

RANCANG BANGUN GAME SIMULASI TRUK PETIKEMAS 3D

SKRIPSI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun Oleh:

WIDA PRAPONCO SUBAGYO

NIM. 0910683095

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

UNIVERSITAS BRAWIJAYA

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

MALANG

2014

LEMBAR PERSETUJUAN

RANCANG BANGUN GAME SIMULASI TRUK PETIKEMAS 3D

SKRIPSI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer



Disusun Oleh:

WIDA PRAPONCO SUBAGYO

NIM. 0910683095

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Eriq M. Adams J., S.T., M.Kom.
NIK. 850410 06 1 1 0027

Denny Sagita Rusdianto, S.Kom, M.Kom.
NIK. 851124 06 1 1 0250



KATA PENGANTAR

Alhamdulillah,

Segala puji syukur penulis ucapkan atas kehadiran Allah SWT dengan seluruh limpahan rahmat serta ridhoNya, penulis dapat menyelesaikan skripsi ini. Dengan terselesaikannya skripsi ini tentu juga ada dukungan berupa semangat dan doa dari orang-orang yang berada di sekeliling penulis yang telah terlibat secara langsung maupun tidak langsung. Oleh karena itu penulis mengucapkan terima kasih yang tidak terhingga kepada :

1. Kedua orang tua Bpk. Subagyo dan Ibu Endang Werdiningsih, Andyanto dan Maleona Sarah yang terus menyemangati penulis untuk menyelesaikan skripsi secepat mungkin, meskipun jauh dari tepat waktu.
2. Terima kasih yang tak terhingga kepada Dosen Pembimbing Eriq M. Adams J., S.T., M.Kom. dan Denny Sagita Rusdianto, S.Kom, M.Kom., yang telah membimbing penulis selama 3 semester ini serta tidak lupa mengucapkan terima kasih kepada seluruh dosen dan para staf Program Teknologi Informasi dan Ilmu Komputer dalam membantu penulis hingga di akhir-akhir semester ini.
3. Terima kasih untuk sahabat-sahabat penulis, Ardi, Evan, Heri, Iput dan semua anggota B20, terima kasih juga untuk teman jurusan seperjuangan yang telah berjuang bersama hingga menyelesaikan studi di jenjang perguruan tinggi ini yang tidak bisa penulis sebutkan satu per-satu.

Semoga Allah SWT membalas kebaikan dan ketulusan semua pihak yang telah membantu menyelesaikan skripsi ini dengan melimpahkan rahmat dan karunia-Nya dan semoga karya penelitian tugas akhir ini dapat memberikan manfaat dan kebaikan bagi banyak pihak demi kemaslahatan bersama serta bernilai ibadah di hadapan Allah SWT. Amin.

Malang, Januari 2014

Penulis

ABSTRAK

Wida Praponco Subagyo. 2013. Rancang Bangun Game Simulasi Truk Petikemas 3D. Skripsi Program Studi Teknik Informatika. Program Teknologi Informasi dan Ilmu Komputer. Universitas Brawijaya. Malang. Pembimbing : Eriq Muhammad Adams J., ST., M.Kom dan Denny Sagita Rusdianto S.Kom., M.Kom.

Simulasi kendaraan merupakan *game* yang mensimulasikan pilot atau menyetir sebuah kendaraan. Operator truk merupakan seorang yang menjalankan transportasi petikemas dari dermaga pelabuhan menuju terminal atau sebaliknya. Dalam menjalankan prosedur pelaksanaan operator truk memiliki beberapa instruksi kerja. Instruksi kerja merupakan peraturan atau tata cara beroperasi di dalam terminal petikemas. Peraturan yang dibuat digunakan untuk menunjang keselamatan kerja dan menjaga serta menertibkan agar meminimalisir kejadian yang tidak diharapkan di area terminal petikemas. Permasalahan ini diambil untuk merancang *gameplay* dari *game* simulasi ini. Oleh karena itu, dibutuhkan sebuah alternatif yang dapat membantu pemahaman diri dan memberikan pengalaman lebih dengan menggambarkan suatu aturan serta pengenalan lingkungan dalam sebuah media yakni melalui *game* simulasi yang dirancang menjadi sebuah *game* yang berjudul “Game Simulasi Truk Petikemas 3D”. Metode dan perancangan *game* ini memerlukan 3 fase yaitu observasi, perancangan dan implementasi. Ketiga fase tersebut diperlukan dalam menyelesaikan *game* simulasi ini, dengan memberikan pemodelan bentuk 3D dan peraturan seperti pada lingkungan sebenarnya. Dari hasil pengujian yang dilakukan, mulai dari uji validitas, unit, integrasi, dan performa didapatkan kesimpulan bahwa *game* simulasi ini telah memenuhi kebutuhan fungsional dan non fungsional serta menghasilkan persyaratan minimum spesifikasi perangkat keras untuk mengeksekusi *game* ini dengan mengambil tingkat performa berdasarkan FPS (*Frames per Second*).

Kata Kunci : Simulasi Kendaraan, *Game*, Petikemas, Simulasi, 3D

ABSTRACT

Wida Praponco Subagyo. 2013. *Design and Build 3D Container Truck Simulation Game. Undergraduate Thesis of Informatic Engineering Study Program, Information Technology and Computer Science Program, Brawijaya University, Malang. Advisor : Eriq Muhammad Adams J., ST., M.Kom and Denny Sagita Rusdianto S.Kom., M.Kom.*

Vehicle Simulation is a game that simulate pilot/driving a vehicle. Truck operator is a man who drive the container transportation truck from the dock to the port terminal or vice versa. In carrying out the procedures for implementing, truck operators have some work instructions. Work instructions are operating procedures at the container terminal. Regulations made used to support and maintain the safety and discipline in order to minimize unexpected events in the area of container terminal. Therefore , it takes an alternative that can help provide an understanding and give more experience with describe as well as the introduction the rules and the environment in a media that is through a simulation game designed into a game entitled " Container Truck Simulation Games in 3D ". Methods and design of this game requires three phases there are observation, design and implementation. All three phases are necessary in completing this simulation game, by providing 3D modeling and regulations as in the actual environment. As a result from testing, ranging from the validity test , unit , integration , and performance, it can be concluded that this simulation game has fulfilled the functional and non functional requirements and produce the minimum specification hardware requirements to execute the game by taking the level of performance based FPS (Frames per Second).

Keywords : *Vehicle Simulation, Game, Container, Simulation, 3D*

DAFTAR ISI

KATA PENGANTAR	I
ABSTRAK	II
ABSTRACT	III
DAFTAR ISI	IV
DAFTAR GAMBAR	VII
DAFTAR TABEL	IX
DAFTAR LAMPIRAN	XI
BAB I	1
PENDAHULUAN	1
1.1. LATAR BELAKANG	1
1.2. RUMUSAN MASALAH	4
1.3. BATASAN MASALAH	4
1.4. TUJUAN PENELITIAN	5
1.5. MANFAAT	5
1.6. SISTEMATIKA PENULISAN	5
BAB II	7
LANDASAN TEORI	7
2.1. GAME SIMULASI	8
2.2. SIMULASI KENDARAAN	8
2.3. PENGEMBANGAN KARAKTER MELALUI SIMULASI	8
2.4. PEMODELAN SIMULASI DALAM SEBUAH GAME	9
2.5. GAME ENGINE (WAVEENGINE)	10
2.6. BLENDER	12
2.7. PENGUJIAN PERANGKAT LUNAK	14
2.7.1. White-Box Testing	14
2.7.2. Black-Box Testing	15
2.8. PENGUJIAN PERFORMA	15
BAB III	16
METODOLOGI DAN PERANCANGAN	16
3.1. FASE OBSERVASI	17
3.1.1. Studi Literatur	17
3.1.2. Pengumpulan Data	17
3.2. FASE PERANCANGAN	18
3.2.1. Game Concept Design	19
3.2.1.1. Main Game Concept	19

3.2.1.2. Kebutuhan Teknologi	20
3.2.1.3. Kontrol Permainan	20
3.2.1.4. Sudut Kamera	22
3.2.1.5. Alur Permainan	22
3.2.1.6. Antarmuka	23
3.2.1.7. Art Concept	25
3.2.1.8. Low Polygon Modeling	25
3.2.1.9. UV Layout	26
3.2.1.10. Pemberian Tekstur	26
3.2.1.11. Desain HUD	27
3.2.2. <i>Concept Level Design</i>	27
3.2.2.1. Game Goals Level	27
3.2.2.2. Story	28
3.2.2.3. Gameplay Description	29
3.2.2.4. Penempatan Aset	32
3.2.2.5. Level Design Dalam Game Simulasi Truk Petikemas 3D	33
3.2.3. <i>Technical Design</i>	33
3.2.3.1. Use Case	34
3.2.3.2. Identifikasi Aktor	34
3.2.3.3. Daftar Kebutuhan	34
3.2.3.4. Diagram Use Case	36
3.2.3.5. Diagram Class	37
3.2.3.6. Diagram Activity	42
3.3. IMPLEMENTASI	43
3.4. PENGUJIAN	43
3.5. PENULISAN LAPORAN SKRIPSI	44
BAB IV	45
IMPLEMENTASI	45
4.1. OBJEK INTEGRASI DI LUAR WAVEENGINE	46
4.1.1. <i>Pengumpulan Data</i>	46
4.1.2. <i>Modeling</i>	46
4.1.3. <i>Musik Latar</i>	47
4.1.4. <i>Ekspor</i>	47
4.2. OBJEK INTEGRASI DI DALAM WAVEENGINE	48
4.2.1. <i>Implementasi Prosedur Program</i>	48
4.2.1.1. Implementasi Akselerasi Truk	48
4.2.1.2. Implementasi Kendali Roda Truk	49
4.2.1.3. Implementasi Pemasangan Chassis	50
4.2.1.4. Implementasi Angkut Petikemas	51
4.2.1.5. Implementasi Nilai Ukur Kerusakan Mesin	52
4.2.1.6. Implementasi Nilai Ukur Jumlah Bahan Bakar	53
4.2.2. <i>Implementasi Kontrol Permainan</i>	54
4.2.2.1. Mouse	54

4.2.2.2. Keyboard.....	54
4.2.2.3. Joystick	55
4.2.3. <i>Implementasi Kamera</i>	55
4.2.4. <i>Implementasi Alur Permainan</i>	56
4.2.5. <i>Implementasi Collision Detection</i>	57
4.2.6. <i>Implementasi Frustum Culling</i>	57
4.2.7. <i>Implementasi HUD</i>	59
4.2.8. <i>Implementasi Aset-Aset</i>	61
4.2.9. <i>Implementasi Integrasi</i>	61
BAB V	632
PENGUJIAN DAN ANALISIS	632
5.1. PENGUJIAN VALIDITAS	63
5.2. PENGUJIAN UNIT	63
5.2.1. <i>Prosedur Pergerakan Akselerasi</i>	63
5.2.2. <i>Prosedur Kendali Roda</i>	65
5.2.3. <i>Prosedur Pemasangan Chasis Petikemas</i>	66
5.2.4. <i>Prosedur Angkut Petikemas</i>	67
5.2.5. <i>Prosedur Nilai Ukur Kerusakan Mesin</i>	69
5.2.6. <i>Prosedur Niali Ukur Jumlah Bahan Bakar</i>	70
5.3. PENGUJIAN INTEGRASI	71
5.4. PENGUJIAN PERFORMA	73
5.5. ANALISIS PENGUJIAN VALIDITAS	75
5.6. ANALISIS PENGUJIAN UNIT	76
5.7. ANALISIS PENGUJIAN INTEGRASI	82
5.8. ANALISIS PENGUJIAN PERFORMA	82
BAB VI.....	814
PENUTUP	814
6.1. KESIMPULAN.....	814
6.2. SARAN	85
DAFTAR PUSTAKA	DP-1
LAMPIRAN	L-1

DAFTAR GAMBAR

Gambar 2.1. Screenshot situs *waveengine*..... 11

Gambar 2.2. Blender 2.68 a. 13

Gambar 3.1. Diagram alur produksi *game* secara umum..... 16

Gambar 3.2. Lingkungan terminal petikemas surabaya..... 18

Gambar 3.3. Ilustrasi implementasi kamera orang ketiga..... 22

Gambar 3.4. Diagram alur permainan..... 23

Gambar 3.5. Konsep truk. 25

Gambar 3.6. Uv layout truk 26

Gambar 3.7. Pemberian tekstur truk (diffuse dan normal map) 26

Gambar 3.8. Level design dalam konteks relasi dengan proses kerja lainnya. 32

Gambar 3.9. *Level design* konsep pemodelan lingkungan..... 33

Gambar 3.10. Diagram use case..... 36

Gambar 3.11. Diagram *class – scene* 40

Gambar 3.12. Diagram *class – behavior*..... 41

Gambar 3.13. Diagram *class - helper*..... 41

Gambar 4.1. Diagram alir implementasi..... 47

Gambar 4.2. Pemodelan lingkungan *game*. 47

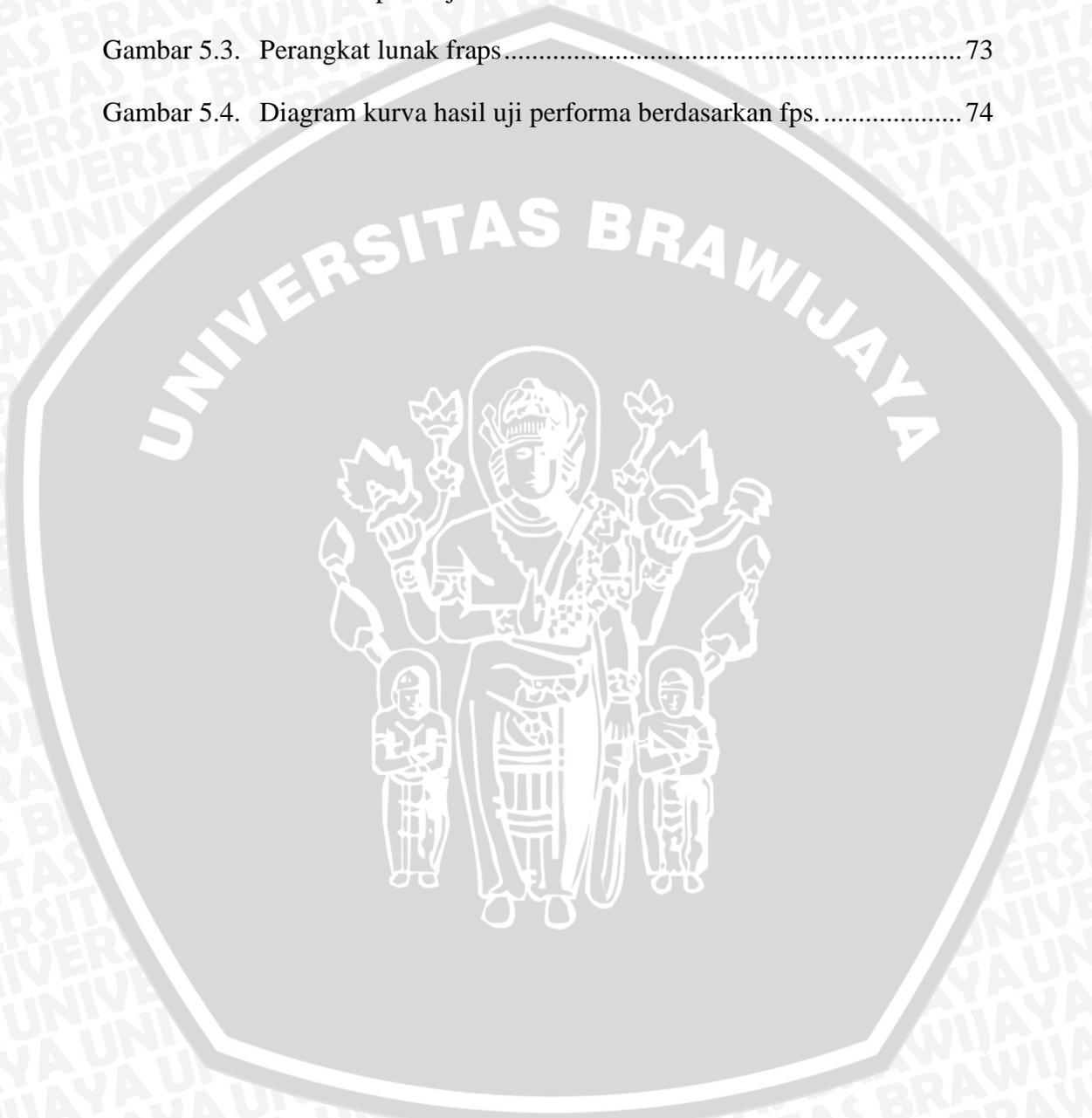
Gambar 4.3. Hasil implementasi penggunaan *third person camera*..... 56

Gambar 4.4. Alur *game* simulasi truk petikemas 3d..... 56

Gambar 4.5. Kondisi *pause* dan *goal*..... 57

Gambar 4.6. Tampilan frustum..... 58

Gambar 4.7. Frustum culling.....	58
Gambar 5.1. Pengujian black box-masukan dan keluaran.....	63
Gambar 5.2. Proses ekspor objek utama.....	71
Gambar 5.3. Perangkat lunak fraps.....	73
Gambar 5.4. Diagram kurva hasil uji performa berdasarkan fps.....	74



DAFTAR TABEL

Tabel 3.1. Main <i>Game</i> Concepts.....	19
Tabel 3.2. Kebutuhan Teknologi.....	20
Tabel 3.3. Rancangan Pengontrol Permainan	21
Tabel 3.4. Rancangan Antarmuka.....	24
Tabel 3.5. <i>Gameplay</i> Consept Design di Atas Kertas.....	29
Tabel 3.6. Identifikasi Aktor	34
Tabel 3.7. Deskripsi Diagram Class.....	37
Tabel 3.8. Deskripsi Abstract Class	39
Tabel 3.9. Diagram Activity.....	42
Tabel 4.1. Prosedur Akselerasi Truk.....	48
Tabel 4.2. Prosedur Kendali Roda Truk.....	49
Tabel 4.3. Prosedur Pemasangan Chassis	50
Tabel 4.4. Prosedur Angkut Petikemas.....	51
Tabel 4.5. Prosedur Nilai Ukur Kerusakan Mesin.....	53
Tabel 4.6. Prosedur Nilai Ukur Jumlah Bahan Bakar.....	53
Tabel 4.7. Informasi Keterangan Fungsional Hud pada Layar Utama	59
Tabel 5.1. Prosedur Fungsi Pergerakan Akselerasi Truk.....	64
Tabel 5.2. Prosedur Kendali Roda Truk.....	65
Tabel 5.3. Prosedur Pemasangan Chasis.....	66
Tabel 5.4. Prosedur Angkut Petikemas.....	68
Tabel 5.5. Prosedur Nlai Ukur Kerusakan Mesin	69

Tabel 5.6. Prosedur Nilai Ukur Jumlah Bahan Bakar.....	70
Tabel 5.7. Prosedur Pemanggilan Truk.....	72
Tabel 5.8. Spesifikasi Perangkat Keras.....	74
Tabel 5.9. Analisis Pengujian Validasi.....	75
Tabel 5.10. Analisis Pengujian Pergerakan Akselerasi Truk.....	76
Tabel 5.11. Analisis Pengujian Kendali Roda Truk.....	77
Tabel 5.12. Analisis Pengujian Pemasangan Chasis.....	78
Tabel 5.13. Analisis Pengujian Angkut Petikemas.....	79
Tabel 5.14. Analisis Pengujian Nilai Ukur Kerusakan Mesin.....	80
Tabel 5.15. Analisis Pengujian Nilai Ukur Jumlah Bahan Bakar.....	81
Tabel 5.16. Analisis Pengujian Pembuatan Truk.....	82



DAFTAR LAMPIRAN

LAMPIRAN L-1



BAB I PENDAHULUAN

1.1. Latar Belakang

Peran pelabuhan sangat penting bagi perkembangan sosial dan ekonomi suatu daerah mengingat pelabuhan merupakan pusat segala kegiatan pelayanan pelayaran yang meliputi pelayanan terhadap kapal dan muatannya (penumpang, barang, dan hewan). Pelabuhan khusus seperti di Indonesia dibuat selain untuk transportasi masal antar pulau juga dibuat khusus untuk transportasi barang/ petikemas yang biasa disebut terminal petikemas. Peti kemas merupakan suatu kemasan yang dirancang secara khusus dengan ukuran tertentu, dapat dipakai berulang kali, dipergunakan untuk menyimpan sekaligus mengangkut muatan yang ada di dalamnya. Petikemas memiliki beberapa jenis dan ukuran, untuk jenisnya petikemas ada 3 yaitu petikemas biasa, petikemas over dimension (ukuran tidak umum), petikemas tanki (berbentuk silinder untuk cairan), dan petikemas *refrigerator* (pendingin). Sedangkan untuk petikemas biasa terdapat 3 ukuran yang sudah terstandarisasi yaitu ukuran 20'', 40'', dan 45''.

Terminal petikemas menjadi lokasi yang sangat rentan terhadap kelalaian karyawan. Kendaraan-kendaraan dan alat-alat berat sangat banyak digunakan di dalam, sehingga sangat beresiko terjadi kecelakaan yang fatal bila karyawan melakukan sedikit kesalahan prosedur. Maka dari itu, terminal petikemas memiliki prosedur-prosedur dalam bentuk instruksi kerja yang berbeda-beda bagi setiap karyawan yang bekerja di dalam area terminal petikemas, menyesuaikan penggunaan alat berat yang digunakan. Ada beberapa alat berat yang berhubungan langsung dengan proses bongkar-muat petikemas yaitu RTG, *container crane*, dan Truk. Dari ketiga alat berat tersebut hanya merupakan sebagian dari alat-alat berat yang digunakan di dalam terminal petikemas khususnya di Tanjung Perak, Surabaya.

Mengambil dari masalah yang pernah diutarakan oleh Kokoh Wahyu Utomo (Staff HRD PT. Gama Mitra Perkasa), peraturan yang dibuat digunakan untuk menunjang keselamatan kerja dan menjaga serta menertibkan agar meminimalisir kejadian yang tidak diharapkan [KOK-12]. Operator truk merupakan seorang yang berperan penting dalam menjalankan transportasi petikemas dari dermaga pelabuhan menuju terminal atau sebaliknya. Operator truk yang beroperasi di dalam terminal petikemas juga memiliki instruksi kerja yang berbeda dengan yang lainnya. Instruksi kerja tersebut merupakan peraturan dan tata cara beroperasi di dalam terminal petikemas. Peraturan dan tata cara tersebut yang menjadi suatu keharusan bagi seorang operator truk agar dapat sesuai dengan prosedur kerjanya.

Prosedur-prosedur pada instruksi kerja untuk operator truk memiliki beberapa poin yang harus dipatuhi dan dilaksanakan. Banyaknya poin-poin tersebut kemudian dicetak menjadi dokumen kerja yang wajib dibawa ketika operator truk bekerja. Hal tersebut tidak menutup kemungkinan dokumen yang dibawa hilang atau tertinggal. Untuk melaksanakan prosedur yang tepat sasaran tidak akan terlaksana bilamana karyawan tersebut apalagi yang baru saja bekerja sebagai operator salah satu alat berat sulit memahami poin-poin instruksi kerja. Oleh karena itu, dibutuhkan sebuah alternatif yang dapat membantu pemahaman diri dengan menggambarkan suatu aturan atau tata cara dalam sebuah media yakni melalui *game* simulasi. Simulasi adalah model dari suatu sistem yang nyata, melakukan pengolahan data terhadap model tersebut dan mengevaluasi hasil simulasi tersebut. Simulasi juga merupakan teknik untuk menyelesaikan persoalan melalui pengolahan data yang nyata untuk memperoleh *output* berupa solusi persoalan ataupun sebagai bahan masukan dalam rangka pengembangan dan perbaikan struktur dan operasi sistem yang nyata [SYA-13]. Pada penelitian ini dilakukan studi kasus pengenalan proses bongkar-muat petikemas di Pelabuhan Tanjung Perak, Surabaya menggunakan truk.

Berdasarkan penelitian De Porter, manusia dapat menyerap suatu materi sebanyak 70% dari apa yang dikerjakan, 50% dari apa yang didengar dan dilihat (audio

visual), sedangkan dari yang dilihatnya saja hanya 30%, dari yang didengarnya saja hanya 20%, dan dari yang dibaca hanya 10% [HID-08]. Dari penelitian tersebut, *game* simulasi menjadi solusi dimana itu merupakan sesuatu yang dilihat dan dikerjakan oleh pengguna secara langsung. Permainan yang menarik dapat memberikan kesan yang dapat diterima di dalam otak. Apalagi sebuah permainan yang kemudian digabungkan dengan sebuah simulasi hasil pemodelan dengan konsep 3D sehingga membuat kesan realitas kepada pengguna. Konten 3D kini menjadi dasar pembuatan simulasi yang lebih menarik dan interaktif untuk memberikan informasi kepada masyarakat seperti menginformasikan tentang pendidikan, sistem kerja, aplikasi, *game*, dan web. Meskipun sebenarnya informasi dalam bentuk atau konten 3D lebih kecil dan sedikit tulisan, namun lebih bersahabat dan mudah dipahami. 3D memiliki daya tarik langsung yang seakan-akan dapat mencerminkan dunia yang hidup [MAC-05].

Perkembangan *game* dewasa ini semakin pesat, hingga beberapa tahun berjalan, *game* telah terbagi menjadi beberapa bentuk jenis yang berbeda. Dari berbagai macam jenis *game* yang ada, *Vehicle Simulation* salah satunya. Jenis *game* ini menjadi dasar dari “Game Simulasi Truk Petikemas 3D”. Permainan ini memberikan simulasi tentang mengendarai sebuah truk petikemas dalam pelabuhan. Selain berfungsi sebagai hiburan, *game* simulasi juga berfungsi untuk menguji pemahaman mengendarai sebuah kendaraan secara langsung dengan media *game*. Seperti halnya beberapa ragam *game* yang muncul semisal simulasi SIM, berawal dari adanya peraturan yang diterapkan dan harus dilaksanakan *game* simulasi SIM ini digunakan untuk membantu memahami bagaimana peraturan itu dibuat bukan untuk dilanggar melainkan untuk dipatuhi dan dilaksanakan.

Berdasarkan uraian pada latar belakang di atas, maka pada penelitian ini dibangun sebuah aplikasi *game* untuk simulasi truk petikemas menggunakan *game engine Waveengine* dengan bahasa pemrograman C# serta *compiler* Visual Studio 2012 dengan harapan dapat memberikan pengalaman lebih untuk pengguna tentang keadaan di dalam terminal petikemas khususnya kegiatan operator truk.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang dipaparkan maka rumusan masalah yang perlu diperhatikan adalah:

1. Bagaimana memodelkan lingkungan terminal petikemas dalam sebuah “Game Simulasi Truk Petikemas 3D” ?
2. Bagaimana menerapkan kegiatan truk petikemas dalam terminal petikemas menjadi sebuah *gameplay* simulasi berkendara ?
3. Bagaimana menguji “Game Simulasi Truk Petikemas 3D” dengan menggunakan teknik *black box*, *white box*, dan performa berdasarkan FPS (*Frames per Second*) ?

1.3. Batasan Masalah

Agar penelitian ini lebih terfokuskan pada proses pengembangan *game*, maka dalam melakukan penelitian akan dibatasi pada hal-hal pokok sebagai berikut:

1. Informasi edukasi “Game Simulasi Truk Petikemas 3D” ini adalah memperkenalkan proses terjadinya kegiatan bongkar-muat petikemas yang dilakukan operator truk di dalam terminal petikemas dan latihan berkendara.
2. Aktivitas di dalam permainan akan lebih difokuskan pada kegiatan operator truk saja.
3. Instruksi kerja operator truk yang digunakan adalah instruksi kerja untuk aktivitas bongkar muat petikemas secara umum.
4. Pengujian pada penelitian ini dibatasi hingga pada proses produksi.
5. Pengujian *game* ini akan menggunakan teknik pengujian *black box*, *white box*, dan performa berdasarkan FPS (*Frames per Second*).
6. Dirancang pada platform PC dengan sistem operasi *windows*

1.4. Tujuan Penelitian

Tujuan penelitian ini adalah membangun “Game Simulasi Truk Petikemas 3D” dengan menerapkan kegiatan-kegiatan yang dilakukan operator truk di pelabuhan petikemas Tanjung Perak, Surabaya.

1.5. Manfaat

Manfaat dari “Game Simulasi Truk Petikemas 3D” adalah sebagai berikut :

1. Menambah variasi *game* dan memberikan hiburan bagi pengguna umum yang ingin mencoba mengendarai truk dan proses kerjanya di pelabuhan.
2. Dapat mengenalkan lingkungan, instrument-instrumen, dan instruksi kerja bagi operator truk khususnya yang baru dipekerjakan.
3. Dapat mengurangi resiko tentang keselamatan kerja karyawan khususnya operator truk.
4. Memberikan pengalaman lebih pada pengguna akan kegiatan operator truk petikemas di dalam terminal petikemas.

Hasil penelitian ini diharapkan dapat dijadikan sebagai bahan referensi dan tambahan dalam pengembangan *game* simulasi selanjutnya.

1.6. Sistematika Penulisan

Sistematika penulisan dalam skripsi ini sebagai berikut:

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

BAB II Landasan Teori

Menguraikan tentang landasan teori dan referensi yang mendasari rancang bangun

“Game Simulasi Truk Petikemas 3D”.

BAB III Metodologi dan Perancangan

Membahas metode yang digunakan dalam penulisan dan *design* perancangan *game*.

BAB IV Implementasi

Membahas tentang implementasi perangkat lunak.

BAB V Pengujian dan Analisis

Memuat proses pengujian dan hasil analisis terhadap *game*.

BAB VI Penutup

Memuat kesimpulan serta saran untuk pengembangan lebih lanjut.



BAB II

LANDASAN TEORI

2.1. Game Simulasi

Game simulasi merupakan interaksi antara alat penerapan yang diberikan dalam memainkan peran pada lingkungan tertentu [SHI-05]. *Game* simulasi membuat pengalaman pengguna dalam situasi nyata dari sebuah tempat latihan yang aman [PED-09]. Dalam *Level Up! The Guide to Great Video Game*, Scott Rogers menyebutkan 2 jenis *game* simulasi yaitu [ROG-10].

1. *Life Simulation* : jenis simulasi ini membuat pemain dapat mengitari bangunan dan memelihara hubungan dengan bentuk kehidupan buatan. Contoh permainan dengan jenis ini adalah *The Sims* dan *Princess Maker*. Jenis ini juga memiliki jenis turunannya yaitu *Pet Simulation*.
2. *Vehicle Simulation* : jenis simulasi yang membuat pemain dapat melakukan kegiatan pilot/menyetir sebuah kendaraan serta fokus kepada penciptaan pengalaman senyata mungkin. Turunan dari jenis permainan ini yaitu *Driving* dan *Flying*.

Jenis *game* simulasi tersebut merupakan jenis yang saat ini sering ditemui, namun tidak menutup kemungkinan perkembangan permainan yang sangat pesat memunculkan jenis-jenis ataupun turunannya yang baru.

2.2. Simulasi Kendaraan

Simulasi kendaraan adalah permainan yang mencoba mensimulasikan secara akurat melalui permainan tersebut. Permainan simulasi kendaraan menciptakan rasa simulasi langsung bagi penggunanya untuk memiliki kemampuan dalam melaksanakan kehidupan nyata seperti kecepatan, belokan, kerusakan, penggunaan bahan bakar, *grip*

dan juga pengaturan suspensi [ANO-09]. Sedangkan secara garis besar permainan simulasi kendaraan sebagai jenis *video game* yang mencoba untuk memberikan interpretasi realistis operasi dari berbagai jenis kendaraan, termasuk mobil, pesawat, perahu, pesawat ruang angkasa, kendaraan militer, dan berbagai kendaraan lainnya. Tantangan pada *game* ini adalah untuk menguasai kemudi kendaraan dari perspektif pilot atau supir. Tantangan lain yang diberikan pada *game* ini misalnya balapan, dan manajemen bahan bakar. Simulasi kendaraan juga dapat disebut sebagai *game* yang mensimulasikan pilot/menyetir sebuah kendaraan serta fokus kepada penciptaan pengalaman senyata mungkin [ROG-10].

Konten 3D kini menjadi dasar pembuatan simulasi yang lebih menarik dan interaktif untuk memberikan informasi kepada masyarakat seperti menginformasikan tentang pendidikan, sistem kerja, aplikasi, *game*, dan web [MAC-12]. Dengan semakin pesatnya dunia informasi dan teknologi kini konten 3D menjadi sesuatu yang dibutuhkan untuk menjadikan suatu informasi lebih interaktif dan menarik, dibandingkan dengan konten 2D atau grafik *pixel*.

2.3. Pengembangan Karakter Melalui Simulasi

Sridadi dalam Syahfitri menyatakan bahwa simulasi adalah merancang model dari suatu sistem yang nyata, mengadakan percobaan-percobaan terhadap model tersebut dan mengevaluasi hasil percobaan tersebut [SYA-13]. Simulasi merupakan teknik atau cara penyelesaian persoalan melalui pengolahan data operasi sistem imitasi untuk memperoleh data *output* penyelidikan atau percobaan penelitian sebagai bahan solusi persoalan ataupun sebagai bahan masukan dalam rangka pengembangan dan perbaikan struktur dan operasi sistem yang nyata.

Manfaat simulasi bagi kehidupan nyata adalah untuk; (1) Menjelaskan kelakuan sistem, (2) Menirukan bekerjanya suatu sistem melalui suatu model, (3) Memecahkan suatu persoalan matematik dengan analisis numerik, (4) Mempelajari dinamika suatu

sistem, (5) Memberikan suatu deskripsi perilaku sistem dalam perkembangan sejalan dengan bertambahnya waktu, (6) Membangun teori atau hipotesa yang mempertanggungjawabkan kelakuan dari sistem yang diamati, (7) Meramalkan kelakuan sistem yang akan datang yaitu pengaruh yang dihasilkan oleh perubahan-perubahan sistem atau perubahan operasinya.

Piaget, Vigotsky, dan Zuhdi menyatakan permainan memiliki peranan penting dalam pengembangan karakter anak [RES-NN]. Melalui sebuah permainan yang memunculkan permasalahan yang kemudian diselesaikan dengan cara menyenangkan memberikan dampak terhadap pengembangan karakter pengguna. Riberman (1977), Garvey (1990), Neumann (1971), dan Schwartzman (1978) menyatakan pendapatnya bahwa simulasi kreatif; (1) bukan merupakan suatu pekerjaan yang nyata (manipulatif), (2) memiliki tiga elemen (criteria, proses, tujuan), (3) kegiatan fisik yang spontan dan sukarela, (4) tidak produktif, (5) hal yang menyenangkan/menggembirakan, (6) motivasinya intrinsic, (7) tidak memiliki tujuan ekstrinsik, (8) melibatkan aktif pelakunya, dan (9) merupakan spontanitas fisik, sosial, dan kognitif [RES-NN].

2.4. Pemodelan Simulasi Dalam Sebuah Game

Pemodelan sistem kendaraan sangat penting dilakukan dalam sebuah *game* simulasi kendaraan baik itu pesawat, mobil, kapal, dan lain-lain. Pemodelan tersebut diperlukan agar dalam sebuah *game* terlihat realistis. Menurut Gao, Mi, dan Emadi pemodelan tersebut dilakukan dengan merincikan beberapa pemodelan yaitu pemodelan umum yaitu getaran, penanganan, dan kebisingan, pemodelan performa kendaraan (misalnya, percepatan, kemampuan tanjakan, dan kecepatan laju maksimum), pemodelan untuk prediksi, evaluasi, dan optimalisasi bahan bakar, ekonomi, pemodelan untuk keamanan, stabilitas, dan kecelakaan kelayakan pemodelan kontrol kendaraan, pemodelan untuk structural integritas; pemodelan untuk memfasilitasi pengujian komponen dan validasi, pemodelan untuk konsep desain awal

/desain eksplorasi, pemodelan untuk biaya dan kemasan, dan pemodelan untuk prediksi emisi [GAO-09].

Dengan beberapa pemodelan yang ditentukan untuk memberikan sentuhan nyata dalam sebuah *game* kendaraan, kini dapat dilakukan dengan mudah dengan menggunakan sebuah *framework* atau *engine* fisika sehingga dapat membentuk objek representasi kendaraan yang tampak nyata.

2.5. Game Engine (*Waveengine*)

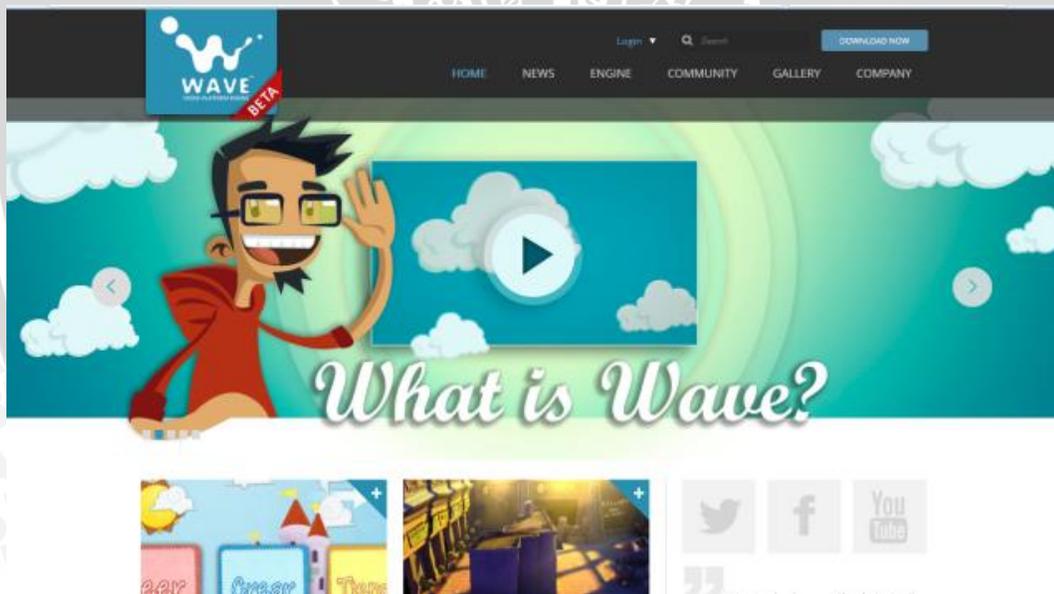
Waveengine adalah sebuah *game engine* yang lahir dua tahun lalu dan membawa misi untuk membantu komunitas pengembang *game mobile* [WAV-13]. Seperti dalam Gambar 2.1 menampilkan halaman awal situs *waveengine.net* yang merupakan web utamanya. *Game engine* merupakan salah satu cara mempermudah para pengembang *game* dalam membuat *game* dan *Waveengine* salah satunya. *Waveengine* berjalan pada beberapa OS atau *game console* di antaranya yaitu Windows, iOS, OUYA, Android. *Waveengine* juga merupakan *game engine* yang free source sehingga dapat diunduh dengan gratis namun juga dapat dikembangkan dan dilihat beberapa komponen didalamnya. *Waveengine* terintegrasi dalam *compiler* visual studio 2010, dan 2012, serta *monodevelop* sehingga *software* ini perlu digunakan untuk menggunakan *Waveengine* sebagai *framework*-nya.

Waveengine memiliki beberapa kelebihan diantaranya yaitu :

1. Merupakan *game engine* yang dapat mengkonversi pada beberapa platform mobile dan desktop.
2. Berbasis pada sistem komponen sehingga dapat digunakan dengan mudah dan digunakan kembali
3. Penulisan kode menggunakan .NET Platform
4. Ringan digunakan di semua komputer

Waveengine juga memiliki beberapa alat untuk membantu dalam pembuatan *game* diantaranya yaitu.

1. *Wave Asset Exporter*, alat ini digunakan untuk mengkonversikan semua aset yang dibutuhkan baik itu aset 2D ataupun 3D serta aset *sound*.
2. *Project Converter*, alat ini digunakan untuk mengkonversi *game* ke semua platform yang tersedia.
3. *Advertising configurator*, alat ini digunakan untuk mengatur iklan yang akan dimunculkan pada *game* yang dibuat. Bagi pengembang *game* biasanya digunakan untuk memasang iklan pada *game store*.
4. *Sample Browser*, ini merupakan beberapa file dan contoh kode dari beberapa *game* sederhana yang telah dibuat dengan menggunakan *Waveengine* untuk mempermudah pemula dalam menggunakan *Waveengine*.



Gambar 2.1. Screenshoot situs *Waveengine*.

Sumber : [WAV-13]

2.6. Blender

Aplikasi blender merupakan aplikasi Pengolah grafis 3D *open source* yang dikembangkan oleh Blender Foundation, sebuah organisasi independen dari Belanda. Aplikasi blender ini sangat berguna bagi penggemar grafis 3D, animasi bergerak, dan lain-lain. Aplikasi ini cocok digunakan bagi siapapun yang ingin berkarya dengan biaya yang murah. Meskipun Blender merupakan Aplikasi yang gratis, aplikasi ini juga menawarkan fitur yang sangat canggih dan tidak bisa dipandang remeh [FLA-10].

Ton Roosendaal, cofounder dari studio animasi NeoGeo, bertanggung jawab terhadap perusahaan rumah *software*-nya. Namun perusahaan tersebut tertimpa masalah financial, dan di tahun 2002 dengan investor mulai menarik aset, Ton mengambil langkah radikal dengan menawarkan *software* Blender sebagai *software open source*, dengan ketentuan bahwa penggemar Blender dapat meningkatkan €100.000 untuk melepaskan utang. Dalam beberapa minggu Blender telah memperoleh pengikut setia, dan tawaran berhasil dalam beberapa minggu. Sesuai dengan janji Ton, Blender dikeluarkan sebagai *open source software* dibawah lisensi GNU *General Public Lisence* (GPL), dan Ton telah mendedikasikan dirinya untuk akan mengembangkan Blender. Hingga kini Blender berkembang karena kuatnya penggemar-penggemar dan pengembang-pengembang yang tanpa lelah bekerja untuk menyempurnakan aplikasi dan penggunaannya [FLA-10]. Sperti yang ditampilkan dalam Gambar 2.2 merupakan tampilan Blender versi 2.68 a.



Gambar 2.2. Blender 2.68 a.

Sumber : [BLE-13]

Blender memiliki peran yang sangat penting dalam pengembangan animasi, modeling, dan *game* khususnya pada konten 3D. Dengan beberapa fitur yang tidak kalah bagus dengan *software* 3D berbayar lainnya, Blender mampu menunjukkan kemampuannya dan menghasilkan beberapa karya hebat yang memotivasi para *blender artist* (sebutan untuk pengguna Blender) dalam menghasilkan karya-karya lainnya. Blender memiliki fitur yang hampir menyerupai dengan *software* 3D yang berbayar dan dapat digunakan secara maksimal walaupun memiliki ukuran yang kecil dibandingkan *software* 3D lainnya.

Untuk membuat transisi 3D menggunakan blender, ada beberapa hal yang perlu diingat yang akan membuat lebih terasa mudah :

- Blender adalah aplikasi gratis. Seorang yang ingin membuat kreasi dengan aplikasi ini tidak perlu memikirkan lagi masalah pembayaran untuk meng-upgrade atau isu mengenai lisensi.

- Blender adalah aplikasi yang mempunyai komunitas. Jika dalam penggunaannya terdapat kesulitan kita dapat langsung mengunjungi halaman forum di <http://blenderartist.org>

Banyaknya sumber-sumber yang ada pada tutorial online, termasuk <http://blendercookie.com>, <http://blenderguru.com>, dan <http://blendernewbies.com> [JON-12].

2.7. Pengujian Perangkat Lunak

Pengujian perangkat lunak memerlukan perancangan kasus uji (test case) agar dapat menemukan kesalahan dalam waktu singkat dan usaha minimum. Berbagai macam metode perancangan kasus uji telah berevolusi. Metode ini menyediakan pendekatan sistematis untuk pengujian oleh developer. Terlebih lagi metode ini menyediakan mekanisme yang dapat membantu memastikan kelengkapan dari pengujian dan menyediakan kemungkinan tertinggi untuk menemukan kesalahan-kesalahan dalam perangkat lunak. Teknik atau metode perancangan kasus uji yang digunakan adalah black-box testing dan white-box testing [PRE-10:443].

2.7.1. White-Box Testing

White box testing adalah teknik pengujian yang menguji berdasarkan jalur internal, struktur dan implementasi dari perangkat lunak yang sedang diuji. White box testing adalah teknik pengujian yang menggunakan struktur kontrol dari prosedur yang terdapat dalam perancangan untuk membuat kasus uji. Ada dua jenis pengujian yang termasuk white box testing yaitu basis path testing dan control structure testing. Basis path testing adalah teknik dalam metode white box testing untuk menyediakan desain test case yang dapat digunakan untuk menghitung kompleksitas desain prosedural untuk menghitung banyaknya jalur eksekusi. Test case tersebut digunakan untuk menjamin apakah tiap statement telah dijalankan minimal satu kali tiap pengujian. Dalam konteks basis path, nilai yang dihitung di dalam cyclomatic complexity

menjelaskan banyaknya jumlah jalur independent yang harus disediakan dalam pengujian basis path [PRE-10:445].

2.7.2. Black-Box Testing

Black box testing adalah teknik pengujian yang menguji hanya berdasarkan kebutuhan dan spesifikasi. *Black box testing* juga disebut sebagai behavioral testing dan berfokus pada kebutuhan fungsi dari perangkat lunak [PRE-10:460]. Proses umum yang terjadi pada black box testing yaitu:

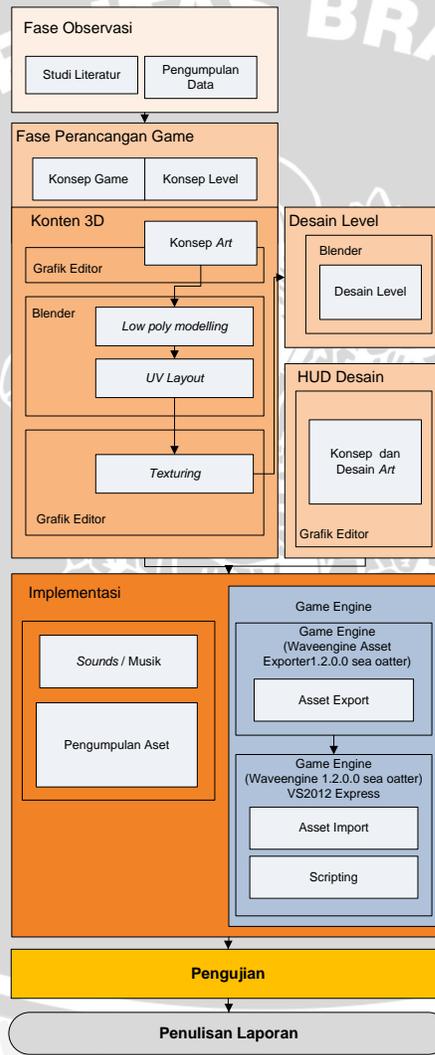
- a. Kebutuhan atau spesifikasi dianalisa terlebih dahulu.
- b. Penentuan input valid terpilih berdasarkan spesifikasi untuk menentukan perangkat lunak berjalan dengan benar. Input yang tidak valid juga harus dipilih untuk memverifikasi bahwa perangkat lunak dapat mendeteksinya dan menanganinya dengan baik.
- c. Penentuan output yang diharapkan sesuai dengan input yang telah dipilih.
- d. Pengujian dibuat dengan input yang telah dipilih.
- e. Pengujian dijalankan.
- f. Output yang sebenarnya dibandingkan dengan output yang diharapkan
- g. Penentuan dibuat menyangkut perangkat lunak berfungsi sesuai dengan spesifikasi yang telah ditentukan.

2.8. Pengujian Performa

Pengujian ini dilakukan dengan melihat *frame rate (Frames Per Second)*, merupakan jumlah perubahan frame yang ditampilkan dalam setiap detik. *Gameplay* yang berjalan secara halus (lancar) di bangun berdasarkan *frame rate* yang halus dan menyentuh nilai hingga 60 *frames per second*. Nilai tersebut ditemukan berdasarkan pada pengembangan *game shadow blade* yang berjalan pada iPhone dan iPad [FAS-13]. Dari referensi nilai FPS tersebut dapat kemudian nantinya menjadi acuan nilai FPS *game* dikatakan berjalan dengan baik.

BAB III METODOLOGI DAN PERANCANGAN

Metodologi Penelitian yang dilakukan penulis dalam skripsi ini adalah sebagai berikut digambarkan dengan diagram alir yang terdiri dari 3 fase yaitu fase observasi, fase perancangan *game*, dan fase implementasi dalam Gambar 3.1.



Gambar 3.1. Diagram alir produksi *game* secara umum.

Sumber : Metodologi dan perancangan.

Gambar 3.1 ketiga fase berjalan secara berurutan, mulai dari observasi, perancangan, implementasi, pengujian, hingga penulisan laporan. Mulainya aplikasi dikerjakan adalah ketika masuk dalam fase perancangan *game* semua konsep dan desain terfokuskan di fase tersebut. Di dalam fase implementasi konsep dan desain yang telah dibuat diolah melalui proses pengkodean. Untuk penjelasan yang lebih detail akan diulas pada sub-bab selanjutnya.

3.1. Fase Observasi

3.1.1. Studi Literatur

Sebelum masuk ke dalam fase perancangan, diperlukan fase observasi berupa studi literatur dan pengumpulan data yang dikerjakan berdasarkan lingkungan yang akan dijadikan sebuah *game*. Dalam kasus ini penulis menggunakan lingkungan terminal petikemas Surabaya sebagai acuan untuk masuk dalam fase perancangan.

3.1.2. Pengumpulan Data

Dalam melakukan pengumpulan data penelitian, penulis menggunakan studi kepustakaan yaitu pengumpulan data sekunder. Data sekunder dapat diperoleh dari berbagai buku, jurnal, dan artikel-artikel yang memuat mengenai data dan artikel-artikel terkait dengan *game* simulasi kendaraan. Data sekunder dimaknai sebagai data yang dikumpulkan dari tangan kedua atau dari sumber-sumber lain yang telah tersedia sebelum penelitian dilakukan [SIL-06].

Gambar 3.2 menunjukkan hasil observasi pada terminal petikemas di sisi dermaga ketika truk-truk mengangkut petikemas dari kapal dan kemudian diletakkan di area terminal. Hasil observasi ini akan digunakan untuk proses pemodelan pada fase perancangan.



Gambar 3.2. Lingkungan terminal petikemas Surabaya

Sumber : Metodologi dan perancangan.

3.2. Fase Perancangan

Tujuan utama dari permainan ini adalah agar pemain dapat mengendarai truk dengan benar dan sesuai peraturan serta mengenali semua kegiatan yang ada di dalam terminal petikemas. Kegiatan ini termasuk uji keterampilan mengendarai truk hingga melakukan kegiatan muat dan bongkar petikemas di dalam terminal. Setelah mengetahui tujuan dari pembuatan *game* ini maka pada fase perancangan di dalam pembuatan *game* dibagi menjadi dua perancangan yaitu *Game Concept Design*, *Concept Level Design*, dan *Technical Design*.

3.2.1. Game Concept Design

Dalam perancangan *Game concept design* pada “Game Simulasi Truk Petikemas 3D” terdiri dari beberapa sub-bagian yaitu (1)*Main Game Concept*, (2)Kebutuhan Teknologi, (3)Kontrol Permainan, (4)Dunia Dalam Permainan, (5)Sudut Kamera, (6)Alur Permainan, (7)Antarmuka, (8)*Art Concept*, (9)*Low Poly Modelling*, (10)*UV Layout*, (11)Pemberian Tekstur, (12)Desain *Heads Up Display* (HUD).

3.2.1.1. Main Game Concept

Konsep utama dari “Game Simulasi Truk Petikemas 3D” ditunjukkan pada Tabel 3.1.

Tabel 3.1. *Main Game Concepts*

No.	Elemen	Keterangan
1.	Judul <i>game</i>	Game Simulasi Truk Petikemas 3D
2.	<i>Game platform</i>	Windows
3.	<i>Target device</i>	Desktop/ PC
4.	Target usia	Semua Umur
5.	<i>Genre game</i>	<i>Vehicle Simulation</i>
6.	<i>USP</i> (<i>Unique Selling Point</i>)	<ol style="list-style-type: none"> Memudahkan pengguna dalam mengenali lingkungan area terminal petikemas tanpa memasukinya karena tidak semua orang bisa masuk ke dalamnya. Media alternatif pengenalan kegiatan truk operator petikemas. Konsep 3D yang membuat lingkungan tampak nyata.

	4. Terdapat hingga 11 level yang berbeda untuk melatih keterampilan berkendara.
--	---

Sumber: Metodologi dan perancangan

3.2.1.2. Kebutuhan Teknologi

Berikut kebutuhan teknologi untuk membangun permainan ditunjukkan pada Tabel 3.2.

Tabel 3.2. Kebutuhan Teknologi

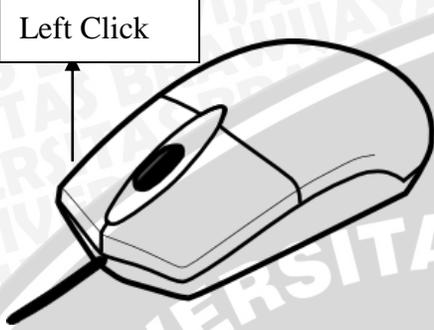
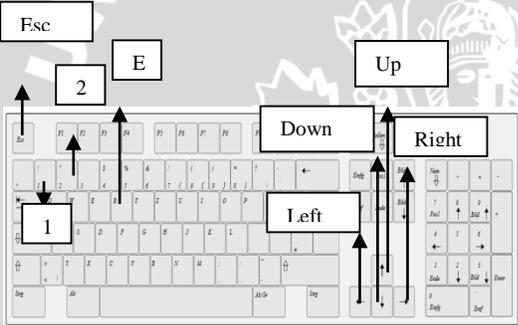
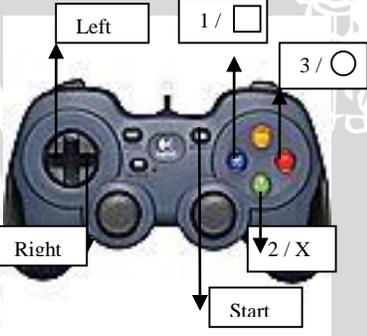
No	Kebutuhan	Kegunaan
1.	Aplikasi pengolah grafis	Sebagai media gambar digital.
2.	Blender	Sebagai media pemodelan 3D
3.	Aplikasi Visual Studio 2012 Express	Untuk membangun aplikasi.
4.	Waveengine v1.2.0 sea otter	Sebagai <i>game engine</i> yang digunakan.
5.	Komputer Desktop	Sebagai alat uji perangkat lunak.

Sumber : Metodologi dan perancangan

3.2.1.3. Kontrol Permainan

Dalam “Game Simulasi Truk Petikemas 3D” terdapat 3 alat kontrol yang akan digunakan yaitu *Mouse*, *Keyboard*, dan *Joystick*. Berikut perancangan penggunaan pengontrol dalam permainan ini yang dijelaskan dalam Tabel 3.3.

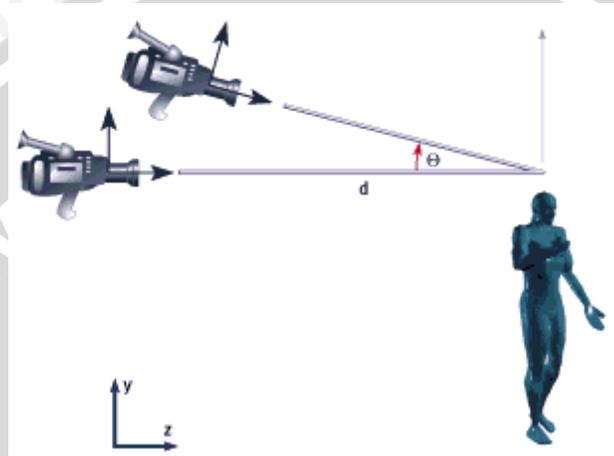
Tabel 3.3. Rancangan pengontrol permainan

No.	Jenis Pengontrol	Keterangan
1.	<p><i>Mouse</i></p> 	<p>Penggunaan <i>mouse</i> pada permainan hanya digunakan pada saat penekanan HUD <i>game</i> yang butuh penekanan. Pada permainan hanya left click yang digunakan.</p>
2.	<p><i>Keyboard</i></p> 	<p>Penggunaan <i>keyboard</i> pada permainan digunakan untuk mengendalikan truk, kamera, dan mengaitkan <i>chasis</i>. Pada permainan tombol-tombol disamping yang digunakan.</p>
3.	<p><i>Joystick</i></p> 	<p>Penggunaan <i>joystick</i> pada permainan secara keseluruhan sama dengan penggunaan <i>keyboard</i> yaitu digunakan untuk mengendalikan truk, kamera, dan mengaitkan <i>chasis</i>. Pada permainan tombol-tombol disamping yang digunakan.</p>

Sumber : Metodologi dan perancangan

3.2.1.4. Sudut Kamera

Permainan ini menggunakan tipe *third person camera*. Kamera orang ketiga ini mengikuti objek utama permainan dalam hal ini adalah truk. Dalam perancangannya kamera orang ketiga ini digunakan untuk memaksimalkan *gameplay* yang akan dibuat. Gambar 3.3 menunjukkan ilustrasi penggunaan kamera orang ketiga.



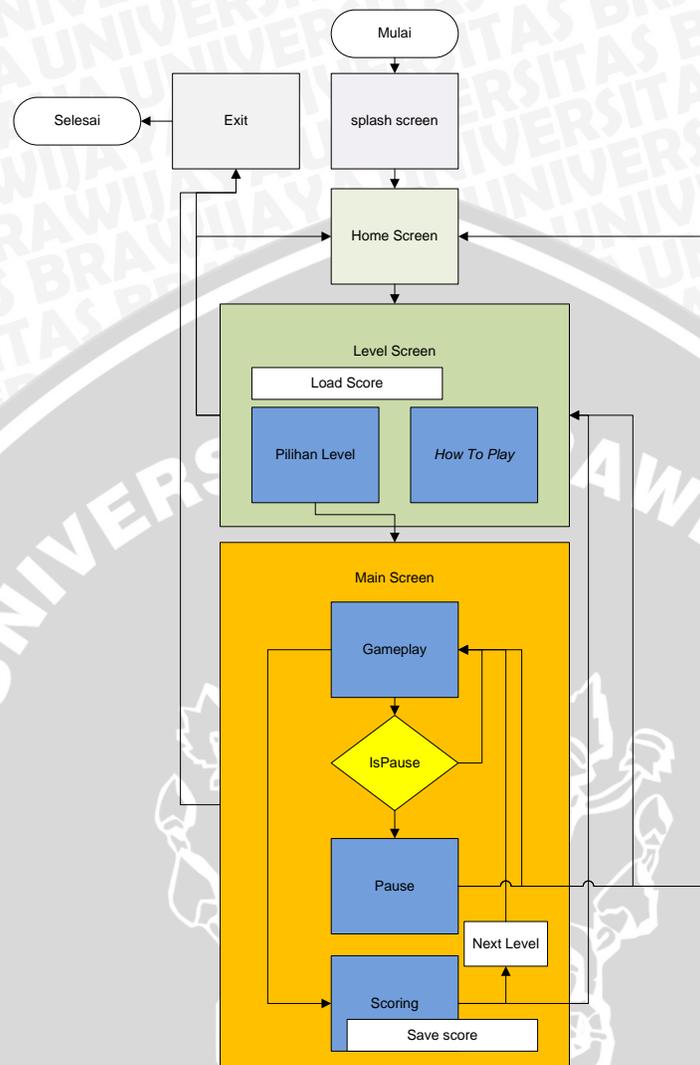
Gambar 3.3. Ilustrasi implementasi kamera orang ketiga.

Sumber : [BOB-98]

Gambar 3.3 menunjukkan kamera yang menginisialisasikan objek di depannya sebagai target sehingga kamera tersebut akan mengikuti objek target kemanapun objek tersebut bergerak baik itu maju atau mundur. Bahkan dalam Gambar 3.3 ditampilkan perubahan sudut sorot kamera pada objek target akibat perubahan rotasi pada objek target juga akan selalu diikuti.

3.2.1.5. Alur Permainan

Alur layar permainan yang akan dilihat oleh pemain akan dibagi menjadi 3 bagian, yang (1)*Home Scene*, (2)*Level Scene*, (3)*Main Scene*. Bagian-bagian layar yang akan ditampilkan tersebut dapat dilihat berdasarkan alur diagram dalam Gambar 3.4 berikut ini.



Gambar 3.4. Diagram alur permainan.

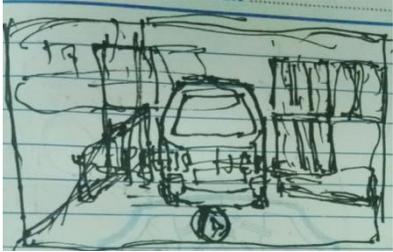
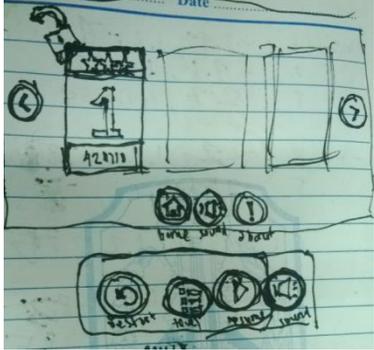
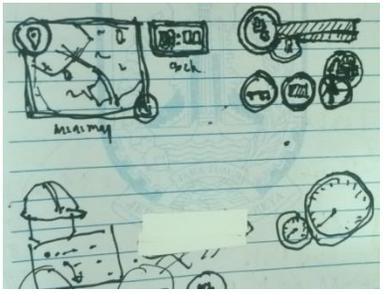
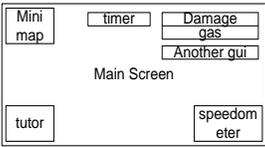
Sumber : Metodologi dan perancangan.

Gambar 3.4 menjelaskan alur permainan dalam sebuah diagram alur yang saling berhubungan. Terdapat beberapa tampilan yang akan dilihat oleh pengguna mulai dari *splash screen*, *home scene*, *level scene*, sampai *main scene*. Ketika *splash screen* permainan akan menampilkan logo *game engine* yaitu *Waveengine*, kemudian pada *home scene* dan selanjutnya akan dijelaskan lebih detail pada sub-bab selanjutnya.

3.2.1.6. Antarmuka

Desain tampilan antarmuka setiap *scene* ditunjukkan pada Tabel 3.4.

Tabel 3.4. Rancangan antarmuka.

Nama Layar	Desain Sketsa	Desain Antarmuka	Keterangan
Home Scene			<ol style="list-style-type: none"> 1. Home scene (judul permainan). 2. "Play Button" tombol akses menuju <i>level scene</i>
Level Scene			<ol style="list-style-type: none"> 1. Level list menampilkan seluruh <i>level game</i> 2. Menu terdapat tombol kembali ke home scene, cara bermain, dan atur suara.
Main Scene			<ol style="list-style-type: none"> 1. Pada main scene terdapat beberapa HUD yang penting untuk menunjang permainan

Sumber : Metodologi dan perancangan

3.2.1.7. Art Concept

Artist Concept bekerja khusus pada pembuatan aset utama *game* (lingkungan dan objek utama). Pembuatan aset-aset yang dibutuhkan selain dari segi pemodelan juga dibutuhkan penggunaan *UV layout* dan pembuatan *texture* untuk menampilkan objek lebih nyata dan berwarna. Gambar 3.5 menampilkan tampilan truk dalam bentuk ortografis sehingga memudahkan proses pemodelan.



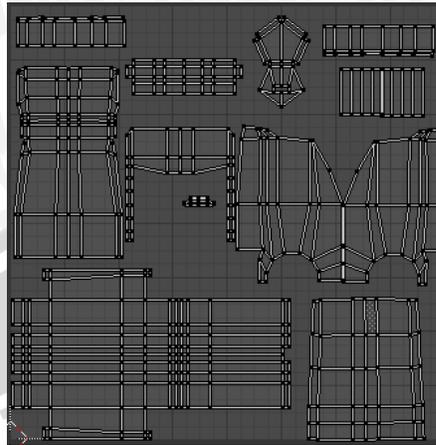
Gambar 3.5. Konsep truk.

Sumber : <http://www.the-blueprints.com>

3.2.1.8. Low Polygon Modeling

Langkah awal dalam membuat konsep konten 3D adalah membuat objek yang *low poly*. Dalam pembuatan *game*, objek *low poly* sangat dianjurkan karena akan mengurangi pembacaan poligon dalam *game engine*. Tidak pernah sekalipun kebanyakan *modeling artist* menggunakan *high poly* untuk membuat aset 3D, meskipun penggunaan *low poly* akan terlihat lebih kaku namun hal tersebut dapat diatasi dengan teknik *smooth* dan penggunaan beberapa jenis *texture* tertentu.

3.2.1.9. UV Layout

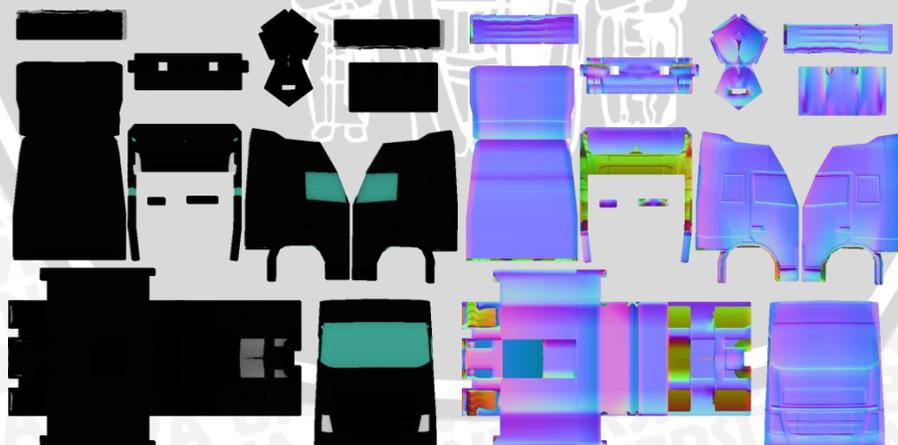


Gambar 3.6. UV layout truk

Sumber : Metodologi dan perancangan.

Sebelum proses *texturing* dapat dilakukan, *UV layout* harus sudah selesai dipetakan. Seperti dalam Gambar 3.6 *UV layout* untuk truk dipetakan berdasarkan vertex yang terbentuk. Sebuah *UV* dalam Blender pada pengaturan normal memiliki satu *UV layout*. Sebuah objek dengan banyak jenis tekstur dapat memiliki banyak *UV Layout*.

3.2.1.10. Pemberian Tekstur



Gambar 3.7. Pemberian tekstur truk (diffuse dan normal map)

Sumber : Metodologi dan perancangan.

Pemberian tekstur dimulai ketika *UV Layout* sudah dipetakan dalam Blender. Dalam “Game Simulasi Truk Petikemas 3D” proses ini dilakukan berdasarkan *Art Concept* yang sudah ditentukan, seperti halnya truk dengan warna yang sesuai dengan tekstur konsep dan beberapa aset lainnya. Gambar 3.7 menunjukkan pemberian tekstur truk dengan 2 kombinasi metode penteksturan yaitu *diffuse* dan *normal map*.

3.2.1.11. Desain HUD

Desain HUD merupakan informasi penting untuk pemain yang ditampilkan dalam layar utama, termasuk di dalamnya Health/Status/Live/Money dan lain sebagainya. Seperti pada Tabel 3.4 pada layar utama (*main scene*) menampilkan rancangan HUD permainan dan peletakannya.

3.2.2. Concept Level Design

Dalam perancangan *level design* akan menjelaskan bagaimana semua aset-aset yang dibutuhkan di dalam *game* dikombinasikan dari bentuk *virtual environment* menjadi *game* yang dapat dimainkan. Dalam proses ini dibagi menjadi *Game Goals Level, Story, Game Play Description, Penempatan Aset, Level Design* “Game Simulasi Truk Petikemas 3D”.

3.2.2.1. Game Goals Level

Game Goals Levels merupakan salah satu komponen dalam *game* yang sangat penting, bertujuan untuk membuat permainan lebih menarik dan menantang. *Game Goals Levels* menciptakan perbandingan simetris antara level permainan dengan tingkat kesusahan, khususnya dalam *game* simulasi truk petikemas 3D.

1. Level 1 : Pengenalan area dan penggunaan bantuan panah sebagai penunjuk arah tempat tujuan.
2. Level 2 : Pengenalan penggunaan *mini map* pada HUD *game* untuk melihat tempat tujuan..

3. Level 3 : Pengenalan mengendarai truk dengan menguji kemampuan haluan kiri dan kanan.
4. Level 4 : Pengenalan penggunaan kotak dialog perintah pada HUD *game* yang dapat disembunyikan atau ditampilkan.
5. Level 5 : Pengenalan mengendarai truk dengan menguji kemampuan bergerak mundur.
6. Level 6 : Pengenalan menggunakan chasis.
7. Level 7 : Pengenalan memuat petikemas pada area terminal.
8. Level 8 : Pengenalan memuat petikemas pada area dermaga.
9. Level 9 : Melakukan kegiatan bongkar-muat sebanyak 3 kali dari terminal menuju dermaga (Eksport).
10. Level 10 : Melakukan kegiatan bongkar-muat sebanyak 3 kali dari dermaga menuju terminal (Import).
11. Level 11 : Melakukan kegiatan bongkar-muat sebanyak-banyaknya dan mendapat nilai tertinggi.

3.2.2.2. Story

Dalam permainan “Game Simulasi Truk Petikemas 3D” dari beberapa konsep level dibagi menjadi 3 kategori. 8 level awal pemula, 2 level tengah tingkat lanjut, dan 1 level akhir bermain bebas.

1. Story level awal pemula, 8 level awal ini merupakan level dimana pemain diharuskan mengerti cara mengendarai, menggunakan *chasis*, dan melakukan kegiatan muat petikemas. Semua level awal tersebut harus dapat diselesaikan secara berurutan dengan waktu yang sudah ditentukan di awal, untuk dapat memudahkan pemain menyelesaikan story level tingkat selanjutnya.
2. Story level tengah tingkat lanjut, 2 level tengah ini merupakan level dimana pemain diharuskan sudah dapat melakukan kegiatan bongkar-muat petikemas baik dari terminal menuju dermaga, ataupun dari dermaga menuju terminal. 2 level tersebut harus dapat diselesaikan secara berurutan dengan waktu yang sudah

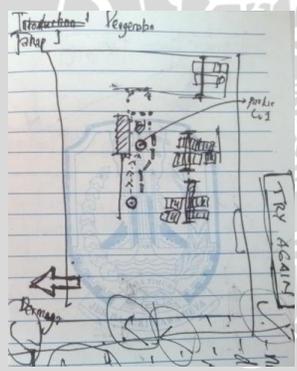
ditentukan di awal, untuk dapat memudahkan pemain menyelesaikan story level tingkat selanjutnya.

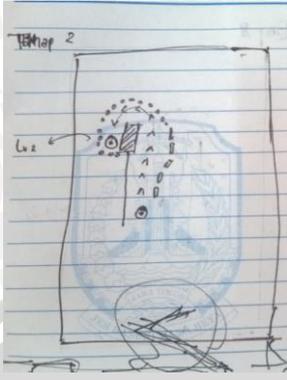
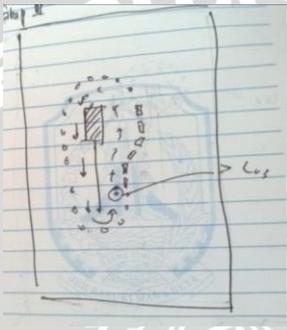
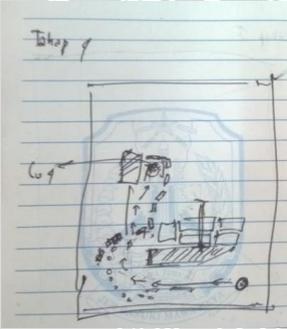
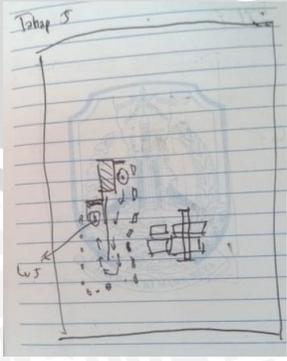
3. Story level akhir bermain bebas, dalam level ini pemain bukan berarti dapat mengendarai truk dengan seenaknya, namun dalam level ini pemain diharuskan melakukan kegiatan bongkar-muat petikemas sebanyak-banyaknya dengan waktu yang telah ditentukan untuk mendapatkan nilai skor tertinggi, dan setiap kali selesai melakukan peletakan petikemas, pemain akan mendapatkan nilai skor dan waktu yang terus bertambah.

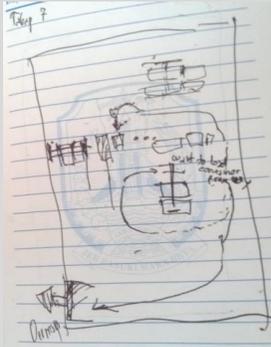
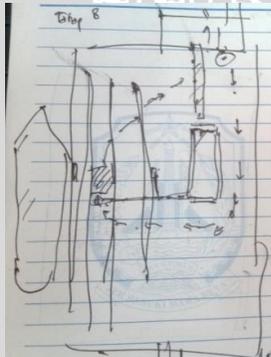
3.2.2.3. Gameplay Description

Dari 11 level yang ada pada permainan “Game Simulasi Truk Petikemas 3D” terdapat 8 *gameplay* khusus dan beberapa memiliki kesamaan namun tujuan dan lokasi yang berbeda. Berikut konsep *gameplay* yang telah di rancang di atas kertas dijelaskan pada Tabel 3.5 di bawah ini.

Tabel 3.5. *Gameplay* Concept Design di Atas Kertas

No. / Level	<i>Gameplay</i> Concept Design	Keterangan
1.		<p><i>Gameplay</i> concept di samping merupakan terapan pada level 1</p> <p>Tanda panah menunjukkan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area terminal petikemas.</p>

<p>2.</p>		<p><i>Gameplay concept</i> di samping merupakan terapan pada level 2</p> <p>Tanda panah menunjukan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area terminal petikemas.</p>
<p>3.</p>		<p><i>Gameplay concept</i> di samping merupakan terapan pada level 3.</p> <p>Tanda panah menunjukan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area terminal petikemas.</p>
<p>4.</p>		<p><i>Gameplay concept</i> di samping merupakan terapan pada level 4.</p> <p>Tanda panah menunjukan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area terminal petikemas.</p>
<p>5.</p>		<p><i>Gameplay concept</i> di samping merupakan terapan pada level 5.</p> <p>Tanda panah menunjukan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area terminal petikemas.</p>

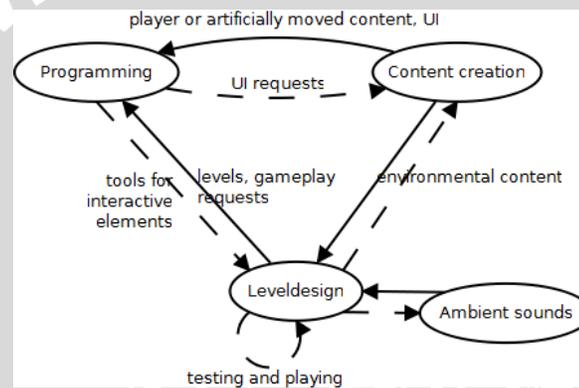
<p>6.</p>		<p><i>Gameplay concept</i> di samping merupakan terapan pada level 6.</p> <p>Tanda panah menunjukkan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area terminal petikemas.</p>
<p>7.</p>		<p><i>Gameplay concept</i> di samping merupakan terapan pada level 7.</p> <p>Tanda panah menunjukkan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area terminal petikemas.</p>
<p>8.</p>		<p><i>Gameplay concept</i> di samping merupakan terapan pada level 8.</p> <p>Tanda panah menunjukkan lokasi yang harus dituju oleh truk dengan melewati jalur sesuai dengan penunjuk arah panah di samping. Lokasi dalam konsep desain di samping berada pada area dermaga.</p>
<p>9.</p>	<p><i>Gameplay concept design no. 1-8</i></p>	<p><i>Gameplay concept</i> pada level ini menerapkan seluruh <i>gameplay concept</i> level-level sebelumnya.</p>

10.	<i>Gameplay concept design no. 1-8</i>	<i>Gameplay concept</i> pada level ini menerapkan seluruh <i>gameplay concept</i> level-level sebelumnya.
11.	<i>Gameplay concept design no. 1-8</i>	<i>Gameplay concept</i> pada level ini menerapkan seluruh <i>gameplay concept</i> level-level sebelumnya.

Sumber : Metodologi dan perancangan.

3.2.2.4. Penempatan Aset

Terrain terbuka biasanya sering membutuhkan banyak objek yang harus ditempatkan di dalamnya untuk meningkatkan dan menciptakan *game* yang menarik.



Gambar 3.8. Level design dalam konteks relasi dengan proses kerja lainnya.

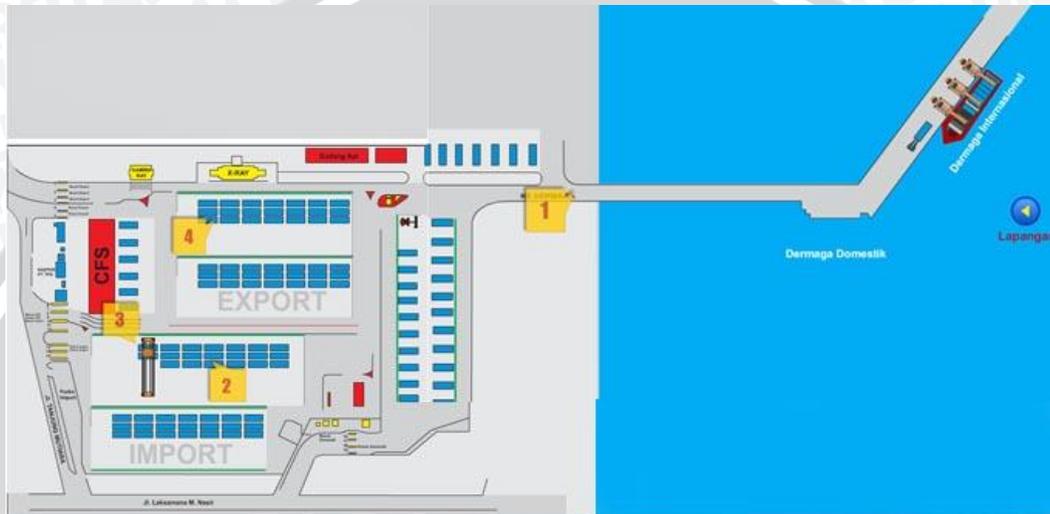
Sumber : [LAB-11]

Dalam Gambar 3.6 menampilkan kedekatan dalam relasi pada proses kerja lainnya selama pembuatan *game*. Garis hitam penuh mendeskripsikan jalannya konten yang melewati proses kerja lainnya selama proses pengembangan *game*. Sedangkan garis putus-putus menunjukkan permintaan dan peralatan yang tidak akan dimasukkan secara langsung dalam akhir proses *output*.

Level Design adalah langkah antara pembuatan konten dan pemrograman. Untuk itu, perubahan dalam level design secara interaktif dibutuhkan atau pengekseskuan secara cepat untuk melihat hasilnya di dalam *game* [LAB-11].

3.2.2.5. Level Design Dalam Game Simulasi Truk Petikemas 3D

Level design dalam *game* ini menggunakan objek dasar yaitu *plane* untuk membuat *environment* seperti jalan, jembatan, dan objek *environment* lainnya sebagai *game area*. Di dalam Blender, aset-aset tersebut kemudian di export dan di import pada proses pengkodean nantinya.



Gambar 3.9. *level design* konsep pemodelan lingkungan.

Sumber : <http://tps.co.id>.

Dalam Gambar 3.7 menampilkan *level design* berdasarkan denah yang didapat dari sumber situs terminal petikemas Surabaya. Pada proses ini pemodelan dirancang sesuai dengan pemetaan lingkungan menjadi beberapa objek dalam bentuk 3D.

3.2.3. Technical Design

Technical design menjelaskan aspek teknis dari pengembangan *game* yang meliputi pembuatan diagram *use case*, diagram *class*, dan diagram *activity*.

3.2.3.1. Use Case

Use case digunakan untuk menunjukkan kebutuhan dari sebuah aplikasi. *Use case* digambarkan dengan sebuah diagram *use case*. Pembuatan diagram *use case* dimulai dengan identifikasi aktor dan identifikasi kebutuhan.

3.2.3.2. Identifikasi Aktor

Identifikasi aktor digunakan untuk menjelaskan interaksi yang dilakukan aktor ketika bermain permainan. Tahap ini akan dijelaskan pada Tabel

Tabel 3.6. Identifikasi Aktor

Aktor	Interaksi	Keterangan
Pemain	Memilih level	Pengguna yang memainkan permainan “Game Simulasi Truk Petikemas 3D” dengan segala interaksi yang dapat dilakukan terhadap permainan.
	Menekan tombol HUD	
	Mengontrol Permainan	

Sumber : Metodologi dan Perancangan

3.2.3.3. Daftar Kebutuhan

Daftar kebutuhan merupakan penjabaran dari kebutuhan fungsional yang ada pada permainan “Game Simulasi Truk Petikemas 3D”, ditunjukkan pada Tabel 3.7.

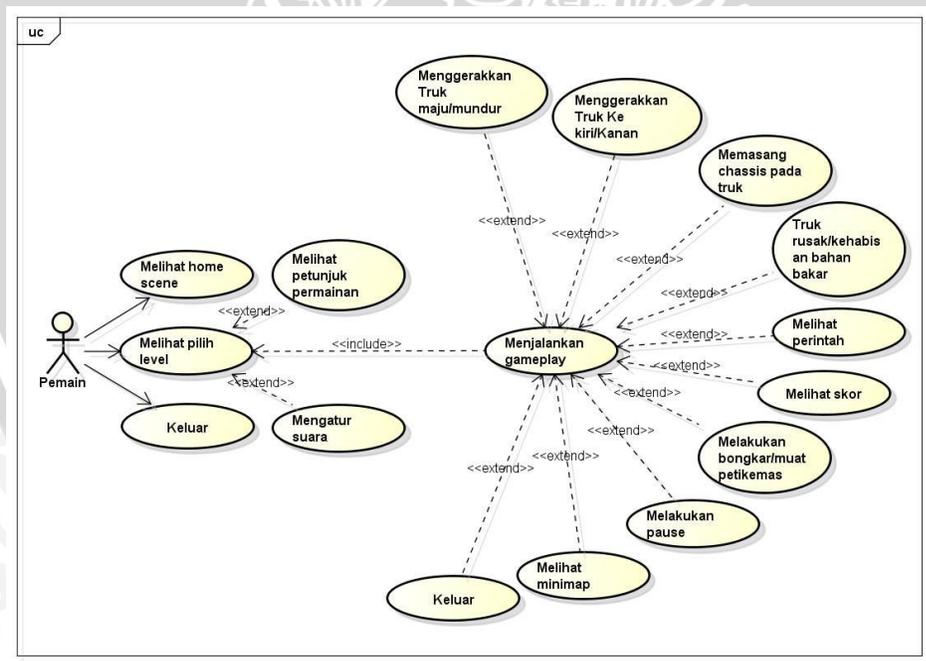
No	Kebutuhan	Use Case
1	<i>Game</i> harus dapat menampilkan halaman <i>Home Scene</i> .	Melihat <i>Home Scene</i>
2	<i>Game</i> harus dapat menampilkan halaman Pilih Level.	Melihat Pilih Level

5	<i>Game</i> harus dapat menampilkan halaman Petunjuk Permainan.	Melihat Petunjuk Permainan
4	<i>Game</i> harus mampu memulai permainan dengan membuka halaman untuk bermain saat pemain menekan tombol suatu level pada Pilih Level.	Menjalankan <i>Gameplay</i>
5	<i>Game</i> harus mampu menggerakkan truk ke kiri/kanan saat pemain menekan tombol kiri/kanan.	Bergerak Kiri/Kanan
6	<i>Game</i> harus mampu menggerakkan truk maju dan mundur saat pemain menekan tombol untuk maju atau mundur.	Bergerak Maju/Mundur
7	<i>Game</i> harus mampu membuat truk dapat terpasang dengan <i>chassis</i> petikemas saat posisi truk telah tepat dengan <i>chassis</i> dan pemain menekan tombol untuk memasang.	Memasang <i>chassis</i> pada truk
8	<i>Game</i> harus mampu membuat truk dapat melakukan muat/ bongkar petikemas saat menggunakan <i>chassis</i> dan pada posisi dan arah yang benar	Melakukan bongkar/ muat petikemas
9	<i>Game</i> harus mampu membuat truk dapat berhenti ketika mesin atau bahan bakar habis	Berhenti bergerak
10	<i>Game</i> harus mampu menampilkan/ tidak menampilkan <i>minimap</i> saat pemain menekan tombol <i>minimap</i>	Melihat <i>minimap</i>
11	<i>Game</i> harus mampu menampilkan/ tidak menampilkan perintah saat pemain menekan tombol perintah	Melihat perintah

12	Game harus mampu melakukan pengaturan suara ketika pemain ingin bermain dengan menggunakan suara atau tidak.	Mengatur <i>Sound</i>
13	Game harus mampu melakukan <i>pause game</i> ketika pemain ingin menghentikan <i>game</i> untuk sementara waktu.	<i>Pause Game</i>
14	Game harus mampu menampilkan skor nilai permainan ketika permainan telah terselesaikan	Menampilkan skor nilai
15	Game harus menyediakan fasilitas untuk keluar dari <i>game</i> . Sehingga pemain dapat mengakhiri permainan dan keluar dari <i>game</i> .	Keluar

Sumber : Metodologi dan Perancangan

3.2.3.4. Diagram Use Case



powered by astah

Gambar 3.10. Diagram use case.

Sumber : Metodologi dan perancangan

Diagram *use case* dalam perancangan “Game Simulasi Truk Petikemas 3D” digunakan untuk memodelkan fungsionalitas dari *game*. Dalam Gambar 3.10 menampilkan diagram *use case* yang melibatkan pengguna sebagai aktor dan beberapa *use case*. Sebelum merancang diagram *use case*, yang harus dilakukan adalah mengidentifikasi aktor dan mengidentifikasi kebutuhan. Aktor dalam permainan ini adalah pemain itu sendiri.

3.2.3.5. Diagram Class

Diagram *class* merupakan gambaran pemodelan elemen-elemen *class* yang membentuk sebuah perangkat lunak. *Class* diperoleh dengan cara menganalisis secara detail *use case* yang dimodelkan. Perancangan diagram *class* ditunjukkan pada Tabel 3.7.

Tabel 3.7. Deskripsi Diagram Class

No	Class	Keterangan
1	CoverScene	<i>Class</i> yang berfungsi sebagai tampilan <i>home scene</i> .
2	Game	<i>Class</i> yang berfungsi proses permainan dieksekusi.
3	MyScene	<i>Class</i> yang berfungsi sebagai halaman utama permainan.
4	MySceneBehavior	<i>Class</i> yang berfungsi sebagai proses halaman utama permainan.
5	LoadScene	<i>Class</i> yang berfungsi sebagai halaman tunggu untuk masuk pada halaman berikut yang dipilih pemain
5	SelectCamera	<i>Class</i> yang berfungsi sebagai proses penentuan kamera
6	LevelBehavior	<i>Class</i> yang berfungsi sebagai penentu level yang dipilih dan instrumen-

		instrumen apa saja yang akan dimuat pada saat permainan
7	RandomBrushBehavior	<i>Class</i> yang berfungsi sebagai pengacak
8	FinalScoreBehavior	<i>Class</i> yang berfungsi sebagai tampilan skor
9	Helper	<i>Class</i> yang berfungsi sebagai pembantu melakukan proses muat asset-aset dan <i>behavior</i> entitas/objek.
10	SteeringBehavior	<i>Class</i> yang berfungsi sebagai proses prosedur melakukan rotasi roda truk kanan/kiri.
11	DriveBehavior	<i>Class</i> yang berfungsi sebagai proses prosedur melakukan rotasi roda truk maju/mundur.
12	ChasisBehavior	<i>Class</i> yang berfungsi sebagai proses prosedur melakukan pemasangan <i>chassis</i>
13	FollowBehavior	<i>Class</i> yang berfungsi sebagai proses prosedur untuk mengikuti suatu objek/entitas.
14	TargetBehavior	<i>Class</i> yang berfungsi sebagai proses prosedur melakukan rotasi arah panah menuju ketempat tujuan saat permainan.
15	MapScene	<i>Class</i> yang berfungsi sebagai tampilan <i>minimap</i> .
16	OrthoCameraBehavior	<i>Class</i> yang berfungsi sebagai pengaturan kamera pada <i>minimap</i>

17	ArrowBehavior	Class yang berfungsi sebagai proses prosedur ikon truk pada <i>minimap</i>
18	FinishBehavior	Class yang berfungsi sebagai proses prosedur ikon lokasi akhir pada <i>minimap</i> .
19	LoadBehavior	Class yang berfungsi sebagai proses prosedur ikon muat petikemas pada <i>minimap</i> .
20	UnloadBehavior	Class yang berfungsi sebagai proses prosedur ikon bongkar petikemas pada <i>minimap</i> .
21	MenuScene	Class yang berfungsi sebagai halaman pilih level..
22	MouseOverButtonBehavior	Class yang berfungsi sebagai proses prosedur tombol ketika mouse berada diatas.
23	MouseRollBehavior	Class yang berfungsi sebagai proses prosedur mouse roll.

Sumber : Metodologi dan Perancangan

Perancangan diagram *abstract class* yang digunakan dalam permainan dideskripsikan pada Tabel 3.8. *Abstract class* pada Tabel 3.8 merupakan class yang secara langsung terdapat pada *game engine Waveengine* tanpa perlu merancang kembali.

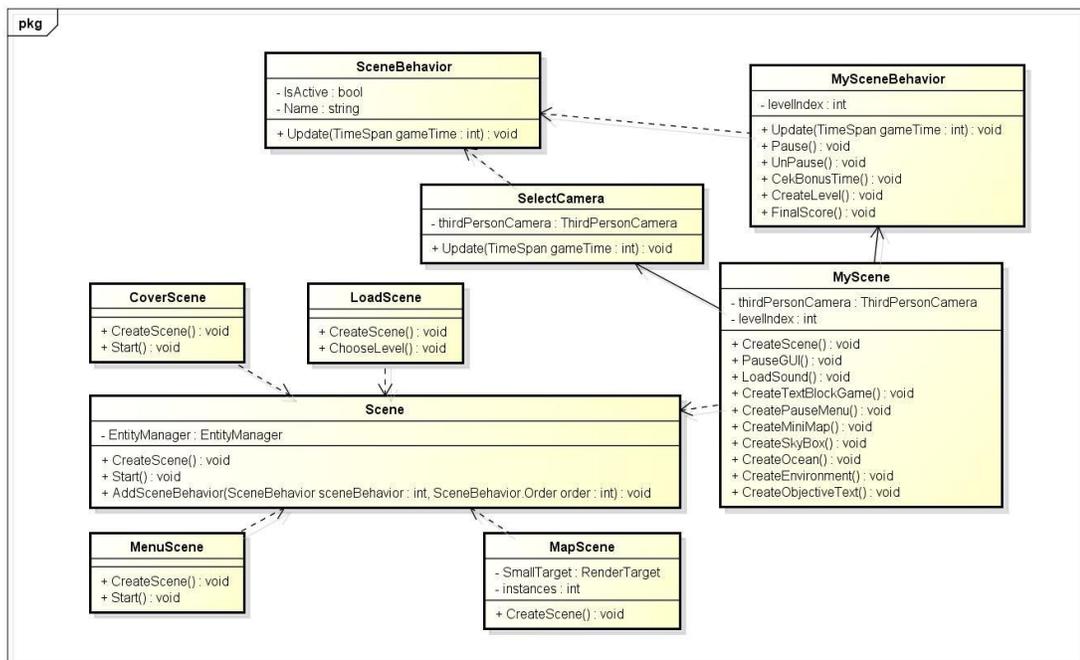
Tabel 3.8. Deskripsi Abstract Class

No.	Abstract	Keterangan
1	Scene	<i>Abstract class</i> yang berfungsi untuk diimplementasikan pada semua <i>class scene</i> .

2	SceneBehavior	<i>Abstract class</i> yang berfungsi untuk diimplementasikan pada semua <i>class SceneBehavior</i> .
3	Behavior	<i>Abstract class</i> yang berfungsi untuk diimplementasikan pada semua <i>class behavior</i> .
4	Component	<i>Abstract class</i> yang berfungsi untuk diimplementasikan pada semua <i>class behavior</i> .

Sumber : Metodologi dan Perancangan

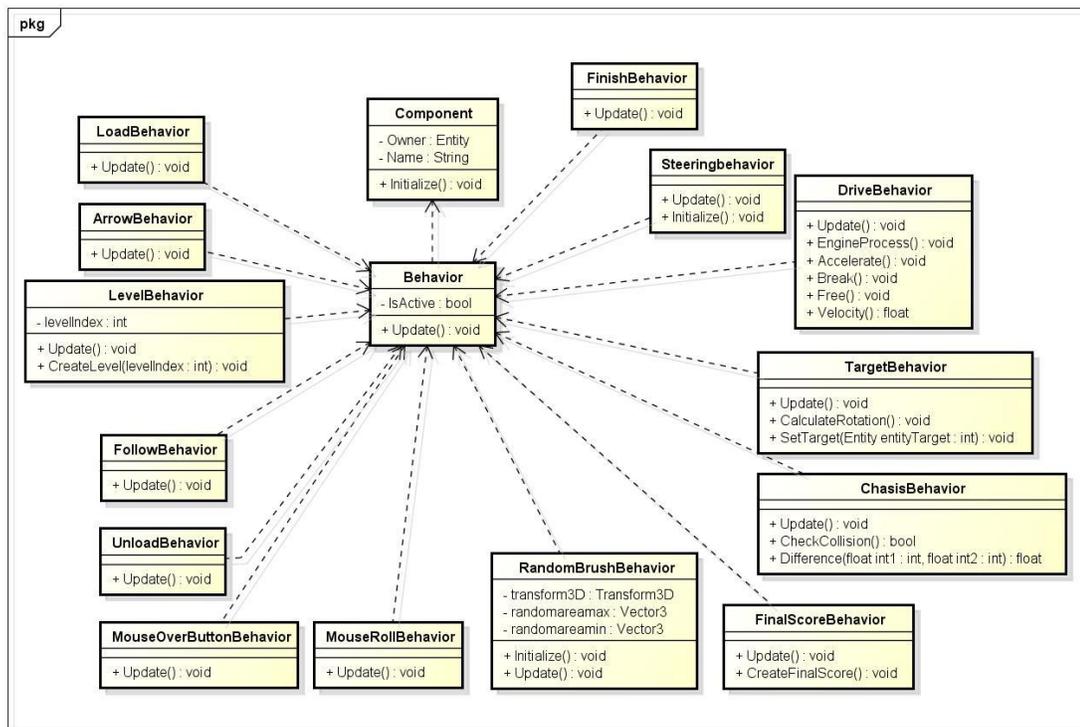
Perancangan diagram class kategori kontrol layar permainan ditunjukkan dalam Gambar 3.11.



Gambar 3.11. Diagram Class – Scene

Sumber : Metodologi dan Perancangan

Perancangan diagram class kategori *behavior* permainan ditunjukkan dalam Gambar 3.12.

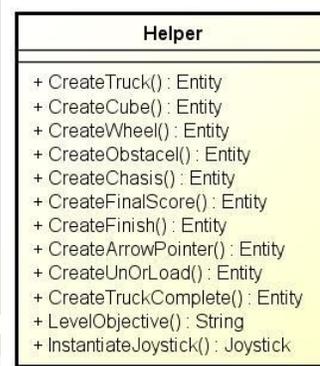


powered by astah

Gambar 3.12. Diagram Class – Behavior

Sumber : Metodologi dan Perancangan

Perancangan diagram class kategori class pembantu permainan ditunjukkan dalam Gambar 3.13.



powered by astah

Gambar 3.13. Diagram Class - Helper

Sumber : Metodologi dan Perancangan

3.2.3.6. Diagram Activity

Diagram *activity* digunakan untuk menjelaskan urutan dari aktivitas. Diagram *activity* menggambarkan alur kerja mulai dari *starting point* hingga *finish point* dengan memberikan detail alur percabangan yang terdapat pada proses suatu *event* dalam aktivitas. Diagram *activity* dari interaksi system dan pemain akan dideskripsikan pada Tabel 3.9.

Tabel 3.9. Diagram Activity

No.	Aktivitas	Diagram	Keterangan
1	Eksekusi Permainan Awal		Pemain melakukan eksekusi permainan pertama kali
2	Pemilihan Level		Pemain memilih level permainan yang akan dimainkan

<p>3</p>	<p>Memulai Permainan</p>		<p>Pemain memainkan permainan hingga permainan selesai, jika pemain berhasil menyelesaikan permainan maka sistem akan menampilkan dialog berhasil, bila tidak dapat menyelesaikan maka akan muncul dialog gagal.</p>
----------	--------------------------	--	--

Sumber : Metodologi dan Perancangan

3.3. Implementasi

Dalam tahap ini, dilakukan implementasi berdasarkan studi literatur dan hasil analisa yang diperoleh sehingga menjadi sebuah *game*. Implementasi yang dilakukan di sini dibagi menjadi 2 yaitu implementasi yang dilakukan di dalam *game engine* meliputi pengumpulan data dan pemodelan dan implementasi yang dilakukan didalam *game engine* yang meliputi prosedur program, kontrol permainan, teknik kamera, alur permainan, desain, dan teknik-teknik pendukung lainnya yang dapat dilakukan di dalam *game engine*.

3.4. Pengujian

Pada tahap ini dilakukan uji “Game Simulasi Truk Petikemas 3D” yang sudah dibuat dan mengevaluasi semua komponen *game* menggunakan pengujian unit, integrasi, validasi, dan pengujian performa berdasarkan nilai FPS (*Frame per Seconds*). Pengujian sistem merupakan hal terpenting yang bertujuan untuk

menemukan kesalahan-kesalahan atau kekurangan-kekurangan pada perangkat lunak yang diuji. Pada pengujian unit dan integrasi, akan digunakan teknik pengujian *White Box* (*White Box Testing*). Pada pengujian validasi akan digunakan teknik pengujian *Black Box* (*Black Box Testing*). Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut.

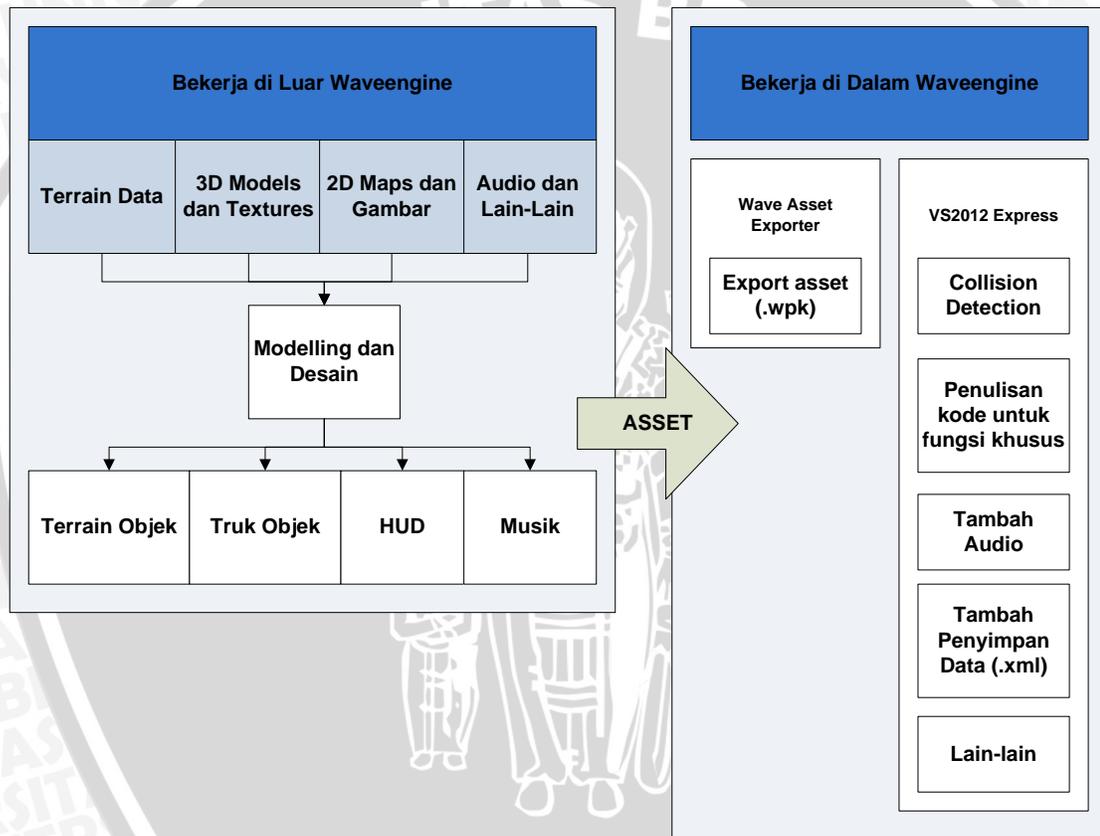
3.5. Penulisan Laporan Skripsi

Pada tahap terakhir ini dimulai penyusunan karya tulis sebagai dokumentasi dari pelaksanaan skripsi. Dokumentasi ini dibuat untuk menjelaskan *game* yang telah dibuat agar memudahkan orang lain yang ingin mengembangkan lebih lanjut.



BAB IV IMPLEMENTASI

Dalam fase implementasi terdapat dua bagian yaitu Objek Integrasi di Dalam *Waveengine* dan Objek integrasi di luar *Waveengine* seperti yang ditampilkan dalam Gambar 4.1.



Gambar 4.1. Diagram alir implementasi.

Sumber : Implementasi

Dalam Gambar 4.1 akan menjelaskan tentang proses implementasi dengan membagi pekerjaan menjadi 2 bagian, yaitu pekerjaan di luar *game engine* (*Waveengine*) dan pekerjaan di dalam *game engine*.

Proses pekerjaan di luar akan masuk dalam pengumpulan data secara garis besar, dan proses pekerjaan di dalam akan masuk dalam pengkodean. Adapun penjelasan yang lebih detil akan dipaparkan pada sub-bab selanjutnya.

4.1. Objek Integrasi di Luar Waveengine

Dalam bekerja di luar *Waveengine* ada beberapa hal penting yang digunakan dalam pembuatan “Game Simulasi Truk Petikemas 3D” yaitu Pengumpulan Data, Modeling, Musik Latar, dan Ekspor.

4.1.1. Pengumpulan Data

Terdapat dua jenis data yang dibutuhkan yaitu spasial data dan atribut data. Data - data tersebut telah dikumpulkan dari 3 sumber :

1. *Game* terrain maps terminal petikemas didapat dari observasi dan mengambil gambar dalam area terminal.
2. Desain arsitektur terminal petikemas dan sketsa area terminal petikemas
3. Tekstur jalan, petikemas, truk, dan lain-lain diperoleh berdasarkan observasi sebelumnya

4.1.2. Modeling

“Game Simulasi Truk Petikemas 3D” environment termasuk jalan, petikemas, kapal, RTG, CC, bangunan, dan lain-lain telah diimplementasikan dalam objek 3D dengan baik. Setelah mendapatkan semua environment objek dalam bentuk 3D kemudian ditempatkan pada posisi yang dibutuhkan dalam *gameplay*. Seperti dalam Gambar 4.2 menampilkan lingkungan terminal petikemas dengan penempatan aset yang sudah mereferensi denah lingkungan petikemas pada proses perancangan.



Gambar 4.2. Pemodelan lingkungan *game*.

Sumber : Implementasi

4.1.3. Musik Latar

Musik dan suara di dalam *game* sangat penting untuk meningkatkan emosional pemain ketika bermain *game* agar lebih menjiwai permainan, oleh karena itu *game* sangat lekat dengan penggunaan musik. Tanpa musik seakan *game* tidak menarik.

4.1.4. Ekspor

Ketika semua model telah selesai dan disimpan menjadi format *.x* yang dapat dibaca oleh *wave asset exporter*. Mulai dari tekstur dengan format *.jpg* /*.png*, model, dan suara/musik dengan format *.wav* diekspor menjadi file *.wpk* yang akan dibaca oleh *Waveengine* dalam pengkodean, sedangkan untuk file musik/suara berformat *.mp3* dapat langsung digunakan kedalam pengkodean.

4.2. Objek Integrasi di Dalam *Waveengine*

Dalam bekerja di dalam *Waveengine* ada beberapa hal penting yang digunakan dalam pembuatan “Game Simulasi Truk Petikemas 3D” yaitu Implementasi Prosedur Program, Implementasi Kontrol Permainan, Implementasi Kamera, dan Implementasi Alur *Game*.

4.2.1. Implementasi Prosedur Program

“Game Simulasi Truk Petikemas 3D” ini memiliki beberapa proses atau *method*. Beberapa *method* yang akan dicantumkan dalam penulisan makalah skripsi ini hanya untuk algoritma dari beberapa proses (operasi) utama saja, sehingga tidak semua *method* akan dicantumkan. Adapun beberapa algoritma tersebut adalah:

4.2.1.1. Implementasi Akselerasi Truk

Dalam prosedur program akselerasi truk terdapat beberapa proses yang digunakan agar truk dapat bergerak maju dan mundur dengan percepatan seperti pergerakan kendaraan yang sebenarnya.

Tabel 4.1. Prosedur Akselerasi Truk

Pseudocode Prosedur untuk Akselerasi Truk	
DEKLARASI AWAL	
Float[]	kcptnTrtnggi;
Float[]	akselerasi;
Float[]	kcptnMndrMaks;
Float	kcptn;
Int	i;
1	DESKRIPSI
2	Mulai
3	Jika Maju
4	Maka, Jika kcptn kurang dari kcptnTrtnggi[i]
5	Maka i++, kcptn <- kcptnTrtnggi[i]+akslerasi[i]
6	
7	Jika Mundur
8	Maka, Jika kcptn kurang dari kcptnTrtnggi[i] dan
9	kurang dari (-kcptnMndrMaks[i])
10	Maka i--, kcptn <-
11	kcptnTrtnggi[i]+akslerasi[i]

```

12
13     Jika Diam
14     Maka, Jika kcptn kurang dari kcptnTrtnggi[i-1]
15     Maka kcptn <- (-akslerasi[i])
Selesai

```

Sumber : Implementasi

Penjelasan dari prosedur untuk akselerasi truk pada Tabel 4.1 adalah:

1. Baris 3 – 5 merupakan prosedur untuk melakukan pergerakan maju dengan percepatan yang dihasilkan, sehingga terjadi pergerakan pada truk bergerak maju dengan penambahan kecepatan berdasarkan tingkat akselerasi yang telah disimpan di dalam array. Jika tombol maju semakin lama ditekan maka truk akan bergerak semakin cepat hingga mencapai batas maksimum kecepatan tertiggi.
2. Baris 7 – 10 merupakan prosedur untuk melakukan pergerakan mundur dengan percepatan yang dihasilkan, sehingga terjadi pergerakan pada truk bergerak mundur dengan penambahan kecepatan berdasarkan tingkat akselerasi yang telah disimpan di dalam array. Jika tombol mundur semakin lama ditekan maka truk akan bergerak semakin cepat hingga mencapai batas maksimum kecepatan untuk mundur.
3. Baris 12 – 14 merupakan prosedur dimana tidak adanya tombol yang ditekan sebagai pemicunya, sehingga jika pada saat truk sedang bergerak maju/mundur dan seketika tombol maju/mundur tidak ditekan, kecepatan yang ada pada truk akan dikurangi dengan nilai minus akselerasi. Kondisi ini akan menimbulkan akselerasi pada perlambatan truk.

4.2.1.2. Implementasi Kendali Roda Truk

Dalam prosedur program kendali roda truk terdapat beberapa proses yang digunakan agar roda truk dapat berputar ke kiri dan kanan. Prosedur ini membantu pergerakan truk dapat ke kanan dan ke kiri seperti kendaraan sebenarnya.

Tabel 4.2. Prosedur Kendali Roda Truk

Pseudocode Prosedur untuk Kendali Roda Truk
DEKLARASI AWAL

```

Float maxAngle;
Vector3 kndlRts;// kendaliRotasi
1  DESKRIPSI
2  Mulai
3    Jika, Kanan
4    Maka, Jika kndlRts.Y kurang dari maxAngle
5    Maka kndlRts.Y <- kndlRts.Y + 1
6
7    Jika Kiri
8    Maka, Jika kndlRts.Y kurang dari (-maxAngle)
9    Maka kndlRts.Y <- kndlRts.Y - 1
10 Selesai

```

Sumber : Implementasi

Penjelasan dari prosedur untuk akselerasi truk pada Tabel 4.2 adalah:

1. Baris 3 – 5 merupakan prosedur untuk melakukan kendali roda depan truk untuk berotasi dengan sumbu vertikal (Y) ke kanan. Jika tombol kanan ditekan maka roda akan berotasi dengan penjumlahan rotasi sebesar +1.
2. Baris 3 – 5 merupakan prosedur untuk melakukan kendali roda depan truk untuk berotasi dengan sumbu vertikal (Y) ke kiri. Jika tombol kiri ditekan maka roda akan berotasi dengan penjumlahan rotasi sebesar -1.

4.2.1.3. Implementasi Pemasangan Chassis

Dalam prosedur program pemasangan *chassis* terdapat beberapa proses yang digunakan agar truk dapat terpasang dengan *chassis* dengan memperhitungkan posisi antara truk dan *chassis* sehingga digunakan objek pemicu antar keduanya agar terpasang sesuai.

Tabel 4.3. Prosedur Pemasangan Chassis

Pseudocode Prosedur untuk Pemasangan Chassis	
DEKLARASI AWAL	
ENTITY Truk	
ENTITY Chasis	
Vector3 trukPemicu	
Bool IsReady, CekCollision	
1	DESKRIPSI
2	Mulai

```

3   Truk add ChasisBehavior(Chasis)
4   //chasis behavior
5   Vector3 jarak
6   Jika jarak <- Chasis.position - trukPemicu kurang
7   dari sama dengan .2f
8   Maka, IsReady <- true
9       Jika Baca masukkan kontrol 'E'/'O'/'3'
10      Maka Chasis Joint Truk
11  Jika Tidak Maka, IsReady <- false
      Selesai

```

Sumber : Implementasi

Penjelasan dari prosedur untuk akselerasi truk pada Tabel 4.3 adalah:

1. Baris 3 merupakan pemberian sifat pada truk agar dapat memasang *chassis*.
2. Baris 5 – 11 merupakan prosedur ChasisBehavior yang di dalamnya terdapat prosedur – prosedur agar truk dapat memasang *chassis*. Baris 6 – 7 merupakan seleksi kondisi dimana selisih jarak *chassis* dan truk adalah 0.2f. Jika hasil *true* maka kondisi truk akan siap terpasang *chassis*. Bila kondisi sudah siap terpasang pemain dapat menekan tombol (E/O/3) untuk mengaitkan *chassis* dengan truk. Jika hasil *false* maka posisi antara *chassis* dan truk tidak tepat.

4.2.1.4. Implementasi Angkut Petikemas

Dalam prosedur program angkut petikemas terdapat beberapa proses yang digunakan agar truk dapat melakukan proses angkut petikemas. Proses ini mewajibkan truk menggunakan *chassis* sebelum melakukan kegiatan bongkar/muat petikemas.

Tabel 4.4. Prosedur Angkut Petikemas

Pseudocode Prosedur untuk Angkut Petikemas	
DEKLARASI AWAL	
Bool ChasisOn, angkut, kosong, akses;	
Vector3 ChasisPos, angkutPos, letakPos;	
1	DESKRIPSI
2	Mulai
3	Vector3 jarak
4	Jika, ChasisOn <- true
5	Maka, jika, jarak <- ChasisPos-angkutPos kurang
6	dari 1 dan Akses = true dan angkut = false

```
7     Maka, angkut <- true
8     kosong <- false
9     Lain jika, jarak <- ChasisPos-letakPos kurang
10    dari 1 dan Akses = true dan kosong = false
11    Maka, kosong <- true,
12    angkut <- false
13    Selesai
```

Sumber : Implementasi

Penjelasan dari prosedur untuk akselerasi truk pada Tabel 4.4 adalah:

1. Baris 4 merupakan seleksi kondisi dimana menentukan kondisi *chassis* telah terpasang dengan truk atau tidak.
2. Baris 5 – 8 merupakan seleksi kondisi untuk proses angkut yang akan tereksekusi ketika baris 4 menghasilkan nilai *true*. Seleksi kondisi menentukan posisi *chassis* yang terpasang pada truk tepat pada area bongkar/muat (*angkutPos*) dan Akses harus sama dengan *true* (artinya yaitu arah masuk area bongkar/muat harus sesuai), serta angkut harus sama dengan *false* (artinya yaitu *chassis* belum mengangkut petikemas). Jika hasil *true* maka angkut akan *true* (petikemas terangkut) dan kosong menjadi *false* (artinya yaitu *chassis* dalam keadaan tidak kosong).
3. Baris 9 – 12 merupakan seleksi kondisi untuk proses peletakan petikemas yang akan tereksekusi ketika baris 4 menghasilkan nilai *true*. Seleksi kondisi menentukan posisi *chassis* yang terpasang pada truk tepat pada area bongkar/muat (*letakPos*), dan Akses harus sama dengan *true* (artinya yaitu arah masuk area bongkar/muat harus sesuai), serta angkut harus sama dengan *true* (artinya yaitu *chassis* mengangkut petikemas). Jika hasil *true* maka angkut akan *false* (*chassis* telah diletakan) dan kosong menjadi *true* (artinya yaitu *chassis* dalam keadaan tidak mengangkut petikemas/kosong).

4.2.1.5. Implementasi Nilai Ukur Kerusakan Mesin

Dalam prosedur program pada “Game Simulasi Truk Petikemas 3D” terdapat beberapa proses yang digunakan agar truk dapat memiliki nilai ukur kerusakan mesin. Kerusakan mesin terjadi ketika truk melakukan tumbukan/ tabrakan dengan objek lain.

Tabel 4.5. Prosedur Nilai Ukur Kerusakan Mesin

Pseudocode Prosedur untuk Nilai Ukur Kerusakan Mesin	
DEKLARASI AWAL	
Int MeterKerusakan	
Entity Truk	
1	DESKRIPSI
2	Mulai
3	Jika, <code>Truk.RigidBody3D.onPhysic3DCollision</code> <-
4	<code>Physic3DCollisionEventHandler()</code>
5	Maka, Jika, MeterKerusakan lebih dari 0
6	Maka, <code>MeterKerusakan</code> <- <code>MeterKerusakan - 1</code>
7	Selesai

Sumber : Implementasi

Penjelasan dari prosedur untuk akselerasi truk pada Tabel 4.5 adalah:

1. `RigidBody3D` adalah komponen pada *game engine Waveengine* yang memberikan efek fisika pada objek dalam “Game Simulasi Truk Petikemas 3D”.
2. Baris 3 - 5 merupakan seleksi kondisi dimana menentukan kondisi badan truk bertumbukan/ bertabrakan dengan objek yang memiliki `RigidBody3D` lainnya.
3. Baris 5 – 6 merupakan seleksi kondisi yang akan tereksekusi ketika baris 3 – 5 menghasilkan nilai *true*. Pada prosedur ini menentukan nilai ukur kerusakan, jika hasil yang didapat adalah *true* maka nilai ukur kerusakan akan dikurangi sejumlah -1.

4.2.1.6. Implementasi Nilai Ukur Jumlah Bahan Bakar

Dalam prosedur program pada “Game Simulasi Truk Petikemas 3D” terdapat beberapa proses yang digunakan agar truk dapat memiliki nilai ukur jumlah bahan bakar. Jumlah bahan bakar berkurang terjadi ketika truk bergerak dengan roda yang berotasi ke depan atau ke belakang.

Tabel 4.6. Prosedur Nilai Ukur Jumlah Bahan Bakar

Pseudocode Prosedur untuk Nilai Ukur Jumlah Bahan Bakar	
DEKLARASI AWAL	
Int MeterBensin	
Entity Truk	

```

1  DESKRIPSI
2  Mulai
3      Entity rodaTruk <- Truk.FindChild("Wheel")
4      Jika, rodaTruk.Rotasi lebih dari 360° + 100f
5      Maka, MeterBensin <- MeterBensin - 1
6  Selesai

```

Sumber : Implementasi

Penjelasan dari prosedur untuk akselerasi truk pada Tabel 4.6 adalah:

1. Baris 3 merupakan prosedur untuk memasukkan nilai pada variabel entity rodaTruk menjadi entity anak dari truk yaitu roda.
2. Baris 4 – 5 merupakan seleksi kondisi yang menentukan rotasi terhadap sumbu horizontal sebesar 360 derajat ditambah 100 sebagai nilai penyeimbang untuk pengurangan nilai ukur bahan bakar dengan jarak pergerakan truk. Jika hasil tersebut adalah *true* maka nilai ukur bahan bakar akan dikurangi sejumlah -1. Pada prosedur ini nilai ukur bahan bakar dipengaruhi oleh putaran roda truk.

4.2.2. Implementasi Kontrol Permainan

Dalam “Game simulasi truk petikemas 3D” terdapat 3 alat kontrol yang dapat digunakan yaitu *Mouse*, *Keyboard*, dan *Joystick*.

4.2.2.1. Mouse

Mouse dalam *game* simulasi truk petikemas digunakan untuk memilih setiap menu atau tombol yang muncul di permainan dengan cara menekan bagian kiri *mouse*.

4.2.2.2. Keyboard

Fungsi *keyboard* dalam permainan ini adalah

ESC : Untuk menutup *game*

Key Up : Maju

Key Down : Mundur

Key Left : Putar roda ke kiri

Key Right : Putar roda ke kanan

Key E : Mengaitkan chasis

Key P : *Pause*

Key Num 1 : Kamera tampak depan truk

Key Num 2 : Kamera tampak belakang truk

4.2.2.3. Joystick

Fungsi *joystick* dalam permainan ini adalah untuk mengontrol truk berjalan. Di dalam implementasi kontrol permainan ini joystick yang digunakan adalah pabrikan *Logitech*. berikut uraiannya.

Key Y/ Key 2 : Maju

Key X/ Key 1 : Mundur

Key O/ Key 3 : Mengaitkan chasis

Key Left : Putar roda ke kiri

Key Right : Putar roda ke kanan

Key Start/ Key 10 : *Pause*

4.2.3. Implementasi Kamera

Third person camera adalah teknik sorot permainan yang akan digunakan dalam perancangannya, permainan ini menggunakan objek target yaitu truk. Dalam Gambar 4.3 menampilkan hasil implementasi kamera pada “Game Simulasi Truk Petikemas 3D”.



Gambar 4.3. Hasil implementasi penggunaan *third person camera*.

Sumber : Implementasi

Dalam Gambar 4.3 menampilkan hasil penggunaan *third person camera* dimana objek target yaitu truk menjadi target kamera yang akan selalu mengikuti perubahan posisi dan rotasi truk.

4.2.4. Implementasi Alur Permainan



Gambar 4.4. Alur *Game* Simulasi Truk Petikemas 3D.

Sumber : Implementasi

Alur *game* adalah alur permainan dimulai dari tampilan awal hingga permainan berlangsung. Berikut alur “Game Simulasi Truk Petikemas 3D” dalam Gambar 4.4. Selain layar-layar yang sering ditampilkan pada permainan, biasanya dalam permainan juga memiliki kondisi *pause* dan *goal* atau tujuan permainan telah diselesaikan. Kondisi *pause* dan *goal* ditampilkan dalam Gambar 4.5 dibawah ini.



Gambar 4.5. Kondisi *pause* dan *goal*.

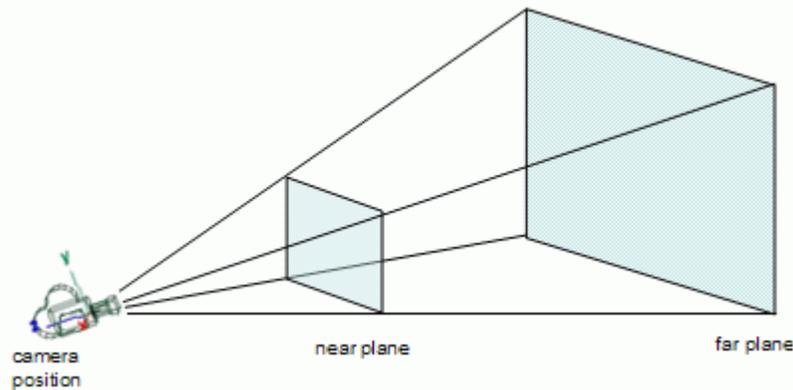
Sumber : Implementasi

4.2.5. Implementasi Collision Detection

Terdapat beberapa *collision* yang diberikan pada objek-objek tertentu dalam “Game Simulasi Truk Petikemas 3D”. *Collision* digunakan untuk membuat objek yang saling bersentuhan mendeteksi objek sentuh sehingga terjadi adanya gaya *physic* yang tampak nyata.

4.2.6. Implementasi Frustum Culling

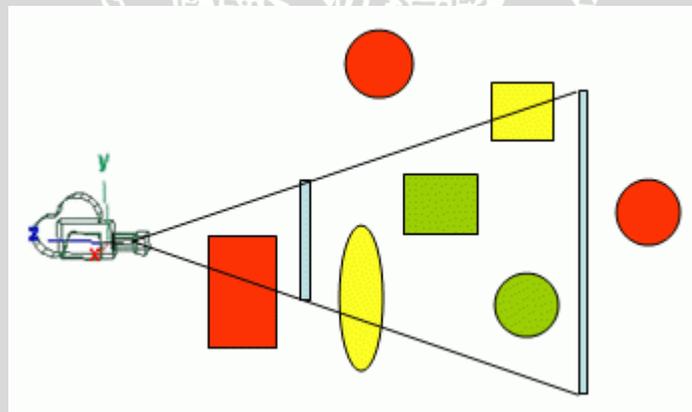
Frustum merupakan volume yang di dalamnya memiliki semua yang berpotensi (terdapat *occlusion*) untuk terlihat di atas layar. Volume ini bergantung pada pengaturan kamera, dan ketika menggunakan proyeksi perspektif akan membuat bentuk potongan piramida seperti dalam Gambar 4.6. Puncak piramida adalah posisi kamera, dan dasar piramida adalah batas jauh penglihatan kamera (*plane* terjauh). Piramida dipotong pada batas penglihatan terdekat (*plane* terdekat) karenanya disebut dengan frustum [LIG-11].



Gambar 4.6. Tampilan frustum.

Sumber : [LIG-11]

Culling sendiri merupakan teknik *rendering* yang digunakan untuk mereduksi proses *rendering* objek dalam area yang sudah ditentukan. Sehingga teknik *frustum culling* ini dapat membantu meningkatkan performa permainan dikarenakan tidak perlu *me-render* objek diluar area frustum.



Gambar 4.7. Frustum culling.

Sumber : [LIG-11]

Dalam Gambar 4.7 semua objek berwarna hijau secara total masuk di dalam area *frustum* dan semua objek berwarna kuning yang sebagian masuk di dalam area *frustum* juga di-*render* sedangkan semua objek berwarna merah yang tidak sedikitpun masuk di dalam area *frustum* tidak akan di-*render*.

4.2.7. Implementasi HUD

Terdapat beberapa HUD di dalam “Game Simulasi Truk Petikemas 3D”. HUD digunakan untuk menambah informasi yang dapat membantu permainan saat berlangsung. Dalam Gambar 4.3 dan 4.4 di setiap alur permainan menampilkan beberapa HUD yang digunakan dalam menunjang permainan ketika berlangsung.

Dari seluruh alur permainan yang ada pada “Game Simulasi Truk Petikemas 3D” memiliki HUD yang akan dijelaskan fungsionalnya pada Tabel 4.7 sebagai berikut.

Tabel 4.7. Informasi keterangan fungsional HUD pada layar utama

Nama	Gambar	Keterangan
Timer		Digunakan untuk menunjukkan batas waktu tempuh permainan.
Indikator kerusakan		Digunakan untuk mengindikasikan kerusakan truk.
Indikator bahan bakar		Digunakan untuk mengindikasikan bahan bakar truk.
Tombol panah		Digunakan untuk menampilkan panah penunjuk arah tujuan.
Indikator petikemas		Digunakan untuk mengindikasikan petikemas.
Indikator chasis		Digunakan untuk mengindikasikan chasis.
Speedo Meter		Digunakan untuk menunjukkan kecepatan truk.

Tombol manajer		Digunakan untuk menampilkan petunjuk manajer.
Tombol GPS		Digunakan untuk menampilkan mini map.
Tombol "Play Now"		Digunakan untuk melanjutkan permainan pada layar level.
Tombol "How To Play"		Digunakan untuk menampilkan tata cara permainan
Tombol "Back to Home"		Digunakan untuk kembali pada layar awal
Tombol "musik/suara"		Digunakan untuk mematikan suara/musik
Petikemas terbuka		Kondisi ini membolehkan pemain untuk masuk ke level tersebut
Petikemas tertutup		Kondisi ini tidak memperbolehkan pemain untuk masuk ke level tersebut
Bintang		Menunjukkan kemampuan pemain berdasarkan skor. Tampilan bintang antara 1-3, dengan 3 adalah skor tertinggi
Tombol ulang		Digunakan untuk mengulang permainan
Tombol kembali ke layar level		Digunakan untuk kembali ke layar level
resume		Digunakan untuk mengkondisikan pada

		posisi <i>unpause</i> setelah <i>pause</i>
Tombol lanjut		Digunakan untuk melanjutkan permainan langsung ke level selanjutnya

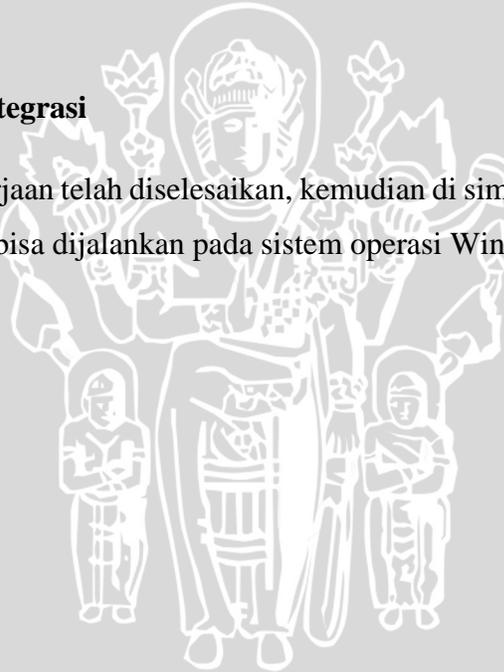
Sumber : Implementasi

4.2.8. Implementasi Aset-Aset

Suara, Musik, dan aset-aset lain yang dibutuhkan dan telah di ekspor kedalam bentuk format .wpk di import kedalam pengkodean menggunakan VS2012 dalam folder content.

4.2.9. Implementasi Integrasi

Ketika semua pekerjaan telah diselesaikan, kemudian di simpan / eksekusi dalam bentuk format .exe yang bisa dijalankan pada sistem operasi Windows.

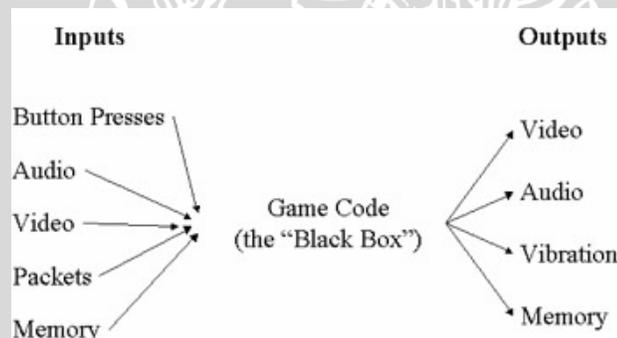


BAB V

PENGUJIAN DAN ANALISIS

Pengujian merupakan salah satu proses penting dalam pembuatan ataupun pengembangan sebuah aplikasi. Pengujian sangat penting karena untuk dapat membantu pengembangan selanjutnya. Adapun beberapa pengujian yang akan dilakukan pada “Game Simulasi Truk Petikemas 3D” diantaranya yaitu pengujian validitas (*Black Box Testing*), pengujian unit, dan pengujian integrasi (*White Box Testing*), serta pengujian performa *game* yang diukur berdasarkan nilai FPS (*Frames per Second*).

5.1. Pengujian Validitas



Gambar 5.1. Pengujian Black box-masukan dan keluaran.

Sumber : [SCH-05].

Pengujian validasi merupakan pengujian yang berdasar pada pengujian *black box*. Dalam Gambar 5.1 menampilkan cara pengujian berdasarkan *black box*. Pengujian *black box* merupakan cara paling efektif untuk menguji jaringan dan modul sistem yang sangat kompleks menjadi sederhana [SCH-05]. Pengujian ini digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan.

Daftar kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *black box*, karena tidak diperlukan konsentrasi terhadap alur jalannya algoritma program. Adapun hasil analisis uji validasi ditampilkan dalam Tabel 5.9.

5.2. Pengujian Unit

Pengujian unit merupakan kebalikan dari pengujian *black box* atau biasa disebut dengan pengujian *white box*. Pengujian *white box* memberikan penguji sebuah kesempatan untuk mengamati pengkodean secara langsung tanpa pengguna akhir nantinya yang pernah bisa [SCH-05]. Pada pengujian unit permainan ini adalah mengacu pada beberapa fungsionalitas truk diantaranya yaitu (1) prosedur pergerakan akselerasi, (2) kendali roda truk, (3) pemasangan *chasis* petikemas, (4) bongkar-muat petikemas, (5) nilai ukur kerusakan mesin, dan (6) nilai ukur jumlah bahan bakar (bensin). Semua pengujian unit ini akan diintegrasikan pada pembuatan truk yang memiliki kemampuan dan atau sifat-sifat di dalam permainan.

5.2.1. Prosedur Pergerakan Akselerasi

Pada pengujian pergerakan akselerasi yang akan menjadi acuan untuk pengujian adalah algoritma pergerakan akselerasi truk yang kemudian digambarkan menjadi *flow graph* dan kemudian dihitung kompleksitas siklomatisnya, yang mana akan menghasilkan jalur-jalur berjalannya algoritma.

Tabel 5.1. Prosedur Fungsi Pergerakan Akselerasi Truk

Pseudocode Akselerasi	Flow Graph
<p>DEKLARASI AWAL Float[] kcptnTrtnggi; Float[] akselerasi; Float[] kcptnMndrMaks; Float kcptn; Int i;</p> <p>DESKRIPSI Mulai ————— 1 Jika Maju ————— 2 Maka, Jika kcptn kurang dari } kcptnTrtnggi[i] } 3 Maka i++, kcptn <- } kcptnTrtnggi[i]+akslerasi[i] } 5 Jika Mundur ————— 4 Maka, Jika kcptn kurang dari } kcptnTrtnggi[i] dan kurang dari (- } 6 kcptnMndrMaks[i]) } Maka i--, kcptn <- } kcptnTrtnggi[i]+akslerasi[i] } 8 Jika Diam ————— 7 Maka, Jika kcptn kurang dari } kcptnTrtnggi[i-1] } 9 Maka kcptn <- (-akslerasi[i]) - } 10 Selesai ————— 11</p>	<pre> graph TD 1((1)) --> 2((2)) 2 --> 3((3)) 2 --> 4((4)) 3 --> 5((5)) 4 --> 6((6)) 4 --> 7((7)) 5 --> 8((8)) 6 --> 8 7 --> 9((9)) 8 --> 10((10)) 9 --> 10 10 --> 11((11)) </pre>

Sumber : Pengujian dan Analisis

Pada Tabel 5.1 pemodelan algoritma ke dalam *flow graph* yang telah dilakukan untuk akselerasi menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 12 - 11 + 2 = 3
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 3 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1-2-3-5-11

Jalur 2 : 1-2-4-6-8-11

Jalur 3 : 1-2-4-7-9-10-11

5.2.2. Prosedur Kendali Roda

Pada pengujian kendali roda yang akan menjadi acuan untuk pengujian adalah algoritma pergerakan kendali roda truk yang kemudian digambarkan menjadi *flow graph* dan kemudian dihitung kompleksitas siklomatisnya, yang mana akan menghasilkan jalur-jalur berjalannya algoritma.

Tabel 5.2. Prosedur Kendali Roda Truk

Pseudocode Kendali Roda Truk	Flow Graph
<pre> DEKLARASI AWAL Float maxAngle; Vector3 kndlRts;// kendaliRotasi DESKRIPSI Mulai _____ 1 Jika, Kanan _____ 2 Maka, Jika kndlRts.Y kurang dari maxAngle } 4 Maka kndlRts.Y <- kndlRts.Y + 1 _____ 6 Jika Kiri _____ 3 Maka, Jika kndlRts.Y kurang dari (-maxAngle) } 5 Maka kndlRts.Y <- kndlRts.Y - _____ 7 1 _____ 8 Selesai _____ </pre>	<pre> graph TD 1((1)) --> 2((2)) 1((1)) --> 3((3)) 2((2)) --> 4((4)) 2((2)) --> 5((5)) 3((3)) --> 5((5)) 4((4)) --> 6((6)) 5((5)) --> 7((7)) 6((6)) --> 8((8)) 7((7)) --> 8((8)) </pre>

Sumber : Pengujian dan Analisis

Pada Tabel 5.2 pemodelan ke dalam *flow graph* yang telah dilakukan untuk kendali roda truk menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah

kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$V(G) = E - N + 2$$

$$= 8 - 8 + 2 = 2$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 3 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1-2-4-6-8

Jalur 2 : 1-3-5-7-8

5.2.3. Prosedur Pemasangan Chasis Petikemas

Pada pengujian pemasangan *chasis* yang akan menjadi acuan untuk pengujian adalah algoritma pemasangan *chasis* truk yang kemudian digambarkan menjadi *flow graph* dan kemudian dihitung kompleksitas siklomatisnya, yang mana akan menghasilkan jalur-jalur berjalannya algoritma.

Tabel 5.3. Prosedur Pemasangan Chasis

Pseudocode Pemasangan Chasis Petikemas	Flow Graph
<pre> DEKLARASI AWAL ENTITY Truk ENTITY Chasis Vector3 trukPemicu Bool IsReady, CekCollision DESKRIPSI Mulai _____ 1 Truk add ChasisBehavior(Chasis) _____ 2 Vector3 jarak _____ 3 Jika jarak <- Chasis.position - trukPemicu kurang dari sama dengan .2f } 4 Maka, IsReady <- true _____ 5 Jika Baca masukkan kontrol 'E'/'O'/'3' } 7 Maka Chasis Joint Truk _____ 8 Jika Tidak Maka, IsReady <- false _____ 6 Selesai _____ 9 </pre>	<pre> graph TD 1((1)) --> 2((2)) 2 --> 3((3)) 3 --> 4((4)) 4 --> 5((5)) 4 --> 6((6)) 5 --> 7((7)) 7 --> 8((8)) 6 --> 9((9)) 8 --> 9 </pre>

Sumber : Pengujian dan Analisis

Pada Tabel 5.3 pemodelan ke dalam *flow graph* yang telah dilakukan untuk pemasangan *chasis* truk menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}V(G) &= E - N + 2 \\ &= 10 - 9 + 2 = 3\end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 3 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1-2-3-4-5-7-9

Jalur 2 : 1-2-3-4-5-7-8-9

Jalur 3 : 1-2-3-4-6-9

5.2.4. Prosedur Angkut Petikemas

Pada pengujian angkut petikemas yang akan menjadi acuan untuk pengujian adalah algoritma angkut petikemas pada keadaan truk berada pada titik picu melakukan proses angkut petikemas pada permainan yang kemudian digambarkan menjadi *flow graph* dan kemudian dihitung kompleksitas siklomatisnya, yang mana akan menghasilkan jalur-jalur berjalannya algoritma.

Tabel 5.4. Prosedur Angkut Petikemas

Pseudocode Angkut Petikemas	Flow Graph
<p>DEKLARASI AWAL Bool ChasisOn, angkut, kosong, akses; Vector3 ChasisPos, angkutPos, letakPos;</p> <p>DESKRIPSI Mulai ————— 1 Vector3 jarak ————— 2 Jika, ChasisOn <- true ————— 3 Maka, jika, jarak <- ChasisPos- angkutPos kurang dari 1 dan Akses = true dan angkut = false } 4 Maka, angkut <- true } 5 kosong <- false } Lain jika, jarak <- ChasisPos- letakPos kurang dari 1 dan Akses } 6 = true dan kosong = false Maka, kosong <- true, } 7 angkut <- false } Selesai ————— 8</p>	<pre> graph TD 1((1)) --> 2((2)) 2 --> 3((3)) 3 --> 4((4)) 3 --> 8((8)) 4 --> 5((5)) 4 --> 6((6)) 5 --> 7((7)) 6 --> 7 7 --> 8 </pre>

Sumber : Pengujian dan Analisis

Pada Tabel 5.4 pemodelan ke dalam *flow graph* yang telah dilakukan untuk prosedur angkut petikemas menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 9 - 8 + 2 = 3
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 3 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1-2-3-4-6-7-8

Jalur 2 : 1-2-3-4-5-8

Jalur 3 : 1-2-3-8

5.2.5. Prosedur Nilai Ukur Kerusakan Mesin

Pada pengujian nilai ukur kerusakan mesin yang akan menjadi acuan untuk pengujian adalah algoritma nilai ukur kerusakan mesin truk pada permainan yang kemudian digambarkan menjadi *flow graph* dan kemudian dihitung kompleksitas siklomatisnya, yang mana akan menghasilkan jalur-jalur berjalannya algoritma.

Tabel 5.5. Prosedur Nilai Ukur Kerusakan Mesin

Pseudocode Nilai Ukur Kerusakan Mesin	Flow Graph
<p>DEKLARASI AWAL Int MeterKerusakan Entity Truk</p> <p>DESKRIPSI Mulai _____ 1 Jika, Truk.RigidBody3D.onPhysic3DCollision } 2 <- Physic3DCollisionEventHandler() } Maka, Jika, MeterKerusakan lebih dari 0 } 3 Maka, MeterKerusakan <- } 4 MeterKerusakan - 1 } Selesai _____ 5</p>	<pre> graph TD 1((1)) --> 2((2)) 2 --> 3((3)) 2 --> 4((4)) 3 --> 4 </pre>

Sumber : Pengujian dan Analisis

Pada Tabel 5.5 pemodelan ke dalam *flow graph* yang telah dilakukan untuk nilai ukur kerusakan mesin menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 4 - 4 + 2 = 2
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 3 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1-2-3-4

Jalur 2 : 1-2-4

5.2.6. Prosedur Niali Ukur Jumlah Bahan Bakar

Pada pengujian nilai ukur jumlah bahan bakar yang akan menjadi acuan untuk pengujian adalah algoritma nilai ukur jumlah bahan bakar truk pada permainan yang kemudian digambarkan menjadi *flow graph* dan kemudian dihitung kompleksitas siklomatisnya, yang mana akan menghasilkan jalur-jalur berjalannya algoritma.

Tabel 5.6. Prosedur Nilai Ukur Jumlah Bahan Bakar

Pseudocode Nilai Ukur Jumlah Bahan Bakar	Flow Graph
<p>DEKLARASI AWAL Int MeterBensin Entity Truk</p> <p>DESKRIPSI Mulai ——— 1 Entity rodaTruk <- Truk.FindChild("Wheel") } 2 Jika, rodaTruk.Rotasi lebih dari 360° + 100f } 4 Maka, MeterBensin <- MeterBensin - 1 — 3 Selesai ——— 5</p>	<pre> graph TD 1((1)) --> 2((2)) 2 --> 3((3)) 3 --> 4((4)) 3 --> 5((5)) 4 --> 5 </pre>

Sumber : Pengujian dan Analisis

Pada Tabel 5.6 pemodelan ke dalam *flow graph* yang telah dilakukan untuk bongkar-muat petikemas menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 5 - 3 + 2 = 2
 \end{aligned}$$

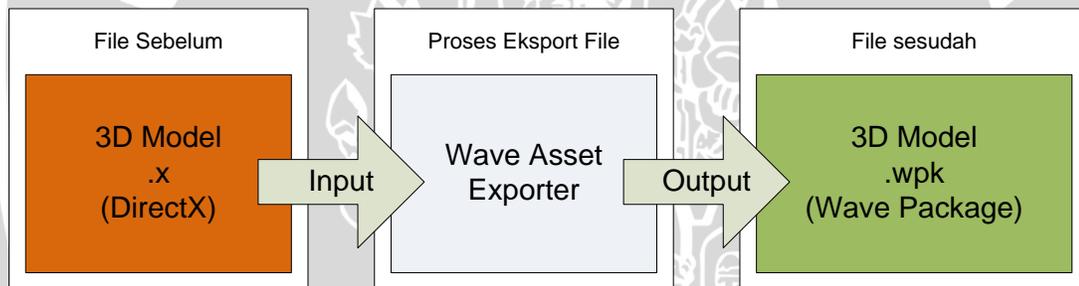
Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 2 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1-2-3-4-5

Jalur 2 : 1-2-3-5

5.3. Pengujian Integrasi

Pembuatan model truk 3D sebagai objek utama dalam *game* simulasi truk petikemas dibuat menggunakan perangkat lunak Blender dan untuk dapat dieksekusi dalam *framework* dibutuhkan konversi file dari *.x* menjadi *.wpk*. Berikut flow diagram dan prosedur pembuatan truk yang di tampilkan dalam Gambar 5.2.



Gambar 5.2. Proses ekspor objek utama.

Sumber : Pengujian dan Analisis

Setelah memperoleh file ekspor *.wpk* kemudian dilakukan pemanggilan objek dengan prosedur pemanggilan pada *framework Waveengine*. Setelah mendapatkan objek truk di dalam permainan pada prosedur pemanggilan diintegrasikan dengan unit-unit fungsionalitas truk menjadi beberapa *behavior* truk atau tingkah laku yang dimiliki truk sesuai dengan pengujian unit sebelumnya di dalam satu entitas yaitu truk.

Tabel 5.7. Prosedur Pemanggilan Truk

Pseudocode Pembuatan Objek Truk Dalam Waveengine	Flow Graph
<p>DEKLARASI AWAL Camera maincamera Entity truk</p> <p>DESKRIPSI Mulai SetCamera(maincamera) Buat truk("direktori_file") Buat 4 Roda Roda Joint truk truk add kendaliRodaDepanObj truk add akselerasi truk add ChasisBehavior truk add UkuranKerusakan truk add UkuranBahanBakar EntityManager(truk) Selesai</p>	

Sumber : Pengujian dan Analisis

Pada Tabel 5.7 pemodelan ke dalam *flow graph* yang telah dilakukan untuk pembuatan objek truk menghasilkan jumlah kompleksitas siklomatis (*cyclomatic complexity*) melalui persamaan $V(G) = E - N + 2$, dimana $V(G)$ merupakan jumlah kompleksitas siklomatis, E merupakan sisi atau *edge* (garis penghubung antar *node*) dan N merupakan jumlah simpul (*node*).

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 &= 0 - 1 + 2 = 1
 \end{aligned}$$

Berdasarkan dari nilai *cyclomatic complexity* yang telah didapatkan dari perhitungan maka ditentukan 1 buah basis set dari jalur *independent*, yaitu :

Jalur 1 : 1

5.4. Pengujian Performa

Pada pengujian performa permainan, FPS (*Frames per Second*) menjadi nilai ukur untuk mendapatkan hasil uji yang sesuai yaitu memperoleh permainan yang berjalan dengan lancar dan stabil. Dikarenakan pada pengujian FPS ini menggunakan nilai ukur, maka untuk menjadikan batas ukuran baku yang sesuai diambil sampel nilai ukur ketika permainan berjalan lancar dan stabil diperoleh nilai sebesar 60 FPS. Di dalam pengujian akan dimulai pengukuran dari awal mula permainan dieksekusi hingga permainan dalam satu level berakhir. Level yang akan menjadi sampel pengujian adalah level 9 dikarenakan pada level tersebut truk akan mengelilingi hampir seluruh area terminal. Diambilnya sampel level tersebut agar mendapatkan hasil pengujian yang valid. Pengujian ini akan dibantu dengan menggunakan aplikasi *Fraps* (tampilan aplikasi dalam Gambar 5.3) untuk mendokumentasikan perubahan FPS setiap detiknya.



Gambar 5.3. Perangkat lunak Fraps

Sumber : [FRA-11]

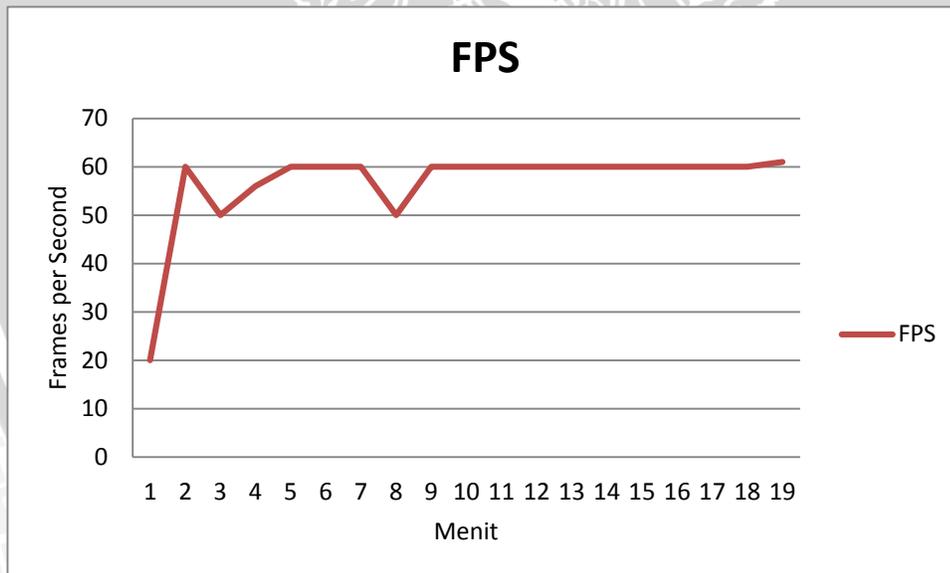
Adapun spesifikasi perangkat keras yang menjadi perangkat uji performa dijabarkan pada Tabel 5.8.

Tabel 5.8. Spesifikasi perangkat keras

Spesifikasi	Nama/Jenis/Versi Spesifikasi
OS	Windows 8 Pro 64-bit (6.2, Build 9200) (9200.win8_gdr.130531-1504)
Processor	Intel(R) Core(TM) i5-2320 CPU @ 3.00GHz (4 CPUs), ~3.0GHz
Memory	4096MB RAM
DirectX Version	DirectX 11
VGA Card	AMD Radeon HD 6450

Sumber : Pengujian dan Analisis

Berikut hasil uji performa berdasar FPS yang diperoleh dengan bantuan aplikasi *Fraps* untuk merekam setiap detik terjadinya perubahan FPS yang digambarkan dalam diagram kurva dalam Gambar 5.4.



Gambar 5.4. Diagram kurva hasil uji performa berdasarkan FPS.

Sumber : Pengujian dan Analisis

5.5. Analisis Pengujian Validitas

Tabel 5.9. Analisis Pengujian Validasi

No	Kasus Uji	Hasil
1.	Melihat <i>Home Scene</i>	Valid
2.	Melihat Pilih Level	Valid
3.	Melihat Petunjuk Permainan	Valid
4.	Menjalankan <i>Gameplay</i>	Valid
5.	Bergerak Kiri/Kanan	Valid
6.	Bergerak Maju/Mundur	Valid
7.	Memasang <i>chassis</i> pada truk	Valid
8.	Melakukan bongkar/ muat petikemas	Valid
9.	Berhenti bergerak	Valid
10.	Melihat <i>minimap</i>	Valid
11.	Melihat perintah	Valid
12.	Mengatur <i>Sound</i>	Valid
13.	<i>Pause Game</i>	Valid
14.	Menampilkan skor nilai	Valid
15.	Keluar	Valid

Sumber : Pengujian dan Analisis

Berdasarkan kesesuaian antara hasil uji terhadap implementasi dan fungsionalitas “Game Simulasi Truk Petikemas 3D” pada Tabel 5.9 menghasilkan

output yang diharapkan dalam daftar kebutuhan *game* ini. Dapat disimpulkan bahwa implementasi dan fungsionalitas *game* telah memenuhi kebutuhan yang telah dijabarkan dalam daftar kebutuhan. Pada saat pengujian *black box* ini tidak ada permasalahan di dalam menguji kebutuhan permainan sehingga dapat di katakan hasil pengujian ini berhasil.

5.6. Analisis Pengujian Unit

Pada pengujian pergerakan akselerasi truk menggunakan kompleksitas siklomatis (*cyclomatic complexity*) dihasilkan 3 jalur kasus uji yang di deskripsikan pada Tabel 5.10.

Tabel 5.10. Analisis Pengujian Pergerakan Akselerasi Truk

Jalur	Kasus Uji	Penekanan Joystick / Keyboard	Hasil yang diharapkan	Hasil yang di Dapatkan
1	Maju	2/Up Arrow	Truk bergerak maju	Truk bergerak maju
2	Mundur	1/Down Arrow	Truk bergerak mundur	Truk bergerak mundur
3	Diam/Perlam batan	Tidak ada penekanan	Truk dalam keadaan diam/ perlambatan	Truk dalam keadaan diam/ perlambatan

Sumber : Pengujian dan Analisis

Berdasarkan kasus uji pada Tabel 5.10 yang sesuai dengan jalur hasil dari kompleksitas siklomatis, menghasilkan *output* yang sesuai dengan harapan. Berikut penjabaran Tabel 5.10.

1. Jalur 1 merupakan jalur prosedur melakukan pergerakan maju pada truk dengan akselerasi. Kondisi ini terjadi jika pemain menekan tombol maju (2/Up Arrow).
2. Jalur 2 merupakan jalur prosedur melakukan pergerakan mundur pada truk dengan akselerasi. Kondisi ini terjadi jika pemain menekan tombol maju (1/Down Arrow).
3. Jalur 3 merupakan jalur prosedur melakukan perlambatan pada truk yang sedang bergerak dan diam bila tidak ada penekanan sebelumnya.

Pada pengujian kendali roda truk menggunakan kompleksitas siklomatis (*cyclomatic complexity*) dihasilkan 2 jalur kasus uji yang di deskripsikan pada Tabel 5.11.

Tabel 5.11. Analisis Pengujian Kendali Roda Truk

Jalur	Kasus Uji	Penekanan Joystick/ Keyboard	Hasil yang diharapkan	Hasil yang di Dapatkan
1	Roda depan berputar ke kanan	Right/Right Arrow	Roda truk depan berotasi searah sumbu vertical ke kanan	Roda truk depan berotasi searah sumbu vertical ke kanan
2	Roda depan berputar ke kiri	Left/Left Arrow	Roda truk depan berotasi searah sumbu vertical ke kiri	Roda truk depan berotasi searah sumbu vertical ke kiri

Sumber : Pengujian dan Analisis

Berdasarkan kasus uji pada Tabel 5.11 yang sesuai dengan jalur hasil dari kompleksitas siklomatis, menghasilkan *output* yang sesuai dengan harapan. Berikut penjabaran Tabel 5.11.

1. Jalur 1 merupakan jalur prosedur melakukan putar roda depan ke kanan pada sumbu vertikal (Y). Kondisi ini terjadi jika pemain menekan tombol kanan (Right/Right Arrow).
2. Jalur 2 merupakan jalur prosedur melakukan putar roda depan ke kiri pada sumbu vertikal (Y). Kondisi ini terjadi jika pemain menekan tombol kiri (Left/Left Arrow).

Pada pengujian pemasangan *chassis* menggunakan kompleksitas siklomatis (*cyclomatic complexity*) dihasilkan 3 jalur kasus uji yang di deskripsikan pada Tabel 5.12.

Tabel 5.12. Analisis Pengujian Pemasangan Chasis

Jalur	Kasus Uji	Penekanan Joystick/ Keyboard	Hasil yang diharapkan	Hasil yang di Dapatkan
1	Posisi belakang truk berada tepat pada bagian depan chasis	Tidak ada penekanan	Chasis tidak terpasang dengan truk	Chasis tidak terpasang dengan truk
2	Posisi belakang truk berada tepat pada bagian depan chasis	3/O/E	Chasis terpasang dengan truk	Chasis terpasang dengan truk
3	Posisi belakang truk tidak berada tepat pada bagian depan chasis	3/O/E	Chasis tidak terpasang dengan truk	Chasis tidak terpasang dengan truk

Sumber : Pengujian dan Analisis

Berdasarkan kasus uji pada Tabel 5.12 yang sesuai dengan jalur hasil dari kompleksitas siklomatis, menghasilkan *output* yang sesuai dengan harapan. Berikut penjabaran Tabel 5.12.

1. Jalur 1 merupakan jalur prosedur melakukan pemasangan *chassis* dengan kondisi posisi tepat pada titik picu pemasangan *chassis* dengan tanpa penekanan kontrol permainan, sehingga *chassis* tidak terpasang.
2. Jalur 2 merupakan jalur prosedur melakukan pemasangan *chassis* dengan kondisi posisi tepat pada titik picu pemasangan *chassis* dengan penekanan kontrol permainan, sehingga *chassis* dapat terpasang dengan baik pada truk.
3. Jalur 3 merupakan jalur prosedur melakukan pemasangan *chassis* dengan kondisi posisi tidak tepat pada titik picu pemasangan *chassis* dengan penekanan kontrol permainan, sehingga *chassis* tidak dapat terpasang pada truk.

Pada pengujian angkut petikemas menggunakan kompleksitas siklomatis (*cyclomatic complexity*) dihasilkan 3 jalur kasus uji yang di deskripsikan pada Tabel 5.13.

Tabel 5.13. Analisis Pengujian Angkut Petikemas

Jalur	Kasus Uji (Chassis)	Kegiatan	Hasil yang diharapkan	Hasil yang di Dapatkan
1	Terpasang pada truk, terisi petikemas, dan posisi dan arah yang tepat pada lokasi bongkar/muat	Kegiatan Bongkar/ Muat Petikemas	Petikemas menghilang/diletakkan	Petikemas menghilang/diletakkan
2	Terpasang pada truk, kosong, dan posisi dan arah yang tepat pada lokasi bongkar/muat		Petikemas diangkut	Petikemas diangkut
3	Tidak terpasang pada truk		Tidak terjadi proses angkut/ peletakan petikemas	Tidak terjadi pengangkutan/ peletakan

Sumber : Pengujian dan Analisis

Berdasarkan kasus uji pada Tabel 5.13 yang sesuai dengan jalur hasil dari kompleksitas siklomatis, menghasilkan *output* yang sesuai dengan harapan. Berikut penjabaran Tabel 5.13.

1. Jalur 1 merupakan jalur prosedur melakukan peletakan petikemas dengan kondisi *chassis* telah terpasang pada truk dengan mengangkat petikemas, dan berada pada posisi dan arah yang tepat pada lokasi bongkar/ muat. Sehingga dengan kondisi tersebut akan meletakkan/ menghilangkan petikemas dari atas *chassis*.
2. Jalur 2 merupakan jalur prosedur melakukan pengangkutan petikemas dengan kondisi *chassis* telah terpasang pada truk dan tanpa mengangkat petikemas, serta berada pada posisi dan arah yang tepat pada lokasi bongkar/ muat. Sehingga dengan kondisi tersebut akan membuat truk mengangkat petikemas.

- Jalur 3 merupakan jalur prosedur kondisi *chassis* tidak terpasang pada truk, sehingga dengan kondisi tersebut truk tidak dapat mengangkut/ meletakkan petikemas.

Pada pengujian nilai ukur kerusakan mesin menggunakan kompleksitas siklomatis (*cyclomatic complexity*) dihasilkan 2 jalur kasus uji yang di deskripsikan pada Tabel 5.14.

Tabel 5.14. Analisis Pengujian Nilai Ukur Kerusakan Mesin

Jalur	Kasus Uji	Objek (RigidBody 3D)	Hasil yang diharapkan	Hasil yang di Dapatkan
1	Truk menabrak sesuatu objek	Pin / Halang rintang	HUD indikator mesin berkurang	HUD indikator mesin berkurang
		Petikemas	HUD indikator mesin berkurang	HUD indikator mesin berkurang
		Bangunan	HUD indikator mesin berkurang	HUD indikator mesin berkurang
		RTG/CC	HUD indikator mesin berkurang	HUD indikator mesin berkurang
2	Truk tidak menabrak sesuatu objek	-	HUD indikator mesin tidak berkurang	HUD indikator mesin tidak berkurang

Sumber : Pengujian dan Analisis

Berdasarkan kasus uji pada Tabel 5.14 yang sesuai dengan jalur hasil dari kompleksitas siklomatis, menghasilkan *output* yang sesuai dengan harapan. Berikut penjabaran Tabel 5.14.

- Jalur 1 merupakan jalur prosedur kerusakan mesin/ pengurangan nilai ukur mesin pada tampilan *game* dengan kondisi dimana sesaat truk bertumbukan/ bertabrakan dengan objek yang memiliki komponen *RigidBody3D*. Sehingga seketika nilai ukur mesin pada tampilan *game* akan berkurang.
- Jalur 2 merupakan jalur prosedur kerusakan mesin/ pengurangan nilai ukur mesin pada tampilan *game* dengan kondisi dimana sesaat truk tidak bertumbukan/ bertabrakan dengan objek yang memiliki komponen *RigidBody3D*. Sehingga nilai ukur mesin pada tampilan *game* tidak akan berubah.

Pada pengujian nilai ukur jumlah bahan bakar menggunakan kompleksitas siklomatis (*cyclomatic complexity*) dihasilkan 2 jalur kasus uji yang di deskripsikan pada Tabel 5.15.

Tabel 5.15. Analisis Pengujian Nilai Ukur Jumlah Bahan Bakar

Jalur	Kasus Uji	Roda Truk	Hasil yang diharapkan	Hasil yang di Dapatkan
1	Truk berjalan	Berputar $360^0 + 100f$	HUD indikator bahan bakar berkurang	HUD indikator bahan bakar berkurang
2	Truk tidak berjalan	Tidak berputar	HUD indikator bahan bakar tidak berkurang	HUD indikator bahan bakar tidak berkurang

Sumber : Pengujian dan Analisis

Berdasarkan kasus uji pada Tabel 5.15 yang sesuai dengan jalur hasil dari kompleksitas siklomatis, menghasilkan *output* yang sesuai dengan harapan. Berikut penjabaran Tabel 5.15.

1. Jalur 1 merupakan jalur prosedur pengurangan nilai ukur bahan bakar pada tampilan *game* dengan kondisi dimana roda truk telah berputar sebesar ($360^0 + 100f$). Putaran penuh roda truk dan penambahan angka 100f merupakan cara untuk membuat hubungan jarak pergerakan yang dilakukan truk dan pengurangan bahan bakar seimbang.
2. Jalur 2 merupakan jalur prosedur pengurangan nilai ukur bahan bakar pada tampilan *game* dengan kondisi dimana roda truk tidak berputar sebesar ($360^0 + 100f$). Sehingga tidak terjadi pengurangan bahan bakar.

Hasil analisis yang didapatkan dari pengujian unit yang berdasarkan kesesuaian antara hasil pengujian tiap unit dengan *output* unit seperti pada Tabel 5.10-5.15, maka dapat diambil kesimpulan bahwa unit modul dari program sudah sesuai dengan *output* yang diharapkan.

5.7. Analisis Pengujian Integrasi

Tabel 5.16. Analisis Pengujian Pembuatan Truk

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang di Dapatkan
1	Membuat entitas truk dan meletakkannya pada Entity Manager untuk ditampilkan dalam layar utama dan Fungsionalitas setiap unitnya	Menampilkan truk pada layar utama dengan parameter yang sudah ditentukan	Menampilkan truk pada layar utama dengan parameter yang sudah ditentukan

Sumber : Pengujian dan Analisis

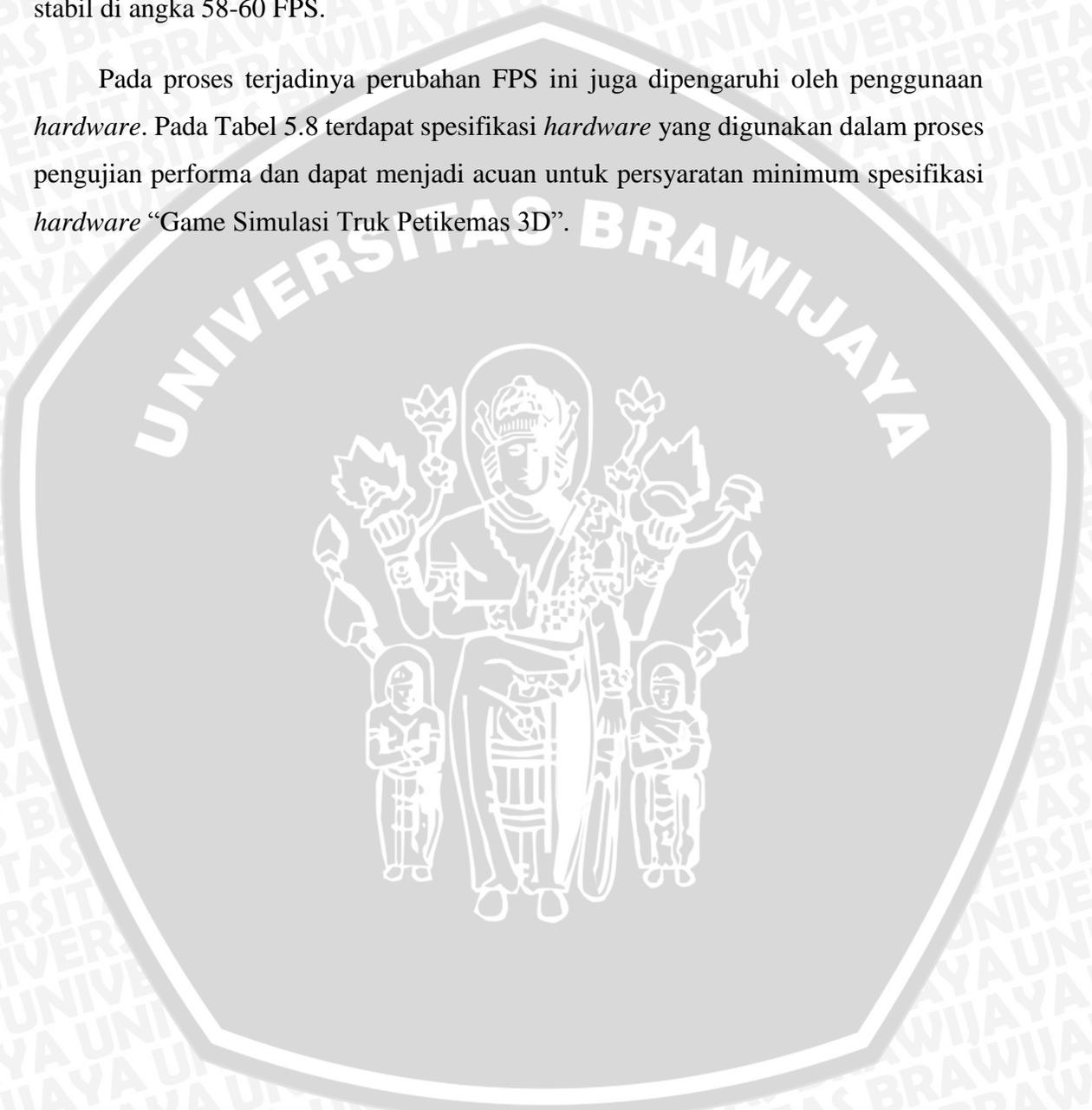
Hasil analisis yang didapatkan dari pengujian integrasi berdasarkan kesesuaian antara hasil pengujian integrasi dengan *output* integrasi yang dituliskan pada Tabel 5.7, dan kemudian di analisis berdasarkan fungsional permainan seperti pada Tabel 5.16 dengan menghasilkan 1 jalur kasus uji maka dapat diambil kesimpulan bahwa modul integrasi dari program sudah sesuai dengan *output* yang diharapkan. Kondisi ini dibuktikan pada Tabel 5.16 truk dapat ditampilkan di dalam *game* dengan beberapa unit yang diintegrasikan pada objek truk berjalan dengan baik.

5.8. Analisis Pengujian Performa

Berdasarkan hasil uji dalam Gambar 5.4 yang menampilkan FPS dalam bentuk diagram kurva mendapatkan FPS sampai ketitik normal yaitu sebesar 60 FPS. Dalam analisis pengujian performa berdasarkan Gambar 5.4 terdapat beberapa kali FPS berubah turun kemudian kembali stabil. Ketika permainan dieksekusi, FPS di mulai pada posisi 20 FPS kemudian naik ke angka 60 FPS ketika layar awal permainan ditampilkan dan ketika beberapa saat sebelum masuk pada layar level, FPS kembali turun di angka 50 FPS kemudian naik kembali dan stabil. Sesaat sebelum memulai permainan atau setelah memilih level yang tersedia, FPS kembali turun di angka 50 dan naik kembali ketika permainan berlangsung dan beranjak stabil hingga permainan selesai. Ketika permainan berlangsung sempat beberapa saat perubahan naik dan turun

terjadi namun tidak terlalu signifikan. Perubahan tersebut terjadi dikarenakan ketika kamera yang menampilkan area pepohonan dan rerumputan FPS berkurang dan kemudian ketika tidak menampilkan area pepohonan FPS kembali naik dan cenderung stabil di angka 58-60 FPS.

Pada proses terjadinya perubahan FPS ini juga dipengaruhi oleh penggunaan *hardware*. Pada Tabel 5.8 terdapat spesifikasi *hardware* yang digunakan dalam proses pengujian performa dan dapat menjadi acuan untuk persyaratan minimum spesifikasi *hardware* “Game Simulasi Truk Petikemas 3D”.



BAB VI

PENUTUP

6.1. Kesimpulan

Kesimpulan yang didapatkan dari hasil pengujian adalah sebagai berikut :

1. “Game Simulasi Truk Petikemas 3D” adalah permainan yang mengungkap jenis *game* simulasi kendaraan. Kendaraan yang digunakan di dalam permainan adalah truk dan menggunakan lingkungan terminal petikemas. Permainan ini memiliki 11 level dengan kesulitan yang berbeda dan semakin susah seiring dengan tingginya level yang akan dimainkan. Metodologi dan perancangan permainan secara umum dimulai dari 3 fase yaitu fase observasi, perancangan *game*, dan implementasi. Pada fase implementasi dibagi lagi menjadi dua sistem kerja objek integrasi yaitu di luar dan di dalam *game engine* (*Waveengine*). Pada proses pengujian skripsi ini telah diuji menggunakan beberapa metode yaitu dengan *black box*, *white box*, dan performa berdasarkan FPS (*Frames per Second*).
2. Berdasarkan hasil pengujian validitas menggunakan metode *black box*, didapatkan keseluruhan fungsional aplikasi permainan dapat berjalan sesuai daftar kebutuhan yang telah dibuat dan sesuai dengan diagram *use case*.
3. Berdasarkan hasil pengujian unit dan integrasi diketahui dengan menggunakan metode *white box* yang menguji fungsional truk di dalam permainan diantaranya yaitu akselerasi, kendali roda, pemasangan *chassis*, angkut petikemas, pengukuran kerusakan mesin, dan bahan bakar. Dari beberapa uji unit dan integrasi tersebut didapatkan kasus uji terbanyak adalah prosedur akselerasi truk, pemasangan *chassis*, dan angkut petikemas dengan memiliki 3 buah basis set dari jalur independent berdasarkan nilai *cyclomatic complexity*.

4. Berdasarkan hasil pengujian performa dengan berdasar pada nilai FPS normal yaitu 60 FPS, dengan bantuan perangkat lunak *Fraps* dihasilkan nilai FPS cukup stabil meskipun pada beberapa saat terjadi penurunan nilai FPS hingga 20%, hal tersebut dikarenakan pergantian *scene* yang membutuhkan proses muat seluruh aset. Selain itu pengaruh perangkat keras yang digunakan juga sangat mempengaruhi perubahan FPS. Sehingga pada pengujian performa dapat diketahui persyaratan spesifikasi minimum perangkat keras yang harus digunakan untuk menjalankan “Game Simulasi Truk Petikemas 3D”.

Dari beberapa kesimpulan di atas diharapkan dapat digunakan menjadi sebuah dasar yang berguna untuk penelitian selanjutnya dan tentunya *game* ini juga dapat digunakan untuk pengembangan diri, baik dari segi keterampilan ataupun dari segi ilmu.

6.2. Saran

Adapun saran-saran yang dapat digunakan untuk pengembangan aplikasi permainan yang selanjutnya sebagai berikut.

1. Pada permainan “Game Simulasi Truk Petikemas 3D” belum di dapati penggunaan AI (*Artificial Intelligence*) untuk kendaraan lainnya.
2. Penambahan animasi pengangkutan petikemas dari RTG/ *container crane* pada truk juga dapat ditambahkan dalam pengembangan selanjutnya.
3. Dalam membuat sebuah pemodelan lingkungan / *terrain* yang besar ada perlunya menggunakan teknik *culling* untuk meningkatkan performa permainan.
4. Pengembangan aplikasi ini juga dapat dengan menambahkan kegiatan-kegiatan di dalam terminal petikemas lainnya seperti untuk operator RTG, *container crane*, atau yang lainnya.
5. Pengembangan *game* ini juga dapat dilakukan secara bebas, namun ada baiknya untuk tetap mencantumkan nama pembuat sebelumnya agar dapat dijadikan referensi untuk pengembangan *game* kedepannya.

6. Pada penelitian selanjutnya dapat dilakukan pengujian pada fase *post-production* dengan 4 tahapan umum sebagai berikut:

- 1) Pre-Alpha testing
- 2) Alpha testing
- 3) Beta testing
- 4) Release



DAFTAR PUSTAKA

- ANO-09 Anonim, 2009, “*Game Genres*” diakses pada 8 Mei 2013
(<http://jccict.wikidot.com/call-of-duty-5>)
- BLE-13 Roosendaal, Ton. Blender 2.69 a. Perangkat Lunak tersedia di
<http://www.blender.org/download/>
- BOB-98 Bobic, Nick. 1998. *Rotating Object Using Quaternions*. diakses pada 25
November 2013
(http://www.gamasutra.com/view/feature/131686/rotating_objects_using_quaternions.php?print=1)
- FAS-13 Fassihi, Amir. “0 – 60 fps in 14 days!” What we learned trying to
optimize our *game* using Unity3D. diakses pada 27 Desember 2013.
(http://www.gamasutra.com/blogs/AmirFassihi/20130828/199134/0__60_fps_in_14_days_What_we_learned_trying_to_optimize_our_game_using_Unity3D.php)
- FLA-10 Flavell, Lance. 2010. *Beginning Blender: Open Source 3D Modeling, Animation, and Game Design*. USA : Paul Manning.
- FRA-11 Fraps 3.5.99. Beepa Pty Ltd, 2011; Perangkat Lunak tersedia di
<http://www.fraps.com/download.php>
- GAO-09 Gao, David Wenzhong, Chris Mi, dan Ali Emadi. 2009. *Modeling and Simulation of Electric and Hybrid Vehicles*. IEEE : University of Michigan Library
- JON-12 Jonathan, William. 2012. *Character Development in Blender 2.5*. Course Technology Cengage Learning.

- LAB-11 Labschutz, Matthias, Katharina Krost. *Content Creation for a 3D Game with Maya and Unity 3D*. 2011. Vienna University of Technology, Austria
- LIG-11 Arf. View Frustum Culling. diakses pada tanggal 28 Desember 2013. (www.lighthouse3d.com/tutorials/view-frustum-culling/)
- PED-09 Pedersen, Roger E. 2009. *Game Design Foundation*. 2nd ed. Jones and Bartlett : United States of America
- PRE-10 Pressman, Roger S. 2010. "Software Engineering : A Practitioner's Approach, Seventh Edition". McGraw Hill.
- SCH-05 Schultz, Charles P., Robert Bryant, dan Tim Langdell. 2005. *Game Testing All in One*. Thomson Course Technology PTR : United States of America
- SHI-05 Shiratori, Rei, Kiyoshi Arai dan, Fumitoshi Kato. 2005. *Gaming Simulations, and Society Research Scope and Perspective*. Springer : Verlag Tokyo
- SYA-13 Syahfitri, Dina. 2013. Simulasi Pencarian Jarak Terdekat (Shortest Path) Dengan Menggunakan Algoritme A* (Studi Kasus Pada Perpustakaan Politeknik Negeri Medan) diakses pada 8 mei 2013 (<http://repository.usu.ac.id/handle/123456789/35192>).
- KOK-12 Wawancara dengan Kokoh Wahyu Utomo. Instruksi Kerja Operator Truk Petikemas di PT. GAMA MITRA PERKASA, Tanjung Perak, Surabaya(Juli, 2, 2012).
- HID-08 Hidayatullah, Priyanto, Amrullah, Zaky. 2008. *Making Educational Animation using Flash*. Bandung : Informatika Bandung.

- MAC-05 Mac Glivery, Carrol and Anthony Head. 2005. *3D for the web interactive animation using 3ds max, flash and directer*. Oxford. Focal Press. Burlington.
- RES-NN Resmi, Novi. Simulasi Kreatif Dalam Pengajaran Bahasa Indonesia Di Sekolah Dasar. UPI : Makalah.
- ROG-10 Rogers, Scott. 2010. *Level Up! The Guide to Great Video Game*. West Sussex: John Wiley & Sons, Ltd
- SIL-06 Silalahi, Ulber. 2006. *Metode Penelitian Sosial*. Bandung : Unpar Press.
- WAV-13 *Waveengine 1.2.0 sea otter*. *Wave Engine*. Perangkat Lunak tersedia di <http://Waveengine.net/Download/Index>



LAMPIRAN

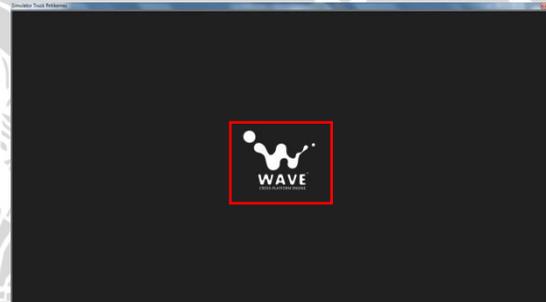
PETUNJUK PENGGUNA

GAME SIMULASI TRUK PETIKEMAS 3D

1. Eksekusi file containertruksimulator.exe di dalam folder *game*.

Name	Date modified	Type	Size
Content	12/31/2013 7:44 PM	File folder	
AForge.Video.DirectShow.dll	10/7/2013 11:16 AM	Application extens...	64 KB
AForge.Video.dll	10/7/2013 11:16 AM	Application extens...	23 KB
Bepu.dll	10/7/2013 5:20 PM	Application extens...	633 KB
Bepu.xml	10/7/2013 5:20 PM	XML Document	1,168 KB
containertruksimulator.exe	12/20/2013 4:04 AM	Application	718 KB
FarseerPhysics.dll	10/7/2013 5:20 PM	Application extens...	311 KB
FarseerPhysics.xml	10/7/2013 5:20 PM	XML Document	269 KB

2. Setelah tereksekusi *splash screen game engine Waveengine* akan muncul, dan tunggu beberapa saat.



3. Layar awal (*Home*) akan muncul setelah *splash screen game engine* selesai ditampilkan.
4. Tekan tombol “PLAY NOW” di tengah bawah layar awal.



5. Tombol “PLAY NOW” akan membawa pengguna pada layar pemilihan level.
6. Pada layar pemilihan level akan muncul level ikon petikemas dan urutan nomor, untuk level yang belum terselesaikan petikemas akan berwarna hijau dan memiliki skor nilai 0 dan level selanjutnya akan dikunci dan tidak bisa dipilih (petikemas abu-abu).

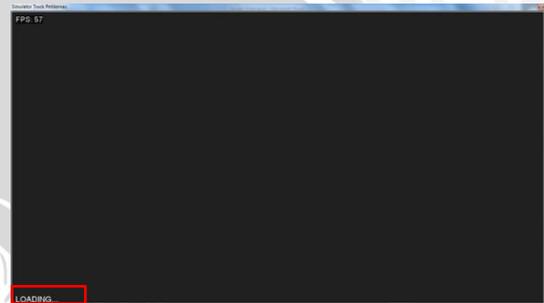


7. Pada layar level terdapat tombol tanda seru, bagi pemula tombol ini digunakan untuk menampilkan layar bagaimana cara untuk bermain.



8. Pada awal permainan pengguna hanya bisa bermain pada level 1 dan bila ingin bermain pada level lebih tinggi maka pengguna harus menyelesaikan level-level sebelumnya.

9. Setelah pengguna memilih level permainan maka akan muncul layar tunggu bertuliskan "Loading" di pojok kiri bawah layar.



10. Permainan langsung dimulai dengan waktu tempuh permainan yang telah ditentukan setelah layar tunggu selesai memuat aset yang ada pada layar utama permainan.



11. Pada level pemula antara 1-10 pengguna belajar bagaimana mengendarai truk hingga melakukan kegiatan bongkar-muat petikemas dan tujuan setelah terselasaikannya tugas yang diberikan truk harus berada pada aura berwarna biru untuk menyelesaikan permainan.

12. Tekan tombol "P/start/10" pada keyboard atau joystick untuk Pause maka akan muncul menu Pause di tengah layar utama



13. Setelah truk telah sampai pada aura berwarna biru maka skor akan terakumulasi mulai dari kerusakan mesin dan waktu tempuhnya.



14. Bila kembali ke layar pemilihan level maka skor dan tanda bintang yang didapatkan ketika bermain akan ditampilkan di layar pemilihan level dan membuka akses level selanjutnya.



15. Permainan akan berlanjut hingga level mencapai level tertinggi yaitu level 11. Pada level 11 pengguna akan bermain mengumpulkan skor dari setiap melakukan kegiatan bongkar-muat petikemas.

