

**GAME PETUALANG EDUKASI BAHASA INGGRIS
PADA ANAK SEKOLAH DASAR**

Oleh

NURIMAN HAMKA GENTI

T3116043

SKRIPSI

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



**PROGRAM SARJANA
TEKNIK INFORMATIKA
UNIVERSITAS ICHSAN GORONTALO
GORONTALO
2020**

PENGESAHAN SKRIPSI

GAME PETUALANG EDUKASI BAHASA INGGRIS PADA ANAK SEKOLAH DASAR

Oleh

NURIMAN HAMKA GENTI

T3116043

SKRIPSI

Untuk memenuhi salah satu syarat ujian
guna memperoleh gelar Sarjana
Program Studi Teknik Informatika,
ini telah disetujui oleh Tim Pembimbing

Gorontalo, 01 Agustus 2020

Pembimbing I



Irvan A Salihi, M.Kom
NIDN: 0928028101

Pembimbing II



Muis Nanja, M.Kom
NIDN: 0905078703

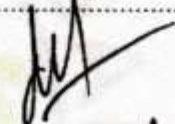
PERSETUJUAN SKRIPSI

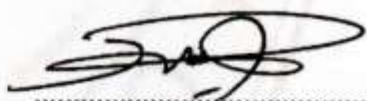
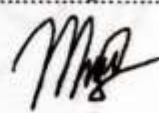
GAME PETUALANG EDUKASI BAHASA INGGRIS PADA ANAK SEKOLAH DASAR

Oleh
NURIMAN HAMKA GENTI
T3116043

Diperiksa oleh Panitia Ujian Strata Satu (S1)
Universitas Ichsan Gorontalo
Gorontalo, 06 Agustus 2020

1. Ketua Penguji
Asmaul Husna Nasrullah, M.Kom
2. Anggota
Sudirman S. Panna, M.Kom
3. Anggota
Andi Bode, M.Kom
4. Anggota
Irvan Abraham Salihi, M.Kom
5. Anggota
Muis Nanja, M.Kom


.....

.....

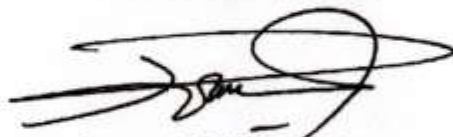
.....

.....

.....

Mengetahui :

Dekan Fakultas Ilmu Komputer


Zohrahayaty, M.Kom
NIDN: 0912117702

Ketua Program Studi


Irvan A Salihi, M.Kom
NIDN: 0928028101

PERNYATAAN SKRIPSI

Dengan ini saya menyatakan bahwa:

1. Karya tulis (Skripsi) saya ini adalah asli dan belum pernah diajukan untuk mendapat gelar akademik (Sarjana) baik di Universitas Ichsan Gorontalo maupun di perguruan tinggi lainnya.
2. Karya tulis (Skripsi) saya ini adalah murni gagasan, rumusan, dan penelitian saya sendiri tanpa bantuan pihak lain, kecuali arahan dari Tim Pembimbing.
3. Dalam karya tulis (Skripsi) saya ini tidak terdapat karya atau pendapat yang telah dipublikasikan orang lain, kecuali secara tertulis dicantumkan sebagai acuan/sitasi dalam naskah dan dicantumkan pula dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma-norma yang berlaku di Universitas Ichsan Gorontalo

Gorontalo, 01 Agustus 2020

Saya Membuat Pernyataan,



Nuriman Hamka Genti

ABSTRAK

Seiring dengan kebanyakan sekolah dasar masih menerapkan metode pembelajaran menggunakan media buku panduan, begitu pula dengan metode belajar pada Sekolah Dasar Negeri, adapun sekolah-sekolah masih sering menggunakan metode cerama maka dari itu siswa hanya jadi penonton saja, maka, penulis berinisiatif membuat suatu aplikasi game edukasi untuk suatu pembelajaran pada anak-anak dalam bahasa inggris. Kelebihan dari game ini adalah dijalankan dalam keadaan *offline* dan dapat membuat si anak mudah untuk mengikuti pelajaran tanpa harus menunggu guru dan memotivasi si anak untuk giat belajar mempelajari bahasa inggris. Hasil Penelitian menunjukan bahwa peneliti berhasil membuat sebuah aplikasi yang akan membuat anak-anak sekolah dasar senang ketika belajar dan akan menambah kosa kata bahasa inggris yang mereka ketahui,

Kata kunci: Game Edukasi, Kosa-Kata, Bahasa Inggris, Sekolah Dasar,

ABSTRACT

Along with most elementary schools still applying the learning method using the guidebook media, as well as the learning methods at the State Elementary Schools, while schools still often use the cerama method, therefore students are only spectators, hence, the author took the initiative to make a game application education for learning in children in English. The advantage of this game is that it is run offline and can make the child easy to take lessons without having to wait for the teacher and motivate the child to actively study English. The results showed that the researchers succeeded in making an application that would make elementary school children happy when studying and would add to the English vocabulary they knew,

Keywords: Educational Games, Vocabulary, English, Elementary School,

KATA PENGANTAR

Alhamdulillah, penulis dapat menyelesaikan usulan penelitian ini dengan judul “ ***Game Petualang Edukasi Bahasa Inggris Pada Anak Sekolah Dasar*** ” untuk memenuhi salah satu syarat penyusunan Skripsi Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Penulis menyadari sepenuhnya bahwa usulan penelitian ini tidak mungkin terwujud tanpa bantuan dan dorongan dari berbagai pihak, baik bantuan moril maupun materil. Untuk itu, dengan segala keikhlasan dan kerendahan hati, penulis mengucapkan banyak terima kasih dan penghargaan yang setinggi-tingginya kepada:

1. Muhammad Ichsan Gaffar, S.E, M.Ak, selaku Ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Ichsan Gorontalo;
2. DR. Abdul Gaffar La Tjokke, M.Si, selaku Rektor Universitas Ichsan Gorontalo;
3. Zohrahayaty, S.Kom, M.Kom, selaku Dekan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
4. Sudirman S. Panna, S.Kom, M.Kom, selaku Wakil Dekan I Bidang Akademik Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
5. Irma Surya Kumala Idris, M.Kom, selaku Wakil Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
6. Sudirman Melangi, M.Kom, selaku Wakil Dekan III Bidang Kemahasiswaan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
7. Irvan A Salihi, M.Kom, selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo;
8. Irvan A Salihi, M.Kom, selaku Pembimbing I, yang selalu membantu atau membimbing penulis untuk menyelesaikan proposal penelitian;
9. Muis Nanja, M.Kom, selaku Pembimbing II, yang selalu membantu atau membimbing penulis untuk menyelesaikan proposal penelitian;
10. Bapak dan Ibu Dosen Universitas Ichsan Gorontalo yang telah mendidik dan mengajarkan berbagai disiplin ilmu kepada penulis;

11. Kedua Orang Tua saya yang tercinta, atas segala kasih sayang, jerih payah dan doa restunya dalam membesarkan dan mendidik penulis, serta kakak, adik dan keluarga penulis yang selalu menemani dalam suka maupun duka;
12. Rekan-rekan seperjuangan Nurlaila, Rizki Malahedi, Rifandi, Rizki Mada, Fatma, Serliani, Bella, Nurfadillah, Shandy, Dana, Nurain yang telah banyak memberikan bantuan dan dukungan moril yang sangat besar kepada penulis;
13. Kepada semua pihak yang ikut membantu dalam penyelesaian proposal ini yang tak sempat penulis sebutkan satu-persatu.

Semoga Allah, SWT melimpahkan balasan atas jasa-jasa mereka kepada kami. Penulis menyadari sepenuhnya bahwa apa yang telah dicapai ini masih jauh dari kesempurnaan dan masih banyak terdapat kekurangan. Oleh karena itu, penulis sangat mengharapkan adanya kritik dan saran yang konstruktif. Akhirnya penulis berharap semoga hasil yang telah dicapai ini dapat bermanfaat bagi kita semua, Aamiin

Gorontalo, Juli 2020

Penulis

DAFTAR ISI

PENGESAHAN SKRIPSI.....	iii
PERSETUJUAN SKRIPSI.....	iv
PERNYATAAN SKRIPSI	v
ABSTRAK.....	vi
<i>ABSTRACT</i>	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL	xiv
LAMPIRAN	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	3
1.3 Rumusan Masalah	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.5.1 Manfaat Teoritis.....	4
1.5.2 Manfaat Praktis.....	4
BAB II LANDASAN TEORI.....	5
2.1 Tinjauan Studi	5
2.2 Tinjauan Pustaka	7
2.2.1 Definisi Game.....	7
2.2.2 Model Pembelajaran Bahasa Inggris.....	10
2.2.3 Unified Modelling Language (UML).....	16

2.2.4 Pengujian Sistem	21
2.3 Kerangka Pikir.....	22
BAB III METODE PENELITIAN	23
3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian.....	23
3.2 Pengumpulan Data	23
3.2.1 Observasi	23
3.2.2 Study Literatur	23
3.3 Pengembangan Game	24
3.3.1 Specification	24
3.3.2 Functional Modelling, Menggunakan Alat Bantu UML	24
3.3.3 Pengujian Sistem	24
BAB IV HASIL PENELITIAN	26
4.1 Hasil Pengumpulan Data	26
4.2 Hasil Pembangunan/ Perancangan Game	26
4.2.1 Menyiapkan Desain	27
4.2.2 Menyiapkan Musik Dan Sound.....	27
4.2.3 Membuat Permainan	27
4.3 Hasil Pengembangan Game.....	27
4.2.1 Aktor.....	27
4.2.2 Hewan Dan Buah-Buahan.....	28
4.2.3 Enemy.....	28
4.2.4 Story Board.....	29
4.2.5 Specification	37
4.2.6 Functional Modelling, Menggunakan Alat Bantu UML	37
4.2.7 Pengujian Sistem	39

BAB V PEMBAHASAN.....	42
5.1 Pembahasan Model.....	42
5.1.1 Rancangan Permainan Petualang	42
5.2 Pembahasan Sistem	42
5.2.1 Menu Utama	42
5.2.2 Halaman Sub Level.....	43
BAB VI PENUTUP	50
6.1 Kesimpulan	50
6.2 Saran	50
DAFTAR PUSTAKA	51
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2.1: Model Pembelajaran 1.....	12
Gambar 2.2: Latihan Medel Pembelajaran 1.....	13
Gambar 2.3: Model Pembelajaran 2.....	14
Gambar 2.4: Latihan Model Pembelajaran 2	15
Gambar 2.5: Kerangka Pikir	22
Gambar 4.1: Struktur Jalannya Game	26
Gambar 4.2: Aktor Berjalan.....	27
Gambar 4.3: Desain Hewan Yang Ada Pada Game.....	28
Gambar 4.4: Desain Buah Yang Ada Pada Game.....	28
Gambar 4.5: Karakter Enemy	29
Gambar 4.6: Use Case Diagram Aplikasi	37
Gambar 4.7: Activity Diagram Menu	38
Gambar 4.8: Flowchart Pengujian White Box	40
Gambar 4.9: Flowgraph Pengujian White Box	40
Gambar 4.10: Black Box Testing.....	41
Gambar 5.1: Tampilan Menu Utama	43
Gambar 5.2: Tampilan Level 1	44
Gambar 5.3: Tampilan Level 2	44
Gambar 5.4: Tampilan Level 3	45
Gambar 5.5: Tampilan Level 4	46
Gambar 5.6: Tampilan Level 5	46
Gambar 5.7: Tampilan Menu Aturan Permainan.....	47
Gambar 5.8: Tampilan Menu Tentang Aplikasi	47
Gambar 5.9: Tampilan Nilai Tertinggi.....	48
Gambar 5.10: Tampilan Quis Perlevel.....	48
Gambar 5.11: Tampilan Quis Akhir.....	49

DAFTAR TABEL

Tabel 2.1: Tinjauan Studi	5
Tabel 2.2: Simbol Use Case Diagram	16
Tabel 2.3: Simbol Class Diagram	17
Tabel 2.4: Multipliciti Class Diagram.....	18
Tabel 2.5: Simbol Activity Diagram	19
Tabel 2.6: Simbol Sequence Diagram.....	20
Tabel 4.1: Tabel Storyboard Untuk Bagian Menu	29
Tabel 4.2: Tabel Storyboard Untuk Level.....	29
Tabel 4.3: Tabel Storyboard Untuk Animasi Level	33
Tabel 4.4: Tabel Storyboard Untuk Aturan Bermain.....	35
Tabel 4.5: Tabel Storyboard Untuk Nilai Tertinggi.....	36
Tabel 4.6: Tabel Storyboard Untuk Tentang Aplikasi	36

LAMPIRAN

Lampiran 1: Kode Program.....	53
Lampiran 2: Permohonan Izin Penelitian.....	89
Lampiran 3: Surat Keterangan Telah Melakukan Penelitian	90
Lampiran 4: Bukti Penerimaan Softcopy Skripsi.....	91
Lampiran 5: Surat Pernyataan Pengecekan Kembali	92
Lampiran 6: Surat Rekomendasi Bebas Plagiasi	93
Lampiran 7: Surat Keterangan Bebas Pustaka	94
Lampiran 8: Biografi Penulis	95

BAB I

PENDAHULUAN

1.1 Latar Belakang

Di Indonesia teknologi berpengaruh besar terhadap proses pembelajaran di sekolah dasar dan berhubungan penuh juga terhadap materi pembelajaran dan juga cara penyampaian materi di dalam pembelajaran, pada suatu lingkup pendidikan pada anak sekolah dasar, anak didik akan lebih tertarik pada suatu permainan yang mudah dipahami dan menarik, yang di dalamnya terdapat gambar, suara, maupun animasi yang menarik, pada langkah ini anak didik akan di sajikan agar dapat mempermudah mengingat sebuah bentuk atau tulisan[1].

Tetapi yang kita ketahui kebanyakan sekolah dasar masih menerapkan metode pembelajaran menggunakan media buku panduan, begitu pula dengan metode belajar pada Sekolah Dasar Negeri, adapun sekolah-sekolah masih sering menggunakan metode cerama maka dari itu siswa hanya jadi penonton saja. Pembelajaran bahasa inggris pada sekolah dasar ini meliputi pengenalan dasar bahasa inggris, dengan materi dasar pengenalan huruf dan membaca yang ada di lingkungan sekitar sesuai kompetensi dasar pada mata pelajaran bahasa inggris[1][2].

Madrasah Ibtidaiyah (MI) Al Islam Program Plus Dibal merupakan salah satu dari sekian sekolah dasar yang ada di Indonesia yang memberikan mata pelajaran Bahasa Inggris untuk siswanya. Bahasa Inggris merupakan bahasa yang dipakai sebagai bahasa internasional saat ini dan berguna untuk siswa yang akan menuntut ilmu diluar Negeri nanti, bahasa Inggris juga termasuk dalam salah satu mata pelajaran wajib yang di terapkan di indonesia saat ini. Materi yang diberikan salah satunya adalah tentang pengenalan kata-kata dalam bahasa inggris sesuai kurikulum 2013. Pada saat kegiatan belajar mengajar berlangsung di kelas banyak kita jumpai, daya tangkap beberapa siswa dalam mengingat kosakata masih lemah serta rendahnya kemauan siswa untuk belajar karena masih menerapkan metode pembelajaran menggunakan media buku panduan, adapun kita sering menjumpai

menggunakan metode cerama dan ketika diberikan latihan pasti akan berbalik pada buku lagi[1][2][3]. Penulis berpendapat bahwa game edukasi petualang dapat digunakan sebagai media pembelajaran yang menarik dan menyenangkan.

Dengan penerapan metode karya wisata pada model pembelajaran dapat menciptakan kegiatan belajar yang menyenangkan, seperti membawa siswa keluar kelas. Pada metode karya wisata ini, titik utama pembelajaran adalah dengan penggalian kreatifitas siswa dan pengaitan antara apa saja yang di pelajari di sekolah dengan lebih nyata. Pembelajaran di luar diluar kelas ini dapat membuat siswa memiliki pengalaman tersendiri[4]. Metode karya wisata mempunyai kelebihan yaitu memiliki tujuan pembelajaran modern yang memiliki manfaat menggunakan lingkungan nyata, Membuat sesuatu yang di pelajari lebih tertuju pada kenyataan, Dapat melatih memunculkan kreativitas siswa, terdapat informasi yang lebih luas dan benar nyata. Serta kekurangannya adalah Sulitnya fasilitas yang harus disediakan siswa atau sekolah, memerlukan persiapan dan perencanaan, harus memiliki koordinasi dengan guru serta bidang studi yang lain, Sering unsur rekreasi menjadi prioritas dari pada tujuan utama, Sulit mengontrol siswa yang banyak[5].

Penulis telah mendapat inisiatif untuk membuat suatu aplikasi game edukasi untuk suatu pembelajaran pada anak-anak dalam bahasa inggris. Kelebihan dari game ini adalah dijalankan dalam keadaan *offline* dan dapat membuat si anak mudah untuk mengikuti pelajaran tanpa harus menunggu guru dan memotivasi si anak untuk giat belajar mempelajari bahasa inggris[6]. Untuk itu, perencanaan dan persiapan lingkungan belajar anak harus dirancang dengan seksama sehingga segala sesuatu dapat merupakan kesempatan belajar yang sangat menyenangkan bagi anak itu sendiri[7]. Pembelajaran bahasa inggris bagi anak – anak memerlukan banyak gambar, permainan, relia, tape recorder, video (CD / DV), dan alat bantu lainnya agar anak dapat mudah mengingat apa yang ia lihat, dan mudah mengerti apa yang kita ajarkan[7].

Dalam penyusunan proposal skripsi ini yang menjadi pembahasan utama adalah game edukasi yang ditujukan untuk siswa SD, yang bertujuan untuk memberikan sistem pembelajaran model baru berupa permainan. Berdasarkan latar belakang inilah penulis tertarik untuk mengambil judul skripsi “ **Game**

Petualang Edukasi Bahasa Inggris Pada Anak Sekolah Dasar “. Dengan menggunakan media game sebagai bahan pembelajaran tentunya akan bisa menarik minat siswa dan siswi dalam menerima materi pembelajaran yang diberikan dengan baik.

1.2 Identifikasi Masalah

Berdasarkan uraian dari latar belakang yang di paparkan pada halaman sebelumnya, dapat diidentifikasi masalah sebagai berikut :

Daya tangkap beberapa siswa dalam mengingat kosakata masih lemah serta rendahnya kemauan siswa untuk belajar karena masih menerapkan metode pembelajaran menggunakan media buku panduan.

Dengan menerapkan model pembelajaran karyawisata terlalu banyak kendala seperti sulitnya fasilitas yang harus disediakan siswa atau sekolah, memerlukan persiapan dan perencanaan, harus memiliki koordinasi dengan guru serta bidang studi yang lain, Sering unsur rekreasi menjadi prioritas dari pada tujuan utama, Sulit mengontrol siswa yang banyak

1.3 Rumusan Masalah

Berdasarkan uraian dari latar belakang diatas, dapat dirumuskan masalah sebagai berikut :

1. Bagaimana Merancang game petualang edukasi bahasa inggris agar dapat meningkatkan minat belajar anak-anak SD untuk lebih mengenal kosa kata bahasa inggris??
2. Bagaimana cara menerapkan model pembelajaran karyawisata pada pembelajaran bahasa inggris??

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah :

1. Untuk menghasilkan aplikasi pembelajaran bahasa Inggris pada anak sekolah dasar untuk lebih mengenal kosa kata.

2. Agar siswa lebih muda untuk mengenal nama-nama hewan dan Buah-Buahan dalam Bahasa Inggris tanpa harus keluar rumah dan biaya banyak.

1.5 Manfaat Penelitian

Dampak dari Manfaat penelitian ini adalah :

1.5.1 Manfaat Teoritis

Memberikan masukan bagi perkembangan ilmu pengetahuan dan teknologi, khususnya pada bidang ilmu komputer.

1.5.2 Manfaat Praktis

Sumbangan pemikiran, karya, bahan pertimbangan, atau solusi bagi *game developer* guna mendukung pengambilan keputusan dalam rangka menghasilkan *game* yang berkualitas sehingga berdampak pula pada peningkatan minat belajar anak SD, pelajar, dan terutama bagi anak yang kurangnya pengetahuan tentang kosa kata bahasa inggris.

BAB II

LANDASAN TEORI

2.1 Tinjauan Studi

Adapun penelitian terdahulu yang berhubungan dengan topik peneliti untuk di jadikan sebagai bahan referensi dalam menggunakan yang di tentukan nantinya.

Tabel 2.1: Tinjauan Studi

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
1.	Marzuki, Fandri Chandra Program	Game Berbasis Adventur e Sebagai Penduku ng Pembelaj aran Pengenal an Kata Bahasa Inggris Untuk Anak Usia Dini	2009		Hasil penelitian yang dilakukan mulai dari tahap perancangan hingga pengujian terhadap aplikasi game berbasis adventure sebagai pendukung pembelajaran pengenalan kata bahasa inggris untuk anak usia dini, maka mendapat hasil, proses pembelajaran pengenalan kata bahasa inggris untuk anak usia dini akan lebih mudah dimengerti oleh anak – anak dan dapat menjadi sebuah alat bantu, jadi dengan adanya aplikasi game tersebut maka akan mempermudah dalam

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
					pembelajaran pengenalan kata bahasa inggris.
2.	F Y Al Irsyadi, R Annas <i>and</i> Y I Kurniawan	Game Edukasi Pembelaj aran Bahasa Inggris untuk Pengenal an Benda- Benda di Rumah bagi Siswa Kelas 4 Sekolah Dasar	2019		Dari hasil pengujian blackbox didapatkan bahwa aplikasi sudah berjalan dengan baik, sesuai dengan harapan developer. Dari hasil pengujian User Acceptance Test, rata-rata nilai 'Sangat Setuju' dari 5 pernyataan yang diberikan adalah 69.19% yang menunjukkan tingkat acceptance yang cukup tinggi dari pengguna terhadap aplikasi tersebut.
3.	Syifaul Fuada	Perancan gan Game Petualan gan Pramuka Berbasis Android		2016	Game ini memberikan manfaat khususnya guru dan siswa ataupun masyarakat luas yang terjangkau teknologi. Lebih jauh lagi dapat menjadikan pembelajaran pramuka dapat dilakukan dikelas ataupun luar kelasdengan tanpa bantuan

NO	PENELITI	JUDUL	TAHUN	METODE	HASIL
					dari guru. Penelitian ini menggunakan development model yang terdiri atas: concept, design, material collecting, assembly, testing dan distribution. Makalah ini merupakan studi awal (pre-eliminary study) tentang di implementasikan rancangan permainan digital, sehingga konten sebatas dalam tahap konsep dan design

2.2 Tinjauan Pustaka

Dalam penelitian ini, tinjauan pustaka yang digunakan adalah teori – teori yang menjadi landasan dalam penelitian, selain itu kajian pustaka juga melalui jurnal – jurnal penelitian nasional.

2.2.1 Definisi Game

Game adalah sebuah aplikasi yang di dalamnya ada aturan dimana seorang pemain akan menghadapi sebuah masalah/konflik yang dibuat dalam permainan tersebut. Dan jika pemain mengikuti aturan tersebut pemain akan mendapatkan hasil/kemenangan. Sebuah *game* harus memiliki akhir yang bisa di capai oleh pemain, pada pelibatan pemain dalam mencapai akhir game pemain harus memilih Kontrol terhadap apa yang terjadi pada jalannya *game* tersebut[8].

Berdasarkan *genre* permainannya, game dapat dibagi menjadi beberapa kategori[8], yaitu:

- *Action – Shooting*: suatu game yang memerlukan kecepatan refleks, keseimbangan mata-tangan, ketepatan waktu.

- *Action – Adventure*: suatu game yang sudah memiliki kemajuan yang pesat, dan menjadi game campuran *action beat-em up*, dan saat ini sudah tersedia dalam bentuk visual 3D dan sudut pandang orang ke tiga.
- *Fighting* (pertarungan): suatu game yang mengutamakan penguasaan permainan, permainan karakter dan waktu, dan kerjasama antara mata-tangan.
- *Adventure*: suatu game yang mengikuti suatu kehidupan seperti aslinya atau hanya tiruan kehidupan sehari-hari.
- Simulasi Konstruksi dan manajemen: adalah suatu *game* yang seringkali menggambarkan keadaan dalam *game* sama dengan atau hampir sama dengan keadaan dunia nyata dengan memperhatikan detailnya
- *Role Playing*: sebuah *game* yang cenderung bermain peran, memiliki penekanan pada tokoh atau peran perwakilan pada pemain pada saat *game* berjalan.
- Strategi: Sebuah *game* yang mengutamakan sebuah kecerdasan atau logika yang terencana yang nantinya menjadi sebuah jalan keluar atau akhir dari *game*
- *Puzzle*: Sebuah game yang bertujuan untuk menyelesaikan sebuah teka-teki. Baik dalam bentuk balok, menyamakan warna bola, menyelesaikan perhitungan matematika, cara melewati labirin dan lain-lain.
- Simulasi kendaraan: sebuah *game* yang memberikan percobaan pada pemain, sebelum mencobanya yang asli. apa saja yang ada di dalam game menyerupai dengan kenyataannya.
- Olahraga: sebuah *game* yang di anggap realistis yang tinggi pada kehidupan pada saat berolahraga.

2.2.1.1 Game Edukasi

Sebuah *game* sering kali di salah artikan dengan pembawa dampak negatif terhadap anak. Tapi itu berbanding terbalik dengan faktanya, banyak *game* yang di ciptakan dengan membawa positif bagi anak, dengan game anak bisa mengenal teknologi komputer, pelajaran untuk mengikuti pengarahan dan aturan, latihan memecahkan masalah dan melatih logika, melatih saraf motorik dan

keterampilan spesial, menjalin komunikasi anak-orangtua ketika sedang bermain bersama, serta dapat memberikan hiburan bagi si pemain, bahkan bisa menjadi suatu ilmu terapi penyembuhan. Edukasi adalah proses dimana seseorang di belajarkan oleh alat atau guru, yang dilakukan harus di lakukan dengan cara belajar mengamati kemudian belajar terus diterapkan dalam kehidupan. Secara umum anak usia dini adalah anak yang memiliki usia 0-6 tahun. Pada usia dini inilah anak-anak sudah berada pada fase pertumbuhan dan perkembangan yang paling pesat. Baik fisik maupun mental[9].

Maka anak-anak sangat membutuhkan *game* yang memiliki konten pendidikan atau yang lebih dikenal game edukasi. *Game* ini dapat merangsang minat belajar anak-anak, sehingga anak-anak yang jenuh terhadap pelajaran, akan merasa lebih senang ketika bermain game ini. Dikarenakan selain bisa bermain anak-anak bisa belajar atau menambah ilmu yang di ajarkan pada game tersebut[10].

2.2.1.2 Game Petualang

Game adventure atau *game* petualang adalah suatu aplikasi perangkat lunak yang di sajikan menyerupai lingkungan asli yang mana pemain akan di tantang untuk berinteraksi dan memecahkan masalah agar bisa menyelesaikan permainan tersebut[7].

Game ini menuntut pemain lebih pada penyelesaian jalan cerita, *game* ini pula membutuhkan ketajaman analisis dan kekuatan hapalan dan pemecahan teka-teki, dan tidak melibatkan kecepatan refleks pemain. Pemain dapat memahami karakter pada setiap pemain dan fungsi dari benda-benda yang ada pada game[10].

2.2.1.3 Desain Dan Karakter Dalam Game

Karakter adalah salah satu hal yang penting dan harus ada dalam dunia *game*, karakter juga menjadi sebagai pelengkap dalam sebuah *game*, yang akan memberikan *game* bernyawa. Karakter dalam *game* dapat di bagi menjadi dua jenis yaitu *playing characters* dan *none playing characters* yang biasa disingkat NPC[11].

Dalam Desain dan Pembuatan karakter penulis menggunakan aplikasi Adobe Photoshop. Adobe Photoshop adalah suatu software/perangkat lunak buatan adobe systems yang di gunakan pengguna mengolah/mengedit/pembuatan efek pada sebuah gambar/citra. Sekarang perangkat ini telah banyak diminati oleh fotografer digital professional, desainer grafis dan perusahaan iklan[11].

Popularitas yang dihasilkan perusahaan yang termasuk jajaran perintis ini masih sulit disaingi sehingga adobe dapat di katakan sebagai market leader untuk perangkat lunak pengolah gambar atau foto[11].

Versi kedelapan aplikasi ini disebut dengan nama photoshop CS (creative suite), versi kesembilan disebut adobe photoshop CS2, versi kesepuluh disebut photoshop CS3, versi kesebelas adalah adobe photoshop CS4, versi keduabelas adalah adobe photoshop CS5 dan versi yang ketigabelas adalah adobe photoshop CS6, dan yang terbaru adalah Adobe photoshop CC[11].

2.2.2 Model Pembelajaran Bahasa Inggris

Bahasa Inggris termasuk salah satu hal berpengaruh dan harus di pelajari dari masa anak-anak, bahasa internasional ini sangat wajib bagi kita bagi yang ingin melanjutkan studi yang lebih tinggi, tepatnya yang bertaraf internasional, maka kita patut untuk membelajarkan bahasa ini kepada mereka[7].

Menurut Scott dan Ytreberg (1990:108) Cara yang utama menyampaikan makna dalam proses pembelajaran bahasa asing kepada anak-anak adalah melalui berbagai variasi alat Bantu pembelajaran. Pelajaran akan jauh lebih mudah dan lebih menarik bagi anak-anak jika guru dengan sepenuhnya memanfaatkan benda-benda atau objek serta bahasa untuk menyampaikan makna. Maka itu pelajaran akan dirasa mudah dan menarik bagi anak didik jika guru mau dan senantiasa memanfaatkan benda-benda[12].

2.2.2.1 Metode Karya Wisata

Metode Karyawisata adalah metode pembelajaran yang dilakukan dengan cara mengajak siswa didik agar dapat mengenal lingkungan luar sekolah dan

sekaligus mengamatinya. Dengan metode Karyawisata dapat membuat siswa didik merasa senang dan termotivasi dalam kegiatan belajar mengajar. Dengan penelitian yang dilakukan terhadap kegiatan pembelajaran, dapat ditemukan kurangnya kegiatan pembelajaran yang memotivasi peserta didik untuk lebih bersemangat dalam pembelajaran menulis. Hal ini disebabkan oleh, siswa didik tidak langsung mengamati dengan menggunakan objek nyata, sehingga materi dianggap sulit. Dengan itu solusi untuk mengatasi masalah tersebut dengan menerapkan metode karyawisata. Metode pembelajaran karyawisata dirancang dengan memiliki tujuan agar siswa didik dapat mengamati objek secara langsung, dapat belajar dengan nyaman dan menyenangkan[13].

Pada suatu penggunaan metode karya wisata pada siswa sekolah dasar dapat sepenuhnya membantu untuk meningkatkan ide-ide baru dan apresiatif mereka, karena yang sudah kita ketahui siswa sekolah dasar merupakan anak – anak yang memasuki tahap perkembangan kognitif “operasional konkrit”. Dalam kata lain cepat mengerti dengan adanya pengalaman[4].

Kelebihan Metode karyawisata adalah memiliki tujuan pembelajaran modern yang memiliki manfaat menggunakan lingkungan nyata, Membuat sesuatu yang di pelajari lebih tertuju pada kenyataan, Dapat melatih memunculkan kreativitas siswa, terdapat informasi yg lebih luas dan benar nyata. Serta kekurangannya adalah Sulitnya fasilitas yang harus disediakan siswa atau sekolah, memerlukan persiapan dan perencanaan, harus memiliki koordinasi dengan guru serta bidang studi yang lain, Sering unsur rekreasi menjadi prioritas dari pada tujuan utama, Sulit mengontrol siswa yang banyak[5].

2.2.2.2 Model Pembelajaran 1

Buah-Buahan dan Sayuran/*Fruits And Vegetables*

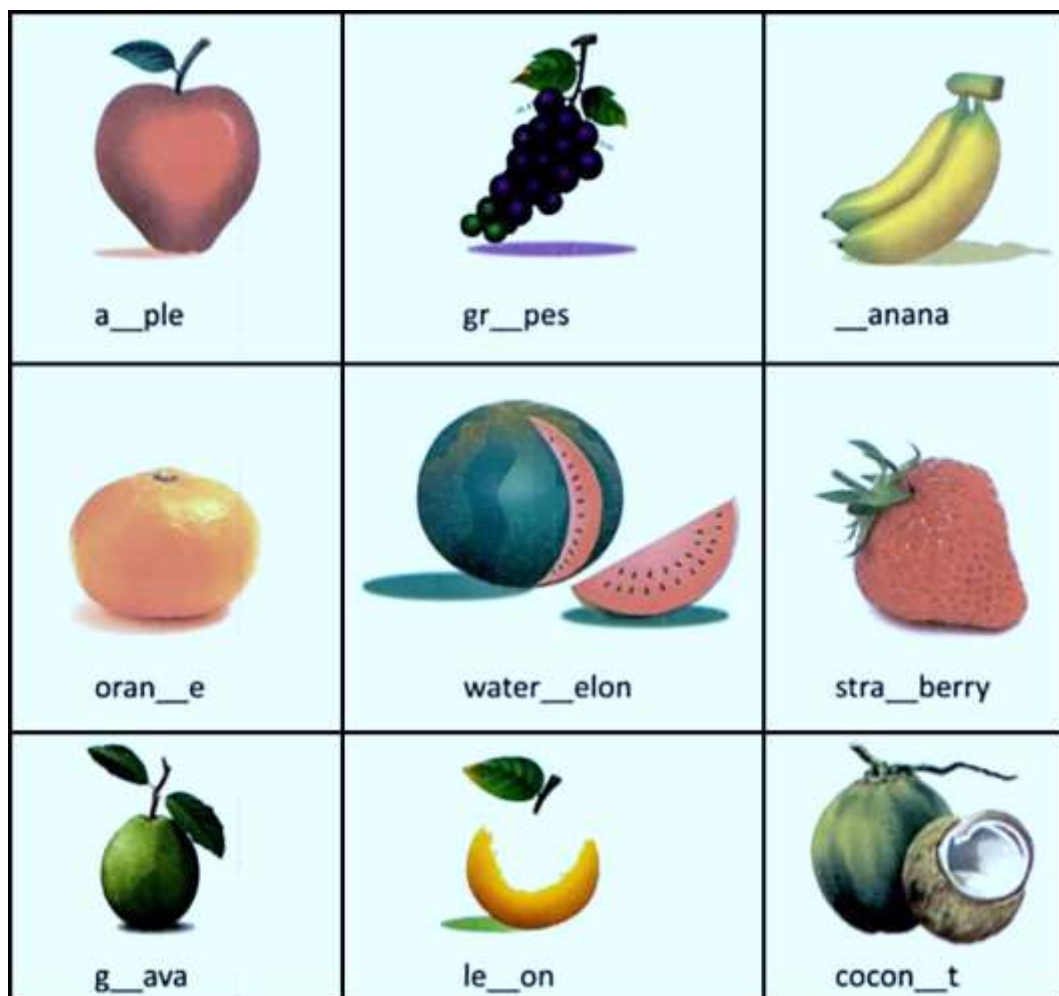
Model Pembelajaran Pertama ditunjukkan pada Gambar 2.1 Berikut ini.

 apple (apel)	 cherry (ceri)	 pear (pir)
 star fruit (belimbing)	 lemon (lemon)	 strawberry (stroberi)
 watermelon (semangka)	 orange (jeruk)	 guava (jambu)
 durian (durian)	 lychee (leci)	 mango (mangga)
 banana (pisang)	 grape (anggur)	 pineapple (nanas)

Gambar 2.1: Model Pembelajaran 1[14]

Latihan Model Pembelajaran 1/ *Exercise For Learning Model 1*

Isilah huruf-huruf yang hilang dalam kata-kata tentang buah/ *Fill in the missing letters in fruit words.*

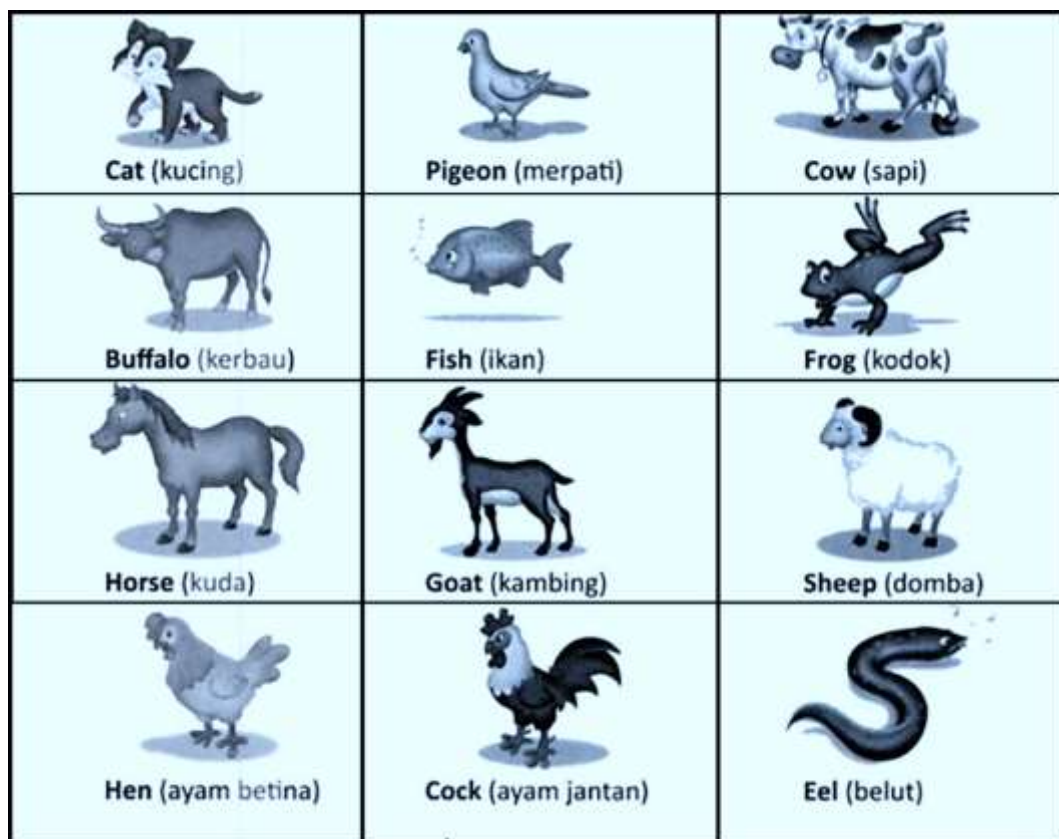


Gambar 2.2: Latihan Medel Pembelajaran 1[14]

2.2.2.3 Model Pembelajaran 2

Binatang/*Animals*

Model Pembelajaran ke 2 ditunjukkan pada Gambar 2.3 Berikut ini.



Gambar 2.3: Model Pembelajaran 2[14]

Latihan Untuk Model Pembelajaran 2/Exercise For Learning Model 2

Pilihlah jawaban yang benar dengan memberikan tanda silang pada huruf A, B, C, atau D!/Choose the correct answer by crossing (X) A, B, C, or D!

1.		These are my father's
	a. birds	c. cocks
	b. ducks	d. sheeps
2.		Yudha sees a at Taman Safari Zoo.
	a. hippopotamus	c. komodo
	b. giraffe	d. elephant
3.		I have long neck and small ears. I eat leaves, shrubs, and vines. I am a
	a. elephant	c. giraffe
	b. goat	d. panda
4.		 is pet animal.
	a. porcupine	c. goldfish
	b. squirrel	d. tiger
5.		 is wild animal.
	a. goat	c. buffalo
	b. cow	d. lion

Gambar 2.4: Latihan Model Pembelajaran 2[14]

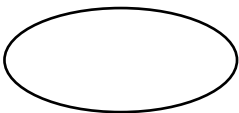
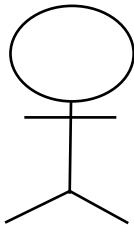
2.2.3 Unified Modelling Language (UML)

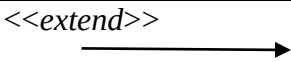
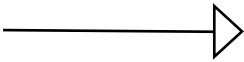
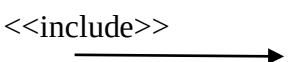
UML merupakan suatu metode atau proses pemodelan yang berbentuk visual sebagai fasilitas untuk menganalisa dan membuat *software* berorientasi objek. UML juga dapat di artikan termasuk bahasa visual untuk menjelaskan, memberikan rincian, mengonsep, membuat model, dan memaparkan aspek-aspek dari sebuah sistem[15]. Alat Bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

2.2.3.1 Use Case Diagram

Use Case Diagram adalah penggambaran dari suatu kegiatan atau interaksi antara aktor dengan aturan dan fungsionalitas dari sebuah aplikasi, sistem atau *game* [16].

Tabel 2.2: Simbol Use Case Diagram[17]

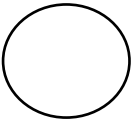
Gambar	Keterangan
	<i>Use Case</i> : Fungsionalitas yang disediakan pada sistem yang berfungsi sebagai penukar antara pesan antara unit atau aktor, biasanya digunakan menggunakan kata kerja pada awal <i>frase</i> nama <i>use case</i> .
	Aktor: Yang menjadi peran Orang, atau suatu sistem lain yang berinteraksi dengan sistem informasi atau alat ketika berkomunikasi dengan <i>use case</i> .
	Asosiasi: garis komunikasi penghubung antara aktor dan <i>use case</i> yang berperan terhadap <i>use case</i> atau <i>use case</i> yang bertatap muka langsung dengan aktor.

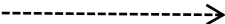
Gambar	Keterangan
	Ekstensi: relasi <i>use case</i> tambahan ke sebuah <i>use case</i> , pada <i>use case</i> yang di sdah di tambahkan akan mampu berdiri sendiri walauoun tanpa <i>use case</i> .
	Generalisasi: Generalasiasi dan spesialisasi dapat di artikan dangan umum dan khusus, yakni antara dua buah <i>use case</i> . Salah satu memiliki fungsi yang lebih umum dari lainnya.
	Include merupakan relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>include</i> merupakan kebalikan dari ekstensi yaitu memerlukan <i>use case</i> .

2.2.3.2 Class Diagram

Class Diagram adalah penggambaran dan deskripsi *class* dan hubungannya antara *class* [18].

Tabel 2.3: Simbol Class Diagram [17]

Simbol	Keterangan
	Kelas yang ada pada struktur sistem.
	Antarmuka/ <i>interface</i> : konsep interfacenya menyerupai dengan pemograman Berorientasi objek.
	Asosiasi/ <i>association</i> : Relasi antarkelas yang bermakna umum, asosiasi biasa di libatkan dengan multiplicity.

Simbol	Keterangan
	Asosiasi berarah/ <i>directed association</i> : relasi antarkelas yang memiliki arti salah satu kelas digunakan oleh kelas lainnya. Asosiasi biasanya melibatkan dengan menggunakan multiplicity.
	Generalisasi: Relasi antarkelas dengan bersifat umum atau khusus.
	Kebergantungan/ <i>dependency</i> : Relasi antarkelas dengan bersifat ketergantungan pada antarkelas.
	Agregasi/ <i>aggregation</i> : Relasi antarkelas dengan memiliki arti semua bagian.

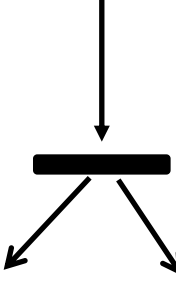
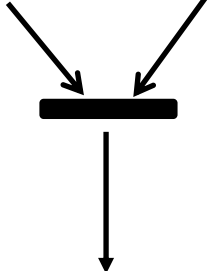
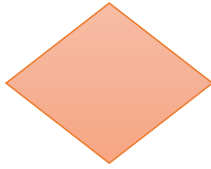
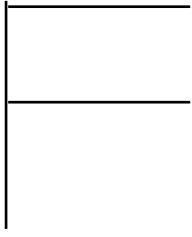
Tabel 2.4: Multipliciti Class Diagram[17]

Multiplicity	Penjelasan
1	Cuman Satu dan hanya satu
0..*	Bisa tidak ada, bisa 1 dan atau lebih
1..*	Satu atau lebih
0..1	Bisa tidak ada dan maksimal satu
n.n	Batasan antara. Contoh 3..5 mempunyai arti minimal 3 maksimal 5

2.2.3.3 Activity Diagram

Activity Diagram adalah suatu penggambaran dari berbagai aliran kerja atau aktivitas di dalam sistem, dari awal mulai, pemilihan dan mungkin sampai akhir *game/Aplikasi*[16].

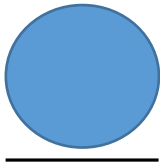
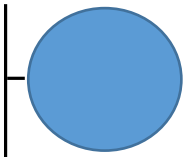
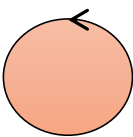
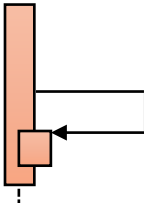
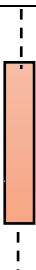
Tabel 2.5: Simbol Activity Diagram[19]

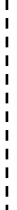
Gambar	Keterangan
	<i>Start Point</i> : terletak pada pojok kiri atas dan sebagai penanda mulainya aktivitas.
	<i>End Point</i> : sebagai penanda akhir aktivitas.
	<i>Activities</i> : suatu penggambaran dari kegiatan bisnis.
	<i>Fork/Percabangan</i> : di gunakan untuk sebagai penunjuk pada suatu kegiatan yang bersifat paralel atau sebagai penghubung atau sebagai penyatu dua kegiatan paralel.
	<i>Join/penggabungan</i> : di gunakan sebagai penunjuk adanya proses perubahan menjadi bentuk yang sederhana atau penguraian.
	<i>Decision Points</i> : Suatu penggambaran untuk memilih suatu keputusan dari dua pilihan yang pasti. <i>true</i> atau <i>false</i> .
	<i>Swimlane</i> : suatu pembagian <i>activity</i> diagram. Yang menunjukan semua pembagian tugas agar bisa di ketahui.

2.2.3.4 Sequence Diagram

Sequence Diagram adalah suatu penggambaran interaksi antara objek pada suatu *game*, dimana dalam Diagram *Sequence* tersebut kita dapat melihat langkah-langkah yang di lakukan dalam sistem. Yang dimana, apapun yang pemain lakukan akan di gambarkan yang pada akhirnya pemain dapat melihat nilai akhir[16].

Tabel 2.6: Simbol Sequence Diagram[19]

Gambar	Keterangan
	<i>Entity Class</i> : adalah sebuah bagian dari sistem yang di dalamnya ada kumpulan kelas berupa entitas-entitas, yang di gambarkan pada awal sistem serta bisa menjadi landasan untuk merangkai basis data.
	<i>Boundary Class</i> : adalah suatu tempat yang di dalamnya ada kumpulan-kumpulan kelas yang menjadi interaksi.
	<i>Control class</i> : adalah suatu tempat yang di dalamnya terdapat logika aplikasi yang tidak ada tanggung jawabnya terhadap entitas.
	<i>Messege</i> : adalah symbol garis yang menandakan pengiriman pesan terhadap class-class.
	<i>Recursive</i> : adalah suatu penggambaran atau simbolpada pengiriman pesan teruntuk dirinya.
	<i>Activation</i> : adalah sebagai perwakilan sebuah perjalanan operasi dari objek, sebagaimana yang kita harus tahu bahwa panjang kotak ini sama dengan panjang aktifitas sebuah operasi.

Gambar	Keterangan
	<i>Lifeline</i> : adalah sebuah garis putus-putus yang satu garis lurus dengan objek, pada setiap garis <i>lifeline</i> ada <i>activation</i> .

2.2.4 Pengujian Sistem

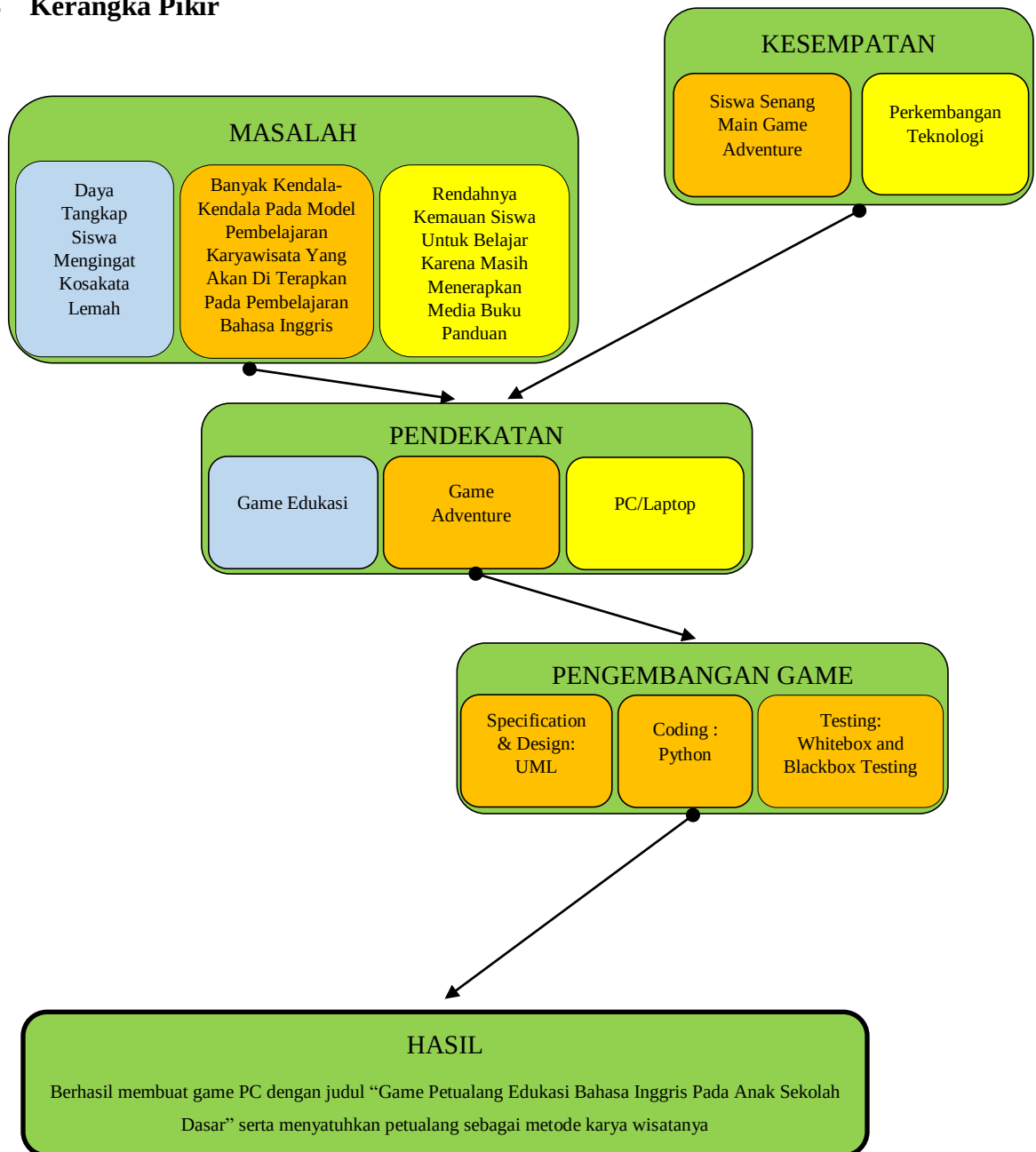
2.2.4.1 White Box Testing

White BoxTesting adalah suatu cara pengujian yang di lakukan pada suatu aplikasi atau *software* dengan cara melihat modul untuk dapat mengetahui kesalahan atau tidak pada koding dari program yang telah dibuat. Jika telah di temukan kesalahan maka akan di kompilasi ulang dan di cek lagi secara manual koding sehingga di perbaiki kembali[20].

2.2.4.2 Black Box Testing

Black BoxTesting adalah suatu pengujian yang tertuju pada spesifikasi fungsional dari perangkat lunak. Tester dapat mengartikan dari kondisi masukan dan mengetes spesifikasi fungsional pada program[20].

2.3 Kerangka Pikir



Gambar 2.5: Kerangka Pikir

BAB III

METODE PENELITIAN

3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian

Dalam Penelitian ini jika dipandang dari tingkat penerapannya, maka penelitian ini merupakan penelitian terapan.

Penelitian ini menggunakan metode penelitian eksperimen, dengan melakukan penilaian pengaruh terhadap aplikasi game terhadap siswa yang sedang belajar bahasa inggris. Dengan demikian jenis penelitian ini adalah penelitian eksperimental.

Subjek penelitian ini adalah dapat menambah pengetahuan kosa kata bahasa Inggris dengan menggunakan model pembelajaran metode karyawisata, serta membuat hati senang ketika belajar. pada objek Siswa Sekolah Dasar. Penelitian ini dimulai dari Oktober 2019 s/d Maret 2020 yang berlokasi pada SD Negeri 2 Tolangohula, Kabupaten Gorontalo, Provinsi Gorontalo.

3.2 Pengumpulan Data

3.2.1 Observasi

Studi lapangan (Observasi) adalah teknik pengumpulan data dengan terjun langsung ke lapangan dengan mengadakan suatu pengamatan dan menentukan suatu permasalahan yang ada pada tempat kejadian, di susun secara rapi kejadian-kejadian tersebut yaitu objek-objek yang yang perlu akan kita teliti agar dapat membantu atau mendukung penelitian yang sedang berlangsung.

3.2.2 Study Literatur

Pengumpulan data dengan cara mengumpulkan literature, paper, jurnal dan semua yang terkait dalam judul penelitian yang akan dibuat.

3.3 Pengembangan Game

3.3.1 Specification

Berdasarkan spesifikasi dan kapabilitas game yang dibuat, perangkat keras yang digunakan untuk membangunnya adalah sebagai berikut :

- CPU dengan kecepatan 3.00 GHz
- Kapasitas Harddisk 100 GB
- RAM 4 GB
- Monitor dengan resolusi 1366 x 768p
- Keyboard
- Mouse

3.3.2 Functional Modelling, Menggunakan Alat Bantu UML

3.3.2.1 Dalam Bentuk Use Case Diagram

Dalam penggambaran atau desain suatu rancangan game, penulis menggunakan Alat bantu UML dalam bentuk Use Case Diagram, untuk mengetahui kegiatan antara aktor dan fungsi.

3.3.2.1 Dalam Bentuk Actifity Diagram

Adapun penulis juga menggunakan penggambaran dalam bentuk Actifity Diagram yang menggambarkan suatu aktifitas dari mulai game, pemilihan, dan sampai akhir game.

3.3.3 Pengujian Sistem

Adapun dalam pengujian suatu sistem menggunakan 2 pengujian yaitu sebagai berikut:

3.3.3.1 White Box Testing

Pengujian yang di lakukan pada penelitian ini salah satunya menggunakan metode white box testing dengan cara melakukan pengecekan pada koding yang di buat agar jika terdapat kesalahan maka si penulis langsung mengubahnya.

3.3.3.2 Black Box Testing

Setelah itu melakukan pengujian menggunakan metode black box testing yakni si penulis pengujian lebih pada detailnya lagi seperti interface agar di lebih relevan agar tidak membuat ketidaksukaan pada pemakai, fungsi-fungsi di dalam aplikasi dan kesesuaian alur fungsi terhadap apa yg ingin di buat.

BAB IV

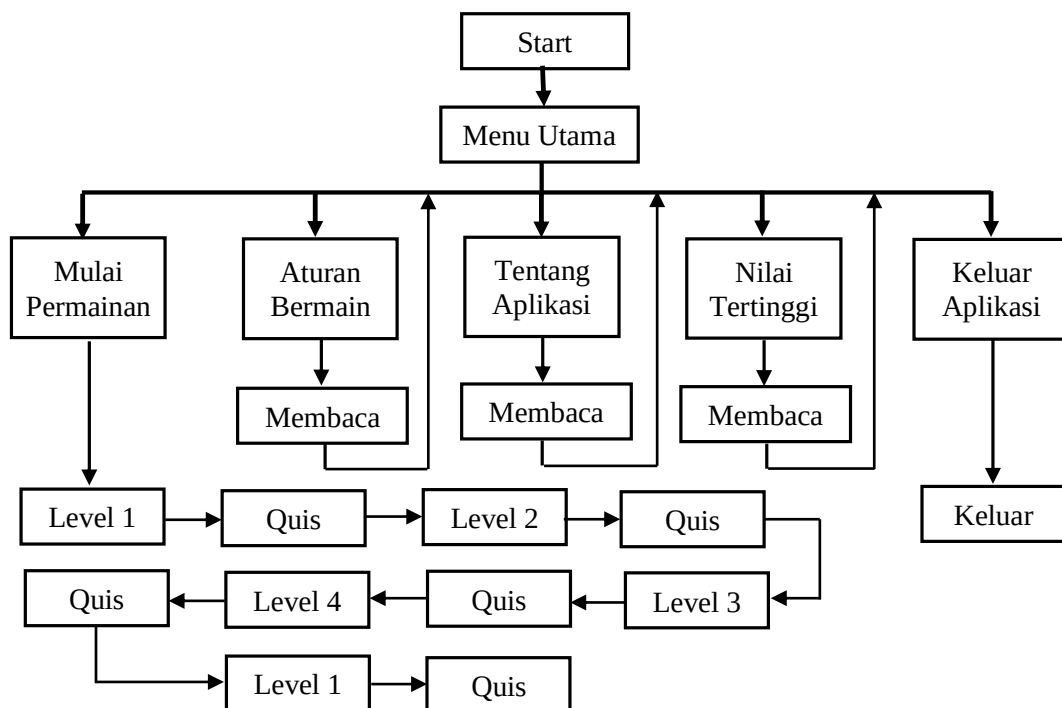
HASIL PENELITIAN

4.1 Hasil Pengumpulan Data

Berdasarkan hasil yang di dapatkan dari pemikiran diatas, penulis berhasil mengumpulkan data-data yang diperlukan termasuk untuk pembuatan *game* petualang edukasi dan program-program pendukung. Adapun bahan-bahan yang penulis persiapkan dapat berupa gambar-gambar hewan dan buah-buahan yang nantinya akan di pakai pada *game*, dan kosa-kata dari hewan dan buah-buahan tersebut

4.2 Hasil Pembangunan/ Perancangan Game

Struktur aplikasi *game* petualang yang di rancang bisa dilihat pada gambar berikut:



Gambar 4.1: Struktur Jalannya Game

Awalnya pemain akan masuk pada Menu Utama, akan disajikan 6 menu yang memiliki fungsi berbeda-beda, selanjutnya jika pemain membuka menu aturan

bermain terlebih dahulu untuk mengetahui aturan-aturannya, setelah itu menekan kembali untuk kembali ke Menu Utama, selanjutnya Klik Mulai Permainan.

4.2.1 Menyiapkan Desain

Guna untuk membuat sebuah game ini, penulis menyiapkan kurang lebih 116 macam gambar atau desain yang berbeda-beda, penulis membuat dan mengedit gambar tersebut menggunakan aplikasi Adobe Photoshop CC2017.

4.2.2 Menyiapkan Musik Dan Sound

Untuk menambah menarik dari *game* yang di buat penulis menyertakan musik dan *sound* pada game tersebut, untuk *sound* yang di masukan pada saat objek muncul penulis menggunakan suara dari google voice yang rekam lalu di potong menggunakan aplikasi MP3 Cutter dan Ringtone Maker, serta di konversikan menjadi format file wav yang mendukung aplikasi python menggunakan aplikasi MP3 To Wav Converter.

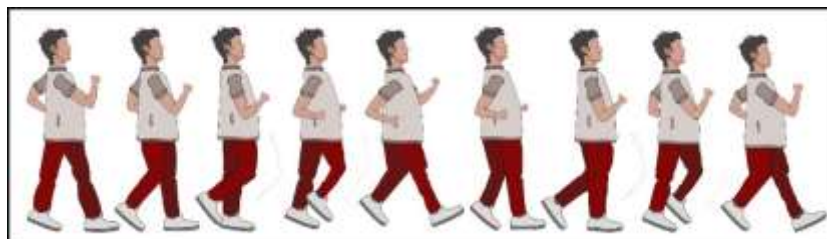
4.2.3 Membuat Permainan

Setelah persiapan selesai, penulis memulai pembuatan *Game* dengan menggunakan aplikasi WinPython64-3.6.7.0Qt5

4.3 Hasil Pengembangan Game

4.2.1 Aktor

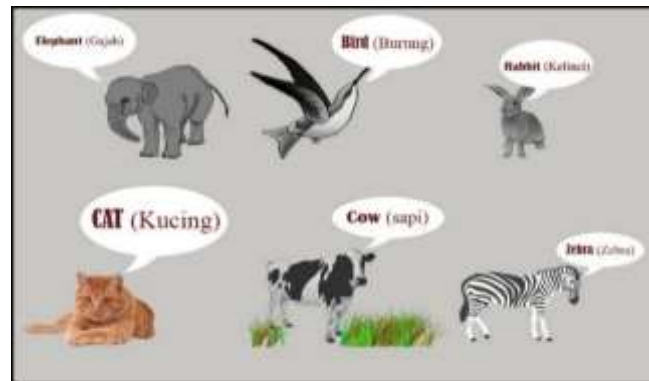
Aktor yang di pakai dalam game ini adalah seorang anak muda laki-laki yang berfungsi untuk dijalankan dan mengatur jalannya permainan, peneliti menggunakan 9 macam pergerakan. Cara menggerakkannya dengan cara menekan tombol kekiri atau kekanan, dan untuk melompat dengan cara menekan tombol Space. Untuk lebih jelas desainya bisa dilihat pada gambar berikut ini :



Gambar 4.2: Aktor Berjalan

4.2.2 Hewan Dan Buah-Buahan

Dalam sebuah permainan pemain nantinya akan di pertemukan dengan beberapa objek dapat berupa hewan dan buah-buahan, objek-objek tersebut yang nantinya akan jadi pelajaran buat pemain, Untuk desainnya bisa dilihat pada gambar berikut ini :



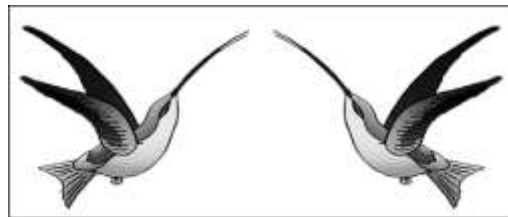
Gambar 4.3: Desain Hewan Yang Ada Pada Game



Gambar 4.4: Desain Buah Yang Ada Pada Game

4.2.3 Enemy

Karakter ini di buat berbentuk burung yang kan muncul pada level 3 dan 4 di pertengahan permainan, karakter enemy ini akan sewaktu-waktu akan menyerang, serangan dari burung ini akan mengakibatkan score anda berkurang 5, solusi utama, dengan cara melumpuhkan kerakter tersebut, yakni menembak kerakter tersebut, dengan cara menekan tombol (t) ketika menembak score bertambah 1, untuk melihat karakter tersebut bisa dilihat pada gambar tersebut:



Gambar 4.5: Karakter Enemy

4.2.4 Story Board

Tabel 4.1: Tabel *Storyboard* Untuk Bagian Menu

Visual	Sketsa	Audio
Keterangan: Tampilan sebuah menu yang menyajikan pilihan yang berbeda-beda	 1. Mulai Permainan 2. Aturan Bermain 3. Nilai Tertinggi 4. Tentang Aplikasi 5. Keluar Aplikasi	Suara: Klik.wav (Jika menu dipilih)

Tabel 4.2: Tabel *Storyboard* Untuk Level

Visual	Sketsa	Audio
Keterangan : Menampilkan halaman level untuk mengetahui apa-apa saja yang pemain akan lakukan jika akan menyelesaikan pada level 1	 Level 1 Tahap 1: Berjalan Tahap 2: Melompat (Menghindari Lubang) Tahap 3: Melompat (Melewati Tembok) Tahap 4: Mencermati kata disetiap Yang di lewati Tahap 5: Menjawab Soal Pertanyaan	Musik: music1.wav Suara: detik.wav (jika 20 lagi waktu habis) lose.wav (jika kalah) win.wav (jika mendapatkan peti quis) lemon.wav (lemon ditampilkan)

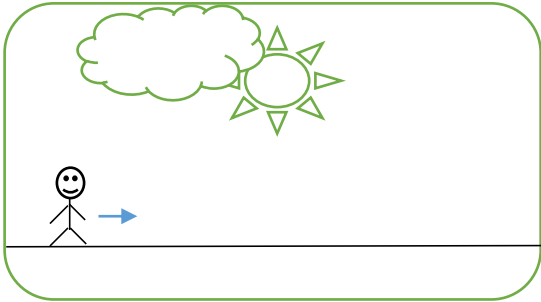
Visual	Sketsa	Audio
		pisang.wav (pisang ditampilkan) kelinci.wav (kelinci ditampilkan) kucing.wav (kucing ditampilkan)
Keterangan : Menampilkan halaman level yang menjelaskan apa yang harus dilakukan oleh pemain untuk dapat menyelesaikan permainan pada level 2	Level 2 Tahap 1: Berjalan Tahap 2: Melompat (Menghindari Lubang) Tahap 3: Melompat (Melewati Tembok) Tahap 4: Mencermati kata disetiap Yang di lewati Tahap 5: Menjawab Soal Pertanyaan 	Musik: music1.wav Suara: detik.wav (jika 20 lagi waktu habis) lose.wav (jika kalah) win.wav (jika mendapatkan peti quis) apel.wav (apel ditampilkan) burung.wav (burung ditampilkan) sapi.wav (sapi ditampilkan) kelapa.wav (kelapa ditampilkan)
Keterangan : Menampilkan halaman level yang menjelaskan apa yang harus dilakukan oleh pemain untuk	Level 3 Tahap 1: Berjalan Tahap 2: Melompat (Menghindari Lubang) Tahap 3: Melompat (Melewati Tembok) Tahap 4: Mencermati kata disetiap Yang di lewati Tahap 5: Melumpuhkan Musuh Tahap 6: Menjawab Soal Pertanyaan 	Musik: music.wav Suara: detik.wav (jika 20 lagi waktu habis) lose.wav (jika kalah)

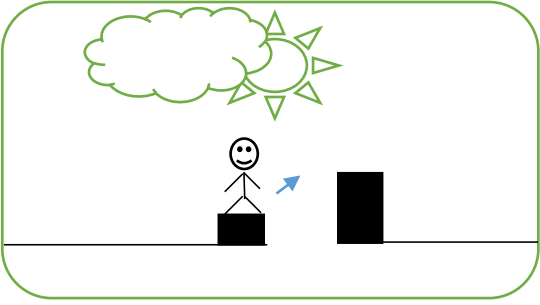
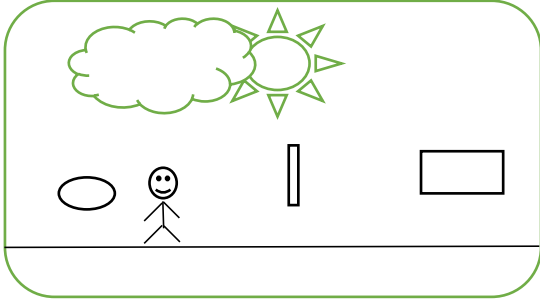
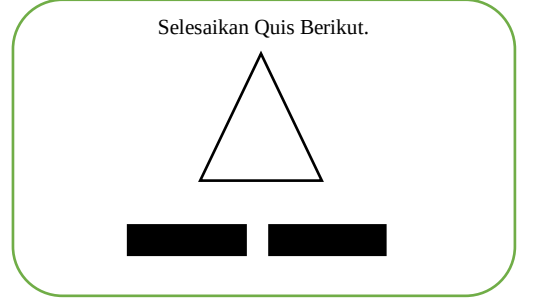
Visual	Sketsa	Audio
dapat menyelesaikan permainan pada level 3		win.wav (jika mendapatkan peti quis) ceri.wav (ceri ditampilkan) anggur.wav (anggur ditampilkan) monyet.wav (monyet ditampilkan) zebra.wav (zebra ditampilkan)
Keterangan : Menampilkan halaman level yang menjelaskan apa yang harus dilakukan oleh pemain untuk	Level 4 Tahap 1: Berjalan Tahap 2: Melompat (Menghindari Lubang) Tahap 3: Melompat (Melewati Tembok) Tahap 4: Mencermati kata disetiap yang di lewati Tahap 5: Melumpuhkan Musuh Tahap 6: Menjawab Soal Pertanyaan 	Musik: music.wav Suara: detik.wav (jika 20 lagi waktu habis) lose.wav (jika kalah)
dapat menyelesaikan permainan pada level 4		win.wav (jika mendapatkan peti quis) mangga.wav (mangga ditampilkan) semangka.wav (semangka ditampilkan) singa.wav (singa ditampilkan) jerapah.wav (jerapah ditampilkan)

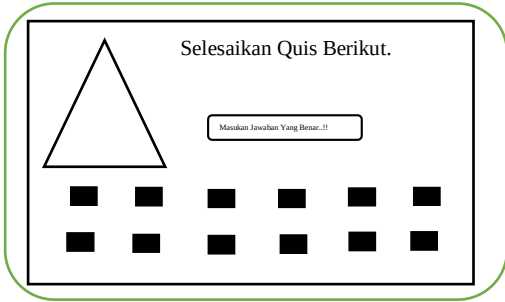
Visual	Sketsa	Audio
Keterangan : Menampilkan halaman level yang menjelaskan apa yang harus dilakukan oleh pemain untuk dapat menyelesaikan permainan pada level 5	 Level 5 Tahap 1: Berjalan Tahap 2: Melompat (Menghindari Lubang) Tahap 3: Melompat (Melewati Tembok) Tahap 4: Mencermati kata disetiap Yang di lewati Tahap 5: Menjawab Soal Pertanyaan	Musik: music.wav Suara: detik.wav (jika 20 lagi waktu habis) lose.wav (jika kalah) win.wav (jika mendapatkan peti quis) lemon.wav (lemon ditampilkan) apel.wav (apel ditampilkan) pisang.wav (pisang ditampilkan) kelinci.wav (kelinci ditampilkan) kucing.wav (kucing ditampilkan) sapi.wav (sapi ditampilkan) kelapa.wav (kelapa ditampilkan) burung.wav (burung ditampilkan) ceri.wav (ceri ditampilkan) anggur.wav (anggur ditampilkan) monyet.wav (monyet ditampilkan) zebra.wav (zebra ditampilkan)

Visual	Sketsa	Audio
		mangga.wav (mangga ditampilkan) semangka.wav (semangka ditampilkan) singa.wav (singa ditampilkan) jerapah.wav (jerapah ditampilkan) stroberi.wav (stroberi ditampilkan) nanas.wav (nanas ditampilkan) buaya.wav (buaya ditampilkan) gajah.wav (gajah ditampilkan)

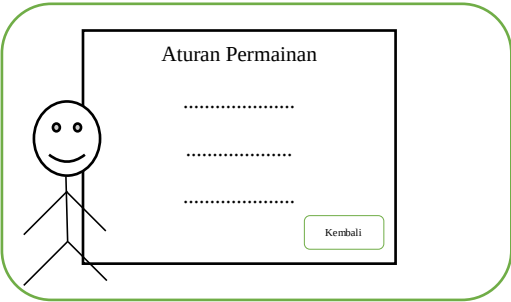
Tabel 4.3: Tabel *Storyboard* Untuk Animasi Level

Visual	Sketsa	Audio
Keterangan animasi : Menampilkan halaman sublevel berisi tutorial berjalan		Musik: music1.wav (lv 1,2,5) music.wav (lv 3,4)

Visual	Sketsa	Audio
Keterangan animasi : Menampilkan halaman sublevel berisi tutorial melompat menghindari lubang		Musik: music1.wav (lv 1,2,5) music.wav (lv 3,4)
Keterangan animasi : Menampilkan halaman sublevel berisi tutorial mengamati Kosakata		Musik: music1.wav (lv 1,2,5) music.wav (lv 3,4) Suara: Mengikuti objek yang di tampilkan
Keterangan animasi : Menampilkan halaman sublevel berisi tampilan quis		Musik: mengikuti level

Visual	Sketsa	Audio
Keterangan animasi : Menampilkan halaman sublevel berisi tampilan depan quis dan tutorial Menjawab soal Pertanyaan		Musik: music1.wav (lv 1,2,5) music.wav (lv 3,4) Suara: lose.wav (jika kalah) win.wav (jika menang) Clik.wav (Jika tombol di tekan)

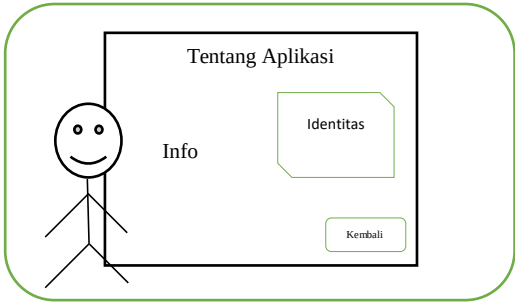
Tabel 4.4: Tabel *Storyboard* Untuk Aturan Bermain

Visual	Sketsa	Audio
Keterangan animasi : Menampilkan halaman menu aturan permainan berisi tutorial ketika bermain game beserta aturan-aturanya.		

Tabel 4.5: Tabel *Storyboard* Untuk Nilai Tertinggi

Visual	Sketsa	Audio
Keterangan animasi : Menampilkan halaman menu nilai tertinggi permainan berisi suatu hal ketika kita ingin melihat nilai tertinggi dari semua yang sudah pernah bermain		

Tabel 4.6: Tabel *Storyboard* Untuk Tentang Aplikasi

Visual	Sketsa	Audio
Keterangan animasi : Menampilkan halaman menu Tentang Aplikasi yang berisi info tentang aplikasi yang dimaksud serta identitas dari si peneliti		

4.2.5 Specification

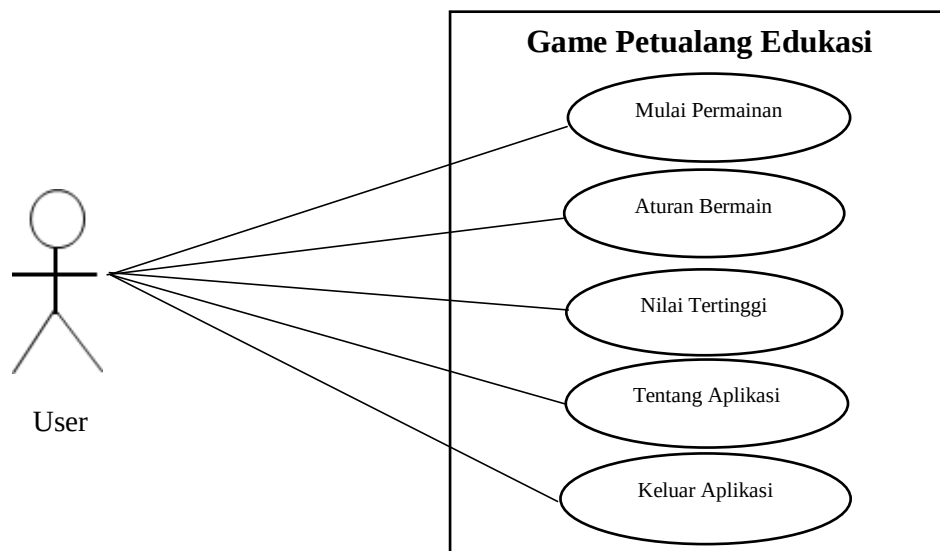
4.2.5.1 Perangkat Keras Untuk Menjalankan Game

Berdasarkan spesifikasi dan kapabilitas game yang dibuat, perangkat keras yang digunakan untuk menjalankan adalah sebagai berikut :

- CPU dengan kecepatan 0.00 GHz
- Kapasitas Harddisk 000 GB
- RAM 0 GB
- Monitor dengan resolusi 1366 x 768p
- Keyboard
- Mouse

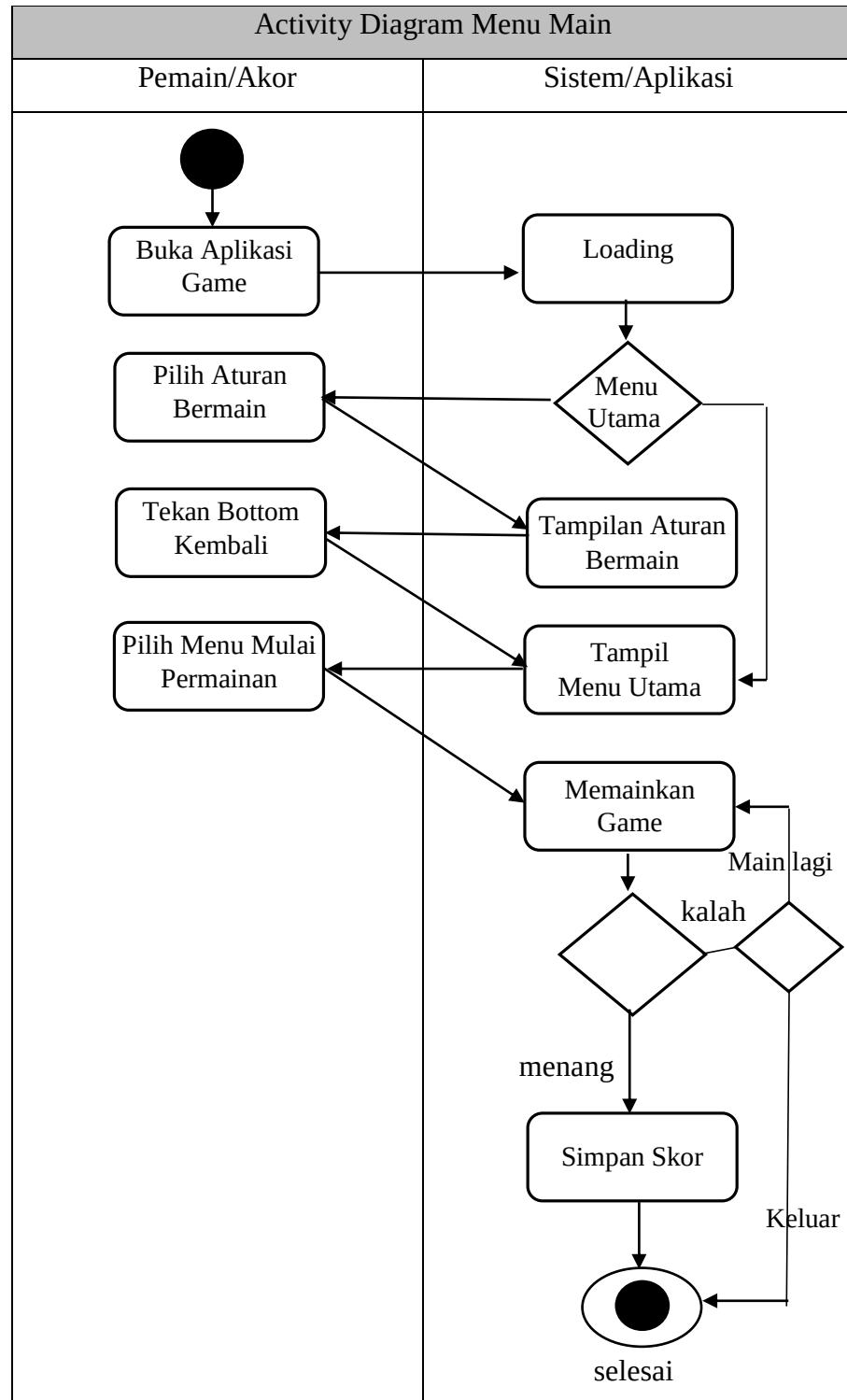
4.2.6 Functional Modelling, Menggunakan Alat Bantu UML

4.2.6.1 Dalam Bentuk Use Case Diagram



Gambar 4.6: Use Case Diagram Aplikasi

4.2.6.2 Dalam Bentuk Activity Diagram



Gambar 4.7: Activity Diagram Menu

4.2.7 Pengujian Sistem

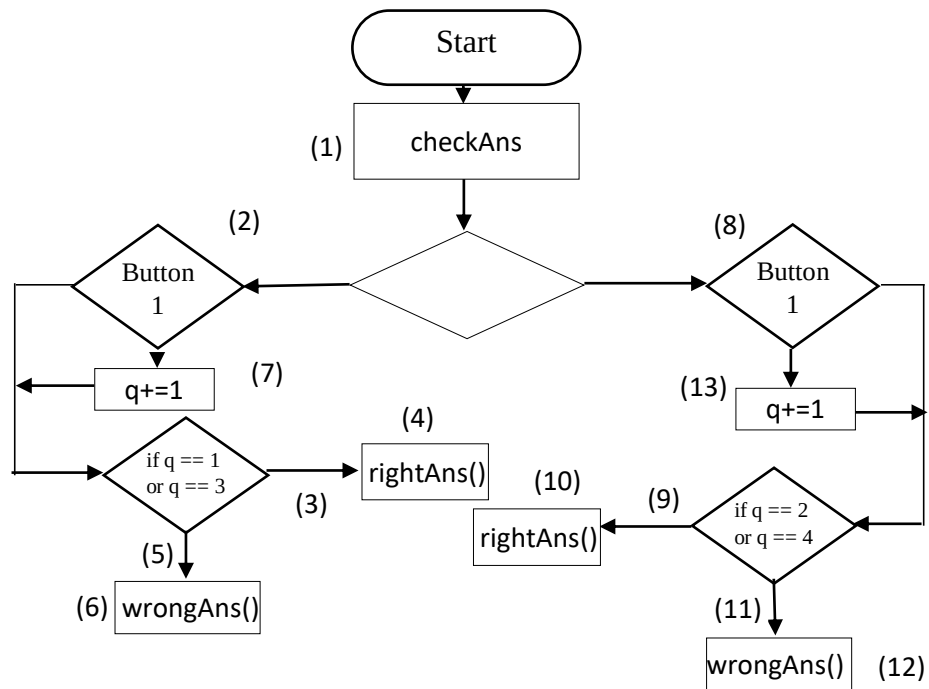
4.2.7.1 White Box Testing

Fungsi If

```
def checkAns(mousePos):.....( 1 )
    if (mousePos[0] in range(160, 420)) and (mousePos[1] in range(410, 482)):.....( 2 )
        global q.....( 3 )
        if q == 1 or q == 3:.....( 3 )
            correctAudio.play().....( 4 )
            rightAns() .....( 4 )
        else:.....( 5 )
            wrongAudio.play().....( 6 )
            wrongAns().....( 6 )
        q+=1.....( 7 )

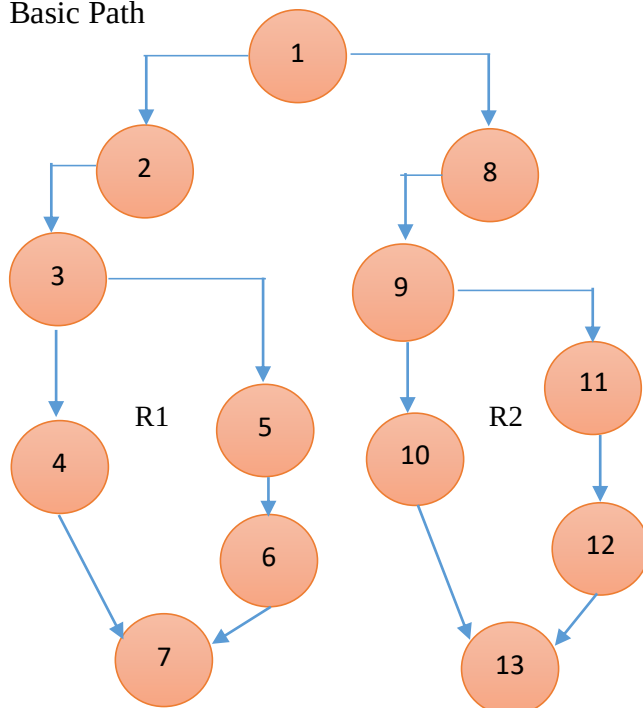
    elif (mousePos[0] in range(588, 847)) and (mousePos[1] in range(410, 482)):.....( 8 )
        if q == 2 or q == 4:.....( 9 )
            correctAudio.play().....( 10 )
            rightAns().....( 10 )
        else:.....( 11 )
            wrongAudio.play().....( 12 )
            wrongAns().....( 12 )
        q+=1.....( 13 )
```

Flowchart



Gambar 4.8: Flowchart Pengujian White Box

Basic Path



Diketahui:

$$R = 2$$

$$N = 13$$

$$E = 14$$

$$P = 2$$

$$\text{Rms: } V(G) = (E - N) + 2$$

$$V(G) = P + 1$$

Penyelesaian:

$$V(G) = (14 - 13) + 2 = 3$$

$$V(G) = 2 + 1 = 3$$

Jalur

1, 2, 3, 4, 7

1, 2, 3, 5, 6, 7

1, 8, 9, 10, 13

1, 8, 9, 11, 12, 13

Gambar 4.9: Flowgraph Pengujian White Box

4.2.7.2 Black Box Testing

No	Nama Pengujian	Kondisi Pengujian	Hasil Pengujian
1.	Menu Mulai Permainan	Menampilkan Awal Permainan pada level 1	Sesuai
2.	Menu Aturan Bermain	Menampilkan aturan bermain sebagai petunjuk di dalam permainan	Sesuai
3.	Menu Nilai Tertinggi	Menampilkan Nilai Tertinggi yang sudah di dapatkan	Sesuai
4.	Menu Tentang Aplikasi	Menampilkan informasi tentang aplikasi	Sesuai
5.	<i>Bottom</i> Main Lagi	Menampilkan kembali ke permainan awal	Selesai
6.	<i>Bottom</i> Keluar	Menyudahi permainan	Selesai
7.	<i>Bottom</i> Pilihan Buah	Menampilkan Quis dalam genre Buah	Sesuai
8.	<i>Bottom</i> Pilihan Hewan	Menampilkan <i>Quis</i> dalam genre Hewan	Sesuai
9.	Papan <i>Keyboard</i>	Memilih sesuai huruf per huruf	Sesuai
10.	<i>Bottom</i> Kembali	Menampilkann halaman sebelumnya	Sesuai
11.	Menu Kaluar Aplikasi	Untuk keluar dari aplikasi	Sesuai

Gambar 4.10: *Black Box Testing*

BAB V

PEMBAHASAN

5.1 Pembahasan Model

Penelitian ini menghasilkan sebuah media berupa game edukasi yang berbasis petualang yang di peruntukkan kepada siswa sekolah dasar khususnya kelas 4,5,6. Perancangan Umum pada game ini akan membelajarkan bahasa inggris pada anak sekolah dasar, yang membawa anak tersebut untuk masuk atau berperan di dalam game, agar siswa lebih senang ketika belajar, game ini juga bukan sekedar game pengenalan kosa kata dalam bahasa inggris yang biasanya, tetapi terdapat quis pada setiap akhir level yang nantinya akan menguji pengetahuan yang pemain sudah dapatkan selama permainan.

5.1.1 Rancangan Permainan Petualang

Dalam Perancangan Game Petualang ini penulis telah membuat game yang di dalamnya pemain atau aktor akan di tantang untuk menyelesaikan tantangan serta ketangkasan yang ada pemain juga harus melatih daya ingatan mereka untuk mengingat objek-objek yang akan mereka lewati, agar bisa menyelesaikan quis pada akhir level untuk melanjutkan ke level selanjutnya, pemain juga akan di sediakan beberapa bantuan jika mereka bisa menjawab tantang dari sebuah game. Semakin tinggi level yang di mainkan maka semakin sulit pula tantangan yang di selesaikan.

5.2 Pembahasan Sistem

5.2.1 Menu Utama

Tampilan menu utama game edukatif berbentuk petualang terlihat pada gambar di berikut ini:



Gambar 5.1: Tampilan Menu Utama

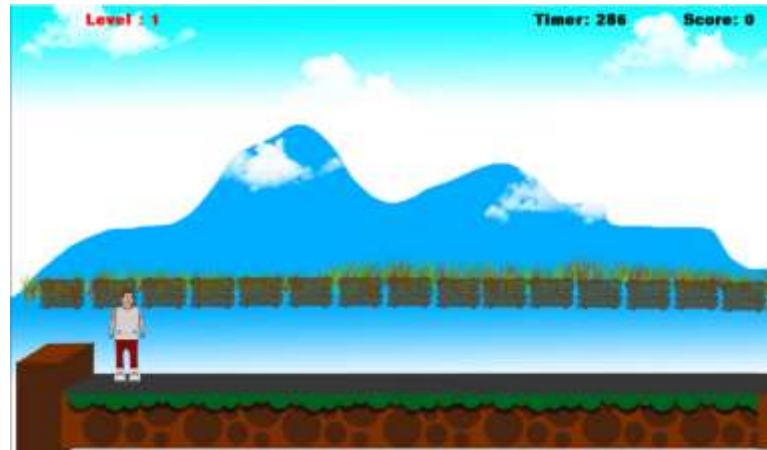
Tampilan ini terdiri dari menu diantara lain:

1. Mulai Permainan : Menu untuk memulai permainan, dengan memperhatikan kosa kata yang di tampilkan
2. Aturan Bermain : Menu untuk pemain dapat mengetahui aturan-aturan yang ada di dalam game
3. Tentang Aplikasi : Menu yang di dalamnya terdapat apa maksud peneliti membuat game tersebut dan identitas peneliti
4. Nilai Tertinggi : Menu yang berisikan Score tertinggi yang pernah di capai saat bermain
5. Keluar Aplikasi : Untuk keluar dari aplikasi atau mengakhiri permainan

5.2.2 Halaman Sub Level

5.2.2.1 Level 1

Pemain akan di tantang untuk memainkan game yang masih sediki mudah dan kata-kata bahasa inggris yang masih mudah untuk dimengerti atau sering di jumpai, pada level ini pemain diberikan waktu 300 timer berjalan dan harus melawati 5 jurang dan 5 tembok yang sudah di atur dan 4 buah objek dalam bahasa inggris yang akan di tampilkan secara random, pada level ini disajikan pepadangan yang menarik pada pagi hari, seperti pada gambar berikut ini :



Gambar 5.2: Tampilan Level 1

5.2.2.2 Level 2

Pemain akan di tantang untuk menyelesaikan permainan, pada level ini akan di sajikan kata-kata bahasa inggris yang masih sediki mudah dan tingkat petualang tidak sulit dan pada level ini pemain diberikan waktu 250 timer berjalan serta harus melewati 8 jurang dan 8 tembok yang sudah di atur dengan tingkat kesulitan perlevel dan 4 buah objek berbeda dengan level 1 dalam bahasa inggris yang akan di tampilkan secara random, pada level ini disajikan pemandangan yang menarik pada senja hari, seperti pada gambar berikut ini :



Gambar 5.3: Tampilan Level 2

5.2.2.3 Level 3

Pemain akan di tantang untuk menyelesaikan permainan, pada level ini Akan di sajikan kata-kata bahasa inggris sudah mulai sulit dan tantangan petualang yang lumayan sulit, dan pada level ini pemain diberikan waktu 240 timer berjalan serta akan melewati 11 jurang dan 11 tembok yang sudah di atur dengan tingkat level berbeda dan 4 buah objek yang berbeda dengan level 1 dan 2 dalam bahasa inggris yang akan di tampilkan secara random, pada level ini disajikan suasana pada malam hari, seperti pada gambar berikut ini :



Gambar 5.4: Tampilan Level 3

5.2.2.4 Level 4

Pemain akan di tantang untuk menyelesaikan permainan, pada level ini Akan di sajikan kata-kata bahasa inggris yang sulit dan tingkat petualang sulit dan pada level ini pemain diberikan waktu 230 timer berjalan serta akan melewati 11 jurang dan 11 tembok yang sudah di atur dengan tingkat level berbeda-beda dan 4 buah objek yang berbeda dengan level sebelumnya dalam bahasa inggris yang akan di tampilkan secara random, pada level ini disajikan suasana pada malam hari, seperti pada gambar berikut ini :



Gambar 5.5: Tampilan Level 4

5.2.2.5 Level 5

Pemain akan di tantang untuk menyelesaikan permainan, pada level ini Akan di sajikan kata-kata bahasa inggris sangat sulit dan tingkat petualang sangat sulit dan pada level ini di berikan waktu 200 timer berjalan serta pemain akan melewati 15 jurang dan 15 tembok yang sudah di atur dengan tingkat level berbeda dan 20 buah objek dalam bahasa inggris yang akan di tampilkan secara random yang merupakan gabungan dari semua objek yang di level sebelumnya dan perjalanannya juga cukup lebih panjang, pada level ini menyajikan pemandangan pada pagi hari, seperti pada gambar berikut ini :

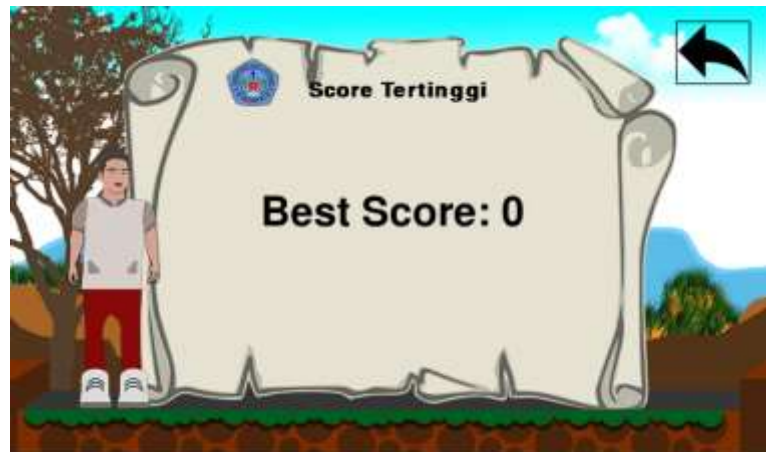


Gambar 5.6: Tampilan Level 5

5.2.2.6 Menu Aturan Permainan

5.2.2.8 Menu Nilai Tertinggi

Sebuah menu pilihan yang berfungsi untuk melihat berapa score tertinggi yang pernah di capai oleh pemain, untuk lebih jelasnya bisa dilihat pada gambar berikut ini :



Gambar 5.9: Tampilan Nilai Tertinggi

5.2.2.9 Quis Perlevel

Tahap ini adalah tahap dimana pemain untuk memilih jawaban yang tepat dari dua pilihan yang sudah di sediakan, tahap ini pemain akan jumpai pada akhir permainan level setelah menemukan peti harta karun, untuk menjawab pertanyaan pemain harus menjawab kurang lebih 3 pertanyaan benar agar dapat melanjutkan kelevel selanjutnya, jika pemain menjawab tidak melampaui yang di maksud maka pemain akan dikembalikan ke soal pertama pada quis itu, Untuk tampilannya dapat kita lihat pada gambar berikut ini :



Gambar 5.10: Tampilan Quis Perlevel

5.2.6.0 Quis Akhir

Tahap ini adalah tahap dimana akhir dari permainan yang nantinya akan menguji pengetahuan pemain, cara menjawabnya yaitu dengan memilih huruf perhuruf dalam bahasa inggris yang sesuai dengan gambar yang akan di tampilkan, disediakan 6 nyawa, jika pemain menekan huruf yang tidak terdapat pada nama gambar di tampilkan makan pemain akan kehilangan 1 nyawa, dan seterusnya. Poin bertambah sesuai keadaan ketika menjawab pertanyaan. Untuk tampilannya bisa di lihat pada gambar berikut ini :



Gambar 5.11: Tampilan Quis Akhir

BAB VI

PENUTUP

6.1 Kesimpulan

Dari hasil penelitian, bahwa game Petualangan Edukasi Bahasa Inggris Pada Anak Sekolah Dasar dapat ditarik kesimpulan sebagai berikut:

1. Telah berhasil dibuat game PC dengan judul “Game Petualang Edukasi Bahasa Inggris Pada Anak Sekolah Dasar”
2. Permainan ini mengajarkan kosa kata pada anak sekolah dasar.
3. Untuk metode karya wisata pada game ini penulis berhasil menyatuhkan petualang serta pemandangan pada game edukasi.
4. Pengujian dalam game ini menggunakan pengujian Blackbox dan Whitebox. dengan begitu game ini bisa dapat berjalan sesuai dengan aturan-aturanya.

6.2 Saran

Beberapa saran untuk pembangunan game selanjutnya yakni, antara lain:

1. Menambah jumlah level permainan agar permainan yang telah dibuat lebih menarik dan anak-anak lebih mengetahui kosa kata lebih banyak.
2. Menambah beberapa item di sela-sela perjalanan sebagai bantuan yang akan membantu pemain untuk menyelesaikan quis atau dalam menjawab quis
3. Menambahkan animasi di setiap perpindahan menu dan pada saat mengklik bottom.
4. Untuk Pengembangan Gamenya nanti lebih tertujuh pada edukasi

DAFTAR PUSTAKA

- [1] S. Maesaroh and N. Malkiah, "Media Pembelajaran Interaktif Bahasa Inggris Pengenalan Huruf & Membaca Berbasis Multimedia untuk Sekolah Dasar," *J. Sisfotek Glob.*, vol. 5, no. 1, pp. 81–86, 2015.
- [2] R. S. Pindo Hutauruk, "Meningkatkan Hasil Belajar Siswa Dengan Alat Peraga Pada Mata Pelajaran Ipa Kelas IV SDN Nomor 14 Simbolon Purba," *SEJ (School Educ. Journal)*, vol. 8, no. 2, pp. 172–181, 2018.
- [3] Y. I. K. F Y Al Irsyadi, R. Annas, "Game Edukasi Pembelajaran Bahasa Inggris Untuk Pengenalan Benda-Benda di Rumah Bagi Siswa Kelas 4 Sekolah Dasar," vol. 9, no. September, pp. 78–92, 2019.
- [4] Jumiati, "Penerapan Metode Karya Wisata Pada Konsep Dasar IPA MI/SD Materi Perkembangbiakan Untuk Meningkatkan Motivasi Belajar Mahasiswa PGMI Informasi Artikel," vol. 2, no. 2, pp. 19–27, 2017.
- [5] R. Radhakrishna and C. Ravi, "Metode Mengajar Sebagai Strategi Dalam Mencapai Tujuan Belajar Mengajar," vol. III, no. 1, pp. 66–70, 2004.
- [6] Y. Abbas, E. Winarno, P. Studi, T. Informatika, F. T. Informasi, and U. Stikubank, "Perancangan Game Edukasi Pengenalan Angka Dalam Bahasa," no. 2012, pp. 347–352, 2018.
- [7] F. C. Marzuki, "Game Berbasis Adventure Sebagai Pendukung Anak Usia Dini," pp. 1–11, 2009.
- [8] E. Santoso, G. M. Satia Budhi, and R. Intan Eng, "Pembuatan Game dengan Menerapkan Metode Decision Tree: UCB1, untuk Menentukan Pemilihan Strategy dalam AI."
- [9] V. G. U. Taufik Ramadhan, "Rancang Bangun Aplikasi Mobile Untuk Notifikasi Jadwal," *J. Teknol. Inf. dan Komunikasi*, vol. 5, no. 1, pp. 47–55, 2014.
- [10] S. P. D. D. Anggriani, Arifiati Fitri; Erviana, Nena; Anggriani, "Aplikasi Game Edukasi Petualangan Nusantara," *J. Media Pembelajaran Lewat Game*, vol. 8, pp. 168–172, 2016.
- [11] A. Syafrando, J. Wahyudi, and Y. Arliando, "Pembuatan Aplikasi Game

Merawat Hewan Peliharaan Dengan Menggunakan Macromedia Flash 8.0,” vol. 10, no. 1, pp. 65–71, 2014.

- [12] D. P. Ramendra and N. M. Ratminingsih, “Pemanfaatan Audio Visual Aids (AVA) Dalam Proses Mengajar Mata Pelajaran Bahasa Inggris Di Sekolah Dasar,” *J. Penelit. dan Pengemb. Pendidik.*, vol. 1, no. 2, pp. 78–95, 2007.
- [13] A. Senjaya, F. I. Wahid, D. Y. Saputra, M. Lathfullah, and S. Fasya, “Jurnal Membaca Bahasa Dan Sastra Indonesia,” 2016.
- [14] S. A. Susana, *Buku Pintar Bahasa Inggris Kelas 4,5 & 6 SD*. 2008.
- [15] H. A. Emka, “Game Edukasi Bahasa Indonesia Kelas 1 Sekolah Dasar Berbasis Android Menggunakan DGBL-ID Model,” vol. 2, no. 1, pp. 10–20, 2017.
- [16] M. Yunus, I. F. Astuti, and D. M. Khairina, “Game Edukasi Matematika Untuk Sekolah Dasar,” *Inform. Mulawarman J. Ilm. Ilmu Komput.*, vol. 10, no. 2, p. 59, 2015.
- [17] Yunahar Heriyanto, “Perancangan Sistem Informasi Rental Mobil Berbasis Web Pada PT.APM Rent Car,” *J. Intra-Tech*, vol. 2, no. 2, pp. 64–77, 2018.
- [18] Suendri, “Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan),” *J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, pp. 1–9, 2018.
- [19] K. Kawano, Y. Umemura, and Y. Kano, “Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak),” *Crop Sci.*, vol. 23, no. 2, p. 201, 2016.
- [20] N. Kholis et al., “Praktikalitas Pengembangan Modul Konstruktivisme dan Website pada Materi Lingkaran dan Bola,” *J. Pekommas*, vol. 2, no. 1, pp. 6–12, 2014.
- [21] C. Montolalu and Y. Langi, “Pengaruh Pelatihan Dasar Komputer dan Teknologi Informasi bagi Guru-Guru dengan Uji-T Berpasangan (Paired Sample T-Test),” *d’CARTESIAN*, vol. 7, no. 1, p. 44, 2018.

Lampiran 1: Kode Program

Coding Perlevel

```
import pygame, sys
from pygame.locals import *
import os
import random

pygame.init()
#screen
W, H = 1024, 600
win = pygame.display.set_mode((W,H), pygame.RESIZABLE)
monitor_size = [pygame.display.Info().current_w,
pygame.display.Info().current_h]
pygame.display.set_caption("Game Petualang Bahasa Inggris -
T3116043_Nuriman Hamka Genti")
clock = pygame.time.Clock()
bgX = 0
liX = 0

#image
bg = pygame.image.load(os.path.join('images', 'bgl3-4.png'))
li = pygame.image.load(os.path.join('images', 'lintasan13.png'))
jurber = [pygame.image.load(os.path.join('images', 'jurang2.png'))]
kotak = [pygame.image.load(os.path.join('images', 'box.png'))]
tmbk = [pygame.image.load(os.path.join('images', 'tembok.png'))]
mbox = [pygame.image.load(os.path.join('images', 'mbox.png'))]
walkRight = [pygame.image.load(os.path.join('images', 'R1.png')),
pygame.image.load(os.path.join('images', 'R2.png')),
pygame.image.load(os.path.join('images', 'R3.png')),
pygame.image.load(os.path.join('images', 'R4.png')),
pygame.image.load(os.path.join('images', 'R5.png')),
pygame.image.load(os.path.join('images', 'R6.png')),
pygame.image.load(os.path.join('images', 'R7.png')),
pygame.image.load(os.path.join('images', 'R8.png')),
pygame.image.load(os.path.join('images', 'R9.png'))]
walkLeft = [pygame.image.load(os.path.join('images', 'L1.png')),
pygame.image.load(os.path.join('images', 'L2.png')),
pygame.image.load(os.path.join('images', 'L3.png')),
pygame.image.load(os.path.join('images', 'L4.png')),
pygame.image.load(os.path.join('images', 'L5.png')),
pygame.image.load(os.path.join('images', 'L6.png')),
pygame.image.load(os.path.join('images', 'L7.png'))]
```

```

pygame.image.load(os.path.join('images', 'L8.png')),
pygame.image.load(os.path.join('images', 'L9.png'))]
ulang = pygame.image.load(os.path.join('images', 'icon', 'b.main.png'))
menu = pygame.image.load(os.path.join('images', 'icon', 'b.menu.png'))
depan = pygame.image.load(os.path.join('images', 'dp.png'))

```

```

#music
sounds = {"win":pygame.mixer.Sound("sound/win.wav"),
"lose":pygame.mixer.Sound("sound/lose.wav"),
"click":pygame.mixer.Sound("sound/click.wav"),
"clock":pygame.mixer.Sound("sound/detik.wav"),
         "ceri":pygame.mixer.Sound("sound/ceri.wav"),
"anggur":pygame.mixer.Sound("sound/anggur.wav"),
"monyet":pygame.mixer.Sound("sound/monyet.wav"),
"zebra":pygame.mixer.Sound("sound/jebra.wav")}
music = pygame.mixer.music.load(os.path.join('sound', 'music.wav'))
pygame.mixer.music.play(-1)

```

```

#font
font = pygame.font.SysFont('comicsans', 30, True)
largeFont = pygame.font.SysFont('comicsans', 80)
bigfont = pygame.font.SysFont('comicsans', 100, True)

```

```

class player(object):
    def __init__(self,x,y,width,height):
        self.x = 100
        self.y = 365
        self.width = width
        self.height = height
        self.vel = 5
        self.isJump = False
        self.left = False
        self.right = False
        self.falling = False
        self.walkCount = 10
        self.jumpCount = 10
        self.standing = True
        self.hitbox = (self.x, self.y, 0, 0)
        self.hitbox1 = (self.x, self.y, 0, 0)

```

```

    def draw(self, win):
        if self.falling:
            self.y += 80
            self.standing = True

        if self.walkCount + 1 >= 27:

```

```

        self.walkCount = 0

    if not(self.standing):
        if self.left:
            win.blit(walkLeft[self.walkCount//3], (self.x,self.y))
            self.walkCount += 1
        elif self.right:
            win.blit(walkRight[self.walkCount//3], (self.x,self.y))
            self.walkCount += 1

    else:
        if self.right:
            win.blit(depan, (self.x, self.y))
        else:
            win.blit(depan, (self.x, self.y))
    self.hitbox = (self.x+40, self.y+15, 30, 120)
    #pygame.draw.rect(win, (255,0,0), self.hitbox,2)

def hit(self):
    self.x = 60
    self.y = 410
    self.walkCount = 0
    font1 = pygame.font.SysFont('comicsans', 100)
    text = font1.render('-5', 1, (255,0,0))
    win.blit(text, (500 - (text.get_width()/2),200))
    pygame.display.update()
    i = 0
    while i < 300:
        pygame.time.delay(2)
        i += 1
        for event in pygame.event.get():
            if event.type == pygame.QUIT:
                i = 301
                pygame.quit()

class projectile(object):
    def __init__(self,x,y,radius,color,facing):
        self.x = x
        self.y = y
        self.radius = radius
        self.color = color
        self.facing = facing
        self.vel = 8 * facing

    def draw(self,win):
        pygame.draw.circle(win, self.color, (self.x,self.y), self.radius)

```

```

class enemy(object):
    walkRight = [pygame.image.load(os.path.join('images','birdR.png'))]
    walkLeft = [pygame.image.load(os.path.join('images','birdL.png'))]
    def __init__(self, x, y, width, height, end):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.end = 150
        self.path = [self.x, self.end]
        self.walkCount = 0
        self.vel = 3
        self.hitbox = (self.x + 17, self.y + 2, 31, 57)
        self.health = 10
        self.visible = False

    def draw(self,win):
        self.move()
        if self.visible:
            if self.walkCount + 1 <= 33:
                self.walkCount = 0

            if self.vel > 0:
                win.blit(self.walkLeft[self.walkCount //3], (self.x, self.y))
                self.walkCount += 1
            else:
                win.blit(self.walkRight[self.walkCount //3], (self.x, self.y))
                self.walkCount += 1
            pygame.draw.rect(win, (255,0,0), (self.hitbox[0], self.hitbox[1] - 20, 50,
10))
            pygame.draw.rect(win, (0,128,0), (self.hitbox[0], self.hitbox[1] - 20, 50 -
(5 * (10 - self.health)), 10))
            self.hitbox = (self.x + 17, self.y + 2, 31, 57)
            #pygame.draw.rect(win, (255,0,0), self.hitbox,2)

    def move(self):
        if self.vel > 0:
            if self.x + self.vel > self.path[1]:
                self.x -= 50
            else:
                self.vel = self.vel * -1
                self.walkCount = 0
        else:
            if self.x - self.vel < self.path[0]:
                self.x -= self.vel

```

```

        else:
            self.vel = self.vel * -1
            self.walkCount = 0

    def hit(self):
        if self.health > 0:
            self.health -= 1

        else:
            self.visible = False

    print('hit')

class jurang (object):
    def __init__(self, x, y, width, height, end):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.walkCount = 0
        self.vel = 5.5
        self.hitbox = (self.x, self.y, 70, 5)
        self.visible = True
        self.right = False

    def draw(self,win):
        if self.walkCount + 1 >= 27:
            self.walkCount = 0

        if self.right:
            win.blit(jurber[self.walkCount//100], (self.x,self.y))
            self.walkCount +=1
        else:
            win.blit(jurber[self.walkCount//100], (self.x,self.y))

        if self.visible:
            self.hitbox = (self.x + 55, self.y, 15, 57)
            #pygame.draw.rect(win, (255,0,0), self.hitbox,2)
    def collide(self, rect):
        if rect[0] + rect[2] > self.hitbox[0] and rect[0] < self.hitbox[0] +
self.hitbox[2]:
            if rect[1] + rect[3] > self.hitbox[1]:
                return True
            return False

```

```

class tembok(object):
    def __init__(self, x, y, width, height, end):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.walkCount = 0
        self.vel = 5.5
        self.hitbox = (self.x, self.y, 70, 10)
        self.hitbox2 = (self.x, self.y, 70, 60)
        self.hitbox3 = (self.x, self.y, 70, 60)
        self.visible = True
        self.right = False

    def draw(self, win):
        if self.walkCount + 1 >= 27:
            self.walkCount = 0

        if self.right:
            win.blit(tmbk[self.walkCount//100], (self.x, self.y))
            self.walkCount += 1
        else:
            win.blit(tmbk[self.walkCount//100], (self.x, self.y))

        self.hitbox = (self.x + 5, self.y + 10, 70, 20)
        #pygame.draw.rect(win, (255,0,0), self.hitbox, 2)
        self.hitbox2 = (self.x - 8, self.y + 30, 20, 60)
        #pygame.draw.rect(win, (255,0,0), self.hitbox2, 2)
        self.hitbox3 = (self.x + 60, self.y + 30, 20, 60)
        #pygame.draw.rect(win, (255,0,0), self.hitbox3, 2)

    def collider(self, rect):
        if rect[0] + rect[2] > self.hitbox2[0] and rect[0] < self.hitbox2[0] +
self.hitbox2[2]:
            if rect[1] + rect[3] > self.hitbox2[1]:
                return True

    def collidel(self, rect):
        if rect[0] + rect[2] > self.hitbox3[0] and rect[0] < self.hitbox3[0] +
self.hitbox3[2]:
            if rect[1] + rect[3] > self.hitbox3[1]:
                return True
            return False

    def collidet(self, rect):

```



```

        if rect[0] + rect[2] > self.hitbox[0] and rect[0] < self.hitbox[0] +
self.hitbox[2]:
            if rect[1] + rect[3] > self.hitbox[1]:
                return True
            return False

class misteribox(tembok):
    def draw(self,win):
        if self.walkCount + 1 >= 27:
            self.walkCount = 0

        if self.right:
            win.blit(mbox[self.walkCount//100], (self.x,self.y))
            self.walkCount +=1
        else:
            win.blit(mbox[self.walkCount//100], (self.x,self.y))

        if self.visible:
            self.hitbox = (self.x + 10, self.y + 10, 70, 20)
            #pygame.draw.rect(win, (255,0,0), self.hitbox,2)

        self.hitbox2 = (self.x + 200, 2, 20, 500)
        #pygame.draw.rect(win, (255,0,0), self.hitbox2,2)

        self.hitbox3 = (self.x - 3000, 2, 20, 500)
        #pygame.draw.rect(win, (255,0,0), self.hitbox3,2)

        def collide(self, rect):
            if rect[0] + rect[2] > self.hitbox[0] and rect[0] < self.hitbox[0] +
self.hitbox[2]:
                if rect[1] + rect[3] > self.hitbox[1]:
                    return True
                return False

class zebra(object):
    img = pygame.image.load(os.path.join('images', 'hewan', 'zebra.png'))
    def __init__(self, x, y, width, height):
        self.x = x
        self.y = y
        self.width = width
        self.height = height
        self.vel = 1.4
        self.hitbox = (self.x+80, self.y, 50, 400)

    def draw(self, win):
        self.hitbox = (self.x+80, self.y, 50, 400)

```

```

        #pygame.draw.rect(win, (255,0,0), self.hitbox, 2)
        win.blit(self.img, (self.x, self.y))

    def collide(self, rect):
        if rect[0] + rect[2] > self.hitbox[0] and rect[0] < self.hitbox[0] +
self.hitbox[2]:
            if rect[1] < self.hitbox[3]:
                return True
            return False

class ceri(zebra):
    img = pygame.image.load(os.path.join('images', 'buah', 'ceri.png'))
    def draw(self, win):
        self.hitbox = (self.x+80, self.y, 50, 300)
        #pygame.draw.rect(win, (255,0,0), self.hitbox, 2)
        win.blit(self.img, (self.x, self.y))

class anggur(zebra):
    img = pygame.image.load(os.path.join('images', 'buah', 'anggur.png'))
    def draw(self, win):
        self.hitbox = (self.x+80, self.y, 50, 300)
        #pygame.draw.rect(win, (255,0,0), self.hitbox, 2)
        win.blit(self.img, (self.x, self.y))

class monyet(zebra):
    img = pygame.image.load(os.path.join('images', 'hewan', 'monyet.png'))
    def draw(self, win):
        self.hitbox = (self.x+80, self.y, 50, 300)
        #pygame.draw.rect(win, (255,0,0), self.hitbox, 2)
        win.blit(self.img, (self.x, self.y))

def updateFile():
    f = open('hgscores.txt','r')
    file = f.readlines()
    last = int(file[0])
    if last < int(score):
        f.close()
        file = open('hgscores.txt', 'w')
        file.write(str(score))
        file.close()
        return score
    return last

def updatescores():
    f = open('scores.txt','r')
    file = f.readlines()

```

```
last = int(file[0])
f.close()
file = open('scores.txt', 'w')
file.write(str(score))
file.close()
return score
```

```
click = False
def endScreen():
    global score, speed, speed1, speed2, pause, fallSpeed, bgX, liX, obstacles, man,
    j1, j2, j3, j4, j5, j6, j7, j8, j9, j10, j11, t1, t2, t3, t4, t5, t6, t7, t8, t9, t10, t11, mb
    bgX = 0
    liX = 0
    speed = 0
    speed1 = 0
    speed2 = 0
    pause = 0
    shootLoop = 0
    fallSpeed = 0
    obstacles = []
    bullets = []
    mb.visible = False
    man = player(200, 410, 64, 64)
    bird = enemy(1024, 350, 64, 64, 450)
    j1 = jurang(300, 490, 64, 64, 450)
    j2 = jurang(1000, 490, 64, 64, 450)
    j3 = jurang(2000, 490, 64, 64, 450)
    j4 = jurang(2400, 490, 64, 64, 450)
    j5 = jurang(2800, 490, 64, 64, 450)
    j6 = jurang(3500, 490, 64, 64, 450)
    j7 = jurang(3650, 490, 64, 64, 450)
    j8 = jurang(3800, 490, 64, 64, 450)
    j9 = jurang(4500, 490, 64, 64, 450)
    j10 = jurang(5600, 490, 64, 64, 450)
    j11 = jurang(6050, 490, 64, 64, 450)
    t1 = tembok(1200, 450, 64, 64, 450)
    t2 = tembok(2200, 450, 64, 64, 450)
    t3 = tembok(2600, 450, 64, 64, 450)
    t4 = tembok(4000, 450, 64, 64, 450)
    t5 = tembok(4600, 450, 64, 64, 450)
    t6 = tembok(5000, 450, 64, 64, 450)
    t7 = tembok(5200, 450, 64, 64, 450)
    t8 = tembok(5400, 450, 64, 64, 450)
    t9 = tembok(5690, 450, 64, 64, 450)
    t10 = tembok(5845, 450, 64, 64, 450)
    t11 = tembok(6000, 450, 64, 64, 450)
```

```

mb = misteribox(6300, 470, 64, 64, 450)
run = True
while run:

    pygame.time.delay(1000)
    click = False
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
            pygame.quit()
        if event.type == pygame.MOUSEBUTTONDOWN:
            if event.button == 1:
                click = True

    mx, my = pygame.mouse.get_pos()
    button_1 = pygame.Rect(120, 500, 230, 50)
    button_2 = pygame.Rect(680, 500, 230, 50)
    if button_1.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.time.delay(500)
            run = False
            man.falling = False

    if button_2.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.time.delay(500)
            pygame.quit()

    win.blit(bg, (0,0))
    win.blit(ulang, button_1)
    win.blit(menu, button_2)
    #pygame.draw.rect(win, (255, 0, 0), button_1)
    #pygame.draw.rect(win, (255, 0, 0), button_2)
    largeFont = pygame.font.SysFont('comicsans', 80)
    lastScore = largeFont.render('Best Score: ' +
str(updateFile()),1,(255,255,255))
    currentScore = largeFont.render('Score: '+ str(scores),1,(255,255,255))
    win.blit(lastScore, (W/2 - lastScore.get_width()/2,150))
    win.blit(currentScore, (W/2 - currentScore.get_width()/2, 240))
    pygame.display.update()
    score = 0

click = False
def paused():

```

```

run = False
pause = True
while pause:
    click = False
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            run = False
            pygame.quit()
        if event.type == pygame.MOUSEBUTTONDOWN:
            if event.button == 1:
                click = True
    mx, my = pygame.mouse.get_pos()
    button_1 = pygame.Rect(120, 500, 230, 50)
    button_2 = pygame.Rect(680, 500, 230, 50)
    if button_1.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.time.delay(500)
            pause = False
            man.falling = False

    if button_2.collidepoint((mx, my)):
        if click:
            sounds["click"].play()
            pygame.time.delay(500)
            pygame.quit()
    psd = bigfont.render('PAUSED ', 1, (255,0,0))
    win.blit(bg, (0,0))
    win.blit(ulang, button_1)
    win.blit(menu, button_2)
    win.blit(psd, (355, 200))
    #pygame.draw.rect(win, (255, 0, 0), button_1)
    #pygame.draw.rect(win, (255, 0, 0), button_2)
    pygame.display.update()

def redrawGameWindow():
    scr = font.render('Score: ' + str(scores), 1, (255,255,255))
    tmr = font.render('Timer: ' + str(timer), 1, (255,255,255))
    lv = font.render('Level : 3 ', 1, (255,0,0))
    pr = font.render('Waspada! burung!! Seketika akan menyerang' , 1,
(255,255,255))
    win.blit(bg, (bgX,0))
    win.blit(li, (liX,450))
    j1.draw(win)
    j2.draw(win)
    j3.draw(win)

```

```

j4.draw(win)
j5.draw(win)
j6.draw(win)
j7.draw(win)
j8.draw(win)
j9.draw(win)
j10.draw(win)
j11.draw(win)
t1.draw(win)
t2.draw(win)
t3.draw(win)
t4.draw(win)
t5.draw(win)
t6.draw(win)
t7.draw(win)
t8.draw(win)
t9.draw(win)
t10.draw(win)
t11.draw(win)
for obstacle in obstacles:
    obstacle.draw(win)
for bullet in bullets:
    bullet.draw(win)
win.blit(scr, (850, 10))
win.blit(tmr, (700, 10))
win.blit(lv, (50, 10))

win.blit(pr, (50, 50))
mb.draw(win)
bird.draw(win)
man.draw(win)
pygame.display.update()

```

```

#mainloop
pygame.time.set_timer(USEREVENT+1, 500)
pygame.time.set_timer(USEREVENT+2, 3000)
man = player(200, 410, 64, 64)
bird = enemy(800, 350, 64, 64, 450)
shootLoop = 0
bullets = []
j1 = jurang(300, 490, 64, 64, 450)
j2 = jurang(1000, 490, 64, 64, 450)
j3 = jurang(2000, 490, 64, 64, 450)
j4 = jurang(2400, 490, 64, 64, 450)
j5 = jurang(2800, 490, 64, 64, 450)
j6 = jurang(3500, 490, 64, 64, 450)

```

```
j7 = jurang(3650, 490, 64, 64, 450)
j8 = jurang(3800, 490, 64, 64, 450)
j9 = jurang(4500, 490, 64, 64, 450)
j10 = jurang(5600, 490, 64, 64, 450)
j11 = jurang(6050, 490, 64, 64, 450)
t1 = tembok(1200, 450, 64, 64, 450)
t2 = tembok(2200, 450, 64, 64, 450)
t3 = tembok(2600, 450, 64, 64, 450)
t4 = tembok(4000, 450, 64, 64, 450)
t5 = tembok(4600, 450, 64, 64, 450)
t6 = tembok(5000, 450, 64, 64, 450)
t7 = tembok(5200, 450, 64, 64, 450)
t8 = tembok(5400, 450, 64, 64, 450)
t9 = tembok(5690, 450, 64, 64, 450)
t10 = tembok(5845, 450, 64, 64, 450)
t11 = tembok(6000, 450, 64, 64, 450)
mb = misteribox(6300, 470, 64, 64, 450)
```

```
obstacles = []
speed = 0
speed1 = 0
speed2 = 0
timer = 240
pause = 0
fallSpeed = 0
```

```
run = True
while run:
    clock.tick(27)
```

```
    f = open('scores.txt','r')
    file = f.readlines()
    score = speed//30 + int(file[0])
    timer = speed1//10 + 240
    scores = speed2 + score
```

```
    if shootLoop > 0:
        shootLoop += 1
    if shootLoop > 3:
        shootLoop = 0
```

```
    if timer == 21:
        sounds["clock"].play()
    if timer == 18:
        sounds["clock"].play()
    if timer == 15:
```

```
    sounds["clock"].play()
if timer == 12:
    sounds["clock"].play()
if timer == 9:
    sounds["clock"].play()
if timer == 6:
    sounds["clock"].play()
if timer == 3:
    sounds["clock"].play()
if timer < 1:
    sounds["lose"].play()
    endScreen()
```

#JURANG

```
if pause > 0:
    pause += 1
    if pause > fallSpeed * 2:
        endScreen()
```

```
if j1.collide(man.hitbox):
    sounds["lose"].play()
    man.falling = True
    if pause == 0:
        pause = 1
        fallSpeed = speed1
```

```
if j2.collide(man.hitbox):
    sounds["lose"].play()
    man.falling = True
    if pause == 0:
        pause = 1
        fallSpeed = speed1
```

```
if j3.collide(man.hitbox):
    sounds["lose"].play()
    man.falling = True
    if pause == 0:
        pause = 1
        fallSpeed = speed1
```

```
if j4.collide(man.hitbox):
    sounds["lose"].play()
    man.falling = True
    if pause == 0:
        pause = 1
        fallSpeed = speed1
```



```
if j5.collide(man.hitbox):  
    sounds["lose"].play()  
    man.falling = True  
    if pause == 0:  
        pause = 1  
        fallSpeed = speed1
```

```
if j6.collide(man.hitbox):  
    sounds["lose"].play()  
    man.falling = True  
    if pause == 0:  
        pause = 1  
        fallSpeed = speed1
```

```
if j7.collide(man.hitbox):  
    sounds["lose"].play()  
    man.falling = True  
    if pause == 0:  
        pause = 1  
        fallSpeed = speed1
```

```
if j8.collide(man.hitbox):  
    sounds["lose"].play()  
    man.falling = True  
    if pause == 0:  
        pause = 1  
        fallSpeed = speed1
```

```
if j9.collide(man.hitbox):  
    sounds["lose"].play()  
    man.falling = True  
    if pause == 0:  
        pause = 1  
        fallSpeed = speed1
```

```
if j10.collide(man.hitbox):  
    sounds["lose"].play()  
    man.falling = True  
    if pause == 0:  
        pause = 1  
        fallSpeed = speed1
```

```
if j11.collide(man.hitbox):  
    sounds["lose"].play()  
    man.falling = True
```

```
if pause == 0:
    pause = 1
    fallSpeed = speed1
```

```
#TEMBOK
```

```
if t1.visible == True:
    if man.hitbox[1] < t1.hitbox[1] + t1.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t1.hitbox[1]:
        man.y = 365
```

```
if t2.visible == True:
    if man.hitbox[1] < t2.hitbox[1] + t2.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t2.hitbox[1]:
        man.y = 365
```

```
if t3.visible == True:
    if man.hitbox[1] < t3.hitbox[1] + t3.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t3.hitbox[1]:
        man.y = 365
```

```
if t4.visible == True:
    if man.hitbox[1] < t4.hitbox[1] + t4.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t4.hitbox[1]:
        man.y = 365
```

```
if t5.visible == True:
    if man.hitbox[1] < t5.hitbox[1] + t5.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t5.hitbox[1]:
        man.y = 365
```

```
if t6.visible == True:
    if man.hitbox[1] < t6.hitbox[1] + t6.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t6.hitbox[1]:
        man.y = 365
```

```
if t7.visible == True:
    if man.hitbox[1] < t7.hitbox[1] + t7.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t7.hitbox[1]:
        man.y = 365
```

```
if t8.visible == True:
    if man.hitbox[1] < t8.hitbox[1] + t8.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > t8.hitbox[1]:
        man.y = 365
```

```
if t9.visible == True:
```

```
    if man.hitbox[1] < t9.hitbox[1] + t9.hitbox[3] and man.hitbox[1] +  
man.hitbox[3] > t9.hitbox[1]:  
        man.y = 365
```

```
    if t10.visible == True:  
        if man.hitbox[1] < t10.hitbox[1] + t10.hitbox[3] and man.hitbox[1] +  
man.hitbox[3] > t10.hitbox[1]:  
            man.y = 365
```

```
    if t11.visible == True:  
        if man.hitbox[1] < t11.hitbox[1] + t11.hitbox[3] and man.hitbox[1] +  
man.hitbox[3] > t11.hitbox[1]:  
            man.y = 365
```

```
    if t1.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t2.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t3.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t4.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t5.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t6.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t7.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t8.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t9.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t10.collidet(man.hitbox):  
        man.y -= 32
```

```
    if t11.collidet(man.hitbox):
```

```
man.y -= 32
```

```
#KOTAKSELESAI
```

```
if mb.visible == True:
    if man.hitbox[1] < mb.hitbox[1] + mb.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > mb.hitbox[1]:
        if man.hitbox[0] + man.hitbox[2] > mb.hitbox[0] and man.hitbox[0] <
mb.hitbox[0] + mb.hitbox[2]:
            str(updatescores())
            str(updateFile())
            sounds["win"].play()
            pygame.time.delay(500)
            from quisl3 import quisl3
```

```
if mb.visible == True:
    if man.hitbox[1] < mb.hitbox3[1] + mb.hitbox3[3] and man.hitbox[1] +
man.hitbox[3] > mb.hitbox3[1]:
        if man.hitbox[0] + man.hitbox[2] > mb.hitbox3[0] and man.hitbox[0] <
mb.hitbox3[0] + mb.hitbox3[2]:
            bird.visible = True
```

```
if man.hitbox[1] < bird.hitbox[1] + bird.hitbox[3] and man.hitbox[1] +
man.hitbox[3] > bird.hitbox[1]:
    if man.hitbox[0] + man.hitbox[2] > bird.hitbox[0] and man.hitbox[0] <
bird.hitbox[0] + bird.hitbox[2]:
        man.hit()
        speed2 -= 5
```

```
for bullet in bullets:
    if bullet.y - bullet.radius < bird.hitbox[1] + bird.hitbox[3] and bullet.y +
bullet.radius > bird.hitbox[1]:
        if bullet.x + bullet.radius > bird.hitbox[0] and bullet.x - bullet.radius <
bird.hitbox[0] + bird.hitbox[2]:
            #hitSound.play()
            bird.hit()
            speed2 += 1
            bullets.pop(bullets.index(bullet))
```

```
if bullet.x < 800 and bullet.x > 0:
    bullet.x += bullet.vel
else:
    bullets.pop(bullets.index(bullet))
```

```
keys = pygame.key.get_pressed()
if keys[pygame.K_t] and shootLoop == 0:
```

```

#bulletSound.play()
if man.left:
    facing = -1
else:
    facing = 1

if len(bullets) < 5:
    bullets.append(projectile(round(man.x + 70), round(man.y + 30), 6,
(0,0,0), facing))

shootLoop = 1

for event in pygame.event.get():
    if event.type == pygame.QUIT:
        pygame.quit()
        run = False
    if event.type == USEREVENT+1:
        speed += 0
        speed1 -= 10

keys = pygame.key.get_pressed()
if keys[pygame.K_RIGHT] :
    if event.type == USEREVENT+2:
        r = random.randrange(0,5)
        if r == 1:
            obstacles.append(ceri(1020, 230, 47, 310))
            sounds["ceri"].play()
        elif r == 2:
            obstacles.append(anggur(1020, 230, 47, 310))
            sounds["anggur"].play()
        elif r == 3:
            obstacles.append(monyet(1020, 5, 47, 310))
            sounds["monyet"].play()
        elif r == 4:
            obstacles.append(zebra(1020, 420, 47, 310))
            sounds["zebra"].play()

# tombol
if man.falling == False:
    keys = pygame.key.get_pressed()

    if keys[pygame.K_p]:
        paused()

    elif keys[pygame.K_LEFT] and t1.collidel(man.hitbox):

```

```

    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t2.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t3.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t4.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t5.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1

elif keys[pygame.K_LEFT] and t6.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t7.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t8.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t9.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t10.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_LEFT] and t11.collidel(man.hitbox):
    man.right = False
    man.left = True
    man.x += 1
elif keys[pygame.K_RIGHT] and mb.collider(man.hitbox):
    man.right = True

```

```
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t1.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t2.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t3.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t4.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t5.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t6.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t7.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t8.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t9.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False

elif keys[pygame.K_RIGHT] and t10.collider(man.hitbox):
    man.right = True
    man.left = False
    man.standing = False
elif keys[pygame.K_RIGHT] and t11.collider(man.hitbox):
    man.right = True
    man.left = False
```

```

    man.standing = False
elif keys[pygame.K_LEFT] and man.x > man.vel:
    man.x -= man.vel
    man.left = True
    man.right = False
    man.standing = False
elif keys[pygame.K_RIGHT] and man.x < 500 - man.width - man.vel:
    speed += 1
    bgX -= 0.5
    liX -= 5.5
    man.x += 1
    man.right = True
    man.left = False
    man.standing = False
    j1.x -= j1.vel
    j1.right = True
    j1.left = False
    j2.x -= j1.vel
    j2.right = True
    j2.left = False
    j3.x -= j1.vel
    j3.right = True
    j3.left = False
    j4.x -= j1.vel
    j4.right = True
    j4.left = False
    j5.x -= j1.vel
    j5.right = True
    j5.left = False
    j6.x -= j1.vel
    j6.right = True
    j6.left = False
    j7.x -= j1.vel
    j7.right = True
    j7.left = False
    j8.x -= j1.vel
    j8.right = True
    j8.left = False
    j9.x -= j1.vel
    j9.right = True
    j9.left = False
    j10.x -= j1.vel
    j10.right = True
    j10.left = False
    j11.x -= j1.vel
    j11.right = True

```



```

j11.left = False
t1.x -= j1.vel
t1.right = True
t1.left = False
t2.x -= j1.vel
t2.right = True
t2.left = False
t3.x -= j1.vel
t3.right = True
t3.left = False
t4.x -= j1.vel
t4.right = True
t4.left = False
t5.x -= j1.vel
t5.right = True
t5.left = False
t6.x -= j1.vel
t6.right = True
t6.left = False
t7.x -= j1.vel
t7.right = True
t7.left = False
t8.x -= j1.vel
t8.right = True
t8.left = False
t9.x -= j1.vel
t9.right = True
t9.left = False
t10.x -= j1.vel
t10.right = True
t10.left = False
t11.x -= j1.vel
t11.right = True
t11.left = False
mb.x -= j1.vel
mb.right = True
mb.left = False
for obstacle in obstacles:
    if obstacle.x < -500:
        obstacles.pop(obstacles.index(obstacle))
    else:
        obstacle.x -= 5.5

elif keys[pygame.K_RIGHT] and man.x > 400 - man.width - man.vel:
    speed += 1
    bgX -= 1.0

```

```
liX -= 5.5
man.right = True
man.left = False
man.standing = False
j1.x -= j1.vel
j1.right = True
j1.left = False
j2.x -= j1.vel
j2.right = True
j2.left = False
j3.x -= j1.vel
j3.right = True
j3.left = False
j4.x -= j1.vel
j4.right = True
j4.left = False
j5.x -= j1.vel
j5.right = True
j5.left = False
j6.x -= j1.vel
j6.right = True
j6.left = False
j7.x -= j1.vel
j7.right = True
j7.left = False
j8.x -= j1.vel
j8.right = True
j8.left = False
j9.x -= j1.vel
j9.right = True
j9.left = False
j10.x -= j1.vel
j10.right = True
j10.left = False
j11.x -= j1.vel
j11.right = True
j11.left = False
t1.x -= j1.vel
t1.right = True
t1.left = False
t2.x -= j1.vel
t2.right = True
t2.left = False
t3.x -= j1.vel
t3.right = True
t3.left = False
```

```

t4.x -= j1.vel
t4.right = True
t4.left = False
t5.x -= j1.vel
t5.right = True
t5.left = False
t6.x -= j1.vel
t6.right = True
t6.left = False
t7.x -= j1.vel
t7.right = True
t7.left = False
t8.x -= j1.vel
t8.right = True
t8.left = False
t9.x -= j1.vel
t9.right = True
t9.left = False
t10.x -= j1.vel
t10.right = True
t10.left = False
t11.x -= j1.vel
t11.right = True
t11.left = False
mb.x -= j1.vel
mb.right = True
mb.left = False
for obstacle in obstacles:
    if obstacle.x < -500:
        obstacles.pop(obstacles.index(obstacle))
    else:
        obstacle.x -= 5.5

else:
    man.right = True
    man.left = False
    man.standing = True
    man.walkCount = 0
    j1.right = False
    j1.left = False
    j1.walkCount = 0

if not(man.isJump):
    if keys[pygame.K_SPACE]:
        man.isJump = True
        man.right = False

```

```

        man.left = False
        man.walkCount = 0
    else:
        if man.jumpCount >= -10:
            neg = 1
            if man.jumpCount < 0:
                neg = -1
            man.y -= (man.jumpCount ** 2) * 0.5 * neg
            man.jumpCount -= 1
        else:
            man.isJump = False
            man.jumpCount = 10
    redrawGameWindow()
pygame.quit()

```

Coding Quis

```

import pygame
from pygame.locals import *
from tkinter import *
from tkinter import messagebox
import os

global q,score
q = 1
score = 0

def wrongAns():
    Tk().wm_withdraw()
    messagebox.showinfo('Peringatan!!', 'Jawaban Anda Salah!!!')

def rightAns():
    Tk().wm_withdraw()
    messagebox.showinfo('Peringatan!!', 'Jawaban Anda Benar')
    global score
    score += 1

def checkAns(mousePos):
    if (mousePos[0] in range(160, 420)) and (mousePos[1] in range(410, 482)):
        global q
        if q == 1 or q == 3:
            correctAudio.play()
            rightAns()
        else:
            wrongAudio.play()
            wrongAns()

```

```

    q+=1
elif (mousePos[0] in range(588, 847)) and (mousePos[1] in range(410, 482)):

    if q == 2 or q == 4:
        correctAudio.play()
        rightAns()
    else:
        wrongAudio.play()
        wrongAns()
    q+=1

```

2 – Initialize the game

```

pygame.init()
pygame.mixer.init()
wrongAudio = pygame.mixer.Sound("sound/lose.wav")
correctAudio = pygame.mixer.Sound("sound/win.wav")

```

```

width, height = 1024, 600
screen=pygame.display.set_mode((width, height))

```

3 – Load images

```

bgImage = pygame.image.load("imgs/bg.jpg")
question1 = pygame.image.load("imgs/q1l1.png")
question2 = pygame.image.load("imgs/q2l1.png")
question3 = pygame.image.load("imgs/q3l1.png")
question4 = pygame.image.load("imgs/q4l1.png")
img1 = pygame.image.load(os.path.join('imgs', 'lemon.png'))
img2 = pygame.image.load(os.path.join('imgs', 'pisang.png'))
img3 = pygame.image.load(os.path.join('imgs', 'kelinci.png'))
img4 = pygame.image.load(os.path.join('imgs', 'kucing.png'))

```

```

screen.blit(bgImage, (0, 0))
pygame.display.flip()
Tk().wm_withdraw() # to hide the main window
messagebox.showinfo("Selesaikan Quis", "Mulai QUIZ")

```

4 – Keep looping through
while 1:

```

    # 5 – Clear the screen before drawing it again
    screen.fill(0)

```

```

    # 6 - Draw the screen elements
    if q == 1:

```

```

        screen.blit(question1, (0, 0))
        screen.blit(img1, (350,140))
    elif q == 2:
        screen.blit(question2, (0, 0))
        screen.blit(img2, (350,90))
    elif q == 3:
        screen.blit(question3, (0, 0))
        screen.blit(img3, (380,100))
    elif q == 4:
        screen.blit(question4, (0, 0))
        screen.blit(img4, (350,130))
    elif q > 4:
        if score > 2:
            finalMessage = " Skor anda adalah " + str(score) + " dari 4 Soal"
            Tk().wm_withdraw() # to hide the main window
            messagebox.showinfo("Selamat Ke Level Selanjutnya", finalMessage)
            from lv2 import lv2
        elif score <= 2:
            finalMessage = " Skor anda adalah " + str(score) + " dari 4. " + " Silakan
Ulang Quis!. "
            Tk().wm_withdraw() # to hide the main window
            messagebox.showinfo("Maaf, Skor anda tidak mencukupi", finalMessage)
            score = 0
            q = 1

# 7 – Update the screen
pygame.display.flip()

# 8 – Loop through the events
for event in pygame.event.get():

# Check if the event is the X button
    if event.type==pygame.QUIT:

# If it is quit the game
        Tk().wm_withdraw() # to hide the main window
        messagebox.showinfo('Peringatan!!', 'Apakah Anda Menyerah ??')
        pygame.quit()
        exit(0)
    if event.type == pygame.MOUSEBUTTONDOWN:
        mousePos = pygame.mouse.get_pos()
        checkAns(mousePos)
        break
exit(0)

```

Coding Quis Umum

```
import pygame as pg
import numpy as np
import random
import string
import sys
import os
pg.init()

sw,sh = 1024, 600
sc = (sw/2, sh/2)
screen = pg.display.set_mode((sw,sh))
pg.display.set_caption('QUIS')
default_font = pg.font.SysFont(None, 40)

# Pilihan Warna
colors = {"black":(0,0,0), "darkgray":(70,70,70), "gray":(128,128,128),
"lightgray":(200,200,200), "white":(255,255,255), "red":(255,0,0),
"darkred":(128,0,0), "green":(0,255,0), "darkgreen":(0,128,0),
"blue":(0,0,255), "navy":(0,0,128), "darkblue":(0,0,128),
"yellow":(255,255,0), "gold":(255,215,0), "orange":(255,165,0),
"lilac":(229,204,255), "lightblue":(135,206,250), "teal":(0,128,128),
"cyan":(0,255,255), "purple":(150,0,150), "pink":(238,130,238),
"brown":(139,69,19), "lightbrown":(222,184,135), "lightgreen":(144,238,144),

"turquoise":(64,224,208), "beige":(245,245,220), "honeydew":(240,255,240), "lavender":(230,230,250), "crimson":(220,20,60)}

# images
images =
{"logo":pg.image.load("imgs/logo.png"),0:pg.image.load("imgs/empty.png"),
1:pg.image.load("imgs/v1.png"), 2:pg.image.load("imgs/v2.png"),

3:pg.image.load("imgs/v3.png"),4:pg.image.load("imgs/v4.png"),5:pg.image.load(
"imgs/v5.png"),6:pg.image.load("imgs/v6.png")}
img1 = pg.image.load(os.path.join('imgs', 'gajah.png'))
img2 = pg.image.load(os.path.join('imgs', 'burung.png'))
img3 = pg.image.load(os.path.join('imgs', 'kelinci.png'))
img4 = pg.image.load(os.path.join('imgs', 'sapi.png'))
img5 = pg.image.load(os.path.join('imgs', 'zebra.png'))
img6 = pg.image.load(os.path.join('imgs', 'kucing.png'))
img7 = pg.image.load(os.path.join('imgs', 'lemon.png'))
img8 = pg.image.load(os.path.join('imgs', 'apel.png'))
img9 = pg.image.load(os.path.join('imgs', 'nanas.png'))
img10 = pg.image.load(os.path.join('imgs', 'pisang.png'))
img11 = pg.image.load(os.path.join('imgs', 'semangka.png'))
```

```

img12 = pg.image.load(os.path.join('imgs', 'monyet.png'))
img13 = pg.image.load(os.path.join('imgs', 'buaya.png'))
img14 = pg.image.load(os.path.join('imgs', 'jerapah.png'))
img15 = pg.image.load(os.path.join('imgs', 'singa.png'))
img16 = pg.image.load(os.path.join('imgs', 'stroberi.png'))
img17 = pg.image.load(os.path.join('imgs', 'mangga.png'))
img18 = pg.image.load(os.path.join('imgs', 'ceri.png'))
img19 = pg.image.load(os.path.join('imgs', 'kelapa.png'))
img20 = pg.image.load(os.path.join('imgs', 'anggur.png'))

# music
sounds = {"win":pg.mixer.Sound("sound/win.wav"),
"lose":pg.mixer.Sound("sound/lose.wav"),
         "click":pg.mixer.Sound("sound/click.wav")}

alphabet = list(string.ascii_uppercase)
class Button(object):
    def __init__(self, color, pos, width, height, letter, active = False, type = 1, size
= 40):
        self.type = type
        self.active = active
        self.clicked = False
        self.rollOver = False
        self.size = size
        self.font = pg.font.SysFont(None, self.size)
        self.color = color
        self.letter = letter
        self.pos = pos
        self.width = width
        self.height = height
        self.subsurface = pg.Surface((self.width, self.height))
        self.subsurface.fill(self.color)
        self.text = self.font.render(self.letter, True, colors["white"])

    def Draw(self, surface):
        if self.type == 1:
            if self.rollOver:
                self.subsurface.set_alpha(200)
            else:
                self.subsurface.set_alpha(255)
            if not self.clicked:
                surface.blit(self.subsurface, self.pos)
                self.subsurface.blit(self.text, (self.width/4,self.height/5))
        if self.type == 2:
            if self.active:
                self.subsurface.set_alpha(255)

```



```

        else:
            self.subsurface.set_alpha(100)
            surface.blit(self.subsurface, self.pos)
            self.subsurface.blit(self.text, (self.width / 4, self.height / 5))

notesArea = pg.Surface((sw,700))
notesArea.fill(colors["beige"])
buttonArea = pg.Surface((600, 100))
buttonArea.fill(colors["lavender"])

letters = []
j = 5
for number, letter in enumerate(alphabet):
    if number > 12:
        number = number - 13
        j = 6
    letters.append(Button(colors["gray"], (70+number*70,140+j*60), 40, 40,
letter))

languageButtons = []
languageButtons.append(Button(colors["gray"], (680, 205), 80,40, "Hewan",
False, 2, 20))
languageButtons.append(Button(colors["gray"], (580, 205), 80,40, "Buah", True,
2, 20))
errorCount = 0
needRestart = False
winCount = 0
f = open('scores.txt','r')
file = f.readlines()
pointCount = int(file[0])
spaceCount = 0
# Hewan
wordsH =
["BIRD", "ELEPHANT", "RABBIT", "CAT", "COW", "ZEBRA", "CROCODILE", "
GIRAFFE", "MONGKEY", "LION"]

# Buah
wordsB =
["LIME", "PINEAPLLE", "WATERMELON", "BANANA", "APPLE", "GRAPE", "C
HERRY", "COCONUT", "MANGO", "STRAWBERRY"]

languageChoice = 1
if languageChoice == 1:
    currentLanguage = wordsB
else:
    currentLanguage = wordsH

```

```

currentWord = random.randrange(0, len(currentLanguage))

guessed = []

lw = 40
ls = 10

def updateFile():
    f = open('hgscores.txt', 'r')
    file = f.readlines()
    last = int(file[0])
    if last < int(pointCount):
        f.close()
        file = open('hgscores.txt', 'w')
        file.write(str(pointCount))
        file.close()
    return pointCount
    return last

for letter in currentLanguage[currentWord]:
    if letter == " ":
        spaceCount += 1
print(len(wordsH))
print(len(wordsB))

while True:
    for event in pg.event.get():
        if event.type == pg.QUIT:
            pg.quit()
            sys.exit()
        if event.type == pg.MOUSEMOTION:
            for button in letters:
                currentRect = button.subsurface.get_rect(topleft = (button.pos[0],
button.pos[1]))
                if currentRect.collidepoint(pg.mouse.get_pos()):
                    button.rollOver = True
                else:
                    button.rollOver = False
        if event.type == pg.MOUSEBUTTONDOWN:
            if event.button == 1:
                for button in letters:
                    if button.rollOver == True and button.clicked == False:
                        sounds["click"].play()
                        button.clicked = True
                        guessed.append(button.letter)
                        noError = False

```

```

        for letter in currentLanguage[currentWord]:
            if button.letter == letter:
                noError = True
            if errorCount < 6 and not noError:
                errorCount += 1
        for button in languageButtons:
            currentRectLang = button.subsurface.get_rect(topleft =
(button.pos[0], button.pos[1]))
            if currentRectLang.collidepoint(pg.mouse.get_pos()):
                button.active = True
                sounds["click"].play()
                if button.letter == "Hewan":
                    currentLanguage = wordsH
                else:
                    currentLanguage = wordsB
            currentIndex = languageButtons.index(button)
            for subbutton in languageButtons:
                if languageButtons.index(subbutton) == currentIndex:
                    pass
                else:
                    subbutton.active = False
                    needRestart = True

screen.fill(colors["white"])
screen.blit(notesArea, (0,0))
screen.blit(buttonArea, (360, 300))

screen.blit(images["logo"], (sc[0]-images["logo"].get_rect().width/2, 10))
screen.blit(images[errorCount], (sc[0]-
images[errorCount].get_rect().width/2+370, sc[1]-
images[errorCount].get_rect().height/2-280))

for letter in letters:
    letter.Draw(screen)

for langbut in languageButtons:
    langbut.Draw(screen)

largeFont = pg.font.SysFont('comicsans', 80)
lastScore = largeFont.render('Score: ' + str(pointCount),1,(0,0,0))
stats_font = pg.font.SysFont(None, 25, False, True)
winCountText = stats_font.render("TOTAL WINS      : " + str(winCount),
True, colors["black"])
pointCountText = stats_font.render("TOTAL POINTS   : " + str(pointCount),
True, colors["black"])

```

```
Text = stats_font.render("Dengan Menggunakan Bahasa Inggris", True,
colors["black"])
screen.blit(winCountText, (800, 200))
screen.blit(Text, (350, 100))
screen.blit(pointCountText, (800, 230))

if (currentLanguage[currentWord]) == "ELEPHANT":
    screen.blit(img1, (0,100))

if (currentLanguage[currentWord]) == "BIRD":
    screen.blit(img2, (0,100))

if (currentLanguage[currentWord]) == "RABBIT":
    screen.blit(img3, (0,100))

if (currentLanguage[currentWord]) == "COW":
    screen.blit(img4, (0,100))

if (currentLanguage[currentWord]) == "ZEBRA":
    screen.blit(img5, (0,100))

if (currentLanguage[currentWord]) == "CAT":
    screen.blit(img6, (0,200))

if (currentLanguage[currentWord]) == "LIME":
    screen.blit(img7, (0,200))

if (currentLanguage[currentWord]) == "APPLE":
    screen.blit(img8, (0,100))

if (currentLanguage[currentWord]) == "PINEAPLLE":
    screen.blit(img9, (0,100))

if (currentLanguage[currentWord]) == "BANANA":
    screen.blit(img10, (0,100))

if (currentLanguage[currentWord]) == "WATERMELON":
    screen.blit(img11, (0,100))

if (currentLanguage[currentWord]) == "MONGKEY":
    screen.blit(img12, (0,100))

if (currentLanguage[currentWord]) == "CROCODILE":
    screen.blit(img13, (0,100))
```

```

if (currentLanguage[currentWord]) == "GIRAFFE":
    screen.blit(img14, (0,100))

if (currentLanguage[currentWord]) == "LION":
    screen.blit(img15, (0,100))

if (currentLanguage[currentWord]) == "STRAWBERRY":
    screen.blit(img16, (0,100))

if (currentLanguage[currentWord]) == "MANGO":
    screen.blit(img17, (0,100))

if (currentLanguage[currentWord]) == "CHERRY":
    screen.blit(img18, (0,100))

if (currentLanguage[currentWord]) == "COCONUT":
    screen.blit(img19, (0,100))

if (currentLanguage[currentWord]) == "GRAPE":
    screen.blit(img20, (0,100))

totalShown = 0
if not needRestart:
    for i,letter in enumerate(currentLanguage[currentWord]):
        text = default_font.render(letter, True, colors["black"])
        posX = (1300 - len(currentLanguage[currentWord]) * (lw + ls))/2 + i * (lw
+ ls)
        posY = 350
        if letter != " ":
            pg.draw.rect(screen, colors["black"], (posX, posY, lw, 3))
        else:
            pg.draw.rect(screen, colors["lavender"], (posX, posY, lw, 3))
        if letter in guessed:
            totalShown += 1
            screen.blit(text, (posX+lw/3, posY-30))

pg.display.update()
final_font = pg.font.SysFont(None, 80)
lose_text = final_font.render("Kamu Kalah", True, colors["darkred"])
win_text = final_font.render("Kamu Menang", True, colors["darkgreen"])

if errorCount >= 6 or needRestart:
    if not needRestart:
        sounds["lose"].play()
        screen.blit(lose_text, (350,280))
        screen.blit(lastScore, (sw/2 - lastScore.get_width()/2,350))

```

```

        str(updateFile())
        pg.display.update()
        pg.time.wait(3000)

    pg.time.wait(1000)
    pg.quit()
    sys.exit()
if totalShown == len(currentLanguage[currentWord]) - spaceCount: #menang
    sounds["win"].play()
    screen.blit(win_text, (310, 280))
    pg.display.update()
    pg.time.wait(1000)
    guessed.clear()
    pointCount += 600 + winCount*10 - errorCount * 100 # penghitungan Poin
    errorCount = 0
    winCount+=1

    for letter in letters:
        letter.clicked = False
    currentWord = random.randrange(0, len(currentLanguage))
    spaceCount = 0
    for letter in currentLanguage[currentWord]:
        if letter == " ":
            spaceCount += 1

    pg.time.wait(1000)

```

Lampiran 2: Permohonan Izin Penelitian

20/11/2019

lemlit.ichsan/lemlit/cetak-surat-penelitian-mahasiswa/1760/



**KEMENTERIAN RISET, TEKNOLOGI DAN PENDIDIKAN TINGGI
LEMBAGA PENELITIAN (LEMLIT)
UNIVERSITAS ICHSAN GORONTALO**

Jl. Raden Saleh No. 17 Kota Gorontalo
Telp: (0435) 8724466, 829975; Fax: (0435) 82997;
E-mail: lembagapenelitian@unisan.ac.id

Nomor : 1873/PIP/LEMLIT-UNISAN/GTO/XI/2019
Lampiran : -
Hal : Permohonan Izin Penelitian

Kepada Yth,
Kepala Sekolah SD Negeri 2 Tolangohula
di,-
Gorontalo

Yang bertanda tangan di bawah ini :

Nama : Dr. Rahmisyari, ST., SE
NIDN : 0929117202
Jabatan : Ketua Lembaga Penelitian

Meminta kesediannya untuk memberikan izin pengambilan data dalam rangka penyusunan **Proposal / Skripsi**, kepada :

Nama Mahasiswa : Nuriman Hamka Genti
NIM : T3116043
Fakultas : Fakultas Ilmu Komputer
Program Studi : Teknik Informatika
Lokasi Penelitian : SD Negeri 2 Tolangohula
Judul Penelitian : GAME PETUALANG EDUKASI BAHASA INGGRIS PADA ANAK SEKOLAH DASAR

Atas kebijakan dan kerja samanya diucapkan banyak terima kasih.

Gorontalo, 20 November 2019
Ketua

Dr. Rahmisyari, ST., SE
NIDN 0929117202

Lampiran 3: Surat Keterangan Telah Melakukan Penelitian



PEMERINTAH KABUPATEN GORONTALO
DINAS PENDIDIKAN DAN KEBUDAYAAN
SDN 2 TOLANGOHULA
KECAMATAN TOLANGOHULA KABUPATEN GORONTALO
Jalan Siswa Desa Lakeya Kecamatan Tolangohula Kode Pos 96222

SURAT KETERANGAN

Nomor : 421.01/SDN 2 TOL/75/VIII/2020

Yang bertanda tangan di bawah ini, Kepala Sekolah Dasar Negeri 2 Tolangohula Kecamatan Tolangohula Kabupaten Gorontalo, menerangkan kepada :

Nama : NURIMAN HAMKA GENTI
NIM : T3116043
Fakultas : Ilmu Komputer
Program Studi : Teknik Informatika
Alamat : Desa Lakeya Kec. Tolangohula Kab. Gorontalo

Bahwa yang bersangkutan benar-benar mengambil lokasi penelitian di SD Negeri 2 Tolangohula berdasarkan Permohonan Izin Penelitian dari Ketua Lembaga Penelitian Universitas Ichsan Gorontalo, Nomor : 1873/PIP/LEMLIT-UNISAN/GTO/XI/2019, tanggal 20 November 2019 dalam rangka penyusunan Skripsi dengan judul GAME PETUALANG EDUKASI BAHASA INGGRIS PADA ANAK SEKOLAH DASAR.

Demikian Surat Keterangan ini dibuat untuk dipergunakan seperlunya.

Tolangohula, Agustus 2020

Kepala Sekolah,



DIANA NGAU, S.Pd

NIP 19640606 198406 2 005

Lampiran 4: Bukti Penerimaan Softcopy Skripsi



Pustikom
Universitas Ichsan Gorontalo

BUKTI PENERIMAAN SOFTCOPY SKRIPSI **PENGECEKAN SIMILARITY TURNITIN**

Nama Mahasiswa : NURIMAN HAMKA GENTI
NIM : T3116043
Program Studi : Teknik Informatika (S1)
Fakultas : Fakultas Ilmu Komputer
Judul Skripsi : Game Petualang Edukasi Bahasa Inggris Pada Anak Sekolah Dasar

Nama File (Pdf) : _____

No. HP/WA : 082237470978

e-Mail : _____

Tgl. Terima :

--	--	--	--	--	--

Hasil Pengecekan :

--	--	--	--	--	--

Diterima/Diperiksa Oleh,


Sudirman S. Panna, M.Kom
085340910769

Lampiran 5: Surat Pernyataan Pengecekan Kembali



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN UNIVERSITAS ICHSAN (UNISAN) GORONTALO

SURAT KEPUTUSAN MENDIKNAS RI NOMOR 84/D/O/2001
Jl. Achmad Nadjamuddin No. 17 Telp (0435) 829975 Fax (0435) 829976 Gorontalo

SURAT PERNYATAAN

Yang bertanda tangan di bawah ini :

- | | |
|---------|--------------------------|
| 1. Nama | : Irvan A. Salihi, M.Kom |
| Sebagai | : Pembimbing I |
| 2. Nama | : Muis Nanja, M.Kom |
| Sebagai | : Pembimbing II |

Dengan ini Menyatakan bahwa :

- | | |
|----------------|---|
| Nama Mahasiswa | : NURIMAN HAMKA GENTI |
| NIM | : T3116043 |
| Program Studi | : Teknik Informatika (S1) |
| Fakultas | : Fakultas Ilmu Komputer |
| Judul Skripsi | : Game Petualang Edukasi Bahasa Inggris Pada Anak Sekolah Dasar |

Setelah kami melakukan pengecekan kembali antara softcopy skripsi dari hasil pemeriksaan aplikasi Turnitin dengan hasil Similarity sebesar 15% oleh Tim Verifikasi Plagiasi di Pustikom dengan Skripsi Aslinya, isinya SAMA dan format penulisannya sudah sesuai dengan Buku Panduan Penulisan Skripsi.

Demikian surat pernyataan ini dibuat untuk mendapatkan Surat Rekomendasi Bebas Plagiasi.

Pembimbing I

Irvan A. Salihi, M.Kom
NIDN. 0928028101

Gorontalo, Juli 2020

Pembimbing II

Muis Nanja, M.Kom
NIDN. 0905078703

Mengetahui
Ketua Program Studi,

Irvan A. Salihi, M.Kom
NIDN. 0928028101

Lampiran 6: Surat Rekomendasi Bebas Plagiasi



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN
(UNISAN) GORONTALO**

SURAT KEPUTUSAN MENDIKNAS RI NOMOR 84/D/O/2001
Jl. Achmad Nadjamuddin No. 17 Telp (0435) 829975 Fax (0435) 829976 Gorontalo

SURAT REKOMENDASI BEBAS PLAGIASI

No. 0433/UNISAN-G/S-BP/IV/2020

Yang bertanda tangan di bawah ini:

Nama : Sunarto Taliki, M.Kom
NIDN : 0906058301
Unit Kerja : Pustikom, Universitas Ichsan Gorontalo

Dengan ini Menyatakan bahwa :

Nama Mahasiswa : NURIMAN HAMKA GENTI
NIM : T3116043
Program Studi : Teknik Informatika (S1)
Fakultas : Fakultas Ilmu Komputer
Judul Skripsi : Game Petualang Edukasi Bahasa Inggris Pada Anak Sekolah Dasar

Sesuai dengan hasil pengecekan tingkat kemiripan skripsi melalui aplikasi Turnitin untuk judul skripsi di atas diperoleh hasil Similarity sebesar 15%, berdasarkan SK Rektor No. 237/UNISAN-G/SK/IX/2019 tentang Panduan Pencegahan dan Penanggulangan Plagiarisme, bahwa batas kemiripan skripsi maksimal 35% dan sesuai dengan Surat Pernyataan dari kedua Pembimbing yang bersangkutan menyatakan bahwa isi softcopy skripsi yang diolah di Turnitin SAMA ISINYA dengan Skripsi Aslinya serta format penulisannya sudah sesuai dengan Buku Panduan Penulisan Skripsi, untuk itu skripsi tersebut di atas dinyatakan BEBAS PLAGIASI dan layak untuk diujikan.

Demikian surat rekomendasi ini dibuat untuk digunakan sebagaimana mestinya.

Gorontalo, 26 Juli 2020
Tim Verifikasi,



Sunarto Taliki, M.Kom
NIDN. 0906058301

Terbusan :

1. Dekan
2. Ketua Program Studi
3. Pembimbing I dan Pembimbing II
4. Yang bersangkutan
5. Arsip

Lampiran 7: Surat Keterangan Bebas Pustaka



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
UPT. PERPUSTAKAAN PUSAT**

SURAT KEPUTUSAN MENDIKS RI NO. 84/D/0/2001
Jln. Achmad Nadjamuddin No.17 Telp. (0435) 829975 Fax. (0435) 829976 Gorontalo

SURAT KETERANGAN BEBAS PUSTAKA

No.005/perpus_fikom/VIII/2020

Perpustakaan Rabu, 19 Agustus 2020 Fakultas Ilmu komputer (FIKOM) Universitas Ichsan Gorontalo dengan ini menerangkan bahwa :

Nama : Nuriman Hamka Genti
Nim : T3116043
No anggota : M2016342

Terhitung sejak tanggal 19 Agustus 2020, dinyatakan telah bebas dari pinjaman buku dan koleksi lainnya dipergustakaan Fakultas Ilmu komputer.

Demikian keterangan ini di buat untuk di gunakan sebagaimana mestinya.

Gorontalo, 19 Agustus 2020
Kepala Perpustakaan
Fakultas Ilmu Komputer

Apriyanto Alhamad, M.Kom
NIDN 09240486

Lampiran 8: Biografi Penulis

Riwayat Hidup Peneliti



Nama : Nuriman Hamka Genti

Tempat, Tgl Lahir : Boliyohuto, 22 September 1997

Pekerjaan : Mahasiswa

Email : imangenti.nur02@gmail.com

Riwayat Pendidikan:

1. TK Melati PT.PG Tolangohula (2003)
2. SD Negeri 9 Tolangohula (2004 - 2010)
3. SMP Negeri 1 Tolangohula (2010 - 2013)
4. SMA Negeri 1 Tolangohula (2013 - 2016)
5. Universitas Ichsan Gorontalo (2016 - Sekarang)



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan Mahasiswa

Nama : Nuriman Hamka Genti

NIM : T3116043

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Minggu / 02-Agustus-2020

Waktu : 13.00-15.00 Wita

Tempat :

Maka dapat disimpulkan saran-saran sebagai berikut:

Perbaiki penulisan skripsi
Sesuaikan skripsi dan program
Perhatikan penggunaan gambar di program.

Demikian saran perbaikan ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 02-Agustus-2020

Dosen Penguji

Asmaul Husna, M.Kom



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO**

FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan Mahasiswa

Nama : Nuriman Hamka Genti

NIM : T3116043

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Minggu / 02-Agustus-2020

Waktu : 13.00-15.00 Wita

Tempat :

Maka dapat disimpulkan saran-saran sebagai berikut:

1. Perbaiki kesimpulan, sesuaikan dengan rumusan masalah dan tujuan penelitian
2. flowchart pada pengujian whitebox diperbaiki, sesuaikan dgn coding program yang digunakan

Demikian saran perbaikan ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 02-Agustus-2020

Dosen Penguji

Sudirman S. Panna, M.Kom



**KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO**

FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan Mahasiswa

Nama : Nuriman Hamka Genti

NIM : T3116043

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Minggu / 02-Agustus-2020

Waktu : 13.00-15.00 Wita

Tempat :

Maka dapat disimpulkan saran-saran sebagai berikut:

Demikian saran perbaikan ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 02-Agustus-2020

Dosen Penguji

Andi Bode, M.Kom



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan Mahasiswa

Nama : Nuriman Hamka Genti

NIM : T3116043

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Minggu / 02-Agustus-2020

Waktu : 13.00-15.00 Wita

Tempat :

Maka dapat disimpulkan saran-saran sebagai berikut:

- gambar yang muncul tiap level di bagi saja sesuai dengan tingkat penyebutan
- atau gunakan tetap random, tetapi gambar yang muncul jika ada yang sama dalam level yang sama maka di skip di ganti dengan gambar lainnya.

Demikian saran perbaikan ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 02-Agustus-2020

Dosen Penguji

Irvan Abraham Salihi, M.Kom



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO
FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

SARAN-SARAN PERBAIKAN

Berdasarkan Mahasiswa

Nama : Nuriman Hamka Genti

NIM : T3116043

Program Studi : Teknik Informatika

Yang diselenggarakan pada

Hari/Tanggal : Minggu / 02-Agustus-2020

Waktu : 13.00-15.00 Wita

Tempat :

Maka dapat disimpulkan saran-saran sebagai berikut:

1. bagusnya ditambahkan penambahan level
 2. gamenya harusnya dalam bentuk exe
 3. tingkatkan kualitas obejek atau buat objek lebih menarik
 4. soal kuis harus sesuai dengan objek yang dilewati

Demikian saran perbaikan ini dibuat dengan sebenarnya untuk dapat dipergunakan sebagaimana mestinya.

Gorontalo, 02-Agustus-2020

Dosen Penguji



Muis Nanja, M.Kom



KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN
UNIVERSITAS ICHSAN GORONTALO

FAKULTAS ILMU KOMPUTER

SK MENDIKNAS NOMOR 84/D/O/2001

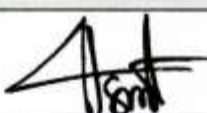
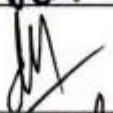
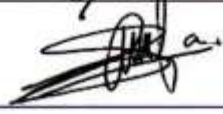
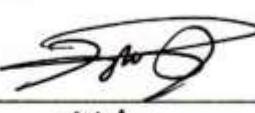
JL. Achmad Nadjamuddin No. 17 Telp. (0435) 829975 Fax (0435) 829976 Gorontalo

Berita Acara Perbaikan/Revisi Ujian

Pada hari ini, Minggu 02-Agustus-2020, Pukul 13.00-15.00 Wita. Telah dilaksanakan Ujian mahasiswa Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Nama : Nuriman Hamka Genti
Nim : T3116043
Pembimbing I : Irvan Abraham Salihi, M.Kom
Pembimbing II : Muis Nanja, M.Kom
Judul : Game Petualang Edukasi Bahasa Inggris Pada Anak Sekolah Dasar

Oleh Komite Seminar sebagai berikut :

No	Komite Seminar	Status	Tanda Tangan
1	Asmaul Husna, M.Kom	Ketua	
2	Sudirman S. Panna, M.Kom	Anggota	
3	Andi Bode, M.Kom	Anggota	
4	Irvan Abraham Salihi, M.Kom	Anggota	
5	Muis Nanja, M.Kom	Anggota	