

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 Media Pembelajaran Interaktif**

Media pembelajaran interaktif merupakan salah satu media yang dapat digunakan untuk menyampaikan materi pembelajaran kepada peserta didik dengan sangat efektif dan efisien. Keunggulan utama media pembelajaran interaktif yaitu interaktivitas itu sendiri membuka berbagai peluang interaksi antara pengguna dengan media. Namun demikian untuk membentuk interaktivitas yang baik diperlukan pengetahuan yang baik tentang desain antar muka dan teknik pemrograman, dua hal ini menjadi kelemahan bagi sebagian besar tenaga pengajar di Indonesia[8].

#### **2.2 Multimedia**

- Multimedia adalah kombinasi dari komputer dan video(Rosch, 1996)
- Multimedia secara umum merupakan kombinasi dari tiga elemen yaitu suara, gambar dan teks (McCormick, 1996).
- Multimedia adalah kombinasi dari paling sedikit dua media input dan output dari data, media ini dapat audio (suara, musik) animasi video, teks, grafik dan gambar (Turban dkk, 2002).
- Multimedia adalah pemanfaatan komputer untuk membuat dan menggabungkan teks, grafik, audio dan gambar bergerak (video dan animasi) dengan menggabungkan link dan tool yang memungkinkan pemakai melakukan navigasi, berinteraksi, berkreasi dan komunikasi (Hofstetter, 2001).
- Dalam definisi ini terkandung empat komponen penting, jika salah satu tidak ada, maka bukan multimedia dalam pendefinisian dalam arti yang luas.
- Dari beberapa definisi di atas, maka multimedia ada yang online(internet) dan ada yang offline(tradisional)[9].

### 2.3 Video

Video adalah media yang menampilkan unsur gambar (visual) dan suara (audio) secara bersamaan pada saat mengkomunikasikan pesan atau informasi. Media video dapat diklasifikasikan sebagai media audio-visual. Walau bentuk fisiknya berbeda, media ini memiliki keasamaan dengan film, yakni sama-sama mampu menayangkan gambar yang bergerak. Media video telah banyak digunakan untuk berbagai keperluan mulai dari hiburan, sampai bidang pendidikan (pembelajaran). Media ini dapat mengungkapkan objek dan peristiwa seperti keadaan yang sesungguhnya. Perencanaan yang baik dalam menggunakan media video akan membuat proses komunikasi (pembelajaran) mejadi lebih efektif. Audiovisual terbagi dua macam, yakni :

1. Audiovisual murni yaitu baik unsur suara maupun unsur gambar berasal dari satu sumber seperti kaset video.
2. Audiovisual tidak murni yaitu unsur suara dan gambarnya berasal dari sumber yang berbeda. Misalnya film bingkai suara yang unsur gambarnya berasal dari slide proyektor dan unsur suaranya berasal dari tape recorder[10].

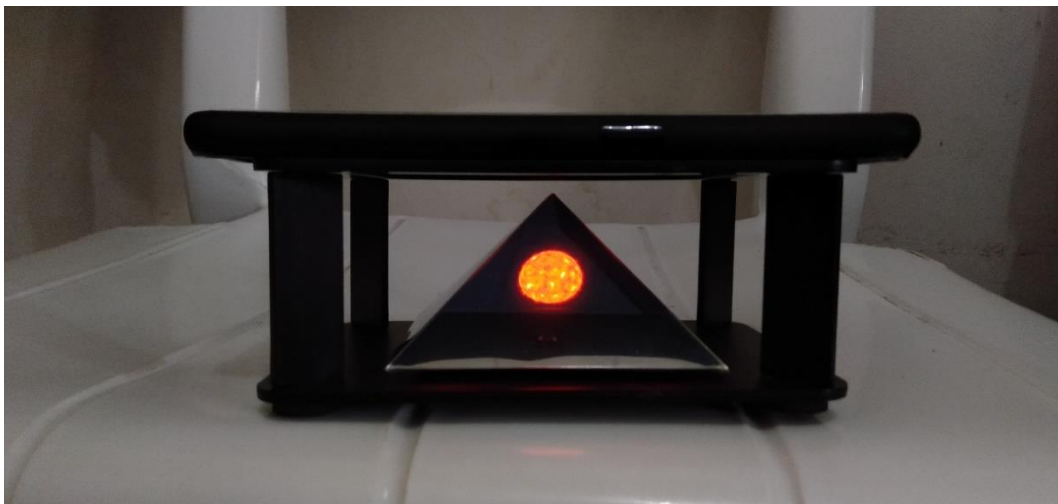
Menurut Munir (2012: 289) "Video adalah teknologi penangkapan, pengolahan, penyimpanan, pemindahan dan perekonstruksian urutan gambar diam dengan menyajikan adegan-adegan dalam gerak secara elektronik". Video menyediakan sumberdaya yang kaya dan hidup bagi aplikasi multimedia[9].

### 2.4 Video 3D Hologram

Hologram merupakan produk dari teknologi holografi. Hologram terbentuk dari perpaduan dua sinar cahaya yang koheren dan dalam bentuk mikroskopik. Hologram bertindak sebagai gudang informasi optik. Informasi-informasi optik ini kemudian akan membentuk suatu gambar, pemandangan, atau adegan dengan prinsip kerja 3D. Hologram juga merupakan tiga dimensi dari iterfensi positif dari gelombang cahaya laser yang mempunyai kelebihan yang mampu menyimpan informasi yang didalamnya memuat objek-objek tiga dimensi

(3D). Perkembangan dunia desain komunikasi visual ke arah alternatif 3D *virtual reality* teknologi yang disebut “*Holografi*” suatu teknik yang memungkinkan cahaya dari suatu benda yang tersebar direkam dan kemudian direkonstruksikan sehingga objek seolah-olah berada pada posisi sama dengan media rekaman yang direkam[3].

3D Video Hologram adalah media yang menampilkan video 3D dan suara secara bersamaan untuk menyampaikan pesan atau informasi, terlihat seperti melihat objek secara langsung, seperti gambar 2.1.



**Gambar 2.1 3D Hologram Menggunakan Frame**

## **2.5 Smartphone**

Smartphone adalah telepon yang menyediakan fitur yang berada diatas dan di luar kemampuan sederhana untuk membuat panggilan telepon. Ponsel pertama yang harus dipertimbangkan smartphone dirancang oleh IBM pada tahun 1992. Sepanjang akhir tahun 1990-an, berbagai jenis perangkat ponsel canggih muncul di pasaran. Layar sentuh mulai menggantikan keysets pada banyak smartphone. Saat ini, fungsi smartphone dengan kaya fitur perangkat komunikasi. Pengguna dapat berselancar di Internet dengan kemudahan yang sama seperti ketika menggunakan laptop atau dekstop.

Banyak produsen smarthphone telah bekerja untuk meningkatkan kejelasan dan integritas sinyal audio dasar ponsel, mempertajam foto kamera, kapasitas memori, kecepatan mengakses internet, 4G dan free install aplikasi internet. Hal ini membantu untuk memastikan penambahan fitur tambahan. Berikut data pengguna smartphone menurut penelitian Aaron Smith (2015):

- a. Dari 5 miliar pengguna ponsel di dunia 1,08 miliar adalah pengguna smartphone.
- b. Android platform smartphone memiliki pangsa pasar 46,9%.
- c. 89% dari pengguna smartphone menggunakan smartphone mereka sepanjang hari
- d. 92% dari pengguna smarthphone menggunakan smartphone mereka untuk mengirim pesan teks ke ponsel lainnya. Sedangkan 84% pengguna menggunakan smartphone mereka untuk browsing internet.
- e. Kelompok usia 25-34 tahun memiliki tingkat penetrasi tertinggi smartphone, sekitar 62%.
- f. Pemilik Android Smartphone mengkonsumsi jumlah tertinggi dari data pada 525 Mbs sebulan, sementara pemilik iPhone pada mengkonsumsi 492 Mbs data sebulan.
- g. Mendownload aplikasi adalah data yang paling populer penggunaan untuk pengguna smartphone[11].

Contoh gambar smartphone seperti pada gambar 2.2.



**Gambar 2.2 Smartphone**

(Sumber : <https://www.freepngimg.com/thumb/smartphone/9-2-smartphone-png-picture-thumb.png>)

## 2.6 Tata Surya

Tata surya adalah kumpulan delapan planet utama, ditambah objek-objek kecil lainnya, yang mengelilingi Matahari. Tata surya terdiri dari Matahari dan segala sesuatu yang terikat oleh gravitasinya. Segala sesuatu tersebut termasuk 8 planet utama, yakni Merkurius, Venus, Bumi, Mars, Jupiter, Saturnus, Uranus dan Neptunus, beserta bulan-bulan mereka; asteroid; planet kerdil; Objek Sabuk Kuiper; meteorid; komet; hingga debu antar planet.

Gravitasi Matahari begitu besar. Bahkan, tarikan gravitasi Matahari diperkirakan hingga mencapai jarak hampir 2 tahun cahaya darinya atau sekitar 18,8 triliun kilometer, hampir setengah jarak ke bintang terdekat dari tata surya kita: Alfa Centauri[12].

Sistem tata surya kita sangat menarik. Ia terbentuk 4,6 miliar tahun lalu dan terdiri atas planet-planet yang mengitari Matahari. Selain planet, tata surya kita juga memiliki satelit alami atau bulan, asteroid, komet, planet-planet kerdil, debu, gas dan objek-objek lain yang menarik[2] seperti pada gambar 2.3.



**Gambar 2.3 Sistem Tata Surya**

(Sumber : [https://i.dlpng.com/static/png/442379\\_preview.png](https://i.dlpng.com/static/png/442379_preview.png))

### 2.2.1 Matahari adalah bintang di pusat tata surya kita

Matahari adalah bintang di pusat tata surya kita dan bertanggung jawab atas iklim dan cuaca Bumi. Matahari berbentuk bola yang hampir sempurna dengan selisih diameter hanya 10 kilometer antara kutub dan ekuatornya. Radius Matahari sendiri adalah sekitar 695.508 kilometer (109,2 kali radius Bumi) di mana 20-25% dari total radius itu adalah bagian intinya.

Matahari diperkirakan terbentuk sekitar 4,6 miliar tahun yang lalu, sehingga kini ia sudah termasuk dalam jenis bintang deret utama kelas G2V. Dengan massa total  $1,99 \times 10^{30}$  kilogram, hal itu membuat Matahari mendominasi sekitar 99,86% total seluruh massa seluruh tata surya.

Suhu permukaan Matahari diketahui mencapai sekitar 5.500 derajat Celsius. Para astronom mengetahui suhu ini dengan mempelajari spektrum atau cahaya Matahari. Semakin panas sebuah bintang, semakin biru warnanya. Dengan kata lain, berarti Matahari termasuk ke dalam bintang dengan suhu rata-rata saja. Walaupun suhu permukaannya terbilang rendah, hal itu berbeda dengan suhu di inti Matahari. Energi Matahari yang dihasilkan melalui proses fusi nuklir, proses ketika Hidrogen dikonversi menjadi Helium. Membuat suhu intinya mencapai

hampir 15 juta derajat Celsius[12]. Gambar matahari ditunjukkan pada gambar 2.4.



**Gambar 2.4 Matahari**

(Sumber : <https://i.pinimg.com/originals/03/e9/89/03e989da104a7685ba95b0a541961d5c.png>)

### **2.2.2 Planet-planet**

Menurut definisi yang dibuat oleh Persatuan Astronomi Internasional pada tahun 2006, planet adalah “Objek langit yang mengorbit Matahari; memiliki massa yang cukup untuk membentuk dirinya menjadi bulat (atau hampir bulat); bukan merupakan satelit alami (bulan) dari objek lain; dan telah mampu membersihkan puing-puing atau benda-benda kecil dari daerah di sekitar orbitnya.”.

Nah, dengan definisi tadi, mulai dari Merkurius hingga Neptunus di tata surya kita, masuk kriteria sebagai “planet”. Tapi tidak dengan pluto. Banyak salah kaprah yang beredar di masyarakat awam yang mengatakan bahwa Pluto telah hilang dari tata surya. Namun, sebenarnya hal itu keliru. Pluto tidak pernah hilang. Hanya saja, Pluto kini diturunkan statusnya sebagai “planet kerdil”.

Definisi “Planet” di atas menjelaskan bahwa sebuah objek langit harus mampu membersihkan puing-puing atau objek lain dari orbitnya. Nah, karena Pluto tidak bisa membersihkan daerah sekitar orbitnya, ditambah ukuran Pluto terlalu kecil, maka definisi planet kerdil pun lahir.

Menurut sejarahnya, istilah “planet” awalnya berasal dari kata Yunani yang berarti “pengembara”. Dulunya orang-orang Yunani mengamati langit dan menemukan adanya “bintang-bintang yang bergerak”, gerakannya berbeda dengan bintang-bintang lainnya, yang sebenarnya merupakan planet.

Tetapi, baru pada saat teleskop dikembangkan pada tahun 1600-an oleh Galileo Galieei, para astronom terdahulu dapat melihat “bintang-bintang bergerak” tersebut secara lebih terperinci. Perkembangan teleskop kala itu tidak hanya berhasil mengungkapkan wujud planet yang sebenarnya, tetapi juga berhasil membuat manusia menemukan bulan-bulan yang mengelilingi planet Jupiter hingga cincin yang mengelilingi Saturnus[12].

#### **A. Merkurius**

Merkurius merupakan planet terkecil dalam Sistem Tata Surya. Sekitar 75% radius planet ini merupakan inti besi. Inti merkurius paling banyak mengandung besi dibandingkan planet lain di Sistem Tata Surya kita. Merkurius merupakan planet terdekat dengan Matahari, tapi terkadang suhunya sangat dingin. Fluktuasi antara suhu tertinggi dan terendah di planet ini adalah yang palig ekstrem dalam Sistem Tata Surya.

Posisi: Planet terdekat dari Matahari

Jarak dari Matahari: 57,91 juta km

1 hari (periode rotasi): 1403,49 jam (58,65 hari Bumi)

1 tahun (periode orbit): 87,97 hari Bumi

Massa: 0,055 massa Bumi

Gravitasi: 0,38 gravitasi Bumi

Suhu: -143 hingga 427<sup>o</sup>C

Atmosfer: Oksigen (O<sub>2</sub>)

Bulan yang diketahui: 0

Cincin: 0

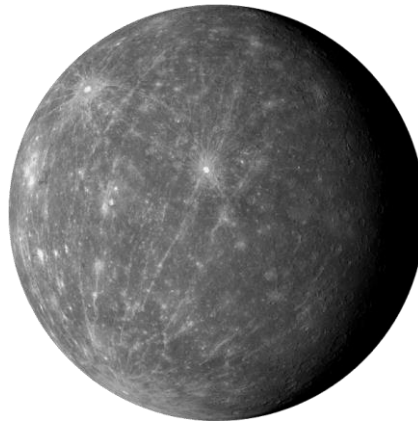
Arti nama: Pembawa pesan dewa-dewa Romawi

Titik tertinggi: Caloris Montes, Pegunungan Lereng Basin Barat Laut

(30,5 KM)



Gambar planet merkurius ditunjukkan pada gambar 2.5.



**Gambar 2.5 Merkurius**

(Sumber : <https://toppng.com/uploads/preview/mercury-planet-mercury-11563652843mkaznrjva.png>)

## **B. Venus**

Satu hari di Venus lebih panjang daripada satu tahun di Bumi karena planet ini berputar berlawanan dengan arah jarum jam. Venus disebut sebagai kembaran Bumi karena memiliki ukuran, komposisi, massa dan gravitasi yang mirip dengan Bumi. Venus sangat panas dan berangin ekstrem. Angin di Venus (724 km/jam) lebih cepat daripada tornado tercepat di Bumi.

Posisi: Planet kedua dari Matahari

Jarak dari Matahari: 108,2 juta km

1 hari (periode rotasi): 5815,47 jam (243,02 hari Bumi)

1 tahun (periode orbit): 224,70 hari Bumi

Massa: 0,815 massa Bumi

Gravitasi: 0,91 gravitasi Bumi

Suhu: 471°C

Atmosfer: Karbon Dioksida (CO<sub>2</sub>)

Bulan yang diketahui: 0

Cincin: 0

Arti nama: Dewi cinta dan kecantikan Romawi

Titik tertinggi: Maxwell Montes, Ishtar Terra (10,9 km)

Titik terendah: Diana Chasma, Aphrodite Terra (3,05 km)

Gambar planet Venus ditunjukkan pada gambar 2.6.



**Gambar 2.6 Venus**

(Sumber : <http://www.pngmart.com/files/4/Venus-PNG-Transparent-Image.png>)

### **C. Bumi**

Bumi merupakan satu-satunya objek di alam semesta yang diketahui memiliki kehidupan. Bentuk Bumi hampir bulat, tapi kutub planet kita datar dan menggembung di sekitar ekuator.

Posisi: Planet ketiga dari Matahari

Jarak dari Matahari: 149,6 juta km

1 hari (periode rotasi): 23,93 jam

1 tahun (periode orbit): 365,26 hari

Massa:  $5,972 \times 10^{24}$  kg

Gravitasi:  $9,807 \text{ m/s}^2$

Suhu: -89,2 hingga  $56,9^\circ\text{C}$

Atmosfer: Oksigen ( $\text{O}_2$ )

Bulan yang diketahui: 1 (Bulan)

Cincin: 0

Titik tertinggi: Gunung Everest, Nepal-Tibet, Cina (8.848 m)

Titik terendah: Challenger Deep, Palung Mariana, Samudera Pasifik  
(10,994 m)

Gambar Bumi ditunjukkan pada gambar 2.7



**Gambar 2.7 Bumi**

(Sumber : [https://pngimg.com/uploads/earth/earth\\_PNG8.png](https://pngimg.com/uploads/earth/earth_PNG8.png))

#### **D. Mars**

Planet Mars dijuluki “Planet Merah” karena permukaannya dilingkupi besi oksida atau dalam kata lain, karat biasa, yang memberi warna kemerahan pada planet ini. Mars memiliki gunung terbesar dalam Sistem Tata Surya bernama Olympus Mons. Gunung tersebut merupakan gunung berapi yang tiga kali lebih tinggi daripada Gunung Everest. Para ilmuwan menemukan bukti keberadaan air garam di Mars. Mars memiliki badai debu paling kuat dan berbahaya di Sistem Tata Surya.

Posisi: Planet keempat dari Matahari

Jarak dari Matahari: 228 juta km

1 hari (periode rotasi): 24,63 jam

1 tahun (periode orbit): 686,98 hari Bumi

Massa: 0,10744 massa Bumi

Gravitasi: 0,38 gravitasi Bumi

Suhu: -153 hingga 20°C

Atmosfer: Karbon Dioksida (CO<sub>2</sub>)

Bulan yang diketahui: 2 (Fobos, Deimos)

Cincin: 0

Arti nama: Dewa perang Romawi

Titik tertinggi: Olympus Mons, Tharsis (27,3 km)

Titik terendah: Hellas Planitia (59,5 km)

Gambar planet Mars ditunjukkan pada gambar 2.8.



**Gambar 2.8 Mars**

(Sumber : [https://pngimg.com/uploads/mars\\_planet/mars\\_planet.PNG7.png](https://pngimg.com/uploads/mars_planet/mars_planet.PNG7.png))

### **E. Jupiter**

Kita dapat melihat Jupiter dengan mata telanjang karena planet ini merupakan objek yang paling bersinar setelah Venus dan Bulan dalam Sistem Tata Surya. The Great Red Spot merupakan badai antisiklon dahsyat di Jupiter, yang berlokasi di selatan ekuator Jupiter dan telah ada setidaknya dalam 350 tahun.

Posisi: Planet kelima dari Matahari

Jarak dari Matahari: 778,4 juta km

1 hari (periode rotasi): 9,92 jam

1 tahun (periode orbit): 4.330,6 hari Bumi (11 tahun Bumi)

Massa: 317,82 massa Bumi

Gravitasi: 24,79 m/s<sup>2</sup>

Suhu: 108,33<sup>o</sup>C

Atmosfer: Hidrogen (H), Helium (He)

Bulan yang diketahui: 79

Cincin: 1

Arti nama: Raja dewa-dewa Romawi

Gambar planet Jupiter ditunjukkan pada gambar 2.9.



**Gambar 2.9 Jupiter**

(Sumber : <https://img2.pngdownload.id/20180205/xze/kisspng-display-resolution-image-file-formats-jupiter-png-photos-5a77ecc8068c83.1264113015178088400268.jpg>)

## **F. Saturnus**

Saturnus berbentuk bola elips. Planet ini berputas sangat cepat pada sumbunya sehingga bagian kutubnya berbentuk pipih. Kepadatan Saturnus sangat rendah, sehingga kalau dimasukkan ke kolam raksasa, planet ini akan mengapung seperti apel.

Posisi: Planet keenam dari Matahari

Jarak dari Matahari: 1,4 miliar km

1 hari (periode rotasi): 10,656 jam

1 tahun (periode orbit): 110,775,7 hari Bumi (29,4 tahun Bumi)

Massa: 95,16 massa Bumi

Gravitasi:  $10,4 \text{ m/s}^2$

Suhu:  $-139^\circ\text{C}$

Atmosfer: Hidrogen (H), Helium (He)

Bulan yang diketahui: 62

Cincin: 7

Arti nama: Dewa pertanian Romawi

Gambar planet Saturnus ditunjukkan pada gambar 2.10.



**Gambar 2.10 Saturnus**

(Sumber : <https://www.pngarts.com/files/4/Saturn-PNG-High-Quality-Image.png>)

### **G. Uranus**

Uranus merupakan planet paling dingin di Sistem Tata Surya. Planet ini merupakan planet pertama yang ditemukan pada era modern dan satu-satunya planet yang namanya diambil dari nama pahlawan dalam mitologi Yunani (bukan Romawi). Uranus sangat berkaitan dengan Inggris. Planet ini ditemukan di Inggris oleh Sir William Herschel dan semua nama bulannya diambil dari nama karakter dalam karya-karya William Shakespeare dan Alexander Pope.

Posisi: Planet ketujuh dari Matahari

Jarak dari Matahari: 2,871 miliar km

1 hari (periode rotasi): 17,23 jam

1 tahun (periode orbit): 30.687 hari Bumi (84 tahun Bumi)

Massa: 14,371 massa Bumi

Gravitasi:  $8,87 \text{ m/s}^2$

Suhu:  $-180,6^\circ\text{C}$

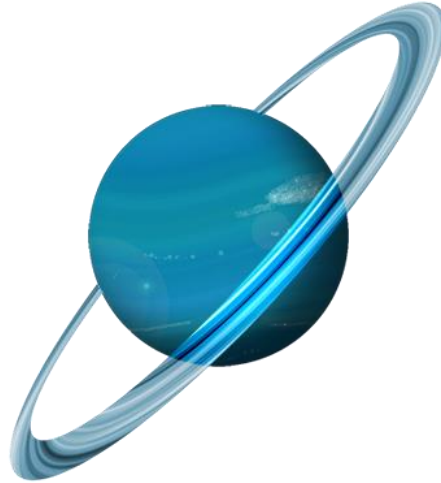
Atmosfer: Hidrogen (H), Helium (He)

Bulan yang diketahui: 27

Cincin: 13

Arti nama: Dewa langit Yunani

Gambar planet Uranus ditunjukkan pada gambar 2.11.



**Gambar 2.11 Uranus**

(Sumber : <https://w0.pngwave.com/png/536/869/rings-of-uranus-planet-natural-satellite-solar-system-planet-png-clip-art.png>)

## **H. Neptunus**

Neptunus tidak langsung diberi nama ketika ditemukan. Planet ini disebut “planet di luar Uranus” atau “planet Le Verrier” karena orang yang menemukannya adalah Urbain Le Verrier. Tetapi setelah mendengarkan banyak masukan terkait nama planet ini, Le Verrier memutuskan menyebutnya Neptunus.

Posisi: Planet kedelapan dari Matahari

Jarak dari Matahari: 4,5 miliar km

1 hari (periode rotasi): 16 jam

1 tahun (periode orbit): 60.190 hari Bumi (165 tahun Bumi)

Massa: 17,147 massa Bumi

Gravitasi: 11,15 m/s<sup>2</sup>

Suhu: -214<sup>o</sup>C

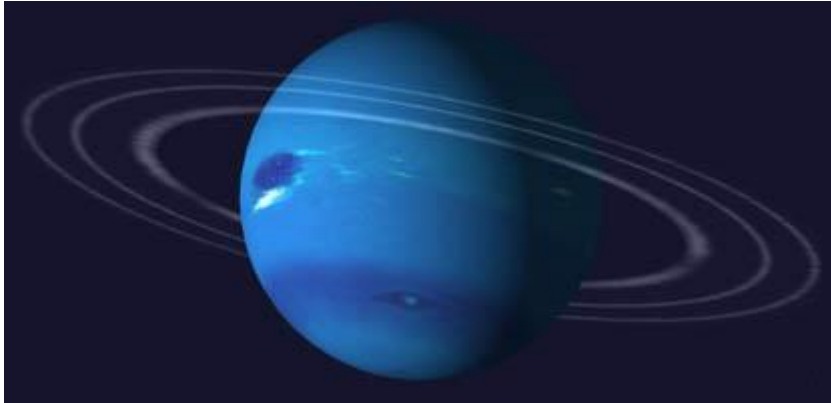
Atmosfer: Hidrogen (H), Helium (He)

Bulan yang diketahui: 14

Cincin: 6

Arti nama: Dewa laut Romawi.

Gambar planet Neptunus ditunjukkan pada gambar 2.12.



**Gambar 2.12 Neptunus**

(Sumber : <https://4.bp.blogspot.com/-HfNpSd9wsjY/U6l7Gz5bZhI/AAAAAAAAAAmo/USMPRvV9Ka0/s1600/Neptunus.png>)

## **I. Planet Kerdil**

Planet kerdil adalah benda langit yang mengitari Matahari, memiliki massa yang cukup untuk membentuk bentuk bulat yang hampir sempurna, tidak mengosongkan area di sekeliling orbitnya, dan bukan bulan.

Lima planet kerdil yang pertama kali dikenali adalah Ceres, Pluto, Eris, Makemake, dan Haumea seperti ditunjukkan pada gambar 2.13, gambar 2.14, gambar 2.15, gambar 2.16 dan gambar 2.17. Ilmuwan percaya masih ada puluhan atau bahkan lebih dari ratusan planet kerdil yang belum ditemukan.





**Gambar 2.13 Ceres**

(Sumber :

<https://icon2.cleanpng.com/20180518/xaw/kisspng-ceres-dwarf-planet-spacepedia-asteroid-belt-5afec4312626d2.5160006615266458091563.jpg>)



**Gambar 2.14 Pluto**

(Sumber :

<https://upload.wikimedia.org/wikipedia/commons/9/94/Pluto-transparent.png>)



**Gambar 2.15 Eris**

(Sumber : <https://nineplanets.org/wp-content/uploads/2019/09/eris.png>)



**Gambar 2.16 Makemake**

(Sumber : [https://vignette.wikia.nocookie.net/thesolarsystem6361/images/6/6d/Makemake\\_spacepedia.png/revision/latest/scale-to-width-down/340?cb=20180302130348](https://vignette.wikia.nocookie.net/thesolarsystem6361/images/6/6d/Makemake_spacepedia.png/revision/latest/scale-to-width-down/340?cb=20180302130348))



**Gambar 2.17 Haumea**

(Sumber :

[https://vignette.wikia.nocookie.net/  
thesolarsystem6361/images/4/  
48/Haumea\\_spacepedia.png/revision/  
latest/scale-to-width-  
down/340?cb=20180302130510\)](https://vignette.wikia.nocookie.net/thesolarsystem6361/images/4/48/Haumea_spacepedia.png/revision/latest/scale-to-width-down/340?cb=20180302130510)

## 2.7 Android

Android adalah sebuah sistem operasi perangkat mobile berbasis linux yang mencakup sistem operasi, middleware dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya *Google Inc* membeli *Android Inc* yang merupakan pendatang baru yang membuat peranti lunak untuk ponsel atau smartphone. Kemudian untuk dalam mengembangkan Android, dibentuk lah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Sebagaimana layaknya yang selalu berkembang. Android juga melalui banyak perubahan dari tujuan awal. Pada saat perilis perdana android, pada tanggal 5 november 2007, android bersama *Open Handset Alliance* menyatakan mendukung pengembangan open source pada perangkat mobile. Di lain pihak, Google merilis kode-kode android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan open platform perangkat seluler.

Kemudian aplikasi android terus dikembangkan, hingga sejak april 2009, versi android dikembangkan diawali dengan nama kode yang dinamai berdasarkan makanan pencuci mulut dan penganan manis. Masing-masing versi dirilis sesuai urutan alfabet, yakni Cupcake (1.5), Donut (1.6), (clair (2.0-2.1), Froyo (2.2-2.2.3), Gingerbread (2.3-2.3.7), Honeycomb (3.0-3.2.6), Ice Cream Sandwich (4.-4.0.4), Jelly Bean (4.1-4.3), Kitkat (4.4+), Lollipop (5.0+). Marshmallow (6.0+), Nougat (7.0+), Oreo (8.0+), dan yang terbaru adalah Pie (9.0+)[13].

Android merupakan sebuah sistem operasi terbuka (open source) yang mempunyai keasamaan dengan sistem operasi LINUX. Pertama kali dikembangkan oleh Android Inc dan mendapat bantuan dana oleh Google Sistem operasi ini dibuat khusus untuk menjadi sistem operasi pada telepon pintar (smartphone) dan komputer tablet. Pada tahun 2005 Android Inc diambil alih secara penuh oleh Google. Selanjutnya pada tahun 2007 Google mendirikan konsorsium perusahaan perangkat keras dan perangkat lunak serta telekomunikasi

untuk memajukan standar terbuka bagi produk perangkat keras. Sehingga pada awal oktober 2008 merupakan awal pertama kali telepon selular berbasis sistem operasi android diluncurkan kepada masyarakat Luas[14]. Gambar logo android ditunjukkan pada gambar 2.18.



**Gambar 2.18 Android**

(Sumber : <http://www.pngmart.com/files/4/Android-PNG-HD.png>)

## **2.8 Android Studio**

Android Studio adalah IDE (Integrated Development Environment) resmi untuk pengembangan aplikasi android dan bersifat open source atau gratis. Peluncuran Android Studio ini diumumkan oleh Google pada 16 Mei 2013 pada event Google I/O Conference untuk tahun 2013. Sejak saat itu, Android Studio menggantikan Eclipse sebagai IDE resmi untuk mengembangkan aplikasi android. Gambar android studio ditunjukkan pada gambar 2.19.



**Gambar 2.19 Android Studio**

Android Studio sendiri dikembangkan berdasarkan IntelliJ IDEA yang mirip dengan Eclipse disertai dengan ADT plugin (Android Development Tools).

Android studio memiliki fitur :

- a. Projek berbasis pada Gradle Build
- b. Refactory dan pembenahan bug yang cepat
- c. Tools baru bernama “Lint” diklaim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
- d. Mendukung *Proguard* and *App-signing* untuk keamanan.
- e. Memiliki GUI aplikasi android lebih mudah.
- f. Didukung oleh Google Cloud Platform untuk setiap aplikasi yang dikembangkan[13].

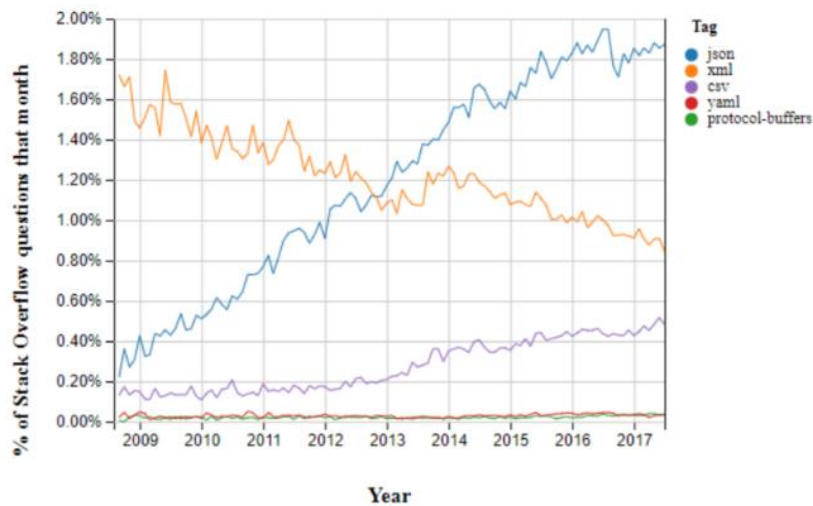
## **2.9 Java Development Kit (JDK)**

JDK merupakan perangkat lunak yang digunakan untuk melakukan proses kompilasi dari kode java ke bytecode yang dapat dimengerti dan dijalankan oleh JRE (Java Runtime Environment). Pada intinya kita berkomunikasi dengan android studio pake java, bukan pake bahasa lainnya. Apa yang ingin kita tuliskan dalam bahasa pemrograman java[15].

## **2.10 Java Script Object Nation (JSON)**

JSON adalah sebuah format data yang digunakan untuk pertukaran dan penyimpanan data. JSON merupakan bagian (subnet) dari javascript. Json bisa dibaca dengan berbagai macam bahasa pemrograman seperti C, C++, C#, Javascript Perl, Python, dan banyak lagi. Hal ini membuat JSON menjadi bahasa yang ideal untuk pertukaran data antar aplikasi.

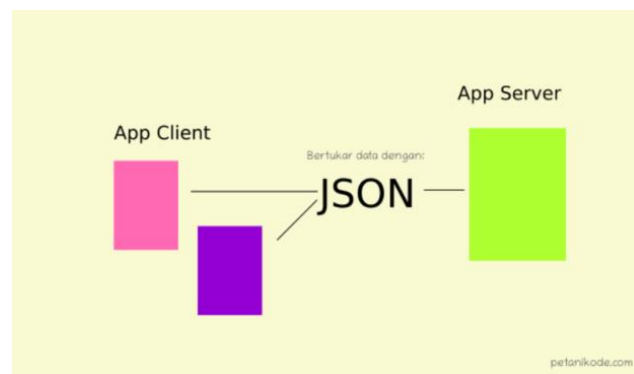
Berikut adalah grafik penggunaan JSON dibandingkan XML, YAML, CSV dan protocol buffers ditunjukkan pada Gambar 2.20.



**Gambar 2.20 Grafik Perbandingan JSON, XMP, YAML, CSV dan Protocol-buffers**

### Penerapan JSON dalam Pemrograman

JSON biasanya digunakan sebagai format standar untuk bertukar data antar aplikasi seperti pada Gambar 2.21.



**Gambar 2.21 Konsep Pertukaran Data Menggunakan JSON**

Tapi Sebenarnya tidak hanya itu saja, masih ada fungsi lain dari JSON. Berikut ini beberapa penerapan JSON yang pernah saya temui:

- JSON sebagai format untuk bertukar data client dan server atau aplikasi.  
Contoh: RESTful API;
- JSON sebagai tempat penyimpana data, contoh: Database Mongoddb;

- JSON digunakan untuk menyimpan konfigurasi project, contoh: file `composer.json` pada project PHP dan `package.json` pada Nodejs;
- JSON digunakan untuk menyimpan konfigurasi dan penyimpanan data pada Hugo;
- Json digunakan untuk menyimpan konfigurasi project pada Nodejs;
- JSON digunakan untuk menyimpan data manifest;
- dan masih banyak lagi[13].

### 2.11 Movavi Video Editing

Movavi Video Editing adalah tools yang membantu kita untuk mengedit video untuk berbagai macam keperluan. Karena ukurannya yang kecil dan ringan aplikasi ini tidak memiliki banyak fitur, tetapi tools ini juga memiliki beberapa fitur yang sudah bisa digunakan untuk edit video yang bervariasi untuk pengguna baru. Gambar movavi video editing ditunjukkan pada gambar 2.22.



**Gambar 2.22 Gambar Movavi Video Editing**

(Sumber : <https://is3-ssl.mzstatic.com/image/thumb/Purple118/v4/1c/0b/61/1c0b61e8-c2bb-b4d3-964c-71dc8bc87d75/icon.png/1200x630bb.png>)

### 2.12 Google Speech API

Google Speech API atau Google Voice Search diluncurkan pada tahun 2008 di Amerika Serikat untuk beberapa tipe smartphone. Google Speech API adalah sebuah framework yang dikembangkan oleh Google untuk mengenali

suara, mengubahnya menjadi string (text) dan memasukkannya ke dalam halaman pencarian Google sehingga akan tampil hasil pencarian berdasarkan input suara[6].

### **2.13 Pocket Sphinx**

Pocketsphinx merupakan library pengenalan ucapan versi mobile dari sistem Sphinx yang dirancang oleh Carnegie Mellon University. Proses pembelajaran unit-unit suara disebut training, sedangkan proses menggunakan pengetahuan yang diperoleh untuk menyimpulkan urutan yang paling mungkin dari unit dalam sinyal yang diberikan disebut decoding, atau secara sederhana disebut pengenalan(recognition). Karena terdapat dua proses tersebut maka diperlukan sphinx trainer untuk decoder. Pocketsphinx adalah library yang berkolerasi dengan library lain yaitu Sphinxbase yang menyediakan fungsionalitas umum untuk semua tools yang ada di CMUSphinxproject. Komponen utama untuk membangun pocketsphinx ini ada 3 yakni pocketsphinx, sphinxbase dan sphinxtrain[16].

1. Komponen untuk training
2. Komponen untuk decoding
3. Komponen sphinxbase

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

Perancangan arsitektur informasi dengan metode OOAD (Object Oriented Analysis and Design) menggunakan pendekatan terhadap masalah dari perspektif objek. Hingga saat ini, metode perancangan menggunakan OOAD menjadi tren dalam mengembangkan sistem informasi, pada OOAD, segala sesuatunya adalah objek serta sistem dipandang sebagai interaksi dari banyak objek. Berbagai perangkat CASE Tool berbasis objek pun terus dikembangkan untuk memenuhi kebutuhan membangun sistem informasi secara efisien.

Jika pada perancangan terstruktur kita mengenal beberapa diagram seperti konteks dan DFD, pada OOAD digunakan Unified Modeling Language



(UML). UML pada dasarnya adalah standar dalam pemodelan arsitektur perangkat lunak berbasis objek.

Sebenarnya, ada banyak spesialis informasi yang mendefinisikan pemodelan objek dalam mengembangkan aplikasi dan ada banyak diagram yang dipresentasikan dalam pemodelan objek. Hal ini akan sangat membingungkan dalam membangun sistem informasi karena terlalu banyak pemodelan objek.

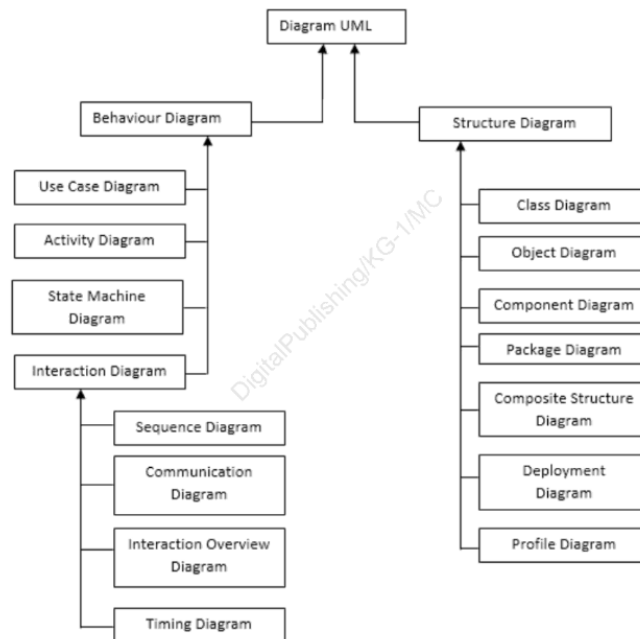
UML tercetus guna membakukan ragam model objek untuk desain perangkat lunak. UML dikembangkan oleh Grady Booch, Ivar Jacobson, dan James Rumbaugh pada tahun 1994. UML terus mengalami perkembangan. Pada tahun 1997, UML diadopsi sebagai standar oleh Object Management Group (OMG) dan Organisasi Internasional untuk Standarisasi (ISO) sebagai standar ISO. Sejak itu standarnya telah direvisi secara berkala.

Pada UML versi 2.2, didefinisikan dua jenis utama diagram UML, yakni diagram struktur (structured diagram) dan diagram perilaku (behavior diagram).

Diagram struktur memperlihatkan struktur statis dari sistem dan bagian-bagian sistem pada tingkat abstraksi dan tahap implementasi yang berbeda dan memperlihatkan struktur yang berkaitan satu sama lain. Unsur-unsur dalam diagram struktur mewakili konsep-konsep yang bermakna dari suatu sistem dan dapat mencakup konsep-konsep abstrak, dunia nyata, dan konsep implementasi.

Diagram perilaku menunjukkan perilaku dinamis dari objek dalam suatu sistem yang dapat digambarkan sebagai serangkaian perubahan pada sistem dari waktu ke waktu.

Diagram UML 2.2 dikategorikan secara hirarkis seperti ditunjukkan pada gambar 2.23.



**Gambar 2.23 Diagram UML**

UML berisi kumpulan diagram-diagram yang sudah memiliki standar untuk pembangunan perangkat lunak berbasis objek.

### 2.14.1 Struktur Diagram

**Class Diagram (Diagram Kelas)** adalah diagram struktur statis yang menggambarkan struktur sistem pada tingkat klasifikasi terendah, seperti kelas dan antarmuka. Diagram ini memperlihatkan pengklasifikasi sistem, atributnya, dan hubungan antara pengkelasan.

**Object Diagram (Diagram Objek)** memperlihatkan objek pengklasifikasi data link objek terasosiasi di antara objek.

**Package Diagram (Diagram Paket)** memperlihatkan paket dan depedensi antar paket.

**Component Diagram (Diagram Komponen)** menunjukkan komponen dan ketergantungan di antara komponennya. Diagram jenis ini digunakan dalam Component-Based Development (CBD).

**Composite Structure Diagram (Diagram Komposit)** digunakan untuk memperlihatkan struktur internal pengkelasan dan perilaku dalam kolaborasi.

Diagram Struktur Internal menunjukkan struktur internal dari suatu classifier – dekomposer dari classifier ke dalam properti, bagian dan keterhubungannya. Collaboration Use diagram memperlihatkan objek suatu sistem yang bekerja sama satu sama lain untuk menghasilkan beberapa perilaku sistem.

**Deployment Diagram** memperlihatkan arsitektur eksekusi dari suatu sistem yang merepresentasikan penugasan (men-deployment) artefak perangkat lunak ke sasaran penugasan (deployment), umumnya ke node.

**Profile Diagram (Diagram Profil)** pertama kali diperkenalkan pada UML versi 2.0. Diagram profil merupakan diagram UML pelengkap yang memungkinkan untuk mendefinisikan stereotip khusus, nilai yang ditandai (tagged values), dan batasan (constraints). Mekanisme Profil telah didefinisikan dalam UML, untuk menyediakan mekanisme standar UML. Profil diagram memungkinkan untuk menyesuaikan metamodel UML untuk berbagai platform dan domain. Contoh platform (seperti J2EE atau .NET), atau domain (seperti pemodelan real-time atau pemodelan proses bisnis).

#### 2.14.2 Behavior Diagram

Kategorisasi lain dari diagram UML adalah Diagram Perilaku (Behavior Diagram) yang memperlihatkan perilaku dinamis dari objek-objek dalam sistem yang dapat digambarkan sebagai serangkaian perubahan pada sistem dari waktu ke waktu.

**Use Case Diagram** adalah perilaku yang digunakan untuk menggambarkan serangkaian tindakan (use cases) bahwa beberapa sistem atau sistem (subjek) harus atau dapat melakukan kolaborasi dengan satu atau lebih user eksternal dari sistem (aktor) untuk memberikan hasil yang dapat diamati pada aktor-aktor atau pengampu kepentingan dalam sistem.

**Activity Diagram** (diagram aktivitas) memperlihatkan urutan dan kondisi untuk mengkoordinasikan perilaku tingkat yang lebih rendah daripada penggolongan yang memiliki perilaku tersebut. ini biasanya disebut aliran kontrol (control flow) dan aliran objek (object flow).

**State Machine** diagram digunakan untuk pemodelan perilaku diskrit melalui transisi status berhingga. Selain mengekspresikan perilaku dari sistem, State Machine juga dapat digunakan untuk mengekspresikan usage protocol pada sistem.

**Interaction Diagrams** (diagram interaksi) mencakup beberapa tipe diagram yakni :

1. **Sequence Diagram:** merupakan jenis Diagram Interaksi yang paling umum yang berfokus pada pertukaran pesan antara sejumlah Lifelines.
2. **Interaction Overview Diagram:** mendefinisikan interaksi melalui varian diagram aktivitas sehubungan aliran kontrol (control flow). Interaction Overview Diagram berfokus pada aliran kontrol secara umum, dalam hal ini Interaksi atau User Interaksi. Lifelines dan Messages tidak ada pada diagram ini.
3. **Communicatuin Diagram (Collaboration Diagram):** adalah Diagram Interaksi yang berfokus pada interaksi antara Lifelines. Urutan pesan diberikan melalui skema penomoran berurutan.
4. **Timing Diagram:** digunakan untuk memperlihatkan interaksi berbasis waktu. Timing diagram fokus pada perubahan kondisi pada dan di antara Lifelines sepanjang waktu linier.

Konsep penting lainnya dari perancangan berbasis objek adalah karakteristik dari objek itu sendiri, yakni Encapsulation, Inheritance, dan Polymorphism.

**Pembungkusan (Encapsulation):** Encapsulation ditunjukkan untuk membatasi akses program terhadap data yang diproses. Pembungkusan dilakukan terhadap data dan prosedur atau fungsi ke dalam suatu objek. Hal ini menjadi pembatas (encapsulate) di mana prosedur atau fungsi lain yang tidak termasuk di dalamnya tidak dapat mengakses komponen data, dungsi, atau prosedur yang di-encapsulate.

**Pewarisan (Inheritance):** pewarisan merupakan karakteristik di mana anak (child) dari objek akan mewarisi atribut dan metode dari induknya, atau dengan kata lain atribut dan metode suatu objek anak (child) diturunkan dari objek induk

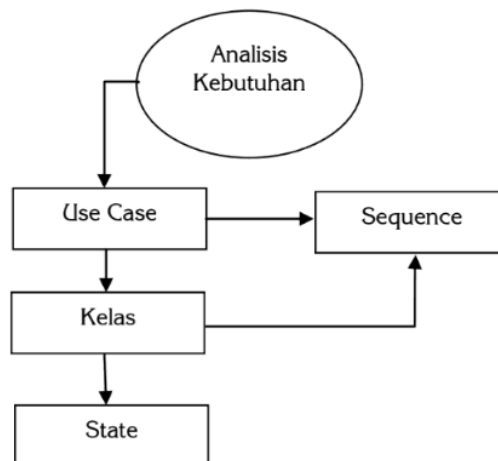
(parent). Pewarisan ini memperlihatkan bahwa atribut dan operasi yang dimiliki suatu kelas (class) mempunyai hubungan secara hirarki.

**Berubah Bentuk atau Banyak Bentuk (polymorphism):** diambil dari rangkaian kata poly dan morph. Poly berarti banyak dan morph berarti bentuk. Polimorfisme merupakan konsep yang menyatakan bahwa kelas-kelas yang sama yang terdiri atas objek-objek dapat mempunyai bentuk dan perilaku berbeda, sewaktu melakukan metode tertentu.

Beberapa konsep ini menjadi ciri dari pemrograman berorientasi objek yang menjadi dasar dalam pemodelan berbasis objek. Objek yang dimodelkan umumnya akan ditransiasikan ke dalam bahasa pemrograman berbasis objek. Beberapa model dapat ditransformasikan langsung menjadi kode pemrograman berbasis objek dan hal ini umumnya dilakukan dengan tool pemodelan.

UML sebetulnyamemiliki banyak diagram selain 4 yang disebutkan sebelumnya, tetapi sebagian besar jarang dibuat oleh pengembang aplikasi. Keempat diagram di atas merupakan diagram inti dari UML. Penggunaan diagram selain empat yang utama, ditunjukkan untuk mendapatkan gambaran arsitektur sistem informasi dengan sudut pandang berbeda. Hal ini terpaut pada faktor kebutuhan.

Penting untuk dipahami, sewaktu spesialis informasi memilih membuat pemodelan arsitektur dengan pendekatan terstruktur, maka pemodelan arsitektur berbasis objek sebagian atau seluruhnya tidak digunakan, hal ini berlaku sebaliknya. Pada gambar 2.24 menunjukkan urutan pembuatan diagram UML.





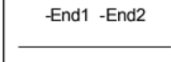
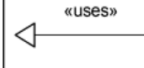
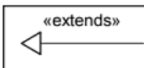
## Gambar 2.24 Urutan Pembuatan Diagram UML

### 2.14.3 Diagram Use Case

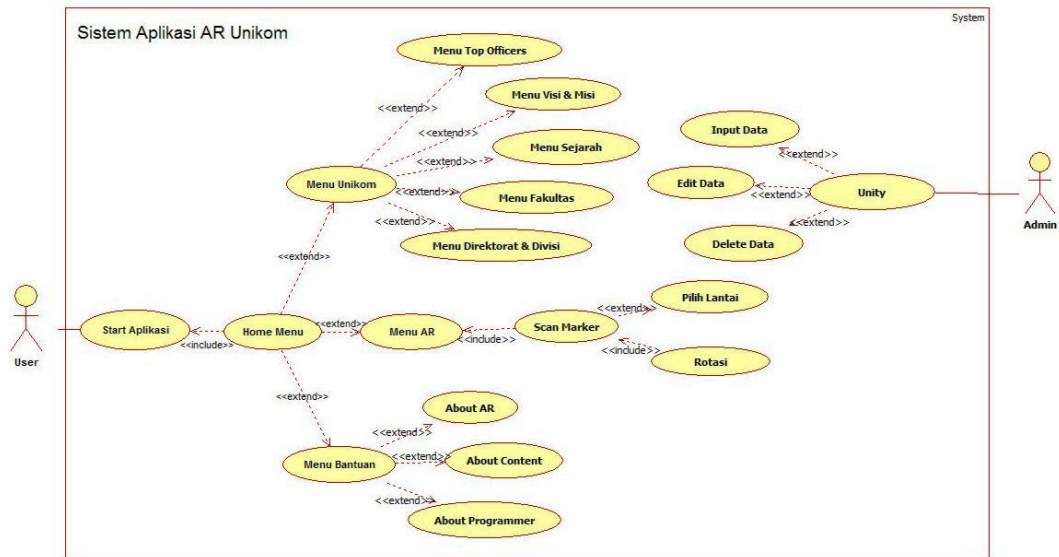
Diagram use case merupakan diagram yang harus dibuat pertama kali saat pemodelan perangkat lunak berorientasi objek dilakukan. Diagram use case akan menggambarkan apa yang dikerjakan oleh aktor. Yang disebut aktr di sini adalah pengguna aplikasi. Sama seperti pembangunan perangkat lunak terstruktur saat membuat DFD, untuk menggambar diagram use case mangacu pada proses sebelumnya, yaitu analisis kebutuhan pada RPL.

Simbol yang digunakan untuk membuat diagram use case ini seperti pada gambar 2.25 :

**Gambar 2.25 Simbol Use Case Diagram**

Simbol	Nama	Keterangan
	Aktor	Merupakan pengguna dari sistem. Penamaan aktor menggunakan kata benda.
	Use case	Merupakan pekerjaan yang dilakukan oleh aktor. Penamaan use case dengan kata kerja.
	Asosiasi	Hubungan antara aktor dengan use case.
	Include	Hubungan antara use case dengan use case. Include menyatakan bahwa sebelum pekerjaan dilakukan harus mengerjakan pekerjaan lain terlebih dahulu.
	Extends	Hubungan antara use case dengan use case. Extends menyatakan bahwa jika pekerjaan yang dilakukan tidak sesuai atau terdapat kondisi khusus, maka lakukan pekerjaan itu.

Sebagai contoh, di sini digunakan contoh pengenalan gedung baru UNIKOM. Use case diagram ditunjukkan seperti pada gambar 2.26 :



**Gambar 2.26 Diagram Use Case Pengenalan Gedung Baru Unikom**

#### 2.14.4 Diagram Kelas

Diagram kelas dibuat setelah diagram use case dibuat terlebih dahulu. Pada pembuatan diagram ini harus dijelaskan hubungan apa saja yang terjadi antara suatu objek dengan objek lainnya sehingga terbentuklah suatu sistem aplikasi.

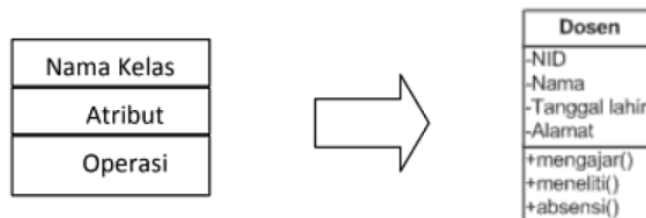
Kelas adalah rancangan dari suatu objek, sebagai contoh kelas mobil merupakan rancangan dari objek mobil sedan, pick up, mini bus, truk, dan lain-lain. Jadi di sini kelas mobil adalah rancangan saja, belum menjadi objek, belum bisa dikendarai, dan belum bisa dioperasikan. Sebaliknya jika rancangan mobil tersebut sudah menjadi objek nyata, mobil sedan misalnya, objek tersebut barulah dapat dikendarai dan dioperasikan.

Pembuatan diagram kelas dibagi menjadi dua, yaitu kelas itu sendiri dan relasi antar kelas. Kelas dibagi menjadi 3 bagian :

**Nama Kelas** adalah nama dari kelas itu sendiri, misalnya kelas mobil, kelas dosen, kelas mahasiswa, dan lain-lain. Penamaan kelas menggunakan kata benda.

**Atribut** adalah data yang dimiliki oleh kelas tersebut, misalnya kelas mahasiswa memiliki atribut NIM nama, tanggal lahir, alamat, jenis kelamin, dan sebagainya.

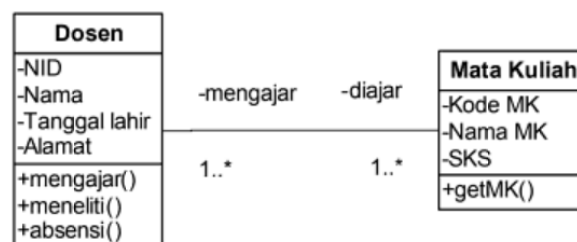
**Operasi Kelas (methods)** menunjukkan apa yang kelas tersebut bisa lakukan, misalnya kelas dosen dapat melakukan operasi mengajar, absensi, penelitian, dan lain-lain. Contoh kelas ditunjukkan pada gambar 2.27



**Gambar 2.27 Contoh Kelas**

**Catatan :** Diagram kelas berbeda dengan Entity Relationship Diagram (ERD). Meskipun menggunakan OOAD, dalam bentuk banyak kasus ERD dibuat jika ditujukan untuk pemodelan basis data dan komponen diagram kelas dapat dijadikan dasar membuat ERD.

Setelah membuat kelas, selanjutnya membuat relasi antar-kelas. Misalnya dosen mengajar mata kuliah, dalam hal ini dosen dan mata kuliah adalah kelas, mengajar adalah relasi antar kelas. Berikut contoh diagram kelasnya ditunjukkan pada gambar 2.28:



**Gambar 2.28 Diagram Kelas**

Relasi memiliki multiplicity, misalnya satu dosen dapat mengajar banyak mata kuliah dan satu mata kuliah adapat diajar oleh banyak dosen, berarti multiplicity-nya adalah banyak ke banyak.

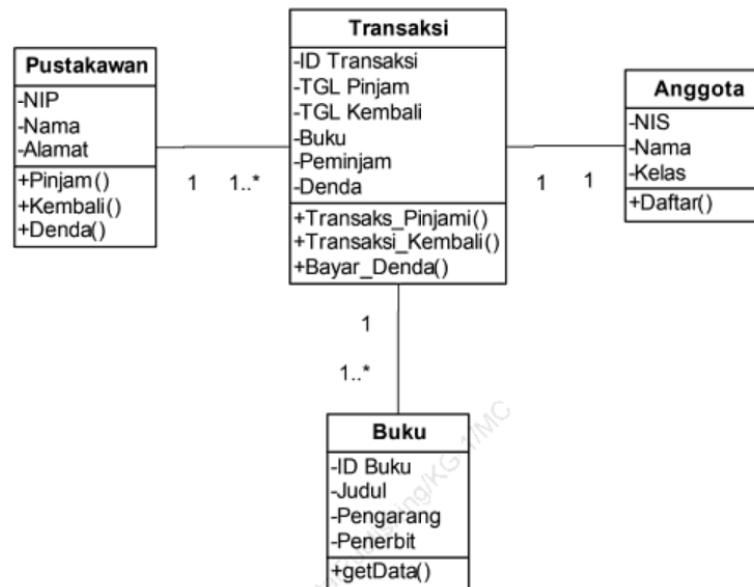
**Catatan :** Sedikit berbeda dengan derajat relasi pada ERD, dalam menggambar kelas, multiplicity dijelaskan oleh gambar 2.29.



**Gambar 2.29 Simbol Multiplicity**

Indikator	Arti	Penjelasan
0..1	Kosong ke satu	Dalam hal ini objek yang berinteraksi dengan objek dari kelas lainnya hanya 1 atau tidak ada sama sekali
1	Hanya satu	Objek yang berinteraksi dengan objek dari kelas lainnya hanya 1
0..*	Kosong ke banyak	Objek yang berinteraksi dengan objek dari kelas lainnya minimal 1 atau tidak sama sekali
1..*	Satu ke banyak	Objek yang berinteraksi dengan objek dari kelas lainnya minimal 1 atau lebih
N	Banyak (n) di mana n lebih besar dari 1	Objek yang berinteraksi dengan objek dari kelas lainnya berjumlah n

Bila melanjutkan pembuatan model dari apa yang sudah dibuat pada use case, yaitu perpustakaan SMA XYZ, diagram kelas yang dapat dibuat seperti gambar 2.30 :



**Gambar 2.30 Diagram Kelas Perpustakaan SMA XYZ**

#### 2.14.5 Diagram Sequence

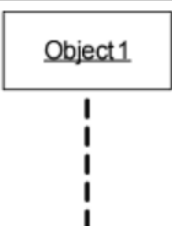

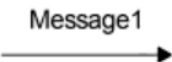
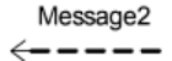
Diagram sequence adalah diagram yang dibuat untuk mengetahui alur dari interaksi antar-objek. Isi dari diagram sequence harus sama dengan use case dan

diagram kelas. Satu use case tunggal akan digambarkan satu diagram sequence-nya.

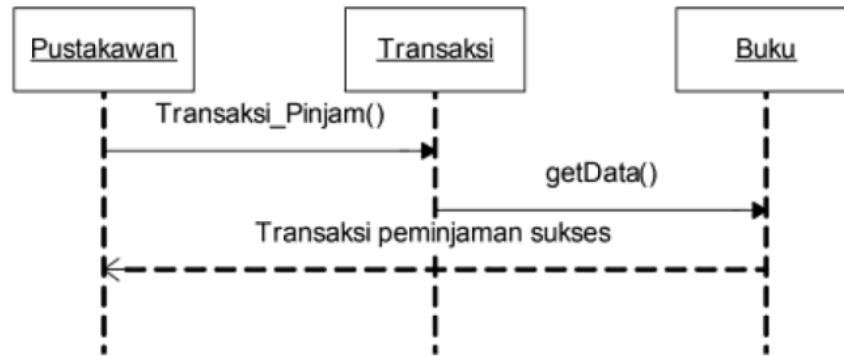
Sebagai contoh, dari use case aplikasi perpustakaan SMA XYZ, di sini terdapat 4 use case tunggal, yaitu perpustakaan dengan peminjaman, pustakawan dengan pengembalian, pustakawan dengan denda, dan anggota dengan pendaftaran. Karena ada 4 use case tunggal, maka nantinya akan digambar 4 diagram sequence.

Penjelasan tentang simbol sequence diagram dijelaskan pada gambar 2.31.

**Gambar 2.31 Simbol Sequence Diagram**

Simbol	Nama	Keterangan
	Objek/aktor	Sebuah objek yang berasal dari kelas. Atau dapat dinamai dengan kelasnya saja. Aktor termasuk objek. Garis putus-putus menunjukkan garis hidup suatu objek.
	Aktivasi	Menunjukkan masa hidup dari objek.
	Pesan	Interaksi antara satu objek dengan objek lainnya. Objek dapat mengirimkan pesan ke objek lain. Interaksi antar-objek ditunjukkan pada bagian operasi pada diagram kelas.
	Return	Pesan kembalian dari objek yang diajak berkomunikasi.

Contoh menggambar diagram sequence pada use case tunggal pustakawan dengan usecase peminjaman seperti pada gambar 2.32 :



**Gambar 2.32 Diagram Sequence Pada Use Case Perpustakaan Dengan Peminjaman**

Dapat dilihat dari diagram di atas bahwa pustakawan memanggil operasi `Transaksi_Pinjam()` dari kelas transaksi dan kelas transaksi memanggil operasi `getData()` pada kelas buku, di mana operasi `Transaksi_Pinjam()` dimiliki oleh kelas transaksi pada diagram kelas dan operasi `getData()` dimiliki oleh kelas buku pada diagram kelas buku.

#### 2.14.6 Diagram State

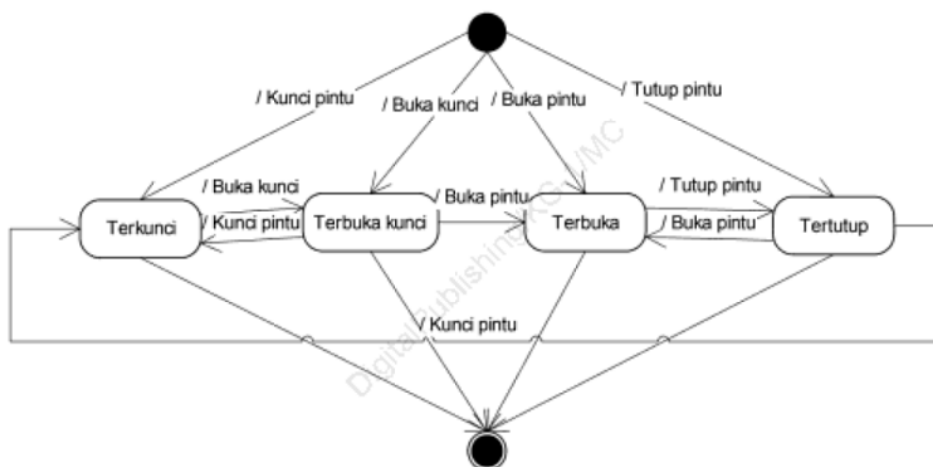
Diagram state menggambarkan perilaku dari objek. Saat menggambar diagram state, diagram kelas harus sudah dibuat terlebih dahulu. Karena objek yang dibuat dari kelas yang sama akan memiliki perilaku yang sama, maka jumlah diagram state yang dibuat harus sama dengan jumlah kelas yang dibuat. Berdasarkan contoh gambar 2.30, pada aplikasi perpustakaan di SMA XYZ terdapat 4 buah kelas, yaitu kelas pustakawan, transaksi, anggota, dan buku, sehingga ada 4 buah diagram state yang harus dibuat. Penjelasan Simbol Diagram State ditunjukkan pada gambar 2.33.

**Gambar 2.33 Simbol Diagram State**

Simbol	Nama	Keterangan
●	State awal	Merupakan simbol dari awal kondisi suatu objek
●	State akhir	Simbol yang menunjukkan akhir dari suatu objek
State1	State	Status perubahan kondisi objek setelah diberlakukan transisi
→	Transisi	Aksi untuk mengubah status objek

Guna lebih memahami mekanisme diagram state, perhatikan contoh berikut :

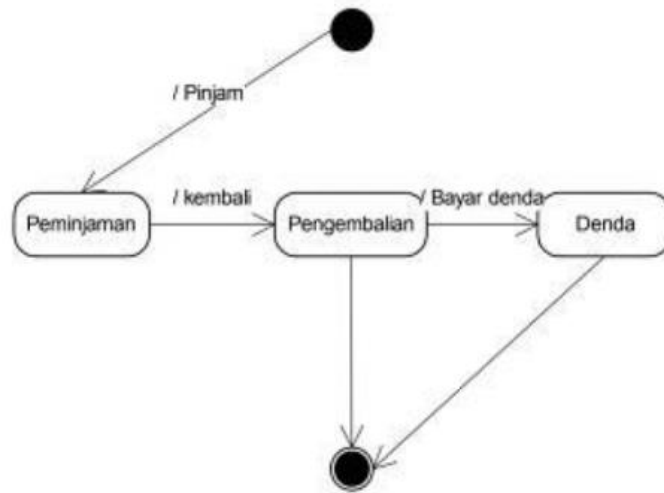
Misalnya objek yang diketahui adalah pintu yang berasal dari kelas pintu. Seperti yang telah diketahui, pintu tersebut memiliki kondisi status bilamana diberi aksi, misalnya pintu akan memiliki status terkunci bila dilakukan aksi kunci d=pintu, statusnya akan terbuka kunci bila aksi yang dilakukan adalah buka kunci, setatusnya akan menjadi terbuka jika aksi yang dilakukan adalah buka pintu, dan statusnya akan berubah lagi menjadi tertutup jika aksi yang dilakukan adalah tutup pintu. Skenario ini bila digambar ke dalam state diagram seperti pada gambar 2.34:



**Gambar 2.34 Diagram State Dari Objek/Kelas Pintu**

Meneruskan contoh yang sebelumnya, ambil satu kelas dari aplikasi perpustakaan di SMA XYZ, misalnya kelas transaksi. Kelas ini memiliki 3

layanan yaitu peminjaman, pengembalian, dan denda. Kelas ini akan memiliki status perminjaman jika aksinya adalah pinjam, kana memiliki status pengembalian jika aksinya adalah kembali, dan akan menjadi status denda jika aksinya adalah bayar denda. Gambar 2.35 menunjukkan diagram state.



**Gambar 2.35 Diagram State Dari Objek/Kelas Transaksi**

[17]

### 2.15 Orientasi Objek

Di dalam kamus orientasi adalah diarahkan menuju. Sedangkan objek adalah kumpulan data dan asosiasi perilaku. Orientasi Objek adalah fungsi yang diarahkan ke memodelkan objek. Orientasi Objek adalah salah satu teknik yang digunakan untuk sistem yang kompleks. Itu didefinisikan dengan menggambarkan kumpulan interaksi objek melalui data dan perilaku[18].

#### **Kelas**

Kelas merupakan kumpulan atas definisi data dan fungsi-fungsi dalam sutau unit untuk tujuan tertentu. Sebagai contoh 'kelas dari kucing' adalah suatu unit yang terdiri atas definisi-definisi data dan fungsi-fungsi yang menunjuk pada berbagai macam perilaku dari kucing.

#### **Objek**

Membungkus data dan fungsi bersama menjadi suatu unit dalam sebuah program komputer: objek merupakan dasar dari modularitas dan struktur dalam sebuah program komputer berorientasi objek.

### **Abstraksi**

Kemampuan sebuah program untuk melewati aspek informasi yang akan diproses olehnya, yaitu kemampuan untuk memfokuskan pada inti. Setiap objek dalam sistem melayani sebagai model dari ‘pelaku’ abstrak yang dapat melakukan kerja, laporan, perubahan keadaannya dan berkomunikasi dengan objek lainnya dalam sistem, tanpa mengungkapkan kelebihan ini diterapkan. Proses, fungsi atau metode dapat juga dibuat abstrak dan beberapa teknik digunakan untuk mengembangkan sebuah pengabstrakan.

### **Enkapsulasi**

Memastikan pengguna sebuah objek tidak dapat mengganti keadaan dalam dari sebuah objek dengan cara yang tidak layak: hanya metode dalam objek tersebut yang diberi izin untuk mengakses keadaannya. Setiap objek mengakses interface yang menyebutkan bagaimana objek lainnya dapat berinteraksi dengannya. Objek lainnya tidak akan mengetahui dan tergantung kepada representasi dalam objek tersebut.

## **2.16 HTML**

HTML adalah Hyper Text Markup Language yang artinya adalah sebuah text berbentuk link dan mungkin juga foto atau gambar yang saat itu di klik, akan membawa si pengakses internet dari satu dokumen ke dokumen lainnya. Dalam praktiknya, Hypertext berwujud sebuah link yang bisa mengantar anda ke dunia internet yang sangat luas. Untuk membantu si pengakses berpindah dari satu tempat ke tempat lainnya, dibuatlah semacam dokumen yang nanti akan disebut dengan istilah website. Untuk membuat website, kita membutuhkan Markup, yaitu Tag (semacam kode) yang mengatur bagaimana website tersebut akan ditampilkan di jendela browser, seperti layout dan tampilan-tampilan visual yang biasa kita lihat di dalam sebuah website. Terakhir, HTML adalah semacam bahasa yang ditunjukkan oleh kata Language merupakan petunjuk bahwa HTML adalah

semacam script atau bahasa pemrograman[19]. Halaman-halaman di WWW disusun dengan menggunakan bahasa yang disebut HTML. Bahasa ini mampu memberikan perintah pada browser untuk menampilkan teks, gambar, grafik, suara, dan elemen weblainnya. Halaman-halaman Web dihubungkan satu dengan lainnya menggunakan apa yang disebut dengan “Hypertext”. Menggunakan hypertext kita dapat berpindah dari halaman ke halaman lain baik secara berurutan maupun secara acak[20].

### **2.17 PHP**

PHP adalah singkatan dari PHP Hypertext Preprocessing, merupakan bahasa scripting untuk web yang cukup populer. Dengan PHP, bisa membuat web dinamis dimana kode PHP diselipkan diantara script kode-kode HTML yang merupakan bahasa markup standar untuk dunia web.

Untuk membedakan kode PHP dan kode HTML sebagai wadahnya, digunakan tag-tag di PHP. PHP dapat dipakai untuk memprogram situs web dinamis tipe apapun, bahkan PHP dapat digunakan untuk membangun CMS.

PHP adalah bahasa scripting server dan merupakan tool yang kuat untuk membuat webpage yang dinamis dan interaktif. PHP banyak digunakan dan merupakan alternatif untuk menggantikan bahasa pemrograman lain, seperti ASP dari Microsoft. Logo PHP ditunjukkan pada gambar 2.36[21].



**Gambar 2.36 Logo PHP**

## 2.18 MySQL

MySQL adalah sistem database yang paling populer karena hampir sebagian besar aplikasi berbasis website seperti Wordpress, dilengkapi dengan MySQL. Selain itu, MySQL juga ditawarkan dalam berbagai versi, termasuk versi yang gratis[22]. Database sendiri memiliki arti kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak untuk menghasilkan informasi.


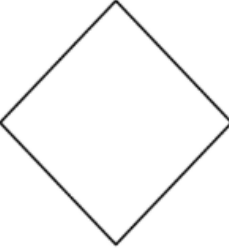




## 2.19 Flowchart

Para ahli telah mendefinisikan flowchart menurut pandangannya masing-masing diantaranya yang paling terkenal adalah sebagai berikut :

1. Pahlevy (2010) menyatakan bahwa flowchart (bagian alir) merupakan sebuah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program yang menyatakan arah alur program.
2. Krismaji (2010) menyebutkan bahwa bagan alir merupakan teknik analitis yang digunakan untuk menjelaskan aspek-aspek sistem informasi secara jelas, tepat dan logis.
3. Indrajani (2011) mengatakan bahwa flowchart merupakan gambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program.
4. Sariadin Siallagan (2009) berpendapat bahwa flowchart adalah bagan atau suatu diagram alir yang mempergunakan simbol atau tanda untuk menyelesaikan suatu masalah.
5. Sterneckert (2003) mengatakan flowchar atau diagram alir merupakan sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran algoritma atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing-masing langkah tersebut dengan menggunakan tanda panah. Diagram ini bisa memberi solusi selangkah demi selangkah untuk penyelesaian masalah yang ada di dalam proses atau algoritma tersebut.



Pada gambar 2.37 adalah simbol-simbol flowchart dan penjelasannya.

Gambar	Simbol untuk...	Keterangan
	Proses/Langkah	Menyatakan kegiatan yang akan ditampilkan dalam diagram alir.
	Titik Keputusan	Proses/langkah di mana perlu adanya keputusan atau adanya kondisi tertentu. Di titik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
	Masukan/Keluaran Data	Digunakan untuk mewakili data masuk, atau data keluar.
	Terminasi	Menunjukkan awal atau akhir sebuah proses.
	Garis alir	Menunjukkan arah aliran proses atau algoritma.
	Kontrol/Inspeksi	Menunjukkan proses/langkah di mana ada inspeksi atau pengontrolan.

**Gambar 2.37 Simbol Flowchart**

## 2.20 Frame Piramid

Frame piramid merupakan alat atau perangkat keras yang digunakan untuk memproyeksikan 3D hologram dari smartphone yang digunakan sebagai proyektor. Frame piramid ini dapat dimiliki dengan membeli secara online atau membuat sendiri menggunakan bahan kaca akrilik.