

BAB II

LANDASAN TEORI

2.1 Matematika Diskrit

Farida (2016:41) menjelaskan bahwa Matematika Diskrit adalah cabang matematika yang membahas segala sesuatu yang bersifat diskrit. Objek yang dibahas dalam Matematika Diskrit seperti bilangan bulat, graf, atau kalimat logika. Matematika diskrit merupakan mata kuliah utama dan dasar untuk bidang ilmu komputer atau informatika.

2.2 Multimedia

2.2.1 Pengertian Multimedia

Arfida & Harahap (2014:409) menguraikan bahwa Multimedia adalah kombinasi data atau media untuk menyampaikan sebuah informasi sehingga informasi tersebut tersaji dengan lebih menarik. Sedangkan definisi lainnya menyatakan bahwa Multimedia merupakan kombinasi dari teks, grafis, seni, suara, animasi dan video yang dikirimkan oleh komputer atau peralatan elektronik lain.

2.2.2 Objek - Objek Multimedia

(Arfida & Harahap (2014:409) menjelaskan bahwa objek - objek multimedia adalah sebagai berikut :

a. Teks

Teks adalah bentuk data multimedia yang paling mudah disimpan dan dikendalikan. Kebutuhan teks tergantung pada kegunaan aplikasi multimedia.

b. Grafik

Grafik menjadi nilai dan unsur tambah suatu penyajian data. Gambar digunakan dalam presentasi multimedia untuk menarik perhatian.

c. Gambar Vektor

Gambar vektor disimpan sebagai serangkaian instruksi yang digunakan untuk membuat suatu gambar yang dinamakan algoritma, yang menentukan bentuk kurva, garis dan berbagai bangun yang diwakilkan oleh gambar (*picture*).

d. Gambar Bitmap

Gambar bitmap adalah gambar yang tersimpan dalam rangkaian *pixel* (titik – titik). Komputer akan mengatur tiap titik di layar sesuai dengan detil warna bitmap.

e. Suara (*Audio*)

Penyampaian sebuah informasi yang sering disertai desain grafis dan teks yang menarik, akan terasa membosankan apabila tidak disertai dengan suara.

f. Video

Video menyediakan sumber daya yang kaya dan membuat aplikasi multimedia lebih hidup. Namun kendala yang dihadapi adalah ukuran file yang terlalu besar. Untuk itu diperlukan software lain untuk memperkecil ukuran file video.

g. Animasi (*Animation*)

Animasi dalam multimedia merupakan penggunaan komputer untuk menciptakan gerak pada layar.

2.3 Media Pembelajaran

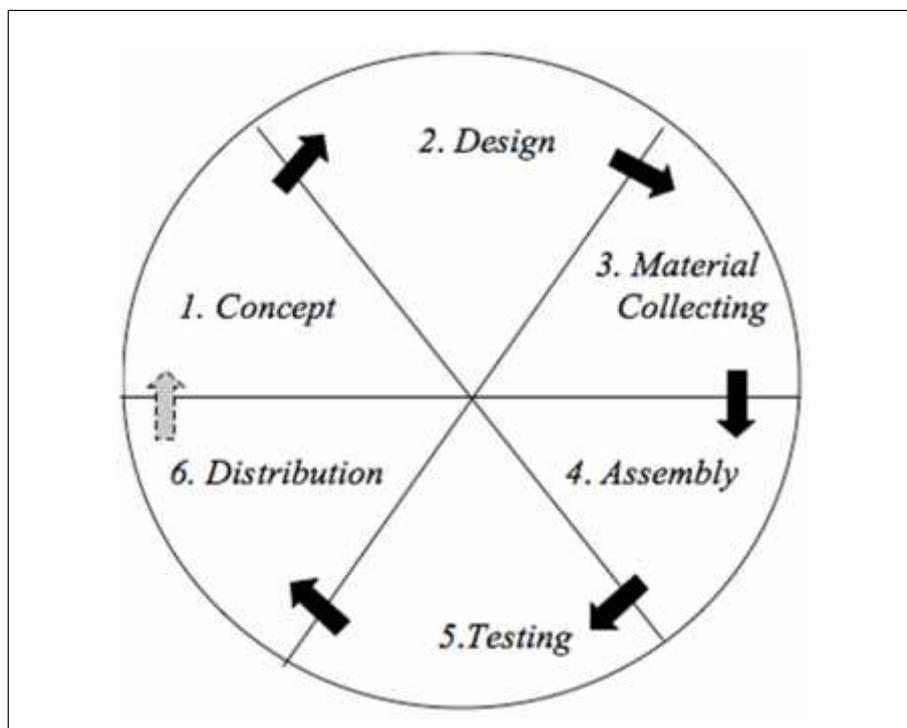
Yohana (2011:9) mendefinisikan bahwa media pembelajaran sebagai “segala sesuatu yang dapat menyampaikan dan menyalurkan pesan dari sumber secara terencana sehingga tercipta lingkungan belajar yang kondusif dimana penerimanya dapat melakukan proses belajar secara efisien dan efektif”. Definisi lainnya menjelaskan “media pembelajaran adalah alat yang dapat membantu proses belajar mengajar dan berfungsi untuk memperjelas makna pesan yang disampaikan, sehingga dapat mencapai tujuan pembelajaran dengan lebih baik dan sempurna”.

2.4 Android

Murtiwiwati & Lauren (2013:2) menjelaskan bahwa Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, middleware dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, dan Nvidia.

2.5 Metode Pengembangan Perangkat Lunak

Binanto (2010:259) menjelaskan bahwa metodologi pengembangan multimedia terdiri dari enam tahap, yaitu *concept* (pengonsepan), *design* (pendesainan), *material collecting* (pengumpulan materi), *assembly* (pembuatan), *testing* (pengujian), dan *distribution* (pendistribusian). Seperti yang terlihat pada gambar 2.1 di bawah ini.



Gambar 2.1 Tahapan Pengembangan Multimedia

Penjelasan dari gambar 2.1 adalah sebagai berikut :

1. *Concept*

Tahap *concept* (pengkonsepan) adalah tahapan untuk menentukan tujuan dan siapa pengguna program (identifikasi *audience*). Karakteristik pengguna termasuk kemampuan pengguna juga perlu dipertimbangkan karena dapat memengaruhi pembuatan desain.

2. *Design*

Design (perancangan) adalah tahap membuat spesifikasi mengenai arsitektur program, gaya, tampilan dan kebutuhan material/bahan untuk program. Spesifikasi dibuat serinci mungkin sehingga pada tahap berikutnya, yaitu material collecting dan assembly, pengambilan keputusan baru tidak diperlukan lagi, cukup menggunakan keputusan yang sudah ditentukan pada tahap ini.

3. *Material Collecting*

Material collecting adalah tahap pengumpulan bahan yang sesuai dengan kebutuhan yang dikerjakan. Bahan-bahan tersebut, antara lain gambar, foto, animasi, video, audio, dan lain-lain. Tahap ini dapat dikerjakan secara paralel dengan tahap *assembly*.

4. *Assembly*

Tahap *assembly* adalah tahap dimana semua objek atau bahan multimedia dibuat. Pembuatan aplikasi didasarkan pada tahap design yang telah dikerjakan sebelumnya.

5. *Testing*

Dilakukan setelah menyelesaikan tahap pembuatan (*assembly*) dengan menjalankan aplikasi/program dan dilihat apakah ada kesalahan atau tidak. Tahap ini disebut juga sebagai tahap pengujian alpha (*alpha test*) dimana pengujian dilakukan oleh pembuat atau lingkungan pembuatnya sendiri.

6. *Distribution*

Tahapan dimana aplikasi disimpan dalam suatu media penyimpanan di internet. Pada tahap ini jika media penyimpanan tidak cukup untuk menampung aplikasinya, maka dilakukan kompresi terhadap aplikasi tersebut.

2.6 Perangkat Lunak Pengembangan Sistem

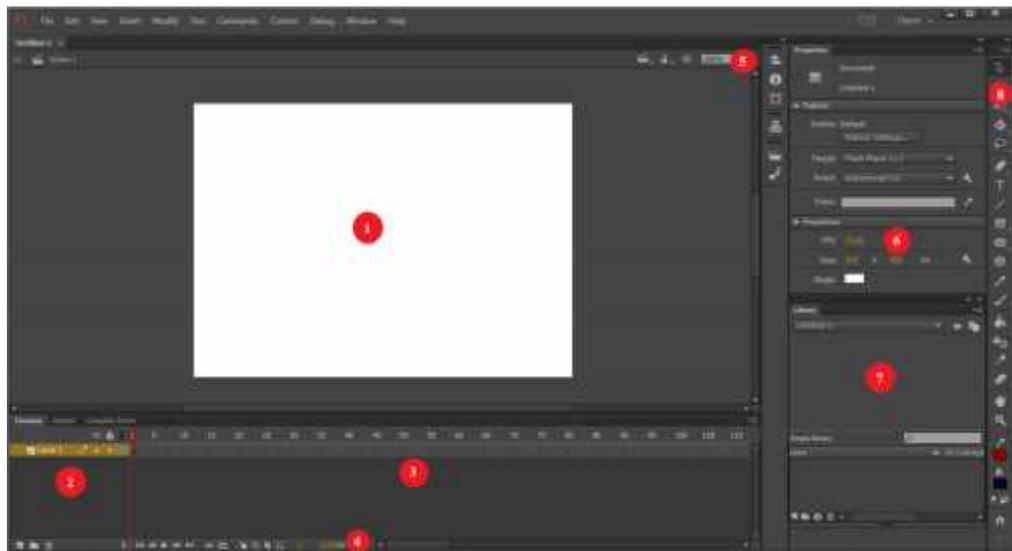
Aplikasi media pembelajaran matematika diskrit berbasis Android, diperlukan beberapa jenis perangkat lunak, yaitu sebagai berikut :

2.6.1 Adobe Flash Profesional

Purwandari, Anggriani & Noperman (2016:272) menjelaskan bahwa *Adobe Flash Profesional* adalah salah satu *software* dari perusahaan *Adobe* yang digunakan untuk membuat animasi, *banner* iklan, *web site*, presentasi, *game*, pembelajaran dan lain sebagainya. Selain itu flash juga dapat dikombinasikan dengan program yang lain, misalnya grafis seperti *AutoCAD*, *Photoshop*, *Camtasia* dan lain sebagainya. Kehandalan *Adobe Flash Profesional* dibandingkan dengan program lain adalah dalam hal ukuran *file* dari hasil animasinya yang kecil, untuk animasi yang dihasilkan oleh program *Adobe Flash Profesional* banyak digunakan untuk membuat sebuah *web* agar menjadi tampil lebih interaktif

1. Antarmuka Adobe Flash Profesional

Binanto (2010:232) menjelaskan bahwa antarmuka *Adobe Flash Profesional* seperti tampilan pada gambar 2.2 berikut ini:



Gambar 2.2 Antarmuka Adobe Flash Profesional

Penjelasan dari gambar 2.2 adalah sebagai berikut :

1. Stage

Stage adalah tempat untuk membuat objek/jendela proyek. Di sinilah letak gambar-gambar yang dianimasikan.

2. Layer

Layer digunakan untuk mempermudah dalam memilah objek maupun animasi yang sedang dibuat.

3. Frame/Time line

Frame ini digunakan untuk membuat animasi maupun mengatur kecepatan masing-masing klip movie/objek yang diletakkan pada frame tersebut.

4. Frame rate

Frame rate memiliki satuan fps atau frame per second sehingga kita dapat mengatur jumlah frame yang dapat dibaca setiap detiknya dari menu ini.

5. Scale Project Windows

Mempermudah proses pembuatan maupun perbaikan pada objek yang dibuat, kita dapat memperbesar maupun memperkecil tampilan jendela proyek dengan mengatur ukuran jendela proyek.

6. Setting Project Windows/Publish

Menu ini digunakan untuk mengatur ukuran jendela proyek, background jendela proyek.

7. Library, Color mixer, dan lain-lain

Objek, shape, klip movie, maupun tombol apa saja yang telah dibuat bisa kita dapatkan di jendela library. Sedangkan control mixer digunakan untuk memanipulasi warna pada objek secara lebih mendetail.

8. Drawing tool bar

Pada tool ini kita dapat melakukan proses memanipulasi pada objek yang ingin kita buat. Selain itu, kita juga dapat membuat objek dari shape maupun garis, sekaligus memainkan warna pada objek tersebut.

2. *Action Script*

Widiartha, Kendengis & Radion (2016:2) menjelaskan bahwa Adobe *ActionScript* merupakan bahasa pemrograman yang bekerja di dalam platform Adobe Flash, *Adobe ActionScript* dibangun untuk mengembangkan pemrograman interaktif secara efisien menggunakan platform aplikasi *Adobe Flash*. Pertama kali dikenalkan dalam Flash Player 9, *ActionScript* merupakan bahasa pemrograman berorientasi objek didasarkan pada ECMAScript-standar yang menjadi dasar JavaScript dan memberikan hasil yang luar biasa dalam kinerja dan produktifitas pengembang.

ActionScript 3.0 adalah bahasa *object-oriented programming* yang kuat dan menandakan sebuah langkah penting dalam kemajuan kemampuan *runtime Flash*. *ActionScript* dijalankan oleh *ActionScript Virtual Machine (AVM)* yang dijalankan pada Flash Player. *Action Script* hanya dapat dituliskan pada objek yang bertipe *Movie Clip*, *keyframe*, *Button*, dan objek *components*. *Action Script* tidak dapat digunakan pada objek tulisan atau gambar lain yang bukan bertipe *Movie Clip*. Jadi bila ingin menggunakan *Action Script* pada suatu objek, objek tersebut harus diubah menjadi *Movie Clip* terlebih dahulu.

2.6.2 **Adobe Premiere Pro**

Bentelu, Sentinuwo & Lantang (2016:4) menjelaskan bahwa Adobe Premiere adalah salah satu software yang populer dan digunakan secara luas dalam pengeditan video. Adobe Premiere merupakan program yang sudah umum digunakan oleh rumah-rumah produksi, televisi dan praktisi di bidangnya. Keuntungan belajar melakukan edit video menggunakan Fungsi utama Adobe Premiere Pro lebih untuk merangkai gambar, video dan audio, bukan untuk animasi. Agar penampilan multimedia anda lebih menarik.

2.6.3 Adobe AIR

Widiartha, Kendengis & Radion (2016:2) menjelaskan bahwa Adobe Integrated Runtime atau juga lebih sering diketahui dengan nama Adobe AIR adalah runtime lintas platform yang dikembangkan oleh Adobe Systems untuk membangun Rich Internet Application (RIA) dengan menggunakan Adobe Flash. Apache Flex (sebelumnya disebut Adobe Flex), HTML, dan Ajax, yang dapat dijalankan sebagai aplikasi desktop atau aplikasi mobile. Runtime ini mendukung aplikasi-aplikasi Windows, Mac OS dan beberapa sistem operasi android seperti BlackBerry Tablet OS, iOS, dan Android. Sebelumnya Adobe AIR juga mendukung Linux, tetapi dukungan untuk Linux sudah diberhentikan.

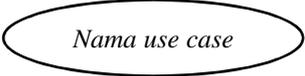
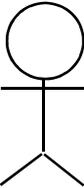
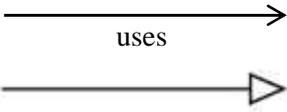
2.7 *Unified Modeling Language (UML)*

Aprianti & Maliha (2016:22) menjelaskan bahwa *Unified Modeling Language (UML)* adalah standarisasi bahasa pemodelan untuk membangun perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek. Diagram - diagram yang digunakan pada UML antara lain adalah *Class Diagram*, *Use Case Diagram*, *Activity Diagram*, dan *Sequence Diagram*.

2.7.1 *Use Case Diagram*

Aprianti & Maliha (2016:23) menguraikan bahwa *Use case diagram* menggambarkan interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Simbol - simbol *Use Case* dapat dilihat pada tabel 2.1.

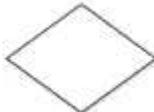
Tabel 2.1 Simbol *Use Case Diagram*

Simbol	Keterangan
<p data-bbox="549 495 660 524"><i>Use Case</i></p> 	<p data-bbox="815 506 1251 607">Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit dan aktor.</p>
<p data-bbox="533 689 676 719">Aktor/<i>Actor</i></p> 	<p data-bbox="815 734 1251 869">Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi.</p>
<p data-bbox="485 981 724 1010"><i>Asosiasi/association</i></p> 	<p data-bbox="815 981 1251 1048">Komunikasi antar aktor dan <i>Use Case</i> yang berpartisipasi.</p>
<p data-bbox="512 1122 692 1151">Ekstensi/<i>extend</i></p> <p data-bbox="533 1173 676 1202"><<extend>></p> 	<p data-bbox="815 1115 1251 1249">Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambah dapat berdiri sendiri walau tanpa <i>Use Case</i> tambahan.</p>
<p data-bbox="443 1308 772 1337"><i>Generalisasi/generalization</i></p> 	<p data-bbox="815 1285 1251 1420">Hubungan generalisasi dan spesialisasi antara dua buah <i>Use Case</i> yang mana fungsi yang satu lebih umum dari yang lainnya.</p>
<p data-bbox="501 1458 708 1487"><i>Include/Use Case</i></p> <p data-bbox="533 1509 676 1538"><<include>></p> <p data-bbox="580 1561 628 1590">uses</p> 	<p data-bbox="815 1464 1251 1644">Relasi <i>Use Case</i> tambahan ke sebuah <i>Use Case</i> dimana <i>Use Case</i> yang ditambahkan memerlukan <i>Use Case</i> ini untuk menjalankan fungsinya.</p>

2.7.2 Activity Diagram

Aprianti & Maliha (2016:23) menguraikan bahwa *Activity Diagram* menggambarkan rangkaian aliran kerja dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Simbol – simbol *Activity Diagram* dapat dilihat pada tabel 2.2.

Tabel 2.2. Simbol *Activity Diagram*

Simbol	Keterangan
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Status awal 	Bagaimana objek dibentuk atau diawali
Status akhir 	Bagaimana objek dibentuk dan diakhiri.
Percabangan/ <i>join</i> 	Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu.
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

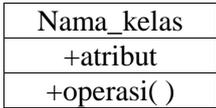
2.7.3 Sequence Diagram

Aprianti & Maliha (2016:23) menguraikan bahwa *Sequence Diagram* menggambarkan interaksi objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek.

2.7.4 Class Diagram

Aprianti & Maliha (2016:22) menguraikan bahwa *Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan di buat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Simbol-simbol yang ada pada *Class diagram* ditunjukkan oleh Tabel 2.3.

Tabel 2.3 Simbol *Class Diagram*

Simbol	Deskripsi
<p>Kelas</p> 	Kelas pada struktur system
<p>Antarmuka/<i>interface</i></p>  <p>Nama_interface</p>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
<p>Asosiasi/<i>association</i></p> 	Relasi antarkelas dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .
<p>Asosiasi berarah/<i>directed association</i></p> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
<p>Generalisasi</p> 	Relasi antarkelas dengan makna Generalisasi-spesialisasi (umum khusus).
<p>Kebergantungan/<i>dependency</i></p> 	Relasi antarkelas dengan makna Kebergantungan antarkelas
<p>Agregasi/<i>aggregation</i></p> 	Relasi antar kelas dengan makna semua-bagian

2.8 Penelitian Terdahulu

Tabel 2.4 berikut merupakan penelitian yang terkait dengan aplikasi pembelajaran berbasis android :

Tabel 2.4 Penelitian Terdahulu

No	Nama	Judul	Terbit / Tahun	Keterangan
1.	Raka Tamagola dan Puput Budi Wintoro	Visualisasi 3D Aset Kendaraan Tempur Brigade Infanteri 3 Marinir Lampung Berbasis Android	Prosiding Seminar Nasional IIB Darmajaya / 2017	Penelitian ini membahas mengenai visualisasi 3D aset kendaraan tempur yang dirancang sebagai alat bantu untuk memperkenalkan aset kendaraan tempur darat dan laut yang dimiliki oleh Brigade Infanteri (Brigif) 3 Korps Marinir (TNI AL) Lampung kepada masyarakat umum. Metode pengembangan yang digunakan untuk mengembangkan aplikasi ini yaitu metode Luther Sutopo (1994).
2.	Ahmad Farhan Afifi dan Amnah	Pemanfaatan Media Pembelajaran Berbasis Cloud Computing Menggunakan Edmodo Pada SMA Al-Kautsar	Prosiding Seminar Nasional IIB Darmajaya / 2017	Penelitian ini membahas mengenai pemanfaatan metode edmodo untuk media pembelajaran cloud computing untuk membantu dalam proses belajar – mengajar. Keunggulan dari Edmodo adalah menyediakan fasilitas yang mudah dan aman dalam mengembangkan kelas sesuai dengan keinginan. Dengan Fasilitas Edmodo memberi kesempatan komunikasi bagi guru, siswa dan orang tua/wali murid secara personal.

Tabel 2.4 (Lanjutan)

No	Nama	Judul	Terbit / Tahun	Keterangan
3.	Septilia Arfida dan Rahman E. Harahap	Implementasi Media Pembelajaran Teknik Pengkodean Barcode Berbasis Multimedia Dalam Meningkatkan Kualitas Kegiatan Belajar Mengajar	Prosiding Seminar & Bisnis Teknologi IIB Darmajaya /2014	Penelitian ini membahas mengenai aplikasi media pembelajaran teknik pengkodean barcode berbasis multimedia untuk membantu guru dalam meningkatkan kualitas kegiatan belajar mengajar sehingga menjadi lebih menarik bagi para pelajar. Karena dapat memberikan materi pelajaran secara interaktif dengan didukung berbagai aspek seperti audio, video, animasi, teks, dan grafik. Metode pengembangan perangkat lunak media pembelajaran ini menggunakan Siklus Hidup Pengembangan Multimedia.
4.	Ossy Dwi Endah Wulansari, TM Zaini, dan Bobby Bahri	Penerapan Teknologi <i>Augmented Reality</i> Pada Media Pembelajaran	Jurnal informatika /2013	Penelitian ini membahas mengenai Pemanfaatan Aplikasi grafis komputer yang memungkinkan mengilustrasikan gambar - gambar dalam bentuk animasi (gambar bergerak dan tiga dimensi) yang dapat dieksplorasi dalam berbagai sudut pandang (atas,bawah, kanan, kiri) sehingga akan memberikan pemahaman yang lebih baik kepada <i>user</i> . Penerapan aplikasi komputer ini dikembangkan menggunakan salah satu metode pengembangan perangkat lunak yaitu "waterfall".

Tabel 2.4 (Lanjutan)

No	Nama	Judul	Terbit / Tahun	Keterangan
5.	Nurul Huda dan Fitri Purwaningtias	Pemanfaatan Teknologi Augmented Reality (AR) Pembelajaran Matematika Menggunakan 3 (tiga) Bahasa Pada Tingkat Sekolah Dasar Berbasis Android	Prosiding Seminar Nasional IIB Darmajaya / 2017	Penelitian ini membahas mengenai pemanfaatan teknologi augmented reality untuk pembelajaran matematika yang memanfaatkan telepon genggam dengan menggunakan 3 bahasa yaitu bahasa indonesia, bahasa inggris, dan bahasa arab. Dengan tujuan bisa membuat siswa menjadi lebih tertarik untuk mempelajari mata pelajaran matematika. Metode yang di pakai dalam penelitian ini adalah metode waterfall