

**IMPLEMENTASI *LEFT CORNER PARSING* UNTUK PEMBELAJARAN  
GRAMMAR BAHASA INGGRIS PADA *GAME 3D ADVENTURE "GO TO  
LONDON"***

**SKRIPSI**

Oleh:

**FACHRY KHUSAINI**

**NIM. 09650137**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM  
MALANG  
2013**

**IMPLEMENTASI *LEFT CORNER PARSING* UNTUK PEMBELAJARAN  
GRAMMAR BAHASA INGGRIS PADA *GAME 3D ADVENTURE "GO TO  
LONDON"***

**SKRIPSI**

Diajukan Kepada:

Fakultas Sains dan Teknologi

Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang

Untuk Memenuhi Salah Satu Persyaratan Dalam

Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:

**FACHRY KHUSAINI**

**NIM. 09650137**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UNIVERSITAS ISLAM NEGERI (UIN) MAULANA MALIK IBRAHIM**

**MALANG**

**2013**

**IMPLEMENTASI *LEFT CORNER PARSING* UNTUK PEMBELAJARAN  
GRAMMAR BAHASA INGGRIS PADA *GAME 3D ADVENTURE "GO TO  
LONDON"***

Oleh:

**FACHRY KHUSAINI**

**NIM. 09650137**

Telah disetujui oleh:

**Dosen Pembimbing I**

**Dosen Pembimbing II**

**Hani Nurhayati, M.T**  
**NIP. 19780625 200801 2 006**

**Fachrul Kurniawan M.MT**  
**NIP. 19771020 200912 1 001**

**Tanggal, Juli 2013**

**Mengetahui dan Mengesahkan,  
Ketua Jurusan Teknik Informatika**

**Ririen Kusumawati, M.Kom**  
**NIP. 19720309 200501 2 002**

**IMPLEMENTASI *LEFT CORNER PARSING* UNTUK PEMBELAJARAN  
GRAMMAR BAHASA INGGRIS PADA *GAME 3D ADVENTURE "GO TO  
LONDON"***

**SKRIPSI**

Oleh:

**FACHRY KHUSAINI**

**NIM. 09650137**

Telah Dipertahankan di Depan Dewan Penguji Tugas akhir dan  
Dinyatakan Diterima Sebagai Salah Satu Persyaratan Untuk  
Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal, 9 Juli 2013

Susunan Dewan Penguji	Tanda Tangan
1. Penguji Utama : <u>Irwan Budi Santoso, M.Kom</u> NIP. 19770103 201101 1 004	( )
2. Ketua : <u>A'la Svauqi, M.Kom</u> NIP. 19771201 200801 1 007	( )
3. Sekretaris : <u>Hani Nurhayati, M.T</u> NIP. 19780625 200801 2 006	( )
4. Anggota : <u>Fachrul Kurniawan, M.MT</u> NIP. 19771020 200901 1 001	( )

**Mengetahui dan Mengesahkan,  
Ketua Jurusan Teknik Informatika**

**Ririen Kusumawati, M.Kom  
NIP. 19720309 200501 2 002**

**ORISINALITAS PENELITIAN**

Saya yang bertanda tangan di bawah ini:

Nama : Fachry Khusaini

NIM : 09650137

Fakultas / Jurusan : Sains dan Teknologi / Teknik Informatika

Judul Penelitian : **IMPLEMENTASI *LEFT CORNER PARSING*  
UNTUK PEMBELAJARAN *GRAMMAR*  
BAHASA INGGRIS PADA *GAME 3D ADVENTURE*  
*“GO TO LONDON”***

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur penjiplakan, maka saya bersedia untuk mempertanggung jawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 5 Juli 2013

Yang Membuat Pernyataan,

Fachry Khusaini  
NIM. 09650137

## HALAMAN PERSEMBAHAN

*Dengan mengucap rasa syukur dan mengharap ridho Allah*

*Kupersembahkan karya ini kepada:*

*Bapak dan Mama tercinta*

***Jaenuri dan Darlina***

*Atas Segalanya.*

*Semoga Allah SWT selalu melindungi*

*Dan menyayangi keduanya ...*

*Serta adik tersayang **Liza Pangestika** yang memberi  
kelengkapan di dalam keluarga...*

*Teman-teman kontrakkan MSC08 dan TI09 yang telah  
berbagi ilmunya*

*Khususnya **Mas Udin, Mas Ardi, dan Mbak Atim** yang  
membantu dalam memberi solusi dalam setiap masalah...*

*Terspecial **Zifora Nur Baiti** yang tak henti memberi  
semangat selama pengerjaan tugas akhir ini...*

## KATA PENGANTAR

*Assalamu'alaikum Wr. Wb.*

Alhamdulillah, segala puja dan puji syukur Alhadulillah kami panjatkan kehadirat Allah SWT yang Maha Pengasih lagi Maha Penyayang yang telah memberikan rahmat, taufik, hidayah dan inayah-Nya kepada kita serta memberikan nikmat Islam dan Iman serta tak lupa nikmat kesehatan yang diberikan kepada penulis khususnya sehingga penulis dapat menyelesaikan karya ilmiah dengan judul “IMPLEMENTASI LEFT CORNER PARSING UNTUK PEMBELAJARAN GRAMMAR BAHASA INGGRIS PADA GAME 3D ADVENTURE “GO TO LONDON””. Penelitian ini dimaksudkan untuk memenuhi salah satu syarat dalam meraih gelar Sarjana Komputer (S.Kom) di Fakultas Sains dan Teknologi Jurusan Teknik Informatika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.

Penulis menyadari bahwa banyak pihak yang telah membantu dalam menyelesaikan penulisan tugas akhir ini. Untuk itu, iringan doa dan ucapan terima kasih yang sebesar-besarnya penulis sampaikan kepada:

1. Prof. Dr. H. Mudjia Raharjo, M.Si, selaku rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
2. Dr. Hj. Bayyinatul Muchtaromah, drh., M.Si, selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
3. Hani Nurhayati, M.T dan Fachrul Kurniawan, M.MT selaku dosen pembimbing I dan II, yang telah meluangkan waktu untuk membimbing, memotivasi, mengarahkan, serta memberikan saran, kemudahan dan kepercayaan dalam dalam penyelesaian tugas akhir ini.
4. Ririen Kusumawati, M.Kom selaku ketua jurusan Teknik Informatika Fakultas Sains Dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.

5. Seluruh civitas akademik dan dosen Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang yang telah mengalirkan ilmu, pengetahuan, pengalaman, dan wawasannya, sebagai pedoman dan bekal bagi penulis.
6. Teman-teman satu jurusan Teknik Informatika angkatan 2009 yang telah bersedia berbagi ilmu dan informasi selama menimba ilmu di Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang ini.
7. Dan semua pihak yang tidak dapat penulis sebutkan satu-persatu. Penulis ucapkan terimakasih banyak atas bantuan, dan motivasinya.

Akhirnya atas segala kekurangan dari penyusunan skripsi ini, sangat diharapkan kritik dan saran yang bersifat konstruktif dari semua pembaca demi memperbaiki kualitas penulisan selanjutnya. Semoga apa yang telah tertulis di dalam skripsi ini dapat memberikan kontribusi yang bermanfaat dan menambah khasanah ilmu pengetahuan. Amien...

*Wassalamu'alaikum Wr. Wb.*

Malang, Juli 2013

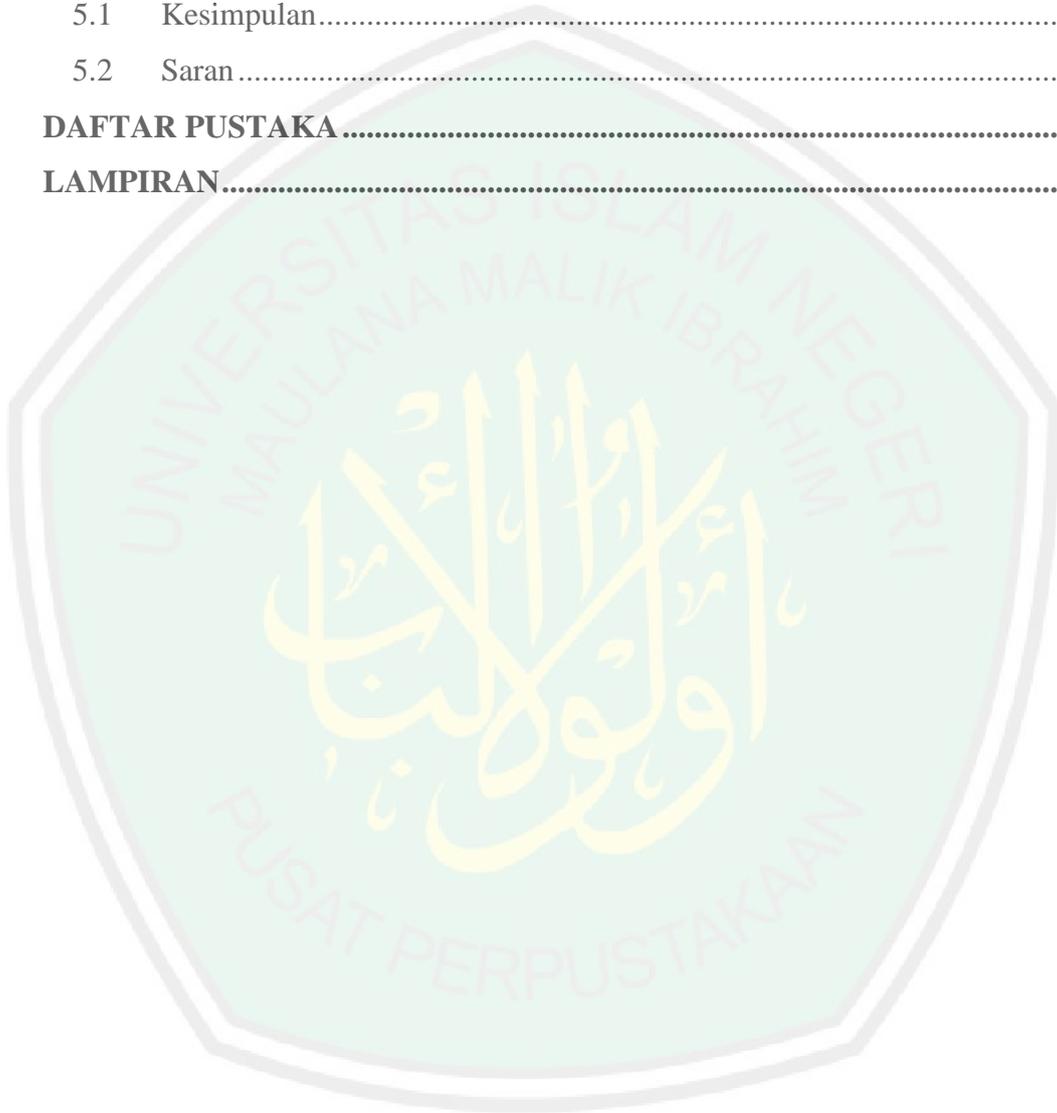
Penulis

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>HALAMAN PENGAJUAN .....</b>	<b>ii</b>
<b>HALAMAN PERSETUJUAN .....</b>	<b>iii</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>iv</b>
<b>HALAMAN PERNYATAAN.....</b>	<b>v</b>
<b>HALAMAN PERSEMBAHAN .....</b>	<b>vi</b>
<b>KATA PENGANTAR.....</b>	<b>vii</b>
<b>DAFTAR ISI.....</b>	<b>ix</b>
<b>DAFTAR GAMBAR.....</b>	<b>xii</b>
<b>DAFTAR TABEL .....</b>	<b>xiv</b>
<b>ABSTRAK .....</b>	<b>xv</b>
<b>ABSTRACT .....</b>	<b>xvi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian.....	5
1.5 Manfaat Penelitian.....	5
1.6 Sistematika Penulisan.....	5
1.7 Metode Penelitian.....	7
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>8</b>
2.1 Permainan ( <i>game</i> ) dan Karakteristik.....	8
2.1.1 Elemen Game.....	9
2.1.2 Genre Game .....	12
2.2 Bahasa Inggris .....	18
2.3 <i>Grammar</i> (Tata Bahasa) .....	20
2.4 <i>AI (Artificial Inteligence)</i> .....	22
2.4.1 Sejarah <i>AI (Artificial Inteligence)</i> .....	23

2.5	Parsing .....	24
2.6	<i>Left Corner Parsing</i> .....	25
2.7	3D Game Studio .....	26
2.7.1	Editor .....	28
2.8	Pendidikan Dini Usia Sekolah Dasar .....	30
<b>BAB III RANCANGAN DESAIN SISTEM .....</b>		<b>32</b>
3.1	Perancangan Sistem .....	32
3.1.1	Keterangan Umum Game .....	32
3.1.2	Storyboard Game .....	34
3.1.3	Penampilan Umum Game .....	35
3.1.4	Deskripsi Karakter .....	36
3.2	Perancangan Aplikasi .....	37
3.2.1	Perancangan Antarmuka Pengantar .....	38
3.2.2	Perancangan Antarmuka Menu Game .....	38
3.2.3	Perancangan Antarmuka Game .....	39
3.2.4	Perancangan Antarmuka Penutup .....	39
3.2.5	Perancangan Alur Permainan Pada Level Pertama, Level Kedua, dan Level Ketiga .....	40
3.2.6	Perancangan Alur Permainan Pada Level Keempat .....	45
3.3	Perancangan Left Corner Parsing Sebagai Pemeriksa Kalimat .....	47
3.4	Kebutuhan Sistem .....	52
3.5	Cara Memainkan Game .....	53
3.6	Kebutuhan PC Pemain .....	54
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>		<b>55</b>
4.1	Implementasi .....	55
4.2	Implementasi <i>Artificial Intelligence</i> Proses Pemeriksaan Kalimat .....	55
4.3	Implementasi Aplikasi Game .....	59
4.3.1	Antarmuka Intro .....	59
4.3.2	Antarmuka Menu .....	60
4.3.3	Antarmuka Latar Level 1, 2, dan 3 .....	62
4.3.4	Antarmuka Latar Level 4 .....	66

4.4	Pengujian Game .....	67
4.5	Integrasi Game Pembelajaran Bahasa Inggris pada Ayat Al-Qur'an.....	69
<b>BAB V PENUTUP.....</b>		<b>73</b>
5.1	Kesimpulan.....	73
5.2	Saran.....	73
<b>DAFTAR PUSTAKA.....</b>		<b>75</b>
<b>LAMPIRAN.....</b>		<b>77</b>



## DAFTAR GAMBAR

Gambar 2.1 <i>Parse Tree</i> .....	22
Gambar 2.2 <i>Task Domains of Artificial Intelligence</i> .....	24
Gambar 2.3 <i>Left Corner Parsing</i> .....	26
Gambar 2.4 Tampilan <i>World Editor</i> .....	28
Gambar 2.5 Tampilan <i>Model Editor</i> .....	29
Gambar 2.6 Tampilan <i>Script Editor</i> .....	30
Gambar 3.1 <i>Storyboard Game</i> .....	35
Gambar 3.2 Karakter Utama (Imran) .....	36
Gambar 3.3 NPC Penolong (Pak Rizal) .....	37
Gambar 3.4 NPC Perantara (Pak Bagus) .....	37
Gambar 3.5 Pengantar <i>Game</i> .....	38
Gambar 3.6 Menu <i>Game</i> .....	39
Gambar 3.7 Awal Permainan <i>Game</i> .....	39
Gambar 3.8 Hasil Akhir <i>Game</i> .....	40
Gambar 3.9 Alur Level Ke-1 .....	41
Gambar 3.10 Alur Level Ke-2 .....	43
Gambar 3.11 Alur Level Ke-3 .....	44
Gambar 3.12 Alur Level Ke-4 .....	46
Gambar 3.13 Alur Sistem Pemeriksaan .....	47
Gambar 3.14 Proses Tokenization .....	49
Gambar 3.15 Proses Pencocokan Kata .....	50

Gambar 3.16 Pemeriksaan Pola Grammar .....	51
Gambar 4.1 Intro Pengantar Game .....	60
Gambar 4.2 Intro Pengantar Game Ke-2 .....	60
Gambar 4.3 Intro Pengantar Game Ke-3 .....	60
Gambar 4.4 Menu <i>Game</i> .....	61
Gambar 4.5 Menu <i>New Game</i> .....	61
Gambar 4.6 Menu <i>Setting</i> .....	61
Gambar 4.7 Menu <i>About</i> .....	61
Gambar 4.8 Menu <i>Help</i> .....	61
Gambar 4.9 Menu <i>Exit</i> .....	61
Gambar 4.10 Latar Level .....	62
Gambar 4.11 Pencarian Kartu di Dalam Rum .....	62
Gambar 4.12 Pemberian Tugas .....	62
Gambar 4.13 Pencarian Portal .....	63
Gambar 4.14 Tugas Menyusun yang Diberika .....	63
Gambar 4.15 Tugas yang Harus Dipenuhi Level 1 .....	64
Gambar 4.16 Tugas yang Harus Dipenuhi Level 2 .....	65
Gambar 4.17 Tugas yang Harus Dipenuhi Level 3 .....	65
Gambar 4.18 Tugas yang Harus Dipenuhi Level 4 .....	65
Gambar 4.19 Latar Level-4 .....	66
Gambar 4.20 Latar Lain di Level-4 .....	67
Gambar 4.21 Menemui NPC Akhir .....	67
Gambar Uji Coba .....	77

## DAFTAR TABEL

Tabel 3.1 Fungsi Keyboard dan Mouse untuk Memainkan Game .....	54
Tabel 3.2 Kebutuhan PC Pemain .....	54
Tabel 4.1 Hasil Penerapan <i>Left Corner Parsing</i> .....	56
Tabel 4.2 Fungsi Menu Game .....	60
Tabel 4.3 Hasil Rekapitulasi Uji Coba .....	68



## ABSTRAK

Khusaini, Fachry. 2013. **Implementasi Left Corner Parsing Untuk Pembelajaran Grammar Bahasa Inggris Pada Game 3D Adventure “Go To London”**. Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Pembimbing: (I) Hani Nurhayati, M.T (II) Fachrul Kurniawan M.MT.

---

---

**Kata kunci:** *Left Corner Parsing, Top Down Parsing, Bottom Up parsing, Pemeriksa, Bahasa Inggris.*

Bahasa Inggris adalah bahasa internasional yang digunakan seluruh orang di dunia untuk berinteraksi satu sama lain. Mempelajari bahasa Inggris sangat diperlukan pada saat ini. Banyak metode pembelajaran yang digunakan untuk mempelajari bahasa Inggris. Game merupakan media pembelajaran yang menyenangkan, terutama bagi anak. Melalui game, anak dapat belajar sambil bermain dalam mengembangkan kemampuannya. Game yang dibangun adalah game pembelajaran grammar bahasa Inggris menggunakan algoritma left corner parsing sebagai pemeriksa dalam kalimat yang dibuat.

Algoritma yang digunakan dalam memeriksa sebuah kalimat sangat banyak macamnya, salah satunya algoritma left corner parsing. Algoritma left corner parsing merupakan gabungan dari dua algoritma, yaitu algoritma top down parsing dan bottom up parsing. Tugas algoritma ini memeriksa setiap kata dalam sebuah kalimat, kemudian mencocokkan sebuah pola grammar terhadap hasil pemeriksaan tersebut. Proses pemeriksaan ini yang akan menjadi pemeriksa kata dalam game.

Dari uji coba yang dilakukan, metode left corner parsing dapat mengenali pola grammar dengan sangat baik, akan tetapi dalam mengenali kata yang tepat untuk membentuk sebuah kalimat yang benar masih menjadi kekurangan.

## ABSTRACT

Khusaini, Fachry. 2013. **Implementation Left Corner Parsing for Learning English Grammar In 3D Adventure Game "Go To London"**. Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang, Supervisor: (I) Hani Nurhayati, M.T. (II) Fachrul Kurniawan M.MT.

---

---

**Kata kunci:** *Left Corner Parsing, Top Down Parsing, Bottom Up parsing, Examiner, English Language.*

English is the international language that is used all over the world to interact with each other. Learning the English language is necessary at this time. Many learning methods are used to study the English language. Game is a fun learning medium, especially for children. Through games, children can learn while playing in developing his abilities. Game is a game built learning English grammar using the left corner parsing algorithm as an examiner in the phrase made.

Algorithms used in examining a sentence so many kinds, one of which left corner parsing algorithm. Left corner parsing algorithm is a combination of two algorithms, algorithms top down parsing and bottom up parsing. The assignment algorithms examine each word in a sentence, then match a grammar pattern of the results of the investigation. The examination process will be the examiner said in the game.

From the experiments performed, the method can recognize the left corner parsing grammar patterns very well, but in recognizing the right words to form a sentence that was still a shortage.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Bahasa Inggris adalah salah satu dari banyak macam bahasa yang digunakan. Bahasa Inggris sering digunakan dikarenakan bahasa Inggris adalah bahasa internasional yang digunakan oleh banyak negara di dunia untuk berkomunikasi. Dahulu begitu banyak invasi negara Inggris terhadap negara-negara lain, dan akhirnya bahasa Inggris diakui sebagai bahasa internasional (Priyana, 2013). Bahasa Inggris juga dipakai sebagai pengantar dalam acara-acara atau pertemuan-pertemuan resmi tingkat internasional (Hidayat, 2006).

Negara Indonesia dalam hal bahasa Inggris sudah mulai dipelajari dari tingkat sekolah dasar (SD). Dimulai dari bagaimana cara menulis yang benar, lalu kemudian berlanjut dengan cara pengucapannya yang benar. Dalam faktanya walaupun bahasa Inggris sudah dipelajari dari SD, akan tetapi masih banyak juga yang kurang memahami bagaimana cara pengucapan bahasa Inggris dan juga penulisannya. Bahkan yang sudah menjadi guru/pengajar pun belum tentu bisa dalam hal menulis ataupun mengucapkan bahasa Inggris dengan benar.

Pada dasarnya mempelajari sebuah bahasa itu bersifat penting, dikarenakan ketika seseorang yang kita tidak kenal bertanya atau berkomunikasi dengan kita, akan tetapi kita tidak paham akan apa yang dikatakan orang tersebut. Seperti yang disebutkan pada terjemahan Q.S. Al-Kahfi ayat 93 yang berbunyi:

حَتَّىٰ إِذَا بَلَغَ بَيْنَ السَّدَّيْنِ وَجَدَ مِنْ دُونِهِمَا قَوْمًا لَّا يَكَادُونَ يَفْقَهُونَ قَوْلًا ﴿١٣﴾

*Artinya: “hingga apabila Dia telah sampai di antara dua buah gunung, Dia mendapati di hadapan kedua bukit itu suatu kaum yang hampir tidak mengerti pembicaraan”.*

Maksud dari ayat diatas yaitu mereka tidak bisa memahami bahasa orang lain karena bahasa mereka amat beda dari bahasa yang lain, dan merekapun tidak dapat menerangkan maksud mereka dengan jelas karena kekurangan kecerdasan mereka dalam berbahasa.

Dalam pembelajaran bahasa Inggris terdapat sebuah tata bahasa yang digunakan sebagai standar untuk mengetahui bahasa yang digunakan itu sudah benar atau belum. Standar yang digunakan biasa disebut *grammar*. Di dalam *grammar* terdapat beberapa cara atau metode untuk menyusun sebuah kalimat dalam membuat sebuah cerita atau untuk berkomunikasi dengan orang lain dengan benar. Dengan mempelajari *grammar* dengan baik dan benar akan menghasilkan sebuah pemahaman tentang struktur bahasa inggris yang baik dan benar. Dari permasalahan pemahaman *grammar* ini muncullah ide untuk membuat sebuah pembelajaran tentang *grammar* bahasa Inggris dimulai dari anak usia dini melalui game *adventure “Go to London”* yang akan dibangun.

Game ini menceritakan sebuah Sekolah Dasar (SD) yang memberikan sebuah beasiswa untuk sekolah di negeri Inggris yaitu di kota London. Syarat untuk mendapatkan beasiswa tersebut ialah dengan membuat sebuah cerita/pengalaman. Bertepatan dengan liburan sekolah maka cerita/pengalaman yang digunakan ialah cerita/pengalaman selama liburan selama satu minggu.

Kemudian ada seorang murid yang menginginkan beasiswa tersebut, setelah itu dimulailah petualangan dari murid tersebut untuk mendapatkan beasiswa sekolah di London. Berawal dari cerita petualangan untuk mendapatkan beasiswa ke London tersebut, maka game ini diberi nama “*Go to London*”, dikarenakan negara Inggris dimana kota London berada adalah pusat dari grammar bahasa Inggris atau biasa disebut Inggris British.

Tugas dari siswa tersebut yaitu membuat beberapa kalimat yang akan dikumpulkan setiap harinya. Untuk membuat kalimat siswa itu harus menemukan kartu untuk digunakan. Kartu tersebut bisa ditemukan di dalam sebuah rumah-rumah, dan di dalam rumah-rumah tersebut terdapat penjaganya. Penjaga disini diistilahkan dengan NPC (Non-Player Character) Jika kartu tersebut telah diambil oleh siswa, maka NPC penjaga akan menghadangnya untuk memberi tugas membuat kalimat dari kartu yang diperoleh siswa tersebut.

Setiap jawaban yang diterima oleh NPC penjaga, NPC penjaga akan membukakan pintu untuk siswa tersebut. Dalam hal penilaian jawaban yang benar dan salah, algoritma *Left Corner Parsing* diterapkan untuk memeriksa kalimat yang dibuat. Setelah keluar dari ruangan tersebut, siswa meneruskan petualangannya. Siswa tersebut diberikan sebuah kesehatan dalam waktu satu level. Setelah membuat tugas kalimat kesehatannya akan berkurang, sehingga apabila kesehatannya sudah habis siswa tersebut harus pulang untuk istirahat dan melanjutkan petualangan pada esok harinya.

Setelah liburan berakhir kalimat-kalimat yang dibuat selama 4 hari dikumpulkan ke gurunya untuk dinilai oleh pihak pemerintah. Apabila jawaban

yang dinilai oleh pihak pemerintah tersebut memiliki jawaban benar lebih dari 70 persen, maka siswa tersebut berhasil mendapatkan beasiswa ke London. Apabila tidak maka harus mengulangi permainan kembali.

## 1.2 Rumusan Masalah

Dari pemaparan latar belakang di atas, maka perumusan masalah dalam penelitian ini, yaitu bagaimana menerapkan algoritma *left corner parsing* sebagai pemeriksa *grammar* bahasa Inggris di dalam sebuah *game adventure* “Go to London”?

## 1.3 Batasan Masalah

Agar penelitian ini lebih terarah dengan jelas dan sistematis, maka permasalahan kami batasi sebagai berikut:

1. Game yang dibangun berjenis *game adventure*.
2. Game bersifat *single player*.
3. *Game engine* yang digunakan adalah *3D Game Studio A8*.
4. Bahasa pemrograman yang digunakan adalah *Lite-C*.
5. Algoritma yang digunakan adalah *Left Corner Parsing*.
6. Bentuk kalimat yang digunakan dalam game yaitu positif tidak termasuk kalimat pertanyaan.
7. Game ini untuk anak-anak usia 8-13 tahun.
8. Kalimat yang digunakan di dalam game menggunakan kalimat sederhana.

9. Setiap kata dianggap hanya memiliki satu arti untuk menghindari ambiguitas yang terjadi dalam pemilihan tipe atau jenis dari kata tersebut.
10. Kalimat tidak mengandung *preposition* atau *to be* yang disingkat seperti “*i’m*” yang seharusnya “*I am*”.
11. Tense yang digunakan yaitu *simple present tense*, *simple past tense*, dan *simple future tense*.
12. Pemeriksaan yang dilakukan hanya sebatas pemeriksaan pola kalimat.

#### 1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk menerapkan algoritma *left corner parsing* sebagai pemeriksa kalimat pada game 3D “*Go to London*” adventure.

#### 1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah untuk mempermudah anak dalam mempelajari bahasa Inggris dari segi *grammar*. Pembuatan game ini dapat membantu mengenali pola *grammar* sebelum membuat sebuah kalimat bahasa Inggris yang sesuai dengan aturan *grammar*.

#### 1.6 Sistematika Penulisan

Adapun sistematika penulisan skripsi ini adalah sebagai berikut:

### **BAB I Pendahuluan**

Pendahuluan, membahas tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, metodologi dan sistematika penyusunan laporan tugas akhir. Uraian pada bab pendahuluan ini berisi gambaran tentang maksud dan

tujuan dalam penelitian ini. Diawali dengan berbagai cara atau metode belajar yang digunakan untuk memahami sebuah pelajaran. Hal tersebut dijawab dengan merancang dan membangun sebuah aplikasi permainan atau *game* pembelajaran *grammar* bahasa Inggris menggunakan metode *Left Corner Parsing* yang akan dipaparkan lebih lanjut di bab selanjutnya.

## **BAB II Tinjauan Pustaka**

Berisikan beberapa teori yang mendasari dalam penyusunan tugas akhir ini. Adapun yang dibahas dalam bab ini adalah dasar teori yang berkaitan dengan pembahasan tentang cara pembelajaran *grammar* bahasa Inggris, dan metode *Left Corner Parsing*. Teori yang dijabarkan ini akan menjadi acuan dalam melakukan perancangan sistem yang akan dibangun dan dibahas di bab selanjutnya.

## **BAB III Rancangan Desain Sistem**

Rancangan desain sistem berisi tentang tahap perancangan dalam pembuatan *game* yang akan dibangun. Perancangan aplikasi terdiri perancangan proses-proses utama dan desain aplikasi yang terdiri desain input, desain output, desain antarmuka, desain karakter dan desain lingkungan *game*. Penjelasan proses utama yang digunakan sistem untuk memproses input yang masuk

## **BAB IV Hasil dan Pembahasan**

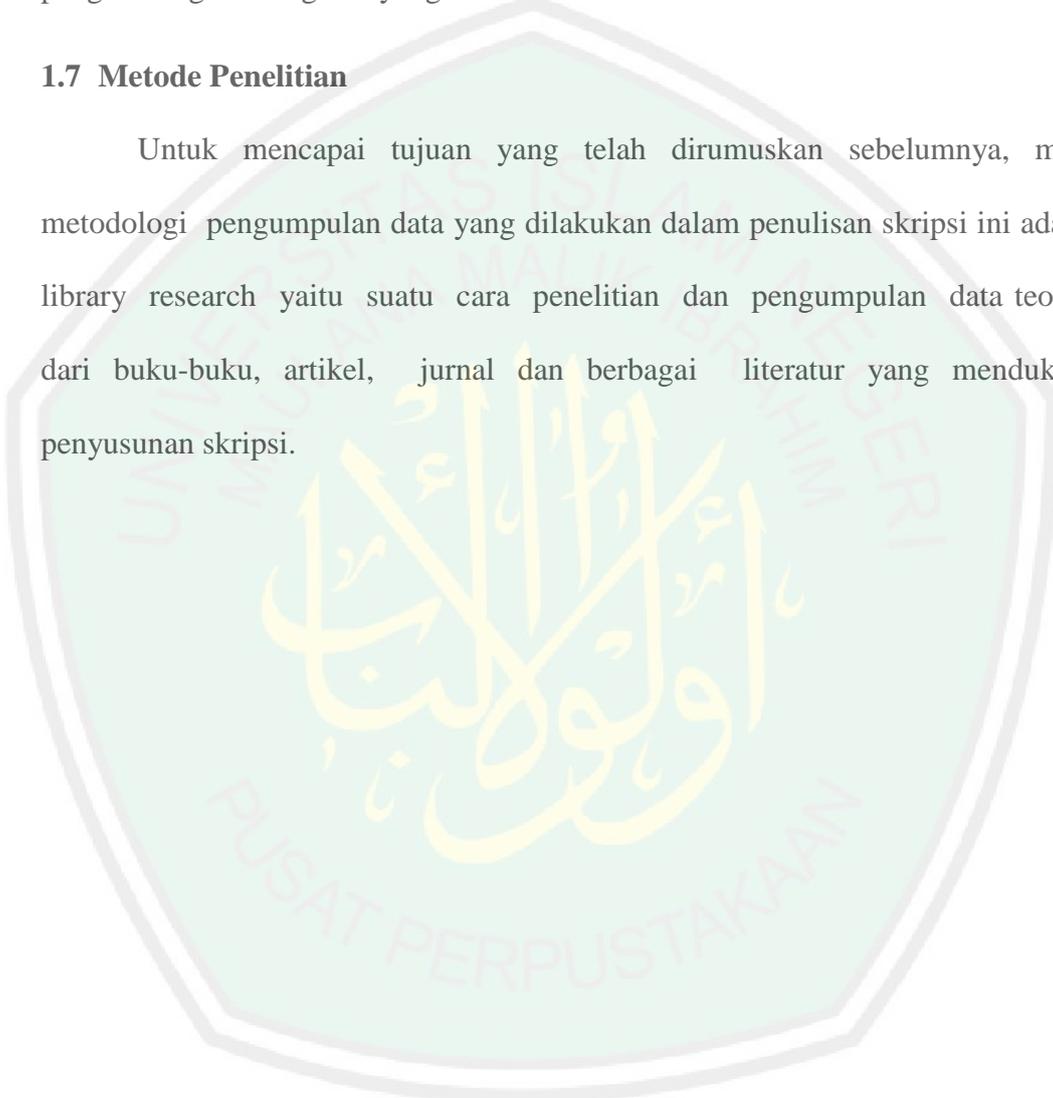
Bab ini menjelaskan tentang hasil uji aplikasi *game* yang dibangun. Perancangan dan desain diimplementasikan dengan bahasa pemrograman *Lite-C* dan juga *engine 3D Game Studio*. Hasil uji digunakan untuk mengetahui seberapa permasalahan yang terdapat *game* dapat diselesaikan, serta mengetahui solusi yang didapatkan dari permasalahan tersebut.

## **BAB V Penutup**

Berisi tentang kesimpulan dan saran yang dapat digunakan untuk pengembangan dari game yang telah dibuat.

### **1.7 Metode Penelitian**

Untuk mencapai tujuan yang telah dirumuskan sebelumnya, maka metodologi pengumpulan data yang dilakukan dalam penulisan skripsi ini adalah library research yaitu suatu cara penelitian dan pengumpulan data teoritis dari buku-buku, artikel, jurnal dan berbagai literatur yang mendukung penyusunan skripsi.



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Permainan (*game*) dan Karakteristik

Permainan (*game*) dapat diklasifikasikan menjadi dua bagian besar yaitu *game* fisik dan *game* elektronik. *Game* fisik sering dijumpai ketika masih anak-anak, seperti lompat tali, petak umpet, dan sebagainya. Permainan ini berhubungan dengan gerak fisik. Dan *game* elektronik merupakan permainan yang hampir semua kalangan menyukainya. *Game* elektronik berkembang sangat pesat hingga berbagai macam *game* elektronik baru banyak bermunculan.

*Game* elektronik atau yang sekarang disebut dengan *video game* pertama kali ditemukan oleh *Thomas T. Goldsmith Jr.* Dan *Estle Ray Mann* pada Januari tahun 1947. Perkembangannya semakin terlihat ketika mereka menemukan *Chatode-Ray Tube* sebuah tabung *vacum* yang digunakan sebagai media untuk membuat simulasi keceptaan tembakan dan arah tembakan sebuah roket. Dan penemuan baru terus berkembang hingga tahun 1959 (Sibero, 2009). Berikut adalah perkembangan *game* setelah masa ini berdasarkan generasi.

- a) Generasi Pertama (1972-1977)
- b) Generasi Kedua (1976-1983)
- c) Generasi Ketiga (1983-1992)
- d) Generasi Keempat (1987-1996)
- e) Generasi Kelima (1993-2002)
- f) Generasi Keenam (1998-2006)

g) Generasi Ketujuh (2004)

### 2.1.1 Elemen Game

Didalam sebuah game terdapat beberapa elemen-elemen yang akan membentuk sebuah game itu sendiri yaitu :

1) Rules (aturan-aturan)

Sebuah game mengambil tempat atau setting di dunia buatan yang diatur oleh aturan-aturan (rules). Rules istilah yang menentukan aksi dan gerakan dalam sebuah game. Pada game komputer, kebanyakan rules ini tersembunyi. Karena pemain berinteraksi dengan game hanya melalui suatu alat masukan atau yang biasanya disebut dengan input device dan mesin mengabaikan input yang tidak sesuai tanpa harus memberitahu aturan kepada pemain.

2) Victory condition (kondisi menang) dan lose condition (kondisi kalah)

Dalam sebuah game ada kondisi menang dan ada juga kondisi kekalahan.

3) Setting (setting)

Sebuah game mengambil tempat pada suatu dunia atau setting tertentu. Contohnya pada permainan bola yang bersetting pada lapangan dan batasan-batasan tertentu.

4) Interaction Model (model interaksi)

Yaitu cara pemain berinteraksi dengan game dan melakukan aksi untuk menghadapi tantangan dari game tersebut. Model interaksi yang biasa dipakai pada game komputer ada dua jenis. Jenis yang pertama, jika pemain mengendalikan tokoh tunggal yang mempresentasikan dirinya dalam game dan tokoh tersebut dapat mempengaruhi sekitarnya, maka tokoh tersebut

disebut sebagai avatar-nya. Sedangkan jenis yang kedua adalah jika pemain memiliki kemampuan untuk melihat berbagai bagian dari gameworld (dunia permainan) dan melakukan aksi pada banyak tempat maka pemain tersebut sebagai omnipresent. Namun hal ini dapat diterapkan pada tokoh atau unit yang menjadi miliknya, dengan memberikan perintah kepada mereka satu-persatu. Salah satu contoh game yang termasuk jenis ini adalah permainan catur.

5) Perspective (sudut pandang)

Yaitu menjelaskan bagaimana pemain melihat gameworld dari suatu game pada layar.

6) Role (peran)

Yaitu tokoh yang dimainkan oleh pemain dalam suatu game. Dengan adanya peran ini maka pemain akan lebih mudah untuk memahami apa yang sebenarnya ingin dicapai dan aturan apa yang dimainkan. Sebagai contoh, pada permainan monopoli kita berperan sebagai pialang perumahan. Pada game Football Manager kita berperan sebagai manager sebuah tim sepak bola.

7) Mode

Beberapa game, seperti catur, berlaku sama dari awal sampai akhir. Pemain selalu ingin mencapai atau menyelesaikan hal yang sama dengan cara yang berbeda. Namun ada juga game yang memiliki mode yang nyata, yaitu dimana gameplay-nya berubah dari satu mode ke mode lainnya. Contohnya pada game perang, yaitu sebelum pemain turun ke medan perang biasanya

pemain diberi briefing terlebih dahulu, lalu pindah ke mode pemilihan senjata, dan terakhir adalah perang itu sendiri.

8) Structure (struktur)

Hubungan antara mode dan aturan menentukan kapan dan mengapa game berubah secara bersama-sama membentuk suatu permainan.

9) Realism (realisme)

Game menggambarkan sebuah dunia, bahkan mungkin dunia fantasi. Sebuah game yang menerapkan akal sehat dan logika pada aturan permainannya dapat dikatakan sebagai game yang realistis. Contoh dari game ini adalah Microsoft Flight Simulator, game ini mencoba mengimplementasikan bagaimana mensimulasikan pesawat terbang yang sesungguhnya.

10) Story (cerita)

Game komputer merupakan perpaduan antara media pasif, pasif seperti televisi dan film dan media aktif, non pasif seperti permainan poker dan domino. Beberapa komputer, seperti tetris tidaklah mempunyai cerita. Lain halnya dengan seri Metal Gear Solid, dan Final Fantasy. Beberapa game memiliki alur cerita yang linier atau hanya satu jalan cerita dan bersifat non interaktif, namun ada juga yang bersifat interaktif dimana cerita akan berbeda atau bercabang tergantung pada pilihan atau tindakan yang diambil pemain. Sehingga cerita akhir (ending) yang didapat akan berbeda-beda pula, atau biasanya disebut dengan multiple ending.

### 2.1.2 Genre Game

Game berdasarkan jenisnya memiliki banyak macam game atau biasa disebut *genre game*, berikut beberapa *genre game* yang diketahui (Henry, 2005) yaitu:

#### 1. *Maze Game*

Jenis game ini adalah jenis game yang paling awal muncul. Pada game ini pemain hanya mengitari maze (lorong-lorong yang berhubungan) untuk mencapai tujuan tertentu. Dalam beberapa maze game, terkadang terdapat musuh yang selalu mengejar pemain. Sebagai contoh adalah game Pacman.

#### 2. *Board Game*

Jenis game ini tidak berbeda jauh dengan game tradisional. Sampai saat ini tidak ada variasi yang berbeda pada game ini, hanya tampilannya saja yang berbeda dengan game tradisional. Contohnya adalah game monopoli.

#### 3. *Card Game*

Hampir sama dengan board game, genre ini tidak memberikan perubahan berarti dari game versi tradisional yang sejenis. Contohnya, game Solitaire dan Hearts, versi asli dan versi elektroniknya nyaris tidak ada bedanya. Variasi yang ada adalah kemampuan multiplayer dan tampilan yang lebih bervariasi dari versi tradisional. Game ini termasuk game yang muncul pada awal game komputer seperti genre maze dan board game.

#### 4. *Battle Card Game*

Jenis game ini jarang masuk ke Indonesia. Contoh yang populer di luar negeri adalah battle card Pokemon. Ada beberapa stasiun televisi Indonesia

yang menayangkan kartun tentang battle card ini. Versi game elektroniknya sangat digemati di luar negeri. Kita bisa membeli card untuk dikoleksi dan dipertarungkan dengan pemain lain. Sama seperti card game, genre ini tidak banyak variasinya dibanding versi tradisionalnya.

#### 5. *Quiz Game*

Jenis ini juga agak jarang di Indonesia. Salah satu yang umum dikenal adalah game kuis *Who Wants to Be Billionaire*, sebuah game dengan nama yang sama dari acara kuis televisi. Game ini sederhana dalam cara bermain. Kita hanya perlu memilih jawaban yang benar dari beberapa pilihan jawaban. Biasanya pertanyaan yang diberikan memang memiliki topik tertentu. Tetapi ada yang topiknya beragam seperti di kuis *Who Wants to be Billionaire*.

#### 6. *Puzzle Game*

Jenis game ini memberikan tantangan kepada pemainnya dengan cara memecahkan masalah yang terdapat pada game yang dimainkan. Contoh dari jenis game ini adalah jigsaw.

#### 7. *Shoot Them Up*

Jenis ini banyak diminati karena mudah dimainkan. Biasanya musuh kita adalah berbentuk pesawat maupun jenis lain. Datang dari sebelah atas dengan jumlah yang banyak dan tugas kita adalah menembaki dan menghancurkannya secepat dan sebanyak mungkin. Pada awalnya bentuk game ini adalah game 2D.

#### 8. *Side Scroller Game*

Game jenis ini sangat terkenal dengan banyaknya game yang dikeluarkan dan berhasil di pasaran. Game ini didasarkan kepada 2D dimana pemain bergerak ke sepanjang alur permainan ke satu arah dan menyelesaikan tugasnya. Ada yang meloncat, berlari, mengendap, dan menghindari halangan seperti jurang dan proyektil baik dari musuh maupun yang lainnya. Contoh beberapa game yang terkenal, seperti Duke Nukem Asli, Commander Keen, Prince of Persia, Sonic the Hedgehog, dan sebagainya.

#### 9. Fighting Game

Sesuai dengan namanya, game ini mengutamakan pertarungan. Game ini memberikan kesempatan kepada pemain untuk mengkombinasikan berbagai gerakan dalam pertarungan. Contoh yang populer adalah game Street Fighter.

#### 10. Racing Game

Game ini memberikan permainan lomba kecepatan dari kendaraan yang dimainkan oleh pemain. Contoh game yang terkenal adalah Need For Speed.

#### 11. Flight Sim

Game ini berfokus kepada simulasi penerbangan. Simulasi yang diberikan meniru kondisi dari penerbangan sebenarnya, baik kondisi pesawat dan peralatannya maupun kondisi pemandangan. Contoh game yang terkenal dari jenis genre ini adalah Microsoft Flight Simulator, IL-2 Sturmovik, Apache Air Assault. Kita bisa belajar mengemudikan berbagai jenis pesawat, mulai dari jenis kecil sampai Boeing 747. Ada juga game yang memberikan simulasi pesawat jenis lain seperti helikopter (Apache Air assault). Konon teroris yang menyerang WTC belajar dengan game Microsoft Flight

Simulator karena presisi dan keakuratan game ini memang dinilai nyaris sempurna. Namun genre ini tidak hanya menyediakan simulasi zaman sekarang. Zaman futuristik pun sudah dipersiapkan. Seperti game Wing Comander dan X-Wing dimana kita bisa belajar mengendarai pesawat tempur antariksa.

#### *12. Turn Based Strategy Game*

Merupakan jenis game yang melibatkan strategi dan logika dari pemain untuk memenangkan permainan. Aturan dari game ini adalah setiap pemain melakukan gerakan secara bergantian. Contohnya adalah game Civilization.

#### *13. RTS (Real Time Strategy) Game*

Game yang melibatkan strategi dan logika dari pemain untuk memenangkan permainan. Hampir sama dengan turn based strategy, akan tetapi dalam game ini seorang pemain tidak perlu menunggu pemain lain untuk melakukan gerakan. Jadi pemain tercepatlah yang besar kemungkinannya untuk menang. Contoh game yang populer dari jenis ini adalah Warcraft.

#### *14. Sim*

Genre ini menyetengahkan permainan simulasi yang berbeda dengan Flight Simulator. Di sini kita sebagai pemain membangun secara simulasi sebuah kota, negara, atau koloni. Kita berperan sebagai “Dewa” di sini. Kita mengatur berbagai sumber daya dan menentukan berbagai keputusan yang kita inginkan dalam proses pembangunan yang sedang terjadi.

#### *15. First Person Shooter*

Mungkin hampir semua dari kita mengenal game Doom atau Counterstrike. Itulah contoh dari jenis FPS ini. Game ini mengutamakan kecepatan gerakan kita di dalam permainan. Banyak baku tembak dan kita harus bertahan selama mungkin. Disebut First Person Shooter karena pandangan pemain adalah pandangan orang pertama (first person). Kita melihat tampilan dilayar seperti kita melihat dari mata kita sendiri.

#### *16. First Person 3D Vehicle Based*

Ini sama dengan genre FPS di atas hanya bedanya pandangan kita bukan dari mata tetapi dari sudut pandang kendaraan atau mesin yang kita naiki. Kendaraan bisa berbentuk kapal, tank, atau robot raksasa. Di sini kecepatan bukan menjadi faktor utama. Kita harus berjuang dengan kemampuan kita sebaiknya mungkin untuk tidak dibunuh. Mirip dengan FPS memang tetapi lebih lambat karena kita berada di dalam sebuah kendaraan atau mesin

#### *17. Third Person 3D Games*

Sebenarnya genre ini sama dengan FPS kita melihat dari sudut pandang orang pertama (yaitu mata kita sendiri) maka pada genre ini kita melihat dari sudut pandang orang ketiga. Beberapa game menyediakan fitur kedua sudut pandang, tetapi umumnya game yang dipasaran hanya menggunakan satu sudut pandang saja.

#### *18. Role Playing Game (RPG)*

Game bertipe ini ini lebih mengutamakan cerita dan biasanya pemain diajak masuk ke dalam cerita tersebut untuk menyelesaikan sebuah misi. Dalam game ini pemain akan berperan menjadi sebuah karakter dengan

berbagai atribut, seperti kesehatan, intelegensi, kekuatan dan keahlian. Salah satu game yang terkenal dengan RPG adalah Final Fantasy.

#### 19. *Adventure Game*

Game jenis ini adalah game dengan mengambil konsep petualangan. Pemain berjalan menuju ke suatu tempat dan disepanjang perjalanan pemain menemukan hal-hal baru untuk dieksplorasi. Sebagai contohnya adalah game Crash Bandicoot.

#### 20. *Educational and Edutainment*

Banyak pengamat game di Indonesia yang mengatakan hanya genre ini yang berhasil secara komersil (dalam artian tidak dibajak, seperti game keluaran luar). Genre ini sebenarnya lebih mengacu kepada isi dan tujuan game, bukan genre yang seandainya. Misalnya game Bobby Bola sebenarnya campuran dari genre arcade dan side scroller. Terkadang ada genre lain yang dicampurkan sehingga genre edutainment, bertujuan untuk memancing minat belajar anak sambil bermain.

#### 21. *Sport*

Game ini sama dengan *genre edutainment*. Genre ini hanya berdasarkan jenisnya saja, bukan berdasarkan teknologi atau spesifikasi teknis apapun. Selama game itu menyetengahkan *genre olahraga* maka disebut *genre Sport*. Tidak peduli apakah game itu menggunakan gaya *arcade* 2D atau 3D maupun lainnya.

## 2.2 Bahasa Inggris

Bahasa Inggris adalah sebuah bahasa yang berasal dari Inggris. Bahasa ini menjadi bahasa utama di daerah Britania Raya (Irlandia, Wales, serta Inggris), Amerika Serikat dan banyak negara yang menggunakannya (Hidayat, 2006). Bahasa ini berawal dari kombinasi antara bahasa lokal yang dipakai oleh orang-orang Norwegia, Denmark, dan Anglo-Saxon dari abad ke-6 sampai abad ke-10. Lalu pada tahun 1066 dengan ditaklukkan Inggris oleh *William the Conqueror*, sang penakluk dari Normandia, Perancis Utara, maka bahasa Inggris sangat intensif dipengaruhi oleh bahasa Latin dan bahasa Perancis. Dari seluruh kosakata bahasa Inggris modern, diperkirakan kurang lebih 50 persen berasal dari bahasa Perancis dan Latin.

Di dalam sebuah bahasa pasti terdapat struktur dalam pembuatan sebuah kalimat yang nantinya akan dituliskan atau dikomunikasikan oleh seseorang. Di bahasa Inggris lebih diketahui sebagai *grammar* atau tata bahasa. Di dunia saat ini yang menjadi pedoman dalam penggunaan *grammar* yaitu *grammar* dari Amerika dan Inggris. Kalimat dalam tata bahasa Inggris memiliki pola tertentu yang digunakan untuk membentuk sebuah kalimat. Jenis kata yang terdapat di dalam bahasa Inggris ada 24 macam (Nina, 2010), yaitu:

1. *Articles* (Kata Sandang)
2. *Auxiliary Verb* (Kata Kerja Bantu)
3. *Verbs* (Kata Kerja)
4. *Nouns* (Kata Benda)
5. *Adjectives* (Kata Sifat)

6. *Adverbs* (Kata Keterangan)
7. *Possesive Forms* (Bentuk Kepemilikan)
8. *It and There* (Bentuk It Dan There)
9. *Modal Auxiliaries* (Kata Kerja Bantu)
10. *Pronouns* (Kata Ganti)
11. *Preposition* (Kata Depan)
12. *Conjunctions* (Kata Sambung)
13. *Infinitive* (Bentuk Dasar Kata Kerja)
14. *Question Words Questions* (Kata Tanya)
15. *Participles* (Kata Kerja Partisipatif)
16. *Question Tag* (Pertanyaan Penegas)
17. *Gerunds* (Kata Kerja -Ing)
18. *Conditional Sentences* (Kalimat Pengandaian)
19. *Active And Pasive Voice* (Kalimat Aktif Dan Pasif)
20. *Agreement/Concord* (Pesesuaian)
21. *Phrases And Clauses* (Frase Dan Klausa)
22. *Direct And Indirect Speech* (Kalimat Langsung Dan Tak Langsung)
23. *Iregular Verb* (Kata Kerja Tak Berurutan)
24. *Tenses* (Bentuk-Bentuk Waktu)

Sistem kala atau waktu dalam bahasa Inggris dinamakan *tense*. *Tense* terdiri dari tiga wilayah *tense* yaitu: *Past Tense* (waktu lampau), *Present Tense* (waktu sekarang), serta *Future Tense* (waktu akan datang). Dalam setiap *tense* dikembangkan menjadi empat yaitu *Simple*, *Continuous*, *Perfect*, *Perfect*

*Continuous*, terkecuali Future Tense yang memiliki delapan macam yang telah dikembangkan. (Winarno, 2006)

### 2.3 Grammar (Tata Bahasa)

*Grammar* di dalam kamus bahasa Inggris memiliki arti tata bahasa. Secara definisi umum *grammar* adalah aturan-aturan pembentukan suatu kalimat dalam bahasa, atau biasa juga disebut tata bahasa. Banyak para ahli *grammar* memiliki bermacam-macam definisi tentang *grammar* itu sendiri. Seperti di dalam buku (Coghill & Magendanz, 2003) yang mendefinisikan *grammar* sebagai berikut:

***“The grammar of a language is the set of rules that govern its structure. Grammar determines how words are arranged to form meaningful units.”*** (Coghill & Magendanz, 2003)

Maksud dari definisi tersebut yaitu “*Grammar* sebuah bahasa adalah satu kumpulan aturan yang menata bagian susunannya. *Grammar* menentukan bagaimana kata-kata disusun dalam membentuk unit-unit bahasa yang bermakna.”

Sama halnya dengan definisi diatas, (Swan, 2005) ahli bahasa yang lebih cenderung memperhatikan Bahasa Inggris asli Inggris (*British English*) mendefinisikan *grammar* seperti dibawah ini:

***“The rules that show how words are combined, arranged or changed to show certain kinds of meaning.”*** (Swan, 2005)

Arti dari definisi tersebut yaitu “*Grammar* adalah aturan yang menerangkan bagaimana kata digabungkan, disusun atau diubah untuk menunjukkan beberapa jenis makna.”

Terdapat pula definisi menurut pakar *grammar* ternama (Greenbaum, 1996) mengartikan *grammar* sebagai berikut:

***“In the concrete sense of the word grammar, a grammar is a book of one or more volumes. We of course also use grammar for the contents of the book. When we compare grammars for their coverage and accuracy, we are referring***

***to the contents of the book: a grammar is a book on grammar, just as a history is a book on history.***” (Greenbaum, 1996)

“Menurut makna konkrit kata *Grammar*, *grammar* adalah sebuah buku yang berisi satu volume atau lebih. Kita juga tentu mengartikan *grammar* sebagai isi sebuah buku. Ketika kita membandingkan *grammar* dengan bahasan dan kebenarannya, kita tentu mengacu pada isi dari sebuah buku: *grammar* adalah sebuah buku tentang *grammar*, seperti halnya sejarah adalah sebuah buku tentang sejarah.”

Berdasarkan definisi yang telah di utarakan oleh para pakar *grammar* diatas, tidak mudah untuk memungkirinya bahwa definisi dari *grammar* tersebut memiliki banyak sekali pendapat, mulai dari kalangan para ahli *grammar* dan juga pandangan orang lain terhadap *grammar* tersebut.

Dengan adanya *grammar*, proses *parsing* dapat dilakukan secara cepat dengan cara mencocokkan aturan-aturan yang tepat untuk membentuk struktur suatu kalimat (Suciadi, 2001).

Salah satu contoh *grammar* sederhana untuk bahasa Inggris ialah sebagai berikut:

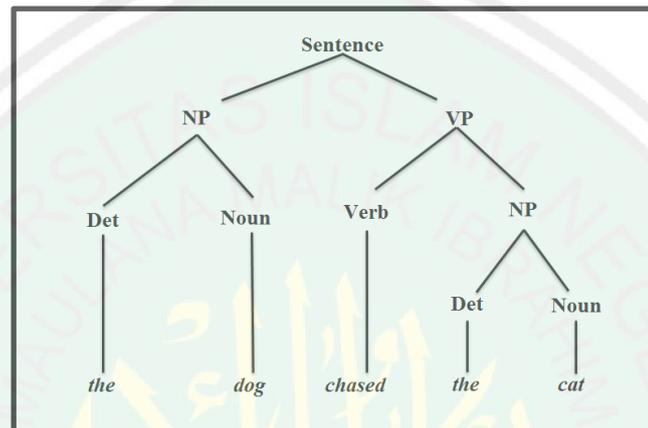
- a) *Sentence = noun phrase + verb phrase*
- b) *Noun phrase = determiner + noun*
- c) *Verb phrase = verb + noun phrase*
- d) *Determiner = the, a*
- e) *Noun = dog, dogs, cat*
- f) *Verb = chased, see, sees*

Dari *grammar* diatas dapat dibentuk beberapa kalimat bahasa Inggris yang sederhana, yaitu:

- a. *The dog chased a cat*

b. *The dog sees a cat*

Proses pembentukan kalimat “*The dog chased a cat*” dari grammar tersebut dapat dilihat pada gambar 2.1.



**Gambar 2.1** Parse Tree  
Sumber: (Suciadi, 2001)

#### 2.4 AI (*Artificial Intelligence*)

Definisi *Artificial Intelligence* (AI) menurut H. A. Simon [1987] adalah “Kecerdasan buatan (*artificial intelligence*) merupakan kawasan penelitian, aplikasi dan instruksi yang terkait dengan pemrograman komputer untuk melakukan sesuatu hal yang -dalam pandangan manusia adalah-cerdas”. Menurut *Encyclopedia Britannica*: “Kecerdasan Buatan (AI) merupakan cabang dari ilmu komputer yang dalam representasi pengetahuan lebih banyak menggunakan bentuk simbol-simbol daripada bilangan, dan memproses informasi berdasarkan metode heuristic atau dengan berdasarkan sejumlah aturan”. AI dapat dipandang dalam berbagai perspektif, antara lain:

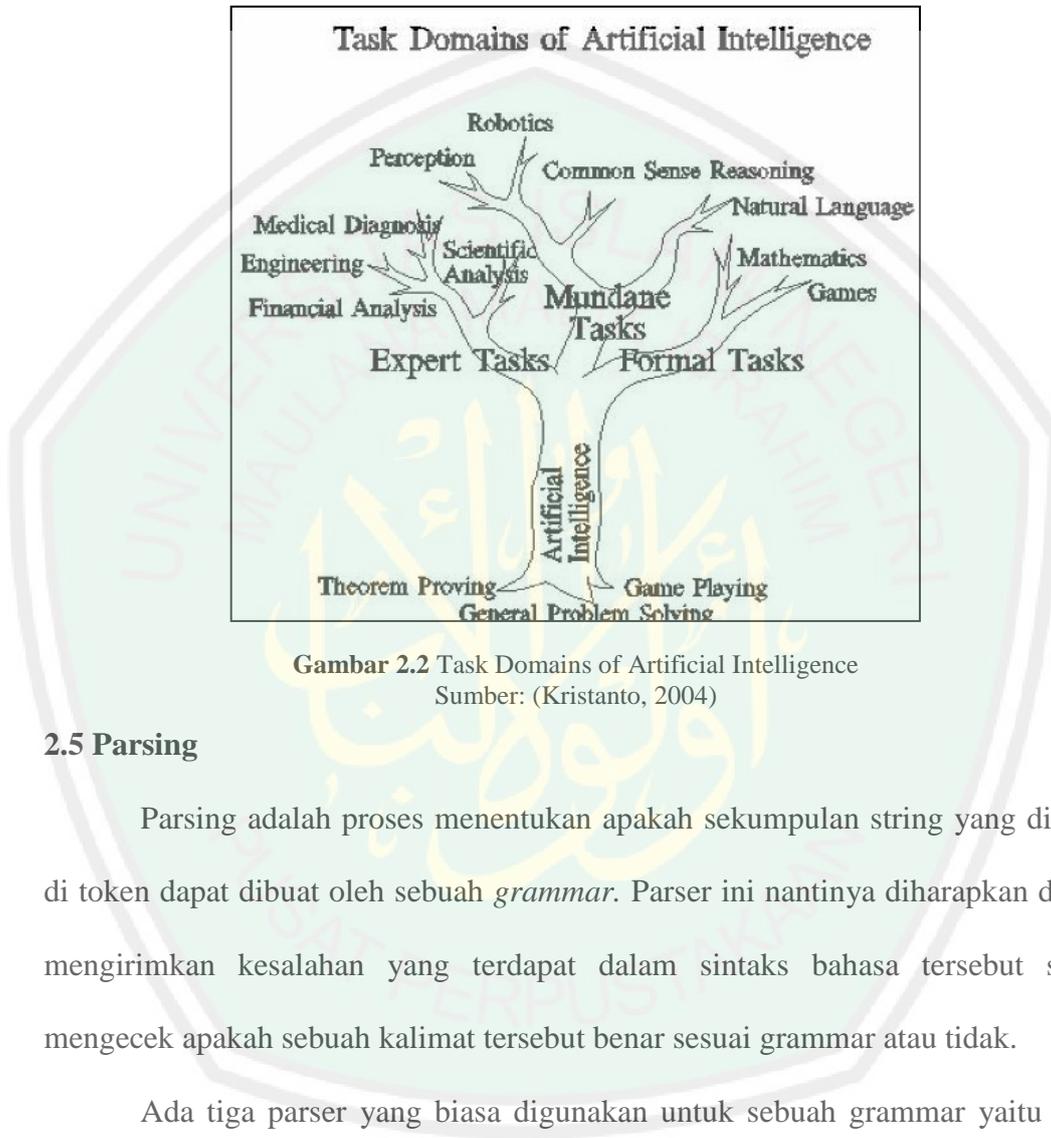
- a. Dari perspektif Kecerdasan (*Intelligence*) AI adalah bagaimana membuat mesin yang “cerdas” dan dapat melakukan hal-hal yang sebelumnya dapat dilakukan oleh manusia
- b. Dari perspektif bisnis, AI adalah sekelompok alat bantu (*tools*) yang berdaya guna, dan metodologi yang menggunakan tool-tool tersebut guna menyelesaikan masalah-masalah bisnis.
- c. Dari perspektif pemrograman (*Programming*), AI termasuk didalamnya adalah studi tentang pemrograman simbolik, pemecahan masalah, proses pencarian (*search*).

Dari perspektif penelitian (*research*), Riset tentang AI dimulai pada awal tahun 1960-an, percobaan pertama adalah membuat program permainan (*game*) catur, membuktikan teori, dan *general problem solving* (untuk tugas-tugas sederhana).

#### 2.4.1 Sejarah AI (*Artificial Intelligence*)

Tahun 1950-an Alan Turing, seorang pionir AI dan ahli matematika Inggris melakukan percobaan Turing (*Turing Test*) yaitu sebuah komputer melalui terminalnya ditempatkan pada jarak jauh. Di ujung yang satu ada terminal dengan software AI dan diujung lain ada sebuah terminal dengan seorang operator. Operator itu tidak mengetahui kalau di ujung terminal lain dipasang software AI. Mereka berkomunikasi dimana terminal di ujung memberikan respon terhadap serangkaian pertanyaan yang diajukan oleh operator. Dan sang operator itu mengira bahwa ia sedang berkomunikasi dengan operator lainnya yang berada pada terminal lain. Turing beranggapan bahwa jika mesin dapat membuat

seseorang percaya bahwa dirinya mampu berkomunikasi dengan orang lain, maka dapat dikatakan bahwa mesin tersebut cerdas (seperti layaknya manusia).



Gambar 2.2 Task Domains of Artificial Intelligence  
Sumber: (Kristanto, 2004)

## 2.5 Parsing

Parsing adalah proses menentukan apakah sekumpulan string yang dibuat di token dapat dibuat oleh sebuah *grammar*. Parser ini nantinya diharapkan dapat mengirimkan kesalahan yang terdapat dalam sintaks bahasa tersebut serta mengecek apakah sebuah kalimat tersebut benar sesuai *grammar* atau tidak.

Ada tiga parser yang biasa digunakan untuk sebuah *grammar* yaitu Top Down Parsing, Bottom Up Parsing, serta gabungan dari keduanya Left Corner Parsing. Left Corner Parsing pertama kali dikemukakan oleh Rosenkrantz dan Lewis II pada tahun 1970. LC (*Left Corner*) parsing merupakan strategi yang menggunakan data secara bottom up parsing dan prediksi dari top down parsing.

## 2.6 Left Corner Parsing

*Left Corner Parsing* adalah gabungan dari dua buah metode yaitu *Top-Down* dan *Bottom-Up Parsing*. *Top-Down Parsing* melakukan penelusuran dari *root/puncak* menuju ke *leaf/daun* (simbol awal sampai simbol terminal). *Top-Down* bekerja dengan cara menguraikan sebuah kalimat mulai dari *constituent* yang terbesar sampai menjadi *constituent* yang terkecil. *Constituent* sendiri yaitu unsur-unsur pembentuk kalimat yang dapat berdiri sendiri seperti *noun*, dan *verb*. Hal ini dilakukan terus-menerus sampai semua komponen yang dihasilkan ialah *constituent* yang terkecil dalam kalimat, yaitu kata.

Sedangkan *Bottom-Up Parsing* bekerja dengan cara mengambil satu demi satu kata dari kalimat yang diberikan, untuk dirangkaikan menjadi *constituent* yang lebih besar. Hal ini dilakukan terus-menerus sampai *constituent* yang terbentuk ialah *sentence* atau kalimat. Dengan demikian metode *Bottom-Up Parsing* ini bekerja dengan cara yang terbalik dari cara kerja *Top-Down Parsing*.

Cara kerja *Left Corner Parsing* ialah dengan mula-mula menerima sebuah kata, menentukan jenis *constituent* apa yang dimulai dengan jenis kata tersebut, kemudian melakukan proses parsing terhadap sisa-sisa *constituent* tersebut dengan cara *Top Down*. Dengan demikian proses *Left-Corner Parsing* dimulai secara *Bottom Up* dan diakhiri secara *Top Down* (Suciadi, 2001)

Berikut adalah proses cara kerja dari *Left Corner Parsing* dalam gambar 2.3:



**Gambar 2.3** Left Corner Parsing  
Sumber: (Suciadi, 2001)

Proses yang terdapat pada gambar di atas yang pertama dilakukan yaitu melakukan pengecekan dengan cara Bottom-Up yaitu diawali dengan memeriksa kata pertama “the”, kemudian memeriksa apakah *constituent* apa yang dimiliki oleh kata tersebut. Kata “the” di atas memiliki jenis *constituent* “Det” atau “determine”. Setelah itu *constituent* terbesar selanjutnya yang dimiliki oleh “Det”. Setelah diketahui *constituent* terbesarnya langkah selanjutnya yaitu dengan melakukan proses Top-Down, yaitu menelusuri *constituent* terbesar dan mencocokkannya dengan *constituent* yang lebih kecil. Apabila terjadi kecocokan, maka dilanjutkan dengan pencocokan kata selanjutnya yaitu “dog”. Begitupun dengan kata-kata yang lain dalam memeriksa apakah terdapat kecocokan antar *constituent*.

## 2.7 3D Game Studio

Gamestudio, sering dikenal sebagai 3D Game Studio atau 3DGS adalah software yang digunakan untuk pengembangan *game* yang memungkinkan

pengguna untuk membuat *game* 3D dan aplikasi lain yang bersifat virtual-reality (Henry, 2005). Di dalamnya terdapat Model Editor (MED), World Editor (WED), Script Editor (SED) / debugger dan dilengkapi dengan koleksi tekstur, model dan karya seni, serta sistem permainan template yang memungkinkan penciptaan dasar permainan penembak atau RPG.

Gamestudio dipasarkan pada pengguna berbagai tingkat keahlian, menyediakan tiga tingkat penggunaan yang berbeda ("pemula", "maju", dan "profesional") untuk penggemar, seniman, serta programmer. Selain itu, Conitec juga pasar sistem untuk penggunaan seperti advergame pengembangan dan penciptaan presentasi real-time, simulasi, dan pameran virtual.

Perkembangan 3D Game Studio:

- 1) 1993 ACK 3D (Animation Construction Kit) by Lary Myers (Wolfenstein-like open source engine).
- 2) 1994 ACK NEXT GENERATION by Johann Christian Lotter / oP Group (improved version of ACK 3D, open source).
- 3) 1995 ACKNEX-2 written for the German TV show X-BASE (engine comparable to Doom).
- 4) 1995 ACKNEX-2 becomes Conitec property and is released as '3D Game Studio'.
- 5) 1997 ACKNEX-3 released.
- 6) 1999 A4 released (Windows Based Quake-like Engine).
- 7) 2000 A5 released (Terrain Engine).
- 8) 2003 A6 released (Physics and Shaders).

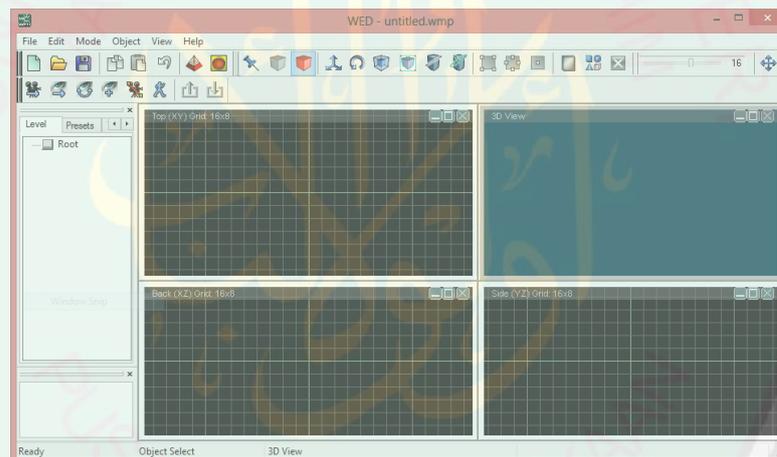
9) 2007 A7 released (new ABT renderer and Lite-C).

10) 2010 A8 released (Enet Newtork Library, Nvidia PhysX, PSSM).

## 2.7.1 Editor

### 2.7.1.1 World Editor (WED)

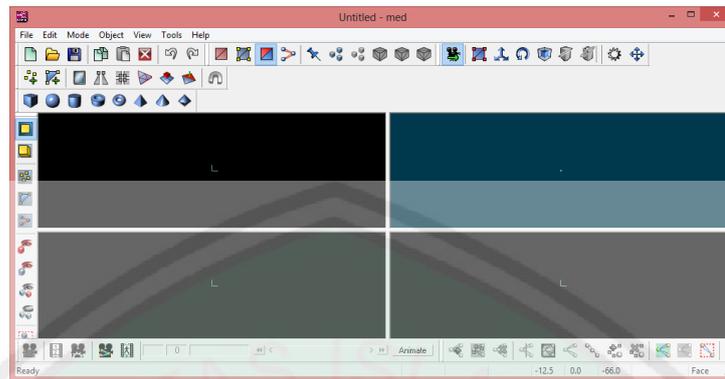
World Editor atau WED merupakan editor utama dari Game Studio. Pada editor ini beberapa entity diatur tata letaknya dan beberapa action dapat terapkan langsung pada editor ini. Interface WED juga memiliki tampilan dengan aplikasi 3D lainnya sehingga mudah untuk menggunakannya.



Gambar 2.4 Tampilan World Editor  
Sumber: (Sibero, 2009)

### 2.7.1.2 Model Editor (MED)

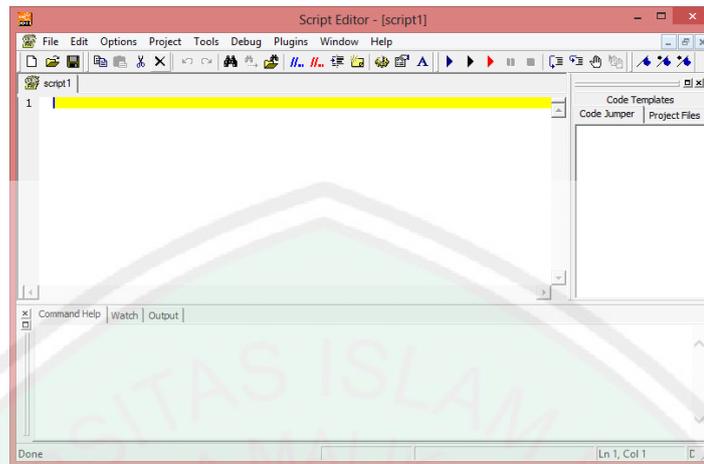
*Model Editor* merupakan aplikasi modelling yang disediakan ketika software ini di-install. *Model editor* ini digunakan untuk mebbuat model 3D yang akan diletakkan pada *World Editor*. *Model Editor* menyediakan *tool* yang cukup lengkap, seperti animasi karakter 3D dengan *bones*, *UV mapping* dan lain-lain.



**Gambar 2.5** Tampilan Model Editor  
Sumber: (Sibero, 2009)

### 2.7.1.3 Script Editor (SED)

Script Editor atau SED merupakan editor khusus yang dibuat untuk keperluan pemrograman untuk game yang lebih kompleks, editor ini digunakan untuk membuat beberapa perintah variabel dan perintah dalam editor ini menggunakan bahasa seperti C. Namun beberapa development dapat digunakan seperti Visual C++ dan Delphi. Editor ini juga dapat melakukan compile dan debug.



**Gambar 2.6** Tampilan Script Editor  
Sumber: (Sibero, 2009)

## 2.8 Pendidikan Dini Usia Sekolah Dasar

Pendidikan bertujuan untuk membentuk bangsa yang cerdas dan bermutu. Pendidikan orang tua terhadap anak akan sangat berpengaruh terhadap kreativitas anak. Menurut (Sumarno, 2011), anak yang memiliki bakat tertentu jika tidak diberikan rangsangan-rangsangan atau motivasi dari orang tua dan lingkungannya, tidak akan mampu memelihara, apalagi mengembangkan bakatnya. Hal ini dapat diminimalkan dengan mengoptimalkan pendidikan anak sejak dini.

Menurut Prof. Dr. Utami Munandar, seorang pakar kreativitas Indonesia yang telah dikutip oleh (Effendi, 2013):

“Kapasitas otak anak pada usia 6 bulan sudah mencapai sekitar 50% dari keseluruhan potensi orang dewasa. Otak seorang anak ternyata sangat luar biasa. Pada masa ini, anak mengalami perkembangan intelektual otak yang sangat cepat. Tingkat perkembangan intelektual otak anak dari sejak lahir sampai usia 4 tahun mencapai 50%. Oleh karena itu, pada masa empat tahun pertama ini sering disebut juga sebagai Golden Age (Masa Keemasan), karena si anak mampu menyerap dengan cepat setiap rangsangan yang masuk. Anak akan mampu menghafal banyak sekali informasi, seperti perbendaharaan kata, nada, dan sebagainya. Hingga usia 8 tahun, anak telah memiliki intelektual otak sekitar 80%” (Sumarno, 2011).

Melihat anak sekolah dasar yang lebih senang bermain dan masa keemasan anak terjadi pada usia muda, serta kapasitas intelektual otak anak yang mencapai 80% pada usia 8 tahun. Hendaknya diangkat menjadi sebuah *game* edukasi yang ditujukan untuk anak usia sekolah dasar, mengingat *game* yang ada saat ini banyak memuat konten yang tidak jelas tujuannya dan cenderung kepada kekerasan (Sujiono, 2009).



## BAB III

### RANCANGAN DESAIN SISTEM

#### 3.1 Perancangan Sistem

*Game* yang dibangun adalah *game single player* yang berjenis *adventure game* atau *game* petualangan. Dalam *game* petualangan ini terdapat beberapa karakter yang digunakan. Satu karakter sebagai karakter utama, dan yang lain sebagai NPC (*Non-Player Character*) yang akan membantu karakter utama dalam menyelesaikan tugas karakter utama. Objek penelitian dalam *game* ini terdapat pada pengecekan atau pemeriksaan jawaban dari kalimat yang dibuat oleh karakter utama.

##### 3.1.1 Keterangan Umum Game

Environment atau lingkungan yang digunakan dalam *game* ini yaitu lingkungan perumahan atau bangunan. Setiap bangunan tersebut ada beberapa kartu yang harus dicari oleh Imran sebagai karakter utama. Imran harus mencari kartu di setiap bangunan yang ada disekitar rumahnya. Setelah ditemukan kartu tersebut, Imran tidak semudah itu keluar dari bangunan itu. Imran akan dihadang oleh sebuah NPC yang menjaga setiap kartu yang diberi nama Pak Rizal. Imran harus menyetujui apa yang dikatakan oleh Pak Rizal untuk menyusun kata-kata yang muncul menjadi sebuah kalimat sederhana. Kemudian Imran dapat keluar dari bangunan tersebut.

Latar *game* pada level 1 terdapat 15 bangunan dan 8 kartu yang diletakkan di dalamnya, serta NPC yang menjaga setiap kartu. Masing-masing kartu terdapat 2 macam model susunan kalimat yang harus disusun. Setelah semua kartu telah

ditemukan dan menyelesaikan seluruh tugas menyusun maka langkah selanjutnya yaitu mencari sebuah gerbang atau portal untuk mengakhiri game pada hari pertama dan melanjutkan game pada hari selanjutnya.

Latar game pada level 2 sama dengan level 1 akan tetapi jumlah bangunan dan kartu berbeda. Dalam level 2 ini karakter utama akan mencari kartu kembali yang berjumlah 6 kartu yang terdapat pada 12 bangunan yang ada. Penelitian ini terdapat pada pengecekan atau pemeriksaan kalimat yang disusun oleh karakter utama setelah memperoleh kartu yang dicari. Kesulitan dalam mencari juga akan dibatasi oleh kesehatan dan waktu yang ditempuh. Semakin lama mencari kartu di dalam sebuah bangunan, semakin cepat juga kesehatan akan berkurang seiring waktu yang terus berjalan.

Latar game pada level 3 sama dengan latar game pada level 1 dan level 2, serta juga mempunyai jumlah bangunan yang sama dengan pada game level 2 dan jumlah kartu yang berbeda. Jumlah kartu pada level 3 terdapat 4 kartu saja. Dikarenakan semakin sedikitnya kartu maka waktu untuk pencarian semua kartu akan memerlukan waktu lebih untuk mencarinya. Setelah menemukan semua kartu segera mencari gerbang atau portal untuk melanjutkan menyelesaikan tugas pada level selanjutnya.

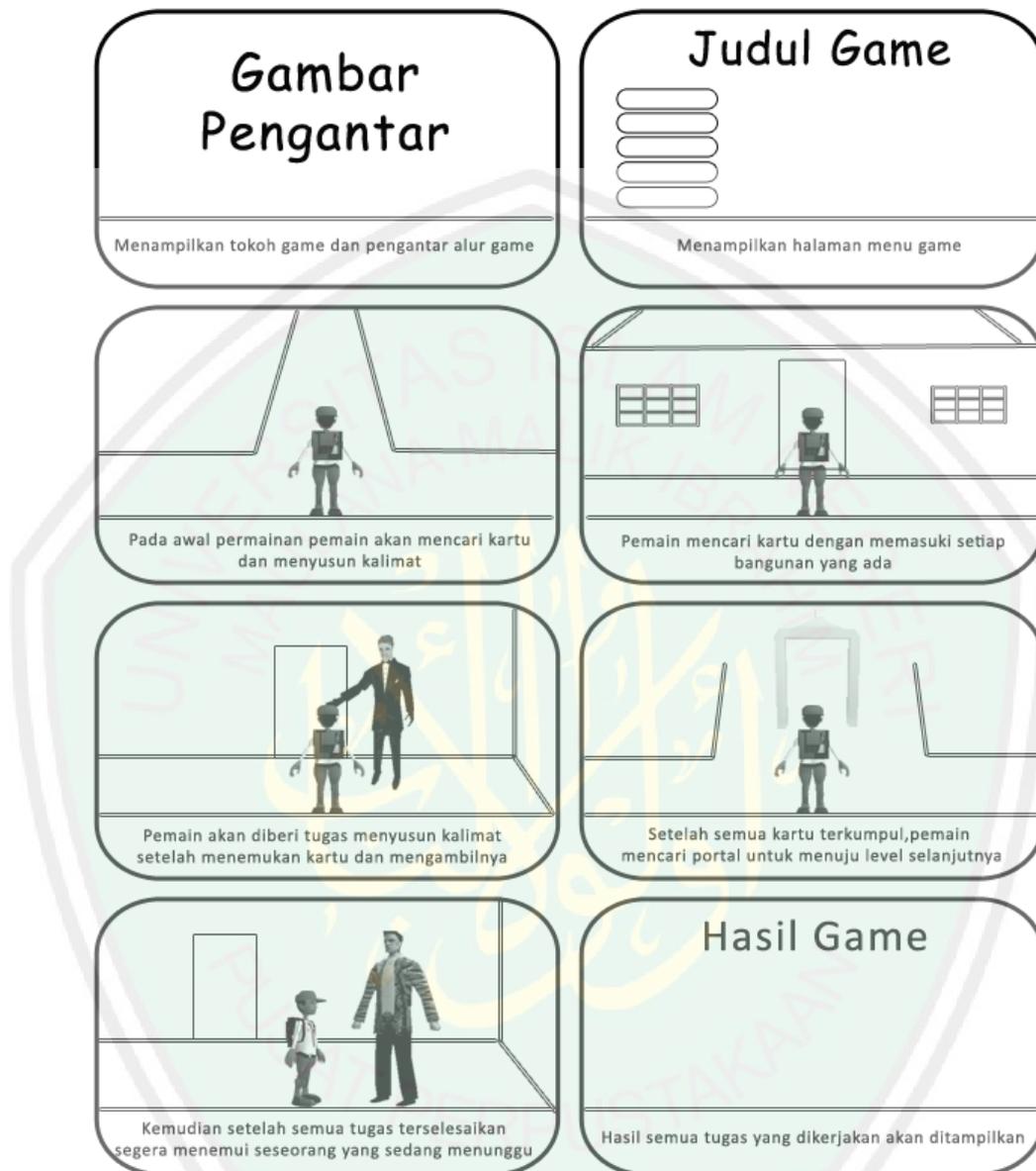
Dalam level 4 tugas pemain tetap mencari kartu yang terdapat di dalam bangunan yang ada. Jumlah bangunan yang ada yaitu 15 bangunan dan mencari 2 kartu yang harus ditemukan sebelum kesehatan yang ada berkurang. Setelah ditemukan kartu yang dicari segera menuju gerbang atau portal besar untuk

menuju sebuah latar baru, yaitu latar tempat NPC lain menunggu untuk menerima semua hasil dari semua tugas yang telah diselesaikan.

Unsur pembelajaran dalam game ini terdapat pada saat menyusun dan setelah menyusun. Saat menyusun kata-kata yang ada, pemain akan berlatih dan berfikir bagaimana untuk menyusunnya dengan kalimat yang benar, dan setelah tersusun semua maka pengetahuan tentang grammar pada setiap kata-kata yang disusun tersebut akan keluar.

### **3.1.2 Storyboard Game**

Dalam subbab ini menjelaskan gambaran rancangan game yang akan dibuat. Dimulai dengan sebuah pengantar yang berisi tentang tokoh dan alur dari cerita game ini, lalu sebuah judul serta menu yang digunakan dalam game. Setelah itu menceritakan tugas sang pemain dalam menjalankan game ini. Apa yang akan dihadapi oleh pemain ditampilkan di dalam storyboard game ini. Berikut ini adalah gambar dari desain storyboard game ini.



Gambar 3.1 Storyboard Game

### 3.1.3 Penampilan Umum Game

Aplikasi game yang dibangun pada umumnya berbentuk 3D, akan tetapi terdapat pula 2D. Seperti teks yang keluar ketika NPC berdialog, jumlah hari yang ada, jumlah skor nilai dan juga kesehatan yang digunakan, serta panel-panel yang digunakan untuk memunculkan susunan kata yang akan disusun oleh pemain.

Adapun yang berbentuk 3D yaitu pemain atau karakter utama, kartu, NPC penjaga kartu, NPC penunggu jawaban, pohon, pagar, pintu, rumah, lampu dan lainnya.

### 3.1.4 Deskripsi Karakter

#### a) Karakter Utama (Pemain)

Karakter utama atau pemain pada game ini mempunyai nama yaitu Imran. Imran adalah seorang siswa Sekolah Dasar Islam (SDI) Indonesia, yang mendapatkan kesempatan untuk melanjutkan belajar ke London di negara Inggris melalui beasiswa yang diadakan oleh pemerintah. Imran harus membuat beberapa kalimat yang sesuai dengan grammar British dalam waktu 4 hari. Dalam membuat kalimat tersebut, Imran dibantu oleh Pak Rizal yang akan membantu dalam hal membuat kalimat tersebut. Berikut adalah desain dari karakter utama.



Gambar 3.2 Karakter Utama (Imran)

#### b) Karakter Pak Rizal

Karakter Pak Rizal adalah sosok laki-laki yang memakai jas hitam berdasi kupu-kupu. Tugas Pak Rizal yaitu membantu Imran dalam membuat kalimat yang dibutuhkan oleh Imran dengan cara menyediakan kata-kata yang nantinya akan disusun oleh Imran. Karakter Pak Rizal termasuk NPC yang diberi animasi ketika karakter utama telah mengambil kartu dan mendekat ke NPC. Berikut adalah desain dari karakter Pak Rizal.



**Gambar 3.3** NPC Penolong (Pak Rizal)

### c) Karakter Pak Bagus

Karakter Pak Bagus disini hanya diam duduk disebuah ruangan untuk menunggu Imran menyerahkan semua tugas yang diselesaikannya dan selanjutnya memberikannya kepada pemerintah. Karakter Pak Bagus juga termasuk NPC yang dapat bereaksi. NPC tersebut akan beraksi ketika karakter Imran mendekatinya. Berikut adalah desain tampilan dari karakter Pak Bagus.



**Gambar 3.4** NPC Perantara (Pak Bagus)

## 3.2 Perancangan Aplikasi

Dalam subbab ini menjelaskan tentang semua gambaran perancangan dari aplikasi yang akan dibuat atau yang sering disebut dengan antarmuka aplikasi serta alur sistem dari sebuah aplikasi yang akan dibuat. Berikut adalah perancangan aplikasi game berupa gambaran antarmuka game dan alur flowchart beserta keterangannya.

### 3.2.1 Perancangan Antarmuka Pengantar

Pada saat pertama game dijalankan, akan tampil beberapa gambar pengantar berupa tokoh karakter dan alur cerita game. Berikut dibawah ini desain dari pengantar game.



Gambar 3.5 Pengantar Game

### 3.2.2 Perancangan Antarmuka Menu Game

Antarmuka menu game ini terdapat lima menu yang dapat digunakan yaitu *New Game*, *Setting*, *About*, *Help*, dan *Exit*. Menu *New Game* akan menampilkan permainan yang akan dimainkan oleh pemain. Dalam menu *Setting*, pemain dapat mengatur kecil besarnya suara dan efek yang ada di dalam game. Dalam menu *About* akan menampilkan tentang pembuat game ini. Sedangkan dalam menu *Help*, pemain dapat melihat petunjuk yang akan dijadikan bantuan dalam bermain game ini. Menu yang terakhir yaitu *Exit*, pemain akan diberi pilihan apakah memainkan keluar dari game atau melanjutkan permainan.



Gambar 3.6 Menu Game

### 3.2.3 Perancangan Antarmuka Game

Antarmuka game yang akan dimainkan menggunakan lingkungan rumah. Pada game ini terdapat 20 kartu yang tiap kartunya mempunyai 2 jenis susunan kalimat. Dua puluh kartu tersebut tersebar di dalam beberapa rumah yang ada di dalam game. Dua puluh kartu tersebut dibagi dalam 4 level.



Gambar 3.7 Awal Permainan Game

### 3.2.4 Perancangan Antarmuka Penutup

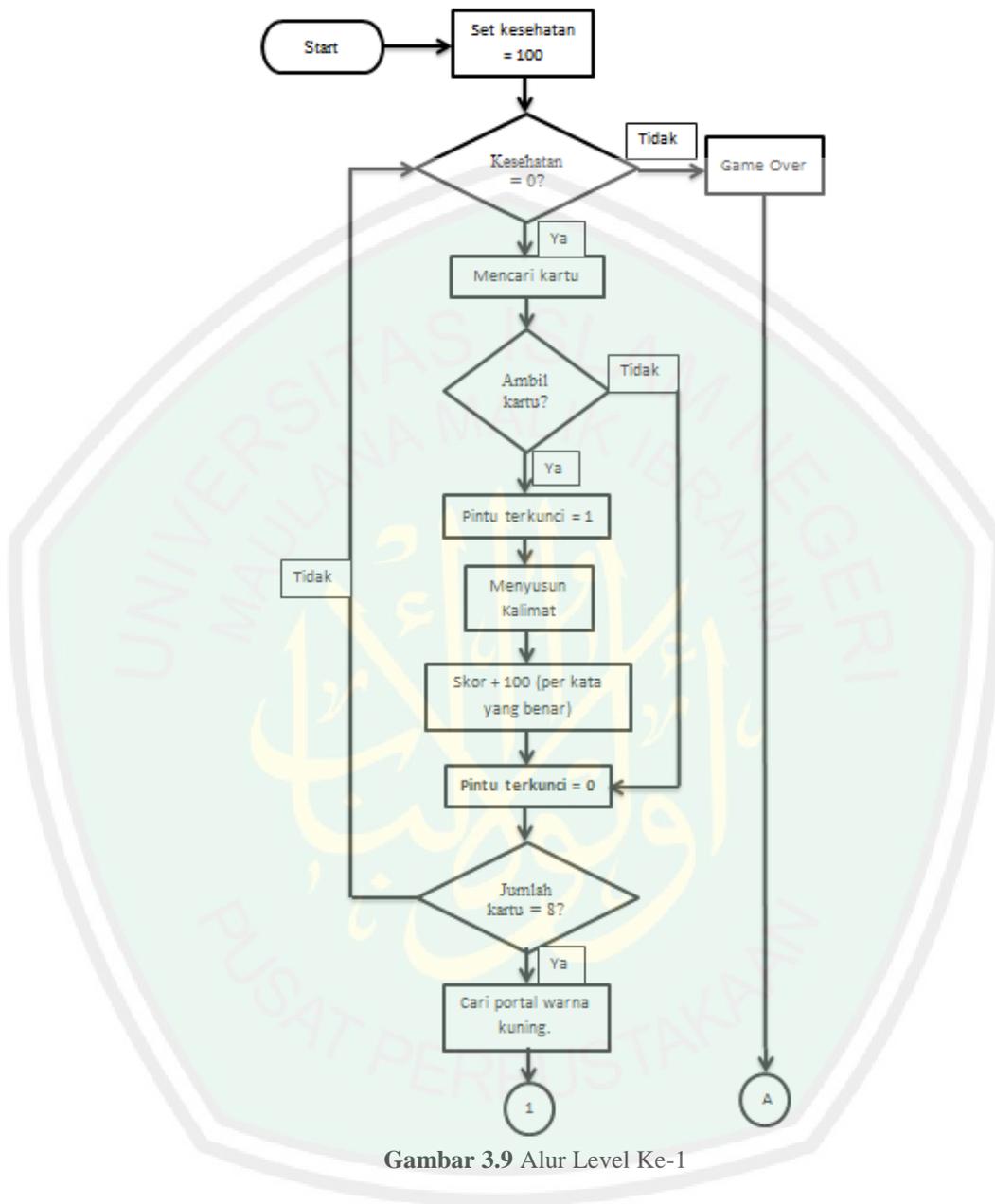
Antarmuka penutup ini melingkupi hasil dari kalimat yang sesuai dengan pola kalimat yang telah ditentukan.



Gambar 3.8 Hasil Akhir Game

### 3.2.5 Perancangan Alur Permainan Pada Level Pertama, Level Kedua, dan Level Ketiga

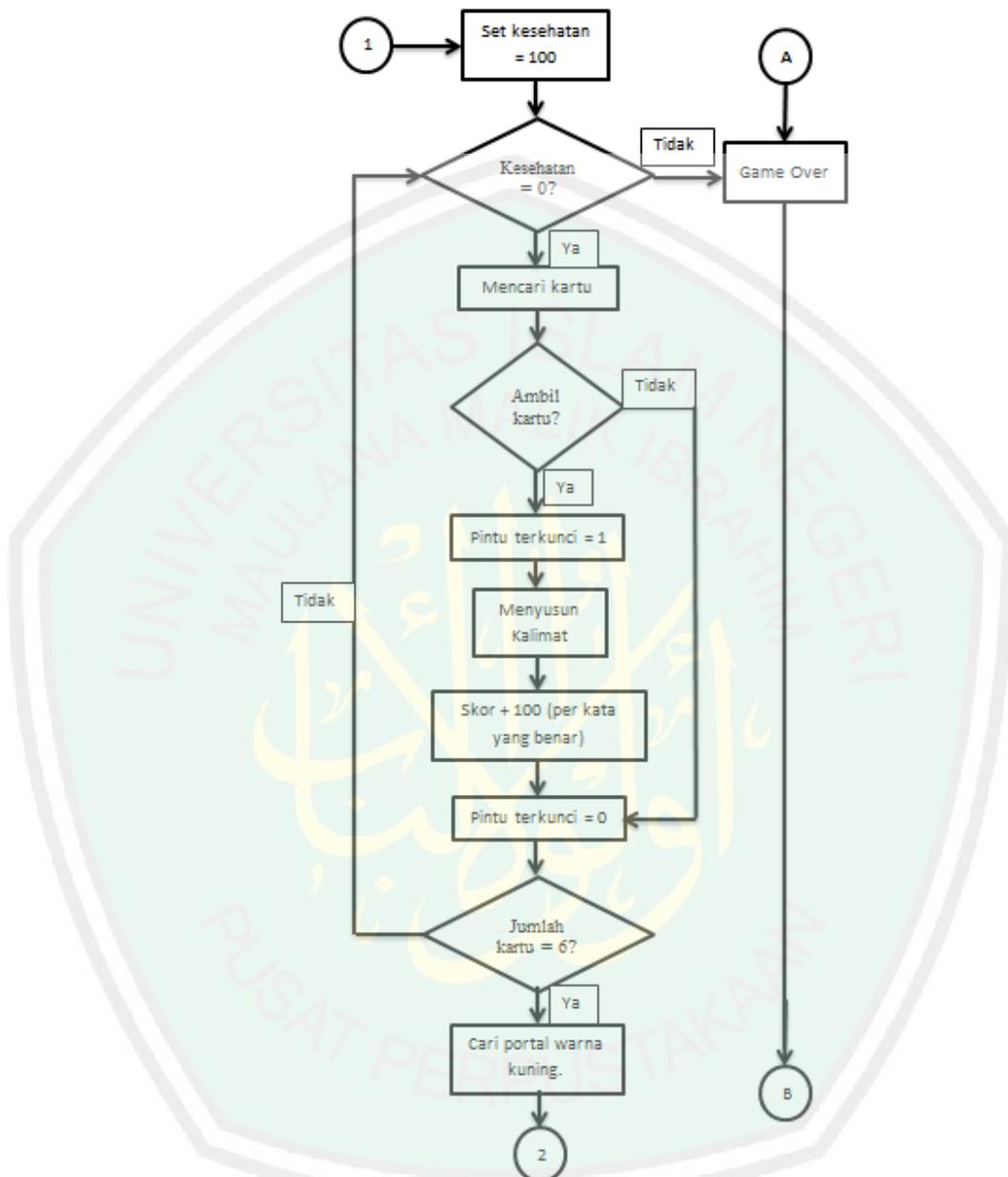
Alur permainan pada level pertama, kedua, ketiga, dan keempat hampir sama. Yang membedakan hanya jumlah kartu yang harus ditemukan pada tiap level. Berikut adalah alur dari level pertama, kedua, dan ketiga.



Dalam alur level ke-1 proses alur yang terjadi pada saat pertama bermain yaitu game akan memasang kesehatan awal sebesar 100% atau dalam keadaan penuh. Proses selanjutnya yaitu pemain akan mencari kartu yang berada di dalam rumah. Setelah menemukan kartu, apakah pemain mengambil kartu tersebut? Apabila pemain tidak mengambil kartu tersebut, pintu rumah akan bernilai = 0

yang artinya pintu akan terbuka, dan apabila kartu diambil oleh pemain maka pintu akan bernilai = 1 yang artinya pintu akan terkunci dan pemain tidak akan bisa keluar dari rumah, kecuali pemain melakukan tugas yang diberikan untuk menyusun kalimat. Selesai menyusun kalimat, hasil inputan akan diproses dan apabila kata-kata yang disusun benar semua, maka pemain akan mendapatkan skor dengan jumlah skor 100 untuk tiap kata yang benar. Jumlah susunan kata dalam setiap level mempunyai jumlah susunan kata yang berbeda. Setelah semua tugas selesai dilakukan, maka pintu akan bernilai = 0 kembali.

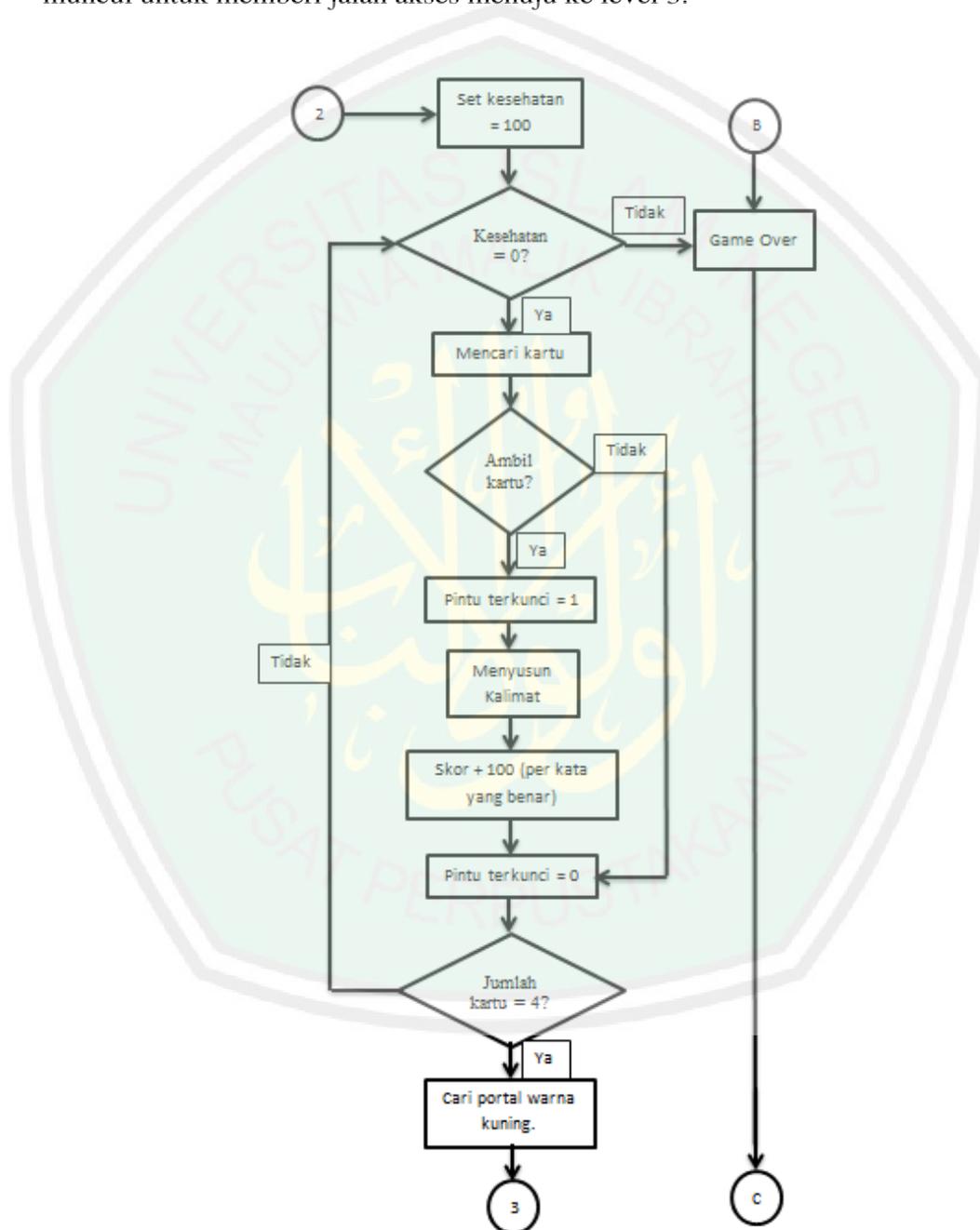
Keluar dari rumah tersebut, sistem akan membaca jumlah kartu yang telah diperoleh pemain. Pembacaan sistem terjadi ketika pemain telah melakukan tugas yang diberikan. Jumlah kartu yang harus dicari dalam game level 1 ini terdapat 8 kartu, sehingga apabila kartu yang diperoleh telah mencapai 8 buah kartu, maka akan muncul sejenis portal berwarna kuning yang mana portal tersebut adalah jalan menuju ke level selanjutnya.



Gambar 3.10 Alur Level Ke-2

Dalam level sebelumnya, pemain telah melewati portal berwarna kuning dan telah menyelesaikan semua tugas yang diberikan. Setelah melewati portal tersebut pemain akan dihadapi dengan level selanjutnya yaitu level 2. Proses yang terjadi di dalam level 2 sama dengan proses pada level 1. Perbedaan yang ada dengan level 1 yaitu jumlah kartu yang harus dicari untuk memenuhi tugas di

level 2. Dalam level 2 terdapat 6 kartu yang harus dicari di dalam rumah-rumah yang ada. Setelah semua kartu terpenuhi maka portal berwarna kuning akan muncul untuk memberi jalan akses menuju ke level 3.



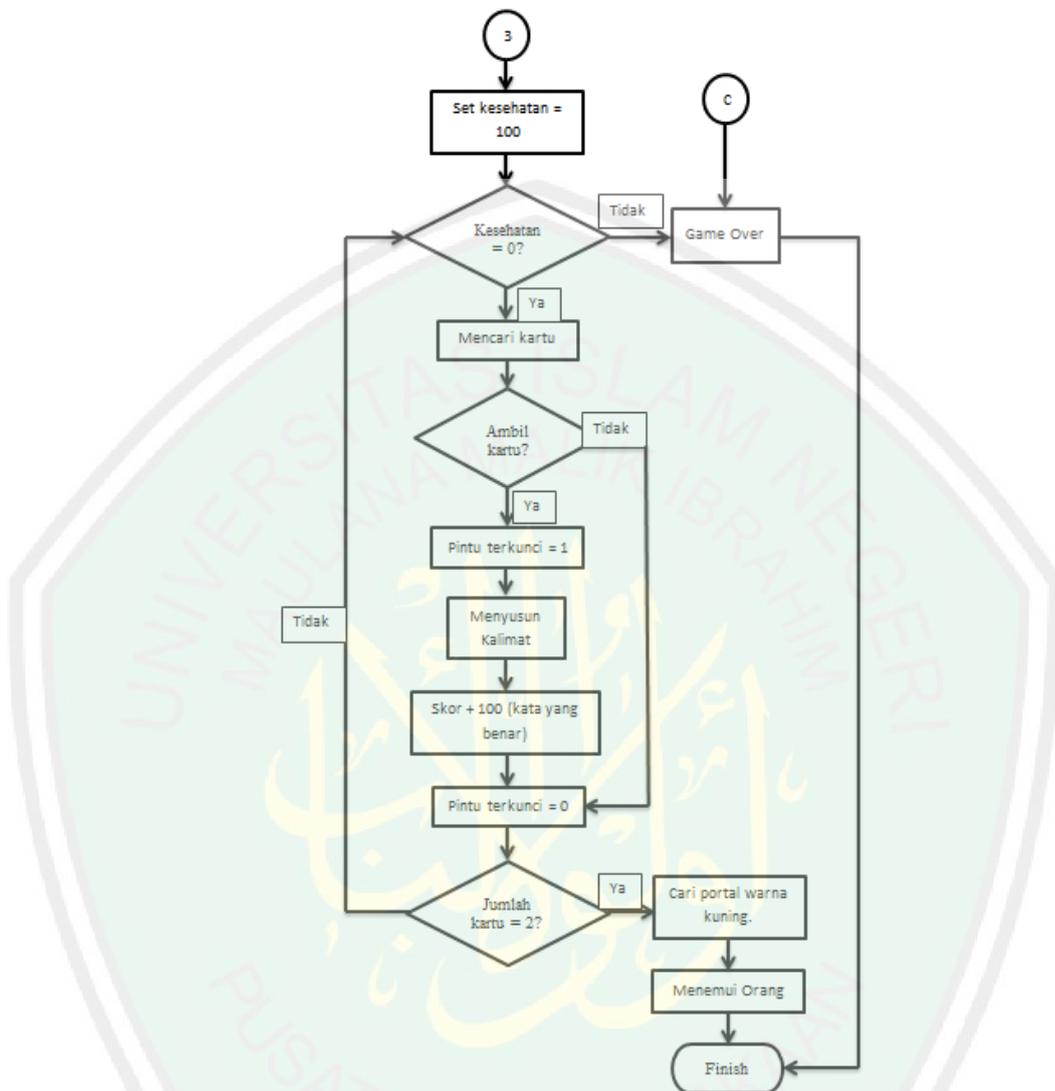
Gambar 3.11 Alur Level Ke-3

Proses yang dilalui pada level 3 sama dengan proses pada level 1 dan level 2. Setelah melewati portal berwarna kuning pada level 2, maka akan melanjutkan

permainan ke level 3. Dalam level 3 ini jumlah kartu semakin sedikit, sehingga dalam pencariannya agak sulit. Jumlah kartu yang berada di dalam level 3 yaitu 4 buah kartu. Semakin sedikit kartu yang ada di dalam setiap level maka tugas yang diberikan juga semakin sulit. Sama halnya dengan level 1 dan level 2, di dalam level 3 juga harus memenuhi tugas tersebut untuk dapat menemukan portal berwarna kuning dan melanjutkan perjalanan ke level selanjutnya.

### **3.2.6 Perancangan Alur Permainan Pada Level Keempat**

Dalam level ke-4 ini terdapat alur tambahan yaitu menemui seseorang yang ada pada lingkungan yang baru dalam game dan juga merupakan akhir dari game ini. Berikut alur level ke-4.



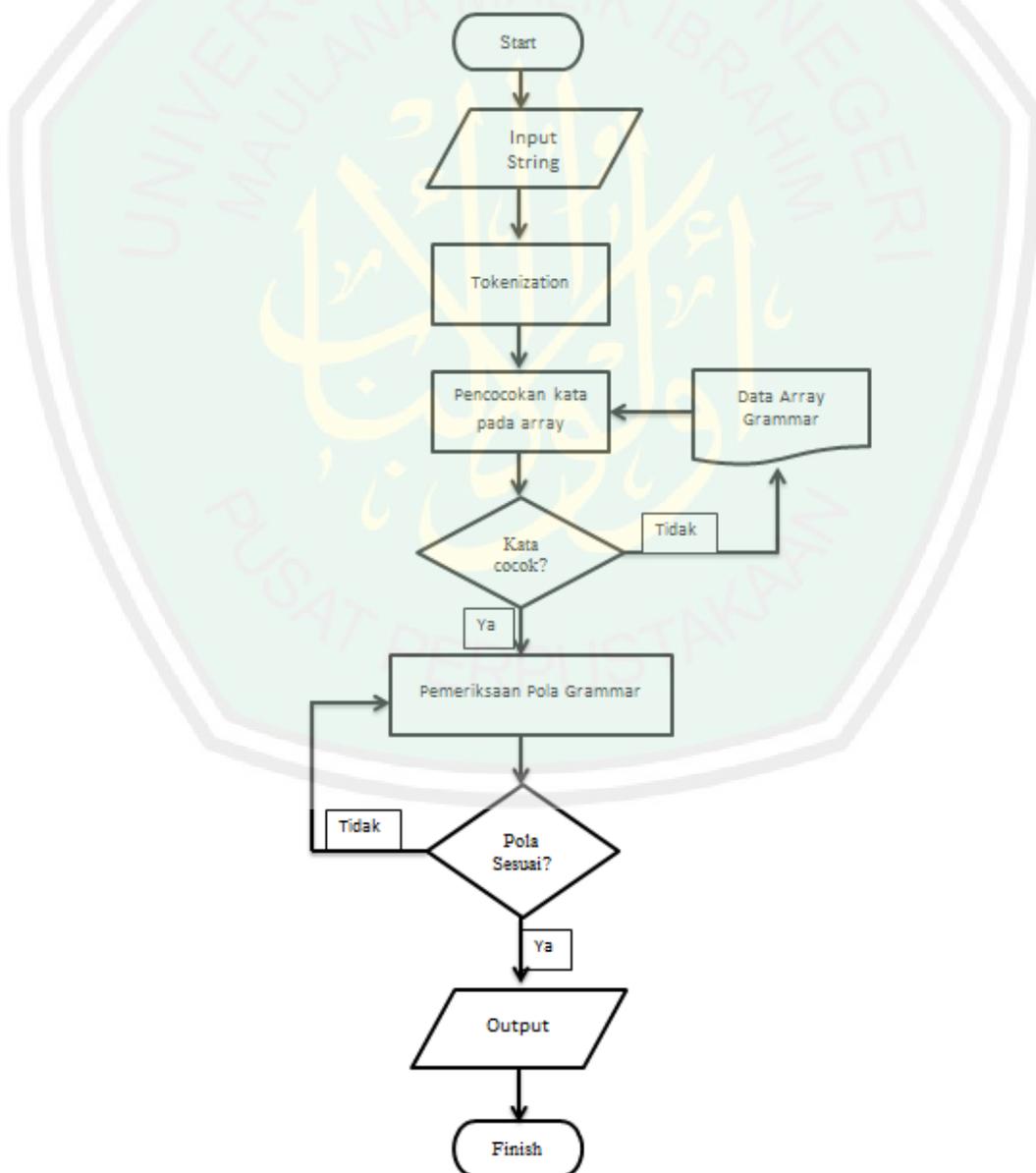
Gambar 3.12 Alur Level Ke-4

Memasuki level 4, jumlah yang harus dicari yaitu hanya 2 kartu tersisa, dikarenakan level 4 adalah level yang terakhir. Total kartu yang harus dicari dalam game ini adalah 20 kartu. Dalam level 4 ini pemain nantinya akan menemukan portal kuning terlebih dahulu sebelum memenuhi semua tugas. Akan tetapi portal tersebut belum bisa dilewati dikarenakan tugas pemain yang belum diselesaikan. Untuk melewati portal tersebut pemain harus memenuhi tugas

tersebut. Setelah melewati portal, pemain akan dihadapkan dengan latar game yang baru dan harus mencari orang di dalam sebuah rumah.

### 3.3 Perancangan Left Corner Parsing Sebagai Pemeriksa Kalimat

Alur algoritma Left Corner Parsing yang akan digunakan untuk pemeriksaan jawaban dari susunan kata yang dimasukkan oleh pemain. Berikut adalah alur pemeriksaan tersebut.



Gambar 3.13 Alur Sistem Pemeriksaan

Proses pertama yaitu proses *tokenization*. Proses *tokenization* adalah tahap yang mengolah input menjadi kata atau karakter (Nayak & Raghavan). Setelah menjadi kata atau karakter, proses selanjutnya mencocokkan dengan data yang berada dalam array. Apabila terjadi pencocokkan, langkah selanjutnya memeriksa setiap kata menggunakan aturan Left Corner Parsing. Aturan tersebut memeriksa setiap kata yang masuk dimana setiap kata tersebut memiliki jenis kata. Kata pertama dalam aturan *left corner* sangat penting, karena menentukan kalimat tersebut benar atau salah melalui aturan pemeriksaan dari sisi kalimat paling kiri. Setelah mengetahui jenis kata yang dimiliki, kemudian mencocokkan dengan pola bangun kalimat yang telah ditentukan, bila sesuai pola, selanjutnya mencocokkan kata kedua dari input yang masuk. Seperti langkah dalam memeriksa kata pertama, begitupun proses langkah kedua. Pemeriksaan kata yang masuk berakhir ketika semua kata telah selesai diperiksa.

Sebagai contoh, ada sebuah kalimat sederhana “i have a car”. Kemudian kalimat tersebut dijadikan ke unit yang lebih kecil menjadi “i, have, a, dan car”. Lalu masing-masing kata tersebut telah mempunyai jenis kata tersendiri. Jenis kata dari kata “i” adalah pronoun, lalu kata “have” adalah verb, kata “a” adalah adjective, kata “car” adalah noun. Setelah semuanya telah diketahui jenis dari setiap kata, maka selanjutnya yaitu mencocokkan dengan pola yang sudah dibentuk.

Sebagai contoh pola yang digunakan yaitu pronoun-verb-adjective-noun. Kemudian proses selanjutnya melakukan pemeriksaan tiap kata yang dimulai dari kata yang paling kiri, yaitu kata pertama. Kata pertama mempunyai jenis kata

pronoun lalu dicocokkan dengan pola yang sudah dibentuk, dan hasilnya sesuai dengan pola yang sudah dibentuk sebelumnya. Kata yang pertama kali diperiksa mempunyai peranan penting dalam sebuah kalimat, apabila kata pertama tidak sesuai, maka kalimat tersebut tidak akan menjadi sebuah kalimat. Setelah sesuai maka proses melanjutkan ke pemeriksaan terhadap kata kedua. Prosesnya pemeriksaannya sama dengan proses yang pertama, begitu juga dengan kata ketiga dan keempat. Proses ini berjalan hingga semua kata yang ada telah diperiksa dengan pola yang sudah dibentuk.

Berikut flowchat dari setiap proses yang berjalan dalam algoritma *left corner parsing*.

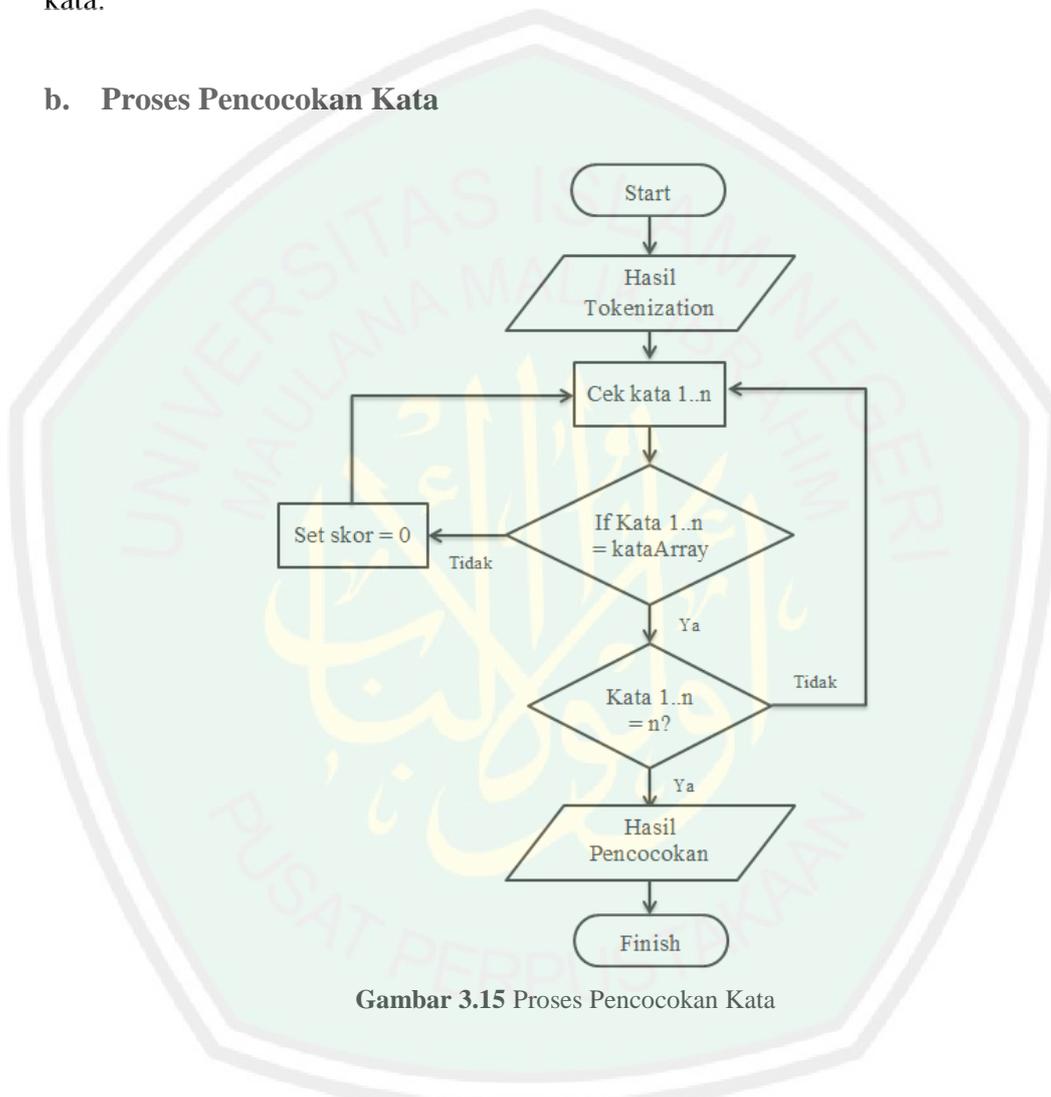
**a. Proses Tokenization**



**Gambar 3.14** Proses Tokenization

Gambar flowchart diatas menjelaskan tentang proses tokenization, mengolah input string yang masuk kemudian memisahkannya menjadi beberapa kata.

#### b. Proses Pencocokan Kata

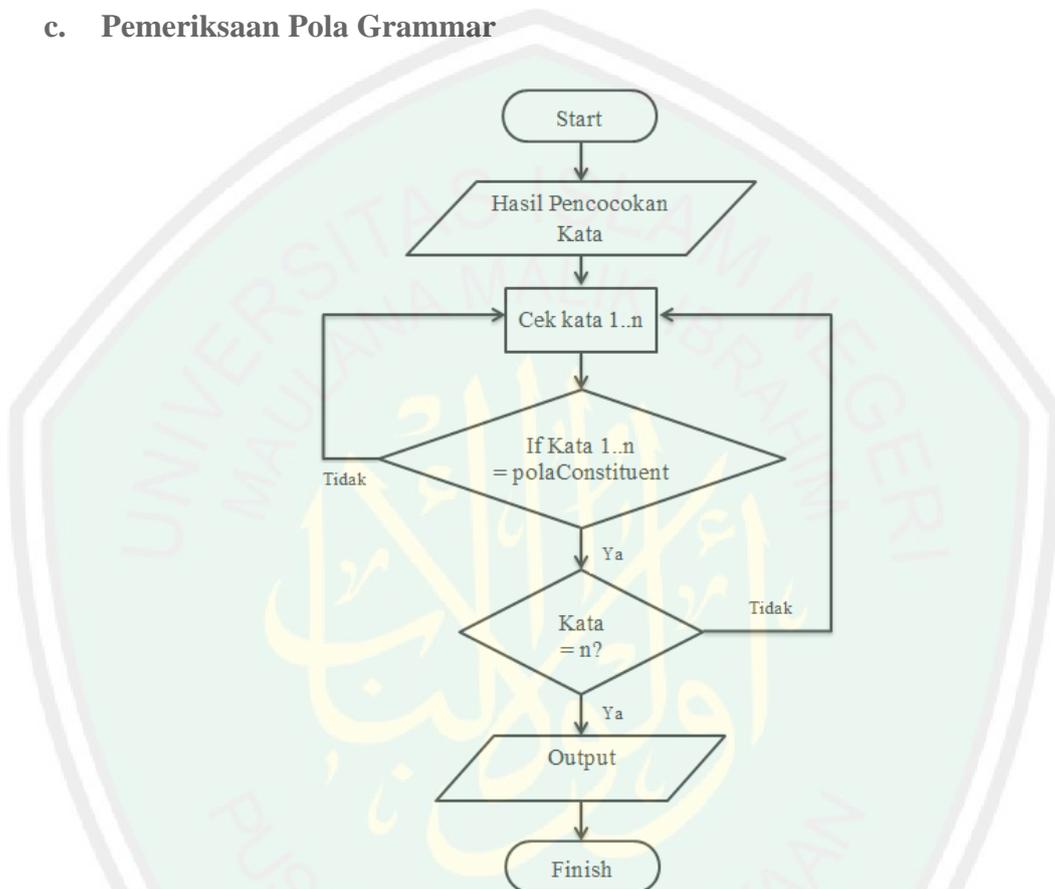


Gambar 3.15 Proses Pencocokan Kata

Flowchart diatas menjelaskan proses pencocokan kata, mengolah hasil tokenization yang kemudian terjadi pengecekan kata 1 sampai kata n dengan kata yang menjadi jawaban di dalam array. Apabila pengecekan tidak sesuai, maka kata yang telah dicek akan bernilai 0, dan bila kata yang akan melakukan pengecekan telah habis, maka didapat hasil dari pencocokan, bila belum habis,

proses selanjutnya mengecek kembali hasil dari tokenization hingga semua kata telah dicek.

### c. Pemeriksaan Pola Grammar



**Gambar 3.16** Pemeriksaan Pola Grammar

Flowchart diatas menjelaskan proses pemeriksaan pola grammar. Hasil dari pencocokkan kata sebelumnya kembali melakukan proses pengecekan kata. Apabila kata yang diolah sama dengan pola grammar yang ditentukan, maka selanjutnya mengecek hasil dari pencocokkan kata yang masih ada. Apabila sudah tidak ada, maka output akan keluar.

### 3.4 Kebutuhan Sistem

Pada bagian spesifikasi kebutuhan sistem ini, diulas tentang kebutuhan sistem perangkat lunak maupun perangkat keras yang mendukung dalam pembuatan maupun pada saat pengoperasian program aplikasi.

#### a. Kebutuhan Perangkat Keras (*hardware*)

Kebutuhan perangkat keras yang digunakan untuk mendukung proses pengembangan sistem aplikasi *game* ini adalah:

1. Intel(R) Core(TM) i3 CPU M370 @ 2.40GHz (4 CPUs), ~2.4GHz 1
2. RAM (Random Acces Memory) 1 GB.
3. VGA 128 MB
4. Hardisk 75 GB
5. Mouse
6. LCD/Monitor yang mendukung resolusi 1280x800 pixel.
7. Speaker

#### b. Kebutuhan Perangkat Lunak (*software*)

Perangkat lunak (*software*) yang digunakan untuk mendukung dalam pembuatan atau pengoperasian aplikasi *game* ini, antara lain:

1. Sistem Operasi yang digunakan adalah Windows 8 Pro 32-bit.

Sistem operasi windows merupakan sistem operasi yang user friendly, serta mendukung 3D Game Studio yang akan digunakan dalam membangun *game*.

2. Software 3D Game Studio

3D Game Studio atau 3DGS juga dikenal sebagai Gamestudio, adalah sistem pengembang permainan komputer 3D yang memperbolehkan pengguna membuat permainan 3D dan aplikasi realitas maya. Gamestudio meliputi model/terrain editor, level editor, script editor/debugger. Contoh permainan yang dibuat dengan Gamestudio adalah Kabus 22 (2006, Son Isik LTD, Merscom LLC) dan Citroën C4 Robot (2008, 2GEN Studio, Citroën Turkey). Versi terbaru adalah A8.

### 3. Adobe Photoshop CS3

Adobe Photoshop adalah program pengolah grafis yang sangat baik dan hampir mendukung semua format gambar. Dalam pembuatan aplikasi *game* ini banyak perannya diantaranya untuk pembuatan tombol. Dan pewarnaan *skin* objek 3D.

## 3.5 Cara Memainkan Game

Cara menjalankan *game* menggunakan kombinasi antara *mouse* dan *keyboard*. Cara tersebut ditunjukkan dalam tabel 3.1 berikut ini.

**Tabel 3.1** Fungsi Keyboard dan Mouse untuk Memainkan Game

<b>Perintah</b>	<b>Fungsi</b>
Tombol A,W, D, S	Berjalan ke kiri, depan, kanan, mundur ke belakang
Tombol anak panah atas, kanan, kiri, bawah	Berjalan ke depan, kanan, kiri, mundur ke belakang
Tombol spacebar	Mengeluarkan kekuatan magis
Mouse klik kiri	Memilih jawaban dari kuis
Geser mouse ke kanan, kiri, atas, bawah	Menggeser layar tampilan ke kanan, kiri, atas, bawah

### 3.6 Kebutuhan PC Pemain

Berikut ini merupakan tabel daftar spesifikasi komputer yang harus dimiliki pemain untuk memainkan *game* ini.

**Tabel 3.2** Kebutuhan PC Pemain

<b>Kebutuhan</b>	<b>Spesifikasi Minimum</b>	<b>Spesifikasi Rekomendasi</b>
Operating System	Windows XP, Vista, 7	Windows XP, Vista, 7
CPU	Pentium 4 1,7GHz	Dual Core 2,1 GHz
RAM	512 MB	1 GB
VGA Card	128 MB, 1024 x 768	256 MB, 1024 x 768
Sound Card	Support DirectX 9.0c	Support DirectX 9.0c
Keyboard & Mouse	-	-

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Implementasi

Dalam bab ini membahas tentang implementasi dari perancangan yang dibuat. Serta melakukan pengujian terhadap aplikasi yang dibuat untuk mengetahui apakah aplikasi tersebut telah berjalan sesuai dengan yang diharapkan.

#### 4.2 Implementasi *Artificial Intelligence* Proses Pemeriksaan Kalimat

Implementasi *artificial intelligence* pada game ini terletak ketika pemain menginputkan susunan kalimat kedalam kotak jawaban. Setelah diinputkan, susunan kata akan melakukan proses mengidentifikasi jenis kata yang dimiliki kata tersebut. Kata tersebut dimasukkan dan dikelompokkan ke dalam array, dan array tersebut nantinya akan digunakan dalam pencocokan pola yang telah dibentuk.

Dalam kasusnya, implementasi dari *left corner parsing* ini terdapat kelemahan yaitu dalam hal mengenal kata dan jenis kata yang ada. Sehingga yang seharusnya kalimat tersebut menurut pandangan seseorang tidak sesuai, akan tetapi memiliki pola grammar yang sama. Dilihat secara aturan dalam grammar, jawaban tersebut adalah benar. Hasil inputan juga dapat dipastikan di (<http://www.grammarly.com>) bahwa kalimat tersebut juga dikenali dengan pola yang sama. Untuk lebih jelasnya dapat dilihat pada tabel dibawah ini.

**Tabel 4.1** Hasil Penerapan Left Corner Parsing

No	Input Kalimat	Pola	Hasil
1.	My <u>hat</u> is <u>blue</u>	Pronoun-noun-verb-noun	Kalimat yang benar
2.	My <u>blue</u> is <u>hat</u>	Pronoun-noun-verb-noun	<i>Blue</i> dan <i>hat</i> memiliki jenis kata yang sama, Secara pelafalan kalimat kurang tepat.
3.	My brother <u>is sick</u>	Pronoun-noun-verb-verb	Kalimat yang benar
4.	My brother <u>sick is</u>	Pronoun-noun-verb-verb	<i>Sick</i> dan <i>is</i> mempunyai sama pola. Secara pelafalan kurang tepat.
5.	<u>Della</u> send <u>me</u> a <u>letter</u>	Noun-verb-noun-adjective-noun	Kalimat yang benar
6.	<u>Letter</u> send <u>Della</u> a <u>me</u>	Noun-verb-noun-adjective-noun	<i>Letter</i> , <i>Della</i> , dan <i>me</i> mempunyai pola yang sama. Secara pelafalan kalimat kurang tepat.
7.	<u>Jakarta</u> is hot <u>weather</u>	Noun-verb-adjective-noun	Kalimat yang benar
8.	<u>Weather</u> is hot <u>Jakarta</u>	Noun-verb-adjective-noun	<i>Jakarta</i> dan <i>weather</i> mempunyai sama pola, secara pelafalan kurang tepat
9.	My <u>father</u> have two <u>cars</u>	Pronoun-noun-verb-adjective-noun	Kalimat yang benar
10.	My <u>cars</u> have two <u>father</u>	Pronoun-noun-verb-adjective-noun	<i>Cars</i> dan <i>father</i> mempunyai sama pola, secara pelafalan kurang tepat
11.	I <u>can speak</u> english <u>very well</u>	Pronoun-verb-verb-noun-adverb-adverb	Kalimat yang benar
12.	I <u>speak can</u> english very well	Pronoun-verb-verb-noun-adverb-adverb	<i>Speak</i> dan <i>can</i> mempunyai sama pola, secara pelafalan kurang tepat
13.	I <u>speak can</u> english <u>well very</u>	Pronoun-verb-verb-noun-adverb-adverb	<i>Very</i> dan <i>well</i> mempunyai sama pola, secara pelafalan kurang tepat
14.	<u>Malang</u> is cold <u>city</u>	Noun-verb-adjective-noun	Kalimat yang benar
15.	<u>City</u> is cold <u>Malang</u>	Noun-verb-adjective-noun	<i>Malang</i> dan <i>city</i> mempunyai sama pola, secara pelafalan kurang tepat

Hasil inputan yang terdapat pada tabel diatas membuktikan bahwa *left corner parsing* belum mampu mengenali kata yang mempunyai jenis kata yang sama. *Left corner* hanya melakukan pemeriksaan terhadap pola dari jenis kata yang dibangun. Sebagai contoh kalimat nomor 5 dan 6. Kedua kalimat mempunyai pola yang sama, akan tetapi secara pelafalan atau pengucapan adalah salah.

Seperti penelitian sebelumnya yang dilakukan oleh (Abdurohman, Hadi, & Rohidin, 2006) dalam pemeriksaan tata bahasa Inggris menggunakan *left corner parsing*. Penelitian tersebut menggunakan visual basic dan hasil dari sistem yang dibangun mengalami hal yang sama. Perbedaannya terdapat pada penerapan yang dilakukan oleh peneliti sebelumnya dengan penulis adalah penulis menerapkannya pada game pembelajaran tentang *grammar*, yang mana dapat membantu sebagai media pembelajaran serta bermain.

Dalam penerapannya pada game, proses pemeriksaan kalimat tersebut dapat ditunjukkan pada *script* berikut:

```

if ((kal == 1) && (soalitem == 1))
{
    if (str_cmp(jawaban1,chr(pnoun[i])) //pencocokkan kata dalam array
    {
        if (str_cmp(jawaban1,J1_la)) //pengecekan kata benar
        {word_right1 = word_right1 + 100;    jml_score = jml_score + 100;}
        else
        {str_cpy(wrong1,"Kata pertama: ");    str_cat(wrong1,jawaban1);
        word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}
    }
    else
    {str_cpy(wrong1,"Kata pertama: ");    str_cat(wrong1,jawaban1);
    word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}

    if (str_cmp(jawaban2,chr(noun[i])) //pencocokkan kata dalam array
    {
        if(str_cmp(jawaban2,J2_la))
        {word_right1 = word_right1 + 100;    jml_score = jml_score + 100;}
        else
        {str_cpy(wrong2,"Kata kedua: ");    str_cat(wrong2,jawaban2);
        word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}
    }
    else
    {str_cpy(wrong2,"Kata kedua: ");    str_cat(wrong2,jawaban2);
    word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}
}

```

```

if (str_cmp(jawaban3,_chr(verb[i])) //pencocokkan kata dalam array
{
    if (str_cmp(jawaban3,J3_1a))
    {word_right1 = word_right1 + 100;    jml_score = jml_score + 100;}
    else
    {str_cpy(wrong3,"Kata ketiga: ");    str_cat(wrong3,jawaban3);
    word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}
}
else
{str_cpy(wrong3,"Kata ketiga: ");    str_cat(wrong3,jawaban3);
word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}

if (str_cmp(jawaban4,_chr(noun[i])) //pencocokkan kata dalam array
{
    if (str_cmp(jawaban4,J4_1a))
    {jml_score = jml_score + 100; word_right1 = word_right1 + 100;}
    else
    {str_cpy(wrong4,"Kata keempat: ");    str_cat(wrong4,jawaban4);
    word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}
}
else
{str_cpy(wrong4,"Kata keempat: ");    str_cat(wrong4,jawaban4);
word_right1 = word_right1 + 0;    jml_score = jml_score + 0;}

hilang();
wait(5);
set(kataSalah1, SHOW);
set(kataSalah2, SHOW);
set(kataSalah3, SHOW);
set(kataSalah4, SHOW);
set(salah, SHOW);
wait(300);
reset(kataSalah1, SHOW);
reset(kataSalah2, SHOW);
reset(kataSalah3, SHOW);
reset(kataSalah4, SHOW);
reset(salah, SHOW);
set(gram1, SHOW);
wait(600);
reset(gram1, SHOW);
jml_card1 = jml_card1 + 1;
viewterkunci = 0;
oke=1;
wakpause=0;
}

```

Script diatas merupakan proses pemeriksaan dan juga pencocokkan kata pada game. Proses yang pertama dilakukan yaitu memeriksa apakah kata pertama yang diinputkan terdapat di dalam data array. Proses tersebut dapat dilihat pada script `if (str_cmp(jawaban1,_chr(pnoun[i]))`. Script tersebut membandingkan inputan kata pertama dengan data array yang ada. Pembanding pada script terdapat pada fungsi `str_cmp`, dengan membandingkan inputan kata `jawaban1` dengan `_chr(pnoun[i])` sebagai data simpanan dalam bentuk array. Setelah pemeriksaan proses selanjutnya yaitu pencocokkan kata dari inputan kata pertama

dengan hasil jawaban yang benar. Dalam script dapat ditunjukkan pada script `if (str_cmp(jawaban1,J1_1a))`. Seperti proses yang pertama, proses kedua ini pun membandingkan inputan kata, tetapi inputan kata membandingkan kata dengan kata yang benar. Script `J1_1a` adalah string yang berisi jawaban dari inputan pertama. Apabila tidak cocok maka inputan yang dicocokkan ditampilkan script `set(kataSalah1, SHOW);`, dengan mengambil data sebelum ditampilkan dari script `str_cpy(wrong1,"Kata pertama: "); str_cat(wrong1,jawaban1);`. Fungsi `str_cat` yaitu memindahkan isi string `jawaban1` kedalam string `wrong1` dan ditampilkan pada `text kataSalah1`.

### 4.3 Implementasi Aplikasi Game

Implementasi merupakan proses pembangunan komponen-komponen pokok suatu sistem, yang didasarkan pada desain dan rancangan yang telah dibuat sebelumnya. Berikut adalah rincian tentang implementasi aplikasi game yang telah dibuat.

#### 4.3.1 Antarmuka Intro

Pada bagian ini akan ditampilkan beberapa gambar berupa tokoh karakter yang ada serta cerita singkat tentang tujuan pemain untuk menyelesaikan game ini.



Gambar 4.1 Intro Pengantar Game



Gambar 4.2 Intro Pengantar Game Ke-2



Gambar 4.3 Intro Pengantar Game Ke-3

### 4.3.2 Antarmuka Menu

Menu dalam *game* ini terdiri dari 5 pilihan, antara lain “*New Game*”, “*Setting*”, “*About*”, “*Help*” dan “*Exit*”. Berikut ini fungsi dari masing-masing tombol:

Tabel 4.2 Fungsi Menu Game

Tombol	Fungsi
New Game	Untuk memulai permainan.
Setting	Untuk memunculkan pengaturan besar-kecilnya volume suara <i>background</i> musik dan suara efek.
About	Berisi tujuan pembuatan <i>game</i> ini dan tentang pembuat <i>game</i> .
Help	Untuk memunculkan keterangan bantuan. Bantuan di sini berupa bagaimana cara menjalankan permainan.
Exit	Untuk keluar dari permainan.



Gambar 4.4 Menu Game



Gambar 4.5 Menu New Game



Gambar 4.6 Menu Setting 1



Gambar 4.7 Menu About



Gambar 4.8 Menu Help



Gambar 4.9 Menu Exit

### 4.3.3 Antarmuka Latar Level 1, 2, dan 3

Antarmuka level 1, 2, dan 3 menggunakan latar yang sama hanya jumlah kartu yang berbeda tiap level.



Gambar 4.10 Latar Level



Gambar 4.11 Pencarian Kartu di Dalam Rumah



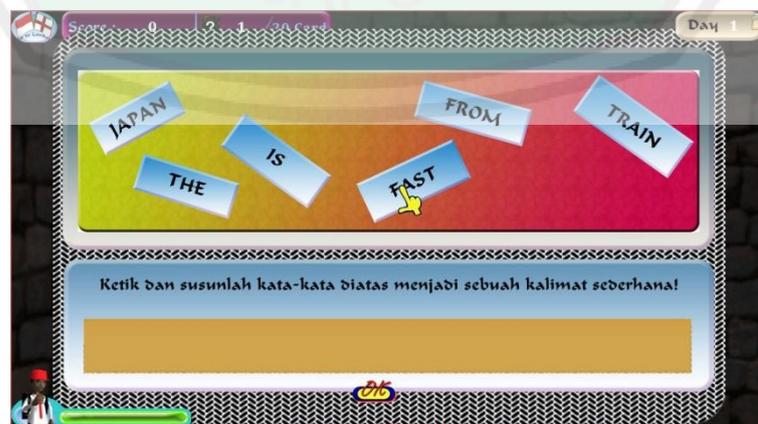
Gambar 4.12 Pemberian Tugas

Dalam gambar 4.12 pemain akan menerima tugas dari NPC untuk menyusun kalimat disetiap pemain mengambil kartu, sekaligus pintu yang akan dilalui akan tutup terkunci, sehingga pemain tidak bisa keluar kecuali pemain tersebut menyusun tugas yang diberikan.



Gambar 4.13 Pencarian Portal

Dalam gambar 4.13 disetiap level akan menemukan sebuah pintu portal untuk melanjutkan ketahap selanjutnya. Syarat yang harus dipenuhi yaitu melengkapi kartu yang dibutuhkan disetiap level. Pada awal setiap level akan diberitahukan berapa jumlah kartu yang harus dicari.

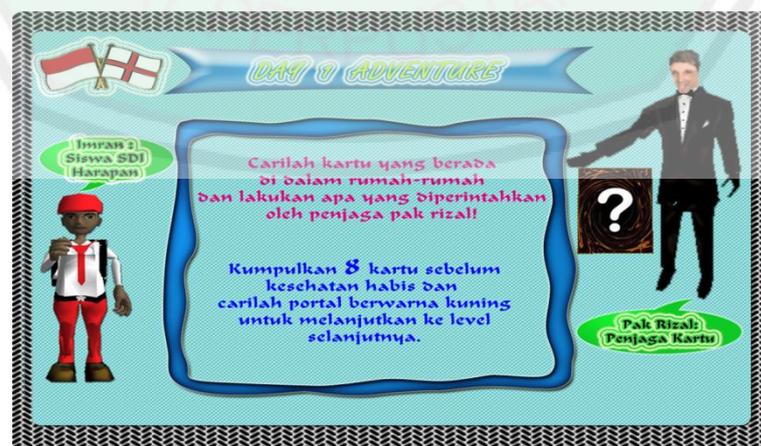


Gambar 4.14 Tugas Menyusun yang Diberikan

Proses pengacakan soal kalimat yang keluar di dalam script yaitu sebagai berikut:

```
function soall_1()
{
    kal=1;
    var randomm = integer(random(2));
    if (randomm == 0)
    {
        soalitem = 1;
        wait(50);
        set(soall_1_a, SHOW);
        set(jawaban1_txt, SHOW);
        set(jawaban2_txt, SHOW);
        set(jawaban3_txt, SHOW);
        set(jawaban4_txt, SHOW);
    }
    else if (randomm == 1)
    {
        soalitem = 2;
        wait(50);
        set(soall_1_b, SHOW);
        set(jawaban1_txt, SHOW);
        set(jawaban2_txt, SHOW);
        set(jawaban3_txt, SHOW);
        set(jawaban4_txt, SHOW);
    }
}
```

Script diatas adalah contoh pengacakan yang terjadi pada salah satu soal kalimat. Apabila variabel `randomm` bernilai `== 0`, maka soal dengan item 1 yang akan keluar, begitupun dengan nilai `== 1`, maka soal dengan item 2 yang akan keluar. Dalam setiap kalimat yang akan keluar pada game, dibuat 2 jenis kalimat yang berbeda dalam setiap kartu yang didapat.



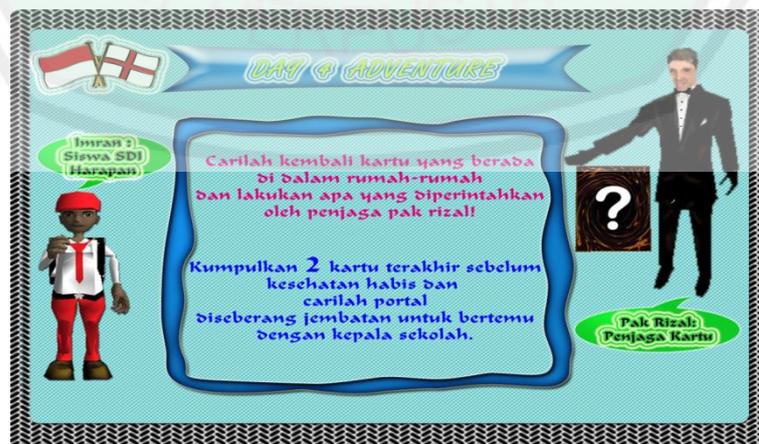
Gambar 4.15 Tugas yang Harus Dipenuhi Level 1



Gambar 4.16 Tugas yang Harus Dipenuhi Level 2



Gambar 4.17 Tugas yang Harus Dipenuhi Level 3



Gambar 4.18 Tugas yang Harus Dipenuhi Level 4

#### 4.3.4 Antarmuka Latar Level 4

Dalam level 4 ini latar yang digunakan sama dengan level sebelumnya akan tetapi hanya mencari kartu yang jumlahnya tidak banyak, yaitu 2 kartu tersisa yang belum terambil. Setelah melengkapinya semua kartu, pemain harus melewati portal seperti level sebelumnya juga, tetapi portal tersebut tidak menuju ke level selanjutnya, melainkan menuju latar yang berbeda dengan level yang masih sama. Berikut gambar dari portal tersebut.



Gambar 4.19 Latar Level-4

Setelah melewatinya tugas pemain yaitu mencari seseorang yang sedang menunggu di sebuah bangunan, untuk memberikan sesuatu yang diinginkan pemain. Kemudian sesi pertemuan tersebut juga adalah sebagai bagian akhir dari game ini diikuti hasil dari penilaian terhadap semua susunan kalimat yang dibuat.



Gambar 4.20 Latar Lain di Level-4



Gambar 4.21 Menemui NPC Akhir

#### 4.4 Pengujian Game

Hasil rekapitulasi dari pengujian game yang diujikan kepada 20 anak berumur 8 sampai 13 tahun di dusun Bendo, desa Ngadirejo, kecamatan Pogalan, kabupaten Trenggalek sebagai responden dari uji coba aplikasi ini.

Uji coba ini dilaksanakan untuk mengetahui kelayakan game ini dibuat. Sistem uji coba yang dilakukan yaitu setiap anak memainkan game secara bergantian hingga game selesai. Kemudian para responden mengisi kuisisioner yang telah disediakan. Berikut adalah tabel rekapitulasi yang telah terdata.

**Tabel 4.3** Hasil Rekapitulasi Uji Coba

No	Uraian	Penilaian	
		Ya	Tidak
1.	Saya mengerti cara bermain game “Go to London Adventure”.	100%	0%
2.	Saya mengerti fungsi setiap menu dalam game.	95%	5%
3.	Saya mengetahui cara memainkan game ini.	100%	0%
4.	Saya mengerti petunjuk yang diberikan dalam di game.	100%	0%
5.	Saya dapat menemukan kartu dengan mudah.	85%	15%
6.	Saya lebih mengerti tentang grammar setelah bermain game ini.	90%	10%
7.	Apakah anda bosan dengan bermain game ini?	0%	100%
8.	Game “Go To London” menarik dan menyenangkan.	100%	0%
9.	Saya ingin belajar grammar dari game ini.	100%	0%
10.	Saya ingin memainkan game ini kembali.	100%	0%
Total keseluruhan		87%	13%
Prosentase ketertarikan terhadap game		97%	

Dari hasil data uji coba diatas kesimpulan yang dapat diperoleh secara umum bahwa responden mengerti bagaimana memainkan game ini. Terbukti seluruh responden dapat mengerti cara bermain game ini dan mengerti petunjuk-petunjuk yang diberikan dalam game dengan nilai prosentase 100%. Kemudian sebanyak 95% responden mengerti fungsi dari setiap menu yang ada. Sebanyak 85% dapat menemukan kartu yang dicari dalam game, 90% lebih dapat mengerti tentang *grammar* setelah bermain game ini, dan menurut seluruh responden game membuktikan dengan nilai prosentasi 97% responden tertarik terhadap game ini

karena menarik dan menyenangkan sehingga responden ingin memainkan kembali game ini, serta dapat memanfaatkan game ini sebagai tempat untuk belajar dan bermain tentang grammar.

Berdasarkan uraian diatas dapat dikatakan uji coba pada aplikasi game yang dibuat dapat dilaksanakan dengan lancar dan berhasil.

#### 4.5 Integrasi Game Pembelajaran Bahasa Inggris pada Ayat Al-Qur'an

Bahasa inggris merupakan bahasa yang digunakan orang diseluruh dunia untuk berkomunikasi, dikarenakan bahasa Inggris adalah bahasa internasional. Dalam pembelajarannya, bahasa Inggris memang agak sulit untuk mempelajarinya. Akan tetapi sekarang banyak cara dalam mempelajari bahasa Inggris, seperti yang terdapat pada Q.S. an-Nahl ayat 78 serta Q.S. al\_israa ayat 36:

وَاللَّهُ أَخْرَجَكُمْ مِنْ بُطُونِ أُمَّهَاتِكُمْ لَا تَعْلَمُونَ شَيْئًا وَجَعَلَ لَكُمُ السَّمْعَ وَالْأَبْصَرَ وَالْأَفْئِدَةَ  
لَعَلَّكُمْ تَشْكُرُونَ ﴿٧٨﴾

*Artinya “Dan Allah mengeluarkan kamu dari perut ibumu dalam keadaan tidak mengetahui sesuatupun, dan dia memberi kamu pendengaran, penglihatan dan hati, agar kamu bersyukur”.*(Q.S. an-Nahl 16:78)

وَلَا تَقْفُ مَا لَيْسَ لَكَ بِهِ عِلْمٌ إِنَّ السَّمْعَ وَالْبَصَرَ وَالْفُؤَادَ كُلُّ أُولَئِكَ كَانَ عَنْهُ مَسْئُولًا ﴿٣٦﴾

*Artinya “Dan janganlah kamu mengikuti apa yang kamu tidak mempunyai pengetahuan tentangnya. Sesungguhnya pendengaran, penglihatan dan hati, semuanya itu akan diminta pertanggung jawaban”.*(Q.S. al-Israa 17:36)

Melihat dari kedua ayat di atas, terdapat urutan “Pendengaran” (as-sam’a), lalu “Penglihatan” (al-abshar/al-bashar), dan “hati/akal-budi” (al-Fuaad). Urutan

tersebut merupakan urutan (tahap-tahap) dalam mempelajari atau menguasai suatu bahasa, seperti bahasa Indonesia, Arab dan Inggris.

**Tahap 1. “Pendengaran” (as-sam’a):**

Dalam bahasa Inggris disebut dengan “Listening Step” (Tahap Pendengaran). Tahap ini menunjukkan bahwa setiap orang (setiap anak balita) menguasai bahasa dimulai dengan aktifitas telinganya, yaitu mendengar suara yang keluar dari mulut manusia (terutama dari mulut seorang ibu), sehingga kemudian bahasa pertama ini disebut dengan “bahasa ibu” (mother tongue). Bayi dilahirkan dalam keadaan “*laa ta’lamuuna syai’an*” (QS 16:78 – tidak tahu apa-apa, tanpa ilmu), lalu dalam menguasai suatu bahasa, kegiatan telinga (mendengar) adalah kegiatan pertama.

**Tahap 2. “Penglihatan” (al-abshar/al-bashar):**

Tahap kedua ini adalah tahap “penglihatan”, yg dalam bahasa Inggris disebut dengan “Reading Step” (Tahap baca). Reading tentunya berpasangan dengan writing, karena memang tidak bisa membaca tanpa ada tulisan. So, tahap kedua ini boleh disebut dengan “Reading-writing Step”. Pelajaran ttg “reading” ada dua; “reading pronunciation” dan “reading comprehension”.

“Reading pronunciation” adalah cara belajar membaca dengan tepat, yaitu membaca dengan “tartil”, dengan “mahraj” atau “tajwid” yg tepat, dengan “kefashihan” yg bagus. Reading jenis ini dapat dilalui dg terlebih dahulu menguasai tahap pertama tsb di atas, yaitu mendengar, menirukan, lalu praktik membacanya. “Reading comprehension” adalah cara belajar membaca yg lebih tinggi, yaitu membaca dalam rangka memahami isi bacaan. Penguasaan kosa kata



karangan, dan sebagainya. Tahap ini dapat dilakukan dengan baik jika sudah benar-benar matang dalam tahap-tahap sebelumnya. Selain itu, tahap ini juga berbicara soal nilai-nilai dalam bahasa, seperti nilai halal-haram, sopan-santun, manfaat, dan sebagainya. Nilai-nilai inilah yang harus dimasukkan dalam listening, speaking, reading, dan writing dalam mempelajari sebuah bahasa.



## BAB V

### PENUTUP

#### 5.1 Kesimpulan

Hasil pengujian dalam menerapkan *left corner parsing* pada sebuah game adventure 3D yang telah dibangun dalam proses memeriksa sebuah susunan kata, menyimpulkan bahwa algoritma ini mampu mengenali pola *grammar* dengan baik, sehingga dapat digunakan untuk memeriksa kalimat tersebut sesuai pola yang dibentuk atau tidak, akan tetapi dilihat dari tabel 4.1 untuk hal memilih kata yang cocok sesuai definisi yang dimiliki sebuah kalimat belum bisa, sehingga terdapat kalimat yang secara pola *grammar* dapat dikenali dan benar, akan tetapi secara pelafalan dirasa kurang tepat.

Dalam hasil uji coba kepada 20 responden anak-anak yang berumur antara 8 sampai 13 tahun, dapat disimpulkan bahwa game “*Go to London Adventure*” ini menyenangkan dan menarik bagi responden, terbukti total keseluruhan responden tertarik terhadap game yang dibuat, yaitu dengan nilai prosentase sebesar 97%.

#### 5.2 Saran

Masih banyak kekurangan dalam game ini. Kekurangan yang ada bisa dijadikan referensi dalam mengembangkan game yang lebih kreatif dan variatif, terutama dalam penerapan *artificial intelligence* pada game. Adapun saran yang disampaikan oleh penulis untuk pengembangan game selanjutnya, yaitu:

1. Game ini dapat dikembangkan lagi dengan cara merubah aturan dalam game atau dari segi desain dan tampilan game sebelumnya.

2. Tidak hanya dalam memeriksa kebenaran suatu pola, akan tetapi game juga dapat dikembangkan dalam memeriksa makna yang ada sebelum kalimat tersebut tersusun.
3. Mengembangkan game ini dalam device mobile, seperti Android, IOS Phone, dan lain-lain.



## DAFTAR PUSTAKA

- Abdurohman, M., Hadi, S., & Rohidin, D. (2006). Pemeriksaan Tata Bahasa Dalam Kalimat Bahasa Inggris Menggunakan Algoritma Left Corner Parsing. *Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT 2006)*, (hal. 76-82). Depok.
- Arikunto, S. (2010). *Prosedur Penelitian Suatu Pendekatan Praktik*. Jakarta: RINEKA CIPTA.
- Coghill, J., & Magendanz, S. (2003). *English Grammar*. New York: Wiley Publishing, Inc.
- Effendi, M. N. (2013). *APLIKASI PERMAINAN MATH MAZE MENGGUNAKAN ALGORITMA GROWING TREE SEBAGAI PEMBANGKIT DAN PENYELESAI PERMAINAN*. Skripsi, Universitas Islam Negeri Maulana Malik Ibrahim, Teknik Informatika, Malang.
- Greenbaum, S. (1996). *English Grammar*. Oxford University Press.
- Henry, S. (2005). *Panduan Praktis Membuat Game 3D*. Yogyakarta: GRAHA ILMU.
- Hidayat, R. T. (2006). *Belajar Mudah Bahasa Inggris*. Yogyakarta: PUSTAKA PELAJAR.
- <http://www.grammarly.com>.
- Kristanto, A. (2004). *Kecerdasan Buatan*. Yogyakarta: GRAHA ILMU.
- Nasution, S. H., Widiastuti, W. L., & Pasaribu, S. M. (1985). *Petunjuk Pemrograman C Edisi Kedua*. Jakarta: Erlangga.
- Nina, A. R. (2010). *English Grammar*. Yogyakarta: Pustaka Widiatama.
- Priyana, D. (2013, Maret 19). *Pentingnya ber-Bahasa Inggris*. (D. Priyana, Penyunting) Dipetik Mei 16, 2013, dari Azzahra News: <http://news.universitassazzahra.ac.id/pentingnya-ber-bahasa-inggris/>
- Sibero, I. C. (2009). *Langkah Mudah Membuat Game 3D*. Yogyakarta: MediaKom.

Suciadi, J. (2001). STUDI ANALISIS METODE-METODE PARSING DAN INTERPRETASI SEMANTIK PADA NATURAL LANGUAGE. *JURNAL INFORMATIKA*, II(1), 13-22.

Sujiono, Y. N. (2009). *Konsep Dasar Pendidikan Anak Usia Dini*. Jakarta: INDEKS.

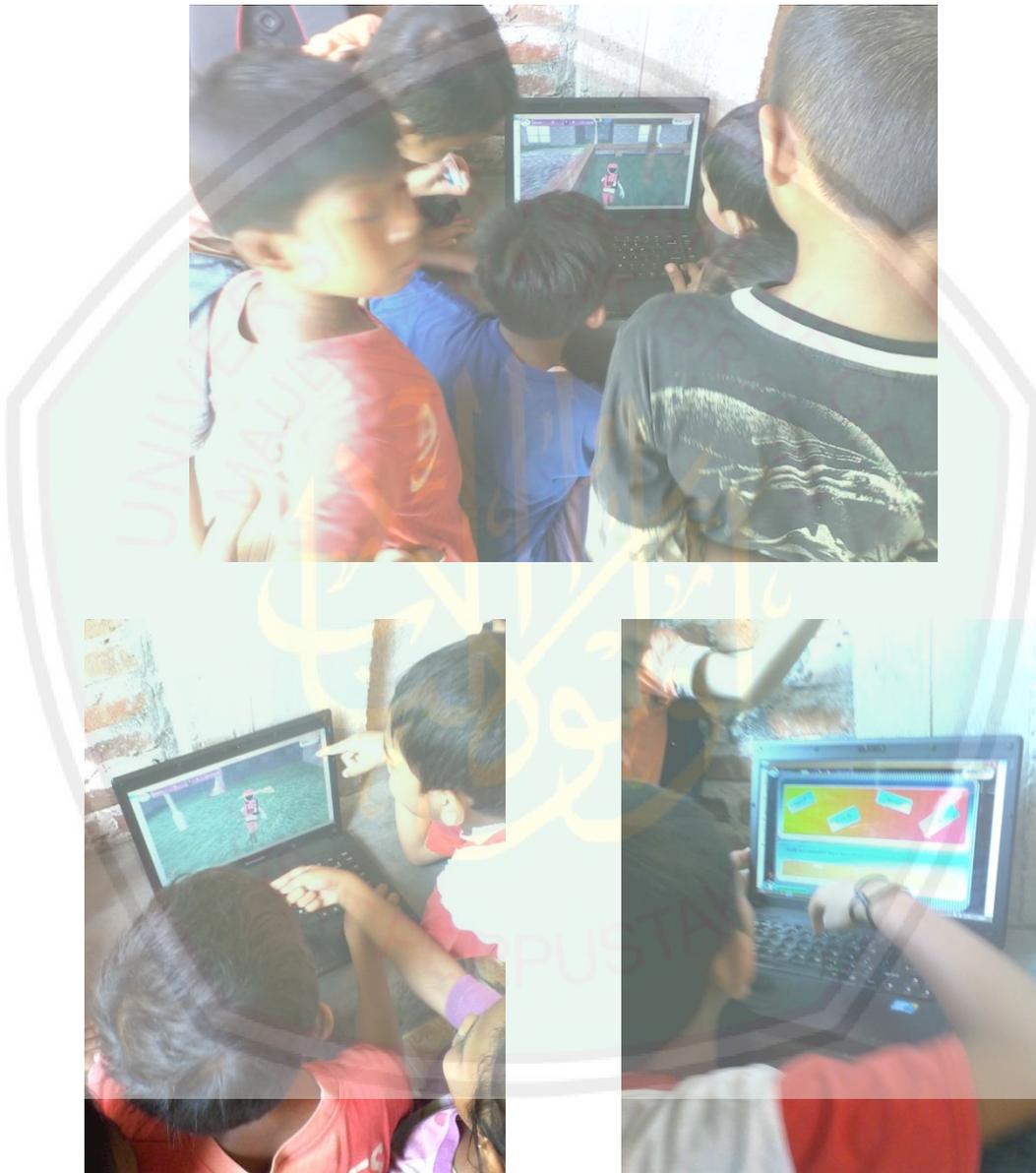
Sumarno. (2011). *Mengembangkan Bakat Anak*. (Universitas Surabaya) Dipetik Juni 29, 2013, dari <http://blog.elearning.unesa.ac.id/alim-sumarno/mengembangkan-bakat-anak>

Swan, M. (2005). *Practical English Usage: 3rd Edition*. Oxford University Press.

Winarno, E. D. (2006). *Belajar Tuntas Tata Bahasa Inggris (Tanpa Guru)*. Yogyakarta: PUSTAKA PELAJAR.



LAMPIRAN



Gambar Uji Coba