

**RANCANG BANGUN SIMULASI ARUS LALU LINTAS
PADA RUAS JALAN SOEKARNO-HATTA MALANG
MENGUNAKAN UNITY 3D**

SKRIPSI

Oleh :
SYAIFUL ARIF
NIM. 07650058



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2014**

**RANCANG BANGUN SIMULASI ARUS LALU LINTAS
PADA RUAS JALAN SOEKARNO-HATTA MALANG
MENGUNAKAN UNITY 3D**

SKRIPSI

Diajukan kepada Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan Guna Memperoleh Gelar
Sarjana Teknik Informatika (S.Kom)

Oleh:

Syaiful Arif
NIM. 07650058

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2014**

LEMBAR PERSETUJUAN

**RANCANG BANGUN SIMULASI ARUS LALU LINTAS PADA
RUAS JALAN SOEKARNO-HATTA MALANG
MENGUNAKAN UNITY 3D**

SKRIPSI

Oleh :

Syaiful Arif

07650058

Telah Disetujui,

11 Juli 2014

Pembimbing I

Pembimbing II

Dr. M. Amin Hariyadi, M.T

A'la Syaugi, M.Kom

NIP. 19670118 200501 1 001

NIP. 19760613 200501 1 004

Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Sains dan Teknologi

Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang

Dr. Cahyo Crysdian

NIP. 19740424 200901 1 008

HALAMAN PENGESAHAN

RANCANG BANGUN SIMULASI ARUS LALU LINTAS PADA RUAS JALAN SOEKARNO-HATTA MALANG MENGUNAKAN UNITY 3D

SKRIPSI

Dipersiapkan dan disusun oleh

SYAIFUL ARIF

NIM.07650058

Telah Dipertahankan Di Depan Dewan Penguji Skripsi
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal, 14 Juli 2014

Susunan Dewan Penguji :

Tanda Tangan

- | | | |
|-----------------------|--|--------------------------|
| 1. Penguji Utama | : <u>Dr. Muhammad Faisal, M.T</u>
NIP. 19740510 200501 1 007 | () |
| 2. Ketua Penguji | : <u>Dr. Cahyo Crysdian</u>
NIP. 19740424 200901 1 008 | () |
| 3. Sekretaris Penguji | : <u>Dr. M. Amin Hariyadi, M.T</u>
NIP. 19670118 200501 1 001 | () |
| 4. Anggota Penguji | : <u>A'la Syauqi, M.Kom</u>
NIP. 19760613 200501 1 004 | () |

Mengetahui,

Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

SURAT PERNYATAAN ORISINALITAS PENELITIAN

Saya yang bertanda tangan di bawah ini :

Nama : Syaiful Arif

NIM : 07650058

Fakultas / Jurusan : Sains dan Teknologi / Teknik Informatika

Judul Penelitian : **RANCANG BANGUN SIMULASI ARUS LALU LINTAS
PADA RUAS JALAN SOEKARNO-HATTA
MENGUNAKAN UNITY 3D**

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

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

Malang, 18 juli 2014

Penulis

Syaiful Arif
NIM. 07650058

PERSEMBAHAN

Terima kasih kepada Ibuku, mbak Ayuk, mas Anang, mas Agus
dan keponakan kecilku, Bagas.

Terima kasih kepada semua teman-teman baikku yang tak pernah menyerah
memberi dukungan, bantuan maupun sentilan keras agar aku tetap mau
berusaha, terutama Iping, Muse, Jon, Hantu, Aji, Cueng, Ivan, Bebek, Pang2,
Kisun, Markisun, Pepi, Ponari adek Udin, dan Mandrak.
dan semua teman komunitas IOC dan WABIR

I love you All ☺

Terima kasih jua yang sebesar besarnya kepada semua pihak yang tidak bisa saya
sebut satu-satu yang telah
membantu dalam penyelesaian tugas akhir ini.

Speciall thanks : kopi hitam dan Ray Novand

”Sungguh tiada daya dan upaya kecuali dengan pertolongan Allah SWT.”

MOTTO

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

**”Jangan Pernah Mau Menyerah Dengan Keadaan
Dan Berjuanglah Melawan Dirimu Sendiri UntukNya”**

KATA PENGANTAR

Dengan menyebut nama Allah yang Maha Pengasih dan Maha Penyayang Puji syukur kita panjatkan kehadiran Allah SWT yang telah melimpahkan rahmat serta hidayah-Nya sehingga skripsi dengan judul "Rancang Bangun Simulasi Arus Lalu Lintas Pada Ruas Jalan Soekarno-Hatta Malang Menggunakan Unity 3D" ini dapat terselesaikan. Sholawat serta salam semoga tetap terlimpahkan kepada Nabi Muhammad SAW, keluarga beliau, para sahabat beliau dan orang-orang yang mengikuti ajaran beliau sampai akhir zaman nanti.

Terselesaikannya skripsi ini dengan baik berkat dukungan, motivasi, petunjuk dan bimbingan dari berbagai pihak. Oleh karena itu penulis mengucapkan terima kasih kepada :

1. Dr. M. Amin Hariyadi, M.T, selaku Dosen Pembimbing I dan Dosen Wali, A'la Syauqi, M.Kom selaku Dosem Pembimbing II.
2. Prof. Dr. H. Mudjia Rahardjo MSc, selaku Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
3. Dr. Bayyinatul Muchtaromah, drh. MSi, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
4. Dr. Cahyo Crysdian, selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
5. Semua Bapak dan Ibu Dosen Teknik Informatika UIN Malang yang telah mengajarkan dan memberikan banyak ilmu dengan tulus. Semoga Ilmu yang di berikan dapat bermanfaat di dunia dan akhirat.

Penulis sadar bahwa tidak ada sesuatu pun yang sempurna kecuali Allah SWT. Oleh karena itu, dengan senang hati penulis menerima kritik dan saran yang bersifat

membangun. Semoga skripsi ini bermanfaat bagi penulis khususnya dan juga bagi pembaca umumnya. Amin.

Malang, 16 Juli 2014
Penulis

Syaiful Arif



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN.....	iii
PERNYATAAN KEASLIAN TULISAN	iv
MOTTO	v
PERSEMBAHAN	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL	xii
ABSTRAK	xiii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian.....	3
1.4 Manfaat Penelitian.....	3
1.5 Batasan Masalah.....	3
1.6 Metode Penelitian	3
1.7 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA	7
2.1 Simulasi	7
2.1.1 Jenis-Jenis Simulasi	7
2.2 Unity 3D	11
2.2.1 Kelebihan Unity 3D	12
2.2.2 Kekurangan Unity 3D	13
2.3 Bahasa Pemrograman C#	13
2.3.1 Kelebihan Bahasa Pemrograman C#.....	14
2.4 Corel Draw	15
2.5 Use Case Diagram	19
2.5.1 Karakteristik Use Case Diagram.....	20
2.5.2 Komponen Pembentuk Use Case Diagram	21
2.6 Jalan Soekarno-Hatta Malang.....	23
2.7 Manajemen Lalu Lintas	24
2.7.1 Tujuan Manajemen Lalu Lintas	25
2.7.2 Sasaran Manajemen Lalu Lintas	26
2.7.3 Bentuk-Bentuk Tindakan Lalu Lintas.....	27
2.8 Kemacetan	28
2.8.1 Penyebab Kemacetan	29
2.8.2 Dampak Negatif Kemacetan	30
2.8.3 Pemecahan Permasalahan Kemacetan	30
BAB III ANALISA DAN PERANCANGAN.....	33
3.1 Analisa Sistem	33
3.1.1 Deskripsi Sistem	33

3.1.2 Analisis Kebutuhan Data	34
3.1.3 Analisis Kemampuan Sistem	36
3.2 Perancangan Sistem	37
3.2.1 Use Case Diagram	37
3.2.2 Flow Chart	39
3.2.3 Sketsa Peta Jalan Soekarno-Hatta Malang	42
3.2.4 Lingkungan Perancangan Sistem	43
3.3 Perancangan Uji Coba	44
3.3.1 Tujuan Pengujian	45
3.3.2 Skenario Pengujian	45
BAB IV IMPLEMENTASI HASIL DAN PEMBAHASAN	46
4.1 Ruang Lingkup Perangkat Keras	46
4.2 Ruang Lingkup Perangkat lunak	47
4.3 Implementasi Dan Antarmuka Simulasi	47
4.4 Pembahasan Sistem Simulasi	49
4.4.1 Unity 3D Scene Pada Aplikasi Simulasi	50
4.4.2 Game Object Pada Scene	50
4.4.3 Dasar logika Perpindahan Kendaraan	53
4.4.4 Kontrol Traffic Light	55
4.5 Pengujian Simulasi Dengan Black Box Testing	56
4.6 Integrasi Dengan Islam	57
BAB V PENUTUP	60
5.1 Kesimpulan	60
5.2 Saran	60
DAFTAR PUSTAKA	61

DAFTAR GAMBAR

Gambar 2.1 Arsitektur .NET Framework..... 14

Gambar 2.2 Jalan Soekarno-Hatta Malang..... 24

Gambar 3.1 Jalan Soekarno-Hatta Malang..... 34

Gambar 3.2 Gambar Kendaraan 35

Gambar 3.3 *Traffic Light* 36

Gambar 3.4 *Use Case diagram* Simulasi 38

Gambar 3.5 *Flowchart* Penambahan Kendaraan..... 39

Gambar 3.6 *Flowchart Traffic Light* 41

Gambar 3.7 Sketsa Jalan Soekarno-Hatta Malang 42

Gambar 4.1 Antar Muka Simulasi 47

Gambar 4.2 Form Atas Simulasi 48

Gambar 4.3 Perempatan Jalan 49

Gambar 4.4 Tampilan Unity 3D 50

Gambar 4.5 Konstruksi Objek 51

DAFTAR TABEL

Tabel 2.1 Perbedaan antara simulasi analog dan simulasi digital 9

Tabel 2.2 Pengujian aplikasi simulasi dengan Black Box..... 56



ABSTRAK

Arif, Syaiful. 2014. 07650058. **Rancang Bangun Simulasi Arus Lalu-Lintas pada Ruas Jalan Soekarno-Hatta Malang Menggunakan Unity 3D.** Pembimbing: (1) Dr. M. Amin Hariyadi, M.T. (2) A'la Syauqi, M.Kom.

Kata kunci : simulasi, arus lalu-lintas, Unity 3D, *traffic light*, jalan Soekarno-Hatta, kendaraan,

Simulasi adalah proses implementasi model menjadi program komputer atau rangkaian elektronik sehingga perilakunya menirukan sistem nyata untuk tujuan mempelajari perilaku sistem. Simulasi ini dibangun menggunakan Unity 3D dengan lokasi penelitian berada pada ruas jalan Soekarno-Hatta Malang. Komponen dari simulasi ini meliputi ruas jalan beserta persimpangan jalannya, *traffic light*, dan kendaraan roda empat yang melintas.

Simulasi ini menggunakan input banyaknya mobil dan pengaturan jeda waktu *traffic light*. Setelah itu input diproses melalui aplikasi simulasi sehingga dapat diketahui seberapa parah kemacetan yang terjadi pada ruas jalan Soekarno-Hatta Malang.

ABSTRACT

Arif, Syaiful. 2014. 07650058. **Simulation Design of Traffic Flow on Soekarno-Hatta Street Malang Using Unity 3D**. Advisors: (1) Dr. M. Amin Hariyadi, M.T. (2) A'la Syauqi, M.Kom.

Keywords : simulation, traffic jam, Unity 3D, *traffic light*, Soekarno-Hatta Street, vehicle,

Simulation is part of implementation from model to computer program or electronic series, that can be imitate real system to learn them. This simulation built with Unity 3D Engine and taking place in Soekarno-Hatta, the biggest yet the crowded street in Malang City. This simulation component consists of main street and some conjunctions, traffic lights, also with some vehicle passing by.

This simulation using the amount of vehicles and regulation of traffic lights timing as input. This input later processed by the simulator, and as a result, we got the information's about severe level of Soekarno-Hatta traffic jam.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pertumbuhan jumlah kendaraan di kota Malang semakin bertambah pesat setiap tahun seiring dengan meningkatnya perekonomian warga, namun pertumbuhan jumlah kendaraan ini tidak mampu di imbangi dengan penambahan ruas jalan karena keterbatasan lahan di kota Malang. Hal ini menyebabkan kemacetan pada lokasi tertentu disaat jam sibuk dan kemacetan semakin parah terjadi pada saat hari libur atau mudik lebaran.

Solusi penanggulangan kemacetan yang paling memungkinkan di kota Malang yaitu dengan melakukan manajemen lalu-lintas. Manajemen lalu lintas ini meliputi penempatan *traffic light* pada persimpangan yang berpotensi terjadi konflik pergerakan arus, perlakuan khusus pada ruas jalan tertentu yang meliputi pemberlakuan jalan satu arah dan pembatasan waktu ijin melintas ruas jalan.

Seiring dengan bertambah padatnya arus lalu-lintas maka diperlukan konfigurasi atau aturan baru pada ruas jalan yang berpotensi terjadi kemacetan. Tetapi dalam penerapannya, konfigurasi baru ini dilakukan secara langsung di ruas jalan yang bersangkutan sekaligus dilakukan pengamatan dalam kurun waktu tertentu. Hal ini menimbulkan masalah baru yaitu kemacetan yang semakin bertambah panjang dan protes dari warga. Untuk itu diperlukan sebuah aplikasi untuk mensimulasikan arus lalu-lintas yang bisa digunakan sebagai bahan pertimbangan dalam menerapkan konfigurasi lalu-lintas yang baru. Penerapan

simulasi arus lalu lintas ini mengambil tempat pada ruas jalan Soekarno-Hatta Malang.

Simulasi arus lalu-lintas ini menggunakan Unity 3D untuk membangun dan merancang . Unity 3D merupakan *game engine* yang digunakan untuk membuat game pada berbagai platform meliputi konsol, desktop dan mobile gaming.

Simulasi ini membahas secara umum tentang berkendara, adapun ayat Al quran yang membahas soal berkendara yaitu:

وَالَّذِي خَلَقَ الْأَزْوَاجَ كُلَّهَا وَجَعَلَ لَكُم مِّنَ الْفُلُكِ وَالْأَنْعَامِ مَا تَرْكَبُونَ ﴿١٢﴾ لِّتَسْتَوُوا عَلَىٰ ظُهُورِهِ ثُمَّ تَذْكُرُوا نِعْمَةَ رَبِّكُمْ إِذَا اسْتَوَيْتُمْ عَلَيْهِ وَتَقُولُوا سُبْحَنَ الَّذِي سَخَّرَ لَنَا هَذَا وَمَا كُنَّا لَهُ مُقْرِنِينَ ﴿١٣﴾

Artinya:

“ dan yang menciptakan semua yang berpasang-pasangan dan menjadikan untukmu kapal dan binatang ternak yang kamu tunggangi. supaya kamu duduk di atas punggungnya kemudian kamu ingat nikmat Tuhanmu apabila kamu telah duduk di atasnya; dan supaya kamu mengucapkan: "Maha suci Tuhan yang telah menundukkan semua ini bagi Kami Padahal Kami sebelumnya tidak mampu menguasainya,” (QS. Az-Zukhruf : 12-13)

1.2 Rumusan Masalah

Dari pemaparan latar belakang, maka perumusan masalah yang akan dikaji dalam penelitian ini adalah “bagaimana membuat aplikasi yang bisa mensimulasikan arus lalu lintas pada ruas jalan Soekarno-Hatta Malang menggunakan Unity 3D”

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah untuk membuat aplikasi yang bisa mensimulasikan arus lalu lintas pada ruas jalan Soekarno-Hatta Malang menggunakan Unity 3D.

1.4 Manfaat Penelitian

Manfaat yang dapat diambil dari penelitian ini adalah sebagai bahan pertimbangan sebelum menerapkan konfigurasi arus lalu lintas yang baru.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Aplikasi simulasi arus lalu-lintas ini berjalan pada desktop dengan sistem operasi dengan sistem operasi Windows.
2. Lokasi penelitian ini berada pada ruas jalan Soekarno-Hatta Malang.
3. Objek yang berjalan pada simulasi ini berupa kendaraan roda empat.

1.6 Metode Penelitian

Peneliti membagi pengerjaan penelitian ini menjadi beberapa tahap, antara lain:

1. Studi literatur

Pada tahap ini dilakukan pengumpulan berbagai informasi terkait beberapa hal berikut:

- a. Pengumpulan informasi mengenai simulasi arus lalu-lintas.
- b. Pengumpulan informasi mengenai lokasi penelitian.

2. Perancangan dan desain aplikasi

Perancangan simulasi ini difokuskan pada penggunaan Unity 3D untuk mengatur objek yang ada pada simulasi. Objek ini meliputi gambar sketsa jalan Soekarno-Hatta, *traffic light* dan kendaraan roda empat. Kendaraan diharapkan bisa mematuhi aturan *traffic light* dan melintas hanya pada ruas jalan saja, tanpa keluar dari lintasan.

3. Pembuatan aplikasi

Pembuatan design aplikasi dimuali dengan pembuatan sketsa peta jalan Soekarno-Hatta Malang dengan Coreldraw X3. Peta jalan Soekarno-Hatta mengacu pada Google Map yang diakses pada tanggal 12 juli 2014. Kemudian perancangan aplikasi simulasi menggunakan Unity 3D.

4. Uji coba dan evaluasi

Uji coba dan evaluasi dilakukan dengan mengamati pergerakan kendaraan yang melintas di dalam peta pada aplikasi dengan membandingkan skenario output yang seharusnya terjadi dengan output yang dihasilkan oleh aplikasi simulasi.

5. Penyusunan laporan

Penyusunan laporan akhir merupakan dokumentasi dari keseluruhan penelitian yang telah dilakukan. Diharapkan dokumentasi penelitian akan memiliki manfaat untuk penelitian atau pengembangan lebih lanjut.

1.7 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini berisi latar belakang mengenai alasan diadakannya penelitian, rumusan masalah dari penelitian, tujuan, manfaat, batasan masalah, serta sistematika penulisan yang bertujuan untuk memudahkan pembaca memahami pokok permasalahan yang dibahas.

BAB II TINJAUAN PUSTAKA

Dalam bab ini dijelaskan teori-teori atau argumentasi ilmiah yang dipakai sebagai referensi sesuai dengan permasalahan yang diambil, diantaranya tentang simulasi, Unity 3D, Bahasa Pemrograman C# , Corel Draw X3, Use Case Diagram, jalan Soekarno-Hatta Malang, kemacetan dan manajemen lalu lintas.

BAB III ANALISIS DAN PERANCANGAN APLIKASI

Bab ini menjelaskan tentang perancangan dan pembuatan simulasi arus lalu lintas, yang meliputi *analisa sistem*, *perancangan sistem* dan perancangan uji coba.

BAB IV HASIL DAN PEMBAHASAN

Hasil penelitian dari aplikasi yang telah dibuat dijelaskan secara rinci pada bab ini. Mulai dari pemaparan ruang lingkup sistem baik dari perangkat keras ataupun perangkat lunak yang dibutuhkan aplikasi, kemudian pemaparan

implementasi antar muka dan pembahasan mengenai simulasi arus lalu lintas yang dirancang pada aplikasi ini.

BAB V PENUTUP

Bab ini merupakan penutup, yang di dalamnya berisi kesimpulan dari hasil uji coba yang telah dilakukan pada bab sebelumnya, dan rangkuman dari pembahasan penelitian ini, serta berisi saran yang diharapkan dapat bermanfaat untuk pengembangan pembuatan program aplikasi selanjutnya.



BAB II

TINJAUAN PUSTAKA

2.1 Simulasi

Simulasi adalah proses implementasi model menjadi program komputer (*software*) atau rangkaian elektronik dan mengeksekusi *software* tersebut sedemikian rupa sehingga perilakunya menirukan atau menyerupai sistem nyata (*realitas*) tertentu untuk tujuan mempelajari perilaku (*behaviour*) sistem, pelatihan (*training*), atau permainan (*gaming*) yang melibatkan sistem nyata (*realitas*). (Sridadi. 2009)

Menurut Kamus Besar Bahasa Indonesia, pengertian simulasi adalah metode pelatihan yang meragakan sesuatu dalam bentuk tiruan yang mirip dengan keadaan sesungguhnya. Simulasi juga berarti penggambaran suatu sistem atau proses dengan peragaan berupa model statistik atau pemeranan. Kata menyimulasi berarti menirukan kepada sesuatu yang besar dengan ukuran yang lebih kecil. Kata menyimulasikan berarti membuat atau menjadikan dalam bentuk simulasi. (<http://kbbi.web.id/simulasi>, diakses tanggal 12 juli 2014)

2.1.1 Jenis-jenis simulasi

Menurut Bambang Sridadi dalam bukunya “Pemodelan dan Simulasi Sistem, 2009” jenis-jenis simulasi digolongkan menjadi tiga, berdasarkan perangkat keras yang digunakan, berdasarkan waktu simulasi dan berdasarkan teknik atau metodologi simulasi.

1. Berdasarkan perangkat keras yang digunakan

Berdasarkan perangkat keras yang digunakan, maka ada tiga jenis simulasi, yaitu: simulasi analog, simulasi digital, dan simulasi *hybrid*.

1. Simulasi analog adalah simulasi yang implementasinya menggunakan rangkaian elektronika analog, seperti *operational amplifier* untuk integrasi, pembanding, pembalik, penjumlah, dan lain-lain.
2. Simulasi digital adalah simulasi yang mana implementasinya menggunakan komputer digital.
3. Simulasi *hybrid* adalah simulasi yang mana implementasinya menggunakan gabungan rangkaian elektronika analog dan komputer digital.

Tabel 2.1 berikut memperlihatkan perbedaan antara simulasi analog dan simulasi digital

Tabel 2.1 perbedaan antara simulasi analog dan simulasi digital (Sridadi, 2009)

	Simulasi analog	Simulasi digital
1.	Menggunakan komputer analog.	Menggunakan komputer digital
2.	Membentuk atau menyusun analogi persoalan.	Menguraikan persoalan menjadi perhitungan.
3.	Menyajikan variable fisis dengan pengukuran	Menyajikan angka-angka dengan pola diskret terkode.
4.	Operasi besar dilakukan oleh piranti khusus (satu tugas) yang jumlahnya relatif sedikit.	Operasi dilakukan oleh piranti hitungan yang jumlahnya relative banyak dan dapat saling tukar tugas
5.	Biaya relatif rendah dan program mudah.	Biaya relatif tinggi dan program sulit.
6.	Unsur-unsur terpisah untuk setiap operasi.	Unsur-unsur identic bekerja beruntun (operasi seri).
7.	Ketelitian besar hingga 1 dalam 10000.	Ketelitian besar hingga 1 dalam 1000000000000 . tapi mempunyai persoalan “ <i>finite word length</i> ” dan <i>delay</i> komputasi.
8.	Simpanan data tersebar pada berbagai piranti yang tidak dapat dipertukarkan.	Simpanan data dipusatkan di tempat tertentu dan dapat dipertukarkan serta tak terbatas waktu.
9.	Sebagai model atau pencerminan sistem yang sebenarnya, operasi biasanya dijalani dalam waktu nyata sistem fisis.	Menghitung data himpunan yang tidak ada hubungannya dengan sistem yang diwakili. Waktu operasi biasanya tak bersangkutan dengan waktu nyata.
10.	Mewakili/menggantikan besaran-besaran matematis atau fisis.	Dapat mewakili angka-angka, maupun huruf-huruf atau simbol-simbol.
11.	Sangat sesuai untuk mewakili besaran-besaran terukur dan menirukan respons sistem-sistem fisis dengan analogi matematis.	Sangat sesuai untuk menangani proses-proses acak diskret, data statistik, dan masalah numeric dalam bidang ilmiah.

2. Berdasarkan waktu simulasi

Berdasarkan waktu simulasi, maka ada simulasi waktu nyata (*real-time*) dan simulasi *offline*.

1. Simulasi waktu nyata adalah simulasi dimana definisi waktu simulasi adalah sama dengan waktu nyata yang ditunjukkan pada jam (*clock*) pada umumnya. Artinya satu detik waktu simulasi adalah sama dengan waktu nyata. Data masukan dan hasil keluaran simulasi waktu nyata harus secara interaktif. Contoh simulasi waktu nyata adalah simulasi terbang (*flight simulation*) simulasi mobil, dan lain-lain.
2. Simulasi *offline* adalah simulasi dimana definisi waktu tidak sama dengan waktu nyata, bisa dipercepat bisa diperlambat. Data masukan dan hasil keluaran simulasi *offline* tidak perlu interaktif. Contoh simulasi *offline* adalah simulasi perang, simulasi antrian bank, simulasi ekonomi, dan lain-lain.

3. Berdasarkan teknik atau metodologi simulasi

Berdasarkan teknik atau metodologi simulasi, maka ada beberapa jenis simulasi Antara lain : simulasi *Monte Carlo*, simulasi kemudi-jejak (*trace driven*), simulasi kejadian diskret, simulasi dinamis kontinu.

1. Pada simulasi *Monte Carlo*, tidak ada elemen waktu, digunakan untuk mengevaluasi ekspresi non-probabilistik (misalnya integral) menggunakan metode probabilistic, melibatkan banyak persoalan matematika.

2. Simulasi kemudi-jejak secara ekstensif digunakan misalnya pada evaluasi untuk kerja sistem komputer, misalnya algoritma penghalaman (*paging*). Keuntungan dari simulasi ini terletak pada kredibilitas, validasi mudah, kurang keacakan, beban kerja akurat. Kekurangan dari simulasi ini terletak pada kompleksitas, hanya *snapshot*, dan validasi yang hanya satu titik.
3. Simulasi kejadian diskret adalah suatu simulasi yang menggunakan model sistem kejadian diskret (*dicrete event*), misalnya pada studi unjuk kerja sistem komputer digital, studi sistem antrian bank, dan lain-lain.
4. Simulasi dinamis kontinu menggunakan model keadaan perubahan kontinu terhadap waktu, misalnya pada studi proses reaksi kimia, gerakan dinamis suatu kendaraan baik darat, laut ataupun udara. (Sridadi. 2009)

2.2 Unity 3D

Unity 3D adalah sebuah game engine yang berbasis cross-platform. Unity dapat digunakan untuk membuat sebuah game yang bisa digunakan pada perangkat komputer, ponsel pintar android, iPhone, PS3, dan X-BOX.

Unity adalah sebuah tool yang terintegrasi untuk membuat game, arsitektur bangunan dan simulasi. Unity bisa digunakan untuk membuat games PC dan games Online. Untuk games Online diperlukan sebuah plugin, yaitu Unity Web Player, sama halnya dengan Flash Player pada Browser.

Unity tidak dirancang untuk proses desain atau modelling, dikarenakan unity bukan tool untuk mendesain. Penggunaan 3D editor lain seperti 3dsmax atau Blender akan banyak membantu untuk modelling 3D. Banyak hal yang bisa dilakukan dengan unity, diantaranya fitur audio reverb zone, particle effect, dan Sky Box untuk menambahkan langit.

Fitur scripting yang disediakan, mendukung 3 bahasa pemrograman, JavaScript, C#, dan Boo. Flexible and EasyMoving, rotating, dan scaling objects hanya perlu sebaris kode. Begitu juga dengan Duplicating, removing, dan changing properties. Visual Properties Variables yang di definisikan dengan scripts ditampilkan pada Editor. Bisa digeser, di drag and drop, bisa memilih warna dengan color picker. Unity 3D berbasis .NET. Artinya penjalanan program dilakukan dengan Open Source .NET platform, Mono. (<http://unity3dindo.forumid.net/t5-mengenal-lebih-dalam-tentang-unity3d>, diakses tanggal 15 juli 2014)

2.2.1 Kelebihan Unity 3D

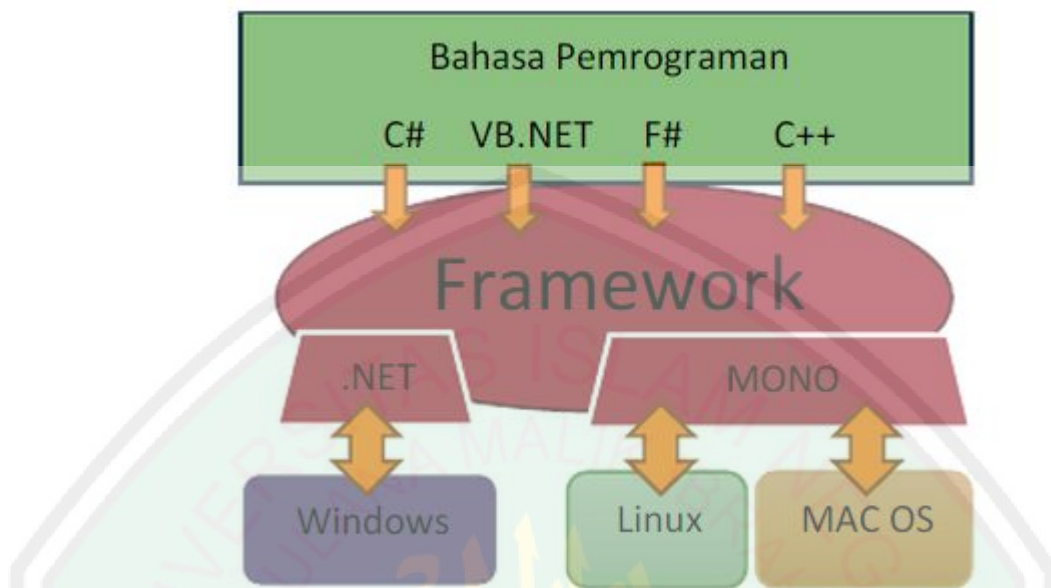
1. Mempunyai tools yang lengkap dalam pembuatan games.
 2. Bisa digunakan pada komputer, ponsel pintar android, iPhone, PS3, dan XBOX.
 3. Cocok untuk pemula atau yang sudah mahir.
 4. Flexible and *EasyMoving*, *rotating*, dan *scaling objects* hanya perlu sebaris kode Begitu juga dengan *Duplicating*, *removing*, dan *changing properties*.
 5. Mendukung 3 bahasa pemrograman, JavaScript, C#, dan Boo
- (<http://unity3dindo.forumid.net/t5-mengenal-lebih-dalam-tentang-unity3d>, diakses tanggal 15 juli 2014)

2.2.2 Kekurangan software Unity 3D

1. *game engine* ini mempunyai kapasitas memory yang sangat tinggi , bisa mencapai 500-600 Mb.
2. *game engine* ini tidak bisa modelling , karena unity 3D bukan *game engine* untuk mendesain (<http://unity3dindo.forumid.net/t5-mengenal-lebih-dalam-tentang-unity3d>, diakses tanggal 15 juli 2014)

2.3 Bahasa Pemrograman C#

Bahasa pemrograman C# adalah sebuah bahasa pemrograman berbasis objek yang di dukung oleh Microsoft .NET Framework. Microsoft .NET Framework adalah perantara agar aplikasi dengan bahasa pemrograman yang di dukung dapat berkomunikasi dengan sistem operasi yang digunakan oleh komputer. Selain itu, .NET Framework juga memungkinkan bahasa pemrograman C# untuk berkomunikasi dengan bahasa pemrograman lainnya yang juga di dukung oleh .NET Framework seperti VB .NET, F# atau C++. Dengan kata lain aplikasi yang kita buat dapat menggunakan komponen-komponen lain yang dibuat dengan menggunakan VB .NET, J# atau C++. (Darmawan. 2011)



Gambar 2.1 Arsitektur .NET Framework (Darmawan, 2011)

2.3.1 Kelebihan Bahasa Pemrograman C#

Ada beberapa hal yang menjadi kelebihan dari bahasa pemrograman C# dibandingkan dengan bahasa pemrograman lainnya, yaitu:

1. Termasuk bahasa pemrograman .NET. Dengan demikian kita dapat menggunakan komponen-komponen yang dibangun dengan bahasa pemrograman .NET lainnya (Integrasi antar bahasa).
2. Bahasa pemrograman C# memiliki Language Integrated Query (LINQ) yang merupakan sintaks query yang dapat digunakan pada setiap kumpulan data.
3. Windows Presentation Foundation (WPF) dapat digunakan untuk membuat tampilan aplikasi kita dengan sangat kreatif

4. Microsoft memberikan IDE (software yang digunakan untuk membangun sebuah program) secara gratis, yaitu Microsoft Visual Studio Express Edition. (Darmawan. 2011)

2.4 Corel Draw

Corel Draw adalah suatu piranti lunak komputer yang digunakan untuk menggambar, membuat logo, edit foto menjadi bitmap, edit foto menjadi gambar kartun, design kaos, dan semua yang berhubungan dengan grafis dan coret-coretan yang menghasilkan kreasi gambar yang menarik.

Kelebihan coreldraw terletak pada tool-toolnya yang lebih bersifat freehand, karenanya coreldraw pada dasarnya sebagai software grafis coreldraw bisa berdiri sendiri, Karena berbasis vektor dan bersifat freehand inilah, software ini bisa dijadikan pilihan utama untuk pembuatan sebuah logo. Hanya saja coreldraw juga memiliki kemampuan untuk dipadupadankan dengan editing foto, terlebih jika ingin menghasilkan suatu gambar grafis dari foto dengan penambahan atau editing tulisan dan karakter-karakter tertentu. Coreldraw juga mampu untuk mengimpor atau ekspor file menjadi file jpg, bmp atau lainnya.

Corel Draw X3 merupakan salah satu software yang cukup populer dikalangan pemakai komputer yang dekat dengan dunia grafis. Banyak aplikasi yang bisa dibuat dengan Corel Draw, mulai dari desain kaos, brosur, iklan, hingga membuat logo untuk perusahaan. Aplikasi yang berada dibawah naungan Corel Corporation ini merupakan penerus dari generasi-generasi sebelumnya. Terdapat beberapa fitur menarik dalam Corel Draw X3 yang tidak dimiliki oleh

pendahulunya, yaitu interface yang semakin modern dan dinamis, lebih flexible dalam memodifikasi sebuah gambar, serta banyaknya efek yang bisa digunakan untuk menggambar sebuah gambar. (Penerbit Andi. 2009)

Kekurangan dari Corel Draw yang sering muncul:

1. Menggunakan memori dan resource lain yang sangat besar apalagi bila gambar yang sedang dibuat mempunyai detail yang banyak. Pada PC yang low end penggunaan CDR sering menimbulkan pesan '*crash*' pada system bahkan dalam proses effect bevel/emboss dalam PC yang bagus pun dapat timbul '*hang*'.
2. Ukuran file yang dibuat membengkak.
3. Warna yang dicetak tidak akurat (tidak sesuai dengan tampilan layar) pada beberapa jenis printer.
4. Pembuatan objek table tidak semudah membuat table dalam MS Word, yaitu dengan cara yang sangat manual.
5. Hasil penggabungan objek vector dan photo/bitmap kualitas cetakannya kurang memuaskan, misalnya membuat cover buku yang terdapat objek text dan photo.
6. Kompatibilitas versi CorelDraw banyak kendala dalam sharing ke versi lainnya.

Jendela Aplikasi Corel Draw

Jendela aplikasi Corel Draw secara default terdiri atas :

1. Tittle bar: berada pada baris paling atas, memanjang ke kanan; berisi keterangan tentang file yang sedang aktif atau sedang kita buka.
2. Menu bar : berada di bawah tittle bar yang berisi pilihan-pilihan menu yang akan menjajar kebawah (drop-down) bila diklik. Terdiri atas menu File, Edit, View, Layout, Arrange, Effect, Bitmap, Text, Tools, Windows, dan Help.
3. Toolbar : berada pada baris ke tiga; berisi jalan pintas (*shortcuts*) yang menuju ke menu dan perintah-perintah lainnya. Secara default di sini akan terdapat shortcut New, Open, Copy, Paste, Undo, Redo, Import, Export, Application Launcher, Corel online dan Zoom Level.
4. Property bar : berisi perintah-perintah yang berhubungan dengan objek atau perangkat yang sedang aktif. Misalnya, jika perangkat Text sedang aktif maka text property bar akan menampilkan perintah-perintah yang dapat kita gunakan untuk menuliskan atau mengedit text.
5. Toolbox : berada pada sisi sebelah kiri, memanjang ke bawah; berisi perangkat-perangkat (*tools*) yang akan kita gunakan dalam membuat gambar, memberi warna, ataupun melakukan modifikasi-modifikasi terhadap objek dan gambar.

Semua bar ini dapat kita pindah-pindahkan letaknya ke tempat yang kita inginkan (*detachable*) atau bahkan bisa kita tidak tampilkan misalnya kita ingin area gambar menjadi lebih luas.

1. Ruler : merupakan batas-batas antara vertikal dan horizontal; berisi angka-angka untuk menunjukkan ukuran dan posisi objek.
2. Halaman (Drawing Page) :
Wilayah kosong (bila belum ada gambar) berbentuk segi empat, di tengah-tengah area putih, di mana gambar yang kita tempatkan di dalamnya dapat kita print.
3. Jendela gambar (Drawing window) : Merupakan area di luar drawing page, dibatasi oleh scroll bars dan application controls.
4. Document navigator : berada pada bagian bawah sebelah kiri; berisi kontrol untuk melakukan perpindahan antar halaman, dan juga untuk menambah atau mengurangi jumlah halaman.
5. Status bar : Berada pada bagian dasar jendela aplikasi; berisikan informasi-informasi mengenai objek seperti jenis, ukuran, warna, dan resolusi.
6. Navigator : Sesuai dengan namanya, navigator yang berada pada sudut kanan bawah berfungsi sebagai alat untuk membantu kita bergerak di sekitar Drawing window atau drawing page.
7. Color pallete : berupa bar di bagian sisi paling kanan yang berisi kotak-kotak warna (secara default berisi warna-warna CMYK) yang akan kita gunakan untuk memberi warna pada objek yang kita pilih.
8. Docker : docker merupakan suatu jendela tersendiri yang berisi perintah-perintah dan setting yang berhubungan dengan perangkat atau fungsi-fungsi tugas tertentu.

Command dan tool-tool Coreldraw X3

Berikut menu commands dan tools yang ada di dalam jendela aplikasi Corel Draw X3 yang terdapat pada masing-masing bagian seperti di jelaskan di atas:

1. Property bar : berisi fungsi-fungsi yang paling umum digunakan yang menampilkan perangkat atau kerja yang sedang kita lakukan. Isi yang ditampilkan akan berubah-ubah tergantung dari perangkat apa yang sedang kita gunakan atau pekerjaan apa yang sedang kita lakukan.
2. Toolbox :berisi perangkat-perangkat (tools) yang akan kita gunakan dalam membuat gambar, memberi warna, ataupun melakukan modifikasi-modifikasi terhadap objek dan gambar. Perangkat-perangkat tersebut adalah (Gambar C2) : Pick Tool, Shape, crop, zoom, curve, smart, rectangle, ellipse, object, perfect shape, text, interactive, efedropper, outline, fill, interactive fill. (Penerbit Andi. 2009)

2.5 Use Case Diagram

Use case diagram merupakan model diagram UML yang digunakan untuk menggambarkan requirement fungsional yang diharapkan dari sebuah sistem. Use case diagram menekankan pada “siapa” melakukan “apa” dalam lingkungan sistem perangkat lunak akan dibangun. Use case diagram terdiri dari dua bagian besar; yang pertama adalah use case diagram (termasuk gambar use case dependencies) dan use case description.

Use case diagram adalah gambaran graphical dari beberapa atau semua actor, use case, dan interaksi diantara komponen-komponen tersebut yang memperkenalkan suatu sistem yang akan dibangun. Use case diagram menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada di luar sistem. Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar.

Use-case diagram dapat digunakan selama proses analisis untuk menangkap requirement system dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, use case diagram berperan untuk menetapkan perilaku (*behavior*) sistem saat diimplementasikan. Dalam sebuah model mungkin terdapat satu atau beberapa use case diagram. Kebutuhan atau requirements system adalah fungsionalitas apa yang harus disediakan oleh sistem kemudian didokumentasikan pada model use-case yang menggambarkan fungsi sistem yang diharapkan (use-case), dan yang mengelilinginya (actor), serta hubungan antara actor dengan use-case (use-case diagram) itu sendiri. (https://www.academia.edu/5295802/Mengenal_Use_Case_Diagram diakses tanggal 15 juli 2014)

2.5.1 Karakteristik Use Case Diagram

Berikut merupakan pemaparan karakteristik dari use case diagram

1. Use case adalah interaksi atau dialog antara sistem dan actor, termasuk pertukaran pesan dan tindakan yang dilakukan oleh sistem.
 2. Use case diprakarsai oleh actor dan mungkin melibatkan peran actor lain.
- Use cases harus menyediakan nilai minimal kepada satu actor.

3. Use case bisa memiliki perluasan yang mendefinisikan tindakan khusus dalam interaksi atau use case lain mungkin disisipkan.
4. Use case class memiliki objek use case yang disebut skenario. Skenario menyatakan urutan pesan dan tindakan tunggal.

(https://www.academia.edu/5295802/Mengenal_Use_Case_Diagram
diakses tanggal 15 juli 2014)

2.5.2 Komponen Pembentuk Use Case Diagram

Komponen Pembentuk Use Case Diagram :

1. Actor

Pada dasarnya actor bukanlah bagian dari use case diagram, namun untuk dapat terciptanya suatu use case diagram diperlukan beberapa actor. Actor tersebut mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. Sebuah actor mungkin hanya memberikan informasi inputan pada sistem, hanya menerima informasi dari sistem atau keduanya menerima, dan memberi informasi pada sistem. Actor hanya berinteraksi dengan use case, tetapi tidak memiliki kontrol atas use case. Actor digambarkan dengan stick man . Actor dapat digambarkan secara umum atau spesifik, dimana untuk membedakannya kita dapat menggunakan relationship.

2. Use Case

Use case adalah gambaran fungsionalitas dari suatu sistem, sehingga customer atau pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.

Cara menentukan Use Case dalam suatu sistem:

1. Pola perilaku perangkat lunak aplikasi.
2. Deskripsi tugas dari sebuah actor.
3. Sistem atau “benda” yang memberikan sesuatu yang bernilai kepada actor.
4. Apa yang dikerjakan oleh suatu perangkat lunak (*bukan bagaimana cara mengerjakannya).

3. Relasi

Ada beberapa relasi yang terdapat pada use case diagram:

1. Association, menghubungkan link antar element.
2. Generalization, disebut juga inheritance (pewarisan), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
3. Dependency, sebuah element bergantung dalam beberapa cara ke element lainnya.
4. Aggregation, bentuk assosiation dimana sebuah elemen berisi elemen lainnya.

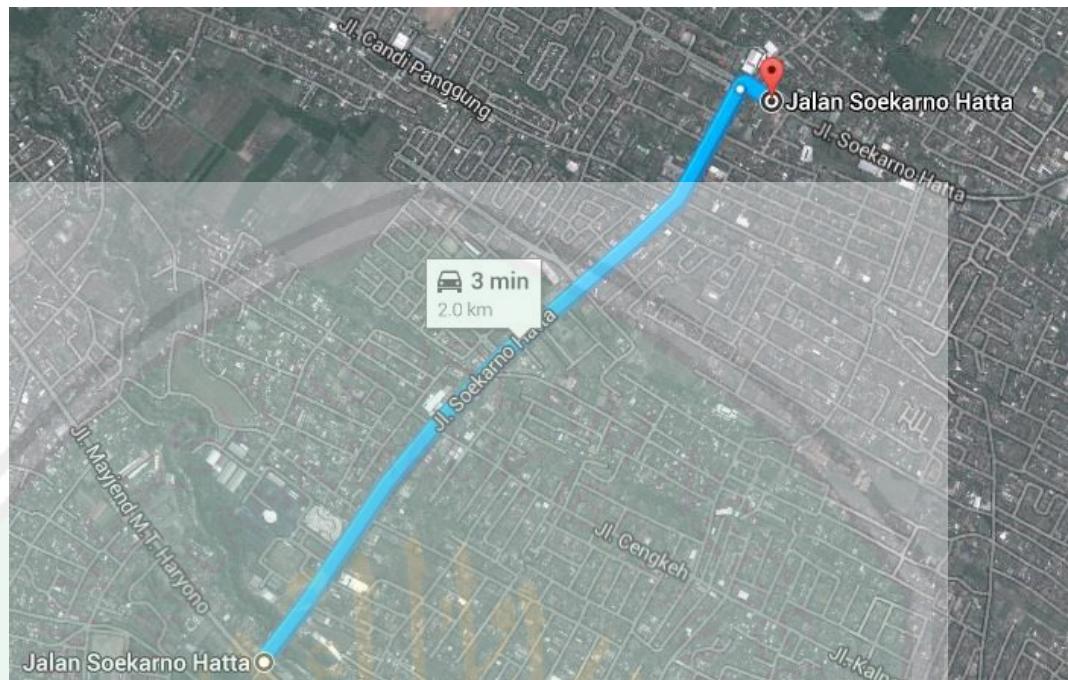
Tipe relasi/stereotype yang mungkin terjadi pada use case diagram:

1. include, yaitu kelakuan yang harus terpenuhi agar sebuah event dapat terjadi, dimana pada kondisi ini sebuah use case adalah bagian dari use case lainnya.

2. kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan roda.
3. communicates, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah communicates association . Ini merupakan pilihan selama asosiasi hanya tipe relationship yang dibolehkan antara actor dan use case. ([https://www.academia.edu/5295802/ Mengenal_Use_Case_Diagram](https://www.academia.edu/5295802/Mengenal_Use_Case_Diagram) diakses tanggal 15 juli 2014)

2.6 Jalan Soekarno-Hatta Malang

Jalan Soekarno-Hatta Malang merupakan salah satu jalan utama yang tergolong padat pengguna lalu-lintasnya. Jalan ini merupakan jalan penghubung antara daerah Blimbing dan Dinoyo. Jalan ini sering terjadi kemacetan karena terdapat balai kota, banyak pertokoan dan kampus. Kampus tersebut adalah STIMIK Asia, Politeknik Negeri Malang (kampus dua) dan Universitas Brawijaya.



Gambar 2.2 jalan Soekarno-Hatta Malang

(sumber dari google maps 2014)

2.7 Manajemen Lalu lintas

Manajemen lalu lintas adalah pengelolaan dan pengendalian arus lalu lintas dengan melakukan optimasi penggunaan prasarana yang ada melalui peredaman atau pengecilan tingkat pertumbuhan lalu lintas, memberikan kemudahan kepada angkutan yang efisien dalam penggunaan ruang jalan serta memperlancar sistem pergerakan.

Manajemen lalu lintas merupakan salah satu tugas yang diserahkan kepada untuk diselenggarakan oleh aparat Daerah Tingkat I untuk jalan propinsi dan aparat Daerah Tingkat II untuk jalan kabupaten/kotamadya berdasarkan Peraturan Pemerintah Nomor 22 Tahun 1990 tentang Penyerahan Sebagian Urusan

Pemerintah Dalam Bidang Lalu Lintas dan Angkutan Jalan Kepada Daerah Tingkat I Dan Daerah Tingkat II.

Manajemen lalu lintas merupakan salah satu strategi pengaturan lalu lintas yang memanfaatkan semaksimal mungkin prasarana dan sarana transportasi yang ada. Pembangunan jalan baru bukan merupakan bagian dari manajemen lalu lintas. Pembangunan yang termasuk di dalam manajemen lalu lintas hanya terbatas pada penyempurnaan fasilitas yang ada akibat diterapkannya suatu strategi dan instrumen (taktik) manajemen lalu lintas di lapangan.

Secara umum yang dimaksud dengan Manajemen Lalu Lintas adalah memanfaatkan semaksimal mungkin sistem jaringan jalan yang ada, atau:

1. Menampung lalu lintas sebanyak mungkin.
2. Menampung penumpang atau barang sebanyak mungkin.
3. Dengan memperhatikan keterbatasan lingkungan (kapasitas lingkungan)
4. Dengan memberikan prioritas untuk kelompok-kelompok yang sangat membutuhkan, melakukan penyesuaian kebutuhan terhadap pemakai jalan lainnya. (<http://www.ilmusipil.com/manajemen-lalu-lintas>, diakses tanggal 16 juli 2014)

2.7.1 Tujuan Manajemen Lalu Lintas

Tujuan dilakukannya manajemen lalu lintas adalah:

1. Mendapatkan tingkat efisiensi dari pergerakan lalu lintas secara menyeluruh dengan tingkat aksesibilitas yang tinggi dengan menyeimbangkan permintaan

2. dengan sarana penunjang yang tersedia.
3. Meningkatkan tingkat keselamatan dari pengguna yang dapat diterima oleh semua pihak dan memperbaiki tingkat keselamatan tersebut sebaik mungkin.
4. Melindungi dan memperbaiki keadaan kondisi lingkungan di mana arus lalu lintas tersebut berada.
5. Mempromosikan penggunaan energi secara efisien ataupun pengguna energi lain yang dampak negatifnya lebih kecil dari pada energi yang ada. (<http://www.ilmusipil.com/manajemen-lalu-lintas>, diakses tanggal 16 juli 2014)

2.7.2 Sasaran Manajemen Lalu Lintas

Sasaran dari manajemen lalu lintas adalah:

1. Mengatur dan menyederhanakan lalu lintas dengan melakukan pemisahan terhadap tipe, kecepatan dan pemakai jalan yang berbeda untuk meminimumkan gangguan terhadap lalu lintas.
2. Mengurangi tingkat kemacetan lalu lintas dengan menaikkan kapasitas atau mengurangi volume lalu lintas pada suatu jalan. Melakukan optimasi ruas jalan dengan menentukan fungsi dari jalan dan kontrol terhadap aktivitasaktivitas yang tidak cocok dengan fungsi jalan tersebut harus dikontrol. (<http://www.ilmusipil.com/manajemen-lalu-lintas>, diakses tanggal 16 juli 2014)

2.7.3 Bentuk-Bentuk Tindakan Lalu Lintas

Bentuk-bentuk tindakan dalam manajemen lalu lintas adalah sebagai berikut:

1. Tindakan untuk melancarkan lalu lintas kendaraan meliputi hal-hal berikut:
 - a. Peningkatan kapasitas pada persimpangan, koordinasi persimpangan dengan cara sistem satu arah (SSA), larangan belok kanan pada persimpangan, pengendalian belokan berputar (U turn), pengendalian jalan akses, pemasangan sinyal lampu lalu lintas di persimpangan, koordinasi sinyal-sinyal lampu lalu lintas.
 - b. Peningkatan kapasitas pada jaringan jalan
 - c. Peningkatan kapasitas pada jalan-jalan utama
2. Tindakan untuk meningkatkan pergerakan manusia meliputi hal-hal berikut:
 - a. Tindakan melakukan prioritas pada bus/angkutan umum dengan cara mengadakan lajur khusus bus (LKB) dapat searah arus maupun berlawanan arus.
 - b. Tindakan pada pejalan kaki dan sepeda dengan cara membuat jalur khusus untuk sepeda dan jalan khusus untuk para pejalan kaki.
3. Tindakan untuk mengendalikan permintaan:
 - a. Tindakan mengendalikan parkir

- b. Tindakan melakukan kekang lalu lintas secara fisik dengan cara membuat Kawasan Pembatasan Lalu Lintas maupun secara fiskal dengan cara mengadakan road pricing
- 4. Tindakan untuk meningkatkan keselamatan lalu lintas
 - a. Pembatasan kecepatan
 - b. Tindakan dengan pengarahan positif (positive guidance)
- 5. Tindakan untuk melindungi lingkungan:
 - a. Manajemen lingkungan lalu lintas
 - b. Tindakan untuk mengatur rute truk dan larangan truk
(<http://www.ilmusipil.com/manajemen-lalu-lintas>, diakses tanggal 16 juli 2014)

2.8 Kemacetan

Kemacetan adalah situasi atau keadaan tersendatnya atau bahkan terhentinya lalu lintas yang disebabkan oleh banyaknya jumlah kendaraan melebihi kapasitas jalan. Kemacetan banyak terjadi di kota-kota besar, terutamanya yang tidak mempunyai transportasi publik yang baik atau memadai ataupun juga tidak seimbangnya kebutuhan jalan dengan kepadatan penduduk, misalnya Jakarta.

Kemacetan lalu lintas menjadi permasalahan sehari-hari di Jakarta, Surabaya, Bandung, Medan, Semarang, Makassar, Palembang, Denpasar, Jogjakarta, dan kota-kota besar lainnya di Indonesia.
(<http://id.wikipedia.org/wiki/Kemacetan>, diakses tanggal 15 Juli 2014)

2.8.1 Penyebab Kemacetan

Berikut merupakan berbagai penyebab kemacetan:

1. Arus yang melewati jalan telah melampaui kapasitas jalan
2. Terjadi kecelakaan terjadi gangguan kelancaran karena masyarakat yang menonton kejadian kecelakaan atau karena kendaraan yang terlibat kecelakaan belum disingkirkan dari jalur lalu lintas,
3. Terjadi banjir sehingga kendaraan memperlambat kendaraan
4. Ada perbaikan jalan,
5. Bagian jalan tertentu yang longsor,
6. Kemacetan lalu lintas yang disebabkan kepanikan seperti kalau terjadi isyarat sirene tsunami.
7. Karena adanya pemakai jalan yang tidak tahu aturan lalu-lintas.
8. Parkir liar dari sebuah kegiatan.
9. Pasar tumpah yang secara tidak langsung memakan badan jalan yang menyebabkan terjadi antrian terhadap sejumlah kendaraan yang akan melewati area tersebut.
10. Pengaturan lampu lalu lintas yang bersifat kaku yang tidak mengikuti tinggi rendahnya arus lalu lintas
11. Adanya tawuran antarpelajar yang menyebabkan kurang lancarnya lalu lintas
12. Banyaknya penyeberang jalan.

(<http://id.wikipedia.org/wiki/Kemacetan>, diakses tanggal 15 Juli 2014)

2.8.2 Dampak Negatif Kemacetan

Kemacetan lalu lintas memberikan dampak negatif yang besar yang antara lain disebabkan:

1. Kerugian waktu, karena kecepatan perjalanan yang rendah
 2. Pemborosan energi.
 3. Keausan kendaraan lebih tinggi, karena waktu yang lebih lama untuk jarak yang pendek, radiator tidak berfungsi dengan baik dan penggunaan rem yang lebih tinggi,
 4. Meningkatkan polusi udara karena pada kecepatan rendah konsumsi energi lebih tinggi, dan mesin tidak beroperasi pada kondisi yang optimal,
 5. Meningkatkan stress pengguna jalan,
 6. Mengganggu kelancaran kendaraan darurat seperti ambulans, pemadam kebakaran dalam menjalankan tugasnya.
- (<http://id.wikipedia.org/wiki/Kemacetan>, diakses tanggal 15 Juli 2014)

2.8.3 Pemecahan permasalahan kemacetan

Beberapa langkah yang bisa dilakukan untuk memecahkan permasalahan kemacetan lalu lintas meliputi langkah-langkah sebagai berikut:

1. Peningkatan kapasitas

Salah satu langkah yang penting dalam memecahkan kemacetan adalah dengan meningkatkan kapasitas jalan atau prasarana seperti:

1. Memperlebar jalan, menambah lajur lalu lintas sepanjang hal itu memungkinkan,
2. Mengubah sirkulasi lalu lintas menjadi jalan satu arah,
3. Mengurangi konflik dipersimpangan melalui pembatasan arus tertentu, biasanya yang paling dominan membatasi arus belok kanan.
4. Meningkatkan kapasitas persimpangan melalui lampu lalu lintas, persimpangan tidak sebidang/flyover,
5. Mengembangkan intelligent transport sistem.
6. Memberikan Sanksi Jika Ada Yang Melanggar

2. Keberpihakan kepada angkutan umum

Untuk meningkatkan daya dukung jaringan jalan dengan adalah mengoptimalkan kepada angkutan yang efisien dalam penggunaan ruang jalan antara lain:

1. Pengembangan jaringan pelayanan angkutan umum
2. Pengembangan lajur atau jalur khusus bus ataupun jalan khusus bus yang di Jakarta dikenal sebagai Busway,
3. Pengembangan kereta api kota, yang dikenal sebagai metro di Perancis, Subway di Amerika, MRT di Singapura
4. Subsidi langsung seperti yang diterapkan pada angkutan kota di Transjakarta, Batam ataupun Jogjakarta maupun tidak langsung melalui keringanan pajak kendaraan bermotor, bea masuk kepada angkutan umum.

3. Pembatasan kendaraan pribadi

Langkah ini dilakukan jika kemacetan semakin parah harus dilakukan manajemen lalu lintas yang lebih ekstrem sebagai berikut:

1. Pembatasan penggunaan kendaraan pribadi menuju suatu kawasan tertentu seperti yang direncanakan akan diterapkan di Jakarta melalui Electronic Road Pricing (ERP). ERP berhasil dengan sangat sukses di Singapura, London, Stockholm. Bentuk lain dengan penerapan kebijakan parkir yang dapat dilakukan dengan penerapan tarif parkir yang tinggi di kawasan yang akan dibatasi lalu lintasnya, ataupun pembatasan penyediaan ruang parkir di kawasan yang akan dibatasi lalu lintasnya,
 2. Pembatasan kepemilikan kendaraan pribadi melalui peningkatan biaya kepemilikan kendaraan, pajak bahan bakar, pajak kendaraan bermotor, bea masuk yang tinggi.
 3. Pembatasan lalu lintas tertentu yang memasuki kawasan atau jalan tertentu, seperti diterapkan di Jakarta yang dikenal sebagai kawasan 3 in 1 atau contoh lain pembatasan sepeda motor masuk jalan tol, pembatasan mobil pribadi masuk jalur busway.
- (<http://id.wikipedia.org/wiki/Kemacetan>, diakses tanggal 15 Juli 2014)

BAB III

ANALISA DAN PERANCANGAN

3.1 Analisa Sistem

Analisis sistem bertujuan untuk mengidentifikasi masalah yang ada pada sistem yang meliputi perangkat lunak (*software*), pengguna (*user*) serta hasil analisis terhadap sistem dan elemen-elemen yang terkait dalam pembuatan aplikasi simulasi arus lalu-lintas ini. Analisis sistem ini diperlukan sebagai awal pembangunan aplikasi dan sebagai dasar bagi tahapan perancangan sistem.

Analisis sistem meliputi deskripsi sistem, analisis kebutuhan data, analisis kemampuan sistem, dan lingkungan operasi perancangan sistem.

3.1.1 Deskripsi Sistem

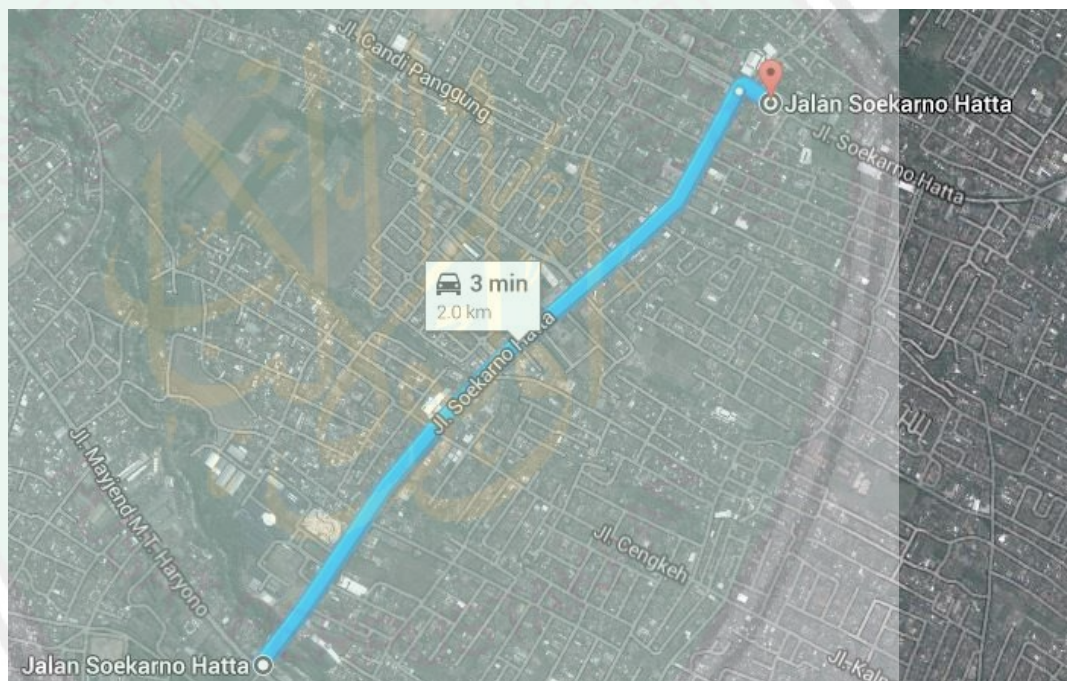
Pembuatan aplikasi simulasi arus lalu lintas ini bertujuan untuk membuat aplikasi yang bisa mensimulasikan arus lalu lintas di jalan Soekarno-Hatta Malang. Aplikasi simulasi ini menggunakan sudut pandang dari atas agar mendapatkan pandangan secara menyeluruh dari setiap bagian jalan. Panjang ruas jalan yang di pakai dalam aplikasi ini adalah 2 Km dengan hanya mengikutsertakan satu perempatan yang mempunyai *traffic light*. Input dari aplikasi simulasi ini adalah banyaknya jumlah kendaraan beserta lokasinya dan jeda waktu *traffic light*.

3.1.2 Analisis Kebutuhan Data

Dalam pembuatan aplikasi simulasi arus lalu lintas ini data yang dibutuhkan adalah :

- a. Peta jalan Soekarno-Hatta Malang.

Peta jalan Soekarno ini dibutuhkan sebagai acuan dalam membuat sketsa peta, sketsa peta kemudian yang akan dipergunakan sebagai background aplikasi simulasi.



Gambar 3.1 jalan Soekarno-Hatta Malang

(sumber google map 2014)

Panjang panjang ruas jalan yang akan dipakai dalam aplikasi simulasi adalah 2 Km terhitung mulai dari persimpangan pertigaan antara jalan Soekarno-Hatta dan jalan Mayjen M. T. Haryono.

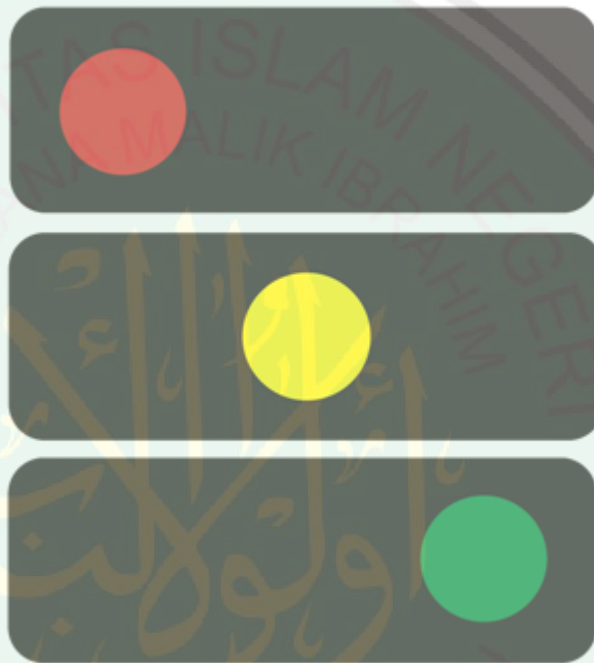
- b. Pembuatan objek sistem yaitu kendaraan dan *Traffic light*.

Kendaraan yang melintas pada aplikasi ini di bedakan menjadi 5 jenis untuk menggambarkan keberagaman kendaraan. Gambar Kendaraan ini mengambil sudut pandang dari atas.



Gambar 3.2 gambar kendaraan
(dibuat dengan Corel Draw)

Object *traffic light* yang akan di gunakan dalam aplikasi ini di gambarkan menjadi 3 bagian, bagian pertama dengan lampu merah berada di sisi kiri, bagian kedua dengan lampu kuning di tengah dan bagian terakhir dengan lampu hijau di kanan.



Gambar 3.3 *traffic light*
(dibuat dengan Corel Draw)

3.1.3 Analisis Kemampuan Sistem

Aplikasi yang di bangun ini memiliki kemampuan mensimulasikan arus lalu lintas yang terjadi di jalan Soekarno-Hatta yang bertujuan untuk mengetahui tingkat kemacetan yang terjadi dan titik titik dimana terjadi potensi kemacetan arus lalu lintas.

3.2 Perancangan Sistem

Perancangan perangkat lunak sesungguhnya salah satu pekerjaan yang paling banyak menuntut keahlian dan keterampilan manusia, yaitu dalam hal analisis dan perancangan, kemampuan teknis pemrograman, serta kemampuan manajerial (pengelolaan). Segala sesuatunya pada umumnya berawal dari upaya pemahaman masalah secara tuntas (*analysis*), perancangan perangkat lunak yang akan dikembangkan (*design*), baru dilanjutkan dengan implemetasi (penulisan hasil perancangan dengan bahasa pemrograman tertentu), dan diakhiri dengan pengujian (*testing*) agar perangkat lunak benar-benar sesuai dengan kebutuhan dan harapan pengguna.

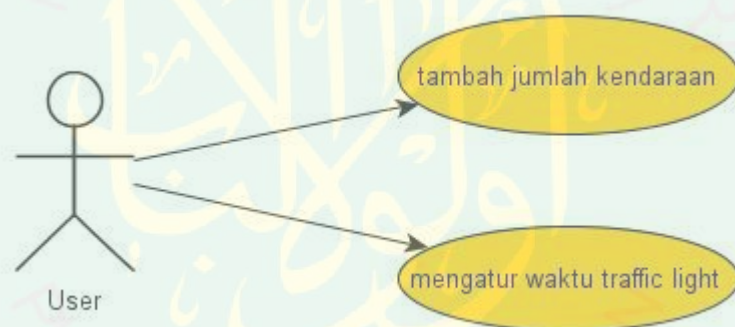
Perancangan sistem akan dibahas seluruh tahapan perancangan sistem menggunakan *Use Case Diagram* dan *Flowchart* sebagai desain prosedural penelitian ini.

3.2.1 Use Case Diagram

Dalam konteks *UML* (*Unified Modeling Language*) tahap konseptualisasi dilakukan dengan pembuatan *use case diagram* yang sesungguhnya merupakan deskripsi peringkat tinggi bagaimana perangkat lunak akan digunakan oleh penggunanya. Selanjutnya, *use case diagram* tidak hanya sangat penting pada tahap analisis, tetapi juga sangat penting untuk perancangan (*design*), untuk mencari kelas-kelas yang terlibat dalam aplikasi, dan untuk meakukan pengujian

(testing). Ketika akan mengembangkan *use case diagram*, hal yang pertama kali harus dilakukan adalah mengenali aktor untuk sistem aplikasi yang akan dikembangkan. Dalam hal ini ada beberapa karakteristik untuk aktor, yaitu (1) aktor yang berada di luar sistem dan (2) aktor yang berinteraksi langsung dengan sistem.

Pada penelitian ini hanya terdapat satu aktor, dimana aktor tersebut adalah seorang user yang berinteraksi langsung dengan sistem aplikasi. Adapun bentuk *use case diagram* seperti gambar 3.4.



Gambar 3.4 *use case diagram* sistem simulasi

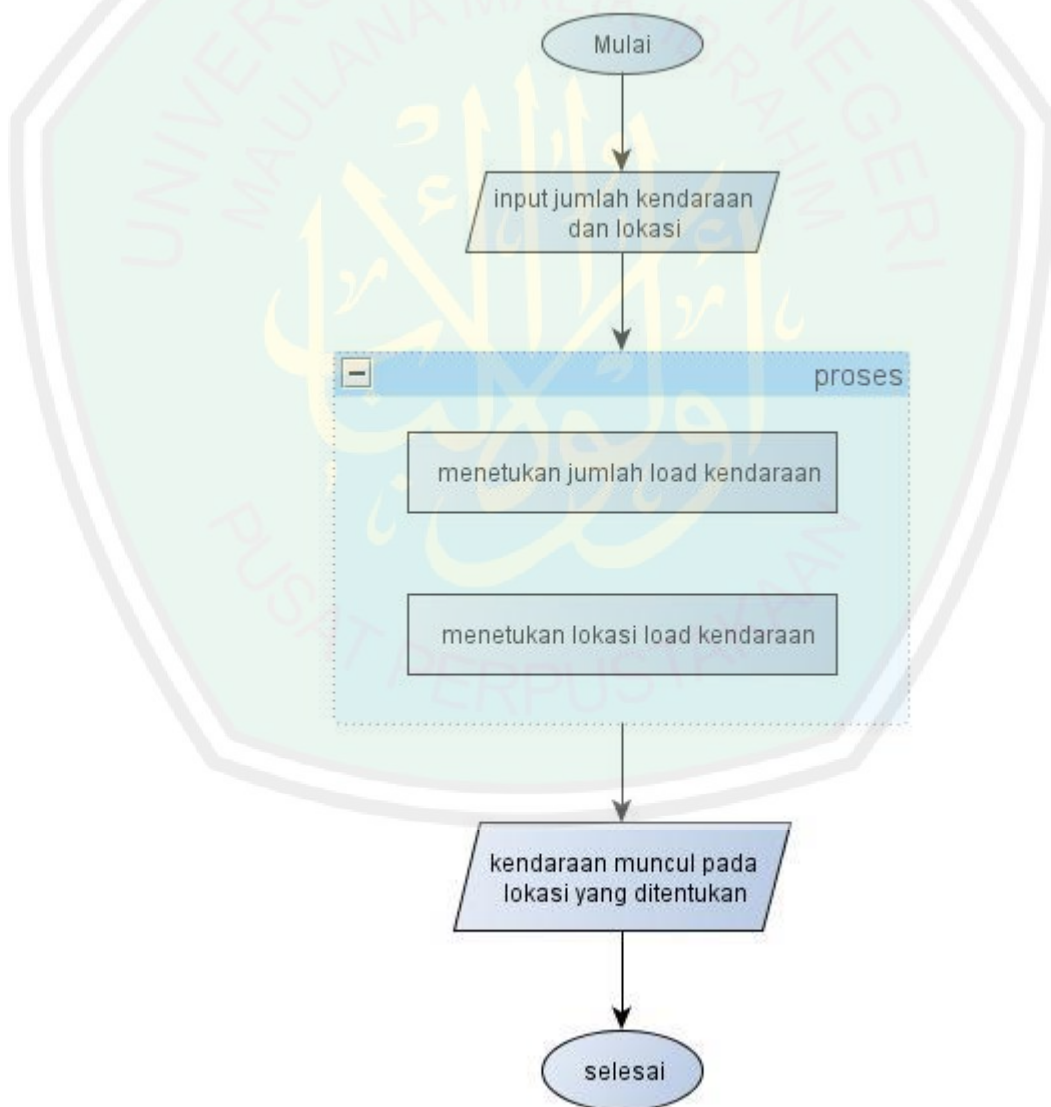
Dalam hal ini terdapat dua perilaku (behaviour) user seperti berikut:

1. User dapat menambah jumlah kendaraan beserta lokasi munculnya kendaraan dalam aplikasi simulasi.
2. User dapat mengatur waktu jeda *traffic light* pada tempat yang dikehendaki.

3.2.2 Flow chart

Flowchart adalah diagram yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma

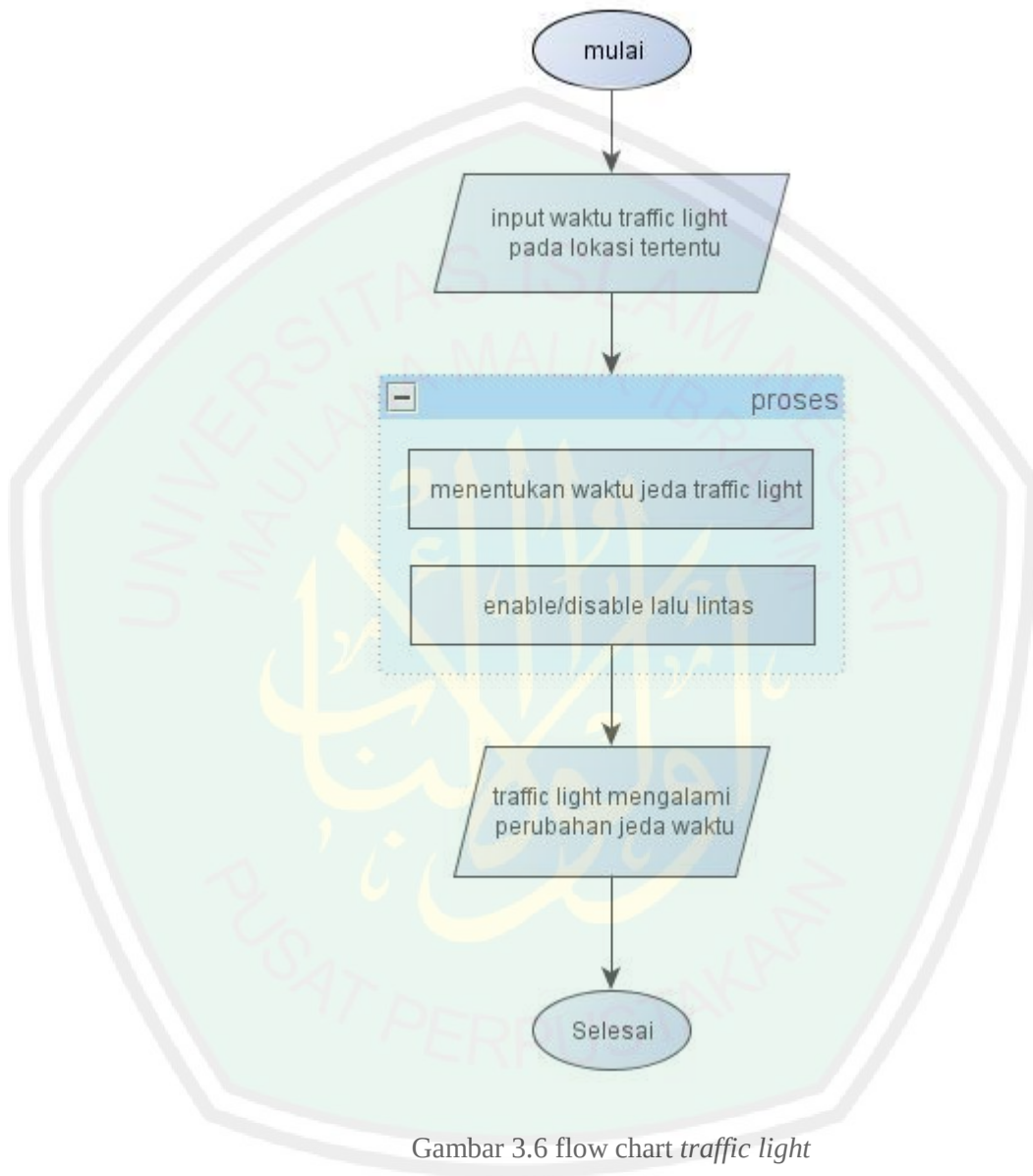
Flowchart diagram penambahan jumlah kendaraan di dalam aplikasi simulasi.



Gambar 3.5 flow chart penambahan kendaraan

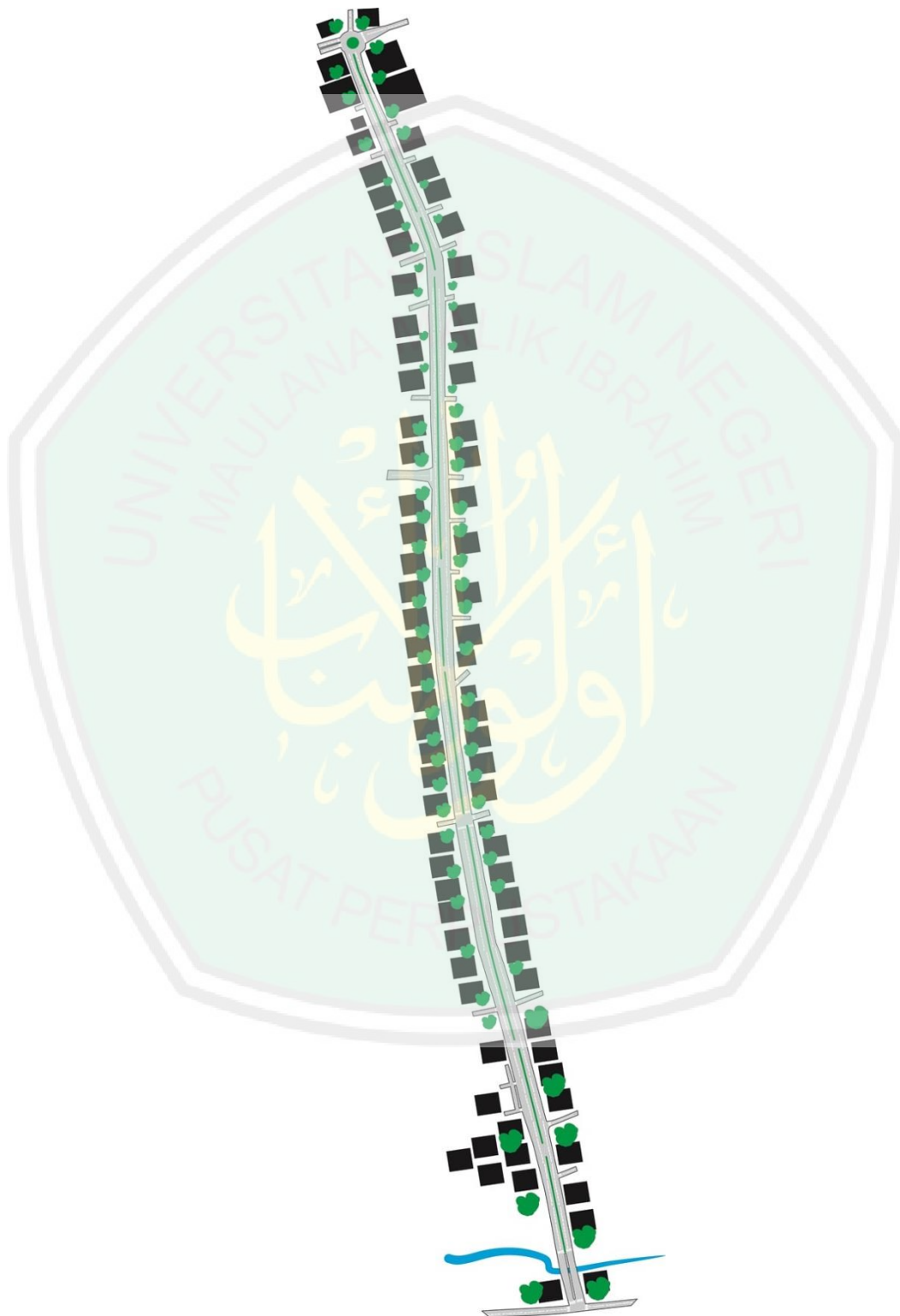
Flow chart pada gambar 3.5 menjelaskan alur penentuan load kendaraan kendaraan pada lokasi yang telah ditentukan. Proses penentuan load kendaraan ini melalui inputan form pada aplikasi simulasi. Inputan form ini terdiri dari dua opsi field yaitu inputan jumlah load kendaraan dalam bentuk text field dan lokasi tertentu dalam bentuk radio button. User menginputkan kedua opsi tersebut untuk mensimulasikan penambahan load kendaraan, kemudian terjadi proses di aplikasi yaitu penentuan jumlah load kendaraan beserta lokasi munculnya. Output dari proses ini yaitu kendaraan muncul pada lokasi yang telah ditentukan.



Flow chart diagram pengaturan *traffic light*

Flow chart pada gambar 3.6 menjelaskan tentang pengaturan jeda waktu *traffic light*. Proses pengaturan jeda waktu ini melalui form waktu jeda (detik) bertipe text field dan enable/disable arus lalu lintas bertipe check box. User menginputkan kedua opsi tersebut untuk mensimulasikan pengaturan traffic light, kemudian terjadi proses penentuan jeda traffic light dan enable/disable arus lalu lintas berdasarkan traffic light.

3.2.3 Sketsa Peta Jalan Soekarno-Hatta Malang



Gambar 3.7 sketsa jalan Soekarno-Hatta Malang

Hasil sketsa jalan Soekarno-Hatta pada gambar 3.7 merupakan adaptasi dari peta jalan yang dilihat dari Google maps (<https://www.google.com/maps/dir/-7.9497593,112.6154847/-7.9372162,112.6269517/@-7.9419535,112.6235952,1572m/data=!3m1!1e3!4m4!4m3!1m0!1m0!3e0>, diakses pada tanggal 16 juli 2014)

Dari gambar 3.7 terdapat beberapa bagian yang perlu diperhatikan yaitu:

1. Jalan Soekarno-Hatta dibagi menjadi dua jalur yang berlawanan arah. Masing-masing jalur dibagi menjadi dua *lane* .
2. Terdapat satu perempatan yang menggunakan *traffic light* yaitu di persimpangan jalan coklat
3. Panjang jalan yang digunakan adalah 2 Km

3.2.4 Lingkungan perancangan sistem

Untuk membangun aplikasi simulasi arus lalu lintas ini membutuhkan sistem pendukung sesuai dengan spesifikasi kebutuhan yaitu :

- a. Sistem operasi windows

Sistem operasi ini dipilih karena sudah banyak di kenal dan kompatibilitasnya baik sehingga mudah untuk di gunakan.

- b. Unity 3D

Game Engine menggunakan sistem navigasi bebas dalam pembuatan game, sehingga pengguna dapat dengan mudah untuk melihat setiap sisi 3D dalam pembuatan objek.

c. Corel Draw

Sebuah program aplikasi yang populer digunakan dalam desain grafis.

Versi software Corel Draw yang bisa digunakan tergantung dari kenyamanan user.

3.3 Perancangan Uji Coba

Teknik pengujian aplikasi simulasi ini menggunakan 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. Proses umum yang terjadi pada black box testing yaitu:

1. Kebutuhan atau spesifikasi dianalisa terlebih dahulu.
2. 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.
3. Penentuan output yang diharapkan sesuai dengan input yang telah dipilih
4. Pengujian dibuat dengan input yang telah dipilih
5. Pengujian dijalankan
6. Output yang sebenarnya dibandingkan dengan output yang diharapkan
7. Penentuan dibuat menyangkut perangkat lunak berfungsi sesuai dengan spesifikasi yang telah ditentukan

3.3.1 Tujuan Pengujian

Tujuan pengujian adalah menguji validitas fungsional semua yang di simulasikan pada aplikasi ini.

3.3.2 Skenario Pengujian

Skenario pengujian dengan membuat test case kemudian membandingkan output yang seharusnya terjadi dengan output yang dihasilkan oleh aplikasi simulasi.



BAB IV

IMPLEMENTASI HASIL DAN PEMBAHASAN

Implementasi merupakan proses pembangunan komponen-komponen pokok sebuah sistem berdasarkan desain yang sudah dibuat. Implementasi sistem juga merupakan sebuah proses pembuatan dan penerapan sistem secara utuh baik dari sisi perangkat keras maupun perangkat lunaknya. Pada bab berikut akan dipaparkan implementasi ruang lingkup sistem baik dari perangkat keras maupun perangkat lunak yang dibutuhkan aplikasi, kemudian pemaparan implementasi antar muka aplikasi dan terakhir juga pembahasan mengenai simulasi arus lalu lintas yang dirancang pada penelitian ini.

4.1 Ruang Lingkup Perangkat Keras

Proses pengembangan dan pengujian aplikasi simulasi ini, menggunakan komputer dengan spesifikasi perangkat keras sebagai berikut:

1. CPU Processor Intel® Core™ i5-520M 2.40GHz
2. Memory Storage 500 GB
3. RAM memory 4 GB
4. Graphic Card VGA ATI Mobility Radeon HD 5650
5. Memory VGA 2733 MB

4.2 Ruang Lingkup Perangkat Lunak

Proses pengembangan dan pengujian aplikasi simulasi ini, menggunakan komputer dengan beberapa tool dan software sebagai berikut:

1. Sistem Operasi Windows 7
2. Unity 3D versi 8
3. Corel Draw X3

4.3 Implementasi dan Antar Muka Simulasi

Aplikasi simulasi ini hanya terdiri dari satu scene. Sebuah scene tunggal yang terdiri dari banyak game object, seperti taman, pembatas jalan, kendaraan (mobil), lampu lalu lintas, dan lain sebagainya. Implementasi antar muka pada bagian ini hanya berfokus pada sebuah scene tunggal tersebut. Berikut adalah beberapa gambar scene simulasi ini.



Gambar 4.1 Antar muka simulasi

Gambar tersebut adalah tampilan awal begitu aplikasi simulasi ini dibuka. Awal kamera aplikasi simulasi berada pada sebelah selatan jalan Soekarno-Hatta. Beberapa game objek kendaraan tampak pada gambar tersebut. Pada bagian atas simulasi, terdapat dua buah form editing yaitu form untuk menambahkan kendaraan pada beberapa poin jalan dan form edit untuk objek traffic light.



Gambar 4.2 Form Atas

Terdapat dua bagian form yang tampak pada gambar 4.2 ini, yaitu form menambahkan kendaraan dan form editing traffic light. Pada form pertama, terdapat dua field yaitu field jumlah kendaraan yang akan ditambahkan dan opsi multi radio button untuk jalan mana kendaraan akan ditambahkan. Pada form kedua, tampak dua traffic light yang bisa diatur, traffic light sebelah jembatan dan traffic light perempatan jalan. Pada setiap traffic light terdapat dua field yang dapat diubah, yaitu waktu atau delay traffic light dan juga menonaktifkan traffic light.



Gambar 4.3 Perempatan Jalan

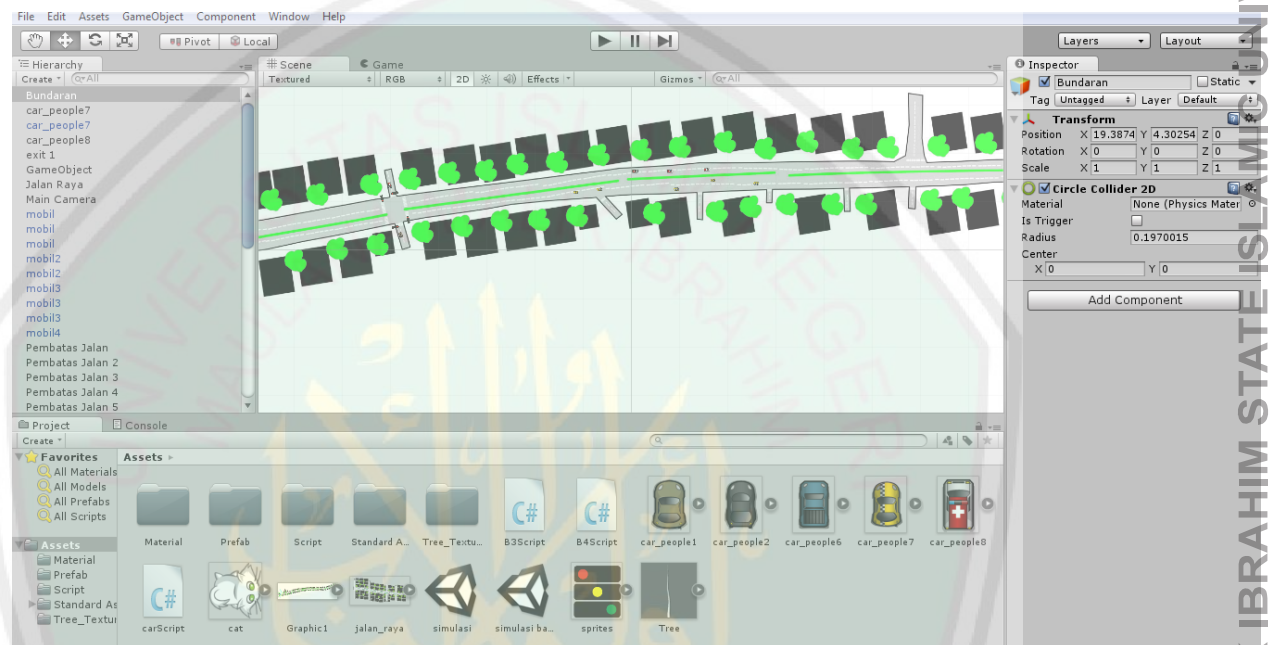
Gambar 4.3 merupakan simulasi perempatan jalan Soekarno-Hatta. Tampak pada gambar empat buah traffic light pada masing-masing sudut perempatan.

4.4 Pembahasan Sistem Simulasi

Pada bagian ini akan dijelaskan implementasi dari aplikasi simulasi yang dibuat. Penjelasan akan terdiri dari beberapa sub-bab yang terkait dari hal-hal teknis terutama pada pengembangan aplikasi simulasi engine unity3D

4.4.1 Unity3D scene pada aplikasi simulasi

Aplikasi simulasi ini hanya memiliki satu scene yaitu scene seperti tampak gambar antar muka bab ini.

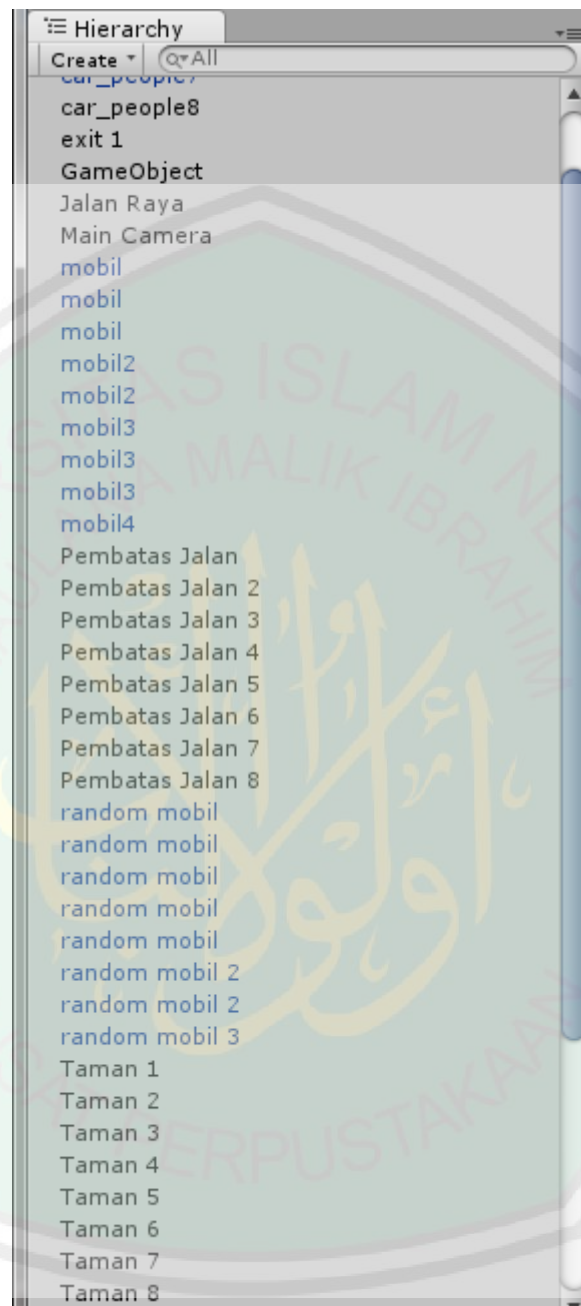


Gambar 4.4 tampilan Unity 3D

Tampak pada gambar adalah sebuah scene simulasi jalan dalam unity 3D. Pada menu samping kiri adalah semua object yang terlibat dalam scene ini dan bagian tengah merupakan layar kerja dari scene ini.

4.4.2 Game object pada scene

Terdapat banyak game object dari scene simulasi ini. Daftar berikut adalah daftar game object yang ada pada scene simulasi.



Gambar 4.5 konstruksi objek

Terdapat juga game object yang dibuat pada saat run-time simulasi. Object ini adalah beberapa kendaraan mobil yang di instantiate pada saat scene selesai terload. Script yang bertanggung jawab untuk instantiate kendaraan ini adalah sebagai berikut.

```
IEnumerator SpawnItems() {
Quaternion newRotation = new Quaternion(
    transform.rotation.x,
    transform.rotation.y,
    transform.rotation.z,
    transform.rotation.w);;
newRotation *= Quaternion.Euler(0, 0, -90);

    while (numberOfCar > 0) {
        Instantiate (prefMobil, new Vector3 (-20, -2.5f, 0),
            newRotation);
        --numberOfCar;
        //Waits for 2 Sec
        yield return new WaitForSeconds(3);
    }
}
```

Fungsi untuk membuat object mobil baru terletak pada fungsi Instantiate(). Parameter pertama adalah bakal resource object, parameter kedua posisi object muncul pada lokasi scene yang diisi dalam bentuk Vector, dan ketiga adalah rotasi object dari keadaan normalnya. Pada script tersebut, object yang akan dibuat adalah mobil dengan prototipe prefMobil dan akan muncul pada lokasi dengan x, y, z adalah -20, -2.5f, 0 serta object akan dirotasi dengan nilai 90 searah jarum jam. Script ini kemudian diletakkan pada fungsi awake di object main camera aplikasi simulasi.

```
void Awake() {
    StartCoroutine(SpawnItems());
}
```

Fungsi StartCoroutine() akan mengeksekusi SpawnItems() dengan membuat prosesnya secara paralel. Kemudian terdapat juga object main camera. Object ini akan selalu ada pada setiap scene engine unity3D. Karena simulasi ini adalah aplikasi dua dimensi, maka mode untuk main camera adalah orthogonal. Terdapat juga script main camera untuk mengikuti kontrol mouse saat di klik dan di drag.

```

if (Input.GetMouseButtonDown(0)) {
    inputPos = Input.mousePosition;
}else if (Input.GetMouseButton(0)) {
    if (inputPos != Input.mousePosition){
        Vector3 camPos = this.transform.position;
        camPos.x += Camera.main.ScreenToWorldPoint(inputPos).x -
Camera.main.ScreenToWorldPoint(Input.mousePosition).x;
        camPos.y += Camera.main.ScreenToWorldPoint(inputPos).y -
Camera.main.ScreenToWorldPoint(Input.mousePosition).y;
        this.transform.position = camPos;
        inputPos = Input.mousePosition;
    }
}

```

4.4.3 Dasar logika perpindahan kendaraan

Proses perpindahan kendaraan dalam simulasi adalah dengan memasukkan posisi lokasi vector yang terbaru. Lokasi vector terbaru ini selalu diberikan pada setiap frame simulasi. Dalam pengembangan di engine unity3D, setiap frame ini di letakkan pada fungsi update() setiap game object. Script yang bertanggung jawab untuk ini adalah sebagai berikut.

```

void Update () {
    moveDirection = Vector3.right;
    Vector3 currentPosition = transform.position;
    Vector3 target = moveDirection * moveSpeed + currentPosition;
    transform.position = Vector3.Lerp( currentPosition, target,
Time.deltaTime );
}

```

Fungsi `Vector3.Lerp` disini adalah pergerakan transisi dari `currentPosition` kepada target. Fungsi ini umum digunakan untuk transisi yang halus (smooth) dari game object di `unity3D`. Sedangkan untuk kendaraan turn over, memakai fungsi `RayCast` yang ditawarkan oleh `unity3D`. `RayCast` disini bisa diibaratkan sebagai sebuah tongkat tidak terlihat didepan object kendaraan. Jadi sebelum kendaraan menabrak dinding pembatas jalan, tongkat akan menyentuh duluan dan kemudian mencoba memutar arah sebanyak seperempat derajat dari posisi sekarang. Script contoh untuk penggunaannya tampak seperti berikut ini,

```
int layerMask = 1 << LayerMask.NameToLayer("top");
RaycastHit2D hit = Physics2D.Raycast(transform.position,
    transform.right, 10.0f, layerMask);
RaycastHit2D hitRight = Physics2D.Raycast(transform.position, (
    transform.right - transform.up), 0.5f, layerMask);
RaycastHit2D hitLeft = Physics2D.Raycast(
    transform.position, (transform.right + transform.up), 0.5f, layerMask);
Debug.DrawRay(transform.position,
    transform.right * 100.0f, Color.red);
Debug.DrawRay(transform.position,
    (transform.right - transform.up) * 0.5f, Color.red);
Debug.DrawRay(transform.position,
    (transform.right + transform.up) * 0.5f, Color.red);
if ((hit.collider != null ||
    hitRight.collider != null ||
    hitLeft.collider != null) && !isTurnOver) {
    counter = 15;
    if (hit.collider == null) {
        Debug.Log("maju depan");
        //do nothing
    }
    else if (hitRight.collider == null) {
        Debug.Log("ke kanan");
        turnOver(transform.right - transform.up);
    }
    else if (hitLeft.collider == null) {
        Debug.Log("ke kiri");
        turnOver(transform.right + transform.up);
    }
}
```

4.4.4 Kontrol Traffic Light

Terdapat dua kelompok *traffic light* pada jalan Soekarno-Hatta, yaitu lampu *traffic light* sebelum masuk jembatan kearah jalan M. T. Haryono dan *traffic light* pada perempatan persimpangan dengan jalan Coklat. Pada implementasinya, kedua kelompok lalu lintas ini memiliki sebuah controller yang terpisah. Sebuah contoh disini adalah controller yang mengatur *traffic light* di jembatan. Script untuk mengatur pergantian lampu ini adalah,

```
while(true){
    trafficJembatan.GetComponent<TrafficJembatan> ().setIndex (index);
    trafficPos.GetComponent<TrafficPosPolisi> ().setIndex ((index+1)%3);
    index=(index+1)%3;
    yield return new WaitForSeconds(timeTraffic);
}
```

Script tersebut tampak mengatur *trafficJembatan* pada jembatan dan satunya lagi di *trafficPos* pada pos polisi. Dua lampu lalu lintas pastinya memiliki state yang berbeda. Delay atau waktu setiap pergantian di tunjukkan pada variabel *timeTraffic*.

4.5 Pengujian Simulasi Dengan Black Box Testing

Skenario pengujian dengan membuat test case kemudian membandingkan output yang seharusnya terjadi dengan output yang dihasilkan oleh aplikasi simulasi.

Tabel 4.1 pengujian aplikasi simulasi dengan Black Box

Skenario	Output yang seharusnya terjadi	Output yang dihasilkan oleh aplikasi simulasi
Menambahkan jumlah kendaraan pada kedua posisi yang disediakan oleh form	Muncul kendaraan sesuai dengan jumlah yang di inputkan user	Muncul kendaraan sesuai dengan jumlah yang di inputkan user
Mengubah waktu jeda perpindahan traffic light yang ada pada simulasi	Waktu jeda traffic light mengalami perubahan sesuai input	Waktu jeda traffic light mengalami perubahan sesuai input
Mengubah jalur atau trek kendaraan	Kendaraan mengikuti trek yang di ubah	Kendaraan mengikuti trek yang di ubah
Membuat perlakuan lampu merah pada kendaraan	Kendaraan berhenti jika lampu merah dan melaju kembali jika lampu hijau	Kendaraan tidak terpengaruh traffic light
Membuat perlakuan persimpangan jalan pada kendaraan	Kendaraan mengikuti perubahan pada persimpangan	Kendaraan tidak terpengaruh oleh perlakuan persimpangan jalan
Mengubah kecepatan satu objek kendaraan	Kecepatan kendaraan bertambah	Bisa dilakukan dengan langsung mengubah variable
Membuat perlakuan tumbukan atau tabrakan antar kendaraan dari belakang	Kecepatan kendaraan yang akan trabakan berkurang sampai menyamai kecepatan kendaraan di depannya	Terjadi tabrakan, kemudian kecepatan tetap tidak berkurang
Membuat perlakuan tumbukan antara dua kendaraan dari depan	Sebelum terjadi tabrakan, kendaraan dapat menghindari tabrakan	Kendaraan bertabrakan

4.6 Integerasi Dengan Islam

Islam memandang bahwa agama adalah dasar dan pengatur kehidupan. Aqidah Islam menjadi basis dari segala ilmu pengetahuan. aqidah islam yang terwujud dalam apa-apa yang ada dalam Al-Qur'an dan Al-Hadits menjadi qaidah fikriyah (landasan pemikiran), yaitu suatu asas yang di atasnya dibangun seluruh bangunan pemikiran dan ilmu pengetahuan manusia.

Islam memerintahjan manusia untuk membngun segala pemikirannya berdasarkan aqidah islam, bukan lepas dari aqidah itu. Ini bisa kita pahami dari ayat yang pertama kali turun :

أَقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ ﴿١﴾

Artinya:

“Bacalah dengan (menyebut) nama Tuhanmu Yang menciptakan”.

(QS. Al-Alaq: 1)

Ayat ini berarti manusia telah diperintahkan untuk membaca guna memperoleh berbagai pemikiran dan pemahaman. Tetapi segala pemikirannya itu tidak boleh lepas dari aqidah Islam, karena iqra' haruslah dengan bismi rabbika, yaitu berasarkan iman kepada Allah, yang merupakan asas Aqidah islam.

Peran Islam sendiri dalam perkembangan sains dan teknologi, adalah bahwa Syariah Islam harus dijadikan standart pemanfaatan sains dan teknologi. Ketentuan halal-haram (hukum-hukum syariat Islam) wajib di jadikan tolok ukur dalam pemanfaatan sains dan teknologi, bagaimana pun juga bentuknya. sains dan

teknologi yang boleh dimanfaatkan, adalah yang telah dihalalkan oleh syariat Islam. Sedangkan sains dan teknologi yang tidak boleh dimanfaatkan, adalah yang telah diharamkan oleh syariat islam. Jika peran ini dapat dimanfaatkan oleh umat islam dengan baik, insyaAllah akan ada berbagai berkah dari Allah kepada umat islam dan juga seluruh umat manusia.

Dalam hal ini aplikasi simulasi arus lalu lintas sangat berkaitan dengan berkendara. Salah satu penyebab masalah dalam berlalu-lintas adalah kurangnya kesadaran pengemudi atau pengendara akan keselamatan dan peraturan lalu lintas. Dalam Islam telah di atur adab berkendara dan berjalan dalam ayat Al-Qur'an. (<http://al-atsariyyah.com/775.html>, diakses tanggal 15 Juli 2014)

Ayat Al-Qur'an tersebut yaitu:

وَالَّذِي خَلَقَ الْأَزْوَاجَ كُلَّهَا وَجَعَلَ لَكُم مِّنَ الْفُلْكِ وَالْأَنْعَامِ مَا تَرْكَبُونَ ﴿١٢﴾ لِّتَسْتَوُوا عَلَىٰ ظُهُورِهِ ثُمَّ تَذْكُرُوا نِعْمَةَ رَبِّكُمْ إِذَا اسْتَوَيْتُمْ عَلَيْهِ وَتَقُولُوا سُبْحَانَ الَّذِي سَخَّرَ لَنَا هَذَا وَمَا كُنَّا لَهُ مُقْرِنِينَ ﴿١٣﴾

Artinya:

“ dan yang menciptakan semua yang berpasang-pasangan dan menjadikan untukmu kapal dan binatang ternak yang kamu tunggangi. supaya kamu duduk di atas punggungnya kemudian kamu ingat nikmat Tuhanmu apabila kamu telah duduk di atasnya; dan supaya kamu mengucapkan: "Maha suci Tuhan yang telah menundukkan semua ini bagi Kami Padahal Kami sebelumnya tidak mampu menguasainya,” (QS. Az-Zukhruf : 12-13)

Kendaraan yang kita gunakan sebagai fasilitas untuk dapat pergi ke suatu tempat merupakan suatu nikmat yang sepatutnya kita syukuri. Rasa syukur hamba

terhadap Rabb yang telah memberinya karunia ditunjukkan melalui lisan berupa pujian dan sanjungan, dan juga melalui anggota badan dengan menundukkannya dalam ketaatan kepada Allah.

Rasulullah shallallahu ‘alaihi wa sallam adalah hamba yang paling bersyukur, beliau memberikan keteladanan bagi umatnya ketika berkendara yaitu dengan cara berdoa ketika naik kendaraan, bertakbir dan bertasbih selama perjalanan, berdoa ketika kendaraan tergelincir, membebani kendaraan sesuai daya angkut, tidak menjadikan kendaraan semata-mata sebagai tempat duduk dan memandang kendaraan yang lebih rendah.

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melakukan analisa, merancang dan mengimplementasikan aplikasi simulasi arus lalu lintas, penelitian ini telah menjawab bagaimana melakukan desain sistem, desain proses dan desain antar muka agar aplikasi simulasi arus lalu lintas ini dapat di implementasikan secara nyata. Berdasarkan implementasi dan uji coba sistem yang telah dilakukan aplikasi simulasi arus lalu lintas ini dapat menampilkan jumlah kendaraan dan menjalankan kendaraan dengan baik.

Namun masih ada banyak kekurangan pada aplikasi simulasi arus lalu lintas ini. Diantaranya yaitu aplikasi ini tidak memiliki rasio panjang dan lebar jalan dengan keadaan nyata dan pengaplikasian traffic light yang belum efektif.

5.2 Saran

Beberapa saran untuk pengembangan lebih lanjut antara lain:

1. Pembuatan aplikasi simulasi ini sangat baik dikembangkan dengan engine Unity 3D.
2. Memberikan setiap kendaraan Artificial Intelligence akan menambah akurasi secara fungsional dari aplikasi ini
3. Penggunaan rasio dengan presisi akan mencerminkan keadaan nyata.

DAFTAR PUSTAKA

- Sridadi, Bambang. 2009 *Pemodelan dan Simulasi Sistem*. Penerbit Informatika
- Ofyar Z. Tamin. *Perencanaan dan Pemodelan Transportasi*. Penerbit ITB.
- Darmawan. Erico H & Laurentius Risal. 2011 *Pemrograman Berorientasi Objek C#*. Penerbit Informatika
- Penerbit Andi. 2009. *Kreasi Grafis Menarik Menggunakan Corel Draw X4*. Penerbit Andi
- Saepuloh. *Belajar Corel Draw X3 (Versi 13)*, ilmuKomputer.com 2008.
- https://www.academia.edu/5295802/Mengenal_Use_Case_Diagram, (diakses tanggal 15 juli 2014)
- <http://id.wikipedia.org/wiki/Kemacetan>, (diakses tanggal 15 Juli 2014)
- <http://al-atsariyyah.com/775.html>, (diakses tanggal 15 juli 2014)
- <https://www.google.com/maps/dir/-7.9497593,112.6154847/-7.9372162,112.6269517/@-7.9419535,112.6235952,1572m/data=!3m1!1e3!4m4!4m3!1m0!1m0!3e0>, (diakses tanggal 15 juli 2014)
- <http://unity3dindo.forumid.net/t5-mengenal-lebih-dalam-tentang-unity3d>, (diakses tanggal 15 juli 2014)
- <http://ilmusipil.com/manajemen-lalu-lintas>, (diakses tanggal 15 juli 2014)