.2 Landasan Teori

### 2.2.1 Bangun Ruang

Materi tentang bangun ruang sudah pernah dipelajari di SMP, di antaranya : Kubus, Balok, Prisma, Limas, Tabung, Kerucut, dan Bola.

#### 2.2.1.1 Kubus

Kubus adalah bangun ruang yang dibatasi enam sisi yang berbentuk persegi yang kongruen. Nama lain dari kubus adalah heksader (bidang enam beraturan).



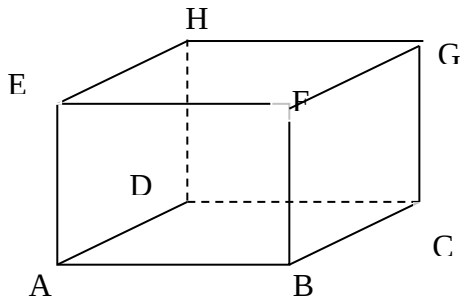

**Gambar 2-1 Bangun Ruang Kubus**

Unsur-unsurnya :

a. Mempunyai 6 sisi berbentuk persegi dan kongruen
b. Mempunyai 12 rusuk yang kongruen
c. Mempunyai 12 diagonal sisi sama panjang
d. Mempunyai 4 diagonal ruang yang sama panjang
e. Mempunyai 6 bidang diagonal berbentuk persegi panjang

#### 2.2.1.2 Balok

Balok adalah bangun ruang yang dibatasi oleh enam bidang datar yang berbentuk persegi panjang dengan tiga pasang sisi yang saling sejajar. Nama lain dari balok adalah prisma siku-siku.




**Gambar 2-2 Bangun Ruang Balok**

Unsur-unsurnya :

a. Mempunyai 12 rusuk yang terbagi menjadi 3 kelompok yang sejajar dan sama panjang, yaitu:
    1. AB, CD, EF, GH
    2. AE, BF, CG, DH
    3. AD, BC, FG, EH
b. Mempunyai 6 bidang sisi yang sepasang-sepasang kongruen berbentuk persegi panjang, yaitu:
    1. ABCD kongruen dengan EFGH (ditulis ABCD EFGH)
    2. ABFE kongruen dengan DCGH
    3. ADHE kongruen dengan BCGF
c. Mempunyai 12 diagonal sisi, yaitu:
    AF, BE, BD, AC, BG, CF, CH, DG, EG, FH, AH, DE

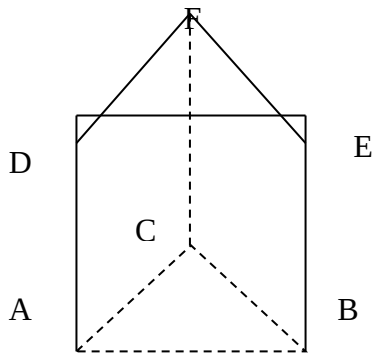
d. Mempunyai 4 diagonal ruang, yaitu: AG, CE, BH, DF
e. Mempunyai 6 bidang diagonal yang kongruen berbentuk persegi panjang, yaitu:
   ACGE, BDHF, ABGH, CDEF, BCEH, AFGD

### 2.2.1.3 Prisma

Macam-macam bentuk Prisma :

a. Prisma segitiga = alasnya berbentuk segitiga
b. Prisma segiempat = alasnya berbentuk segiempat
c. Prisma segilima = alasnya berbentuk segilima, dst.

Sebagai contoh prisma segitiga.



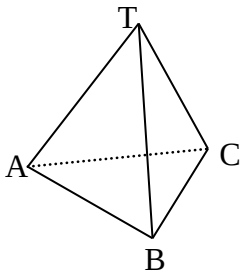

**Gambar 2-3 Bangun Ruang Prisma Segitiga**

Unsur-unsurnya :

a. Bidang sejajar adalah bidang ABC (alas) dengan bidang DEF (atas)
b. Bidang tegaknya ABED, BCFE, ACED
c. Rusuknya :
   1. Rusuk alas = AB, BC, CA
   2. Rusuk atas = DE, EF, FD
   3. Rusuk tegak = AD, BE, CF

### 2.2.1.4 Limas

Limas adalah bangun ruang yang dibatasi oleh segi-n (sebagai bidang alas) dan bidang-bidang yang berbentuk segitiga yang alasnya adalah sisi segi-n dan puncaknya berimpit.



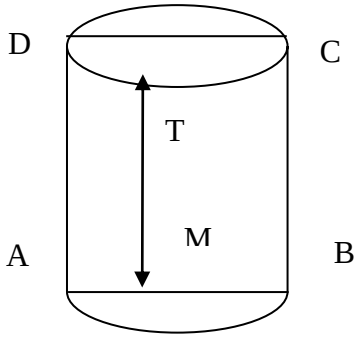

**Gambar 2-4 Bangun Ruang Limas**

Unsur-unsurnya :

a. Rusuk alas : AB, BC, CA
b. Rusuk tegak : TA, TB, TC
c. Sisi tegak : TAB, TBC, TCA
d. Sisi alas : ABC

### 2.2.1.5 Tabung

Tabung adalah prisma tegak beraturan yang bidang alasnya berupa segi-n beraturan dengan n tak terhingga (berupa lingkaran).



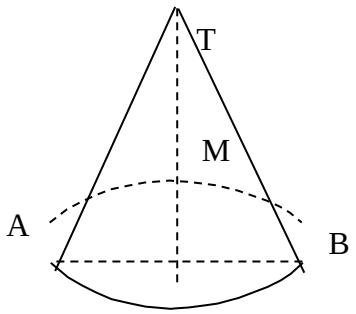

**Gambar 2-5 Bangun Ruang Tabung**

Unsur-unsurnya :

a. Mempunyai 3 bidang sisi, yaitu 2 bidang berbentuk lingkaran dan 1 bidang lengkung.
b. Garis TM adalah tinggi tabung
c. Setiap garis pada bidang lengkung yang sejajar dengan TM disebut garis pelukis.

### 2.2.1.6 Kerucut

Kerucut adalah limas beraturan yang bidang alasnya segi-n beraturan dengan n tak terhingga (alasnya berbentuk lingkaran).



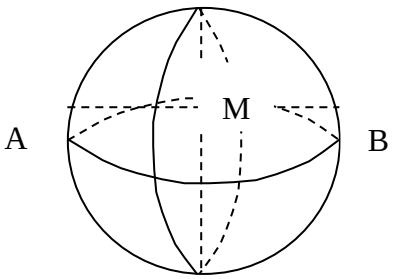

**Gambar 2-6 Bangun Ruang Kerucut**

Unsur-unsurnya :

a. TA dan TB merupakan contoh garis pelukis
b. Titik T adalah puncak kerucut
c. TM adalah tinggi kerucut

### 2.2.1.7 Bola

Bola adalah bangun ruang tiga dimensi yang hanya memiliki satu sisi dan tidak memiliki rusuk maupun titik sudut. Sisi pada bola disebut juga permukaan bola atau kulit bola atau bidang bola.




**Gambar 2-7 Bangun Ruang Bola**

Pusat bola adalah M, AM = MB = r = jari-jari bola dan AB adalah diameter bola.

### 2.2.2 *Augmented Reality*

### 2.2.2.1 Pengertian *Augmented Reality*

*Augmented Reality* (AR) adalah suatu teknologi yang menggabungkan benda maya dua dimensi (2D) ataupun tiga dimensi (3D) kedalam sebuah

lingkungan nyata, lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata. Sistem ini lebih dekat dengan lingkungan nyata. Sistem ini berbeda dengan *virtual reality* (*VR)* yang sepenuhnya merupakan virtual *environment. Augmented Reality* membuat penggunanya dapat berinteraksi secara real-time dengan sistem. Menurut T. Azuma (1997) mendefinisikan *augmented relity* sebagai penggabungan benda nyata dan maya di lingkungan nyata, berjalan secara interaktif dalam waktu nyata, dan terdapat integrasi dalam tiga dimensi, yaitu benda maya terintegrasi dalam dunia nyata [7].

#### 2.2.2.2 Perkembangan *Augmented Reality*

*Augmented reality* pertama kali dimulai pada tahun 1957-1962, Morton Heilig merupakan penemu dan seorang sinematografer yang menciptakan dan mematenkan sebuah simulator yang disebut sensorama dengan visual, getaran, dan bau. Pada tahun 1996, Ivan Suntherland menemukan head-mounted display yang dia claim adalah jendela ke dunia virtual. Tahun 1975 seorang ilmuan bernama Myron Krueger menemukan *Videoplace* yang memungkinkan pengguna dapat berinteraksi dengan objek virtual untuk pertama kalinya. Tahun 1989, Jaron Lanier memperkenalkan *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 Amstrong Labs dan menunjukan manfaatnya pada manusia, dan pada tahun 1992 Steven Fainer, Blair Maclntyre dan Doree Seligmann, merka 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 *Internasional Symposium on Wearable Komputers*.

Pada tahun 2008, Wikitude AR *Travel Guide* memperkenalkan Android Gl *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 I-Phone 3GS.

#### 2.2.2.3 Pengaplikasian Augmented Reality

Seiring berjalannya waktu, *Augmented Reality* berkembang sangat pesat sehingga memungkinkan pengembangan aplikasi ini di berbagai bidang. Contoh, sebagai berikut :

1. Navigasi Telepon Genggam

   Dalam kurun waktu 1 tahun terakhir ini, telah banyak integrasi Augmented reality yang dimanfaatkan pada telepon genggam. Saat ini ada 3 Sistem Operasi telepon genggam besar yang secara langsung memberikan dukungan terhadap teknologi *Augmented reality* melalui antarmuka pemrograman aplikasinya masing-masing. Untuk dapat menggunakan kamera sebagai sumber aliran data visual, maka Sistem Operasi tersebut mesti mendukung penggunaan kamera dalam modus pratayang. *Augmented reality* adalah sebuah presentasi dasar dari aplikasi-aplikasi navigasi. Dengan menggunakan GPS maka aplikasi pada telepon genggam dapat mengetahui keberadaan penggunanya pada setiap waktu. Khusus untuk Sistem Operasi iPhone dan Android, ada 2 pemain besar (Layar dan Wikitude) di dunia Augmented reality.

2. Hiburan

   Bentuk sederhana dari *Augmented reality* telah dipergunakan dalam bidang hiburan dan berita untuk waktu yang cukup lama. Contohnya adalah pada acara laporan cuaca dalam siaran televisi di mana wartawan ditampilkan berdiri di depan peta cuaca yang berubah. Dalam studio, wartawan tersebut sebenarnya berdiri di depan layar biru atau hijau. Pencitraan yang asli digabungkan dengan peta buatan komputer menggunakan teknik yang bernama chroma-keying.

Princeton Electronic Billboard telah mengembangkan sistem Augmented reality yang memungkinkan lembaga penyiaran untuk memasukkan iklan ke dalam area tertentu gambar siaran. Contohnya, ketika menyiarkan sebuah pertandingan sepak bola, sistem ini dapat menempatkan sebuah iklan sehingga terlihat pada tembok luar stadium.

3. Kedokteran (*Medical)*

   Teknologi pencitraan sangat dibutuhkan di dunia kedokteran, seperti misalnya, untuk simulasi operasi, simulasi pembuatan vaksin virus, dll. Untuk itu, bidang kedokteran menerapkan augmented reality pada visualisasi penelitian mereka.

4. *Engineering Design*

   Seorang engineering design membutuhkan *augmented reality* untuk menampilkan hasil desain mereka secara nyata terhadap klien. Dengan *augmented reality* klien dapat mengtahui, tentang spesifikasi yang lebih detail tentang desain mereka.

5. *Robotics* dan *Telerobotics*

   Dalam bidang robotika, seorang operator robot, mengunnakan pengendali pencitraan visual dalam mengendalikan robot itu. Jadi, penerapan *augmented reality* dibutuhkan di dunia robot.

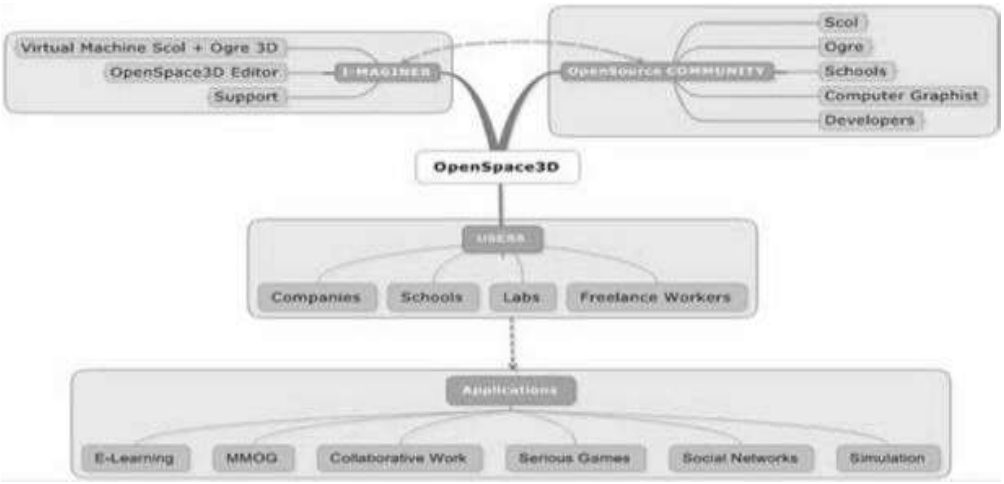
6. Pendidikan

   Dalam dunia pendidikan buku adalah pegangan setiap murid maupun guru, buku hanya berisi tulisan-tulisan dan gambar yang abstrak. Penerapan *augmented reality* bisa diterapkan dalam dunia pendidikan, seperti contoh pada pelajaran matematika dalam materi dimensi tiga (bangun ruang).

### 2.2.3 *OpenSpace* 3D

Sebuah editor atau scene manager objek tiga dimensi yang bersifat *open source* dengan menggunakan OGRE sebagai *Graphic Rendering*. Dengan *Openspace* 3D aplikasi game atau simulasi tiga dimensi bisa dibuat secara mudah

tanpa terlibat secara langsung dengan programing. Aplikasi *Openspace* 3D bersifat sebagai sebuah scene manager dan editor dalam pengaturan skenario sehingga pengguna hanya perlu memasukan *resource* atau sumber daya yang dibutuhkan seperti objek tiga dimensi dalam bentuk mesh OGRE, material, texture dan multimedia lainnya mencakup audio dan video. Untuk menghindari pemrograman yang sulit, *Openspace* 3D menyediakan sebuah hubungan relasional antar objek yang terdiri dari plugin yang cukup lengkap dalam membuat suatu aplikasi tiga dimensi baik simulasi atau game dan masih banyak lagi fitur yang di sediakan oleh aplikasi *Openspace* 3D ini [15].




**Gambar 2-8 Alur kerja Openspace 3D**

Aplikasi *Openspace* 3D ini berbasiskan bahasa pemrograman SCOL, yang merupakan bahasa pemrograman yang berasal dari Perancis dan baru-baru ini dikembangkan. *Openspace* 3D menggunakan *graphic engine* OGRE 3D yang mempunyai komunitas cukup banyak tapi tidak di Indonesia. Kelemahan *Openspace* 3D adalah *output*-nya yang tidak kompatibel, untuk menjalankan aplikasi, diharuskan menginstal SCOLVOY@GER, yaitu sebuah *runtime* dari SCOL. Ada alasan mengapa harus menginstal SCOL, karena sebenarnya *Openspace* 3D ditujukan untuk browser, jadi aplikasi atau simulasi yang dibuat bisa ditampilkan dalam suatu *website* pribadi, meskipun demikian pada versi terbaru dari *Openspace* 3D telah menyediakan fasilitas untuk membuat file eksekusi sehingga menjadi sebuah aplikasi stand alone untuk Windows. Kelebihan lainnya dari Openspace 3D adalah kompatibilitas dengan file

multimedia lainnya seperti *Video Youtube, Chating*, Mp3, Wav, SWF dan lain-lain. *Openspace* 3D juga mendukung *input controler dari joypad, keyboard, mouse, Wii Nintendo joystick*, dan juga *voice controler.*

### 2.2.4 Penjelasan 3D Studio Max

3D Studio Max atau biasa dikenal dengan 3D Max adalah suatu *software* (Perangkat lunak) untuk membuat sebuah grafik vektor 3 dimensi dan animasi. ditulis oleh *Autodesk* Media & *Entertainment*, dulunya dikenal sebagai *Discreet and Kinetix*. 3D Studio Max dikembangkan dari pendahulunya yaitu 3D Studio for DOS, tetapi untuk *platform* Win32 [9]. Kinetix kemudian bergabung dengan akuisisi terakhir *Autodesk, Discreet Logic*. Yang sampai saat penulis membuat artikel ini yang terbaru adalah 3D Studio Max versi 9. Para desain grafis banyak menggunakan *software* ini digunakan untuk membuat sebuah film animasi, arsitektur rumah, ataupun membuat logo suatu perusahaan.

3d Max mempunyai tiga titik koordinat, yaitu dengan memperlihatkan terhadap sumbu-sumbu x, y, z dan sudut yang terjadi. Ketiga sistem koordinat itu ialah :

1. Koordinat *Cartesian (rectangular coordinat)*.

   Menentukan koordinat dengan menggunakan sumbu-sumbu x, y, z. yaitu (x), (y), (z). Penulisannya (0.5,0.9,0.0); (0.42,0.39,0.82).
2. Koordinat *cylindrical*

   Cara ini mengabungkan antara jarak, sudut dan koordinat sumbu z yaitu: (jarak)< (sudut),(z) Penulisannya: (.03<60.95,0.0);(0.57<43,0.82).
3. Koordinat *spherical*

   Cara ini menggabungkan antara jarak dan dua sudut, dan masing-masing besaran dipisahkan dengan tanda<, yaitu: (jarak)<(sudut)<(sudut), penulisannya: (1.03<60.95<0); (1<43<55).

### 2.2.5 Adobe Flash

Adobe flash adalah salah satu aplikasi pembuat animasi yang cukup dikenal saat ini. *Adobe Flash CS4 Profesional* telah membuktikan dirinya sebagai

program animasi dua dimensi berbasis vektor dengan kemampuan profesional. Dalam perkembangannya, *Flash* selalu melakukan banyak penyempurnaan pada setiap versinya. *Adobe Flash CS4 Profesional* menghadirkan fitur-fitur baru yang menjadikan flash semakin diakui sebagai program animasi handal [8].

Hebatnya lagi, dengan menggunakan *Adobe Flash CS4*, berbagai aplikasi animasi 2D dapat dibuat mulai dari animasi kartun, animasi *interaktif, game, company profile*, presentasi, *video* clip, *movie*, web animasi dan aplikasi animasi lainnya sesuai kebutuhan kita. Banyak sekali keunggulan dan kecanggihan Flash dalam membuat dan mengolah animasi, seperti :

1. Dapat membuat tombol interaktif dengan sebuah *movie* atau objek yang lain.
2. Dapat merubah perubahan transparansi warna dalam *movie*
3. Membuat perubahan animasi dari satu bentuk ke bentuk lain
4. Membuat animasi tranformasi 3D dan animasi dekorasi yang merupakan salah satu fitur terbaru
5. Mampu membuat animasi bone yang mengadopsi dari sistem pertualangan sehingga menghasilkan animasi yang semakin atraktif
6. Dapat membuat gerakan animasi dengan mengikuti alur yang telah ditetapkan
7. Dapat dikonversi dan dipublikasikan (publish) ke dalam beberapa tipe diantaranya adalah: .swf, .html, .gif, .jpg, .png, .exe, .mov.

### 2.2.6 *Unified Modeling Language* (UML)

*UML (Unified Modeling Language)* adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (Object-Oriented). [14]

*UML* sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem *software*.

*UML* adalah salah satu *tool* / model untuk merancang pengembangan *software* yang berbasis object oriented. *UML* sebagai sebuah bahasa yang memberikan *vocabulary* dan tatanan penulisan kata-kata dalam '*MS Word'* untuk kegunaan komunikasi. Sebuah bahasa model adalah sebuah bahasa yang mempunyai *vocabulary* dan konsep tatanan / aturan penulisan serta secara fisik mempresentasikan dari sebuah sistem.

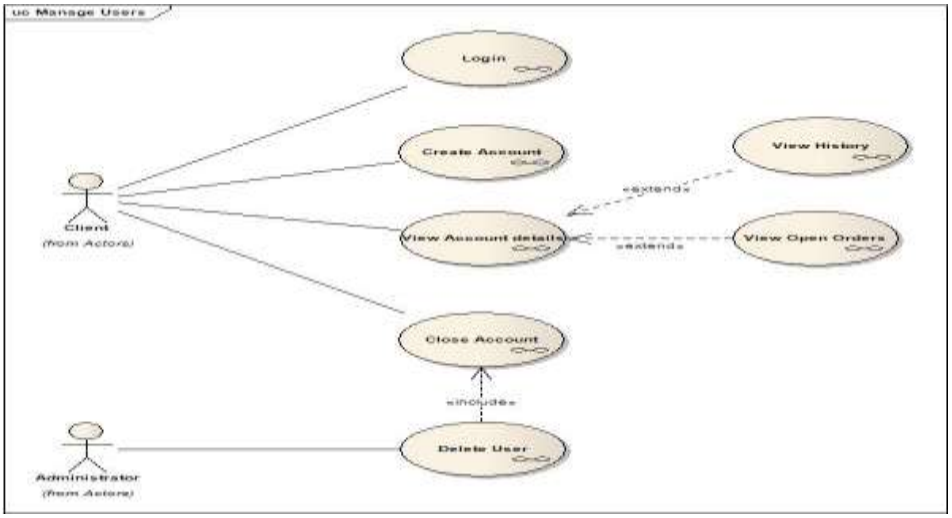
*UML* adalah sebuah bahasa standard untuk pengembangan sebuah *software* yang dapat menyampaikan bagaimana membuat dan membentuk model-model, tetapi tidak menyampaikan apa dan kapan model yang seharusnya dibuat yang merupakan salah satu proses implementasi pengembangan software.

*UML* tidak hanya merupakan sebuah bahasa pemograman visual saja, namun juga dapat secara langsung dihubungkan ke berbagai bahasa pemograman, seperti *JAVA*, C++, *Visual Basic*, atau bahkan dihubungkan secara langsung ke dalam sebuah *object-oriented* database. Begitu juga mengenai pendokumentasian dapat dilakukan seperti; *requirements*, arsitektur, *design*, *source code*, *project plan*, *tests*, *dan prototypes*.

#### 2.2.6.1 Diagram *Unified Modeling Language* (UML)

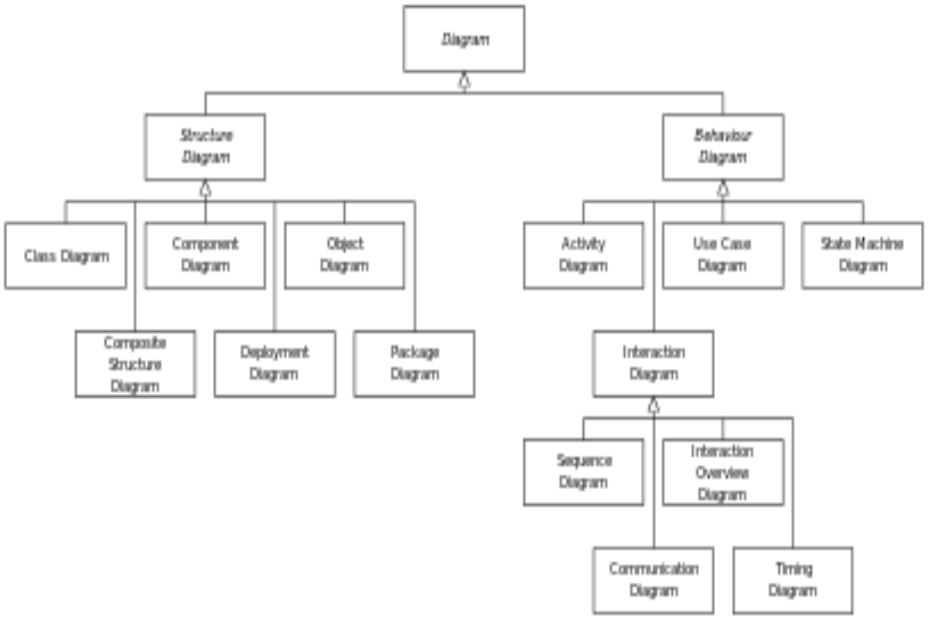
UML terdiri atas pengelompokan diagram-diagram sistem menurut aspek dan sudut pandang tertentu. Diagram merupakan gambaran permasalahan maupun solusi dari permasalahan suatu model. UML mempunyai 9 diagram, yaitu :

1. *Use Case* Diagram, menggambarkan sekelompok use cases dan aktor yang disertai dengan hubungan diantaranya. Diagram *use cases* ini menjelaskan dan menerangkan kebutuhan / requirement yang diinginkan/ dikehendaki *user*/pengguna, serta sangat berguna dalam menentukan struktur organisasi dan model dari pada sebuah sistem.




**Gambar 2-9 Contoh *Use Case Diagram***

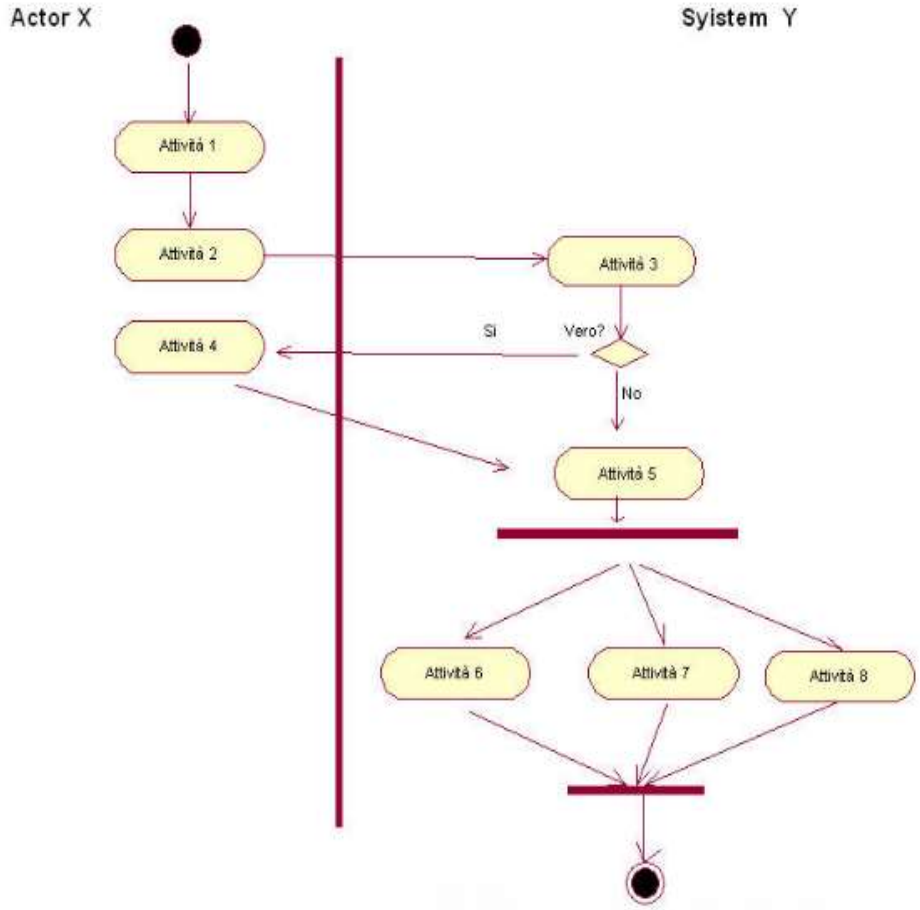
2. *Class Diagram*, yang memperlihatkan struktur statis dari kelas actual didalam sistem.




**Gambar 2-10 Contoh Class Diagram**

3. State Diagram, yang memperliatkan semua keadaan (state) yang dapat dimiliki oleh kelas dan event yang dapat merubah keadaan tersebut.
4. *Sequence* Diagram, yang memperlihatkan kolaborasi dinamik antara objek-objek dengan suatu urutan pesan antar objek tersebut.

5. *Collaboration* Diagram, yang memperlihatkan kolaborasi dinamik antar objek tanpa memperhatikan aspek waktu.
6. *Activity* Diagram, yang memperlihatkan aliran urutan aktifitas.




**Gambar 2-11 Contoh Activity Diagram**

7. *Component* Diagram, yang memperlihatkan struktur fisik dari *source code* dalam terminology code components. Komponen berisi informasi tentang logical *class* dapat berupa komponen *source code*, komponen biner atau komponen yang dapat dieksekusi.
8. Deployment Diagram, yang memperlihatkan arsitektur fisik dari hardware dan software pada sistem.

9. Object Diagram, yang merupakan varian dari kelas diagram yang memperlihatkan lebih detail banyaknya obyek yang mengintantiasi (*instances*) kelas.

### 2.2.7 *Image Processing*

Citra (*image*) adalah istilah lain untuk gambar sebagai salah satu komponen multimedia memegang peranan sangat penting sebagai bentuk informasi visual. Citra mempunyai karakteristik yang tidak dimiliki oleh data teks, yaitu citra kaya dengan informasi. Maksudnya sebuah gambar dapat memberikan informasi lebih banyak daripada informasi tersebut disajikan dalam bentuk teks.

Pengolahan gambar digital atau *Digital Image Processing* adalah bidang yang berkembang sangat pesat sejalan dengan kemajuan teknologi pada industri saat ini. Fungsi utama dari *Digital Image Processing* adalah untuk memperbaiki kualitas dari gambar sehingga gambar dapat dilihat lebih jelas tanpa ada ketegangan pada mata, karena informasi penting diekstrak dari gambar yang dihasilkan harus jelas sehingga didapatkan hasil yang terbaik.

*Image processing* adalah bidang tersendiri yang sudah cukup berkembang sejak orang mengerti bahwa computer tidak hanya dapat menangani data teks, tetapi juga ada citra [6]. Teknik-teknik pengolahan citra biasanya digunakan untuk melakukan transformasi dari suatu citra kepada citra yang lain, sementara tugas perbaikan informasi terletak pada manusia melalui penyusunan algoritmanya. Bidang ini meliputi penajaman citra, penonjolan fitur tertentu dari suatu citra, kompresi citra dan koreksi citra yang tidak fokus atau kabur. Sebaliknya, sistem visual menggunakan citra sebagai masukan tetapi menghasilkan keluaran jenis lain seperti representasi dari kontur objek di dalam citra, atau menghasilkan gerakan dari suatu peralatan mekanis yang terintegrasi dengan sistem visual.

Berkat adanya mata sebagai indera penglihatan yang sangat penting dalam kehidupan sehari-hari, manusia dapat melakukan banyak hal dengan lebih mudah. Berbagai aktifitas seperti berjalan, mengambil sesuatu benda, menulis, apalagi membaca buku, menjadi sangat mudah dilakukan bila melibatkan fungsi mata. Peristiwa melihat yang begitu sederhana bagi kita dalam kehidupan sehari-

hari ternyata melibatkan banyak proses dan aliran data yang besar. Dengan menggunakan sifat-sifat seperti halnya mata, maka hal di atas dapat diaplikasikan dalam perangkat keras pengolahan citra seperti *webcam, handycam, camera digital, scanner,* dan lain-lain.

### 2.2.8 Pemrograman Berorientasi Objek

OOP/PBO merupakan paradigma pemrograman yang popular saat ini yang telah menggantikan teknik pemrograman berbasis prosedur. *Object Oriented* Programing yang berarti pula Pemrograman Beorientasi Objek sudah ditemukan sekitar tahun 1960 dan dikembangkan pada permulaan tahun 1970.

Pemrograman Berorientasi Objek (*Object Oriented Programming*/OOP) merupakan pemrograman yang berorientasikan kepada objek, dimana semua data dan fungsi dibungkus dalam class-class atau *object-object*. Setiap *object* dapat menerima pesan, memproses data, mengirim, menyimpan dan memanipulasi data. Beberapa object berinteraksi dengan saling memberikan informasi satu terhadap yang lainnya. Masing-masing object harus berisikan informasi mengenai dirinya sendiri dan dapat dihubungkan dengan *Object* yang lain. Pemrograman berorientasi objek berbeda dengan pemrograman prosedural yang hanya menggunakan satu halaman kebawah untuk mengerjakan banyak perintah atau statement. Penggunaan pemrograman berorientasi objek sangat benyak sekali, contoh : java, php, perl, c#, cobol, dan lainnya.

Dalam konsep Pemrograman Berorientasi Objek dikenal beberapa istilah umum, yaitu :

1. *Attribute*

   Atribut dari sebuah kelas adalah variabel global yang dimiliki sebuah kelas, Atribut dapat memiliki hak akses private, public maupun protected. Sebuah atribut yang dinyatakan sebagai private hanya dapat diakses secara langsung oleh kelas yang membungkusnya, sedangkan kelas lainnya tidak dapat mengakses atribut ini secara langsung. Sebuah atribut yang dinyatakan sebagai public dapat diakses secara langsung oleh kelas lain di luar kelas yang

membungkusnya. Sebuah atribut yang dinyatakan sebagai protected tidak dapat diakses secara langsung oleh kelas lain di luar kelas yang membungkusnya, kecuali kelas yang mengaksesnya adalah kelas turunan dari kelas yang membungkusnya. Atribut juga biasa disebut state/ properties/ *field* merupakan bagian yang dimiliki oleh sebuah class yang tidak melakukan operasi, tetapi kepadanya dilakukan operasi sehingga dapat merubah nilai dari attribute tersebut.

2. *Method*

   *Method* adalah fungsi atau prosedur yang dibuat oleh seorang programmer didalam suatu *Class*. Dengan kata lain, method pada sebuah kelas hampir sama dengan fungsi atau prosedur pada pemrograman prosedural. Pada sebuah method di dalam sebuah kelas juga memiliki izin akses seperti halnya atribut pada kelas, izin akses itu antara lain private, public dan *protected* yang memiliki arti sama pada izin akses atribut yang telah dibahas sebelumnya. Sebuah kelas boleh memiliki lebih dari satu method dengan nama yang sama asalkan memiliki parameter masukan yang berbeda sehingga kompiler atau interpreter dapat mengenali method mana yang dipanggil. Hal ini dinamakan overloading. Di dalam sebuah kelas, terdapat juga yang disebut sebagai method atau atribut statis yang memiliki kata kunci static. Maksud dari statis di sini adalah method yang dapat diakses secara berbagi dengan semua objek lain tanpa harus membuat objek yang memiliki method statis tadi (tanpa proses new), tapi sebuah method statis mempunyai keterbatasan yaitu hanya dapat mengakses atribut atau method lain di dalam kelas yang membungkusnya yang juga bersifat statis. Method statis biasanya diimplementasikan untuk method main.

3. *Class*

   Merupakan model yang berisi kumpulan attribute dan method dalam suatu unit untuk suatu tujuan tertentu. Sebagai contoh class manusia memiliki attribute berat, tinggi, usia kemudian memiliki method

makan, minum, tidur. *Method* dalam sebuah class dapat merubah attribute yang dimiliki oleh class tersebut. Sebuah class merupakan dasar dari modularitas dan struktur dalam pemrograman berorientasi *object*.

4. *Object*

   Merupakan perwujudan dari *class*, setiap *object* akan mempunyai attribute dan method yang dimiliki oleh *class*-nya, contohnya: amir, ahmad, yani merupakan *object* dari *class* manusia. Setiap *object* dapat berinteraksi dengan *object* lainnya meskipun berasal dari *class* yang berbeda.

5. *Abstraction*

   Adalah suatu cara untuk melihat suatu object dalam bentuk yang lebih sederhana. Sebagai contoh kita tidak melihat sebuah mobil sebagai ribuan komponen elektronik, sistem mekanik dengan empat buah ban, jok, stir kemudi dan sebagainya. Dengan Abstraction, suatu sistem yang kompleks dapat dipandang sebagai kumpulan subsistem-subsistem yang lebih sederhana, seperti halnya mobil merupakan suatu sistem yang terdiri atas berbagai subsistem, seperti subsistem kemudi, subsistem pengereman dan sebagainya.

6. *Encapsulation*

   Merupakan suatu mekanisme untuk menyembunyikan atau memproteksi suatu proses dari kemungkinan interferensi atau penyalahgunaan dari luar sistem dan sekaligus menyederhanakan penggunaan sistem tersebut.

7. *Inheritance*

   Merupakan konsep mewariskan *attribute* dan *method* yang dimiliki oleh sebuah class kepada class turunannya. Dengan konsep ini class yang dibuat cukup mendefinisikan attribute dan *method* yang spesifik didalamnya, sedangkan attribute dan method yang lebih umum akan didapatkan dari class yang menjadi induknya.

### 2.2.9 Review Literatur

Berikut ini adalah penelitian yang berkaitan dengan *Augmented Reality*:

1. Jinseok Seo, Namgyu Kim dan Gerard J. Kim (Lecture Notes in Computer Science, 2006, Volume 3942/2006, 1188-1197, DOI: 10.1007/11736639_149)"Designing Interactions for Augmented Reality based Educational Contents".

   Pada Penelitian ini dibahas mengenai panduan untuk merancang interaksi augmented reality (AR) dengan konsep pendidikan berupa object 3D real-time tentang "sirkulasi air" yang diharapkan mampu memberikan pengalaman nyata yang berguna bagi siswa. Hasil penelitian diperoleh data bahwa penggunaan teknologi AR pada kelompokkelas pembelajaran lebih memudahkan penjelasan materi tentang "sirkulasi air".
2. Penelitian ini dilakukan oleh Mukhlis Yuzti Perdana, Yuli Fitrisia dan Yusapril Eka Putra yang berjudul "Aplikasi Augmented Reality Pembelajaran Organ Pernapasan Manusia Pada Smartphone Android". Penelitian ini membahas mengenai bagaimana memanfaatkan teknologi AR untuk menjadi alat belajar. Dengan pemanfaatan teknologi AR ini, diharapkan akan membuat pemahaman siswa menjadi lebih baik dibandingkan dengan hanya melalui media buku yang biasanya hanya akan mengerti teorinya saja. Sedangkan jika menggunakan alat bantu peraga, biayanya yang cukup mahal dan alat peraga juga memiliki keterbatasan dalam jumlah dan fungsinya.
3. Penelitian ini dilakukan oleh Akhmad Afisuunani, Akuwan Saleh, M. dan Hasbi Assidiqi yang berjudul "Multi Marker *Augmented Reality* untuk Aplikasi *Magic Book*". Penelitian ini membahas mengenai pembuatan *magic book* dengan model animasi 3D, terdapat tiga bagian dari *magic book* tersebut, yaitu: menulis, membaca dan mewarnai. Pada bagian menulis, pengguna harus menggabungkan titik-titik berbentuk huruf menggunakan pensil yang merupakan bagian dari marker. Apabila titik-titik sudah terhubung dari ujung ke ujung, maka program akan mengenai marker sehingga object berupa huruf-huruf 3D akan muncul diatas marker

yang bisa dilihat pada layar monitor. Sebenarnya, dalam penelitian ini lebih ditekankan kepada pemanfaatan multi marker sehingga animasi yang dibuat lebih interaktif dan lebih *real*.

Dari ketiga penelitian tersebut, telah ada penelitian mengenai teknologi *Augmented Reality*, *Augmented Reality* tentang sirkulasi air, tentang pembelajaran organ pernapasan, dan pembahasan mengenai pembuatan magic book. Tetapi, dapat disimpulkan dari ketiga penelitian tersebut. Belum ada peneliti/penulis yang membahasa tentang suatu pembelajaran bangun ruang memanfaatkan teknologi AR.