

## BAB III

### LANDASAN TEORI

#### 3.1 Aplikasi

##### 3.1.1 Pengertian Aplikasi

Menurut Rachmad Hakim S, Aplikasi adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur *Windows &*, permainan (*Game*), dan sebagainya. Menurut Harip Santoso, Aplikasi adalah suatu kelompok file (*Form, Class, Report*) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi *payroll*, aplikasi *fixed asset*, dan lain-lain. (2015).



Gambar 3.1. Contoh Aplikasi

Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket atau *suite* aplikasi (*Application Suite*). Contohnya adalah *Microsoft Office* dan *OpenOffice.org*, Bahasa Pemrograman yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya. Aplikasi – aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Berdasarkan jenisnya, aplikasi – aplikasi komputer dapat dibagi menjadi beberapa kategori, yaitu:

a. *Enterprise*

Digunakan untuk organisasi yang cukup besar dengan maksud menghubungkan aliran data dan kebutuhan informasi antar bagian, contoh : *IT Helpdesk, Travel Management* dan lain-lain.

b. *Enterprise-Support*

Sebagai aplikasi pendukung dari Enterprise, contohnya : *Database Management, Email Server* dan *Networking System*.

c. Individual Worker

Sebagai aplikasi yang biasa digunakan untuk mengolah/edit data oleh tiap individu. Contoh : *Ms.Office, Photoshop, Acrobat Reader* dan lain-lain.

d. Aplikasi Akses Konten

Adalah aplikasi yang digunakan oleh individu (hanya) untuk mengakses konten tanpa kemampuan untuk mengolah atau mengedit datanya melainkan hanya melakukan kustomisasi terbatas. Contoh : *Games, Media Player, Web Browser*.

e. Aplikasi Pendidikan

Biasanya berbentuk simulasi dan mengandung konten yang spesifik untuk pembelajaran.

### 3.2 Budaya

Kata budaya itu sendiri adalah sebagai suatu perkembangan dari bahasa sansekerta “buddhayah” yaitu bentuk jamak dari “buddhi” atau akal, dan kata majemuk budi-daya, yang berarti daya dari budi, dengan kata lain ”budaya adalah daya dari budi yang berupa cipta, karsa dan rasa. Sedangkan kebudayaan merupakan pengembangan dari budaya yaitu hasil dari cipta, karsa dan rasa tersebut”.

Kebudayaan berasal dari kata “budaya”. Budaya diserap dari bahasa Sanskerta “buddhayah” yaitu bentuk jamak dari “buddhi“ yang berarti budi atau akal. Dengan demikian kebudayaan dapat diartikan “hal-hal yang bersangkutan dengan akal”.

Unsur - Unsur Kebudayaan Universal dapat dijelaskan sebagai berikut :

### 3.2.1 Peralatan dan Perlengkapan Hidup

Peralatan dan perlengkapan hidup merupakan semua sarana dan prasarana yang digunakan oleh manusia atau masyarakat dalam setiap proses kehidupan terutama dalam rangka memenuhi kebutuhan hidup. Teknologi merupakan cara atau teknik memproduksi, memakai, serta memelihara segala peralatan dan perlengkapan. Teknologi yang berkembang di masyarakat dan berfungsi sebagai peralatan dan perlengkapan hidup contohnya seperti alat-alat produktif, senjata, wadah, alat-alat menyalakan api, makanan, pakaian, tempat berlindung, serta alat-alat transportasi.

### 3.2.2 Sistem Mata Pencaharian Hidup (Sistem Ekonomi)

Sistem mata pencaharian hidup yang termasuk dalam unsur budaya universal terfokus pada mata pencaharian masyarakat tradisional contohnya seperti berburu, meramu, beternak, bercocok tanam, menangkap ikan.

### 3.2.3 Sistem Keekerabatan dan Organisasi Sosial

Sistem kekerabatan merupakan bagian yang sangat penting dalam struktur sosial. Sistem kekerabatan adalah sistem menghitung garis keturunan atas dasar hubungan perkawinan dan hubungan darah. Dapat pula disebutkan bahwa kekerabatan adalah unit-unit sosial yang terdiri dari beberapa keluarga yang memiliki hubungan darah dan hubungan perkawinan. Ada beberapa system kekerabatan yang dimiliki/dijalani oleh masyarakat di Indonesia, yaitu :

#### a. Sistem Keekerabatan *Bilateral*

Sistem Keekerabatan *Bilateral*, adalah system kekerabatan yang menghitung garis keturunan dari dua pihak, yaitu dari pihak ayah dan ibu secara seimbang atau bersama-sama

#### b. Sistem Keekerabatan *Unilateral*

Sistem kekerabatan *Unilateral*, adalah system kekerabatan yang menghitung garis keturunan dari satu pihak, yaitu dari pihak ibu saja yang disebut *system matrilineal* atau dari pihak ayah saja yang disebut *system patrilineal*.

#### c. Sistem Keekerabatan *Ambilineal*

Sistem Keekerabatan *Ambilineal*, adalah system kekerabatan yang menghitung garis keturunan dari pihak ayah dan pihak ibu secara bergantian, atau bisa dikatakan

menghitung garis keturunan sebagian dari pihak ayah sebagian dari pihak ibu. Organisasi sosial adalah perkumpulan sosial yang dibentuk masyarakat baik formal maupun *non* formal (berbadan hukum maupun tidak berbadan hukum). Manusia sebagai makhluk sosial akan selalu hidup bersama dengan orang lain untuk dapat memenuhi kebutuhannya. Untuk mencapai tujuan-tujuan hidup tertentu yang tidak dapat dicapai sendiri, manusia bersama-sama dengan manusia lain dalam masyarakat akan membentuk perkumpulan atau organisasi sosial. Berdasarkan bidang kegiatannya, organisasi sosial di masyarakat dibedakan menjadi :

- 1) Organisasi Sosial di bidang Pendidikan, misalnya sekolah, lembaga pelatihan, LPK, dll.
- 2) Organisasi Sosial di bidang Kesejahteraan Sosial, misalnya Panti Asuhan, Panti Jompo, dan sebagainya.
- 3) Organisasi Sosial di bidang Kesehatan, misalnya Rumah Sakit, Balai Pengobatan.
- 4) Organisasi Sosial di bidang Keadilan, misalnya LBH.

### **3.2.4 Bahasa**

Bahasa merupakan wujud budaya yang digunakan manusia untuk saling berkomunikasi atau berinteraksi, baik secara lisan, tulisan maupun bahasa isyarat. Secara umum bahasa berfungsi sebagai alat berekspresi, alat komunikasi, serta alat untuk mengadakan integrasi dan adaptasi sosial. Sedangkan secara khusus bahasa berfungsi untuk mengadakan hubungan dalam pergaulan sehari-hari (fungsi praktis), mewujudkan seni (fungsi artistic), mempelajari naskah-naskah kuno (fungsi filosofis), serta usaha mengeksploitasi ilmu pengetahuan dan teknologi.

### **3.2.5 Kesenian**

Kesenian mengacu pada nilai keindahan (estetika) yang berasal dari ekspresi hasrat manusia akan keindahan yang dinikmati dengan mata ataupun telinga. Kesenian secara umum dapat dibedakan menjadi :

- a. Seni Rupa, yaitu kesenian yang dapat dinikmati secara visual (melalui mata).
- b. Seni Suara, yaitu kesenian yang dapat dinikmati melalui telinga atau didengar.

- c. Seni Drama, yaitu kesenian yang dapat dinikmati melalui mata dan telinga (dilihat dan didengarkan). Seni drama mengandung unsur-unsur dari seni lukis, seni musik, sastra, dan tari.

### 3.2.6 Sistem Religi (Kepercayaan)

Kepercayaan atau Religi adalah suatu keyakinan bahwa hal-hal yang dipercayai itu benar dan nyata (Tuhan, manusia, benda-benda, hewan, dll), ada harapan dan keyakinan (akan kejujuran, kebaikan), ada orang-orang yang dipercaya (ditugaskan), dan sebutan untuk sistem religi atau agama yang ada di Indonesia.

Semua aktivitas manusia yang berkaitan dengan kepercayaan atau agama didasarkan pada suatu getaran jiwa, yang disebut emosi keagamaan (religius emotion). Emosi keagamaan inilah yang membuat manusia melakukan tindakan yang bersifat keagamaan.

### 3.3 Multimedia Pembelajaran

*Multimedia* adalah media yang menggabungkan dua unsur atau lebih media yang terdiri dari teks, grafik, gambar, foto, audio, dan animasi secara terintegrasi. *Multimedia* terbagi menjadi dua kategori, yaitu: *multimedia linear*, dan *multimedia interaktif*. *Multimedia linear* adalah suatu *multimedia* yang tidak dilengkapi dengan alat pengontrol apapun yang dapat dioperasikan oleh pengguna. *Multimedia* ini berjalan *sekuensial* (berurutan), contohnya TV dan film. *Multimedia interaktif* adalah suatu *multimedia* yang dilengkapi dengan alat pengontrol yang dapat dioperasikan oleh pengguna, sehingga pengguna dapat memilih apa yang dikehendaki untuk proses selanjutnya. Contoh *multimedia* interaktif adalah: *multimedia* pembelajaran interaktif, aplikasi *game* dan lain-lain.

Sedangkan pembelajaran diartikan sebagai proses penciptaan lingkungan memungkinkan terjadinya proses belajar. Jadi dalam pembelajaran yang utama adalah bagaimana siswa belajar. Belajar dalam pengertian aktivitas mental siswa dalam berinteraksi dengan lingkungan yang menghasilkan perubahan perilaku yang bersifat relatif konstan. Dengan demikian aspek yang menjadi penting dalam aktivitas belajar dan pembelajaran adalah lingkungan. Bagaimana lingkungan ini diciptakan dengan menata unsur-unsurnya sehingga dapat merubah perilaku siswa.

Dari uraian pada halaman sebelumnya dapat disimpulkan bahwa *multimedia* pembelajaran dapat diartikan sebagai aplikasi *multimedia* yang digunakan dalam proses pembelajaran, dengan kata lain untuk menyalurkan pesan (pengetahuan, ketrampilan, dan sikap) serta dapat merangsang pikiran, perasaan, perhatian dan kemauan yang belajar sehingga secara sengaja proses belajar terjadi, bertujuan dan terkendali

### **3.4 Alat Musik**

Alat musik merupakan suatu instrumen yang dibuat atau dimodifikasi untuk tujuan menghasilkan musik. Pada prinsipnya, segala sesuatu yang memproduksi suara, dan dengan cara tertentu bisa diatur oleh musisi, dapat disebut sebagai alat musik. Walaupun demikian, istilah ini umumnya diperuntukkan bagi alat yang khusus ditujukan untuk musik. Bidang ilmu yang mempelajari alat musik disebut organologi. Berikut jenis – jenis alat music dari berdasarkan sumber bunyi dan cara memainkannya.

#### **3.4.1 Berdasarkan Sumber Bunyinya :**

- a. *Idiofon*, adalah alat musik yang sumber bunyinya berasal dari bahan dasarnya. Contoh: kolintang, drum, bongo, kabasa, angklung
- b. *Aerofon*, adalah alat musik yang sumber bunyinya berasal dari hembusan udara pada rongga. Contoh: suling, trompet, harmonika, trombon
- c. *Kordofon*, adalah alat musik yang sumber bunyinya berasal dari dawai. Contoh: bass, gitar, biola, gitar, sitar, piano, kecapi
- d. *Membranofon*, adalah alat musik yang sumber bunyinya dari selaput atau membran. contoh : tifa, drum, kendang, tam-tam, rebana
- e. *Elektrofon*, adalah alat musik yang sumber bunyinya dibangkitkan oleh tenaga listrik (elektronik). Contoh : *keyboard*, gitar listrik, bass listrik, piano listrik

#### **3.4.2 Berdasarkan Cara Memainkan nya :**



- a. Alat musik tiup menghasilkan suara sewaktu suatu kolom udara didalamnya digetarkan. Tinggi rendah nada ditentukan oleh frekuensi gelombang yang dihasilkan terkait dengan panjang kolom udara dan bentuk instrumen,




sedangkan timbre dipengaruhi oleh bahan dasar, konstruksi instrumen dan cara menghasilkannya. Contoh alat musik ini adalah trompet dan suling.

- b. Alat musik pukul menghasilkan suara sewaktu dipukul atau ditabuh. Alat musik pukul dibagi menjadi dua yakni bernada dan tidak bernada. Bentuk dan bahan bagian-bagian instrumen serta bentuk rongga getar, jika ada, akan menentukan suara yang dihasilkan instrumen. Contohnya adalah kolintang (bernada), drum (tak bernada), dan bongo (tak bernada).
- c. Alat musik petik menghasilkan suara ketika senar digetarkan melalui dipetik. Tinggi rendah nada dihasilkan dari panjang pendeknya dawai.
- d. Alat musik gesek menghasilkan suara ketika dawai digesek. Seperti alat musik petik, tinggi rendah nada tergantung panjang dan pendek dawai.

Adapun jenis alat musik daerah di Indonesia dapat dilihat pada tabel 3.1 sebagai berikut :





**Tabel 3.1** Daftar Alat Musik Daerah

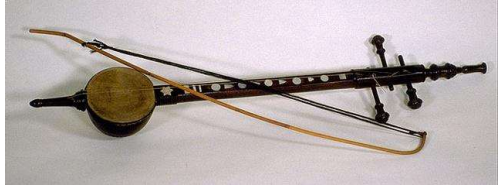



No	Nama Alat	Suku	Provinsi	Gambar
1	Serune Kalee	Suku Aceh	Naggroe Aceh Darussalam	
2	Bangsi Alas	Suku Aceh	Naggroe Aceh Darussalam	
3	Genderang	Suku Aceh	Naggroe Aceh Darussalam	





4	Taktok Trieng	Suku Aceh	Naggroe Aceh Darussalam	 <p>Taktok Trieng</p>
5	Saluang	Suku Minangkabau	Sumatera Barat	
6	Talempong	Suku Minangkabau	Sumatera Barat	








7	Gambus	Suku Akit, Suku Melayu	Riau	
8	Kompang	Suku Akit, Suku Melayu	Riau	
9	Nafiri	Suku Akit, Suku Melayu	Riau	
10	Rebana Sike	Suku Kerinci	Jambi	



11	Serdam	Suku Kerinci	Jambi	
12	Dol	Suku Serawai	Bengkulu	
13	Gendang Panjang	Suku Serawai	Bengkulu	
14	Cetik	Suku Lampung	Lampung	





15	Tehyan	Suku Betawi	DKI Jakarta	
16	Karinding	Suku Sunda	Jawa Barat	
17	Seruling	Suku Sunda	Jawa Barat	
18	Angklung	Suku Sunda	Jawa Barat	

19	Rebab	Suku Sunda	Jawa Barat	
20	Kecapi	Suku Sunda	Jawa Barat	
21	Lesung	Suku Badui	Banten	
22	Kendang	Suku Samin	Jawa Tengah	

23	Saron	Suku Samin	Jawa Tengah	
24	Siter	Suku Samin	Jawa Tengah	
25	Bonang	Suku Jawa	D.I. Yogyakarta	
26	Gong	Suku Jawa	D.I. Yogyakarta	
27	Saronen	Suku Jawa, Suku Madura	Jawa Timur	




				
28	Terompet Reog	Suku Jawa	Jawa Timur	
29	Gengceng	Suku Bali	Bali	
30	Genggong Bali	Suku Bali	Bali	
31	Rindik	Suku Bali	Bali	



32	Gerantang	Suku Bali	Bali	
33	Gendang Beleq	Suku Sasak	NTB	
34	Sasando	Suku Bima	NTT	

35	Sampe	Suku Dayak	Kalimantan Timur, Kalimantan Barat, Kalimantan Utara	
36	Balikan / Kurating	Suku Dayak	Kalimantan Timur, Kalimantan Barat, Kalimantan Utara	
37	Japen	Suku Ot Danum	Kalimantan Tengah	
38	Kolintang	Suku Minahasa	Sulawesi Utara	



39	Keso Keso	Suku Bugis, Suku Toraja	Sulawesi Selatan	
40	Marwas	Suku Gorontalo	Gorontalo	
41	Rumba	Suku Buru	Maluku	

42	Tifa	Suku Asmat, Suku Dani, Suku Arfak	Papua	
43	Pikon	Suku Asmat, Suku Dani, Suku Arfak	Papua	 <p>Alat Musik Tradisional Pikon Gambar: <a href="http://www.indonesiaheritage.org">www.indonesiaheritage.org</a></p>
44	Fu	Suku Togutil	Maluku Utara	

45	Batak Garantung	Suku Batak	Sumatera Utara	
46	Taganing	Suku Batak	Sumatera Utara	
47	Suling Bangka	Suku Melayu	Bangka Belitung	
48	Dambus	Suku Melayu	Bangka Belitung	

49	Panting	Suku Banjar	Kalimantan Selatan	
50	Ole Ole	Suku Batak Mandailing, Suku Nias	Sumatera Utara	

### 3.5 Unified Modeling Language (UML)

*Unified Modeling Language (UML)* adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun.



Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam makna tertentu dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Di dalam UML terdapat beberapa diagram yang di antaranya *Use Case diagram*, *Activity Diagram*, *Sequence Diagram* dan *Class Diagram*.

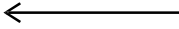
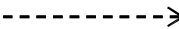
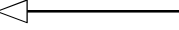


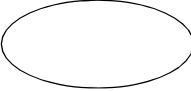
### 3.5.1 Use Case Diagram

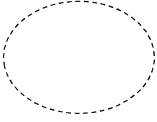

*Use Case* diagram adalah gambaran graphical dari beberapa atau semua *actor*, *Use Case* dan interaksi diantara komponen-komponen tersebut yang memperkenalkan suatu *system* yang akan dibangun. *Use Case* diagram menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada diluar sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar.

*Use Case* diagram dapat digunakan selama proses analisis untuk menangkap *requirements* sistem dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, *use case* diagram berperan untuk menetapkan perilaku (*behavior*) sistem saat diimplementasikan. Dalam sebuah model mungkin terdapat satu atau beberapa *use case* diagram. Kebutuhan atau *requirements* sistem adalah fungsionalitas apa yang harus disediakan oleh sistem kemudian didokumentasikan pada model *use case* yang menggambarkan fungsi *system* yang diharapkan (*Use Case*), dan yang mengelilinginya (*actor*), serta hubungan antara *actor* dengan *use case* (*Use Case* diagram) itu sendiri.

Tabel 3.2 Simbol Use Case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Action</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>Use Case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>Independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri ( <i>Independent</i> ).

3		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> )
4		<i>Include</i>	Menspesifikasikan bahwa <i>Use Case</i> sumber secara eksplisit
5		<i>Extend</i>	Menspesifikasikan bahwa <i>Use Case</i> target memperluas perilaku dari <i>Use Case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Simbol yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .

9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen- elemennya.
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

### 3.5.2 Activity Diagram


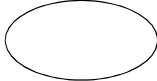



*Activity* Diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity* diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

*Activity* diagram merupakan *state* diagram khusus, dimana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *Activity* diagram tidak menggambarkan *behaviour* internal sebuah *system* secara eksak, tetapi lebih menggambarkan proses-proses dan jalur aktivitas dari level atas secara umum.

Sebuah aktivitas dapat direalisasikan oleh satu *Use Case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *Use Case* menggambarkan bagaimana *actor* menggunakan sistem untuk melakukan aktivitas.

Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan *behaviour* pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork and join*) digunakan titik sinkronisasi yang dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

Tabel 3.3 Simbol Activity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperhatikan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Action</i>	<i>State</i> dari system yang mencerminkan eksekusi dari suatu Aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan Dihancurkan
5		<i>Fork Node</i>	Suatu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

### 3.5.3 Class Diagram





*Class* adalah sebuah spesifikasi yang jika diinstansikan akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).

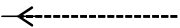
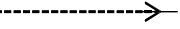



*Class* diagram menggambarkan struktur dan deskripsi *Class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. *Class* dapat merupakan implementasi dari sebuah *interface*, yaitu *Class* abstrak yang hanya memiliki metode.

Interface tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah *Class*. Dengan demikian interface mendukung resolusi metode pada saat *run-time*.

Tabel 3.4 Simbol Class Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak ( <i>descendent</i> ) sebagai perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan system yang menghasilkan suatu hasil yang terukur bagi suatu actor.

5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>Independent</i> ) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.





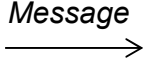
### 3.5.3 Sequence Diagram

*Sequence* Diagram merupakan gambaran secara grafis bagaimana objek berinteraksi satu sama lain melalui pesan saat melaksanakan sebuah operasi atau sebuah *Use Case*.

*Sequence* diagram mendeskripsikan pola interaksi antar objek yang diatur dalam sebuah urutan kronologi. Diagram ini menunjukkan objek-objek yang terlibat dalam interaksi serta *message* yang dikirim.

Object digambarkan dengan garis *vertical* putus-putus yang disebut dengan *lifeline*. *Message* adalah komunikasi antar *object* yang membawa informasi dari suatu *lifeline object* ke *lifeline object* lain. *Focus of Control* merepresentasikan waktu relatif dimana *flow of control* difokuskan dalam sebuah *object*. *Hierarchical numbering* menunjukkan keterangan sebuah *message* ke *message* yang lain. *Script* mendeskripsikan *flow of event* secara *tekstual*.

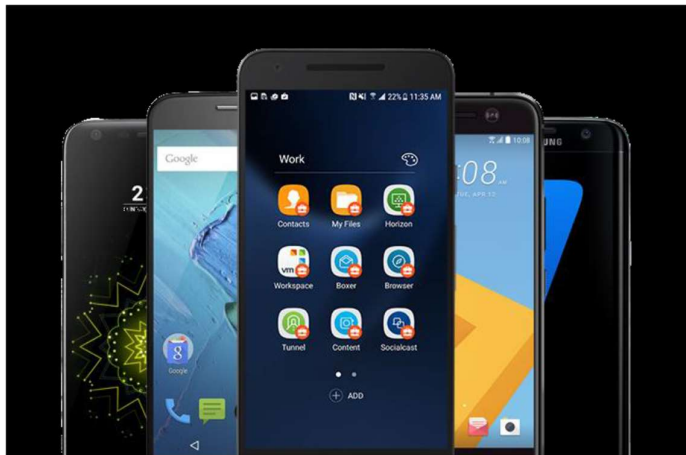
Tabel 3.5 Simbol Sequence Diagram

NAMA	KETERANGAN	GAMBAR
<i>Object</i>	<i>Object</i> merupakan <i>instance</i> dari sebuah <i>Class</i> dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah <i>Class</i> dengan nama objek didalamnya yang diawali dengan sebuah titik koma.	
<i>Actor</i>	<i>Actor</i> juga dapat berkomunikasi dengan objek, maka <i>actor</i> juga dapat durutkan sebagai kolom. Simbol <i>actor</i> sama dengan simbol pada <i>actor Use Case diagram</i> .	
<i>Lifeline</i>	<i>Lifeline</i> mengindikasikan keberadaan sebuah objek dalam basis waktu. Notasi untuk <i>lifeline</i> adalah garis putus-putus vertikal yang ditarik dari sebuah objek.	
<i>Activation</i>	<i>Activation</i> dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah <i>lifeline</i> . <i>Activation</i> mengindikasikan sebuah objek yang akan melakukan sebuah aksi.	
<i>Message</i>	<i>Message</i> digambarkan dengan anak panah horizontal antara <i>Activation</i> . <i>Message</i> mengindikasikan komunikasi antara objek-objek.	

### 3.6 Android

#### 3.6.1 Pengertian Android

*Android* adalah sistem operasi berbasis *Linux* yang dirancang untuk perangkat bergerak layar sentuh seperti telepon pintar dan komputer *tablet*. Antarmuka pengguna *Android* umumnya berupa manipulasi langsung, menggunakan gerakan sentuh yang serupa dengan tindakan nyata, misalnya menggeser, mengetuk, dan mencubit untuk memanipulasi objek di layar, serta papan ketik *virtual* untuk menulis teks menulis teks.



Gambar 3.2. *Smartphones* dengan sistem operasi *Android*

#### 3.6.2. Sejarah Android

Awalnya, *Google Inc.* membeli *Android Inc.*, pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, dan *Nvidia*. Pada saat perilisan perdana *Android*, 5 November 2007, *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, *Google* merilis kode-kode *Android* di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler

### 3.6.3. Perkembangan Android

Sejak dirilis pada tahun 2008, *Android* telah banyak menerima versi baru. Dalam versi-versi tersebut diberi nama seperti makanan, dan di setiap ada pembaruan versi ada penambahan fungsi yang baru.



Gambar 3.3. Versi Android beserta logonya

Pada tanggal 9 Februari 2009, Android versi 1.1 resmi diluncurkan. Pada versi 1.1 ini pihak *Google* telah memperbaiki bug yang ada pada Android versi sebelumnya dan menambah fitur-fitur baru yang belum ada pada Android versi 1.0. perilsan Android berikutnya sudah menggunakan nama-nama makanan penutup sebagai *codenamenya* dengan urutan alfabetis, yaitu dimulai dengan diluncurkannya Android versi 1.5 yang mempunyai naman Cupcake yang diluncurkan pada tanggal 30 April 2009, lalu diikuti dengan peluncuran Android yang memiliki nama Donut, Éclair, Frozen Yogurt (Froyo), dan Gingerbread.

Pada tanggal 22 Februari 2011 *Android Honeycomb* diluncurkan, dan pada tanggal 24 Februari 2011, tablet pertama yang menjalankan *Android Honeycomb* diluncurkan, yaitu *Motorola Xoom*. Lalu pada tanggal 19 Oktober 2011 *Android Meluncurkan Ice Cream Sanwich (ICS)* diluncurkan. Versi ICS ini mampu bekerja secara optimal baik di *smartphone* maupun di tablet. Berikutnya *Google* merilis *Android Jelly Bean* yang bertujuan untuk meningkatkan apa yang sudah tersedia di ICS, dengan memperbaiki bug-bug dan penambahan fitur-fitur.

Pada tanggal 3 September 2013, diumumkan versi *Android* berikutnya, yaitu *Android 4.4 Kit Kat*. *Android* sudah mendapatkan izin dari *Nestle* dan *Harshey* selaku pemilik merek dagang Kit Kat. Dan pada tanggal 12 November 2014, *Android Lollipop* resmi diperkenalkan, dan versi *Android Marshmallow* resmi diluncurkan pada tanggal 5 Oktober 2015, pada *Android Marshmallow* terdapat penambahan fitur-fitur baru yang belum ada pada *Android* versi sebelumnya.

Pada tanggal 22 Agustus 2016, *Android Nougat* resmi diluncurkan, lalu dikikuti oleh *Android Oreo* diluncurkan pada tanggal 21 Agustus 2017. Dan *Android* yang terbaru pada saat penelitian ini dilakukan adalah *Android Pie*. Dikutip dari Kompas, *Android Pie* telah diluncurkan pada tanggal 7 Agustus 2018. Dan smartphone pertama yang kebagian *Android Pie* ini adalah *Google Pixel* melalui pembaruan OTA.



Gambar 3.4. Logo *Android Pie*

### 3.7 Augmented Reality

Realitas tertambah atau *Augmented Reality* adalah teknologi baru yang melibatkan bagian dari komputer grafis dalam dunia nyata. Menurut Ronald Azuma, *Augmented Reality* adalah menggabungkan dunia nyata dan virtual, bersifat interaktif secara *real time*, dan merupakan animasi 3D. Paul Milgram dan Fumio Kishino pada tahun 1994, mendefinisikan Milgram's Reality-Virtuality Continuum. Mereka menggambarkan sebuah kontinum yang membentang dari lingkungan nyata untuk untuk lingkungan virtual murni. Mereka menyimpulkan bahwa AR lebih dekat dengan dunia nyata dan *augmented virtuality* lebih dekat dengan dunia virtual. Dalam penerapannya teknologi Augmented Reality memiliki beberapa komponen yang harus ada untuk mendukung kinerja dari proses

pengolahan citra digital. Adapun komponen-komponen tersebut adalah sebagai berikut :

**a. *Scene Generator***

*Scene Generator* adalah komponen yang bertugas untuk melakukan *Rendering* citra yang ditangkap oleh kamera. Objek *virtual* akan di tangkap kemudian diolah sehingga dapat kemudian objek tersebut dapat ditampilkan.

**b. *Tracking System***

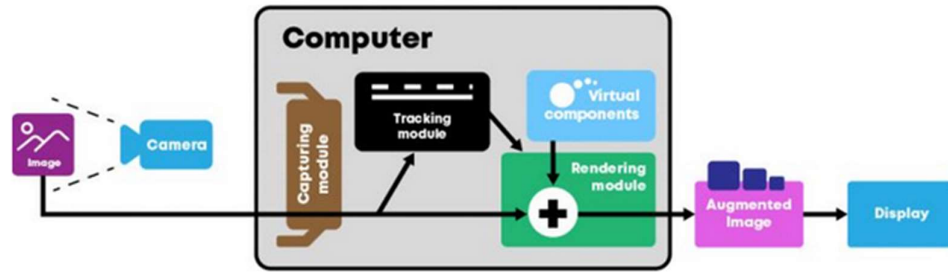
*Tracking system* merupakan komponen yang terpenting dalam *Augmented Reality*. Dalam proses *tracking* dilakukan sebuah pendeteksian pola objek *virtual* dengan objek nyata sehingga sinkron diantara keduanya dalam artian proyeksi *virtual* dengan proyeksi nyata harus sama atau mendekati sama sehingga mempengaruhi validitas hasil yang akan didapatkan.

**c. *Display***

Dalam pembangunan sebuah sistem yang berbasis AR dimana sistem tersebut menggabungkan antara dunia *virtual* dan dunia nyata ada beberapa parameter mendasar yang perlu diperhatikan yaitu optik dan teknologi video. Keduanya mempunyai keterkaitan yang tergantung pada faktor resolusi, fleksibilitas, titik pandang, *tracking area*. Ada batasan-batasan dalam pengembangan teknologi *Augmented Reality* dalam hal proses menampilkan objek. Diantaranya adalah harus ada batasan pencahayaan, resolusi layar, dan perbedaan pencahayaan citra antara citra *virtual* dan nyata.

**d. *AR Devices***

Ada beberapa tipe media yang dapat digunakan untuk menampilkan objek berbasis *Augmented Reality* yaitu dengan menggunakan optik, sistem retina *virtual*, video penampil, monitor berbasis AR dan proyektor berbasis AR. Berikut gambaran umum diagram alur *Augmented Reality*.



**Gambar 3.5** gambaran umum alur *Augmented Reality*

Gambar diatas menunjukkan diagram alur untuk sistem *augmented reality* sederhana. Capturing modul menangkap gambar dari kamera. Modul Tracking menghitung jarak lokasi dan orientasi kamera yang benar untuk menambahkan objek virtual. Modul render akan menyatukan gambar asli dan komponen virtual menggunakan pose yang dihitung dan kemudian merender gambar yang diperbesar pada layar.

### 3.8 Blender 3D Model Animation

Blender adalah perangkat lunak *open source* grafika komputer 3D. Perangkat lunak ini digunakan untuk membuat film animasi, efek *visual*, model cetak 3D, aplikasi 3D interaktif dan permainan video. Blender memiliki beberapa fitur termasuk pemodelan 3D, penteksturan, penyunting gambar bitmap, penulangan, simulasi cairan dan asap, simulasi partikel, animasi, penyunting video, pemahat digital, dan *Rendering* .

Blender 3D memiliki keunggulan sebagai aplikasi terbaik untuk membuat pemodelan 3D diantaranya :

- a. *Interface* yang *user friendly* dan tertata rapi.
- b. *Tool* untuk membuat objek 3D yang lengkap meliputi *modeling*, *UV mapping*, *texturing*, *rigging*, *skinning*, animasi, *particle* dan simulasi lainnya, *scripting*, *Rendering*, *compositing*, *post production* dan *game creation*.
- c. *Cross Platform*, dengan *uniform GUI* dan mendukung semua *platform*. Blender 3D bisa digunakan untuk semua versi *windows*, *Linux*, *OS X*, *FreeBSD*, *Irix*, *Sun* dan sistem operasi yang lainnya.
- d. Kualitas arsitektur 3D yang berkualitas tinggi dan bisa dikerjakan dengan lebih cepat dan efisien.



- e. Dukungan yang aktif melalui forum dan komunitas.
- f. File berukuran kecil.

### 3.9 Unity 3D

*Unity* 3D adalah sebuah *game engine* yang berbasis *cross-platform*. *Unity* dapat digunakan untuk membuat sebuah *game* yang bisa digunakan pada perangkat komputer, ponsel pintar *Android*, *iPhone*, PS3, dan bahkan *X-BOX*. *Unity* adalah sebuah *tool* yang terintegrasi untuk membuat *game*, arsitektur bangunan, *Augmented Reality* dan simulasi. *Unity* bisa untuk *games* PC dan *games Online*. Untuk *games Online* diperlukan sebuah *plugin*, yaitu *Unity Web Player*, sama halnya dengan *Flash Player* pada *Browser*.

*Unity* tidak dirancang untuk proses desain atau *modelling*, dikarenakan *unity* bukan *tool* untuk mendesain. Jika ingin mendesain, pergunakan 3D *editor* lain seperti 3dsmax atau Blender. Banyak hal yang bisa dilakukan dengan *unity*, ada fitur *audio reverb zone*, *particle effect*, dan *Sky Box* untuk menambahkan langit. Fitur *scripting* yang disediakan, mendukung 3 bahasa pemrograman, *JavaScript*, *C++*, dan *Boo*. *Flexible and EasyMoving*, *rotating*, dan *scaling objects* hanya perlu sebaris kode. Begitu juga dengan *duplicating*, *removing*, dan *changing properties*. *Visual Properties Variables* yang di definisikan dengan *scripts* ditampilkan pada *Editor*. Bisa digeser, di *drag and drop*, bisa memilih warna dengan *color picker*. Berbasis .NET. Artinya penjalanan program dilakukan dengan *Open Source .NET platform*, *Mono*. Dalam pembuatan *Augmented Reality* menggunakan *unity* 3D, kita tambahkan *Vuforia* sebagai *plug-in* dari *Augmented Reality* itu sendiri dan *unity* 3D sebagai SDK nya.

### 3.10 Vuforia SDK Qualcomm

*Vuforia SDK Qualcomm* merupakan *library* yang digunakan sebagai pendukung adanya *Augmented Reality* pada *android*. *Vuforia* menganalisa gambar dengan menggunakan pendeteksi *marker* dan menghasilkan informasi 3D dari *marker* yang sudah dideteksi via API. *Programmer* juga dapat menggunakannya untuk membangun objek 3D *virtual* pada kamera. Adapun contoh nyata pembuatan

objek 3D dengan menggunakan vuforia adalah seperti ditunjukkan pada gambar di bawah ini.



**Gambar 3.6** Objek 3D Rumah *Virtual* pada Kamera

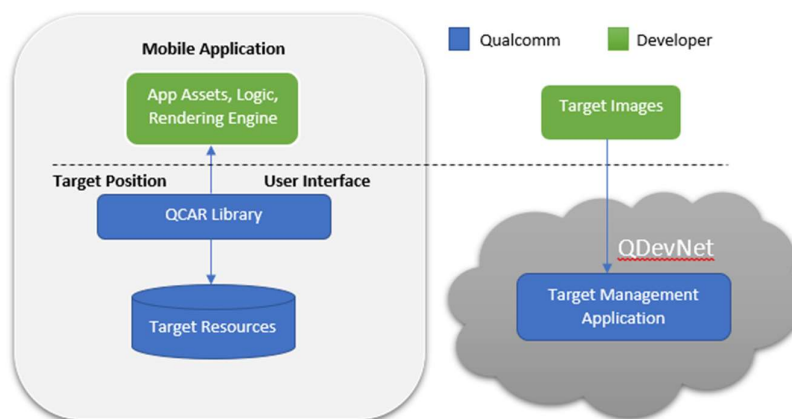
Gambar 3.6 adalah pengembangan aplikasi menggunakan *platform Qualcomm AR*. *Platform* tersebut terdiri dari 2 komponen diantaranya adalah.

**a. Target Management System**

Mengizinkan pengembang melakukan *upload* gambar yang sudah diregistrasi oleh *marker* dan kemudian melakukan *download* target gambar yang akan dimunculkan.

**b. QCAR SDK Vuforia**

Mengizinkan pengembang untuk melakukan koneksi antara aplikasi yang sudah dibuat dengan *library static* i.e libQCAR.a pada iOS atau libQCAR.so pada *android*.



**Gambar 3.7** Arsitektur *Library QCAR SDK*

Gambar 3.3 memberikan gambaran umum pembangunan aplikasi dengan *Qualcomm AR Platform*. *Platform* ini terdiri dari SDK QCAR dan *Target System*

*Management* yang dikembangkan pada *portal* QdevNet. *User* meng-*upload* gambar masukan untuk target yang ingin dilacak dan kemudian men-*download* sumber daya target, yang dibundel dengan *app*. SDK QCAR menyediakan sebuah objek yang terbagi *libQCAR.so* yang harus dikaitkan dengan *app*. Komponen inti dari QCAR Lib adalah sebagai berikut:

**1) *Camera***

Spesifikasi kamera cukup dengan menggunakan kamera tunggal. Kamera mengambil gambar untuk melacak *marker* dan kemudian melakukan registrasi *marker*. Pengembang dapat mengaturnya ketika memulai dan menghentikan pengambilan gambar.

**2) *Image Converter***

Gambar akan dikonversi dari *format* YUV 12 ke *format* RGB565 untuk OpenGL ES kemudian mengatur pencahayaan untuk pelacakan *marker*.

**3) *Tracker***

Menggunakan algoritma *computer vision* untuk mendeteksi dan melakukan pelacakan objek nyata yang diambil dari kamera. Objek tersebut dievaluasi dan hasilnya akan disimpan yang kemudian akan diakses oleh aplikasi.

**4) *Renderer***

Digunakan untuk melakukan *Rendering* hasil objek yang ditangkap oleh kamera ke video yang dimaksudkan untuk optimasi *device*.

**5) *Application code***

*Application code* melibatkan inisialisasi dari semua komponen diatas. Selama objek yang dikehendaki diubah prosesnya, maka *application code* harus diubah berdasarkan lokasi objek *virtual*.

**6) *Target Resources***

*Target resource* dihasilkan dari *target management system*. *Output* yang dihasilkan dari sistem berupa *file binary* yang menyimpan pola *marker* dan file konfigurasi XML. Dan semuanya digabung dalam sebuah aplikasi.

### **3.11 Marker**

*Marker* adalah *real enviroment* berbentuk objek nyata yang akan menghasilkan *virtual reality*, *marker* ini digunakan sebagai tempat *Augmented*

*Reality* muncul, berikut ini beberapa jenis *marker* yang digunakan pada aplikasi *Augmented Reality*

**a. Quick Response (QR)**

*Quick Response* (QR) Kode dua dimensi kode yang terdiri dari banyak kotak diatur dalam pola persegi, Biasanya QR ini berwarna hitam dan putih, kode QR diciptakan di Jepang pada awal 1990-an dan digunakan untuk melacak berbagai bagian dalam manufaktur kendaraan. Dan saat ini QR digunakan sebagai *link* cepat ke *website*, *dial* cepat untuk nomor telepon, atau bahkan dengan cepat mengirim pesan SMS seperti pada gambar 2.3. QR (*Quick Response*) Code.



**Gambar 3.8** QR (*Quick Response*) Code

**b. Fiducial Marker**

*Fiducial Marker* adalah bentuk paling sering digunakan oleh teknologi AR karena *marker* ini digunakan untuk melacak benda-benda di *virtual reality* tersebut. kotak hitam dan putih digunakan sebagai titik referensi atau untuk memberikan skala dan orientasi ke aplikasi. Bila penanda tersebut deteksi dan dikenali maka *Augmented Reality* akan keluar dari *marker* ini seperti pada gambar 2.4. *Fiducial Marker*.



**Gambar 3.9** *Fiducial Marker*

c. ***Markerless Marker***

*Markerless Marker* berfungsi sama seperti *Fiducial Marker* yang namun bentuk *Markerless Marker* tidak harus kotak hitam putih, *Markerless* ini bisa berbentuk gambar yang mempunyai banyak warna seperti pada gambar 3.10 *Markerless Marker*.

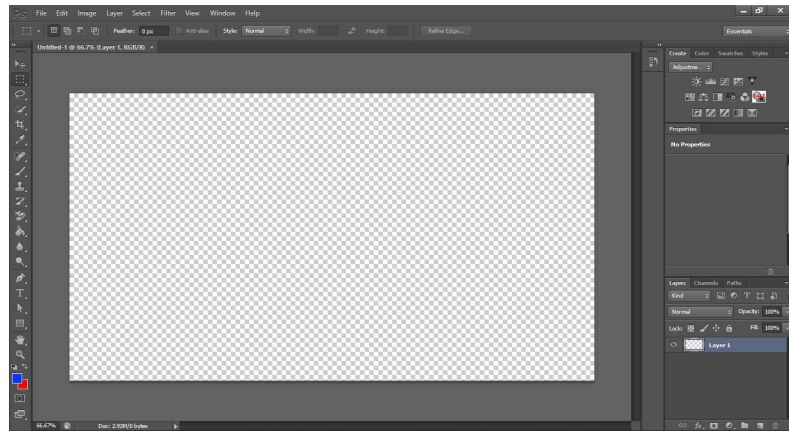


**Gambar 3.10** *Markerless Marker*.

### **3.12 Adobe Photoshop**

Adobe Photoshop, atau biasa disebut Photoshop, adalah perangkat lunak *editor* citra buatan Adobe *Systems* yang dikhususkan untuk pengeditan foto atau gambar dan pembuatan efek. Perangkat lunak ini banyak digunakan oleh fotografer *digital* dan perusahaan iklan sehingga dianggap sebagai pemimpin pasar (*market leader*) untuk perangkat lunak pengolah gambar/foto, dan, bersama Adobe Acrobat, dianggap sebagai produk terbaik yang pernah diproduksi oleh Adobe *Systems*. Versi kedelapan aplikasi ini disebut dengan nama Photoshop CS (*Creative Suite*), versi sembilan disebut Adobe Photoshop CS2, versi sepuluh disebut Adobe Photoshop CS3, versi kesebelas adalah Adobe Photoshop CS4, versi keduabelas adalah Adobe Photoshop CS5, dan versi terbaru adalah Adobe Photoshop CC.

Adobe Photoshop dapat di download di website resminya baik secara free maupun berbayar. Semakin tinggi versinya maka semakin banyak fitur yang didapatkan ketika menginstall aplikasi ini. Untuk dapat menginstallnya, kita membutuhkan spesifikasi sebuah computer atau laptop dengan spesifikasi yang mumpuni untuk melakukan kegiatan desain grafis.



Gambar 3.11 Tampilan Adobe Photoshop CS6 pada windows 10

Gambar 3. 11 adalah tampilan awal lembar kerja dari Adobe Photoshop CS6 pada windows 10. Untuk versi CS6 ini memiliki fitur yang lumayan lengkap dibandingkan versi sebelumnya.

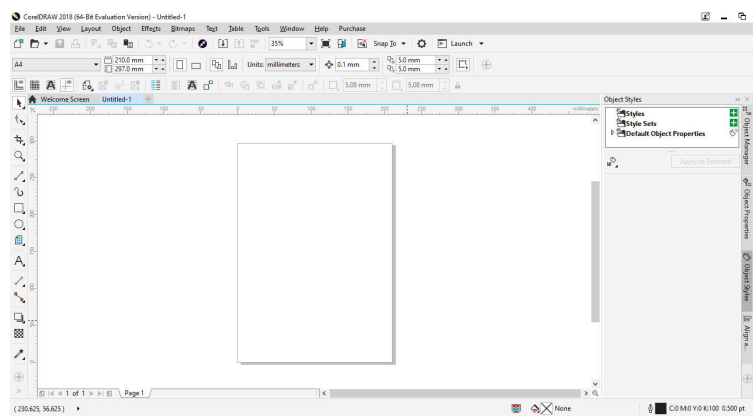
### 3.13 CorelDraw

*CorelDraw* adalah *editor* grafik *vektor* yang dikembangkan oleh *Corel*, sebuah perusahaan perangkat lunak yang bermarkas di Ottawa, Kanada. Versi terbarunya, *CorelDRAW X8* (setara dengan versi 18) dirilis pada tanggal 15 maret 2016. *CorelDRAW* pada awalnya dikembangkan untuk sistem operasi *Windows* 2000 dan seterusnya. Versi *CorelDRAW* untuk *Mac OS* pernah dikembangkan, namun dihentikan karena tingkat penjualannya rendah. Versi terakhir untuk *Linux* adalah Versi 9 (Rilisan tahun 2000, itupun perlu di jalankan dengan *Wine*) dan Untuk *OSX* adalah versi 11 (Rilisan 2001).



Gambar 3.12 tampilan awal *CorelDraw* 2018

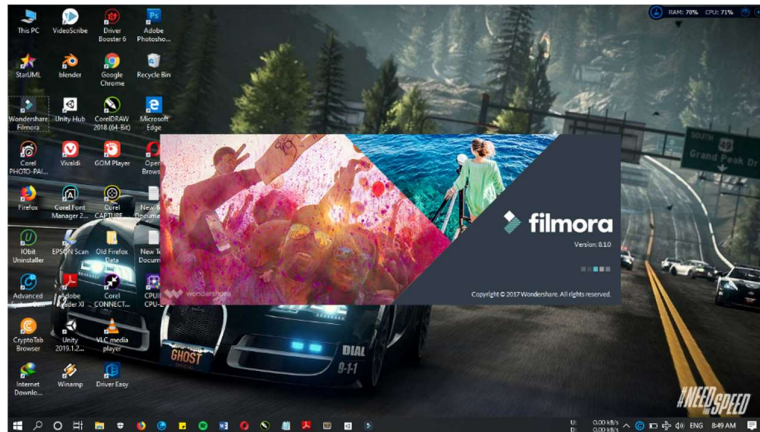
*Format vector* adalah gambar yang membentuk sejumlah objek garis dan objek kurva berdasarkan rumusan matematis. *Format vector* lebih banyak digunakan untuk membentuk objek buatan, seperti menggambar objek dua dimensi, yang lebih ditekankan ke dalam pembuatan objek garis, lingkaran, *polygon* dan persegi panjang. Sedangkan untuk objek tiga dimensi lebih ditekankan ke dalam pembuatan: bola, kubus dan tabung. Objek *vector* banyak digunakan dalam pembuatan pengolahan teks dan *logo*. *Software* yang banyak berhubungan dengan *format vector* yaitu *CoreDRAW*, *freehand*, dan lain-lain. Ruang penyimpanan *format vector* relatif lebih kecil dibandingkan dengan format *bitmap*. Salah satu *Software Desain Grafis* ternama adalah *CorelDraw* yang merupakan *Software* pembantu dalam meng-*edit* bentuk-bentuk desain dari *cover* majalah, surat, brosur membuat pamflet ataupun *cover* dari sebuah buku yang dapat dengan mudah kita untuk mengkreasikan bentuk-bentuk gambar hingga menjadi lebih menarik dengan menggunakan *CorelDRAW* karena terdapat banyak fasilitas penunjang yang disediakan di dalam *software* ini.



Gambar 3.13 Tampilan CorelDraw 2018 pada windows 10

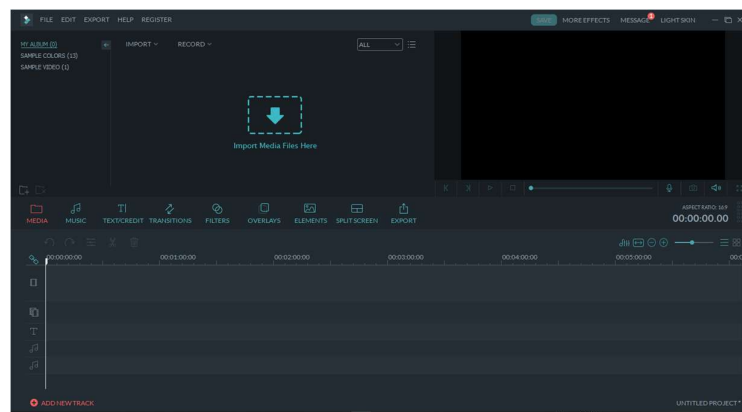
### 3.14 Wondershare Filmora

*Filmora* atau lengkapnya *Wondershare Filmora Video Editor* adalah sebuah aplikasi atau program yang dirancang untuk membuat proses pengeditan video dengan mudah dan sederhana tapi memiliki kualitas yang cukup powerful. *Filmora* juga memiliki beragam fitur seperti fitur – fitur yang ada di *Adobe Premiere Pro*. Berikut tampilan ketika *filmora* sedang dibuka atau dijalankan.



Gambar 3.14 Tampilan Awal Wondershare Filmora

Setelah membuka aplikasi filmora maka akan muncul halaman untuk mengerjakan proyek atau file baru seperti gambar 3.14 pada halaman berikutnya.



Gambar 3.15 tampilan halaman proyek dari filmora

Untuk langkah selanjutnya kita dapat melakukan editing pada sebuah video yang akan kita jadikan sebagai proyek dan aplikasi ini bisa didapatkan secara *free* dan berbayar.

### 3.15 Uji Validitas dan Realibilitas

#### 3.15.1 Uji Validitas

Validitas adalah ketepatan atau kecermatan suatu instrument dalam pengukuran. Dalam pengujian instrument pengumpulan data, validitas dibedakan menjadi validitas factor dan validitas item. Validitas faktor diukur bila item yang disusun menggunakan lebih dari satu faktor (antara faktor satu dengan yang lain ada kesamaan. Pengukuran validitas faktor ini dengan cara mengkorelasikan antara



skor faktor (penjumlahan item dalam satu faktor) dengan skor total faktor (total keseluruhan faktor). Pengukuran validitas item dengan cara mengkorelasikan antara skor item dengan skor total item. Validitas item ditunjukkan dengan adanya korelasi atau dukungan terhadap item total (skor total). Bila kita menggunakan lebih dari satu faktor, berarti pengujian validitas item dengan cara mengkorelasikan antara skor item dengan skor faktor, kemudian dilanjutkan mengkorelasikan antara item dengan skor total faktor (penjumlahan dari beberapa faktor). Dari hasil perhitungan korelasi akan di dapat suatu koefisien korelasi yang digunakan untuk mengukur tingkat validitas suatu item dan menentukan apakah suatu item layak digunakan atau tidak. Dalam menentukan layak atau tidaknya suatu item yang digunakan, biasanya digunakan uji signifikansi valid jika berkorelasi signifikan terhadap skor total. Teknik pengujian SPSS yang sering digunakan untuk uji validitas adalah menggunakan korelasi Bivariate Pearson (Produk Momen Pearson) dan Corrected Item - Total Correlation. Bivariate Pearson (Korelasi Produk Momen Pearson) adalah analisis dengan mengkorelasikan masing-masing skor item dengan skor total. Skor total adalah penjumlahan dari keseluruhan item. Item-item pertanyaan yang berkorelasi signifikan dengan skor total menunjukkan item-item tersebut mampu memberikan dukungan dalam mengungkap apa yang ingin diungkap, rumus korelasi produk moment dari pearsons yang digunakan :

$$r_{XY} = \frac{N\sum XY - \sum X \sum Y}{\sqrt{(N\sum X^2 - (\sum X)^2)(N\sum Y^2 - (\sum Y)^2)}}$$

Gambar 3.16 Rumus korelasi produk moment pearson

Keterangan :

- rx<sub>y</sub> = koefisien korelasi antara variabel X dan Y
- N = jumlah responden
- ΣX = jumlah skor butir soal
- ΣY = jumlah skor total soal
- ΣX<sup>2</sup> = jumlah skor kuadrat butir soal
- ΣY<sup>2</sup> = jumlah skor total kuadrat butir soal

Nilai  $r$  hitung dicocokkan dengan  $r_{\text{tabel}}$  product moment pada taraf signifikan 5%. Jika  $r$  hitung lebih besar dari  $r_{\text{tabel}}$  5%. Maka butir soal tersebut valid.

### 3.15.2 Uji Reliabilitas

Uji reliabilitas digunakan untuk mengetahui konsistensi alat ukur, apakah alat pengukur yang digunakan dapat diandalkan dan tetap konsisten jika pengukuran tersebut diulang. Ada beberapa metode pengujian reliabilitas di antaranya metode tes ulang, formula Flanagan, Cronbach's Alpha, metode formula KR (Kuder-Richardson) – 20, KR – 21, dan metode Anova Hoyt. Metode yang sering digunakan dalam penelitian adalah metode Cronbach's Alpha. Metode ini sangat cocok digunakan pada skor dikotomi (0 dan 1) dan akan menghasilkan perhitungan yang setara dengan menggunakan metode KR-20 dan Anova Hoyt. Reliabilitas berarti dapat dipercaya” Artinya, instrumen dapat memberikan hasil yang tepat. Alat ukur instrument dikategorikan reliabel jika menunjukkan konstanta hasil pengukuran dan mempunyai ketetapan hasil pengukuran sehingga terbukti bahwa alat ukur itu benar-benar dapat dipertanggung jawabkan kebenarannya.

#### a. Reliabilitas Skala.

Untuk mengukur reliabilitas skala atau kuosioner dapat digunakan rumus Cronbach's Alpha sebagai berikut:

$$r_{tt} = \left[ \frac{k}{k-1} \right] \left[ 1 - \left[ \frac{\sum \delta_b^2}{\sum \delta_t^2} \right] \right]$$

Gambar 3.17 Rumus Cronbach's Alpha

Keterangan :

$r_{tt}$  = koefisien reliabilitas instrument (total tes)

$k$  = banyaknya butir pertanyaan yang valid

$\sum \delta_b^2$  = jumlah varian butir

$\sum \delta_t^2$  = varian skor total Perhitungan uji reliabilitas skala  
diterima, jika hasil perhitungan rhitung >  $r_{\text{tabel}}$  5%.

b. Reliabilitas Tes

Untuk mengukur reliabilitas tes menggunakan rumus KR-20. Karena skor tes bersifat dikotomi yaitu untuk jawaban benar diberi skor 1 dan jawaban salah diberi skor 0. Adapun rumus KR-20 adalah sebagai berikut.

$$r_{tt} = \left[ \frac{k}{k-1} \right] \left[ \frac{v_t - \Sigma pq}{v_t} \right]$$

Gambar 3.18 Rumus KR-20

Keterangan :

- rtt : reliabilitas tes
- k : banyaknya butir soal yang sah
- vt : varian total
- p : proporsi subyek yang menjawab soal dengan benar
- q : proporsi subyek yang menjawab soal dengan salah
- $\Sigma pq$  : jumlah hasil perkalian antara p dan q

Instrumen dapat dikatakan valid jika memenuhi kriteria bahwa rhitung > rtabel 5%.

Kategori koefisien reliabilitas (Guilford, 1956: 145) adalah sebagai berikut:

- a.  $0,80 < r_{11} \leq 1,00$  reliabilitas sangat tinggi
- b.  $0,60 < r_{11} \leq 0,80$  reliabilitas tinggi
- c.  $0,40 < r_{11} \leq 0,60$  reliabilitas sedang
- d.  $0,20 < r_{11} \leq 0,40$  reliabilitas rendah
- e.  $-1,00 \leq r_{11} \leq 0,20$  reliabilitas sangat rendah (tidak reliable)

Untuk melakukan uji validitas dan uji reliabilitas dapat dilakukan menggunakan *software* SPSS sebagai alat bantu hitung sebagai alat bantu menghitung nilai statistik untuk menghitung nilai r Bivariate Pearson (Korelasi Produk Momen Pearson) dan cronbach's alpha.

### **3.16 Media Pembelajaran**

Memilih media yang tepat untuk digunakan dalam pembelajaran tidaklah mudah, selain memerlukan analisis mendalam dengan mempertimbangkan berbagai aspek juga di butuhkan prinsip – prinsip tertentu agar pemilihan media bisa lebih tepat. Media pembelajaran sangat berperan untuk keberhasilan proses belajar mengajar. Peranan media pembelajaran terutama adalah untuk membantu penyampaian materi kepada siswa. Dalam hal ini bisa terlihat bahwa tingkat kualitas atau hasil belajar juga dipengaruhi oleh kualitas media pembelajaran yang digunakan.

Untuk mendapatkan kualitas media pembelajaran yang baik agar dapat memberikan pengaruh yang signifikan dalam proses belajar mengajar, maka diperlukan pemilihan dan perencanaan penggunaan media pembelajaran yang baik dan tepat. Pemilihan media pembelajaran yang tepat ini menjadikan media pembelajaran efektif digunakan dan tidak sia-sia jika diterapkan.

Berikut ini beberapa prinsip yang harus diperhatikan saat memilih media untuk pembelajaran yang akan dilaksanakan :

#### **a. Prinsip Efektivitas dan Efisiensi**

Media yang telah memenuhi aspek efektivitas dan efisiensi ini tentunya akan meningkatkan ketertarikan masyarakat umum khususnya anak - anak dalam belajar dan mendukung pencapaian dalam pembelajaran.

#### **b. Prinsip Interaktivitas Media Pembelajaran**

prinsip ketiga yang harus diperhatikan dalam pemilihan media dalam pembelajaran di dalam lingkungan masyarakat umum adalah interaktivitas. Seberapa besar kemungkinan masyarakat umum dan anak - anak dapat berinteraksi dengan media pembelajaran? Makin interaktif, makin bagus media pembelajaran itu karena lebih mendorong mereka untuk terlibat aktif dalam belajar. Dalam hal ini, maka media yang paling cocok untuk dipilih adalah media pembelajaran dalam bentuk interaktif.

### 3.17 Pengukuran Pendidikan

#### 3.17.1 Alat Ukur Pendidikan

Alat ukur pendidikan dapat diklasifikasikan antara lain sebagai berikut :

- a. isian inventori (biasanya berbentuk isian atau kuesioner),
- b. tes (tes hasil belajar, tes kinerja, tes inteligensi, tes bakat, tes kepribadian), kuesioner (berisikan sejumlah butir yang ditanyakan kepada responden),
- c. wawancara (interview atau wawancara dilakukan oleh pewawancara kepada responden),
- d. observasi (pengamatan),
- e. daftar cocok (chek list),
- f. skala (alat ukur kiraan atau rating),
- g. studi kasus,
- h. riwayat hidup,
- i. sosiometri, dan
- j. asesmen portofolio (Koyan, 2012).

#### 3.17.2 Fungsi Tes

Secara harfiah kata “test” berasal dari kata bahasa dari bahasa latin “testum” yaitu alat untuk mengukur tanah (Fresch and Wheaton: 2002). Dalam bahasa Prancis kuno, kata tes berarti ukuran yang dipergunakan untuk membedakan emas dan perak dari logam-logam yang lain. Dalam bahasa Indonesia diterjemahkan dengan tes yang berarti ujian atau percobaan. Fungsi tes dalam pendidikan adalah: Sebagai alat untuk mengukur prestasi belajar siswa.

- a. Sebagai motivator dalam pembelajaran.
- b. Sebagai upaya perbaikan kualitas pembelajaran.
- c. Sebagai penentu berhasil atau tidaknya siswa sebagaisyarat untuk melanjutkan pendidikan ke jenjang yang lebih tinggi (Djaali dan Muljono, 2008).

Kegunaan dari tes adalah:

- a. Memperoleh umpan balik terhadap hasil pembelajaran
- b. Memperbaiki kurikulum dan program pendidikan
- c. Meningkatkan motivasi siswa

- d. Melaksanakan diagnosis dan remedial
- e. Melakukan penempatan
- f. Melakukan seleksi
- g. Mengembangkan khasanah ilmu pengetahuan.

### 3.17.3 Kriteria Tes

Suatu tes dapat dikatakan baik bilamana tes tersebut memiliki kriteria sebagai alat ukur yang baik. Kriterianya antara lain :

- a. Memiliki Validitas (kesahihan) yang cukup tinggi Suatu tes dikatakan valid jika tes tersebut mengukur tujuan atau salah satu aspek tujuan yang peneliti ukur.
- b. Memiliki Reliabilitas (kejegan / kestabilan) yang baik Tes dikatakan reliabel jika mengukur secara konsisten.
- c. Memiliki Nilai Objektivitas Objektivitas suatu tes ditentukan oleh tingkat atau kualitas kesamaan skor-skor yang diperoleh dengan tes tersebut meskipun hasil tes itu dinilai oleh beberapa orang penilai.

### 3.17.4 Kepraktisan Tes

Kepraktisan Suatu tes dikatakan mempunyai kepraktisan yang baik jika kemungkinan untuk menggunakan tes itu besar. Hal-hal yang perlu diperhatikan dalam perencanaan tes adalah:

- a. pengambilan sampel dan pemilihan butir soal,
- b. tipe tes yang akan digunakan,
- c. aspek yang akan diuji,
- d. format butir soal,
- e. jumlah butir soal, dan
- f. distribusi tingkat kesukaran butir soal (Asmawi Z. dan Noehi Nasution, 2005).

### 3.17.5 Bentuk Tes

Secara umum bentuk tes hasil belajar dapat dibedakan atas :

a. Tes Subjektif

Tes Subjektif adalah tes yang dalam pemeriksaannya banyak dipengaruhi oleh unsur-unsur subjektif. Butir-butir tes subjektif biasanya dirumuskan dalam bentuk pertanyaan terbuka seperti: apa, bagaimana, mengapa, siapa, kapan dan sebagainya. Contoh tes subjektif yaitu tes Esei atau uraian, yang penskorannya hanya dapat dilakukan secara subjektif

b. Tes objektif

Tes Objektif adalah tes yang dalam pemeriksaannya dapat dilakukan secara objektif. Adapun klasifikasi bentuk tes objektif adalah sebagai berikut:

1. Tes Pilihan ganda (Multiple Choice Test)
2. Tes Benar-Salah (True-False Test),
3. Tes Menjodohkan (Matching Test).
4. Tes Melengkapi (Completion Test),
5. Tes Jawaban Singkat (Short answer Test).

Petunjuk praktis dalam menyusun tes pilihan ganda, yaitu:

1. Definisikan tugas-tugas dalam stem secara jelas,
2. Tulis alternatif jawaban pada akhir pertanyaan,
3. Tempatkan sebanyak mungkin kata-kata dalam sistem,
4. Hindari penggunaan kata-kata negatif,
5. Hindari sistem yang mengarah pada alternatif jawaban yang salah atau benar,
6. Buat alternatif jawaban yang paralel,
7. Tulis alternatif jawaban secara vertikal,
8. Hindari jawaban “semua di atas”,
9. Buat alternatif jawaban sama panjang,
10. Hilangkan petunjuk ke arah jawaban benar,
11. Buat pengecoh yang masuk akal,
12. Usahakan sistemnya dalam bentuk pertanyaan,

13. Kontrol tingkat kesulitan soal sehingga persentase jawaban benar kira- kira separuhnya,
14. Hindari kemungkinan menebak,
15. Gunakan jawaban “tidak ada jawaban benar” hanya kalau tidak ada jawaban lain,
16. Susun alternatif jawaban sesuai dengan abjad atau urutan lainnya,
17. Soal harus sesuai dengan indikator,
18. Letakkan jawaban benar secara acak, dan
19. Usahakan memiliki empat sampai lima alternatif jawaban (Hopkin dan Antes, 1990)