

**RANCANG BANGUN *GAME* EDUKASI  
UNTUK PENGENALAN DASAR ALGORITMA  
BERBASIS *ANDROID***

**SKRIPSI**



**Oleh:  
Shella  
160210021**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FALKUTAS TEKNIK DAN KOMPUTER  
UNIVERSITAS PUTERA BATAM  
TAHUN 2020**

**RANCANG BANGUN *GAME* EDUKASI  
UNTUK PENGENALAN DASAR ALGORITMA  
BERBASIS *ANDROID***

**SKRIPSI**

**Untuk memenuhi salah satu syarat  
memperoleh gelar Sarjana**



**Oleh:  
Sheila  
160210021**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FALKUTAS TEKNIK DAN KOMPUTER  
UNIVERSITAS PUTERA BATAM  
TAHUN 2020**

## **SURAT PERNYATAAN ORISINALITAS**

Yang bertanda tangan di bawah ini saya:

Nama : Shella  
NPM : 160210021  
Fakultas : Teknik dan Komputer  
Program Studi : Teknik Informatika

Menyatakan bahwa “**Skripsi**” yang saya buat dengan judul:

### **RANCANG BANGUN *GAME* EDUKASI UNTUK PENGENALAN DASAR ALGORITMA BERBASIS *ANDROID***

Adalah hasil karya sendiri dan bukan “duplikasi” dari karya orang lain. Sepengetahuan saya, didalam naskah Skripsi ini tidak terdapat karya ilmiah atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip didalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah Skripsi ini dapat dibuktikan terdapat unsur - unsur PLAGIASI, saya bersedia naskah Skripsi ini digugurkan dan gelar akademik yang saya peroleh dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku.

Demikian pernyataan ini saya buat dengan sebenarnya tanpa ada paksaan dari siapapun

Batam, 15 Februari 2020

Materai 6000

**Shella**  
160210021

**RANCANG BANGUN *GAME* EDUKASI  
UNTUK PENGENALAN DASAR ALGORITMA  
BERBASIS *ANDROID***

**SKRIPSI**

**Untuk memenuhi salah satu syarat  
memperoleh gelar Sarjana**

**Oleh:  
Shella  
160210021**

**Telah disetujui oleh Pembimbing pada tanggal  
seperti tertera di bawah ini**

**Batam, 12 Februari 2020**

**Hotma Pangaribuan, S.Kom., M.SI  
Pembimbing**

## ABSTRAK

Dalam dunia pemrograman, algoritma adalah hal utama bagi *programmer* dalam mendesain dan mengembangkan aplikasi *software*. Namun, tidak semua orang dapat memahami algoritma karena dalam memahami algoritma biasanya menggambar konsep yang sulit dan rumit untuk memecahkan masalah yang kompleks. Media pembelajaran dalam mengenalkan algoritma masih sebatas teori yang dirasa kurang menarik dan tidak mudah memahami materi yang disampaikan. Untuk itu diperlukan sebuah pembaharuan sistem pembelajaran yang dapat menarik minat para pelajar dan mempermudah pendidik dalam mengenalkan materi algoritma. Pembuatan penelitian ini bertujuan untuk merancang sebuah *game* edukasi dan menambah pengetahuan dalam mengembangkan *game* berbasis *android* sebagai media pembelajaran, memberikan hiburan *game* yang menarik dan menambah wawasan tentang algoritma. Penelitian ini menggunakan metode penelitian dan pengembangan (*Research and Development*) dengan model pengembangan versi Luther yang dimulai dari mencari potensi dan masalah yang kemudian dianalisis diiringi pengumpulan data-data guna mendukung proses pengembangan *game*. Selanjutnya tahap perancangan melalui model pengembangan versi Luther dilakukan untuk mendapatkan hasil akhir dari penelitian ini. Dalam mengumpulkan data, metode yang dilakukan studi pustaka dan wawancara. Hasil penelitian ini berupa aplikasi *game* edukasi untuk pengenalan dasar algoritma yang telah diuji melalui pengujian *black box*, dengan hasil pengujian semua indikator dinyatakan baik. Dengan adanya aplikasi *game* edukasi untuk pengenalan dasar algoritma ini pelajar dapat terbantu dalam memahami materi algoritma.

Kata kunci: Algoritma, *Android*, *Game* Edukasi.

## **ABSTRACT**

*Algorithms are the main thing for programmers in designing and developing software applications. However, not everyone can understand the algorithm because in understanding the algorithm usually describes complicated concepts to solve complex problems. Learning media in introducing algorithms is still limited to theories that are felt to be less interesting and hard to understand. For these various conditions, it is necessary to update the learning system that can attract students's interest and facilitate educators to introduce the concept of algorithms. The purpose of this study aims to design educational games and increase knowledge in developing Android-based games as learning media, providing interesting game entertainment and increasing knowledge about algorithms. This study uses research and development methods with Luther's version of the development model that starts from looking for potential and problems which are analyzed and then collects data to support the game development process. Furthermore, the design stage through the Luther version of the development model was carried out to obtain the final results of this study. In collecting data, methods are carried out using literature studies and interviews. The results of this study in the form of applications that have been tested through black box testing, with the results of testing all the indicators declared good. With the educational game application, students can be helped in understanding the concepts of the algorithm.*

*Keywords: Algorithms; Android; Educational Games.*

## KATA PENGANTAR

Segala puji bagi Tuhan Yang Maha Esa yang telah melimpahkan segala rahmat dan karuniaNya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada Program Studi Teknik Informatika di Universitas Putera Batam.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Karena itu, kritik dan saran akan senantiasa penulis terima dengan senang hati. Dengan segala keterbatasan, penulis menyadari pula bahwa skripsi ini takkan terwujud tanpa bantuan, bimbingan, dan dorongan dari berbagai pihak. Untuk itu, dengan segala kerendahan hati, penulis menyampaikan ucapan terima kasih kepada:

1. Rektor Universitas Putera Batam;
2. Dekan Fakultas Teknik dan Komputer;
3. Ketua Program Studi Teknik Informatika;
4. Bapak Hotma Pangaribuan, S.Kom., M.SI selaku pembimbing Skripsi pada Program Studi Teknik Informatika Universitas Putera Batam;
5. Bapak Andi Maslam, S.T., M.SI selaku pembimbing akademik sejak semester pertama hingga semester tujuh;
6. Dosen dan Staff Universitas Putera Batam;
7. SMK Kolese Tiara Bangsa yang telah memberikan izin;
8. Semua teman dan rekan-rekan Kampus Universitas Putera Batam;
9. Teristimewa kepada Orang Tua dan keluarga penulis yang selalu mendoakan, memberikan motivasi dan dukungan kepada penulis sehingga penulis dapat menyelesaikan skripsi ini;
10. Teman-teman seperjuangan Suwarti, Ayna, Ersiska, Daniel, Fetri, dan Afrishella yang juga selalu memberikan motivasi baik berupa *sharing* pendapat, motivasi dan hal lainnya dalam rangka pembuatan skripsi ini.
11. Terima kasih juga kepada semua pihak yang telah membantu dalam penyelesaian skripsi ini yang tidak dapat disebutkan satu per satu;

Semoga Tuhan Yang Maha Esa membalas kebaikan dan selalu mencurahkan hidayah serta taufikNya, Amin.

Batam, 12 Februari 2020

Sheila

## DAFTAR ISI

	Halaman
<b>HALAMAN SAMPUL</b> .....	i
<b>HALAMAN JUDUL</b> .....	ii
<b>SURAT PERNYATAAN</b> .....	iii
<b>HALAMAN PENGESAHAN</b> .....	iv
<b>ABSTRAK</b> .....	v
<b>ABSTRACT</b> .....	vi
<b>KATA PENGANTAR</b> .....	vii
<b>DAFTAR ISI</b> .....	viii
<b>DAFTAR GAMBAR</b> .....	x
<b>DAFTAR TABEL</b> .....	xii
<b>BAB I PENDAHULUAN</b> .....	1
1.1. Latar Belakang .....	1
1.2. Identifikasi Masalah .....	4
1.3. Batasan Masalah .....	4
1.4. Rumusan Masalah .....	5
1.5. Tujuan .....	5
1.6. Manfaat .....	5
1.6.1. Manfaat Teoritis .....	6
1.6.2. Manfaat Praktis .....	6
<b>BAB II TINJAUAN PUSTAKA</b> .....	7
2.1. Teori Dasar .....	7
2.1.1. <i>Game</i> .....	7
2.1.2. <i>Game</i> Edukasi .....	8
2.1.3. Algoritma .....	10
2.1.4. <i>Android</i> .....	11
2.1.5. <i>Unified Modeling Language (UML)</i> .....	12
2.2. Variabel .....	17
2.3. Software Pendukung .....	23
2.3.1. <i>Unity</i> .....	23
2.3.2. <i>Blender</i> .....	24
2.3.3. <i>Adobe Illustrator</i> .....	24
2.3.4. <i>Microsoft Visio</i> .....	25
2.4. Penelitian Terdahulu .....	25
2.5. Kerangka Pemikiran .....	29
<b>BAB III METODE PENELITIAN</b> .....	31
3.1. Desain Penelitian .....	31
3.2. Pengumpulan Data .....	32
3.2.1. Data Primer .....	33
3.2.2. Data Sekunder .....	33
3.3. Operasional Variabel .....	33
3.4. Metode Perancangan Sistem .....	34
3.4.1. <i>Concept</i> (Konsep) .....	35

3.4.2. <i>Design</i> (Perancangan).....	36
3.4.3. <i>Material Collecting</i> (Pengumpulan Bahan).....	50
3.4.4. <i>Assembly</i> (Pembuatan).....	51
3.4.5. <i>Testing</i> (Pengujian).....	51
3.4.6. <i>Distribution</i> (Pendistribusian) .....	51
3.5. Lokasi dan Jadwal Penelitian.....	51
<b>BAB IV HASIL DAN PEMBAHASAN</b> .....	53
4.1. Hasil Penelitian.....	53
4.2. Pembahasan .....	79
<b>BAB V PENUTUP</b> .....	100
5.1. Simpulan.....	100
5.2. Saran .....	100
<b>DAFTAR PUSTAKA</b> .....	102
<b>DAFTAR RIWAYAT HIDUP</b> .....	104
<b>LAMPIRAN</b> .....	104

## DAFTAR GAMBAR

	Halaman
<b>Gambar 2.1</b> Bagan Kerangka Pemikiran .....	29
<b>Gambar 3.1</b> Desain Penelitian .....	31
<b>Gambar 3.2</b> Model Pengembangan Multimedia versi Luther .....	34
<b>Gambar 3.3</b> <i>Flowchart</i> secara umum .....	36
<b>Gambar 3.4</b> <i>Flowchart</i> Materi Struktur Sekuensial.....	37
<b>Gambar 3.5</b> <i>Flowchart</i> Materi Struktur Perulangan.....	38
<b>Gambar 3.6</b> <i>Flowchart</i> Materi Struktur Seleksi .....	39
<b>Gambar 3.7</b> Diagram <i>use case</i> Secara Umum .....	40
<b>Gambar 3.8</b> Diagram <i>use case</i> Struktur Sekuensial .....	41
<b>Gambar 3.9</b> Diagram <i>use case</i> Struktur Perulangan.....	41
<b>Gambar 3.10</b> Diagram <i>use case</i> Struktur Seleksi .....	42
<b>Gambar 3.11</b> Diagram Sekuensial Secara Umum .....	42
<b>Gambar 3.12</b> Diagram Sekuensial Pemilihan Materi .....	43
<b>Gambar 3.13</b> Diagram Sekuensial Menu Pengaturan.....	43
<b>Gambar 3.14</b> Diagram Sekuensial Menu Tentang .....	44
<b>Gambar 3.15</b> Diagram Aktivitas Secara Umum.....	44
<b>Gambar 3.16</b> Diagram Aktivitas Menu Pengaturan .....	45
<b>Gambar 3.17</b> Diagram Aktivitas Menu Tentang .....	45
<b>Gambar 3.18</b> Diagram Aktivitas Pemilihan Materi.....	46
<b>Gambar 3.9</b> Diagram Kelas .....	47
<b>Gambar 3.20</b> Rancangan Antarmuka Halaman Splash .....	47
<b>Gambar 3.21</b> Rancangan Antarmuka Menu Utama.....	48
<b>Gambar 3.22</b> Rancangan Antarmuka Menu Permainan .....	48
<b>Gambar 3.23</b> Rancangan Antarmuka Pemilihan Level .....	48
<b>Gambar 3.24</b> Rancangan Antarmuka Permainan Materi Sekuensial .....	49
<b>Gambar 3.25</b> Rancangan Antarmuka Permainan Materi Seleksi Level 1-4.....	49
<b>Gambar 3.26</b> Rancangan Antarmuka Permainan Materi Seleksi Level 5-7.....	49
<b>Gambar 3.27</b> Rancangan Antarmuka Permainan Materi Seleksi Level 8-10.....	50
<b>Gambar 3.28</b> Rancangan Antarmuka Permainan Materi Perulangan.....	50
<b>Gambar 3.29</b> Peta Lokasi Batam .....	52
<b>Gambar 4.1</b> Tampilan <i>Splash Screen</i> .....	54
<b>Gambar 4.2</b> Tampilan Menu Utama .....	54
<b>Gambar 4.3</b> Tampilan Menu Pengaturan.....	55
<b>Gambar 4.4</b> Tampilan Tentang Game .....	56
<b>Gambar 4.5</b> Tampilan Pemilihan Materi .....	56
<b>Gambar 4.6</b> Menu Level Materi Sekuensial .....	57
<b>Gambar 4.7</b> Tampilan Menu Informasi Sekuensial.....	58
<b>Gambar 4.8</b> Tampilan Informasi Setelah Salah Satu Tombol Ditekan .....	58
<b>Gambar 4.9</b> Tampilan Awal Permainan Materi Sekuensial .....	59
<b>Gambar 4.10</b> Tampilan Ketika Permainan Berhenti Materi Sekuensial .....	59
<b>Gambar 4.11</b> Tampilan Permainan Sedang Berlansung Materi Sekuensial.....	60
<b>Gambar 4.12</b> Tampilan Informasi Berhasil Materi Sekuensial .....	61

<b>Gambar 4.13</b>	Tampilan Informasi Jatuh Materi Sekuensial .....	61
<b>Gambar 4.14</b>	Menu Level Materi Seleksi.....	62
<b>Gambar 4.15</b>	Tampilan Informasi Materi Seleksi 1 .....	63
<b>Gambar 4.16</b>	Tampilan Informasi Materi Seleksi 2 .....	63
<b>Gambar 4.17</b>	Tampilan Informasi Materi Seleksi 3 .....	64
<b>Gambar 4.18</b>	Tampilan Menu Informasi Seleksi 4.....	64
<b>Gambar 4.19</b>	Tampilan Menu Informasi Seleksi 5.....	64
<b>Gambar 4.20</b>	Tampilan Informasi Setelah Salah Satu Tombol Ditekan .....	65
<b>Gambar 4.21</b>	Tampilan Awal Permainan Materi Seleksi.....	65
<b>Gambar 4.22</b>	Tampilan Permainan Setelah Soal Ditutup Level 1 - 4 .....	66
<b>Gambar 4.23</b>	Tampilan Ketika Permainan Berhenti Materi Seleksi .....	67
<b>Gambar 4.24</b>	Tampilan Informasi Blok If Belum Tepat .....	68
<b>Gambar 4.25</b>	Tampilan Informasi Blok Then Belum Tepat.....	68
<b>Gambar 4.26</b>	Tampilan Informasi Blok Input Aksi Belum Benar .....	69
<b>Gambar 4.27</b>	Tampilan Permainan Setelah Soal Ditutup Level 5 - 7 .....	69
<b>Gambar 4.28</b>	Tampilan Informasi Blok Else Belum Benar.....	70
<b>Gambar 4.29</b>	Tampilan Informasi Kondisi Bernilai Benar Beda Dengan Soal... 71	
<b>Gambar 4.30</b>	Tampilan Informasi Kondisi Bernilai Salah Beda Dengan Soal ... 72	
<b>Gambar 4.31</b>	Tampilan Permainan Setelah Soal Ditutup Level 8 - 10 .....	72
<b>Gambar 4.32</b>	Tampilan Informasi Blok Else If Then Belum Benar.....	73
<b>Gambar 4.33</b>	Menu Level Materi Perulangan .....	74
<b>Gambar 4.34</b>	Tampilan Informasi Materi Perulangan 1 .....	75
<b>Gambar 4.35</b>	Tampilan Informasi Materi Perulangan 2.....	75
<b>Gambar 4.36</b>	Tampilan Informasi Setelah Salah Satu Tombol Ditekan .....	76
<b>Gambar 4.37</b>	Tampilan Awal Permainan Materi Perulangan.....	77
<b>Gambar 4.38</b>	Tampilan Permainan Sedang Berlangsung Materi Perulangan ....	78
<b>Gambar 4.39</b>	Tampilan Ketika Permainan Berhenti Materi Perulangan.....	78
<b>Gambar 4.40</b>	Tampilan Informasi Berhasil Level 10 Materi Perulangan .....	79

## DAFTAR TABEL

	Halaman
<b>Tabel 2.1</b> Simbol Notasi dan Relasi dalam Diagram <i>Use Case</i> .....	14
<b>Tabel 2.2</b> Simbol Notasi dan Relasi dalam Diagram Sekuen.....	14
<b>Tabel 2.3</b> Simbol Notasi dan Relasi dalam Diagram Kelas .....	15
<b>Tabel 2.4</b> Simbol Notasi dan Relasi dalam Diagram Aktivitas.....	16
<b>Tabel 3.1</b> Operasional Variabel Dalam Penelitian .....	34
<b>Tabel 3.2</b> Perencanaan Pelaksanaan Jadwal Penelitian .....	52
<b>Tabel 4.1</b> Pengujian Menu Utama .....	80
<b>Tabel 4.2</b> Pengujian Menu Pengaturan.....	81
<b>Tabel 4.3</b> Isi Menu Tentang.....	81
<b>Tabel 4.4</b> Pengujian Menu Pemilihan Materi.....	82
<b>Tabel 4.5</b> Pengujian Menu Pemilihan Level Sekuensial .....	83
<b>Tabel 4.6</b> Pengujian Menu Pemilihan Level Seleksi .....	84
<b>Tabel 4.7</b> Pengujian Menu Pemilihan Level Perulangan .....	85
<b>Tabel 4.8</b> Pengujian Menu Informasi Materi Sekuensial .....	86
<b>Tabel 4.9</b> Pengujian Permainan Sekuensial .....	87
<b>Tabel 4.10</b> Pengujian Permainan Seleksi Level 1 - 4.....	89
<b>Tabel 4.11</b> Pengujian Permainan Seleksi Level 5 - 7 .....	90
<b>Tabel 4.12</b> Pengujian Permainan Seleksi Level 5 - 7.....	93
<b>Tabel 4.13</b> Pengujian Permainan Perulangan.....	97

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Teknologi terus berkembang dan sangat mempermudah kehidupan manusia. Salah satu teknologi tersebut adalah *smartphone* yang memiliki fitur-fitur yang setara dengan komputer, sehingga seringkali *smartphone* disebut sebagai komputer genggam yang dapat digunakan dimana saja tanpa terbatas pada waktu dan tempat. Banyak aktivitas yang dapat dilakukan melalui *smartphone* seperti berkirim pesan, melakukan panggilan, serta mendapatkan hiburan dari aplikasi sosial media dan *game*. Pengguna *smartphone* juga seringkali memanfaatkan *game* sebagai sarana hiburan. Selain ditujukan sebagai hiburan, *game* sudah mulai digunakan sebagai media pembelajaran dalam mengenalkan materi untuk meningkatkan daya pikir dan logika pengguna.

*Game* merupakan salah satu hasil perkembangan teknologi dari pemikiran manusia yang didasari oleh logika dan pemahaman algoritma yang diimplementasikan ke dalam bahasa pemrograman. Dalam buku yang berjudul “Algoritma dan Pemrograman” (Sitorus, 2015) menjelaskan algoritma merupakan suatu urutan tahapan / langkah-langkah logis dalam menyelesaikan suatu masalah yang disusun secara sistematis dan berdasarkan logika (benar atau salah). Logika dapat diartikan bahwa manusia berpikir dengan akal tentang suatu masalah dan menghasilkan sebuah kebenaran yang dapat dibuktikan dan masuk akal. Oleh

karena itu logika dan algoritma merupakan 2 hal yang saling berkaitan dan tidak dapat dipisahkan.

Dalam dunia pemrograman, algoritma adalah hal utama bagi *programmer* dalam mendesain dan mengembangkan *game*. Sebuah *game* terdiri atas kumpulan-kumpulan program dan sintaks. Untuk menyusun program atau sintaks, langkah logis diperlukan dan diatur secara sistematis untuk memecahkan suatu masalah atau tujuan dalam kegiatan perancangan perangkat lunak. Bila terdapat algoritma yang tidak sesuai, *game* dapat berubah tujuan atau bahkan gagal berfungsi. Oleh karena itu, algoritma memainkan peran penting dalam merancang program atau sintaks.

Namun, tidak semua orang dapat memahami algoritma karena dalam memahami algoritma biasanya menggambar konsep yang sulit dan rumit untuk memecahkan masalah yang kompleks. Abadi Nugroho mengemukakan bahwa bila algoritma dikenalkan dengan cara yang sulit dipahami dan terkesan membosankan maka dapat mengurangi minat dalam mempelajari algoritma itu sendiri (Nugroho, 2015). Akibatnya, para pendidik menghadapi kesulitan dalam mengajarkan algoritma yang membutuhkan banyak penjelasan dan ilustrasi. Media pembelajaran dalam mengenalkan algoritma masih sebatas teori yang membuat siswa merasa tidak menarik dan tidak memahami materi yang disampaikan. Alat bantu dalam kegiatan pembelajaran selalu dibutuhkan untuk dapat membantu siswa belajar dan memahami algoritma dengan lebih baik.

Selain itu, dalam beberapa kasus guru yang mengajarkan bahasa pemrograman memberikan contoh sintaks mengalami kendala. Para siswa mampu untuk menuliskan sintaks-sintaks dalam membuat program tersebut dengan melihat

kembali catatan mereka. Namun bila keluaran program yang diminta oleh guru dirubah sedikit, para siswa kebingungan dalam menuliskan sintaks program untuk menghasilkan keluaran program. Hal ini disebabkan karena para siswa menghafal sintaks program dan tidak memahami sintaks yang membutuhkan algoritma. Logika dan algoritma sangat penting untuk belajar pemrograman. Ini dikarenakan suatu program yang dibuat merupakan hasil dari adanya algoritma di dalamnya, sedangkan algoritma itu sendiri didapatkan dari logika pemikiran.

Faktanya, hampir semua remaja di Indonesia memiliki *smartphone* dengan sistem operasi *Android* (Nasution, Neviyarni, & Alizamar, 2017). Namun masih banyak siswa yang kurang memanfaatkan *smartphone* untuk belajar dan menambah wawasan pengetahuan. Para siswa tersebut menganggap *smartphone* digunakan untuk kegiatan pribadi dan sosial yang sifatnya menyenangkan. Hal ini juga dikarenakan masih sedikitnya *game* edukasi yang tersedia di *Playstore* yang dapat menarik minat siswa khususnya *game* yang mencakup materi dasar algoritma berbahasa Indonesia yang dapat dimengerti siswa.

Melihat berbagai kondisi tersebut, maka diperlukan sebuah pembaharuan sistem pembelajaran yang dapat menarik minat para siswa dan dapat mempermudah pendidik dalam mengenalkan materi. Dengan memanfaatkan teknologi yang sedang berkembang, *game* edukasi berbahasa Indonesia yang diimplementasikan dalam *smartphone Android* merupakan solusi yang tepat dan efisien dalam memperkenalkan dasar algoritma, dimana pengguna *Android* di kalangan pelajar sangat tinggi. Dalam hal ini, *game* edukasi tidak hanya memberikan hiburan, tetapi

juga menambah wawasan, serta membangun pemahaman mengenai algoritma dan meningkatkan kecerdasan logika.

Berdasarkan beberapa permasalahan yang disebutkan, peneliti tertarik untuk mengembangkan media pembelajaran melalui judul “**Rancang Bangun Game Edukasi Untuk Pengenalan Dasar Algoritma Berbasis *Android***”.

### **1.2. Identifikasi Masalah**

Adapun hasil identifikasi masalah berdasarkan penjelasan latar belakang tersebut berupa:

1. Alat bantu berupa media edukasi dalam mengenalkan materi dasar algoritma kepada para siswa dibutuhkan oleh pendidik.
2. Masih sedikit *game* edukasi yang tersedia di *Playstore* terutama *game* edukasi yang mengenalkan dasar algoritma berbahasa Indonesia berbasis *android*.
3. Pemanfaatan *smartphone* yang kurang maksimal. Masih sebagai hiburan dan belum diterapkan dalam sektor pendidikan khususnya dalam mengenal logika algoritma.
4. Sulitnya memahami materi algoritma.

### **1.3. Batasan Masalah**

Agar pembahasan penelitian ini lebih fokus dan terarah pada tujuan yang ingin dicapai, maka batasan masalah diberikan dalam penelitian ini, sebagai berikut:

1. Penelitian ini menggunakan *game* edukasi berbasis *Android*.
2. Penelitian ini menggunakan *tools Unity*.

3. Penelitian ini dilakukan di SMK Kolese Tiara Bangsa.
4. Penelitian ini menggunakan metode model pengembangan multimedia (*MDLC*) versi Luther.
5. Penelitian ini meliputi materi pengenalan dasar algoritma secara umum berupa struktur algoritma sekuensial, struktur perulangan, dan struktur seleksi dengan masing-masing 10 level per materi dari tingkat mudah ke tingkat sulit.
6. Penelitian *game* edukasi ini ditujukan khususnya siswa pada tingkat sekolah menengah kejuruan dan umumnya bagi masyarakat yang ingin memahami dasar logika dan algoritma.

#### **1.4. Rumusan Masalah**

Dalam penelitian ini, rumusan masalah yang peneliti angkat adalah sebagai berikut:

1. Bagaimana merancang dan membangun *game* edukasi untuk pengenalan dasar algoritma berbasis *android*?

#### **1.5. Tujuan**

Adapun tujuan dari hasil penelitian ini yang diharapkan sebagai berikut:

1. Merancang dan membangun *game* edukasi untuk pengenalan dasar algoritma berbasis *android*.

#### **1.6. Manfaat**

Adapun manfaat penelitian dalam pembuatan aplikasi *game* edukasi untuk pengenalan dasar algoritma berbasis *android* ini diharapkan dapat memberikan manfaat sebagai berikut:

### 1.6.1. Manfaat Teoritis

Manfaat secara teoritis yang diharapkan dari hasil penelitian sebagai berikut:

1. Dapat digunakan sebagai media belajar mengenai dasar algoritma.
2. Memberikan sumbangan pemikiran dalam pembuatan media pembelajaran serta menjadi bahan kajian lebih lanjut dengan tema sejenis.

### 1.6.2. Manfaat Praktis

Manfaat secara praktis yang diharapkan dari hasil penelitian sebagai berikut:

1. Bagi peneliti

Memperoleh wawasan baru dan pengalaman secara langsung tentang bagaimana membuat *game* edukasi dan meningkatkan kecerdasan logika melalui *game* edukasi yang mengenalkan dasar algoritma.

2. Bagi universitas / sekolah

Diharapkan dapat digunakan sebagai bahan pertimbangan dalam merencanakan kegiatan pembelajaran serta membantu dalam penyampaian informasi dalam mengenalkan dasar algoritma kepada pelajar.

3. Bagi pengguna

Mendapatkan rasa ingin tahu dan senang dalam menerima, memahami serta mempelajari dasar logika dan algoritma. Selain itu juga meningkatkan kecerdasan logika dan minat belajar serta tidak mudah merasa jenuh dalam kegiatan belajar dan lebih tertarik dalam menyelesaikan suatu permasalahan dengan algoritma.

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Teori Dasar**

##### **2.1.1. *Game***

Sudono dalam (Nama, Pamungkas, Mardiana, & Septama, 2019) mengungkapkan bermain merupakan suatu rangkaian kegiatan yang dapat memberikan informasi, hiburan, dan mengembangkan imajinasi pada anak yang dilakukan dengan atau tanpa menggunakan media. Menurut Arief (Sadiman, 2010), *game* adalah sesuatu yang bisa dimainkan dengan berdasarkan aturan tertentu sehingga ada yang menang dan ada yang kalah dalam konteks bukan serius dengan tujuan menghibur diri. *Game* merupakan sebuah aktivitas terstruktur yang merupakan hasil akhir dari proses multimedia dalam bentuk alat dengan tujuan bersenang-senang dan dapat digunakan sebagai media belajar.

Banyak *programmer-programmer* hebat yang berawal dari kegemaran terhadap bermain *game*. Meskipun kata *game* terdengar ditujukan untuk mainan anak, sebenarnya terdapat banyak manfaat yang bisa didapat dari bermain *game*. Salah satu manfaat tersebut meningkatkan kemampuan menalar atau logika. Dalam *game* terdapat permasalahan yang harus dicari jalan keluarnya untuk mencapai tahapan/tujuan tertentu, sehingga diperlukan kemampuan berpikir dan bernalar sampai dengan menyusun strategi untuk dapat menyelesaikan setiap permasalahan dalam *game*. Manfaat *game* sebagai hiburan yang cukup menantang dapat

membantu menyegarkan pikiran setelah melakukan aktivitas seharian, dan dapat digunakan untuk melatih refleks dan ketepatan pengguna terhadap sesuatu.

Menurut Wafda (Rifai, 2015), *game* dapat diklasifikasikan menjadi dua jenis, yaitu *game* 2D dan 3D. *Game* 2D memiliki keadaan ruang yang memiliki dua sisi, yaitu sisi X dan Y sehingga objek yang ditampilkan hanya dapat dilihat dari satu arah dan tidak dapat menampilkan bagian belakang dan samping sebuah objek. Sedangkan *game* 3D melibatkan 3 sisi, yaitu sisi X, sisi Y dan sisi Z sehingga objek dapat dilihat dari berbagai sudut menggunakan kamera. Saat ini *game* dengan desain 3 dimensi sangat diminati oleh masyarakat dibandingkan dengan *game* dengan desain 2 dimensi.

Sementara dilihat dari bagaimana cara pengguna bermain, *game* memiliki beberapa kategori diantaranya: *arcade*, *first person shooter*, *role Playing game*, *simulation*, *racing*, dan sebagainya. Salah satu kategori yang sering dimainkan adalah *game arcade*. *Game arcade* adalah jenis permainan yang bergantung pada ketangkasan tangan pemain dalam mengendalikan kontrol yang umumnya mempunyai karakteristik seperti memiliki konsep dan desain sederhana dan meningkatkan tingkat kesulitan pada setiap level.

### **2.1.2. *Game* Edukasi**

Menurut Efwan (Efwan, Bunyamin, & Wahyudin, 2014), *game* edukasi adalah sebuah *game* berbasis digital yang dirancang sebagai bentuk pengayaan pendidikan dalam mendukung proses pembelajaran dan pengajaran. Efwan, Bunyamin, & Wahyudin melakukan penelitian yang menghasilkan sebuah media *game* edukasi 3 dimensi untuk mempelajari Bahasa Inggris. Sedangkan menurut

Rahman (Rahman & Tresnawati, 2016), mengemukakan *game* edukasi merupakan permainan yang dikemas sedemikian rupa sehingga dapat merangsang daya pikir dan melatih meningkatkan konsentrasi penggunaannya. Berdasarkan pendapat-pendapat diatas dapat disimpulkan *game* edukasi adalah suatu bentuk *game* yang dapat dimanfaatkan untuk mendukung proses belajar mengajar dengan cara yang lebih menyenangkan dan lebih kreatif serta untuk memberikan pengetahuan kepada penggunaannya melalui media yang menarik.

*Game* dengan tujuan mendidik dimanfaatkan sebagai salah satu media edukasi yang memiliki karakteristik pola pembelajaran *learning by doing*. Sama halnya dengan proses belajar mengajar di kelas, materi yang disampaikan oleh guru kepada siswa tidak semua dapat memahami secara langsung. Namun melalui praktik, siswa dapat melihat, mengalami dan mengambil kesimpulan dari materi yang sudah dipraktikkan. Siswa cenderung lebih tertarik dan lebih mudah mengingat materi belajar melalui media interaktif dibandingkan belajar menggunakan buku (Hutabri & Putri, 2019). Ini berarti bila *game* memuat teori pembelajaran dapat lebih mudah dipahami melalui praktik. Dengan pola pembelajaran *learning by doing*, pengguna diharuskan untuk belajar sehingga mereka dapat memecahkan masalah yang ada.

Berdasarkan hasil penelitian terdahulu, ditemukan bahwa *game* edukasi mendukung proses pendidikan (Rifai, 2015). *Game* edukasi unggul dalam banyak hal daripada metode pembelajaran tradisional. Beberapa keunggulannya yakni dengan *game* edukasi bisa membantu melatih pengguna untuk berlogika, menganalisis, dan memecahkan masalah yang dihadapi. Membantu dalam

merangsang stimulus otak serta mengembangkan imajinasi. Selain itu, adanya animasi multimedia yang dapat memacu daya ingat untuk menyimpan materi pelajaran dalam waktu yang lebih lama daripada metode pembelajaran tradisional (Nugroho, 2015).

### **2.1.3. Algoritma**

Definisi algoritma adalah suatu urutan tahapan / langkah-langkah logis dalam menyelesaikan suatu masalah yang disusun secara sistematis dan berdasarkan logika. Algoritma sangat erat kaitannya dengan kata logika, yaitu kemampuan seseorang dalam berpikir menggunakan alasan tentang suatu masalah yang menghasilkan suatu kebenaran, yang terbukti dan dapat diterima dengan akal. Logika sering dikaitkan dengan kecerdasan, seseorang yang mampu berlogika dengan baik sering disebut sebagai orang yang pintar. Logika sangat identik dengan konsep akal dan penalaran. Penalaran adalah bentuk pengetahuan tidak langsung yang didasarkan pada pernyataan berpikir yang mungkin benar dan mungkin juga salah (Manurung, 2016).

Definisi logika sangat sederhana yaitu ilmu yang mengajarkan cara berpikir sesuai dengan aturan yang tertentu yang telah diberlakukan. Pelajaran logis menciptakan kesadaran dalam menggunakan prinsip untuk berpikir secara sistematis. Kata algoritma diambil dari nama seorang ilmuwan Arab yang bernama Abu Jafar Muhammad Ibnu Musa Al Khuwarizmi penulis buku berjudul “Al Jabar Wal Muqabala” (Munir, 2011). Kata Al Khuwarizmi dibaca dengan Bahasa Inggris menjadi *Algorism* yang kemudian secara perlahan berubah menjadi *Algorithm* yang diserap dalam ke Bahasa Indonesia menjadi kata Algoritma.

Dalam menggunakan algoritma dibutuhkan beberapa pertimbangan. Pertama, algoritma harus benar. Ini berarti algoritma yang dikehendaki memberikan output yang sesuai dengan input yang telah diberikan. Bila *output* salah, algoritma yang digunakan bukan termasuk algoritma yang baik. Kedua, hasil dari algoritma mendekati nilai yang sebenarnya. Ketiga, keefisienan algoritma. Algoritma yang baik tidak hanya mengeluarkan output yang benar, tetapi juga mencakup waktu dan memori yang digunakan. Bila algoritma tersebut membutuhkan waktu yang lama dan mengonsumsi memori yang banyak, maka algoritma tersebut bukan algoritma yang baik.

Algoritma secara garis besar dibedakan menjadi 2 bentuk penyajian, yaitu tulisan dan gambar. Salah satu bentuk penyajian dengan tulisan berupa *pseudocode* dimana kode yang mirip dengan kode pemrograman yang sebenarnya sehingga lebih mudah dikomunikasikan kepada pemrogram. Sedangkan penyajian dalam bentuk gambar dapat berupa *flowchart* yang memperlihatkan urutan dan hubungan antar proses beserta pernyataannya.

#### **2.1.4. Android**

*Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis *linux* yang mencakup sistem operasi, *middleware*, dan aplikasi (Efendi, 2018). Kemajuan teknologi saat ini tentunya tidak lepas dari perkembangan teknologi yang semakin canggih. *Android* sendiri akan terus berkembang dan memperbaharui sistem operasinya dengan mengikuti perkembangan teknologi saat ini. Hal ini dapat dilihat dari adanya versi demi versi yang telah di *upgrade* setiap tiga sampai enam bulan kedepan yang terus dikembangkan oleh pihak *android*. Demi menghadapi pesaing

(kompetitor) lainnya supaya konsumen tetap berada atau merasa nyaman dengan *smartphone* yang dimilikinya, *android* juga terus memberi pelayanan yang memuaskan bagi pelanggannya. Berbagai fitur yang telah ditawarkan oleh android bisa menjadikannya platform *smartphone* handal yang menguasai pangsa pasar dunia (global) hingga saat ini.

### 2.1.5. *Unified Modeling Language (UML)*

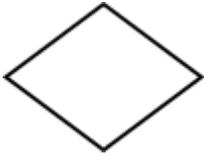
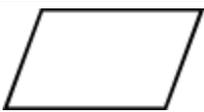
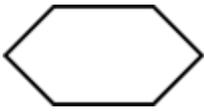
*Unified Modeling Language (UML)* merupakan suatu desain pemodelan secara visual mengenai arus sistem informasi. Umumnya UML digunakan oleh pengembang sistem sehingga perancangan dapat terarah dan terdokumentasi dengan baik. Berdasarkan buku “Rekayasa Perangkat Lunak”, dalam memodelkan suatu sistem, UML terdiri dari 13 macam diagram yang didasarkan pada kebutuhan dan karakteristik sistem (Rosa & Shalahuddin, 2015). Masing-masing model menjelaskan cara kerja dari proses suatu sistem. Berikut beberapa model yang digunakan dalam penelitian ini dan penjelasannya.

#### a. *Flowchart*

*Flowchart* adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program.

Tabel 2. 1 Simbol Notasi dan Relasi dalam Diagram *Use case*

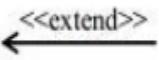
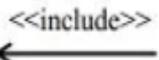
Simbol	Nama	Kegunaan
	Garis Alir	Menunjukkan arah aliran algoritma, dari satu proses ke proses berikutnya.
	Terminal	Menunjukkan awal atau akhir sebuah proses.

	Proses / Langkah	Menyatakan kegiatan yang akan terjadi 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	Digunakan untuk mewakili data masuk, atau data keluar. Hanya bisa dimulai dari masukan menuju keluaran, bukan sebaliknya.
	<i>Predefined Process</i>	Digunakan untuk menunjukkan suatu proses yang begitu kompleks, sehingga tidak bisa dijelaskan di diagram alir ini dan merujuk pada diagram alir yang terpisah.
	Persiapan / Inisialisasi	Menunjukkan operasi yang tidak memiliki efek khusus selain mempersiapkan sebuah nilai untuk langkah / proses berikutnya.
	Konektor dalam Halaman	Digunakan untuk menghubungkan satu proses ke proses lainnya, sama halnya seperti tanda panah dalam pengulangan. Namun hanya bisa menghasilkan satu keluaran.
	Konektor Luar Halaman	Berfungsi untuk menghubungkan satu proses ke proses lainnya, sama halnya seperti tanda panah, hanya saja untuk merujuk ke halaman yang berbeda.
	Kontrol / Inspeksi	Menunjukkan proses / langkah di mana ada inspeksi atau pengontrolan.

b. *Use case diagram*

*Use case* merepresentasikan hubungan antara fungsionalitas yang membentuk sistem secara teratur dengan aktor. *Use case diagram* menggunakan sudut pandang pengguna sistem untuk menggambarkan pola perilaku sistem dan urutan kegiatan yang berlangsung. Gambaran tersebut berupa gambaran singkat hubungan antara *use case*, sistem, dan *actor*. Masing-masing *use case* didefinisikan lebih rinci melalui sebuah skenario. Notasi dan relasi dari *use case* sebagai berikut:

Tabel 2. 2 Simbol Notasi dan Relasi dalam Diagram *Use Case*

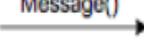
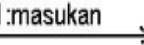
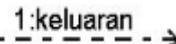
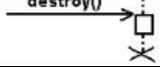
Simbol	Nama	Kegunaan
	<i>Actor</i>	Mewakili manusia, sistem, atau perangkat yang akan menggunakan berinteraksi dengan sistem.
	<i>Use case</i>	Menggambarkan hubungan <i>actor</i> dengan sistem
	<i>Association</i>	Komunikasi antara masing-masing aktor dan setiap <i>use case</i> dalam <i>use case</i> atau <i>use case</i> berinteraksi dengan aktor.
	<i>Extend</i>	Menunjukkan bahwa bagian dari suatu elemen adalah penambahan fungsional dari elemen lain jika kondisi tertentu terpenuhi
	<i>Include</i>	Menunjukkan bahwa suatu bagian dari elemen seluruhnya merupakan bagian dari elemen lainnya
	<i>Generalization</i>	Menunjukkan hubungan antara elemen yang lebih umum dengan elemen yang lebih spesifik.
	<i>Dependency</i>	Hubungan yang menunjukkan bahwa perubahan dalam satu elemen mempengaruhi elemen lainnya. Terdapat dua tipe yaitu <i>include</i> dan <i>extend</i> .

c. *Sequence diagram*

Diagram ini menjelaskan bagaimana mendefinisikan input dan output serta urutan interaksi antara sistem dan pengguna untuk sebuah *use case*. Secara garis besar diagram sekuen memberikan informasi mengenai urutan informasi diantara objek-objek yang berkomunikasi. Jumlah diagram sekuen yang akan dibuat sebanyak *use case* yang telah didefinisikan. Berikut simbol – simbol dalam diagram sekuen:

Tabel 2. 3 Simbol Notasi dan Relasi dalam Diagram Sekuen

Simbol	Deskripsi
	Aktor: mewakili orang, proses maupun sistem yang berinteraksi dengan sistem yang akan dibuat. Dapat berupa objek orang maupun objek nama.

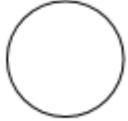
	Objek: berpartisipasi dalam dengan mengirimkan atau menerima pesan
	Garis hidup: menunjuk objek-objek yang saling berinteraksi
	Waktu aktif: mendeklarasikan suatu objek dalam keadaan aktif dan interaktif.
	Pesan: memuat informasi-informasi mengenai kegiatan yang terjadi
	Pesan tipe <i>create</i> : menyatakan bahwa suatu objek dibentuk oleh objek yang lain, arah panah mengarah pada objek yang dibuat
	Pesan tipe <i>send</i> : menyatakan bahwa suatu objek mengirimkan data masuka ke objek lain
	Pesan tipe <i>return</i> : menyatakan bahwa suatu objek telah menjalankan atau menghasilkan suatu kembalian ke objek tertentu.
	Pesan tipe <i>destroy</i> : menyatakan suatu objek mengakhiri keadaan objek yang lain

d. *Class diagram*

*Class diagram* menunjukkan kelas-kelas yang ada dari sebuah sistem dan hubungannya secara logika. Diagram ini digunakan untuk mendokumentasikan dan menggambarkan kelas-kelas yang akan dibuat untuk membangun sistem dan hubungan antara *class object* tersebut. Kelas-kelas tersebut dapat ditentukan/ditemukan dengan cara memeriksa objek-objek dalam *sequence diagram*. Berikut notasi-notasi yang ada dalam *class diagram*:

Tabel 2. 4 Simbol Notasi dan Relasi dalam Diagram Kelas

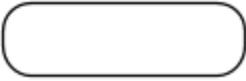
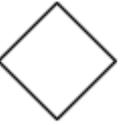
Simbol	Deskripsi
	Kelas: suatu kelas pada struktur sistem

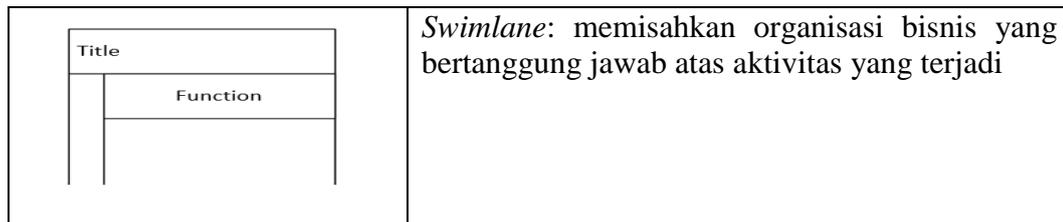
	Antarmuka: sebagai antarmuka komponen
	Asosiasi: Hubungan antara kelas dengan makna umum
	Asosiasi berarah: hubungan antara kelas dengan makna suatu kelas yang digunakan oleh kelas lain
	Generalisasi: relasi antarkelas dengan makna umum khusus
	Agregasi: Hubungan antar kelas dengan makna semua bagian

e. *Activity diagram*

Diagram aktivitas menjelaskan alur kerja sistem bisnis atau proses atau menu yang ditemukan dalam perangkat lunak. Diagram aktivitas menggambarkan aktivitas yang terjadi dalam sistem. Mulai dari awal tahapan sampai akhir sistem yang telah dirancang sesuai langkah-langkah proses kerja sistem. Diagram ini dibuat berdasarkan satu atau lebih *use case* yang ada dalam *use case diagram*.

Tabel 2. 5 Simbol Notasi dan Relasi dalam Diagram Aktivitas.

Simbol	Deskripsi
	Status awal: sebuah diagram aktivitas yang memiliki sebuah kondisi status awal
	Aktivitas: aktivitas yang dilakukan oleh sistem umumnya dimulai dengan kata kerja.
	Percabangan: asosiasi percabangan dimana jika ada lebih dari satu pilihan
	Penggabungan: asosiasi dimana lebih dari satu aktivitas digabungkan menjadi satu
	Status akhir: diagram aktivitas yang memiliki status akhir



## 2.2. Variabel

Algoritma berisi rangkaian langkah-langkah dalam menyelesaikan suatu masalah. Langkah-langkah tersebut digunakan untuk menyelesaikan masalah yang dapat berupa struktur sekuensial (*sequence*), struktur seleksi (*selection*) dan struktur perulangan (*repetition*). Ketiga jenis struktur tersebut akan membentuk konstruksi didalam suatu algoritma (Munir, 2011).

### 1. Struktur sekuensial (*sequence*)

Pada dasarnya sebuah program menjalankan proses dari dasar. Dalam struktur sekuensial, langkah-langkah dalam algoritma dilakukan atau dieksekusi secara berurutan dari awal hingga akhir secara berurutan. Dari yang langkah pertama dieksekusi ke langkah yang kedua, selanjutnya dari langkah kedua ke langkah yang ketiga, dan seterusnya. Urutan dalam eksekusi ini menentukan keadaan akhir algoritma. Bila urutan langkah eksekusinya dirubah maka ada kemungkinan memberikan hasil akhir yang berbeda.

Sebagai contoh dalam sebuah persoalan pertukaran isi dua gelas air. Gelas A yang berisi air berwarna hijau dan gelas B yang berisi air berwarna kuning dengan kondisi volume air sama-sama penuh. Bagaimana menukar isi air gelas A dan gelas B sehingga gelas B berisi air berwarna hijau dan gelas A berisi warna kuning. Bila algoritma solusi dari persoalan diatas dituliskan seperti berikut:

- a. Menuang air berwarna hijau dari gelas A ke gelas B yang berisi air berwarna kuning.
- b. Menuang air berwarna kuning dari gelas B ke gelas A.

Maka algoritma tersebut memberikan hasil berupa gelas B tidak berisi air karena kosong dan gelas A berisi air berwarna kuning. Sehingga algoritma diatas tidaklah benar. Untuk menyelesaikan permasalahan tersebut, dibutuhkan sebuah gelas kosong yaitu gelas C yang digunakan sebagai tempat penampungan sementara. Dengan begitu terdapat 3 langkah dalam menyelesaikan persoalan tersebut. Langkah-langkah tersebut berupa:

- a. Menuang air berwarna hijau dari gelas A ke dalam sebuah gelas C kosong.
- b. Menuang air berwarna kuning dari gelas B ke gelas A yang sudah kosong.
- c. Menuang kembali air berwarna hijau dari gelas C ke gelas B yang kosong.

Dari persoalan diatas, algoritma yang digunakan terdiri atas 3 buah pernyataan. Tiap pernyataan akan dieksekusi secara berurutan. Hasil dari algoritma yang telah dieksekusi adalah gelas A berisi air berwarna kuning dari gelas B dan gelas B berisi air berwarna hijau dari gelas A.

## 2. Struktur seleksi (*selection*)

Dalam struktur seleksi, seleksi langkah-langkah ditetapkan berdasarkan suatu kondisi atau untuk membuat suatu keputusan. Algoritma mengeksekusi instruksi berikutnya jika kondisi yang ditetapkan terpenuhi. Dalam struktur ini, tidak semua instruksi dieksekusi, instruksi akan dijalankan jika memenuhi persyaratan sehingga hanya satu instruksi dari dua atau lebih intruksi yang disediakan akan dieksekusi.

Sebagai contoh, persoalan dalam berkendara diperempatan yang terdapat lampu lalu lintas. Untuk lampu lalu lintas terdapat aturan-aturan yang harus ditaati oleh pengendara. Aturan-aturan tersebut dapat berupa:

- a. Jika lampu dari lampu lalu lintas berwarna merah, maka para pengendara harus berhenti.
- b. Jika lampu dari lampu lalu lintas berwarna kuning sesudah lampu hijau padam, maka para pengendara bersiap untuk berhenti.
- c. Jika lampu dari lampu lalu lintas berwarna kuning sesudah lampu merah padam, maka para pengendara bersiap untuk jalan.
- d. Jika lampu dari lampu lalu lintas berwarna, maka para pengendara diperbolehkan untuk jalan.

Pernyataan-pernyataan diatas dapat dituliskan dalam pernyataan kondisional.

Adapun pernyataan kondisional dapat ditulis sebagai berikut:

*if* kondisi *then* aksi

Jika diterjemahkan dalam Bahasa Indonesia, “*if*” berarti “jika”, “*then*” berarti “maka”, “kondisi” berarti syarat yang dapat ditentukan bernilai benar atau salah, “aksi” berarti bila kondisi benar akan dieksekusi. Sedangkan bila kondisi bernilai salah, maka aksi tidak dieksekusi. Dalam struktur seleksi, kata “*if*” dan “*then*” adalah kata kunci. Struktur seleksi *if-then* hanya memberikan hanya satu pilihan aksi bila kondisi terpenuhi (bernilai benar), dan tidak memberikan pilihan lain bila kondisi tidak terpenuhi (bernilai salah).

Bentuk lain dari algoritma dalam struktur seleksi yang memberikan lebih dari 1 aksi berupa *if-then-else* yang dapat dituliskan seperti berikut:

*If* kondisi *then* aksi 1

*else* aksi 2

Disini “*else*” berarti “selain itu”. Jika syarat dalam kondisi terpenuhi (bernilai benar) maka aksi 1 dieksekusi, selain itu jika kondisi tidak terpenuhi (bernilai salah) maka intruksi mengeksekusi aksi 2. Contoh algoritma dari struktur diatas dapat berupa:

- a. *If* hari sudah malam *then* hidupkan lampu *else* matikan lampu.
- b. *If* air di bak mandi sudah penuh *then* matikan air *else* buka keran air.

Untuk struktur seleksi yang lebih rumit yang memberikan lebih dari 2 aksi dapat berupa seleksi bersarang yaitu *if-then-elseif-then-else* yang dapat dituliskan seperti berikut:

*If* kondisi 1 *then* aksi 1

*else if* kondisi 2 *then* aksi 2

*else* aksi 3

Dari penulisan tersebut dapat diartikan jika kondisi 1 terpenuhi (bernilai benar) maka eksekusi aksi 1 dan jika kondisi 1 bernilai salah periksa kondisi 2. Jika kondisi 2 bernilai benar maka eksekusi aksi 2, selain itu jika kondisi 1 dan 2 bernilai salah, maka eksekusi aksi 3. Contoh algoritma dari struktur diatas dapat berupa:

- a. *If* waktu menunjukkan pukul 12 siang *then* waktunya makan siang *else if* waktu menunjukkan pukul 6 malam *then* waktunya makan malam *else* waktunya istirahat.
- b. *If* cuaca hari ini mendung *then* sedia payung *else if* cuaca hari ini hujan *then* buka payung *else* cuaca cerah.

Dari beberapa struktur seleksi yang telah dijabarkan dapat memberikan keuntungan yaitu kemampuan dalam mengikuti jalur aksi yang kemungkinan berbeda berdasarkan syarat kondisi yang ditentukan.

### 3. Struktur perulangan (*repetition*)

Dalam struktur *repetition*, langkah-langkah tertentu dieksekusi berulang kali hingga syarat suatu kondisi terpenuhi. Terdapat beberapa macam konstruksi perulangan yang berbeda. Dari beberapa konstruksi tersebut, ada yang dapat dipakai untuk masalah yang sama dan ada yang hanya cocok digunakan untuk masalah tertentu. Pemilihan dalam menggunakan konstruksi perulangan untuk masalah tertentu dapat mempengaruhi kebenaran algoritma.

Komputer dapat mengerjakan pekerjaan yang sama tanpa kenal lelah, yang berbeda dengan manusia yang cepat lelah mengerjakan pekerjaan yang berulang-ulang. Sebagai contoh dalam sebuah persoalan untuk menuliskan kalimat yang sama sebanyak 100 kalimat. Bagaimana komputer dapat menuliskan kalimat sebanyak 100 kali. Bagi manusia, solusi yang dapat digunakan dengan menuliskan 1 kalimat per 1 kalimat hingga mencapai 100 kalimat. Berbeda dengan komputer, bila komputer juga menuliskan 1 per 1 kalimat maka menjadi tidak efisien. Oleh karena itu, komputer dapat menggunakan notasi dalam struktur perulangan yang dapat berupa *repeat-until*, *for* dan *while*. Berikut penjelasannya:

#### a. Penulisan *repeat-until*: *repeat* aksi *until* kondisi.

Perulangan ini terjadi berdasarkan nilai benar atau salah. Aksi akan dieksekusi sebanyak kondisi bernilai salah. Jika kondisi sudah bernilai benar, maka aksi akan berhenti dieksekusi.

- b. Penulisan *for*: *for* peubah  $\leftarrow$  nilai awal *to* nilai akhir *do* aksi *endfor*.

Pernyataan *for* digunakan untuk menghasilkan perulangan sebanyak yang telah ditentukan. Jumlah perulangan dapat ditentukan sebelum eksekusi dilakukan. Untuk melakukan perulangan, dibutuhkan sebuah peubah yang nilainya akan bertambah setiap kali perulangan terjadi. Jika peubah tersebut sudah mencapai jumlah yang telah ditentukan, maka perulangan akan berhenti.

- c. Penulisan *while*: *while* kondisi *do* aksi *endwhile*.

Aksi akan di eksekusi berulang kali selama kondisi bernilai benar. Bila kondisi masih bernilai salah maka perulangan akan berhenti. Dalam perulangan ini harus diperhatikan bahwa kondisi akan menghasilkan perulangan yang berhenti. Perulangan yang tidak pernah berhenti mengartikan bahwa logika dari algoritma tersebut salah.

Dari persoalan sebelumnya bila digunakan ketiga konstruksi diatas maka dapat dituliskan sebagai berikut:

- a. Untuk *repeat-until*

*repeat* kalimat until total kalimat sama dengan 100.

- b. Untuk *for*

*for* peubah  $\leftarrow$  1 *to* 100 *do* kalimat *endfor*.

- c. Untuk *while*

*while*  $x < 100$

*do* kalimat

$x \leftarrow x + 1$  *endwhile*

## 2.3. Software Pendukung

### 2.3.1. Unity

Dalam buku yang berjudul “*Programming a Game With Unity*” yang ditulis oleh Andre Infante (Infante, 2014) mengungkapkan kehadiran *videogame* memunculkan banyak *tools* yang dikembangkan sebagai editor untuk mengembangkan *game* modern kepada individu maupun tim *programmer* dan perancang. Salah satu *tools* tersebut adalah *unity* yang dapat digunakan secara mudah dan cepat, dengan biaya yang murah serta dilengkapi fitur-fitur yang membuatnya ideal dalam pengembangan sebuah *game*. *Unity* memiliki antarmuka yang sederhana dan mudah dipelajari serta memiliki tingkat grafis yang tinggi sehingga dapat menghasilkan *game* dengan cepat dan berkualitas.

Kelebihan dari *unity* yaitu dapat digunakan untuk membangun *game* 2D dan 3D. *Unity* mampu berjalan untuk hampir semua jenis sistem operasi. *Scripting* dalam *Unity* 3D juga mudah dipelajari dan cukup sederhana. *Unity* memiliki *framework* lengkap untuk pengembangan profesional. Untuk menjalankan *engine* ini, disediakan beberapa pilihan dalam menggunakan bahasa pemrograman seperti C#, boo dan javascript. Berdasarkan uraian diatas, maka dapat disimpulkan *Unity* merupakan suatu *software editor engine* yang mudah dioperasikan dalam hal mengembangkan *game* yang dapat berjalan diberbagai macam platform. Untuk itu, dalam penelitian ini peneliti menggunakan *software unity* dalam mengembangkan *game* edukasi untuk pengenalan dasar algoritma berbasis *android*.

### **2.3.2. Blender**

Dalam buku berjudul “*The Beginners Guide To Blender*” oleh Jonathan Lampel (Lampel, 2015), menguraikan sebuah *software* 3D bernama *blender*. Jonathan menjelaskan *software blender* merupakan *software open source* yang sering digunakan dalam pembuatan model cetak 3D, film animasi, efek visual dan lain sebagainya. Selain itu *Blender* juga memiliki berbagai macam fitur-fitur yang memudahkan penggunaannya termasuk *modeling, rendering, texturing, rigging, scripting, composite*, post-produksi dan banyak lagi. Dalam penelitian ini, peneliti menggunakan *Blender* sebagai media untuk membuat beberapa model yang akan digunakan dalam mengembangkan *game* edukasi untuk pengenalan dasar algoritma.

### **2.3.3. Adobe Illustrator**

Dalam *ebook* berjudul “*Otodidak Adobe Ilustator*” oleh Jubilee Enterprise (Enterprise, 2018) menjelaskan sebuah *software* desain grafis yang bernama *Adobe Illustrator* yang dimanfaatkan untuk membuat gambar dan ilustrasi dalam bentuk vektor. Untuk objek vektor memiliki keunggulan dalam hal detail ketajaman gambar. *Adobe illustrator* sering digunakan untuk membuat desain grafis, melukis sebuah objek, menata tulisan, membuat desain website dan lain sebagainya. Dalam penelitian ini, peneliti menggunakan *Adobe illustrator* sebagai alat bantu untuk membuat beberapa gambar dan *icon* yang akan digunakan dalam mengembangkan *game* edukasi.

### **2.3.4. Microsoft Visio**

Agar pengembangan *game* edukasi lebih terarah dan dapat memberikan gambaran maka digunakan rancangan diagram UML menggunakan *software* pendukung berupa *Microsoft Visio*. Dalam buku yang berjudul “*Microsoft Visio*” yang diterbitkan oleh Tutorials Point (Tutorials Point (I) Pvt. Ltd., 2017) menjelaskan *Microsoft Visio* adalah sebuah *tools* diagram yang memungkinkan untuk membuat diagram mulai dari tingkatan simple sampai dengan tingkatan kompleks, yang membantu dalam visualisasi data dan pemodelan proses.

*Microsoft Visio* juga membantu untuk membuat bagan organisasi terperinci, *floor plan*, diagram pivot, dan sebagainya. Berdasarkan uraian tersebut, maka dapat disimpulkan *Microsoft Visio* merupakan sebuah perangkat lunak komputer keluaran perusahaan *Microsoft* yang dapat membantu dalam pembuatan diagram dengan banyak fasilitas sehingga informasi dan sistem dapat digambarkan dalam bentuk suatu diagram disertai penjelasan singkat.

## **2.4. Penelitian Terdahulu**

Penelitian terdahulu yang berkaitan tentang pengembangan *game* edukasi berbasis *Android* yang berhubungan dengan pengenalan dasar algoritma sebagai berikut:

1. Penelitian yang dilakukan oleh Lindberg dan Laine dengan judul “*Formative evaluation of an adaptive game for engaging learners of programming concepts in K-12*” - *International Journal of Serious Games*, Volume: 5, Nomor: 2, Juni 2018 ISSN: 2384-8766. Penelitian tersebut mengangkat masalah mengenai permintaan global untuk *programmer* meningkat sehingga

beberapa negara telah mengintegrasikan pemrograman ke dalam kurikulum K-12. Metode yang digunakan untuk permasalahan tersebut berupa penyajian materi pembelajaran melalui permainan, yaitu *game* edukasi pemrograman. Hasil dari 32 siswa kelas 6 Korea yang memainkan *game* dan 32 siswa kelas 6 yang mempelajari konsep yang sama menggunakan *handout* menunjukkan pembelajaran sama efektifnya pada kedua kelompok. (Lindberg & Laine, 2018).

2. Penelitian yang dilakukan oleh Yogi Udjaja dengan judul “***Gamification Assisted Language Learning For Japanese Language Using Expert Point Cloud Recognizer***” - International Journal of Computer Games Technology, Volume: 2018, Nomor: 1, Desember 2018 ISSN: 16877055. Penelitian tersebut dilatarbelakangi oleh tingginya calon siswa Internasional yang melanjutkan pendidikan di Jepang yang terkendala oleh keterampilan berbahasa Jepang. Metode yang digunakan dalam penelitian tersebut berupa *Gamification Assisted Language Learning* (GALL) yang menyajikan materi pembelajaran melalui *game* untuk merangsang sistem saraf sensorik dan motorik dan memotivasi siswa (pemain) untuk belajar lebih keras. Hasil secara keseluruhan dalam hitungan seminggu terbukti dapat meningkatkan kemampuan pemain dari 20% hingga 100%. (Udjaja, 2018).
3. Penelitian yang dilakukan oleh Tumpal Halomoan Manurung dengan judul “**Perancangan Aplikasi Pembelajaran Logika dan Algoritma dengan menggunakan Metode *Computer Based Instruction***” - Jurnal Riset Komputer (JURIKOM) Volume: 3, Nomor: 1, Februari 2016 ISSN: 2407-

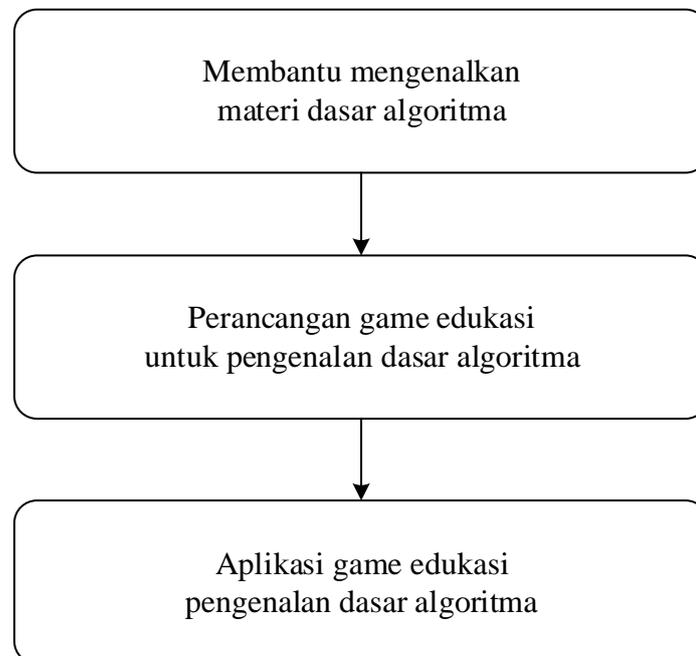
389X. Permasalahan yang diangkat dalam penelitian ini terkait dengan masih banyak orang umum yang dapat menciptakan suatu program namun masih ada yang belum paham dasar logika dan algoritma. Metode yang digunakan oleh peneliti yaitu *Computer Based Instruction* yang menggunakan komputer sebagai media penyampaian materi. Hasil penelitian memberikan kesimpulan bahwa aplikasi memudahkan pengguna untuk memahami pembelajaran logika dan algoritma menggunakan sarana yang menarik dan mudah dipahami dalam bentuk teks, gambar dan video, serta membantu mengarahkan dan memaksimalkan proses belajar mengajar. (Manurung, 2016).

4. Penelitian yang dilakukan oleh Yoyon Efendi dengan judul “**Rancangan Aplikasi Game Edukasi berbasis Mobile menggunakan App Inventor**” - Jurnal Intra-Tech Volume 2 Nomor: 1, April 2018 ISSN: 2549-0222. Penelitian tersebut mengangkat masalah yang ditemui pendidik dalam memberi pelajaran yang berbasis buku cetak berupa kurangnya kemauan belajar anak. Tujuan dari *game* edukasi bagi anak prasekolah untuk memberikan stimulasi terhadap perkembangan anak baik fisik, motorik, kemampuan berpikir, serta psikososial dan keberanian. Metode yang digunakan berupa model *research and development* yang sudah terstandarisasi. Hasil dari penelitian ini menunjukkan bahwa aplikasi *game* yang berisi suara, gambar dan kuis dapat memperkenalkan materi dengan cara yang menarik sehingga mudah diterima dan dipahami oleh anak yang masih dalam usia dini (Efendi, 2018).

5. Penelitian yang dilakukan oleh Anik Vega Vitianingsih dengan judul “**Game Edukasi sebagai Media Pembelajaran Pendidikan Anak Usia Dini**” - Jurnal INFORM Volume 1, Nomor: 1, 2016 ISSN: 2502-3470. Penelitian ini mengangkat masalah dengan mengganti metode pembelajaran konvensional menjadi *game* yang memuat simulasi sehingga dapat mengembangkan kreativitas anak. Tujuan dari penelitian ini untuk membuat suatu aplikasi *game* edukasi sebagai media alternatif dalam menyampaikan pembelajaran. Metode yang digunakan dalam penelitian ini berupa metode *waterfall life cycle*. Hasil dari penelitian ini menunjukkan *game* edukasi dapat membantu dalam mengubah metode pembelajaran dari konvensional menjadi pembelajaran melalui media *game* (Vitianingsih, 2016).
6. Penelitian yang dilakukan oleh Ariadie Chandra Nugraha, Moh.Khairudin, dan Deny Budi Hertanto dengan judul “**Rancang Bangun Game Edukasi sebagai Media Pembelajaran Mata Kuliah Praktik Teknik Digital**” - Jurnal Edukasi Elektro, Volume 1, Nomor: 1, Mei 2017 ISSN: 2548-8260. Penelitian ini dilatarbelakangi oleh metode atau cara pembelajaran konvensional oleh dosen yang mengakibatkan tidak sedikit mahasiswa yang tidak menerima materi secara keseluruhan. Metode yang diterapkan dalam penelitian ini menggunakan metode *ADDIE* yang terdiri dari tahapan analisis, desain, pengembangan, implementasi, dan evaluasi. Hasil dari penelitian ini menunjukkan bahwa *game* edukasi yang telah dibangun layak untuk digunakan sebagai alat bantu pembelajaran (Nugraha, Khairudin, & Hertanto, 2017).

7. Penelitian yang dilakukan oleh De Firman Aonillah Efwan, Bunyamin dan Wahyudin dengan judul “**Pengembangan *Game* Edukasi 3 Dimensi Bahasa Inggris sebagai Media Pembelajaran untuk Anak**”- Jurnal TEKNOIF Volume. 3 Nomor: 1 April 2015 ISSN: 2338-2724. Penelitian tersebut dilatarbelakangi oleh pentingnya Bahasa Inggris yang menjadi Bahasa Internasional yang semestinya sudah diperkenalkan kepada masyarakat terutama kalangan anak. Metode yang digunakan dalam penelitian ini adalah *multimedia development life cycle* (MDLC). Hasil dari penelitian ini adalah sebuah *game* edukasi yang dapat menambah pembendaharaan kata dalam Bahasa Inggris, terampil dalam *writing*, terlatih dalam *listening*, serta pengucapan (Efwan et al., 2014).

## 2.5. Kerangka Pemikiran



Gambar 2. 1 Bagan Kerangka Pemikiran

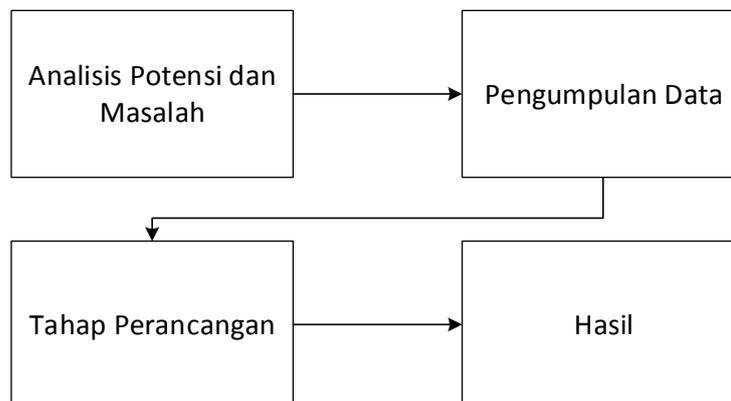
*Game* edukasi untuk pengenalan dasar algoritma ini dimulai dari adanya kesulitan dalam memahami materi algoritma yang masih terbatas oleh teori dan alat bantu ajar yang kurang memadai, kemudian adanya peluang untuk memanfaatkan *smartphone* dengan sistem operasi *android*. Untuk itu, dirancanglah sebuah media *game* edukasi untuk pengenalan dasar algoritma yang mencakup materi struktur sekuensial, struktur seleksi, dan struktur perulangan. Hasil dari penelitian ini berupa sebuah *game* edukasi sehingga dapat mempermudah siswa untuk memahami materi algoritma.

## BAB III

### METODE PENELITIAN

#### 3.1. Desain Penelitian

Dalam proses analisis dan perancangan dalam penulisan penelitian ini, peneliti berpedoman pada metode penelitian dan pengembangan (*Research and Development*) yang sudah terstandarisasi. Metode penelitian ini umumnya digunakan untuk memproduksi dan menguji produk perangkat lunak dan efektivitas produk tersebut.



Gambar 3. 1 Desain Penelitian

1. Penelitian dimulai dari adanya potensi masalah. Peneliti menganalisis potensi apa saja yang dapat menjadi nilai tambah bila diterapkan. Pengenalan dasar algoritma merupakan potensi baru yang menjadi nilai tambah ketika pengguna smartphone dapat memaksimalkan *game* edukasi untuk menambah pengetahuan. Bagi pelajar yang memulai mengenal algoritma pasti merasa kesulitan dan bagi pengajar dalam memperkenalkan algoritma kesusahan

dalam penyampaian untuk dapat diterima oleh pelajar. Oleh karena itu, dengan adanya materi pengenalan dasar algoritma yang dikemas dalam bentuk *game* edukasi, dapat mempermudah pelajar untuk memahami maupun pengajar dalam menjelaskan.

2. Setelah potensi dan masalah ditemukan, dilanjutkan dengan pengumpulan data. Peneliti menggunakan berbagai metode dalam mengumpulkan data baik dari studi pustaka mengenai pembuatan *game* yang nantinya akan mempermudah dalam merancang *game* dengan materi dasar algoritma maupun melalui wawancara untuk mendapatkan lebih banyak data dan informasi dalam penelitian ini.
3. Tahap berikut peneliti akan memasuki tahap perancangan sistem. Disini tahap perancangan menggunakan model pengembangan multimedia (*MDLC*) versi Luther yang terbagi ke dalam beberapa tahapan proses. Selanjutnya akan dijelaskan lebih detail dalam metode perancangan sistem.
4. Tahap akhir yaitu hasil berupa produk *game* edukasi yang telah diujicobakan dan disempurnakan melalui revisi-revisi sesuai saran dan masukan dari tahap ujicoba dalam metode perancangan sistem.

### **3.2. Pengumpulan Data**

Tujuan dari pengumpulan data dalam penelitian ini untuk memperoleh informasi, referensi, sebagai acuan yang dapat dipercaya dan akurat. Dalam penelitian ini, peneliti menggunakan teknik pengumpulan data yang dapat dibedakan berdasarkan sumber data yang diperoleh sebagai berikut:

### 3.2.1. Data Primer

Data primer diperoleh peneliti secara langsung dari sumber data dimana penelitian ini menggunakan teknik pengumpulan data berupa:

1. Wawancara

Teknik wawancara dilakukan untuk mengumpulkan data yang terkait dengan penelitian mengenai *game* edukasi untuk pengenalan dasar algoritma kepada beberapa narasumber. Salah satunya ialah guru yang mengampu bidang mata pelajaran pemrograman dasar di SMK Kolese Tiara Bangsa untuk memperoleh informasi terkait materi dasar algoritma yang diajarkan kepada pelajar SMK Kolese Tiara Bangsa dan beberapa pelajar SMK Kolese Tiara Bangsa.

### 3.2.2. Data Sekunder

Data sekunder dikumpulkan peneliti dari berbagai sumber yang telah ada dimana penelitian ini menggunakan teknik pengumpulan data seperti berikut:

1. Studi pustaka

Teknik pengumpulan data dari berbagai literatur pendukung terkait penelitian tentang *game* edukasi dan berbagai jurnal maupun buku yang menjelaskan bagaimana membangun dan merancang *game* edukasi sehingga *game* edukasi mampu memberikan solusi atas permasalahan dan menghasilkan produk jadi.

### 3.3. Operasional Variabel

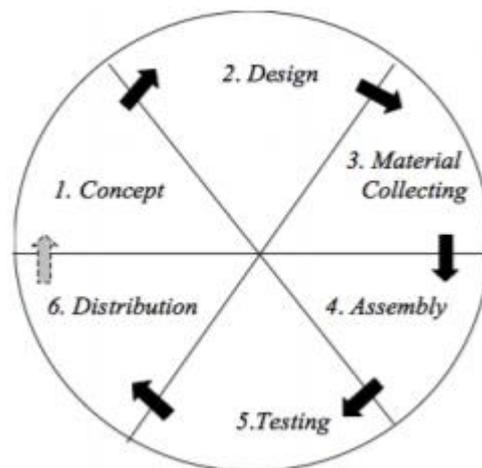
Definisi operasional variabel adalah penjelasan tentang variabel yang digunakan dalam penelitian terhadap indikator yang menyusunnya. Pada penelitian ini dapat dilihat definisi operasional penelitian dalam tabel berikut ini:

Tabel 3. 1 Operasional Variabel Dalam Penelitian

Variabel	Indikator
Algoritma	struktur sekuensial ( <i>sequence</i> )
	struktur seleksi ( <i>selection</i> )
	struktur perulangan ( <i>repetition</i> )

### 3.4. Metode Perancangan Sistem

Dalam merancang dan mengembangkan *game* edukasi pada penelitian ini, peneliti menggunakan model pengembangan multimedia (*MDLC*) versi Luther. Metode ini memiliki 6 tahap, yaitu dimulai dari *concept* (pengonsepan), *design* (perancangan), *material collecting* (pengumpulan bahan), *assembly* (pembuatan), *testing* (pengujian), dan *distribution* (pendistribusian). Menurut Luther dalam Binanto, semua tahap tersebut bisa dikerjakan tanpa secara berurutan. Akan tetapi, tahap *concept* tetap harus dikerjakan sebagai tahap awal (Binanto, 2010).



Gambar 3. 2 Model Pengembangan Multimedia versi Luther

### 3.4.1. *Concept* (Konsep)

Tahap ini merupakan tahap untuk menentukan tujuan dan siapa pengguna program (identifikasi audiens). Pada tahap ini, peneliti menentukan konsep antara lain untuk:

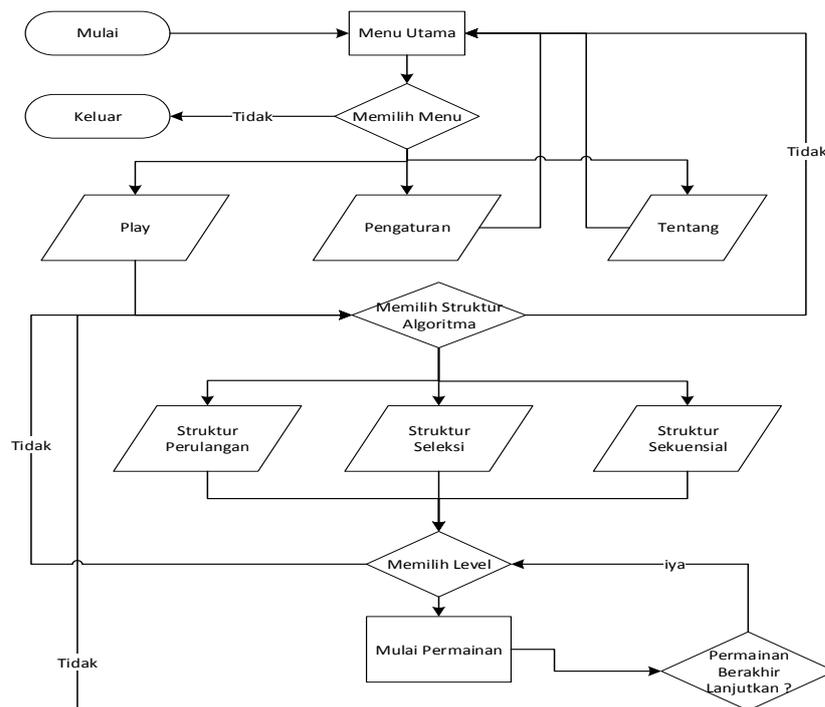
- a. Menentukan tujuan dan manfaat *game* edukasi untuk pengenalan dasar algoritma. Aplikasi ini diharapkan berguna untuk mempermudah pelajar dalam menerima dan tertarik materi dasar algoritma serta meningkatkan rasa ingin tahu. Selain itu diharapkan dapat membantu pihak sekolah untuk mengenalkan materi dasar algoritma kepada pelajar.
- b. Menentukan jenis aplikasi dan spesifikasi untuk pengenalan dasar algoritma. Aplikasi ini dapat berjalan pada sistem operasi *android* dengan spesifikasi minimal sistem operasi *android* 4.1 *Jelly Bean* (*API* Level 16), minimal OpenGL ES 2.0, serta ARMv7 CPU dengan NEON support atau Atom CPU.
- c. Menentukan siapa saja pengguna *game* edukasi untuk pengenalan dasar algoritma. Pengguna *game* edukasi dapat secara umum bagi masyarakat dan secara khusus peneliti tujukan kepada pelajar SMK Kolese Tiara Bangsa.
- d. Mendeskripsikan konsep *game* edukasi untuk pengenalan dasar algoritma yang akan dibangun. Secara konsep *game* edukasi ini menampilkan beberapa pilihan materi dasar algoritma secara umum berupa struktur sekuensial, struktur seleksi dan struktur perulangan. Tiap algoritma disediakan 10 tingkat level dalam masing-masing struktur berbeda sesuai dengan level dari yang mudah dan naik ke level tingkat berikutnya. Misi dari tiap level materi secara umum untuk mengambil semua objek berbentuk bintang. Untuk materi

struktur seleksi, jawabannya bergantung pada jawaban pengguna yang memenuhi kondisi sesuai soal yang diberikan setiap level. Selain itu juga disediakan instruksi berupa penjelasan di setiap halaman pemilihan materi level pengenalan dasar algoritma.

### 3.4.2. Design (Perancangan)

Pada tahap ini, peneliti menerjemahkan kebutuhan perangkat lunak dari tahap konsep ke dalam model desain sehingga dapat diimplementasikan ke dalam *game* pada tahap berikutnya. Tahapan desain ini terdiri dari beberapa diagram seperti:

#### a. Flowchart

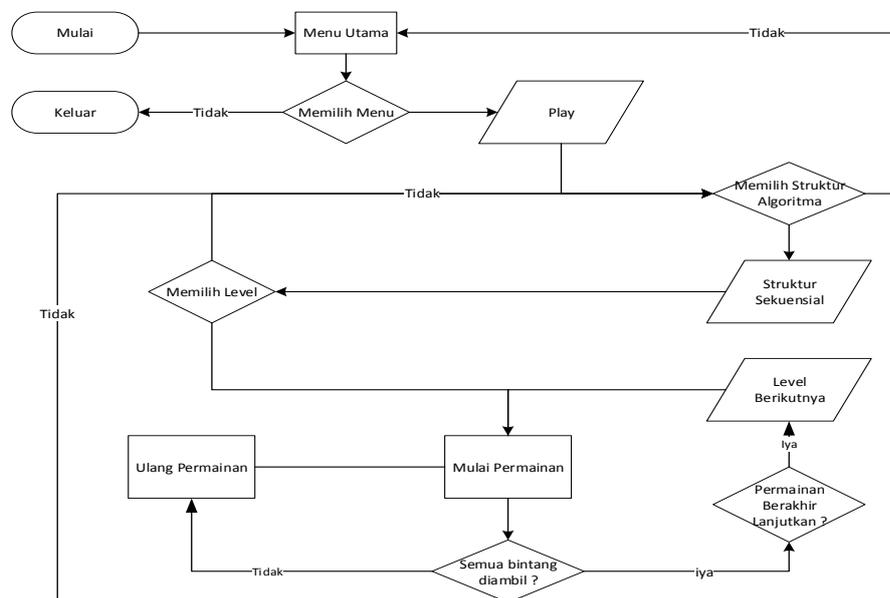


Gambar 3. 3 Flowchart secara umum

Sumber: Data Olahan Peneliti

Gambar diatas menerangkan bagan alir aplikasi secara umum. Ketika aplikasi pertama kali dibuka akan menampilkan halaman menu utama yang memiliki 4

pilihan menu berupa menu mulai, pengaturan, tentang, dan keluar dari aplikasi. Menu pengaturan digunakan untuk mengatur volume audio yang berjalan dalam aplikasi. Menu tentang menjelaskan mengenai aplikasi tersebut. Pengguna dapat keluar dari aplikasi melalui menu keluar. Selanjutnya menu mulai merupakan tampilan yang mengantarkan pengguna ke menu pemilihan materi yang terdiri materi struktur perulangan, struktur seleksi, serta struktur sekuensial. Setelah pengguna memilih materi yang ingin dipelajari, pengguna dapat memilih level yang tersedia dari level 1 sampai level 10. Level yang dipilih semakin tinggi maka tingkat kesulitannya semakin bertambah. Bila pengguna berhasil menyelesaikan level permainan, diberikan pilihan untuk lanjut ke level berikutnya atau keluar dari aplikasi.

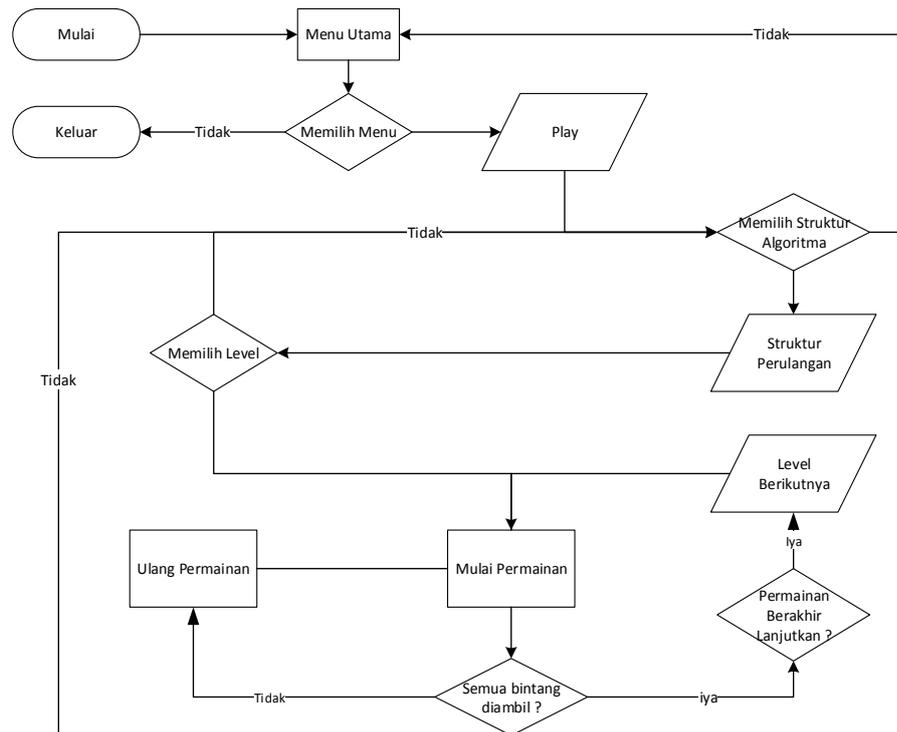


Gambar 3. 4 *Flowchart* Materi Struktur Sekuensial

Sumber: Data Olahan Peneliti

Gambar diatas untuk menjelaskan bagan alir untuk materi struktur sekuensial. Setelah pengguna memilih materi struktur sekuensial, pengguna dapat memilih

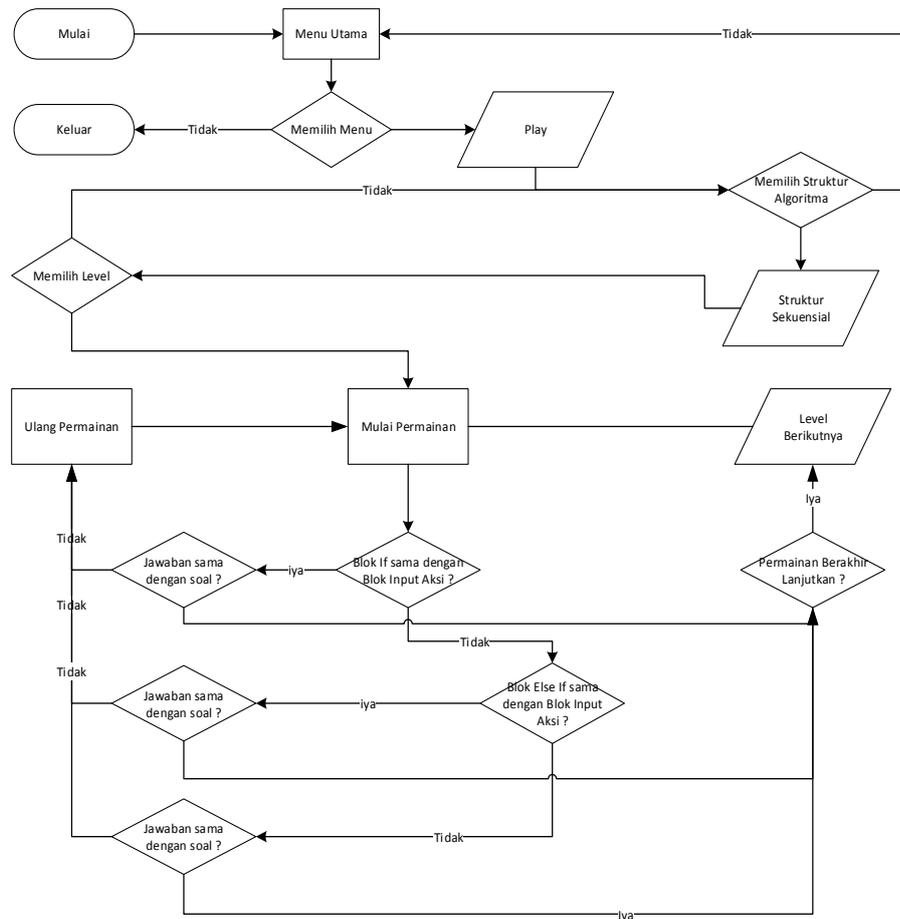
level dari level 1 sampai level 10. Setiap level ini pengguna diminta untuk mengambil objek bintang yang tersedia menggunakan algoritma struktur sekuensial. Jika semua bintang telah diambil, maka pengguna dapat lanjut ke level berikutnya.



Gambar 3. 5 *Flowchart* Materi Struktur Perulangan

Sumber: Data Olahan Peneliti

Gambar diatas menjelaskan bagan alir untuk materi struktur perulangan. Setelah pengguna memilih materi struktur perulangan, pengguna dapat memilih level dari level 1 sampai level 10. Setiap level ini pengguna diminta untuk mengambil objek bintang yang tersedia menggunakan algoritma struktur perulangan secara umum. Jika semua bintang telah diambil, maka pengguna dapat lanjut ke level berikutnya.



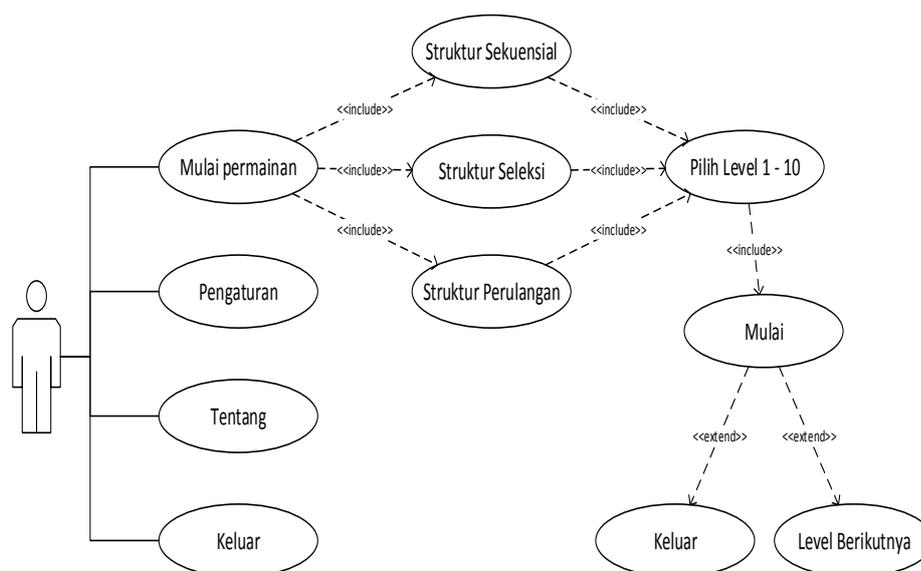
Gambar 3. 6 *Flowchart* Materi Struktur Seleksi

Sumber: Data Olahan Peneliti

Gambar diatas untuk menjelaskan bagan alir untuk materi struktur seleksi. Setelah pengguna memilih materi struktur seleksi, pengguna dapat memilih level dari level 1 sampai level 10. Level 1 sampai level 4 diberikan soal yang jawabannya meminta pengguna menggunakan algoritma struktur seleksi dengan sintaks *if* kondisi *then* aksi. Untuk level 5 sampai level 7, pengguna menggunakan algoritma struktur seleksi dengan sintaks *if* kondisi *then* aksi 1 *else* aksi 2 untuk menyelesaikan soal yang diberikan. Selanjutnya level 8 sampai level 10, pengguna

menggunakan algoritma struktur seleksi dengan sintaks *if* kondisi 1 *then* aksi 1 *else if* kondisi 2 *then* aksi 2 *else* aksi 3 untuk menyelesaikan soal.

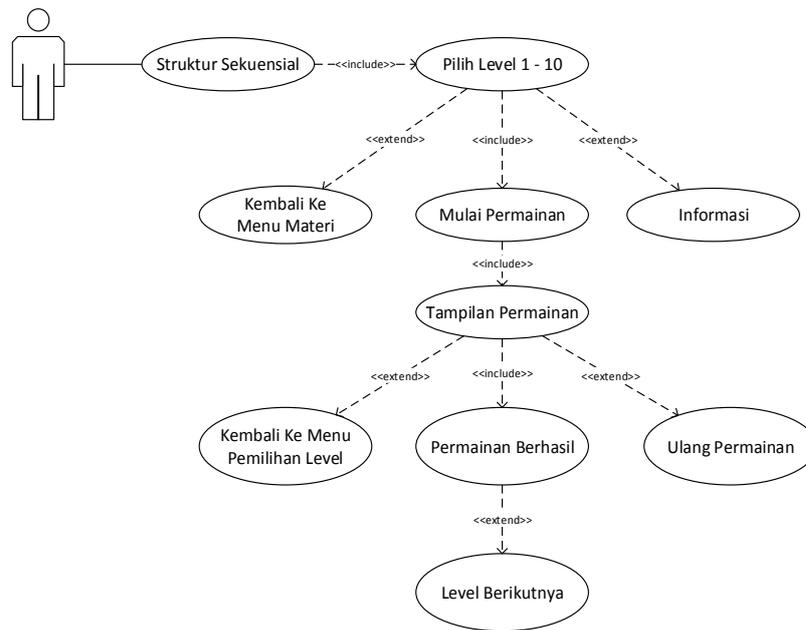
b. *Use case Diagram*



Gambar 3. 7 Diagram *Use case* Secara Umum

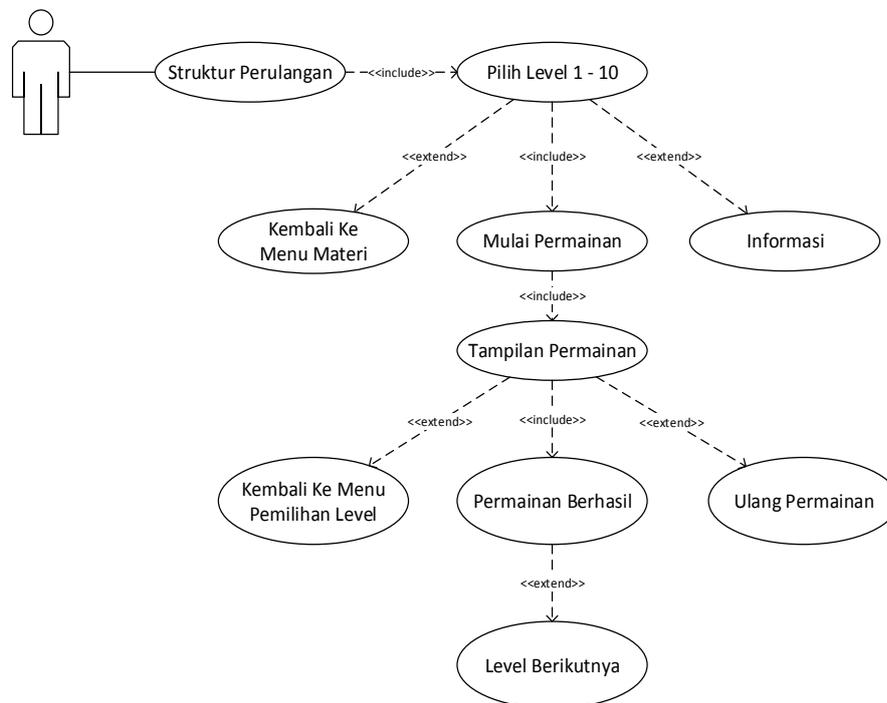
Sumber: Data Olahan Peneliti

Gambar diatas menjelaskan mengenai alur diagram aplikasi secara umum. Awalnya pengguna masuk ke dalam aplikasi. Halaman awal terdapat beberapa tombol, yang pertama tombol “pengaturan” yang berfungsi untuk mengatur volume audio permainan. Yang kedua tombol “tentang” yang berfungsi untuk menampilkan halaman tentang aplikasi. Selanjutnya tombol “Mulai” sebagai pembuka untuk menjalankan permainan. Setelah itu halaman menu pemilihan materi menampilkan tiga pilihan materi, yaitu materi struktur sekuensial, struktur seleksi dan struktur perulangan. Masing-masing materi terdapat 10 level soal yang bisa dimainkan sesuai dengan materi yang dipilih. Bila pengguna berhasil menyelesaikan level, maka pengguna dapat lanjut ke level berikutnya atau memilih untuk keluar dari aplikasi.



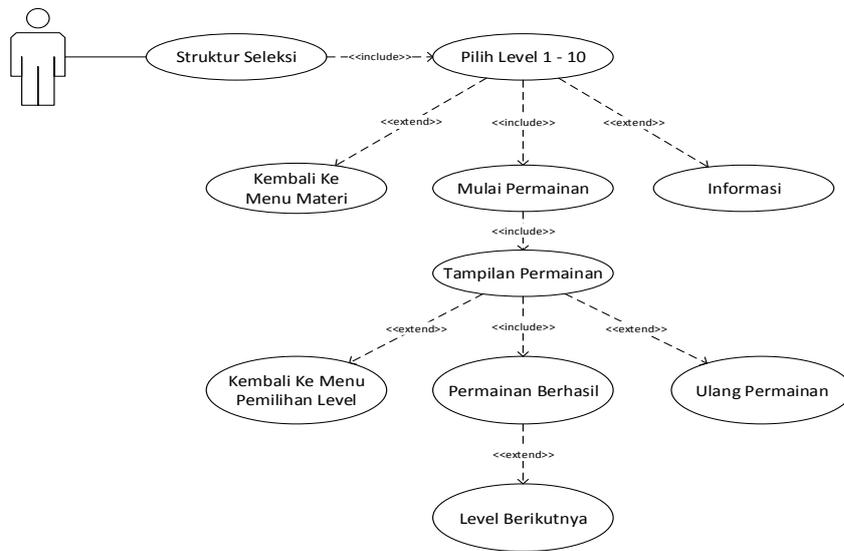
Gambar 3. 8 Diagram *Use case* Struktur Sekuensial

Sumber: Data Olahan Peneliti



Gambar 3. 9 Diagram *Use case* Struktur Perulangan

Sumber: Data Olahan Peneliti

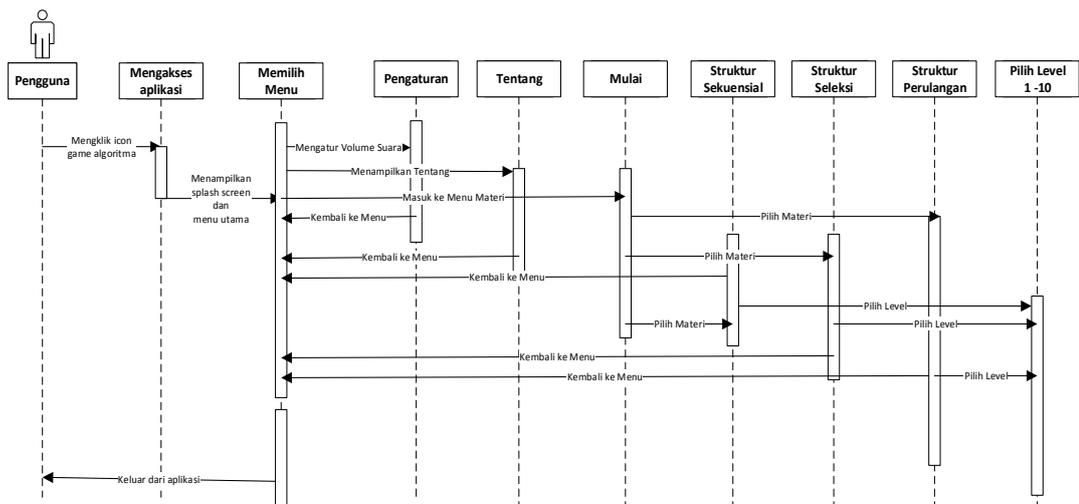


Gambar 3. 10 Diagram Use case Struktur Seleksi

Sumber: Data Olahan Peneliti

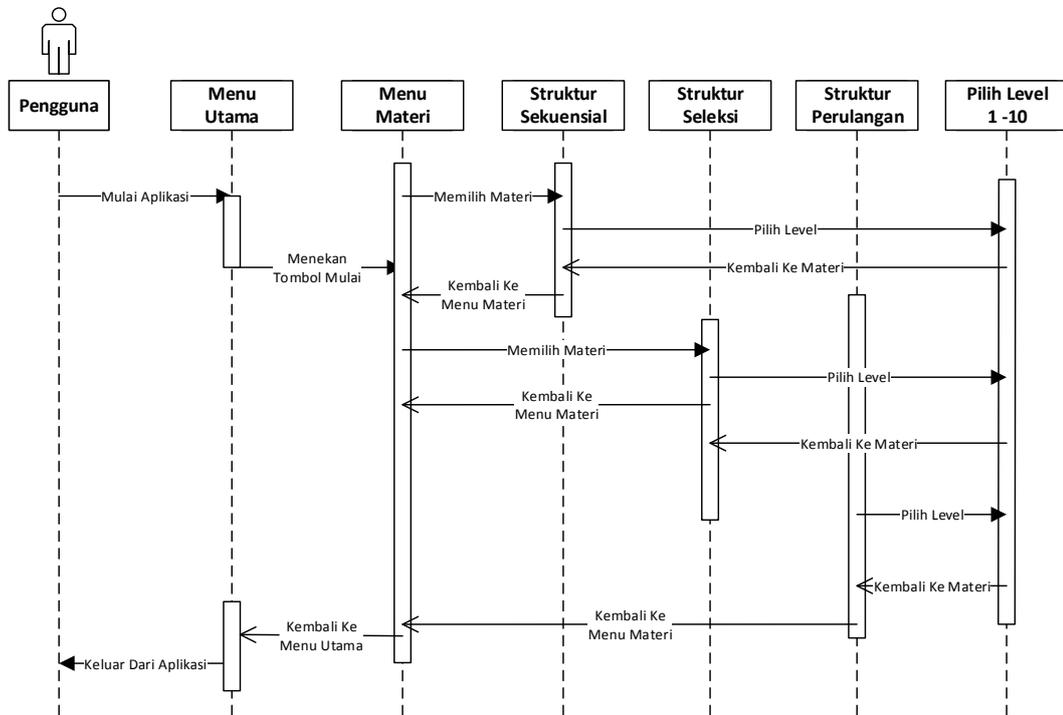
Gambar 3.8, gambar 3.9 dan gambar 3.10 menunjukkan alur diagram masing-masing materi. Setelah pengguna memilih materinya, pengguna dapat langsung memilih level yang ingin dimainkan atau pengguna dapat melihat informasi mengenai materi yang dipilih dengan menekan tombol “informasi”.

c. Sequence Diagram



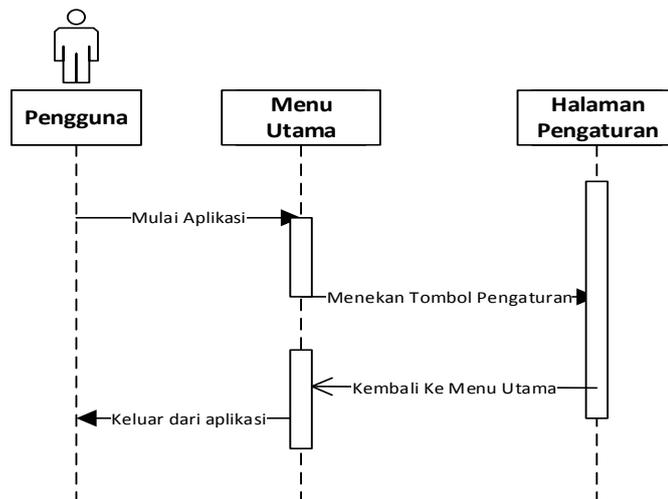
Gambar 3. 11 Diagram Sekuensial Secara Umum

Sumber: Data Olahan Peneliti



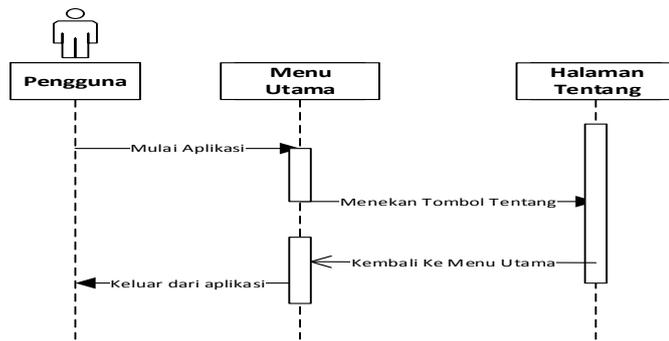
Gambar 3. 12 Diagram Sekuensial Pemilihan Materi

Sumber: Data Olahan Peneliti



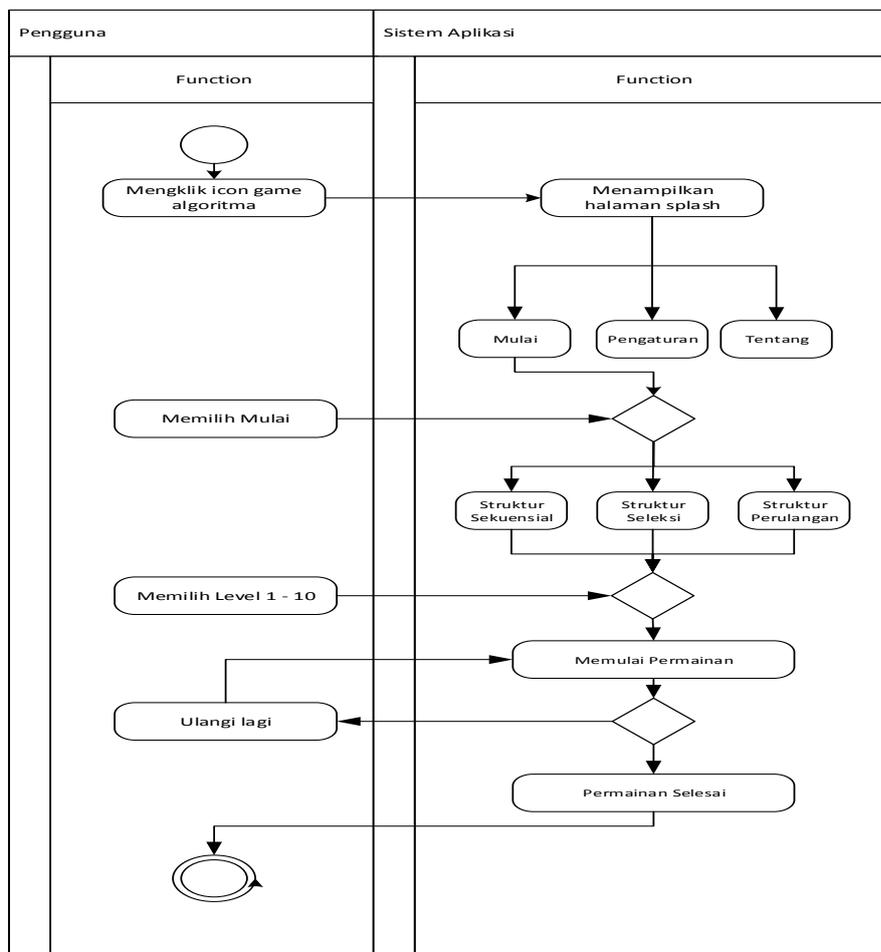
Gambar 3. 13 Diagram Sekuensial Menu Pengaturan

Sumber: Data Olahan Peneliti

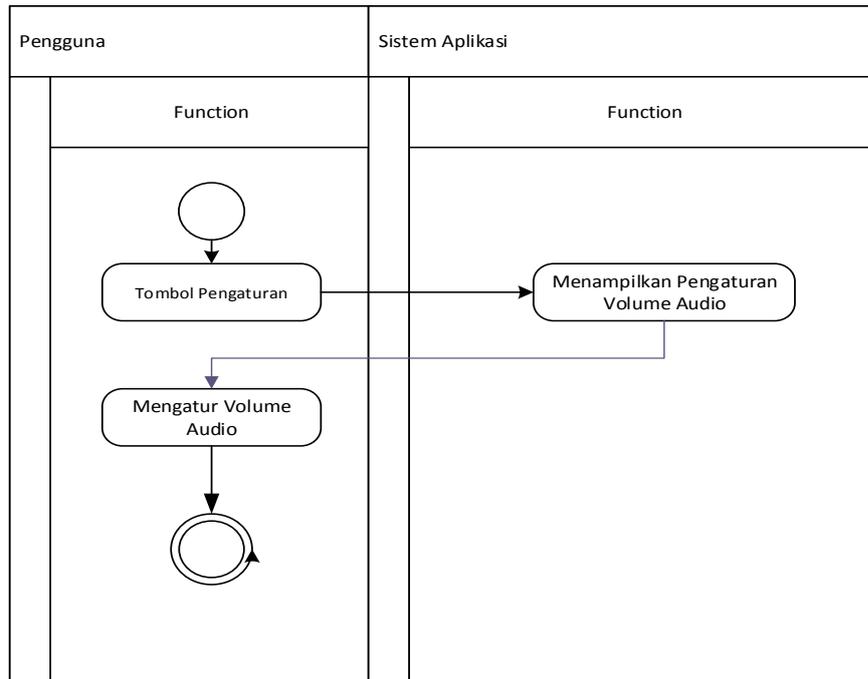


Gambar 3. 14 Diagram Sekuensial Menu Tentang  
 Sumber: Data Olahan Peneliti

d. *Activity diagram*

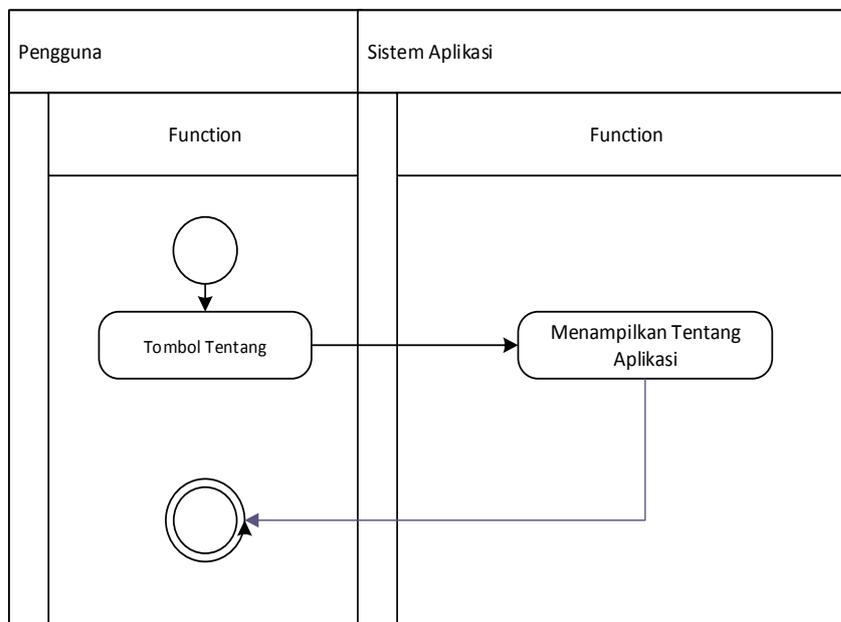


Gambar 3. 15 Diagram Aktivitas Secara Umum  
 Sumber: Data Olahan Peneliti



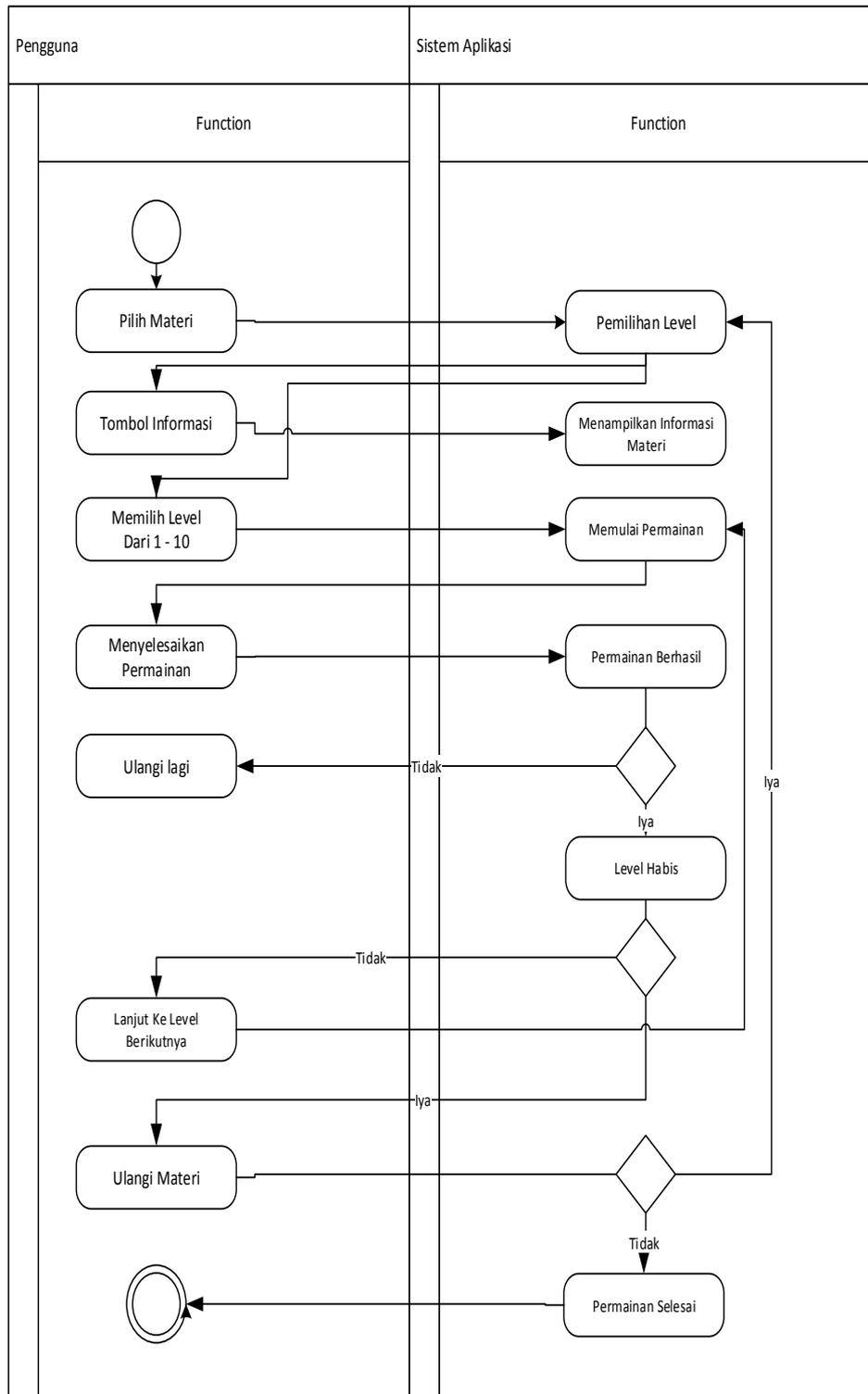
Gambar 3. 16 Diagram Aktivitas Menu Pengaturan

Sumber: Data Olahan Peneliti



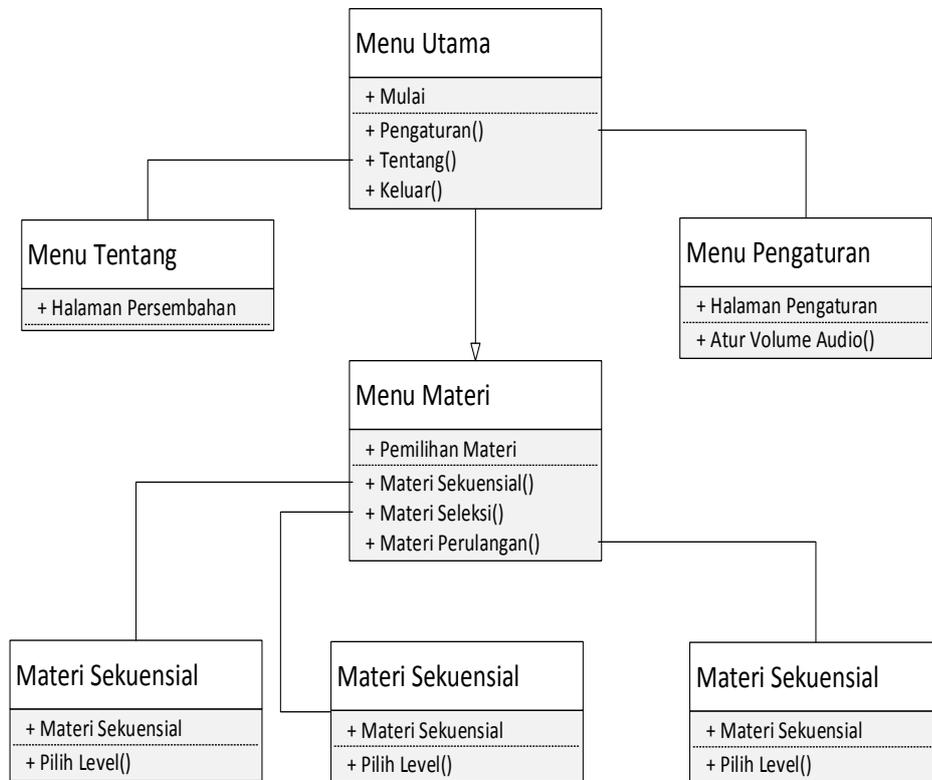
Gambar 3. 17 Diagram Aktivitas Menu Tentang

Sumber: Data Olahan Peneliti



Gambar 3. 18 Diagram Aktivitas Pemilihan Materi

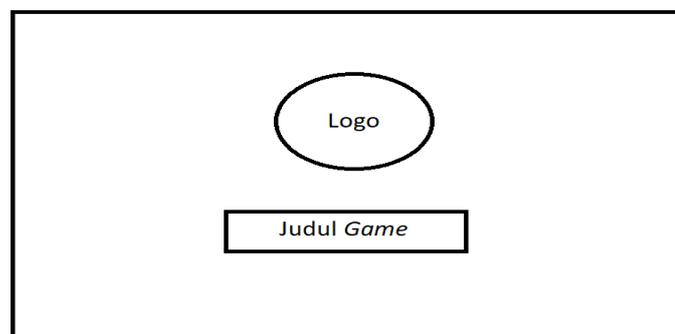
Sumber: Data Olahan Peneliti

e. *Class diagram*

Gambar 3. 19 Diagram Kelas

Sumber: Data Olahan Peneliti

## f. Perancangan Antarmuka

1) Tampilan *Splash*Gambar 3. 20 Rancangan Antarmuka Halaman *Splash*

Sumber: Data Olahan Peneliti

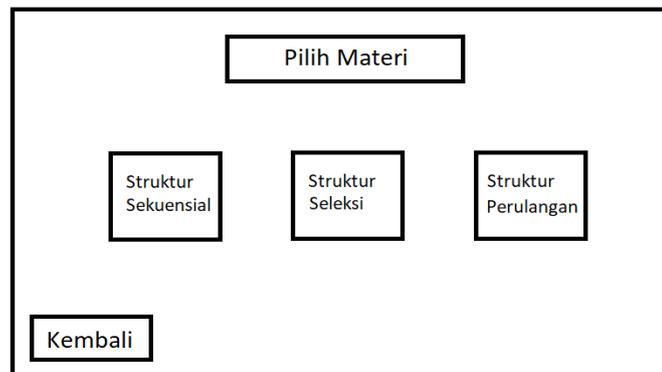
## 2) Tampilan Menu Utama



Gambar 3. 21 Rancangan Antarmuka Menu Utama

Sumber: Data Olahan Peneliti

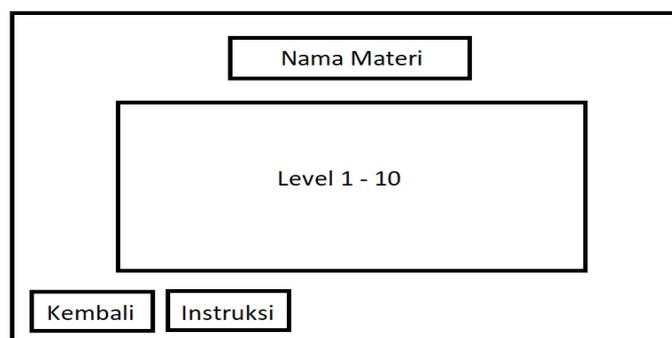
3) Tampilan Menu Permainan



Gambar 3. 22 Rancangan Antarmuka Menu Permainan

Sumber: Data Olahan Peneliti

4) Tampilan Pemilihan Level Untuk Tiap Materi



Gambar 3. 23 Rancangan Antarmuka Pemilihan Level

Sumber: Data Olahan Peneliti

## 5) Tampilan Permainan Materi Sekuensial



Gambar 3. 24 Rancangan Antarmuka Permainan Materi Sekuensial

Sumber: Data Olahan Peneliti

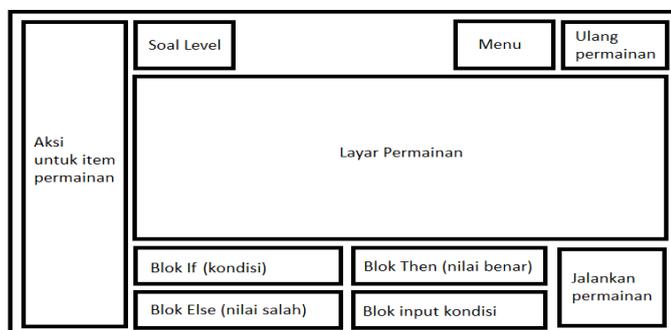
## 6) Tampilan Permainan Materi Seleksi Level 1-4



Gambar 3. 25 Rancangan Antarmuka Permainan Materi Seleksi Level 1-4

Sumber: Data Olahan Peneliti

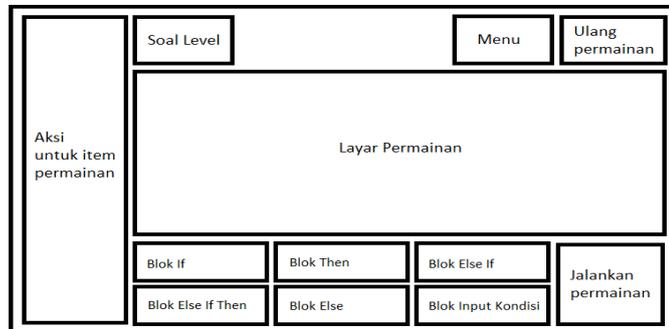
## 7) Tampilan Permainan Materi Seleksi Level 5-7



Gambar 3. 26 Rancangan Antarmuka Permainan Materi Seleksi Level 5-7

Sumber: Data Olahan Peneliti

## 8) Tampilan Permainan Materi Seleksi Level 8-10



Gambar 3. 27 Rancangan Antarmuka Permainan Materi Seleksi Level 8-10

Sumber: Data Olahan Peneliti

## 9) Tampilan Permainan Materi Perulangan



Gambar 3. 28 Rancangan Antarmuka Permainan Materi Perulangan

Sumber: Data Olahan Peneliti

**3.4.3. Material Collecting (Pengumpulan Bahan)**

Tahap ini peneliti mengumpulkan bahan-bahan yang akan digunakan dalam merancang sistem *game*. Mulai dari gambar, audio, dan lain sebagainya yang dapat peneliti peroleh secara gratis atau dapat dibuat menggunakan *software* seperti *Adobe Illustrator* dan *Blender*. Jika memungkinkan peneliti dapat mengerjakan tahap ini bersamaan dengan tahap pembuatan secara paralel.

#### **3.4.4. Assembly (Pembuatan)**

Tahap ini peneliti mulai pembuatan tampilan dalam *editor unity* berdasarkan rancangan serta proses aplikasi yang berjalan sesuai dengan tahap desain.

#### **3.4.5. Testing (Pengujian)**

Tahap *testing* (pengujian) dilakukan setelah peneliti menyelesaikan aplikasi *game* dengan menjalankan *game* dan melihat apakah ada kesalahan dalam program atau tidak. Tahap peneliti menguji *game* dapat disebut sebagai tahap pengujian alpha. Setelah peneliti merasa cukup maka *game* siap untuk diujicobakan ke tahap pengujian Beta. Dalam tahap ini pengujian dilakukan oleh sekelompok pelajar untuk mengujicoba *game* edukasi untuk pengenalan dasar logika dan algoritma.

#### **3.4.6. Distribution (Pendistribusian)**

Peneliti menyimpan *game* dalam suatu media penyimpanan. Tujuannya untuk mendistribusikan dalam bentuk file dengan ekstensi *.apk*. Dalam tahap ini juga peneliti akan melakukan evaluasi untuk pengembangan *game* edukasi yang sudah jadi agar lebih baik kedepannya.

### **3.5. Lokasi dan Jadwal Penelitian**

Penelitian *game* edukasi untuk pengenalan dasar algoritma berbasis *android* ini dilakukan di SMK Kolese Tiara Bangsa yang beralamat di Komplek Permata Hijau No. 5-9 Jl. Engku Putri Batam Center, Teluk Tering, Kec. Batam Kota, Kota Batam, Prov. Kepulauan Riau.



Gambar 3. 29 Peta Lokasi Batam

Sumber: Data Olahan Sendiri

Pelaksanaan pengembangan dan penelitian dilakukan sekitar kurang lebih enam bulan, yaitu pada bulan September 2019 sampai dengan bulan Februari 2020.

Berikut jadwal perencanaan pelaksanaan penelitian:

Tabel 3. 2 Perencanaan Pelaksanaan Jadwal Penelitian

Jadwal	September 2019				Oktober 2019				November 2019				Desember 2019				Januari 2020				Februari 2020			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Kegiatan\Minggu ke-																								
Bab I Pendahuluan	█	█	█	█																				
Bab II Teori Dasar					█	█	█	█	█	█														
Bab III Metode Penelitian										█	█	█	█	█	█									
Merancang dan Menguji <i>game</i> (alpha)											█	█	█	█	█	█								
Merevisi jika perlu													█	█	█	█	█							
Mengujicobakan kepada pelajar (beta)																	█	█	█	█				
Bab IV Hasil dan Pembahasan																				█	█	█	█	
Bab V Kesimpulan																								
Upload Jurnal																								

Sumber: Data Olahan Sendiri