RANCANG BANGUN MEKANIK OPERASIONAL PADA GAME MANAJEMEN JALAN TOL

SKRIPSI

Angga Yusma Prasetia

(2207431014)

PROGRAM STUDI TEKNIK MULTIMEDIA DIGITAL

JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER

POLITEKNIK NEGERI JAKARTA

TAHUN 2026

RANCANG BANGUN MEKANIK OPERASIONAL PADA GAME MANAJEMEN JALAN TOL

SKRIPSI

Dibuat untuk Melengkapi Syarat-Syarat yang Diperlukan untuk Memperoleh Diploma Empat Politeknik

Angga Yusma Prasetia

(2207431014)

PROGRAM STUDI TEKNIK MULTIMEDIA DIGITAL

JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER

POLITEKNIK NEGERI JAKARTA

TAHUN 2026

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN BEBAS PLAGIARISME

Saya yang bertandatangan di bawah ini:

Nama : Angga Yusma Prasetia

NIM : 2207431014

Jurusan/Program Studi : T. Informatika dan Komputer/T. Multimedia Digital

Judul Skripsi : RANCANG BANGUN MEKANIK OPERASIONAL PADA GAME MANAJEMEN JALAN TOL

Menyatakan dengan sebenarnya bahwa skripsi ini benar-benar merupakan hasil karya saya sendiri, bebas dari peniruan terhadap karya dari orang lain. Kutipan pendapat dan tulisan orang lain ditunjuk sesuai dengan cara-cara penulisan karya ilmiah yang berlaku.

Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa dalam skripsi ini terkandung ciri-ciri plagiat dan bentuk-bentuk peniruan lain yang dianggap melanggar peraturan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Depok, 9 Juni 20226

Yang membuat pernyataan

Angga Yusma Prasetia

NIM. 2207431014

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

LEMBAR PENGESAHAN

Skripsi Diajukan Oleh :

Nama	: Angga Yusma Prasetia
NIM	: 2207431014
Program Studi	: Teknik Multimedia Digital
Judul Skripsi	: RANCANG BANGUN MEKANIK OPERASIONAL PADA GAME MANAJEMEN JALAN TOL

Telah diuji oleh tim penguji dalam sidang skripsi pada hari Senin, tanggal 22, Bulan Juni, Tahun 2026, dan Dinyatakan **LULUS**

Disahkan Oleh :

Pembimbing I	:	Ade Rahma Yuly, S.Kom, M.Ds	( )
Penguji I	:	Noorlela Marcheta, S.Kom., M.Kom.	( )
Penguji II	:	Malisa Huzaifa, S.Kom., M.T.	( )
Penguji II	:	Sinantya Feranti Anindya, S.T., M.T.	( )

Mengetahui :
Jurusan Teknik Informatika & Komputer

Ketua

Dr. Anita Hidayati S.Kom, M.Kom

NIP. 197908032003122003

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT atas segala rahmat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul *"Rancang Bangun Mekanik Operasional pada Game Manajemen Jalan Tol"* dengan baik. Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan pendidikan pada Program Studi Teknik Multimedia Digital.

Dalam proses penyusunan skripsi ini, penulis menyadari bahwa banyak pihak yang telah memberikan dukungan, bantuan, bimbingan, serta motivasi baik secara langsung maupun tidak langsung. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada:

1. Ibu Dr. Anita Hidayati, S.Kom., M.Kom. selaku Ketua Jurusan Teknik Informatika dan Komputer.
2. Ibu Malisa Huzaifa, S.Kom., M.T. selaku Kepala Program Studi Teknik Multimedia Digital (D4 TMD).
3. Ibu Ade Rahma Yuly, S.Kom., M.Ds. selaku dosen pembimbing yang telah memberikan arahan, bimbingan, dan masukan selama proses penyusunan skripsi ini.
4. Mama dan Ayah tercinta yang selalu memberikan doa, dukungan, semangat, serta motivasi tanpa henti kepada penulis.
5. Pihak Jasa Marga yang telah memberikan dukungan dan informasi yang membantu proses penelitian ini.
6. Yovan Athailah Supriadi selaku rekan satu tim skripsi yang telah bekerja sama dan saling membantu selama proses pengerjaan penelitian ini.
7. Teman-teman terdekat penulis yang selalu memberikan dukungan, hiburan, dan semangat selama proses penyusunan skripsi ini, yaitu Kim Carenina yang selalu menjadi pendengar yang baik bagi penulis, serta teman-teman grup "Tolongin" yaitu Ajie Muhamad Dzaqi, Muhammad Rafli Pratama, Reynaldo Kurniadi, dan Sagara Sing Sal Sabil yang selalu menemani, menghibur, dan memberikan dukungan kepada penulis di berbagai situasi.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

8. Terakhir, kepada diri penulis sendiri yang telah bertahan sejauh ini. Terima kasih karena tidak memilih menyerah di tengah rasa lelah, tekanan, dan berbagai keraguan yang datang selama proses penyusunan skripsi ini. Terima kasih karena tetap melangkah meskipun sering kali harus berjalan pelan, jatuh, lalu bangkit kembali. Semua proses, air mata, rasa takut, begadang panjang, hingga perjuangan melawan diri sendiri menjadi bagian dari perjalanan yang akhirnya mengantarkan penulis sampai di titik ini. Skripsi ini bukan hanya tentang sebuah penelitian, tetapi juga tentang bukti bahwa penulis mampu bertahan dan menyelesaikan apa yang telah dimulai.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi penyempurnaan karya ini di masa mendatang. Semoga skripsi ini dapat memberikan manfaat bagi pembaca dan menjadi kontribusi positif dalam pengembangan ilmu pengetahuan, khususnya di bidang game dan multimedia digital.

Depok, 9 Juni 2026

Penulis,

Angga Yusma Prasetia

NIM. 2207431014

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

SURAT PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Politeknik Negeri Jakarta, saya bertanda tangan dibawah ini :

Nama : Angga Yusma Prasetia

NIM : 22074301014

Jurusan/ProgramStudi : T.Informatika dan Komputer / T.Multimedia Digital

Demi pengembangan ilmu pengetahuan , menyetujui untuk memberikan kepada Politeknik Negeri Jakarta Hak Bebas Royalti Non-Eksklusif atas karya ilmiah saya yang berjudul :

"RANCANG BANGUN MEKANIK OPERASIONAL PADA GAME MANAJEMEN JALAN TOL"

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini Politeknik Negeri Jakarta Berhak menyimpan, mengalihmediakan/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyatan ini saya buat dengan sebenarnya.

Depok, 9 Juni 2026

Penulis,

Angga Yusma Prasetia

NIM. 2207431014

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

RANCANG BANGUN MEKANIK OPERASIONAL PADA GAME MANAJEMEN JALAN TOL

Abstrak

Perkembangan industri game saat ini tidak hanya dimanfaatkan sebagai media hiburan, tetapi juga mulai digunakan sebagai media simulasi interaktif yang mampu merepresentasikan kondisi operasional nyata. Penelitian ini bertujuan untuk merancang dan membangun mekanik operasional pada game manajemen jalan tol berbasis WebGL menggunakan Unity dengan mengadaptasi konsep operasional jalan tol dari PT Jasa Marga. Mekanik operasional yang dikembangkan meliputi sistem antrean kendaraan, pengelolaan gerbang tol, sistem pendapatan, maintenance, rescue, upgrade gerbang tol, dan monitoring kemacetan. Proses pengembangan dilakukan menggunakan metode iteratif yang mencakup tahap analisis, implementasi, pengujian, dan evaluasi untuk memperoleh balancing gameplay yang sesuai dengan konsep simulasi manajemen. Game dikembangkan menggunakan Unity 6 dengan bahasa pemrograman C# dan didistribusikan dalam bentuk WebGL melalui platform Itch.io. Hasil penelitian menunjukkan bahwa seluruh sistem utama permainan berhasil diimplementasikan dan berjalan dengan baik berdasarkan hasil Alpha Testing dan Beta Testing. Hasil pengujian menunjukkan bahwa game Toll Tycoon mampu memberikan simulasi pengelolaan operasional jalan tol yang baik serta memberikan pengalaman bermain yang positif bagi pengguna.

Kata Kunci: Game Manajemen, Game Simulasi, Jalan Tol, Unity, WebGL, Mekanik Operasional

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR ISI

SURAT PERNYATAAN BEBAS PLAGIARISME iii
LEMBAR PENGESAHAN iv
KATA PENGANTAR v
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI vii
ABSTRAK viii
DAFTAR ISI ix
DAFTAR TABEL xii
DAFTAR GAMBAR x
DAFTAR LAMPIRAN xv
BAB I 1
PENDAHULUAN 1
1.1 Latar Belakang Masalah 1
1.2 Perumusan Masalah 2
1.3 Batasan Masalah 3
1.4 Tujuan dan Manfaat 4
1.4.1 Tujuan 4
1.4.2 Manfaat 4
1.5 Sistematika Penulisan 5
BAB II 7
TINJAUAN PUSTAKA 7
2.1 Game Manajemen 7
2.2 Game Simulasi 7
2.3 Mekanik Operasional dalam Game 8
2.4 Balancing Gameplay 9
2.5 Jalan Tol dan Operasional Gerbang Tol 10
2.6 Unity 11
2.7 Game Development Life Cycle (GDLC) 11
2.8 Penelitian Terdahulu 13
BAB III 17
METODE PENELITIAN 17
3.1 Rancangan Penelitian 17
3.2 Tahapan Penelitian 17
3.2.1 Initiation 19
3.2.2 Pre-Production 21
3.2.3 Production 22
3.2.4 Testing 24
3.2.5 Beta Testing 25
3.2.6 Release 27

POLITEKNIK NEGERI JAKARTA

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.3 Objek Penelitian..... 28
3.4 Teknik Pengumpulan dan Analisis Data..... 28
3.4.1 Teknik Pengumpulan Data..... 29
3.4.2 Teknik Analisis Data..... 30
BAB IV..... 34
HASIL DAN PEMBAHASAN..... 34
4.1 Analisis Kebutuhan..... 34
4.1.1 Analisis Hasil Wawancara..... 34
4.1.2 Analisis Kebutuhan Fungsional..... 35
4.1.3 Analisis Kebutuhan Non-Fungsional..... 37
4.2 Perancangan Sistem..... 38
4.2.1 Flowchart Sistem..... 39
4.2.2 Storyboard..... 40
4.2.3 Perancangan Antarmuka Pengguna (UI Design)..... 43
4.2.4 Material Collection..... 46
4.3 Implementasi Sistem..... 47
4.3.1 Implementasi Antarmuka Pengguna..... 48
4.3.1.3 Implementasi Visual Feedback..... 50
4.3.2 Struktur Implementasi Sistem..... 51
4.3.3 Implementasi Sistem Spawn Kendaraan..... 52
4.3.3.1 Implementasi Spawn Kendaraan Pada Gerbang Tol..... 53
4.3.3.2 Implementasi Spawn Kendaraan Ruas Jalan Tol..... 58
4.3.3.3 Implementasi Sistem Event Rush Hour..... 62
4.3.4 Implementasi Sistem Pergerakan dan Antrean Kendaraan..... 66
4.3.4.1 Implementasi Sistem Pergerakan Kendaraan..... 66
4.3.4.2 Implementasi Sistem Antrean Kendaraan..... 71
4.3.5 Implementasi Sistem Pendapatan Gerbang Tol..... 76
4.3.5.1 Implementasi Sistem Transaksi Kendaraan..... 76
4.3.5.2 Implementasi Sistem Balancing Ekonomi..... 80
4.3.6 Implementasi Sistem Pendapatan Lalu Lintas Tol..... 85
4.3.6.1 Implementasi Sistem Spawn dan Distribusi Kendaraan..... 86
4.3.6.2 Implementasi Sistem Maintenance Jalur..... 90
4.3.6.3 Implementasi Sistem Balancing Lalu Lintas..... 95
4.3.6.4 Implementasi Sistem Rescue Kendaraan..... 99
4.3.6.5 Implementasi Sistem Efisiensi Jalan Tol..... 104
4.3.7 Implementasi Sistem Penyimpanan Data..... 108
4.3.7.1 Implementasi Penyimpanan Pendapatan..... 109
4.3.7.2 Implementasi Penyimpanan Kondisi Jalur..... 112
4.3.7.3 Implementasi Penyimpanan Progress Gerbang Tol..... 115
4.3.8 Implementasi Sistem Game Over..... 121

4.3.9 Implementasi Sistem Audio..... 123
4.3.10 Implementasi Sistem Loading..... 126
4.4 Pengujian..... 132
4.4.1 Deskripsi Pengujian..... 132
4.4.2 Prosedur Pengujian..... 132
4.4.2.1 Alpha Testing..... 133
4.4.2.2 Beta Testing..... 137
4.4.3 Data Hasil Pengujian..... 143
4.4.3.1 Hasil Alpha Testing..... 143
4.4.3.2 Hasil Beta Testing Umum..... 147
4.4.3.3 Hasil Beta Testing Expert Game..... 152
4.4.4 Analisis Data / Evaluasi Pengujian..... 156
4.4.4.1 Analisis Alpha Testing..... 156
4.4.4.2 Analisis Beta Testing oleh Pengguna Umum..... 157
4.4.4.3 Analisis Beta Testing Expert Game..... 159
BAB V..... 161
PENUTUP..... 161
5.1 Simpulan..... 161
5.2 Saran..... 161
DAFTAR PUSTAKA..... 163

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR TABEL

Tabel 3.1 Skor Beta Testing..... 32
Tabel 3.2 Interval Skala Likert..... 32
Tabel 4.1 Hasil Wawancara dan Implementasi..... 36
Tabel 4.2 Daftar Kebutuhan Fungsional..... 37
Tabel 4.3 Kebutuhan Perangkat Keras..... 38
Tabel 4.4 Kebutuhan Perangkat Lunak..... 39
Tabel 4.5 Storyboard Game..... 41
Tabel 4.6 Antarmuka Game..... 44
Tabel 4.7 Material Collection..... 47
Tabel 4.8 Implementasi Antarmuka Pada Unity..... 49
Tabel 4.9 Implementasi balancing ekonomi setiap gerbang..... 84
Tabel 4.10 Perkembangan Ekonomi Pada Gerbang 1..... 85
Tabel 4.11 Balancing kondisi jalur..... 98
Tabel 4.12 Implementasi balancing maintenance..... 99
Tabel 4.13 Pertanyaan Alpha testing..... 135
Tabel 4.14 Pertanyaan Beta Testing Umum..... 139
Tabel 4.15 Pertanyaan Beta Testing Expert Game..... 141
Tabel 4.16 Pertanyaan Beta Testing Kritik dan Saran Expert Game..... 144
Tabel 4.17 Hasil Alpha Testing..... 145
Tabel 4.18 Hasil Beta Testing Pengguna Umum..... 148
Tabel 4.19 Hasil Skor Aktual..... 152
Tabel 4.20 Hasil Beta Testing Expert Game..... 154

POLITEKNIK NEGERI JAKARTA

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR GAMBAR

Gambar 3.1 Struktur Game development Life Cycle.....20
Gambar 4.1 Flowchart Game Toll Tycoon.....41
Gambar 4.2 Implementasi Visual Feedback pada unity.....51
Gambar 4.3 Potongan kode floating text.....52
Gambar 4.4 Struktur Manajemen File.....53
Gambar 4.5 Script Timer-Based Spawning.....56
Gambar 4.6 Script Safety-Based Spawn Checking.....57
Gambar 4.7 Script Adaptive Spawn Interval.....58
Gambar 4.8 Implementasi Script Spawn Pada Unity.....59
Gambar 4.9 Script Timer-Based Spawning.....60
Gambar 4.10 Script Safety-Based Spawn Checking.....61
Gambar 4.11 Script Randomized Vehicle Selection.....62
Gambar 4.12 Implementasi Spawn Kendaraan Ruas pada Unity.....62
Gambar 4.13 Script Timer-Based Event Scheduling.....64
Gambar 4.14 Script Event Rush Hour.....65
Gambar 4.15 Script Dynamic Parameter Adjustment.....65
Gambar 4.16 Implementasi Sistem Event Rush Hour pada Unity.....66
Gambar 4.17 Script Waypoint Following Algorithm.....69
Gambar 4.18 Script Obstacle Detection.....70
Gambar 4.19 Script Dynamic Speed Control.....71
Gambar 4.20 Implementasi Sistem Pergerakan Kendaraan pada Unity.....72
Gambar 4.21 Script Queue Detection Algorithm.....74
Gambar 4.22 Script First In First Out.....75
Gambar 4.23 Implementasi Sistem Antrean Kendaraan pada Unity.....76
Gambar 4.24 Script Transaction Processing Algorithm.....79
Gambar 4.25 Script Accumulator Pattern.....79
Gambar 4.26 Implementasi Sistem Transaksi Kendaraan pada Unity.....80
Gambar 4.27 Script Random Vehicle Selection Algorithm.....88
Gambar 4.28 Script Spawn Safety Detection.....89
Gambar 4.29 Script Waypoint Distribution.....90
Gambar 4.30 Implementasi Sistem Spawn dan Distribusi Kendaraan.....90
Gambar 4.31 Script Health Decay Algorithm.....92
Gambar 4.32 Script Maintenance Recovery Algorithm.....93
Gambar 4.33 Script Maintenance Recovery Algorithm.....94
Gambar 4.34 Implementasi Sistem Maintenance Jalur pada Unity.....95
Gambar 4.35 Script Breakdown Countdown Algorithm.....102

Gambar 4.36 Script Rescue Process Management.....102
Gambar 4.37 Script Rescue Resource Management.....103
Gambar 4.38 Implementasi Sistem Rescue Kendaraan pada Unity.....104
Gambar 4.39 Script Weighted Efficiency Calculation Algorithm.....106
Gambar 4.40 Script Threshold-Based Efficiency Classification.....107
Gambar 4.41 Implementasi Sistem Efisiensi Jalan Tol pada Unity.....108
Gambar 4.43 Script Persistent Money Storage Algorithm.....111
Gambar 4.44 Script Persistent Money Storage Algorithm.....112
Gambar 4.45 Implementasi Penyimpanan Pendapatan pada Unity.....113
Gambar 4.46 Script Lane State Saving.....114
Gambar 4.47 Script Lane State Restoration Algorithm.....115
Gambar 4.48 Script Gate Progress Saving Algorithm.....117
Gambar 4.49 Script Gate Progress Restoration Algorithm.....118
Gambar 4.50 Script Queue Restoration Algorithm.....120
Gambar 4.51 Script Game Over Detection Algorithm.....123
Gambar 4.52 Script Game State Reset Algorithm.....124
Gambar 4.53 Implementasi Musik Pada Unity.....125
Gambar 4.54 Potongan Kode Backgroun Musik.....125
Gambar 4.55 Implementasi Efek Suara Pada Unity.....126
Gambar 4.56 Potongan Kode Efek Suara.....126
Gambar 4.57 Pengaturan Suara.....126
Gambar 4.58 Potongan Kode Pengaturan Suara.....127
Gambar 4.59 Script Loading Screen Management.....129
Gambar 4.60 Script Loading Progress Update Algorithm.....130
Gambar 4.61 Script Loading Animation Algorithm.....131
Gambar 4.62 Implementasi Loading pada Unity.....132

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR LAMPIRAN

Lampiran 1. Game Design Document TollTyconn.....142
Lampiran 2. Transkrip Wawancara..... 159
Lampiran 3. Dokumentasi Wawancara..... 167
Lampiran 4. CV Ahli Game..... 168
Lampiran 5. Hasil Responden Pengguna Umum..... 171
Lampiran 6. Hasil Beta Testing Expert Game.....173
Lampiran 7. Daftar Riwayat Hidup Penulis..... 180

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Video game telah menjadi salah satu medium hiburan yang paling diminati secara global, dengan lebih dari 3 miliar pemain aktif di seluruh dunia pada tahun 2022 (Eshuis et al., 2023). Berbeda dengan medium hiburan pasif seperti film dan televisi, *game* menawarkan pengalaman yang interaktif dan imersif di mana pemain secara aktif terlibat melalui pengambilan keputusan, penyelesaian tantangan, dan eksplorasi sistem. Elemen desain *game* — seperti mekanik permainan, tingkat tantangan, dan interaktivitas sistem — terbukti secara signifikan mempengaruhi rasa *enjoyment* pemain sebagai tujuan utama dari pengalaman bermain (Caroux & Pujol, 2024).

Di antara berbagai genre yang berkembang, *management game* dan *simulation game* memiliki daya tarik hiburan yang khas karena menempatkan pemain pada situasi pengambilan keputusan dalam kondisi yang dinamis (Suparno & Utami, 2025). Daya tarik genre ini bersumber dari kompleksitas sistem yang dikelola pemain — mulai dari pengelolaan sumber daya, pemecahan masalah, hingga merespons kondisi permainan yang terus berubah. *Game* simulasi terbukti mampu menghadirkan keterlibatan pemain yang tinggi secara psikologis maupun behavioral bahkan tanpa tujuan pembelajaran formal, yang membuktikan bahwa nilai hiburan genre ini berdiri sendiri dan tidak bergantung pada unsur edukasi (Rodrigues et al., 2022). Mekanik *gameplay* yang dirancang dengan baik menjadi komponen kritis yang menentukan kualitas pengalaman bermain karena merupakan fondasi interaksi antara pemain dengan sistem permainan.

Konteks operasional jalan tol memiliki potensi besar sebagai fondasi mekanik management *game* karena kekayaan dinamika sistemnya — mencakup antrian kendaraan di gerbang tol, pelayanan transaksi, maintenance fasilitas, rescue kendaraan bermasalah, dan monitoring kemacetan. Pengelolaan antrian di

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

gerbang tol merupakan persoalan yang kompleks dan terus berkembang seiring meningkatnya pengguna jalan tol dari waktu ke waktu (Jurnal Matematika Unisba, 2022). Kompleksitas tersebut justru menjadi sumber tantangan dan keseruan dalam permainan. Meskipun demikian, belum terdapat *game* simulasi manajemen yang secara khusus mengangkat konteks operasional jalan tol di Indonesia dengan mengacu pada Standar Pelayanan Minimum (SPM) dari PT Jasa Marga, sehingga terdapat kesenjangan yang menjadi peluang pengembangan game simulasi manajemen berbasis konteks lokal.

Penelitian ini mengembangkan game simulasi manajemen jalan tol berjudul "Toll Tycoon" menggunakan game engine Unity 6 dengan platform WebGL, yang didasarkan pada hasil wawancara dan observasi langsung bersama pihak PT Jasa Marga serta mengacu pada Standar Pelayanan Minimum (SPM). Mekanik operasional yang dirancang mencakup sistem antrian kendaraan, pengelolaan gerbang tol, maintenance, rescue, sistem pendapatan, upgrade gerbang tol, dan monitoring kemacetan. Kualitas pengalaman bermain yang dihasilkan sangat bergantung pada proses balancing gameplay, mengingat sistem yang tidak seimbang akan menyebabkan pemain merasa frustrasi atau bosan sehingga mengurangi nilai hiburan permainan (Fu et al., 2022). Integrasi algoritma pathfinding dan Finite State Machine (FSM) pada agen kendaraan juga diterapkan untuk menghasilkan perilaku NPC yang lebih realistis dan responsif dalam menunjang kualitas simulasi secara keseluruhan (Amalia et al., 2024). Melalui penelitian ini diharapkan tercipta sebuah game hiburan bergenre management simulation yang engaging, menantang, dan relevan dengan konteks operasional jalan tol di Indonesia.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah dipaparkan, terdapat kesenjangan antara potensi konteks operasional jalan tol sebagai fondasi mekanik permainan dengan ketiadaan game simulasi manajemen yang mengangkat konteks lokal Indonesia. Rumusan masalah dalam penelitian ini meliputi dua hal. Pertama, bagaimana merancang dan membangun mekanik operasional pada game simulasi manajemen — mencakup sistem antrian kendaraan, pengelolaan gerbang tol,

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

maintenance, rescue, dan monitoring kemacetan — yang mampu merepresentasikan logika operasional jalan tol berdasarkan Standar Pelayanan Minimum (SPM) PT Jasa Marga ke dalam bentuk gameplay yang engaging dan menyenangkan. Kedua, bagaimana menerapkan balancing gameplay pada game simulasi manajemen jalan tol agar menghadirkan pengalaman bermain yang menantang sekaligus nyaman bagi pemain.

1.3 Batasan Masalah

Penelitian ini memiliki batasan masalah sebagai berikut agar lebih terarah sesuai tujuan yang ditetapkan. memiliki beberapa batasan masalah sebagai berikut:

1. Penelitian berfokus pada rancang bangun mekanik operasional pada game manajemen dengan konteks pengelolaan lalu lintas jalan tol.
2. Mekanik operasional yang dibahas pada penelitian ini meliputi sistem antrean kendaraan, pengelolaan gerbang tol, sistem pendapatan, *maintenance*, *rescue*, upgrade gerbang tol, dan monitoring kemacetan.
3. Lokasi gerbang tol yang digunakan pada penelitian ini terbatas pada gerbang tol Cililitan dan gerbang tol Bogor. Pemilihan kedua lokasi tersebut didasarkan pada hasil wawancara dan observasi bersama pihak PT Jasa Marga sebagai referensi dalam perancangan lingkungan dan operasional gerbang tol pada permainan.
4. Game dikembangkan menggunakan *game engine* Unity 6 dengan bahasa pemrograman C# dan didistribusikan dalam bentuk WebGL melalui platform Itch.io.
5. Game yang dikembangkan bersifat *single player* dan tidak menggunakan sistem jaringan (*multiplayer*) maupun data lalu lintas secara *real-time*.
6. Representasi operasional jalan tol pada game disusun berdasarkan hasil wawancara, observasi, dan Standar Pelayanan Minimum (SPM) dari PT Jasa Marga, namun tidak bertujuan untuk merepresentasikan simulasi lalu lintas secara realistis penuh.
7. Penentuan parameter permainan seperti durasi *rescue*, waktu *maintenance*, tingkat kemunculan kendaraan, dan biaya operasional dilakukan melalui

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

proses iterasi, pengujian internal, dan *expert testing*, serta tidak menggunakan perhitungan matematis transportasi secara mendalam.

8. Tahap pengembangan pada penelitian ini berfokus pada proses implementasi dan balancing mekanik permainan, serta belum membahas tahap penyempurnaan akhir (*refinement*) secara mendalam.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan dari penelitian ini adalah merancang dan membangun mekanik operasional pada game simulasi manajemen jalan tol "Toll Tycoon" yang mampu menghadirkan pengalaman hiburan yang engaging dan menyenangkan bagi pemain melalui sistem antrian kendaraan, pengelolaan gerbang tol, maintenance, rescue, dan monitoring kemacetan — dengan berlandaskan pada Standar Pelayanan Minimum (SPM) dan hasil observasi operasional PT Jasa Marga — serta menerapkan proses balancing gameplay untuk memastikan keseimbangan antara tingkat tantangan dan kenyamanan bermain.

1.4.2 Manfaat

Manfaat yang diharapkan dari penelitian ini adalah sebagai berikut.

1. Penelitian ini diharapkan menghasilkan produk game hiburan bergenre management simulation yang mengangkat konteks lokal Indonesia, yaitu operasional jalan tol Jagorawi, sehingga memberikan pengalaman bermain yang relevan dan menarik bagi pemain.
2. Penelitian ini diharapkan dapat memberikan pengalaman dalam mengimplementasikan proses pengembangan game menggunakan metode *Game Development Life Cycle* (GDLC), khususnya pada tahap implementasi mekanik permainan dan balancing gameplay..
3. Penelitian ini diharapkan dapat menjadi referensi dalam pengembangan game manajemen dan game simulasi, khususnya

mengenai penerapan logika sistem nyata ke dalam mekanik permainan yang entertaining dan balanced.

1.5 Sistematika Penulisan

Sistematika penulisan disusun untuk memberikan gambaran alur penelitian dan memudahkan pembaca dalam memahami tahapan penelitian mulai dari latar belakang, proses pengembangan game, hingga hasil pengujian dan kesimpulan penelitian.

1. **BAB I PENDAHULUAN**
   Bab ini berisi latar belakang penelitian, perumusan masalah, batasan masalah, tujuan dan manfaat penelitian, serta sistematika penulisan. Pada bab ini dijelaskan alasan dan dasar dilakukannya penelitian mengenai rancang bangun mekanik operasional pada game manajemen berbasis simulasi jalan tol.
2. **BAB II TINJAUAN PUSTAKA**
   Bab ini berisi penelitian terdahulu yang relevan, teori-teori pendukung, serta pembahasan mengenai game manajemen, game simulasi, mekanik operasional, *balancing gameplay*, *player experience*, dan metode pengembangan game yang digunakan dalam penelitian.
3. **BAB III METODOLOGI PENELITIAN**
   Bab ini menjelaskan rancangan penelitian, tahapan penelitian menggunakan metode *Game Development Life Cycle* (GDLC), objek penelitian, teknik pengumpulan data, teknik analisis data, serta metode pengujian yang digunakan dalam penelitian.
4. **BAB IV HASIL DAN PEMBAHASAN**
   Bab ini berisi hasil analisis kebutuhan, perancangan sistem, implementasi game, proses iterasi dan *balancing gameplay*, pengujian sistem, serta hasil evaluasi terhadap mekanik operasional dan pengalaman bermain pemain pada game yang dikembangkan.
5. **BAB V PENUTUP**
   Bab ini berisi kesimpulan dari hasil penelitian yang telah dilakukan serta saran yang dapat digunakan untuk pengembangan penelitian selanjutnya

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB II

TINJAUAN PUSTAKA

2.1 Game Manajemen

Game manajemen (*management game*) merupakan genre game yang berfokus pada proses pengelolaan sumber daya, pengambilan keputusan, dan pengaturan sistem untuk mencapai tujuan tertentu. Dalam game manajemen, pemain berperan sebagai pengelola yang harus mengatur berbagai kondisi dan sumber daya agar sistem permainan dapat berjalan dengan baik. Genre ini umumnya menghadirkan kondisi yang dinamis sehingga pemain dituntut untuk menentukan strategi yang tepat dalam menghadapi berbagai tantangan yang muncul selama permainan berlangsung (Geetha et al., 2024).

Salah satu aspek utama dalam game manajemen adalah *resource management* atau pengelolaan sumber daya. Sumber daya dalam game dapat berupa uang (*revenue*), waktu, fasilitas, maupun unit pendukung yang digunakan untuk mempertahankan kondisi sistem permainan. Pemain harus mampu menggunakan sumber daya yang tersedia secara efektif agar dapat melakukan tindakan tertentu seperti peningkatan fasilitas (*upgrade*), *maintenance*, maupun penanganan kondisi darurat (*emergency handling*) (marwani et al., 2023).

Selain pengelolaan sumber daya, game manajemen juga menekankan pada proses pengambilan keputusan (*decision making*). Pemain dituntut untuk memilih tindakan yang paling efektif di tengah keterbatasan sumber daya dan kondisi permainan yang terus berubah. Keputusan yang diambil pemain akan memengaruhi kondisi sistem permainan, sehingga setiap tindakan memiliki konsekuensi tertentu terhadap jalannya permainan (Geetha et al., 2024).

Game manajemen juga memiliki karakteristik simulasi sistem yang menggambarkan hubungan antar komponen permainan secara dinamis. Perubahan pada satu mekanik permainan dapat memengaruhi kondisi mekanik lainnya, sehingga pemain harus memahami hubungan antar sistem untuk menjaga stabilitas permainan. Dalam konteks penelitian ini, sistem antrean kendaraan, pendapatan,

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

maintenance, *rescue*, dan pengelolaan gerbang tol menjadi bagian dari mekanik operasional yang saling memengaruhi selama permainan berlangsung (Raghothama et al., 2022).

2.2 Game Simulasi

Game simulasi (*simulation game*) merupakan jenis game yang dirancang untuk merepresentasikan kondisi atau sistem tertentu ke dalam bentuk simulasi interaktif. Pada game simulasi, pemain diberikan kesempatan untuk berinteraksi dengan sistem permainan yang dibuat menyerupai kondisi nyata sehingga pemain dapat memahami proses, hubungan antar sistem, dan dampak dari keputusan yang diambil selama permainan berlangsung. Game simulasi tidak hanya digunakan sebagai media hiburan, tetapi juga sebagai media pembelajaran dan pelatihan dalam memahami suatu sistem (Rogers et al., 2022).

Salah satu aspek penting dalam game simulasi adalah unsur realisme (*realism*) dan representasi dunia nyata. Game simulasi yang baik harus mampu menghadirkan kondisi yang memiliki keterkaitan dengan sistem nyata agar proses pembelajaran dan pengambilan keputusan dapat tersampaikan dengan baik kepada pemain. Representasi tersebut tidak harus sepenuhnya realistis, namun perlu memiliki logika sistem yang sesuai dengan kondisi operasional yang disimulasikan (Rogers et al., 2022).

Selain unsur realisme, game simulasi juga memiliki karakteristik *dynamic system* atau sistem yang dinamis. Sistem permainan harus mampu merespons setiap tindakan pemain dan menghasilkan perubahan kondisi secara berkelanjutan. Perubahan tersebut dapat berupa peningkatan tingkat kesulitan, perubahan kondisi lingkungan, maupun munculnya tantangan baru yang memengaruhi jalannya permainan. Dalam konteks penelitian ini, kondisi dinamis ditunjukkan melalui perubahan kepadatan kendaraan, antrean lalu lintas, kendaraan mogok, serta perubahan kondisi operasional jalan tol yang memengaruhi keputusan pemain selama permainan berlangsung (Alyamani et al., 2023).

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.3 Mekanik Operasional dalam Game

Mekanik operasional dalam game merupakan kumpulan sistem dan aturan yang mengatur bagaimana pemain berinteraksi dengan permainan serta bagaimana sistem permainan merespons tindakan pemain. Mekanik ini menjadi dasar utama dalam menciptakan gameplay yang interaktif dan menantang, karena setiap tindakan yang dilakukan pemain akan memengaruhi kondisi permainan secara keseluruhan. Dalam game manajemen, mekanik operasional berperan penting dalam mendukung proses pengambilan keputusan dan pengelolaan sumber daya selama permainan berlangsung (Geetha et al., 2024).

Salah satu mekanik operasional yang umum digunakan adalah sistem antrean (*queue system*). Sistem antrean digunakan untuk mengatur aliran objek atau karakter dalam permainan agar dapat diproses secara teratur. Dalam konteks simulasi operasional jalan tol, sistem antrean digunakan untuk merepresentasikan aliran kendaraan pada gerbang tol. Pada kondisi nyata, proses transaksi pada gerbang tol memiliki standar waktu pelayanan tertentu agar antrean kendaraan tidak menyebabkan kemacetan yang berlebihan (Sewagegn, 2022).

Mekanik resource management dan upgrade system juga menjadi bagian penting dalam game manajemen. Pemain harus mengelola sumber daya seperti uang (*revenue*) untuk melakukan tindakan tertentu, seperti peningkatan fasilitas (*upgrade*) guna meningkatkan kapasitas pelayanan sistem permainan. Dalam penelitian ini, sistem pendapatan digunakan pemain untuk melakukan peningkatan gerbang tol dan mempertahankan kondisi operasional selama permainan berlangsung (Zhou, 2024).

Mekanik *maintenance* dan *emergency handling* digunakan untuk menangani gangguan yang terjadi pada sistem permainan. *Maintenance* digunakan untuk menjaga kondisi fasilitas agar tetap berfungsi dengan baik, sedangkan *emergency handling* digunakan untuk menangani kondisi darurat seperti kendaraan mogok yang dapat menghambat jalannya sistem (Adriansyah, Dwicahyono, 2022). Dalam konteks operasional jalan tol, penanganan gangguan dilakukan menggunakan unit pendukung seperti kendaraan derek (*rescue*) untuk menjaga kelancaran lalu lintas.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Seluruh mekanik operasional tersebut perlu melalui proses *balancing gameplay* agar permainan tidak terlalu mudah maupun terlalu sulit. *Balancing* dilakukan dengan menyesuaikan hubungan antar mekanik permainan seperti tingkat kemunculan kendaraan, durasi penanganan gangguan, biaya operasional, dan sistem pendapatan. Proses balancing memastikan gameplay tetap berjalan secara logis, menantang, dan mampu memberikan pengalaman bermain yang lebih baik bagi pemain (Parreiras et al., 2023).

2.4 Balancing Gameplay

Balancing gameplay merupakan proses penyesuaian mekanik permainan agar tingkat kesulitan, tantangan, dan interaksi antar sistem dalam game dapat berjalan secara seimbang. Pada game manajemen, proses balancing menjadi aspek penting karena setiap mekanik saling memengaruhi kondisi permainan. Balancing bertujuan untuk menjaga agar permainan tidak terlalu mudah maupun terlalu sulit sehingga pemain tetap merasa tertantang selama permainan berlangsung (Parreiras et al., 2023).

Dalam proses pengembangan game, balancing umumnya dilakukan melalui pendekatan *iterative design*. Pendekatan ini dilakukan dengan proses percobaan, analisis, pengujian, dan perbaikan secara berulang terhadap mekanik permainan yang telah dirancang. Melalui proses iterasi tersebut, pengembang dapat mengevaluasi apakah sistem permainan telah berjalan sesuai dengan tujuan gameplay yang diinginkan (Parreiras et al., 2023).

Proses balancing juga dilakukan menggunakan metode trial and error untuk menentukan parameter permainan yang sesuai. Pada penelitian ini, beberapa parameter seperti tingkat kemunculan kendaraan, waktu transaksi gerbang tol, durasi *rescue*, biaya operasional, dan sistem pendapatan ditentukan melalui proses pengujian internal, observasi, dan evaluasi gameplay. Penentuan parameter tersebut dilakukan agar sistem permainan tetap memiliki keterkaitan dengan kondisi operasional jalan tol tanpa menghilangkan unsur hiburan dalam permainan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Balancing gameplay juga berkaitan dengan *difficulty balancing*, yaitu proses pengaturan tingkat kesulitan permainan agar pemain tetap merasa tertantang tanpa menimbulkan rasa frustrasi. Tingkat kesulitan dalam game dapat dipengaruhi oleh peningkatan jumlah kendaraan, antrean lalu lintas, kondisi jalan, maupun keterbatasan sumber daya yang dimiliki pemain. Balancing diperlukan agar pemain tetap terdorong untuk melakukan pengambilan keputusan secara strategis selama permainan berlangsung (Hong et al., 2023).

2.5 Jalan Tol dan Operasional Gerbang Tol

Jalan tol merupakan jalan umum yang dirancang untuk mendukung mobilitas kendaraan dengan tingkat pelayanan tertentu melalui sistem akses terbatas dan pengelolaan operasional khusus. Dalam pengoperasiannya, jalan tol memiliki tujuan untuk menjaga kelancaran, keamanan, dan kenyamanan pengguna jalan melalui berbagai sistem pelayanan dan pengawasan lalu lintas (Yu, 2024).

Dalam pengelolaan operasional jalan tol, terdapat Standar Pelayanan Minimum (SPM) yang digunakan sebagai acuan dalam menjaga kualitas pelayanan kepada pengguna jalan. Standar tersebut mencakup aspek kelancaran lalu lintas, keamanan perjalanan, penanganan gangguan, serta fasilitas operasional pendukung yang tersedia pada ruas jalan tol.

Berdasarkan hasil wawancara dan observasi dengan pihak PT Jasa Marga, operasional jalan tol melibatkan berbagai aktivitas pengelolaan lalu lintas seperti pengaturan antrean kendaraan pada gerbang tol, penanganan gangguan kendaraan, maintenance fasilitas jalan tol, serta monitoring kondisi lalu lintas. Operasional tersebut juga didukung oleh fasilitas pendukung seperti kendaraan patroli dan kendaraan derek (*rescue*) untuk membantu menjaga kelancaran lalu lintas pada ruas jalan tol.

Dalam konteks penelitian ini, konsep operasional jalan tol tersebut digunakan sebagai dasar dalam perancangan mekanik permainan pada game simulasi manajemen. Beberapa mekanik operasional seperti sistem antrean kendaraan, pengelolaan gerbang tol, maintenance, rescue, dan monitoring kemacetan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kemudian direpresentasikan ke dalam bentuk gameplay interaktif yang disesuaikan dengan kondisi operasional jalan tol secara simulatif. Mekanik operasional tersebut kemudian direpresentasikan ke dalam bentuk gameplay interaktif yang disesuaikan dengan kondisi operasional jalan tol secara simulatif.

2.6 Unity

Unity merupakan salah satu *game engine* yang digunakan untuk mengembangkan game dua dimensi maupun tiga dimensi. Unity menyediakan berbagai fitur yang mendukung proses pengembangan game, seperti pengelolaan aset, sistem fisika, animasi, audio, antarmuka pengguna (*User Interface*), serta dukungan bahasa pemrograman C#. Dengan fitur tersebut, Unity banyak digunakan dalam pengembangan game karena mampu mendukung proses implementasi sistem permainan secara fleksibel dan efisien (Nova et al., 2025).

Pada penelitian ini, pengembangan game dilakukan menggunakan Unity 6 dengan bahasa pemrograman C#. Unity digunakan untuk membangun seluruh mekanik permainan, antarmuka pengguna, sistem kendaraan, serta interaksi antar sistem dalam game manajemen yang dikembangkan.

Game yang dikembangkan menggunakan platform WebGL, sebagai media distribusi permainan. WebGL memungkinkan game dijalankan langsung melalui browser tanpa memerlukan proses instalasi tambahan pada perangkat pengguna (Mehanna & Rudametkin, 2023). Pada penelitian ini, hasil build WebGL didistribusikan melalui platform Itch.io sehingga game dapat diakses dan dimainkan secara daring melalui browser.

2.7 Game Development Life Cycle (GDLC)

Game Development Life Cycle (GDLC) merupakan metode pengembangan game yang digunakan untuk mengatur tahapan proses pembuatan game secara terstruktur. Metode ini mencakup beberapa tahapan mulai dari perencanaan konsep, perancangan sistem, implementasi, hingga pengujian permainan. GDLC digunakan agar proses pengembangan game dapat berjalan lebih terarah dan sesuai dengan tujuan yang ingin dicapai (Arifudin et al., 2022).

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tahapan dalam GDLC umumnya terdiri atas *initiation*, *pre-production*, *production*, dan *testing*. Tahap *initiation* merupakan tahap awal yang dilakukan untuk menentukan konsep permainan, ide dasar, serta tujuan pengembangan game. Tahap *pre-production* dilakukan dengan membuat rancangan sistem, mekanik permainan, storyboard, dan kebutuhan aset yang digunakan selama pengembangan. Selanjutnya, tahap *production* merupakan proses implementasi seluruh rancangan ke dalam bentuk game yang dapat dimainkan. Pada tahap ini dilakukan proses pembuatan antarmuka pengguna, implementasi mekanik permainan, integrasi aset, serta pengembangan sistem gameplay menggunakan *game engine* Unity (Syarif et al., 2022).

Tahap *testing* dilakukan untuk menguji fungsi sistem, gameplay, dan mekanik permainan yang telah dikembangkan (Syarif et al., 2022). Pada penelitian ini, proses pengujian dilakukan melalui *alpha testing* dan *beta testing* untuk mengetahui apakah sistem permainan telah berjalan sesuai rancangan serta untuk mengevaluasi hasil balancing gameplay yang telah dilakukan.

Dalam metode GDLC, proses pengembangan game tidak selalu berjalan secara linear dari tahap produksi menuju tahap akhir. Terdapat proses iterasi atau perulangan antara tahap *production* dan *testing* yang memungkinkan pengembang melakukan evaluasi dan perbaikan terhadap sistem permainan secara berulang. Proses iterasi dilakukan ketika ditemukan mekanik permainan yang belum berjalan sesuai tujuan gameplay atau ketika tingkat kesulitan permainan masih belum seimbang (Arief et al., 2023).

Pada penelitian ini, proses iterasi menjadi salah satu bagian penting dalam pengembangan mekanik operasional game manajemen. Beberapa mekanik seperti sistem antrean kendaraan, tingkat kemunculan kendaraan, durasi *rescue*, sistem *maintenance*, dan sistem pendapatan mengalami penyesuaian berdasarkan hasil pengujian internal dan *expert testing*. Proses tersebut dilakukan melalui tahapan analisis masalah, pencarian solusi, implementasi perbaikan, dan evaluasi ulang terhadap gameplay yang dihasilkan. Dengan adanya proses iterasi tersebut, mekanik permainan dapat disesuaikan agar gameplay tetap logis, menantang, dan mampu merepresentasikan kondisi operasional jalan tol secara simulatif.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2.8 Penelitian Terdahulu

Judul Jurnal	Penulis dan Tahun	Mekanisme yang Diambil	Implementasi pada Game Tol
AI-Driven Traffic Simulation using Unity: Implementing Finite State Machines for Adaptive NPC Behaviour	Amalia et al. (2024)	Membangun simulasi lalu lintas berbasis Unity menggunakan FSM untuk mengatur perilaku kendaraan secara adaptif dan responsif terhadap kondisi lalu lintas.	Penelitian ini menjadi referensi teknis dalam penerapan FSM untuk sistem NPC kendaraan pada Toll Tycoon, khususnya dalam mengatur perilaku kendaraan antrian, kecepatan, dan respons terhadap kondisi gerbang tol.
Creating an Interactive Urban Traffic System for the Simulation of Different Traffic Scenarios	Weißmann et al. (2023)	Mengembangkan lingkungan simulasi lalu lintas perkotaan berbasis Unity yang dinamis untuk mensimulasikan kondisi lalu lintas dengan berbagai skenario secara interaktif.	Menjadi referensi dalam perancangan sistem lalu lintas dinamis pada Toll Tycoon, khususnya dalam representasi kepadatan kendaraan dan perubahan kondisi arus lalu lintas selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Implementasi Finite State Machine untuk Dynamic Story dalam Game Simulasi Kampus	Rhamadan (2025)	Menerapkan metode FSM untuk mengatur perubahan kondisi permainan secara dinamis berdasarkan tindakan pemain pada game simulasi berbasis Unity.	Menjadi referensi dalam penerapan FSM untuk perubahan state pada Toll Tycoon, seperti perubahan kondisi gerbang tol, level upgrade, status maintenance, dan kondisi rescue berdasarkan keputusan pemain.

Penelitian mengenai simulasi lalu lintas berbasis game engine telah berkembang dengan berbagai pendekatan teknis. Amalia et al. (2024) melalui penelitian berjudul AI-Driven Traffic Simulation using Unity: Implementing Finite State Machines for Adaptive NPC Behaviour mengembangkan sistem simulasi lalu lintas berbasis Unity dengan memanfaatkan Finite State Machine (FSM) untuk mengatur perilaku NPC kendaraan yang adaptif dan responsif. Setiap agen kendaraan memiliki state yang berubah secara dinamis berdasarkan kondisi lingkungan sekitarnya, seperti kepadatan lalu lintas, hambatan, dan perubahan kecepatan. Pendekatan FSM pada penelitian tersebut menjadi referensi teknis yang relevan dengan Toll Tycoon, khususnya dalam perancangan perilaku kendaraan pada antrian gerbang tol, respons kendaraan terhadap kondisi gerbang, dan pengelolaan alur lalu lintas secara dinamis selama permainan berlangsung.

Penelitian lain dilakukan oleh Weißmann et al. (2023) melalui penelitian berjudul Creating an Interactive Urban Traffic System for the Simulation of Different

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Traffic Scenarios. Penelitian tersebut mengembangkan lingkungan simulasi lalu lintas perkotaan berbasis Unity yang mampu mensimulasikan berbagai skenario lalu lintas secara interaktif dan dinamis. Sistem yang dikembangkan dapat merespons perubahan kondisi lalu lintas secara real-time, termasuk kepadatan kendaraan, arus lalu lintas, dan skenario kemacetan. Konsep sistem lalu lintas dinamis pada penelitian tersebut menjadi referensi dalam perancangan sistem antrian kendaraan dan monitoring kemacetan pada Toll Tycoon, di mana kondisi lalu lintas berubah secara berkelanjutan sesuai dengan keputusan operasional yang diambil pemain selama permainan berlangsung.

Rhamadan (2025) melalui penelitian berjudul Implementasi Finite State Machine untuk Dynamic Story dalam Game Simulasi Kampus menerapkan metode FSM untuk mengatur perubahan kondisi permainan secara dinamis berdasarkan tindakan pemain dalam game simulasi berbasis Unity. Setiap keputusan yang diambil pemain memengaruhi state permainan sehingga menghasilkan alur gameplay yang adaptif. Pendekatan FSM pada penelitian tersebut memiliki kesamaan dengan Toll Tycoon karena kondisi permainan juga berubah secara dinamis berdasarkan keputusan operasional pemain, seperti perubahan level gerbang, status maintenance, kondisi rescue, serta perubahan pendapatan yang diperoleh selama permainan berlangsung.

Ketiga penelitian tersebut menunjukkan bahwa simulasi lalu lintas berbasis Unity dengan pendekatan FSM dan sistem kondisi dinamis merupakan fondasi teknis yang relevan dalam pengembangan game simulasi manajemen. Penelitian mengenai game simulasi yang secara khusus mengangkat konteks operasional manajemen jalan tol sebagai mekanik permainan belum ditemukan dalam literatur yang ada. Hal tersebut menjadi gap penelitian yang memperkuat urgensi Toll Tycoon sebagai game simulasi manajemen pertama yang mengintegrasikan sistem antrian kendaraan, pengelolaan gerbang tol, maintenance, rescue, dan monitoring kemacetan ke dalam gameplay yang didasarkan pada Standar Pelayanan Minimum (SPM) operasional jalan tol Indonesia.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB III

METODE PENELITIAN

3.1 Rancangan Penelitian

Penelitian ini merupakan penelitian pengembangan (*Research and Development*) yang bertujuan untuk merancang dan membangun mekanik operasional pada game manajemen dengan konteks pengelolaan jalan tol sebagai dasar gameplay. Penelitian tidak hanya berfokus pada proses pembuatan game, tetapi juga pada proses analisis, pencarian solusi, evaluasi, dan perbaikan mekanik permainan melalui proses iterasi desain secara berulang. Fokus utama penelitian adalah bagaimana mekanik operasional seperti sistem antrean kendaraan, pengelolaan gerbang tol, *maintenance*, *rescue*, sistem pendapatan, dan monitoring kemacetan dapat direpresentasikan ke dalam gameplay yang logis, dinamis, dan sesuai dengan kondisi operasional jalan tol.

Metode pengembangan yang digunakan dalam penelitian ini adalah *Game Development Life Cycle* (GDLC) yang terdiri atas tahap *initiation*, *pre-production*, *production*, *testing*, dan *release*. Namun, pada penelitian ini proses pengembangan lebih difokuskan pada tahapan *production* dan *testing* untuk melakukan evaluasi dan balancing mekanik permainan melalui proses iterasi secara berulang. Penelitian ini belum membahas tahap *refinement* atau penyempurnaan akhir secara mendalam maupun distribusi luas permainan karena keterbatasan waktu penelitian.

Pendekatan penelitian yang digunakan adalah *mixed method*, yaitu kombinasi antara pendekatan kualitatif dan kuantitatif. Pendekatan kualitatif digunakan pada tahap awal penelitian untuk memperoleh informasi mengenai kondisi operasional jalan tol sebagai dasar representasi dunia nyata ke dalam game simulasi. Data kualitatif diperoleh melalui wawancara dengan pihak PT Jasa Marga sebagai pengelola jalan tol. Informasi yang diperoleh digunakan untuk memahami

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

mekanisme operasional jalan tol, pengelolaan lalu lintas, Standar Pelayanan Minimum (SPM), serta kondisi dinamis yang terjadi pada operasional jalan tol.

Pendekatan kuantitatif digunakan pada tahap pengujian mekanik permainan melalui *alpha testing* dan *expert testing*. Data kuantitatif diperoleh melalui penyebaran kuesioner menggunakan skala Likert untuk mengevaluasi gameplay yang dikembangkan, khususnya terhadap aspek pemahaman pemain terhadap logika operasional permainan, pengambilan keputusan pemain, tingkat ketertarikan untuk memainkan kembali permainan (*replayability*), serta keinginan pemain untuk merekomendasikan permainan kepada pengguna lain. Selain data kuantitatif, pada tahap *expert testing* juga diperoleh data kualitatif berupa saran, masukan, dan evaluasi dari penguji ahli terhadap mekanik operasional dan balancing gameplay yang dikembangkan. Data tersebut digunakan sebagai bahan evaluasi dan perbaikan pada proses iterasi desain permainan.

Melalui kombinasi kedua pendekatan tersebut, penelitian ini diharapkan dapat menghasilkan mekanik operasional game manajemen yang mampu merepresentasikan kondisi operasional jalan tol secara simulatif dengan tetap mempertahankan unsur gameplay yang logis, menantang, dan interaktif.

3.2 Tahapan Penelitian

Tahapan penelitian merupakan serangkaian langkah sistematis yang dilakukan untuk mencapai tujuan penelitian yang telah ditetapkan. Pada penelitian ini, proses pengembangan game simulasi pengelolaan jalan tol dilakukan menggunakan metode *Game Development Life Cycle* (GDLC) yang dimodifikasi untuk mendukung proses kerja iteratif. Berbeda dengan proses pembuatan proyek pada umumnya, penelitian ini tidak hanya berfokus pada proses implementasi game, tetapi juga pada proses analisis, pencarian solusi, evaluasi, dan perbaikan mekanik permainan secara berulang untuk mencapai *balancing gameplay* yang logis dan sesuai dengan tujuan permainan yang dirancang.

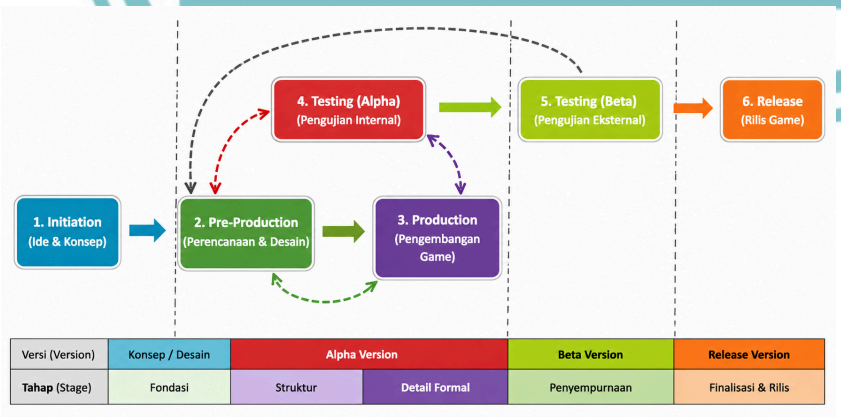
Penelitian ini menerapkan pendekatan *mixed method*, yaitu kombinasi antara pendekatan kualitatif dan kuantitatif. Pendekatan kualitatif digunakan pada tahap

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

awal penelitian untuk memperoleh data primer mengenai kondisi operasional jalan tol melalui wawancara dengan pihak PT Jasa Marga. Data tersebut digunakan sebagai dasar representasi dunia nyata ke dalam mekanik permainan, seperti penerapan sistem antrean kendaraan, pengelolaan gerbang tol, kondisi kepadatan lalu lintas, dan mekanisme penanganan gangguan operasional. Sementara itu, pendekatan kuantitatif digunakan pada tahap pengujian untuk mengevaluasi gameplay yang dikembangkan melalui *alpha testing* dan *expert testing* menggunakan instrumen kuesioner skala Likert. Pengujian dilakukan untuk mengevaluasi aspek gameplay, pemahaman pemain terhadap logika operasional permainan, tingkat ketertarikan untuk memainkan kembali permainan (*replayability*), serta kesediaan pemain untuk merekomendasikan permainan kepada pengguna lain.

Tahapan pengembangan dalam penelitian ini terdiri atas enam fase utama, yaitu *Initiation*, *Pre-production*, *Production*, *Testing*, *Beta Testing*, dan *Release*. Namun, dalam proses pengembangannya penelitian ini lebih difokuskan pada proses iterasi antara tahap *Production* dan *Testing* untuk melakukan evaluasi dan balancing terhadap mekanik operasional yang dikembangkan. Proses iterasi dilakukan melalui tahapan analisis masalah, pencarian solusi, implementasi perbaikan, dan evaluasi ulang terhadap gameplay yang dihasilkan. Penelitian ini berfokus pada pencapaian tahap balancing mekanik permainan dan belum membahas tahap *refinement* atau penyempurnaan akhir secara mendalam karena keterbatasan waktu penelitian.



Gambar 3.1 Struktur Game development Life Cycle

Gambar 3.1 menunjukkan alur *Game Development Life Cycle* (GDLC) yang digunakan dalam penelitian ini. Alur tersebut menekankan adanya proses loop atau perulangan antara tahap *Production* dan *Testing* untuk menyempurnakan desain mekanik permainan secara bertahap. Adapun penjelasan dari setiap tahapan penelitian adalah sebagai berikut:

3.2.1 Initiation

Tahap *Initiation* merupakan tahap awal dalam metode *Game Development Life Cycle* (GDLC) yang bertujuan untuk menentukan konsep dasar permainan yang akan dikembangkan. Pada tahap ini dilakukan proses *brainstorming* untuk mengidentifikasi permasalahan, menentukan tujuan pengembangan, serta mengumpulkan informasi yang dibutuhkan sebagai dasar perancangan mekanik permainan.

Proses inisiasi diawali dengan identifikasi masalah operasional jalan tol, seperti kepadatan pada gerbang tol dan ruas jalan akibat peningkatan volume kendaraan maupun gangguan perjalanan. Permasalahan utama yang diangkat dalam penelitian ini adalah bagaimana merepresentasikan kompleksitas operasional jalan tol ke dalam media simulasi yang interaktif dan mudah dipahami oleh pemain. Berdasarkan identifikasi tersebut, ditetapkan tujuan pengembangan game yaitu menghasilkan game simulasi yang mampu menghadirkan tiga pilar utama efektivitas simulasi, yaitu proses pengambilan keputusan (*decision making*), perubahan kondisi sistem secara dinamis, dan representasi dunia nyata (*realism*) ke dalam gameplay permainan.

Pada tahap ini juga dilakukan penentuan genre permainan, yaitu *simulation game* dengan elemen *management game*. Pemain berperan sebagai pengelola gerbang tol yang bertanggung jawab dalam menjaga kelancaran pelayanan kendaraan sesuai Standar Pelayanan Minimum (SPM). Sistem

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

permainan dirancang agar mampu menghadirkan kondisi dinamis seperti peningkatan volume kendaraan, antrean lalu lintas, kendaraan mogok, serta perubahan kondisi operasional yang memengaruhi jalannya permainan.

Untuk mendukung proses pengembangan, dilakukan studi literatur yang berkaitan dengan game simulasi, game manajemen, mekanik operasional dalam game, *balancing gameplay*, operasional jalan tol, dan metode pengembangan game menggunakan GDLC. Selain itu, dilakukan wawancara dengan pihak PT Jasa Marga untuk memperoleh informasi mengenai kondisi operasional jalan tol, sistem pengelolaan lalu lintas, serta mekanisme penanganan gangguan operasional yang dapat direpresentasikan ke dalam mekanik permainan. Selain itu, diperoleh pula informasi mengenai target pelayanan transaksi gerbang tol dan penggunaan fasilitas pendukung seperti kendaraan derek (*rescue*) untuk penanganan gangguan operasional.

Berdasarkan hasil studi literatur dan wawancara, ditentukan fitur-fitur utama yang akan diimplementasikan sebagai mekanik permainan, yaitu sistem pendapatan (*revenue*), sistem peningkatan (*upgrade*) gerbang tol, sistem *maintenance*, sistem *rescue*, pengelolaan antrean kendaraan, dan monitoring kemacetan. Sistem pendapatan digunakan sebagai sumber daya utama dalam permainan, sedangkan sistem *upgrade* digunakan untuk meningkatkan kapasitas pelayanan gerbang tol. Selain itu, sistem *maintenance* dan *rescue* digunakan untuk menangani gangguan operasional yang terjadi selama permainan berlangsung.

Tahap *Initiation* menghasilkan konsep awal permainan, kebutuhan sistem, gambaran mekanik operasional, serta dasar perancangan gameplay yang menekankan pada proses pengambilan keputusan, analisis kondisi sistem, dan evaluasi terhadap perubahan operasional secara dinamis. Pengembangan game dalam penelitian ini tidak hanya berfokus pada pembuatan produk permainan, tetapi juga pada proses penelitian dan evaluasi mekanik operasional secara terukur.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3.2.2 Pre-Production

Tahap *Pre-production* merupakan tahap perancangan yang dilakukan setelah konsep permainan ditentukan pada tahap *Initiation*. Pada tahap ini dilakukan penyusunan *Game Design Document* (GDD) sebagai dasar pengembangan game. Fokus utama tahap ini adalah menerjemahkan prosedur operasional jalan tol ke dalam bentuk logika sistem, algoritma, dan mekanik permainan yang akan mengatur gameplay di dalam game.

Perancangan gameplay dilakukan dengan menentukan peran pemain sebagai pengelola gerbang tol yang bertugas menjaga kelancaran lalu lintas dan mengelola sumber daya operasional. Mekanik permainan dirancang sebagai komponen yang mendorong pemain untuk melakukan tindakan tertentu, sedangkan aturan (*rules*) digunakan untuk menjelaskan mekanik tersebut ke dalam bentuk instruksi yang dapat dipahami pemain selama permainan berlangsung.

Pada tahap ini juga dilakukan perancangan mekanik operasional seperti sistem *vehicle spawning*, antrean kendaraan, sistem pendapatan (*revenue*), sistem *maintenance*, sistem *rescue*, dan monitoring kemacetan. Mekanik tersebut dirancang untuk merepresentasikan tiga pilar simulasi, yaitu representasi dunia nyata (*realism*), perubahan kondisi sistem secara dinamis (*dynamic system*), dan pengambilan keputusan (*decision making*).

Selain itu, dilakukan perancangan awal terhadap *balancing gameplay* dengan menentukan parameter permainan seperti tingkat kemunculan kendaraan, biaya operasional, durasi *rescue*, dan sistem pendapatan berdasarkan hasil observasi dan wawancara. Parameter awal tersebut digunakan sebagai dasar pengujian yang selanjutnya akan dievaluasi dan disesuaikan kembali melalui proses iterasi pada tahap *Production* dan *Testing*.

Tahap ini juga mencakup perancangan antarmuka pengguna (*User Interface*), flowchart, storyboard, struktur sistem, hubungan antar skrip,

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

serta kebutuhan aset visual dan audio yang akan digunakan dalam pengembangan game. Hasil dari tahap *Pre-production* berupa dokumen perancangan yang digunakan sebagai pedoman implementasi dan evaluasi mekanik permainan pada tahap berikutnya.

3.2.3 Production

Tahap *Production* merupakan tahap implementasi teknis dari seluruh rancangan yang telah disusun pada tahap *Pre-production*. Pada tahap ini dilakukan proses pembangunan game menggunakan Unity *Game Engine* dengan bahasa pemrograman C#. Dalam penelitian ini, tahap produksi tidak berjalan secara linear, melainkan melibatkan proses iterasi internal secara berulang antara implementasi fitur dan pengujian awal (*alpha testing*) untuk melakukan evaluasi dan perbaikan terhadap mekanik permainan.

1. Pembuatan Aset dan Integrasi Lingkungan

Proses produksi diawali dengan pengumpulan dan integrasi aset permainan yang merepresentasikan kondisi operasional jalan tol, seperti model gerbang tol, kendaraan, jalan, elemen *User Interface* (UI), ikon tombol, serta aset visual dan audio pendukung lainnya. Seluruh aset diintegrasikan ke dalam Unity untuk membangun lingkungan simulasi yang dinamis sesuai dengan kebutuhan sistem permainan.

2. Implementasi Logika Algoritma dan Mekanik

Pada tahap ini, prosedur operasional jalan tol yang diperoleh dari hasil observasi dan wawancara diterjemahkan ke dalam bentuk logika algoritma dan mekanik permainan. Implementasi dilakukan pada sistem *vehicle spawning*, pembentukan antrean kendaraan, pergerakan kendaraan menuju gerbang tol, sistem transaksi kendaraan, sistem pendapatan (*revenue*), sistem *maintenance*, sistem *rescue*, sistem *upgrade*, dan monitoring kemacetan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Sistem ekonomi permainan juga diterapkan di mana setiap kendaraan yang berhasil dilayani akan menghasilkan pendapatan yang dapat digunakan pemain untuk melakukan tindakan operasional tertentu. Seluruh sistem tersebut dirancang agar saling terhubung sehingga keputusan pemain akan memengaruhi kondisi permainan secara dinamis.

3. Implementasi Fitur Operasional dan Balancing Awal

Pada tahap ini dilakukan implementasi fitur operasional seperti *upgrade*, *maintenance*, dan *rescue* dengan mempertimbangkan proses *balancing gameplay*. Selama proses pengembangan, dilakukan evaluasi internal terhadap parameter permainan seperti tingkat kemunculan kendaraan, biaya operasional, durasi *rescue*, dan tingkat kesulitan permainan.

Beberapa parameter awal yang digunakan merupakan hasil observasi dan perkiraan awal yang kemudian dievaluasi kembali melalui proses iterasi selama pengembangan berlangsung. Sebagai contoh, durasi fitur *rescue* yang dirancang pada tahap awal mengalami penyesuaian setelah dilakukan pengujian internal karena dinilai terlalu lama dan memengaruhi ritme permainan. Penyesuaian parameter dilakukan agar gameplay tetap logis, dinamis, dan tidak menimbulkan rasa bosan bagi pemain.

4. Integrasi Sistem dan Antarmuka Pengguna

Pembangunan antarmuka pengguna (*User Interface*) seperti menu utama, menu pengaturan, halaman tutorial, tampilan gameplay, panel informasi pendapatan, indikator kondisi gerbang, tombol *upgrade*, tombol *maintenance*, tombol *rescue*, dan tampilan *game over*. Integrasi antara UI dan sistem gameplay dilakukan agar perubahan kondisi permainan dapat ditampilkan secara *real-time* kepada pemain sehingga pemain dapat melakukan analisis situasi sebelum mengambil keputusan operasional.

5. Build Sistem dan Alpha Version

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Setelah seluruh fitur berhasil diimplementasikan, dilakukan proses integrasi sistem secara menyeluruh dan proses *build* ke platform WebGL menggunakan Unity. Hasil dari tahap ini berupa versi awal (*alpha version*) game simulasi yang telah berjalan secara fungsional dan siap untuk memasuki tahap pengujian lebih lanjut pada tahap *Testing* dan *Beta Testing*.

3.2.4 Testing

Tahap *Testing* merupakan tahap pengujian internal yang dilakukan setelah seluruh fitur utama berhasil diimplementasikan pada tahap *Production*. Tujuan utama tahap ini adalah untuk memastikan setiap fungsi berjalan sesuai rancangan, mengidentifikasi kesalahan (*bug*), serta mengevaluasi kesesuaian mekanik permainan dengan logika operasional yang telah dirancang. Pada penelitian ini, tahap *Testing* tidak hanya berfokus pada validasi sistem, tetapi juga menjadi bagian dari proses belajar, analisis, evaluasi, dan perbaikan mekanik permainan melalui proses iterasi secara berulang.

Pengujian dilakukan menggunakan metode *Alpha Testing* dengan pendekatan *Black Box Testing*. Metode ini digunakan untuk menguji fungsionalitas sistem berdasarkan masukan (*input*) dan keluaran (*output*) tanpa memperhatikan struktur kode program yang digunakan. Pengujian dilakukan dengan menjalankan berbagai skenario penggunaan untuk memastikan seluruh fitur utama dapat berjalan sesuai tujuan gameplay yang telah dirancang.

Fitur-fitur yang diuji pada tahap ini meliputi sistem *vehicle spawning*, pergerakan dan antrean kendaraan, proses transaksi gerbang tol, sistem pendapatan (*revenue*), sistem *maintenance*, sistem *rescue*, sistem *upgrade*, tampilan antarmuka pengguna (*User Interface*), serta kondisi *game over*. Setiap hasil pengujian dicatat dan dianalisis untuk mengetahui keberhasilan implementasi sistem dan efektivitas mekanik permainan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Selain pengujian teknis, fokus utama tahap ini adalah melakukan proses iterasi desain untuk mencapai *balancing gameplay*. Proses iterasi dilakukan ketika ditemukan kondisi gameplay yang terlalu mudah, terlalu sulit, atau tidak berjalan sesuai logika operasional yang diharapkan. Peneliti kemudian melakukan analisis masalah, pencarian solusi, implementasi perbaikan, dan pengujian ulang terhadap sistem yang telah diperbaiki.

Dalam proses evaluasi mekanik permainan, digunakan beberapa acuan operasional berdasarkan hasil wawancara dengan PT Jasa Marga, seperti sistem pelayanan transaksi kendaraan, tingkat kepadatan lalu lintas, serta mekanisme penanganan gangguan menggunakan fitur *rescue* dan *maintenance*. Acuan tersebut digunakan untuk memastikan bahwa mekanik permainan tetap memiliki keterkaitan dengan kondisi operasional jalan tol yang direpresentasikan di dalam game.

Beberapa parameter permainan seperti tingkat kemunculan kendaraan, biaya operasional, durasi *rescue*, dan tingkat kesulitan permainan juga mengalami penyesuaian selama tahap pengujian berlangsung. Parameter awal yang digunakan pada tahap *Pre-production* dievaluasi kembali melalui proses *playtesting* internal untuk menemukan keseimbangan antara realisme simulasi dan pengalaman bermain yang tetap menarik bagi pemain.

Hasil dari tahap *Testing* berupa validasi fungsionalitas sistem dan versi awal (*alpha version*) game simulasi yang telah memiliki keseimbangan mekanik dasar dan siap untuk memasuki tahap *Beta Testing* guna memperoleh evaluasi dari penguji eksternal.

3.2.5 Beta Testing

Tahap *Beta Testing* merupakan tahap pengujian yang melibatkan penguji eksternal untuk memperoleh evaluasi terhadap gameplay dan mekanik operasional game yang telah dikembangkan. Berbeda dengan tahap *Testing*

yang berfokus pada validasi fungsionalitas sistem, tahap *Beta Testing* bertujuan untuk mengevaluasi pengalaman bermain, pemahaman pemain terhadap logika permainan, serta efektivitas mekanik operasional yang diterapkan di dalam game.

Pada penelitian ini, *Beta Testing* dilakukan dengan melibatkan penguji ahli game dan masyarakat umum sebagai responden. Penguji ahli game dilibatkan untuk memberikan evaluasi terhadap gameplay, balancing mekanik, dan desain permainan, sedangkan masyarakat umum dilibatkan untuk mengetahui tingkat pemahaman dan pengalaman bermain pengguna terhadap game yang dikembangkan.

Game yang telah lolos tahap *Alpha Testing* kemudian dimainkan oleh responden untuk memperoleh penilaian terhadap gameplay, mekanik permainan, dan sistem operasional yang terdapat di dalam game. Pengumpulan data dilakukan menggunakan kuesioner dengan skala Likert lima tingkat, yaitu Sangat Tidak Setuju, Tidak Setuju, Netral, Setuju, dan Sangat Setuju.

Aspek yang dievaluasi meliputi gameplay, kemudahan penggunaan (*usability*), tampilan antarmuka pengguna (*User Interface*), performa sistem, pemahaman pemain terhadap mekanik operasional, tingkat ketertarikan untuk memainkan kembali permainan (*replayability*), serta kesediaan responden untuk merekomendasikan permainan kepada pengguna lain.

Selain data kuantitatif, pada tahap ini juga diperoleh data kualitatif berupa saran, masukan, dan evaluasi dari penguji terhadap mekanik permainan dan balancing gameplay yang dikembangkan. Data tersebut digunakan sebagai bahan analisis dan evaluasi untuk mengetahui aspek permainan yang masih memerlukan perbaikan maupun penyempurnaan.

Data hasil pengujian kemudian dianalisis menggunakan metode deskriptif kuantitatif dan deskriptif kualitatif untuk mengetahui tingkat penerimaan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

responden terhadap game simulasi yang dikembangkan. Hasil dari tahap *Beta Testing* menjadi dasar dalam melakukan evaluasi akhir terhadap gameplay dan mekanik operasional sebelum memasuki tahap *Release*.

3.2.6 Release

Tahap *Release* merupakan tahap akhir dalam metode *Game Development Life Cycle* (GDLC) yang dilakukan setelah proses implementasi dan pengujian selesai dilaksanakan. Pada penelitian ini, tahap *Release* difokuskan pada finalisasi versi stabil game untuk kebutuhan dokumentasi penelitian dan validasi hasil pengembangan, serta belum mencakup tahap *refinement* atau penyempurnaan akhir secara menyeluruh maupun distribusi publik dalam skala luas.

Keterbatasan waktu penelitian selama satu semester menjadikan proses pengembangan pada penelitian ini lebih difokuskan pada pencapaian *balancing gameplay* melalui proses iterasi antara tahap *Production*, *Testing*, dan *Beta Testing*. Tahap Release dalam penelitian ini tidak berorientasi pada distribusi komersial, melainkan sebagai tahap akhir dokumentasi hasil implementasi mekanik operasional yang telah dikembangkan dan dievaluasi selama proses penelitian berlangsung.

Pada tahap ini dilakukan proses *build* game ke platform WebGL menggunakan Unity agar permainan dapat dijalankan melalui *web browser*. Hasil *build* kemudian dipublikasikan melalui platform Itch.io sehingga dapat digunakan untuk kebutuhan pengujian, validasi penelitian, dan dokumentasi hasil pengembangan game.

Versi game yang dihasilkan pada tahap ini merupakan hasil implementasi mekanik operasional yang telah melalui proses analisis, evaluasi, dan iterasi desain selama tahap pengembangan. Seluruh fitur utama seperti sistem antrean kendaraan, sistem pendapatan (*revenue*), sistem *upgrade*, sistem *maintenance*, sistem *rescue*, dan sistem *game over* telah terintegrasi dan dapat dijalankan sesuai dengan rancangan gameplay yang telah dibuat.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tahap *Release* menghasilkan versi stabil game simulasi pengelolaan jalan tol berbasis WebGL yang telah melalui proses implementasi dan evaluasi sesuai tahapan metode GDLC. Hasil akhir dari tahap ini berupa dokumentasi sistem dan gameplay yang telah tervalidasi sebagai bagian dari laporan proses pengembangan dan penelitian yang dilakukan.

3.3 Objek Penelitian

Objek penelitian pada penelitian ini adalah mekanik operasional pada game simulasi manajemen *Toll Tycoon* yang dikembangkan menggunakan *game engine* Unity berbasis WebGL. Penelitian berfokus pada perancangan dan implementasi sistem operasional jalan tol ke dalam bentuk gameplay interaktif yang melibatkan proses pengambilan keputusan pemain selama permainan berlangsung.

Objek yang diteliti meliputi beberapa mekanik utama permainan seperti sistem antrean kendaraan, pengelolaan gerbang tol, sistem pendapatan, maintenance, rescue, upgrade gerbang tol, dan monitoring kemacetan. Seluruh mekanik tersebut dirancang berdasarkan hasil wawancara, observasi, dan konsep operasional jalan tol dari PT Jasa Marga yang kemudian direpresentasikan ke dalam sistem permainan secara simulatif.

Penelitian juga berfokus pada implementasi balancing gameplay terhadap parameter permainan seperti tingkat kemunculan kendaraan, biaya operasional, sistem pendapatan, maintenance, dan rescue agar gameplay yang dihasilkan tetap menantang, dinamis, dan sesuai dengan konsep game simulasi manajemen yang dirancang.

3.4 Teknik Pengumpulan dan Analisis Data

Teknik pengumpulan dan analisis data digunakan untuk memperoleh informasi yang mendukung proses perancangan, pengembangan, serta evaluasi game simulasi pengelolaan gerbang tol yang dikembangkan dalam penelitian ini. Data yang diperoleh terdiri dari data kualitatif dan data kuantitatif yang dikumpulkan melalui beberapa metode sesuai dengan kebutuhan penelitian.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Data kualitatif digunakan untuk memahami proses operasional gerbang tol dan kebutuhan sistem yang akan direpresentasikan ke dalam permainan. Data kuantitatif digunakan untuk mengevaluasi kualitas game yang dikembangkan berdasarkan tanggapan pengguna pada tahap pengujian.

Data yang telah diperoleh selanjutnya dianalisis menggunakan metode analisis kualitatif dan analisis kuantitatif untuk menghasilkan informasi yang dapat digunakan sebagai dasar dalam penarikan kesimpulan penelitian.

3.4.1 Teknik Pengumpulan Data

Teknik pengumpulan data dilakukan untuk memperoleh informasi yang dibutuhkan dalam proses perancangan, pengembangan, dan evaluasi game simulasi pengelolaan gerbang tol. Data yang dikumpulkan pada penelitian ini terdiri atas data primer dan data sekunder.

Data primer diperoleh melalui wawancara dengan pihak PT Jasa Marga untuk memahami operasional jalan tol, pengelolaan lalu lintas, penyebab kepadatan kendaraan, penanganan gangguan operasional, serta mekanisme pengambilan keputusan di lapangan. Informasi tersebut digunakan sebagai dasar dalam merancang mekanik operasional dan representasi sistem ke dalam gameplay permainan.

Penelitian ini juga menggunakan data sekunder yang diperoleh melalui studi literatur dari buku, jurnal ilmiah, artikel penelitian, dan dokumen pendukung lainnya yang berkaitan dengan game simulasi, game manajemen, operasional jalan tol, sistem transportasi, *balancing gameplay*, dan metode *Game Development Life Cycle* (GDLC).

Pada tahap pengujian, pengumpulan data juga dilakukan melalui penyebaran kuesioner kepada penguji ahli game dan masyarakat umum pada tahap *Beta Testing*. Kuesioner digunakan untuk memperoleh data kuantitatif berupa penilaian terhadap gameplay, tampilan antarmuka, performa sistem, dan pengalaman bermain, serta data kualitatif berupa saran dan masukan terhadap mekanik permainan yang dikembangkan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Data yang diperoleh melalui proses pengumpulan data tersebut digunakan sebagai dasar dalam proses implementasi, evaluasi, dan balancing mekanik permainan selama penelitian berlangsung.

3.4.2 Teknik Analisis Data

Teknik analisis data digunakan untuk mengolah data yang telah diperoleh selama proses penelitian sehingga dapat menghasilkan informasi yang mendukung pencapaian tujuan penelitian. Pada penelitian ini digunakan dua teknik analisis data, yaitu analisis deskriptif kualitatif dan analisis deskriptif kuantitatif sesuai dengan pendekatan *mixed method* yang diterapkan.

1. Analisis Kualitatif

   Analisis kualitatif dilakukan terhadap data hasil wawancara dengan pihak PT Jasa Marga dan data saran dari ahli game pada tahap *Beta Testing*. Data hasil wawancara dianalisis secara deskriptif untuk memahami operasional jalan tol, pengelolaan lalu lintas, penanganan gangguan operasional, serta mekanisme pengambilan keputusan yang kemudian direpresentasikan ke dalam gameplay permainan.

   Data kualitatif dari ahli game digunakan untuk mengevaluasi aspek gameplay, balancing mekanik, alur permainan, dan pengalaman bermain yang terdapat pada game simulasi yang dikembangkan. Hasil analisis tersebut digunakan sebagai dasar dalam proses evaluasi dan perbaikan mekanik permainan selama tahap iterasi berlangsung.

2. Analisis Kuantitatif

   Analisis kuantitatif dilakukan terhadap data hasil *Beta Testing* yang diperoleh melalui penyebaran kuesioner kepada ahli game dan masyarakat umum menggunakan skala Likert lima tingkat, yaitu

Sangat Tidak Setuju (STS), Tidak Setuju (TS), Netral (N), Setuju (S), dan Sangat Setuju (SS).

Tabel 3.1 Skor Beta Testing

Jawaban	**Skor**
Sangat Tidak Setuju (STS)	1
Tidak Setuju (TS)	2
Netral (N)	3
Setuju (S)	4
Sangat Setuju (SS)	5

Data yang diperoleh kemudian dihitung menggunakan persamaan berikut:

$$Persentase \frac{Skor\ Total\ Aktual}{Skor\ Ideal\ Maksimum} \times 100\%$$

Hasil perhitungan persentase kemudian diinterpretasikan menggunakan kategori penilaian yang ditunjukkan pada Tabel 3.2.

Tabel 3.2 Interval Skala Likert

Persentase	**Kategori**
0% – 20%	Sangat Tidak Baik
21% – 40%	Tidak Baik
41% – 60%	Cukup
61% – 80%	Baik
81% – 100%	Sangat Baik

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Melalui analisis kuantitatif tersebut dapat diketahui tingkat penerimaan pengguna terhadap game simulasi pengelolaan gerbang tol yang telah dikembangkan serta kualitas game berdasarkan aspek kemudahan penggunaan, gameplay, tampilan antarmuka, performa sistem, dan kepuasan pengguna.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB IV

HASIL DAN PEMBAHASAN

4.1 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengidentifikasi kebutuhan sistem dan mekanik permainan yang akan diterapkan pada game simulasi pengelolaan gerbang tol. Proses analisis kebutuhan bertujuan untuk memastikan bahwa fitur dan gameplay yang dikembangkan mampu merepresentasikan kondisi operasional jalan tol serta mendukung proses pengambilan keputusan pemain dalam menghadapi kondisi lalu lintas yang dinamis.

Analisis kebutuhan pada penelitian ini dilakukan berdasarkan hasil wawancara dengan pihak PT Jasa Marga serta studi literatur yang berkaitan dengan operasional jalan tol, game simulasi, game manajemen, dan pengembangan game menggunakan metode *Game Development Life Cycle* (GDLC). Hasil analisis tersebut digunakan sebagai dasar dalam menentukan mekanik operasional, fitur permainan, serta kebutuhan teknis yang diperlukan dalam pengembangan game simulasi pengelolaan gerbang tol.

4.1.1 Analisis Hasil Wawancara

Wawancara dilakukan dengan pihak PT Jasa Marga untuk memperoleh informasi mengenai operasional jalan tol, penyebab kepadatan lalu lintas, prosedur penanganan gangguan, penggunaan teknologi pada gerbang tol, serta proses pengambilan keputusan yang dilakukan oleh pengelola jalan tol.

Berdasarkan hasil wawancara, diketahui bahwa tujuan utama pengelolaan jalan tol adalah menjaga kelancaran, keamanan, dan kenyamanan pengguna jalan. Pengelola jalan tol bertanggung jawab untuk memastikan lalu lintas tetap berjalan dengan lancar serta melakukan penanganan

terhadap berbagai gangguan yang dapat memengaruhi mobilitas pengguna jalan.

Hasil wawancara menunjukkan bahwa kepadatan lalu lintas pada ruas Tol Jagorawi umumnya terjadi pada akhir pekan, hari libur panjang, serta jam sibuk pada hari kerja. Kondisi tersebut disebabkan oleh meningkatnya volume kendaraan yang melintas, terutama pada jalur yang mengarah ke kawasan wisata maupun pusat aktivitas kerja. Selain itu, kepadatan juga dapat terjadi akibat gangguan kendaraan, kecelakaan lalu lintas, maupun kegiatan perawatan (*maintenance*) yang dilakukan pada ruas jalan tol.

Menurut narasumber, gangguan kendaraan seperti kendaraan mogok dan kerusakan kendaraan berat merupakan salah satu faktor yang sulit diprediksi dan berpotensi menyebabkan kemacetan yang cukup signifikan. Oleh karena itu, diperlukan tindakan penanganan yang cepat agar gangguan tersebut tidak menghambat kendaraan lain yang sedang melintas.

Selain itu, teknologi yang digunakan pada gerbang tol juga memiliki pengaruh terhadap panjang antrean kendaraan. Semakin cepat proses transaksi kendaraan dilakukan, maka semakin kecil kemungkinan terjadinya antrean yang panjang pada gerbang tol. Oleh karena itu, peningkatan teknologi pelayanan menjadi salah satu faktor penting dalam operasional gerbang tol.

Hasil wawancara juga menunjukkan bahwa dalam operasional jalan tol terdapat berbagai fasilitas pendukung seperti kendaraan patroli, kendaraan derek, ambulans, dan kendaraan *rescue* yang digunakan untuk menangani gangguan lalu lintas maupun keadaan darurat. Keberadaan fasilitas tersebut menjadi bagian penting dalam menjaga kelancaran operasional jalan tol.

Berdasarkan hasil wawancara tersebut, diperoleh beberapa kebutuhan sistem yang kemudian diimplementasikan ke dalam game simulasi

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

pengelolaan gerbang tol. Hubungan antara hasil wawancara dan implementasi sistem dapat dilihat pada Tabel 4.1.

Tabel 4.1 Hasil Wawancara dan Implementasi

No	Temuan Hasil Wawancara	Implementasi Pada Game
1.	Kepadatan lalu lintas terjadi akibat tingginya volume kendaraan pada waktu tertentu.	Sistem spawn kendaraan dan sistem antrean kendaraan.
2.	Kendaraan mogok dan gangguan perjalanan dapat menyebabkan kemacetan.	Sistem rescue.
3.	Maintenance dapat menyebabkan gangguan operasional dalam jangka waktu tertentu.	Sistem maintenance.
4.	Teknologi gerbang tol berpengaruh terhadap kecepatan pelayanan kendaraan.	Sistem upgrade gerbang tol.
5.	Kelancaran lalu lintas menjadi tujuan utama operasional jalan tol.	Sistem monitoring kemacetan dan game over.

4.1.2 Analisis Kebutuhan Fungsional

Kebutuhan fungsional merupakan kebutuhan yang menjelaskan fungsi-fungsi yang harus dimiliki oleh sistem agar dapat berjalan sesuai dengan tujuan yang telah ditetapkan. Kebutuhan fungsional pada game simulasi pengelolaan gerbang tol dapat dilihat pada Tabel 4.2.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 4.2 Daftar Kebutuhan Fungsional

Kebutuhan Fungsional	Deskripsi
Sistem Menu Utama	Menampilkan menu utama dan navigasi permainan.
Sistem Pengaturan	Mengatur suara dan pengaturan permainan.
Sistem Tutorial	Menampilkan panduan bermain kepada pemain.
Sistem Spawn Kendaraan	Memunculkan kendaraan secara otomatis pada interval tertentu.
Sistem Pergerakan Kendaraan	Mengatur pergerakan kendaraan menuju gerbang tol.
Sistem Antrean Kendaraan	Mengatur antrean kendaraan saat terjadi kepadatan.
Sistem Pendapatan	Menambahkan saldo pemain dari kendaraan yang berhasil dilayani.
Sistem Upgrade Gerbang Tol	Meningkatkan kemampuan gerbang tol menggunakan saldo pemain.
Sistem Maintenance	Memulihkan kondisi gerbang tol yang mengalami penurunan performa.
Sistem Rescue	Memberikan bantuan operasional untuk mengurangi dampak kemacetan.
Sistem Monitoring Kemacetan	Memantau tingkat kemacetan selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Sistem Timer Permainan	Mengatur durasi permainan.
Sistem Game Over	Mengakhiri permainan ketika kemacetan tidak dapat ditangani oleh pemain.

4.1.3 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan kebutuhan yang berkaitan dengan perangkat keras dan perangkat lunak yang digunakan dalam proses pengembangan dan pengoperasian game. Kebutuhan ini diperlukan untuk memastikan bahwa sistem dapat dikembangkan dan dijalankan dengan baik.

Kebutuhan non-fungsional pada penelitian ini meliputi perangkat keras yang digunakan dalam proses pengembangan game, perangkat lunak yang digunakan untuk membangun sistem, serta platform yang digunakan untuk mendistribusikan game kepada pengguna.

Tabel 4.3 Kebutuhan Perangkat Keras

No	Komponen	Spesifikasi
1	Processor	AMD Ryzen 5 4600H
2	RAM	8GB DDR4 3200MHz
3	Storage	512GB M.2 NVMe PCIe 3.0 SSD

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4	GPU	NVIDIA GeForce GTX 1650 Ti 4GB GDDR6
5	Monitor	15.6-inch Full HD

Tabel 4.4 Kebutuhan Perangkat Lunak

No	Perangkat Lunak	Fungsi
1	Unity 6	Pengembangan game
2	Visual Studio Code	Penulisan kode C#
3	Google Chrome	Pengujian WebGL
4	Itch.io	Distribusi game
5	Google Docs	Penyusunan laporan

4.2 Perancangan Sistem

Perancangan sistem dilakukan berdasarkan hasil analisis kebutuhan yang telah diperoleh pada tahap sebelumnya. Tahap ini bertujuan untuk menggambarkan alur permainan, antarmuka pengguna, serta aset yang digunakan dalam pengembangan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

game simulasi pengelolaan gerbang tol. Hasil perancangan menjadi acuan dalam proses implementasi sistem pada tahap pengembangan game.

Perancangan sistem pada penelitian ini meliputi perancangan alur permainan menggunakan flowchart, penyusunan storyboard, perancangan antarmuka pengguna (*User Interface*), serta pengumpulan aset visual dan audio (*material collection*) yang digunakan dalam pengembangan game.

4.2.1 Flowchart Sistem

Flowchart sistem digunakan untuk menggambarkan alur permainan pada game simulasi pengelolaan gerbang tol yang dikembangkan. Flowchart ini menunjukkan proses yang dilalui pemain mulai dari memasuki permainan hingga mencapai kondisi akhir permainan.

Berdasarkan Gambar 4.1, permainan diawali pada halaman utama (*Home Page*). Pada halaman ini pemain dapat memilih untuk memulai permainan, membuka menu pengaturan (*settings*), atau keluar dari permainan (*exit*). Apabila pemain memilih keluar dari permainan, maka sistem akan menutup aplikasi. Sebaliknya, apabila pemain memilih memulai permainan, maka sistem akan mengarahkan pemain menuju halaman *Map Gameplay*.

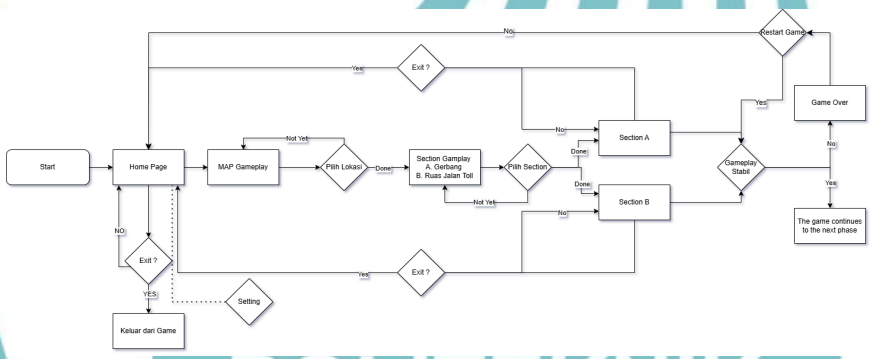
Pada halaman *Map Gameplay*, pemain diminta untuk memilih lokasi permainan yang tersedia. Setelah lokasi dipilih, pemain akan masuk ke bagian gameplay utama yang terdiri atas beberapa section permainan. Pemain kemudian dapat memilih section yang ingin dimainkan sesuai dengan alur permainan yang telah disediakan.

Setiap section memiliki tujuan dan tantangan yang harus diselesaikan oleh pemain. Setelah pemain menyelesaikan suatu section, sistem akan melakukan pemeriksaan terhadap kondisi gameplay untuk menentukan apakah permainan masih berada dalam kondisi stabil atau tidak. Apabila kondisi gameplay masih stabil, maka permainan akan berlanjut ke fase

berikutnya. Namun apabila kondisi gameplay tidak stabil, maka sistem akan mengarahkan pemain ke kondisi *Game Over*.

Pada kondisi *Game Over*, pemain diberikan pilihan untuk memulai permainan kembali (*Restart Game*) atau mengakhiri permainan. Jika pemain memilih untuk memulai kembali permainan, maka sistem akan mengarahkan pemain kembali ke awal permainan sehingga permainan dapat dimainkan kembali dari awal.

Flowchart sistem ini digunakan sebagai acuan dalam proses implementasi game karena menggambarkan alur interaksi pemain dengan sistem secara keseluruhan.



Gambar 4.1 Flowchart Game Toll Tycoon

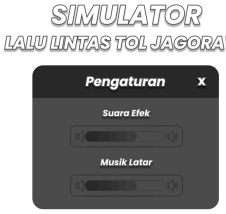
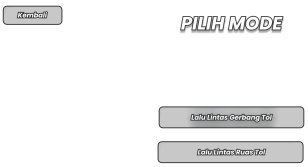
4.2.2 Storyboard

Storyboard digunakan untuk menggambarkan rancangan tampilan dan alur interaksi yang terdapat pada game simulasi lalu lintas Tol Jagorawi. Storyboard berfungsi sebagai acuan dalam proses implementasi antarmuka pengguna (*User Interface*) dan membantu menggambarkan pengalaman yang akan diperoleh pemain selama memainkan game. Rancangan storyboard pada game simulasi lalu lintas Tol Jagorawi dapat dilihat pada Tabel 4.5.

Tabel 4.5 Storyboard Game

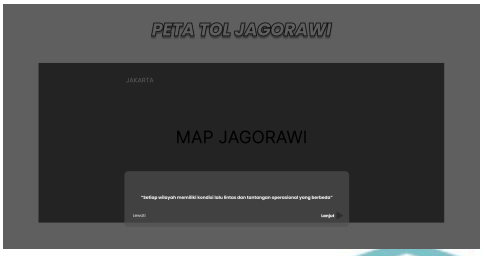
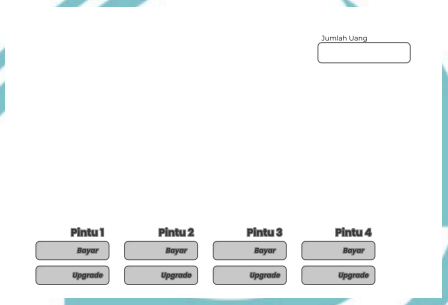
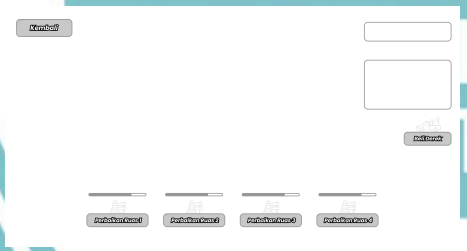
No	Scene	Deskripsi

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

1.		Halaman utama yang ditampilkan ketika permainan dijalankan. Pada halaman ini tersedia tombol Start Game, Settings, dan Exit.
2.		Halaman yang digunakan pemain untuk mengatur volume suara efek dan musik latar permainan.
3.		Halaman yang menampilkan peta ruas Tol Jagorawi beserta titik lokasi yang dapat dipilih oleh pemain sebagai area permainan.
4.		Halaman yang digunakan pemain untuk memilih jenis simulasi yang akan dimainkan, yaitu simulasi lalu lintas gerbang tol atau simulasi lalu lintas ruas tol.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

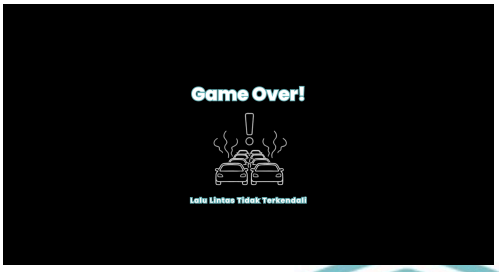
5.		Halaman yang menampilkan informasi dan petunjuk permainan sebelum pemain memasuki gameplay utama.
6.		Halaman permainan utama yang berfokus pada pengelolaan gerbang tol. Pemain bertugas mengatur operasional gerbang, melakukan upgrade, serta mengelola arus kendaraan untuk menjaga kelancaran lalu lintas.
7.		Halaman permainan yang berfokus pada pengelolaan kondisi ruas jalan tol. Pemain dapat melakukan perbaikan ruas jalan dan menggunakan fasilitas pendukung untuk mengurangi dampak gangguan lalu lintas.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

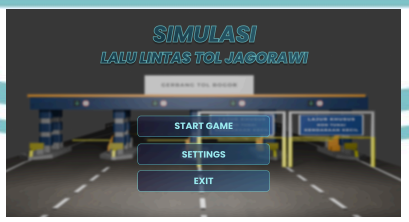
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

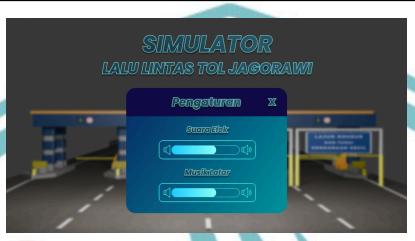
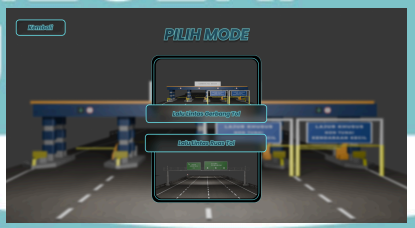
8.		Halaman yang muncul ketika kondisi lalu lintas tidak dapat dikendalikan sehingga permainan berakhir. Halaman ini menampilkan informasi kegagalan pemain dalam menjaga kelancaran lalu lintas.

4.2.3 Perancangan Antarmuka Pengguna (UI Design)

Perancangan antarmuka pengguna (User Interface) dilakukan untuk memberikan gambaran tampilan yang akan digunakan pada game simulasi lalu lintas Tol Jagorawi. Setiap antarmuka dirancang untuk mendukung interaksi pemain selama permainan berlangsung. Adapun rancangan antarmuka yang digunakan dapat dilihat pada Tabel 4.6.

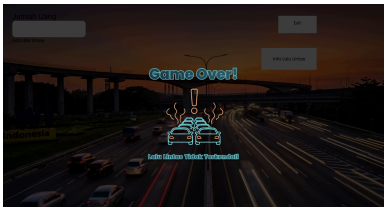
Tabel 4.6 Antarmuka Game

No	Nama Antarmuka	Gambar	Deskripsi
1.	Home Page		Halaman utama yang ditampilkan saat permainan dijalankan. Terdapat tombol Start Game, Settings,

			dan Exit sebagai navigasi utama permainan.
2.	Pengaturan		Halaman yang digunakan pemain untuk mengatur volume suara efek dan musik latar permainan.
3.	Peta Tol Jagorawi		Menampilkan peta Tol Jagorawi yang digunakan pemain untuk memilih lokasi permainan.
4.	Pilih Mode		Menampilkan pilihan mode permainan yaitu Lalu Lintas Gerbang Tol dan Lalu Lintas Ruas Tol.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

5.	Tutorial		Menampilkan informasi dan petunjuk permainan sebelum pemain memasuki gameplay utama.
6.	Gameplay Gerbang Tol		Menampilkan area permainan pengelolaan gerbang tol, termasuk sistem pendapatan dan upgrade gerbang.
7.	Gameplay Ruas Tol		Menampilkan area permainan pengelolaan ruas jalan tol yang dilengkapi fitur perbaikan ruas dan derek (rescue).
8.	Game Over		Ditampilkan ketika pemain gagal mengendalikan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

			lalu lintas sehingga permainan berakhir.

4.2.4 Material Collection

Material collection merupakan tahap pengumpulan aset yang digunakan dalam pengembangan game simulasi lalu lintas Tol Jagorawi. Aset yang dikumpulkan meliputi aset visual dan aset audio yang digunakan untuk mendukung tampilan serta pengalaman bermain pemain selama permainan berlangsung.

Tabel 4.7 Material Collection

No	Aset	Fungsi	Sumber
1.	Gerbang Tol	Objek utama pada mode simulasi gerbang tol	Dibuat oleh tim pengembang (Model 3D)
2.	Jalan Tol	Lingkungan utama permainan	Dibuat oleh tim pengembang (Model 3D)
3.	Kendaraan Mobil	Kendaraan yang melintas pada area permainan	Unity Asset Store
4.	Kendaraan Truk	Kendaraan berat yang digunakan dalam simulasi	Unity Asset Store
5.	Terrain	Membentuk lingkungan dan kontur area permainan	Unity Asset Store
6.	Peta Tol Jagorawi	Media pemilihan lokasi permainan	Dibuat oleh tim pengembang

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

7.	Tombol UI	Navigasi antarmuka permainan	Dibuat oleh tim pengembang
8.	Ikon Upgrade	Representasi fitur peningkatan gerbang tol	Dibuat oleh tim pengembang
9.	Ikon Maintenance	Representasi fitur perawatan gerbang tol	Dibuat oleh tim pengembang
10.	Ikon Rescue/Derek	Representasi fitur bantuan operasional	Dibuat oleh tim pengembang
11.	Background UI	Latar belakang antarmuka permainan	Dibuat oleh tim pengembang
12.	Background Music Menu	Musik latar pada game	Pixabay.com (bebas lisensi)
13.	Efek Suara	Efek suara untuk setiap kondisi pada game	Pixabay.com (bebas lisensi)

4.3 Implementasi Sistem

Implementasi sistem merupakan tahap penerapan hasil perancangan yang telah dibuat pada tahap sebelumnya ke dalam bentuk aplikasi game yang dapat dijalankan dan digunakan oleh pengguna. Pada penelitian ini, implementasi dilakukan menggunakan *game engine* Unity 6 dengan bahasa pemrograman C#. Seluruh fitur yang telah dirancang pada tahap perancangan sistem diimplementasikan menjadi game simulasi lalu lintas Tol Jagorawi yang dapat dimainkan melalui platform WebGL.

Implementasi sistem dilakukan dengan mengintegrasikan berbagai komponen yang telah dirancang sebelumnya, seperti antarmuka pengguna, sistem perpindahan antar scene, sistem lalu lintas kendaraan, sistem pengelolaan gerbang

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

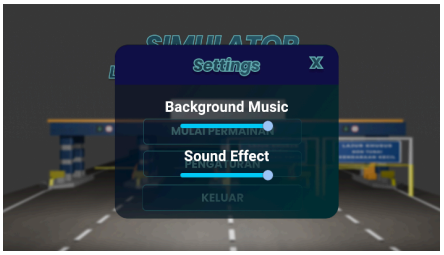
tol, sistem pendapatan, sistem maintenance, sistem rescue, sistem monitoring kemacetan, serta sistem game over. Implementasi tersebut bertujuan untuk menghasilkan simulasi yang mampu merepresentasikan kondisi operasional lalu lintas jalan tol secara interaktif.

4.3.1 Implementasi Antarmuka Pengguna

Implementasi antarmuka pengguna (*User Interface*) dilakukan berdasarkan rancangan antarmuka yang telah dibuat pada tahap perancangan sistem. Antarmuka pengguna berfungsi sebagai media interaksi antara pemain dengan sistem selama permainan berlangsung. Melalui antarmuka yang disediakan, pemain dapat mengakses berbagai fitur permainan, memperoleh informasi terkait kondisi permainan, serta melakukan pengambilan keputusan dalam simulasi lalu lintas Tol Jagorawi.

Implementasi antarmuka pengguna pada game simulasi lalu lintas Tol Jagorawi terdiri atas halaman utama (*Home Page*), halaman pengaturan (*Settings*), halaman peta Tol Jagorawi, halaman pemilihan mode permainan, halaman tutorial, halaman gameplay, dan halaman *Game Over*.

Tabel 4.8 Implementasi Antarmuka Pada Unity

Antarmuka	Hasil
	

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

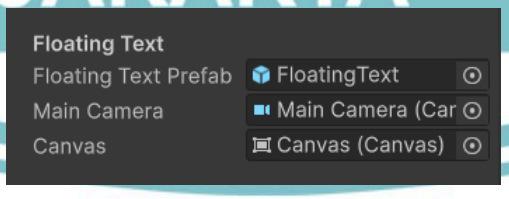
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Berdasarkan hasil implementasi yang telah dilakukan, seluruh antarmuka pengguna berhasil diterapkan sesuai dengan rancangan yang telah dibuat pada tahap perancangan sistem. Antarmuka yang diimplementasikan mampu mendukung interaksi pemain dengan sistem serta memberikan informasi yang dibutuhkan selama permainan berlangsung.

4.3.1.3 Implementasi Visual Feedback

Sistem visual feedback digunakan untuk memberikan respon visual terhadap berbagai kondisi dan tindakan selama permainan berlangsung agar pemain dapat memahami kondisi permainan dengan lebih cepat melalui informasi yang tampil secara langsung pada area permainan. Sebelum implementasi dilakukan, permainan belum memiliki sistem umpan balik visual sehingga kondisi seperti penambahan uang, penalti kendaraan, upgrade gerbang tol, dan kemacetan sulit dipahami pemain secara langsung serta membuat interaksi permainan terasa kurang responsif. Oleh karena itu, pada penelitian ini diterapkan sistem visual feedback menggunakan script FloatingText pada Unity untuk menampilkan informasi secara dinamis pada berbagai kondisi gameplay.

Gambar 4.2 menunjukkan implementasi sistem *visual feedback* pada Unity.



Gambar 4.2 Implementasi Visual Feedback pada unity

Pada implementasinya, sistem akan menampilkan teks informasi yang bergerak ke atas secara otomatis selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
0 references
void Update()
{
    // 🔥 POP SCALE
    transform.localScale = Vector3.Lerp(transform.localScale, targetScale, popSpeed * Time.deltaTime);
    // gerak naik
    transform.Translate(Vector3.up * moveSpeed * Time.deltaTime);

    // fade out
    canvasGroup.alpha -= Time.deltaTime / fadeDuration;

    // destroy kalau udah hilang
    if (canvasGroup.alpha <= 0)
    {
        Destroy(gameObject);
    }
}
```

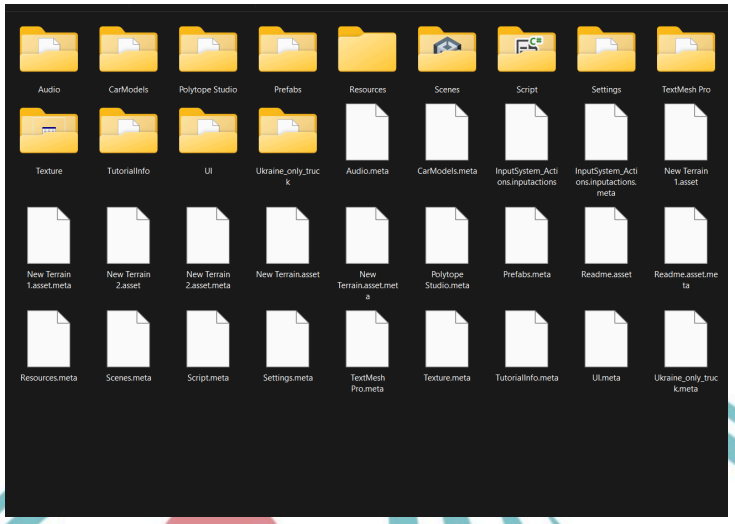
Gambar 4.3 Potongan kode floating text

Potongan kode tersebut digunakan untuk menjalankan animasi floating text secara dinamis selama permainan berlangsung dengan memberikan efek pembesaran teks saat muncul, bergerak ke atas secara bertahap, dan menghilang menggunakan efek fade out. Implementasi visual feedback ini digunakan pada berbagai kondisi gameplay seperti penambahan pendapatan kendaraan, penalti kemacetan, kondisi game over, proses upgrade gerbang tol, dan sistem peringatan lalu lintas. Penerapan sistem tersebut membantu pemain memahami perubahan kondisi permainan secara lebih cepat dan responsif tanpa harus terus memperhatikan seluruh indikator antarmuka selama simulasi berlangsung.

4.3.2 Struktur Implementasi Sistem

Implementasi sistem pada game simulasi pengelolaan gerbang tol dilakukan menggunakan Unity *Game Engine* dengan bahasa pemrograman C#. Dalam proses pengembangan, aset dan komponen sistem disusun secara terstruktur untuk memudahkan implementasi, pengujian, serta proses iterasi gameplay selama pengembangan berlangsung.

Penyusunan aset pada Unity dibagi ke dalam beberapa folder utama, seperti folder *Scripts* untuk script mekanik permainan, folder *Scenes* untuk pengelolaan scene permainan, folder *Prefabs* untuk objek permainan yang dapat digunakan kembali, folder *UI* untuk elemen antarmuka pengguna, serta folder *Audio* dan *Texture* untuk kebutuhan suara dan visual permainan.



Gambar 4.4 Struktur Manajemen File

Gambar 4.5 menunjukkan struktur penyusunan aset dan komponen sistem pada proyek Unity yang digunakan dalam pengembangan game simulasi pengelolaan gerbang tol.

4.3.3 Implementasi Sistem Spawn Kendaraan

Sistem spawn kendaraan merupakan salah satu komponen utama dalam game *Toll Tycoon* yang berfungsi untuk menghasilkan kendaraan secara otomatis pada gerbang tol maupun ruas jalan tol. Sistem ini dirancang untuk menciptakan simulasi lalu lintas yang dinamis sehingga jumlah dan variasi kendaraan yang muncul dapat memberikan tantangan bagi pemain dalam mengelola operasional jalan tol.

Implementasi sistem spawn kendaraan menggunakan mekanisme berbasis waktu (*timer-based spawning*) yang dipadukan dengan pemilihan kendaraan secara acak (*randomized vehicle selection*). Selain itu, sistem juga dilengkapi dengan mekanisme pemeriksaan area spawn (*safety-based spawn checking*) untuk mencegah kendaraan muncul pada posisi yang telah ditempati kendaraan lain. Pada kondisi tertentu, seperti *Rush Hour*, sistem akan menyesuaikan parameter spawn sehingga tingkat kepadatan lalu lintas meningkat sesuai kondisi permainan. Implementasi dari masing-masing mekanisme tersebut dijelaskan pada subbab berikutnya.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3.3.1 Implementasi Spawn Kendaraan Pada Gerbang Tol

Sistem spawn kendaraan pada gerbang tol bertugas menghasilkan kendaraan secara otomatis di setiap jalur masuk untuk mensimulasikan arus lalu lintas yang dinamis, sehingga pemain harus mengelola kapasitas gerbang melalui upgrade atau pengaturan operasional agar antrean terkendali. Sebelum memunculkan kendaraan, sistem memeriksa area spawn untuk mencegah tabrakan atau penumpukan, serta menyesuaikan interval spawn secara otomatis berdasarkan kepadatan lalu lintas guna menjaga keseimbangan dan mendukung pengalaman bermain yang realistis.

Sistem spawn kendaraan pada gerbang tol menggunakan algoritma Timer-Based Spawning sebagai mekanisme utama untuk menghasilkan kendaraan berdasarkan interval waktu tertentu sehingga frekuensi kemunculan kendaraan dapat dikontrol secara konsisten. Dalam implementasinya, algoritma ini didukung oleh Randomized Vehicle Selection untuk memilih jenis kendaraan secara acak, Safety-Based Spawn Checking untuk memastikan area spawn dalam kondisi aman sebelum kendaraan dibuat, serta Adaptive Spawn Interval yang menyesuaikan interval spawn berdasarkan tingkat kepadatan kendaraan di sekitar gerbang tol. Kombinasi keempat mekanisme tersebut memungkinkan sistem menghasilkan arus kendaraan yang dinamis, mengurangi potensi tabrakan pada titik spawn, serta menjaga keseimbangan tingkat kesulitan permainan.

Proses spawn kendaraan pada gerbang tol diawali dengan menjalankan *coroutine* yang bekerja secara terus-menerus selama permainan berlangsung. Sistem akan menunggu hingga interval waktu yang telah ditentukan, kemudian melakukan pemeriksaan terhadap area spawn untuk memastikan tidak terdapat kendaraan lain yang dapat mengganggu proses kemunculan kendaraan.

Apabila area spawn dalam kondisi aman, sistem akan memilih salah satu prefab kendaraan secara acak dan membuat objek kendaraan baru pada titik spawn yang telah ditentukan. Setelah proses spawn selesai, sistem menghitung kembali tingkat kepadatan kendaraan di sekitar gerbang tol untuk menyesuaikan interval spawn berikutnya sehingga laju kemunculan kendaraan tetap seimbang sesuai kondisi lalu lintas dalam permainan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
IEnumerator SpawnWithDetection()
{
    if (showDebugLogs)
        Debug.Log("🚗 SpawnWithDetection coroutine started");

    while (isActive)
    {
        bool safeToSpawn = IsSafeToSpawn();

        // DETEKSI JUMLAH MOBIL DEKAT SPAWN
        int nearbyCars = Physics.OverlapSphere(
            spawnPoint.position,
            detectionDistance,
            carLayer
        ).Length;

        // BASE SPAWN
        currentSpawnInterval =
            Random.Range(minSpawnInterval, maxSpawnInterval);

        // TRAFFIC ADAPTIVE SYSTEM
        if (nearbyCars >= 2)
        {
            currentSpawnInterval += 1f;
        }

        if (nearbyCars >= 4)
        {
            currentSpawnInterval += 2f;
        }

        if (safeToSpawn)
        {
            spawnBlockedPenaltyGiven = false;

            SpawnCar();
        }
        else
        {
            currentSpawnInterval += 1.5f;

            if (!spawnBlockedPenaltyGiven)
            {
                TollGate gate =
                    GetComponentInParent<TollGate>();

                if (gate != null)
                {
                    gate.ApplyPenalty();
                }

                spawnBlockedPenaltyGiven = true;
            }
        }

        yield return new WaitForSeconds(currentSpawnInterval);
    }

    if (showDebugLogs)
        Debug.Log("🚗 SpawnWithDetection coroutine ended");

    spawnCoroutine = null;
}
```

Gambar 4.5 *Script Timer-Based Spawning*

Gambar 4.5 menunjukkan implementasi algoritma *Timer-Based Spawning* menggunakan *coroutine* pada Unity. *Coroutine* dijalankan secara terus-menerus selama permainan berlangsung dengan memberikan jeda berdasarkan nilai currentSpawnInterval sebelum menghasilkan kendaraan baru. Penggunaan mekanisme ini

memungkinkan proses spawn dilakukan secara periodik tanpa dijalankan pada setiap *frame*, sehingga performa permainan tetap terjaga meskipun jumlah kendaraan yang aktif terus bertambah.

```
2 references
public bool IsSafeToSpawn()
{
    if (!isActive)
        return false;

    if (spawnPoint == null)
        return false;

    Vector3 center =
        spawnPoint.position + Vector3.up * 0.5f;

    Vector3 halfExtents =
        new Vector3(1.5f, 1f, 2f);

    Collider[] hits =
        Physics.OverlapBox(
            center +
            spawnPoint.forward * 2f,
            halfExtents,
            spawnPoint.rotation,
            carLayer
        );

    foreach (Collider hit in hits)
    {
        if (hit.CompareTag("Car"))
        {
            return false;
        }
    }

    return true;
}
```

Gambar 4.6 *Script Safety-Based Spawn Checking*

Gambar 4.6 menunjukkan implementasi *Safety-Based Spawn Checking* yang dilakukan sebelum kendaraan dibuat. Sistem melakukan pemeriksaan terhadap area di sekitar titik spawn menggunakan fungsi deteksi fisika untuk memastikan tidak terdapat kendaraan lain yang berada pada lokasi tersebut. Apabila area masih ditempati kendaraan, proses spawn akan ditunda hingga area kembali kosong. Mekanisme ini bertujuan mencegah terjadinya tabrakan maupun penumpukan kendaraan pada titik spawn sehingga simulasi lalu lintas dapat berjalan lebih stabil.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

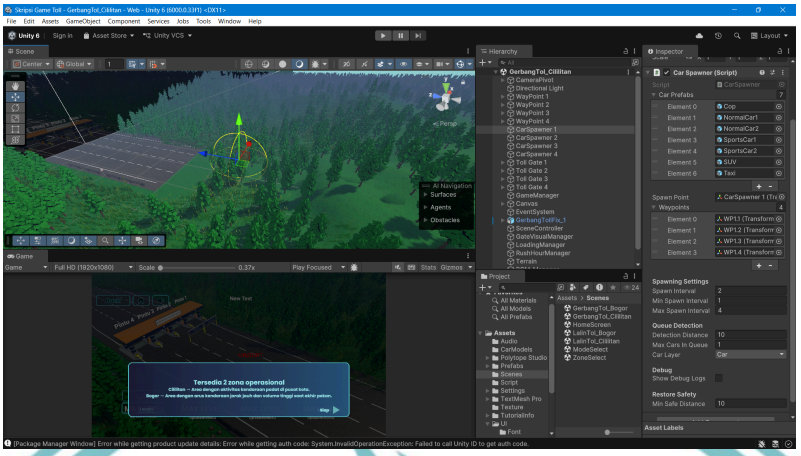
```
// DETEKSI JUMLAH MOBIL DEKAT SPAWN
int nearbyCars = Physics.OverlapSphere(
    spawnPoint.position,
    detectionDistance,
    carLayer
).Length;

// BASE SPAWN
currentSpawnInterval =
    Random.Range(minSpawnInterval, maxSpawnInterval);

// TRAFFIC ADAPTIVE SYSTEM
if (nearbyCars >= 2)
{
    currentSpawnInterval += 1f;
}

if (nearbyCars >= 4)
{
    currentSpawnInterval += 2f;
}

if (safeToSpawn)
{
    spawnBlockedPenaltyGiven = false;

    SpawnCar();
}
```

Gambar 4.7 *Script Adaptive Spawn Interval*

Gambar 4.7 menunjukkan implementasi *Adaptive Spawn Interval* yang berfungsi menyesuaikan interval kemunculan kendaraan berdasarkan tingkat kepadatan lalu lintas di sekitar gerbang tol. Sistem menghitung jumlah kendaraan yang berada dalam radius tertentu, kemudian mengubah nilai currentSpawnInterval secara dinamis. Semakin tinggi kepadatan kendaraan, semakin panjang interval spawn yang diterapkan sehingga laju kemunculan kendaraan dapat dikendalikan. Pendekatan ini membantu menjaga keseimbangan permainan serta mengurangi risiko terbentuknya antrean kendaraan yang berlebihan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.8 Implementasi Script *Spawn* Pada Unity

Pada tahap awal pengembangan, sistem spawn kendaraan menggunakan interval tetap sehingga kendaraan terus muncul dengan kecepatan yang sama tanpa mempertimbangkan kondisi lalu lintas di sekitar gerbang tol. Hasil pengujian menunjukkan bahwa pendekatan tersebut menyebabkan antrean kendaraan terbentuk terlalu cepat ketika jumlah kendaraan meningkat, sehingga tingkat kesulitan permainan menjadi kurang seimbang. Berdasarkan hasil evaluasi tersebut, sistem kemudian dikembangkan dengan menerapkan Adaptive Spawn Interval, yaitu mekanisme yang menyesuaikan interval spawn berdasarkan jumlah kendaraan di sekitar area spawn. Perubahan ini menghasilkan arus kendaraan yang lebih stabil, mengurangi risiko penumpukan kendaraan, serta memberikan pengalaman bermain yang lebih seimbang bagi pemain.

4.3.3.2 Implementasi Spawn Kendaraan Ruas Jalan Tol

Sistem spawn kendaraan pada ruas jalan tol berfungsi menghasilkan kendaraan secara otomatis pada beberapa titik spawn yang tersebar di sepanjang jalan tol. Berbeda dengan sistem spawn pada gerbang tol yang berfokus pada simulasi kendaraan memasuki jalan tol, sistem ini bertujuan menjaga arus lalu lintas tetap aktif pada setiap ruas jalan sehingga lingkungan permainan terlihat lebih dinamis. Dengan adanya kendaraan yang muncul dari berbagai

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

titik, pemain dihadapkan pada kondisi lalu lintas yang lebih bervariasi dalam mengelola operasional jalan tol.

Implementasi sistem spawn kendaraan pada ruas jalan tol menggunakan algoritma Timer-Based Spawning untuk mengatur interval kemunculan kendaraan secara berkala. Setelah interval terpenuhi, sistem menerapkan Randomized Vehicle Selection untuk memilih jenis kendaraan secara acak dari kumpulan prefab yang tersedia. Selanjutnya, kendaraan ditempatkan pada salah satu titik spawn yang telah ditentukan sehingga distribusi kendaraan dapat berlangsung secara merata di sepanjang ruas jalan tol. Kombinasi mekanisme tersebut menghasilkan variasi lalu lintas yang lebih alami tanpa mengganggu performa permainan.

Proses spawn kendaraan pada ruas jalan tol diawali dengan menjalankan *coroutine* yang bekerja secara berulang selama permainan berlangsung. Setelah mencapai interval waktu yang telah ditentukan, sistem memilih salah satu jenis kendaraan secara acak dari kumpulan prefab yang tersedia, kemudian menentukan titik spawn sesuai konfigurasi yang telah ditetapkan pada ruas jalan tol. Kendaraan yang berhasil dibuat selanjutnya diarahkan untuk bergerak mengikuti jalur yang telah ditentukan sehingga arus lalu lintas pada setiap ruas jalan tetap terjaga dan terlihat lebih dinamis.

```
1 reference
IEnumerator SpawnLoop()
{
    while (isActive)
    {
        yield return new WaitForSeconds(currentSpawnInterval);
        if (IsSafeToSpawn())
        {
            SpawnCar();
        }
        currentSpawnInterval = Random.Range(minSpawnInterval, maxSpawnInterval);
    }
}
```

Gambar 4.9 *Script Timer-Based Spawning*

Gambar 4.9 menunjukkan implementasi algoritma Timer-Based Spawning menggunakan *coroutine* SpawnLoop(). Selama sistem

dalam kondisi aktif (isActive), proses spawn akan berjalan secara berulang dengan jeda waktu yang ditentukan oleh nilai currentSpawnInterval. Setelah interval terpenuhi, sistem melakukan validasi keamanan area spawn sebelum memanggil fungsi SpawnCar(). Selanjutnya, nilai interval spawn diperbarui secara acak agar waktu kemunculan kendaraan berikutnya menjadi lebih bervariasi sehingga simulasi lalu lintas terlihat lebih alami.

```
1 reference
bool IsSafeToSpawn()
{
    if (spawnPoint == null) return false;

    Collider[] hits = Physics.OverlapSphere(
        spawnPoint.position,
        detectionDistance,
        carLayer
    );

    return hits.Length == 0;
}
```

Gambar 4.10 *Script Safety-Based Spawn Checking*

Gambar 4.10 menunjukkan implementasi Safety-Based Spawn Checking yang digunakan untuk memastikan area spawn berada dalam kondisi aman sebelum kendaraan dibuat. Sistem memanfaatkan fungsi Physics.OverlapSphere() untuk mendeteksi keberadaan kendaraan lain di sekitar titik spawn berdasarkan nilai detectionDistance. Apabila tidak ditemukan objek kendaraan pada area tersebut, proses spawn akan dilanjutkan. Sebaliknya, apabila area masih ditempati kendaraan lain, sistem akan menunda proses spawn hingga area kembali kosong sehingga dapat mengurangi risiko tabrakan antar kendaraan.

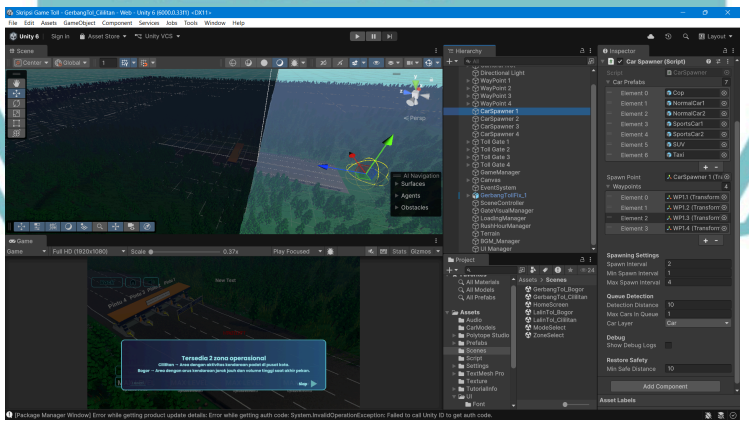
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
int index = Random.Range(0, carPrefabs.Length);
GameObject car = Instantiate(carPrefabs[index], spawnPoint.position, Quaternion.identity);
CarAI ai = car.GetComponent<CarAI>();
ai.currentLane = laneController;
if (ai != null)
{
    ai.waypoints = waypoints;
    ai.currentMode = CarAI.CarMode.Highway;

    spawnedCars.Add(car);

    if (showDebugLogs)
        Debug.Log($"🚗 Car spawned: {ai.carID}");
}
else
{
    Debug.LogWarning("Spawned car has no HighwayCarAI component! Adding one...");
    HighwayCarAI newAI = car.AddComponent<HighwayCarAI>();
    newAI.waypoints = waypoints;
    spawnedCars.Add(car);
}
```

Gambar 4.11 *Script Randomized Vehicle Selection*

Gambar 4.11 menunjukkan implementasi Randomized Vehicle Selection dan proses distribusi kendaraan pada ruas jalan tol. Sistem memilih salah satu prefab kendaraan secara acak menggunakan fungsi Random.Range(), kemudian membuat objek kendaraan pada titik spawn yang telah ditentukan. Setelah kendaraan berhasil dibuat, sistem menginisialisasi komponen CarAI dengan menetapkan jalur (laneController), waypoint, serta mode kendaraan sebagai Highway, sehingga kendaraan dapat bergerak mengikuti lintasan yang sesuai pada ruas jalan tol. Pendekatan ini menghasilkan variasi kendaraan sekaligus menjaga distribusi lalu lintas tetap terorganisir pada setiap jalur.



Gambar 4.12 Implementasi Spawn Kendaraan Ruas pada Unity

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Pada tahap awal pengembangan, sistem spawn kendaraan pada ruas jalan tol menggunakan interval spawn yang bersifat tetap sehingga pola kemunculan kendaraan terlihat kurang bervariasi. Berdasarkan hasil pengujian, interval spawn kemudian diubah menjadi acak dalam rentang minSpawnInterval hingga maxSpawnInterval untuk menghasilkan distribusi kendaraan yang lebih dinamis. Selain itu, ditambahkan mekanisme Safety-Based Spawn Checking agar kendaraan hanya dihasilkan ketika area spawn berada dalam kondisi aman. Perubahan tersebut menghasilkan arus lalu lintas yang lebih alami serta mengurangi kemungkinan terjadinya penumpukan kendaraan pada titik spawn.

4.3.3.3 Implementasi Sistem Event Rush Hour

Sistem Event *Rush Hour* merupakan mekanisme yang dirancang untuk mensimulasikan kondisi lalu lintas padat pada waktu tertentu selama permainan berlangsung. Ketika event aktif, sistem akan meningkatkan intensitas lalu lintas dengan mengubah beberapa parameter permainan, seperti interval kemunculan kendaraan dan kecepatan kendaraan, sehingga volume kendaraan yang melintas menjadi lebih tinggi dibandingkan kondisi normal. Implementasi mekanisme ini bertujuan memberikan variasi tingkat kesulitan sekaligus meningkatkan tantangan bagi pemain dalam mengelola operasional jalan tol.

Implementasi Sistem Event *Rush Hour* menggunakan algoritma Timer-Based Event Scheduling sebagai mekanisme utama untuk mengatur waktu dimulainya dan berakhirnya event berdasarkan penghitung waktu (*timer*). Setelah event aktif, sistem menerapkan Dynamic Parameter Adjustment, yaitu mekanisme yang menyesuaikan parameter permainan seperti interval spawn kendaraan dan kecepatan kendaraan secara dinamis sesuai kondisi *Rush Hour*. Kombinasi kedua algoritma tersebut memungkinkan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

perubahan kondisi lalu lintas terjadi secara otomatis tanpa memerlukan intervensi pemain sehingga simulasi permainan menjadi lebih dinamis dan realistis.

Proses Event *Rush Hour* diawali dengan penghitung waktu yang berjalan secara terus-menerus selama permainan berlangsung. Ketika penghitung waktu mencapai nol, sistem akan mengaktifkan status *Rush Hour* dan mengubah parameter permainan melalui penyesuaian interval spawn kendaraan serta kecepatan kendaraan. Selama event berlangsung, kendaraan akan muncul lebih cepat sehingga kepadatan lalu lintas meningkat. Setelah durasi *Rush Hour* berakhir, sistem akan mengembalikan seluruh parameter permainan ke kondisi normal dan memulai kembali proses perhitungan waktu menuju event berikutnya. Dengan mekanisme tersebut, kondisi lalu lintas dapat berubah secara otomatis sesuai siklus permainan tanpa memerlukan kontrol langsung dari pemain.

```
0 references | Unity Message
void Update()
{
    if (currentTimer > 0)
    {
        currentTimer -= Time.deltaTime;
        UpdateTimerUI();

        if (currentTimer <= 0 && !isRushHour)
        {
            StartRushHour();
            UpdateBreakdownChance();
        }
    }

    if (isRushHour)
    {
        UpdateAllCarSpeeds();
    }
}
```

Gambar 4.13 *Script Timer-Based Event Scheduling*

Potongan kode pada Gambar 4.13 menunjukkan implementasi algoritma Timer-Based Event Scheduling pada sistem Event *Rush Hour*. Algoritma ini memanfaatkan fungsi Update() untuk mengurangi nilai currentTimer menggunakan Time.deltaTime

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

sehingga waktu dihitung secara real-time selama permainan berlangsung. Ketika nilai penghitung mencapai nol, sistem akan melakukan pengecekan status *Rush Hour*. Apabila event belum aktif, sistem akan menjalankan fungsi StartRushHour(), sedangkan apabila event sedang berlangsung maka sistem akan memanggil fungsi EndRushHour(). Pendekatan ini memungkinkan Event *Rush Hour* berlangsung secara periodik dan otomatis tanpa memerlukan interaksi langsung dari pemain.

```
1 reference
void StartRushHour()
{
    isRushHour = true;
    currentTimer = rushHourDuration;

    // Ubah spawn interval semua spawner jadi 2x lebih cepat
    foreach (HighwayCarSpawner spawner in allSpawners)
    {
        if (spawner != null)
        {
            spawner.minSpawnInterval = rushHourSpawnMin;
            spawner.maxSpawnInterval = rushHourSpawnMax;
            Debug.Log($"🚗 Spawner {spawner.name} speed up: {rushHourSpawnMin}-{rushHourSpawnMax}s");
        }
    }

    // Ubah kecepatan semua mobil yang sudah ada
    UpdateAllCarSpeeds();

    Debug.Log($"🚨 RUSH HOUR STARTED! {rushHourDuration} seconds of chaos!");
}
```

Gambar 4.14 *Script Event Rush Hour*

Potongan kode pada Gambar 4.14 menunjukkan proses aktivasi Event *Rush Hour*. Ketika fungsi StartRushHour() dipanggil, sistem akan mengubah status permainan menjadi *Rush Hour* melalui variabel isRushHour, kemudian mengatur ulang nilai currentTimer sesuai durasi event yang telah ditentukan. Aktivasi tersebut menjadi pemicu bagi sistem untuk menerapkan perubahan parameter permainan sehingga kondisi lalu lintas berubah dari kondisi normal menjadi kondisi padat secara otomatis.

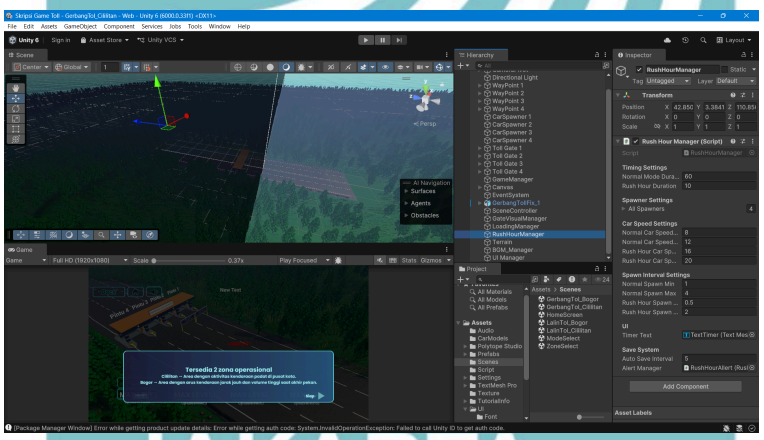
```
// Ubah spawn interval semua spawner jadi 2x lebih cepat
foreach (HighwayCarSpawner spawner in allSpawners)
{
    if (spawner != null)
    {
        spawner.minSpawnInterval = rushHourSpawnMin;
        spawner.maxSpawnInterval = rushHourSpawnMax;
        Debug.Log($"🚗 Spawner {spawner.name} speed up: {rushHourSpawnMin}-{rushHourSpawnMax}s");
    }
}
```

Gambar 4.15 *Script Dynamic Parameter Adjustment*

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Potongan kode pada Gambar 4.15 menunjukkan implementasi algoritma Dynamic Parameter Adjustment pada sistem Event *Rush Hour*. Ketika event dimulai, sistem memperbarui nilai minSpawnInterval dan maxSpawnInterval pada seluruh objek CarSpawner menggunakan parameter khusus *Rush Hour* sehingga interval kemunculan kendaraan menjadi lebih singkat. Selain itu, fungsi UpdateAllCarSpeeds() dijalankan untuk menyesuaikan kecepatan seluruh kendaraan yang sedang aktif. Setelah Event *Rush Hour* berakhir, sistem mengembalikan seluruh parameter spawn serta kecepatan kendaraan ke nilai normal sehingga kondisi lalu lintas kembali stabil. Pendekatan ini memungkinkan perubahan tingkat kesulitan permainan dilakukan secara dinamis tanpa mengubah mekanisme utama sistem spawn kendaraan.



Gambar 4.16 Implementasi Sistem Event Rush Hour pada Unity

Pada tahap awal pengembangan, Event *Rush Hour* hanya mengurangi interval spawn kendaraan sehingga peningkatan kepadatan lalu lintas masih terasa kurang signifikan. Berdasarkan hasil pengujian, sistem kemudian dikembangkan dengan menambahkan penyesuaian kecepatan kendaraan serta pengaturan interval spawn secara terpisah untuk kondisi normal dan *Rush Hour*. Selain itu, durasi event dan waktu kemunculan *Rush Hour* juga disesuaikan melalui beberapa tahap pengujian agar pemain memiliki waktu yang cukup untuk mempersiapkan strategi sebelum

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

menghadapi peningkatan volume kendaraan. Hasil iterasi tersebut menghasilkan kondisi lalu lintas yang lebih menantang namun tetap seimbang sehingga pengalaman bermain menjadi lebih menarik.

4.3.4 Implementasi Sistem Pergerakan dan Antrean Kendaraan

Sistem pergerakan dan antrean kendaraan merupakan salah satu komponen utama dalam game *Toll Tycoon* yang berperan dalam menciptakan simulasi lalu lintas di area gerbang tol. Sistem ini mengatur bagaimana kendaraan bergerak mengikuti jalur yang telah ditentukan, mendeteksi kendaraan lain untuk menghindari tabrakan, menyesuaikan kecepatan berdasarkan kondisi di sekitarnya, serta membentuk antrean ketika memasuki area gerbang tol. Selain itu, sistem juga mengatur urutan pelayanan kendaraan menggunakan mekanisme antrean sehingga proses pembayaran dapat berlangsung secara teratur. Implementasi sistem ini memanfaatkan beberapa algoritma yang saling terintegrasi untuk menghasilkan perilaku kendaraan yang realistis dan menjaga kelancaran arus lalu lintas selama permainan berlangsung.

4.3.4.1 Implementasi Sistem Pergerakan Kendaraan

Sistem pergerakan kendaraan merupakan komponen utama yang mengatur perilaku kendaraan setelah berhasil dihasilkan oleh sistem *spawn*. Pada permainan *Toll Tycoon*, setiap kendaraan bergerak secara otomatis mengikuti jalur yang telah ditentukan menggunakan kumpulan *waypoint*. Selama proses pergerakan, sistem secara terus-menerus memantau kondisi lalu lintas di depan kendaraan untuk menyesuaikan kecepatan maupun menghentikan kendaraan apabila terdapat hambatan. Implementasi sistem ini bertujuan menciptakan simulasi lalu lintas yang lebih dinamis dan realistis sehingga setiap kendaraan dapat bergerak secara teratur sesuai kondisi yang terjadi di dalam permainan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Implementasi sistem pergerakan kendaraan menggunakan kombinasi tiga algoritma utama, yaitu Waypoint Following Algorithm, Obstacle Detection berbasis Raycasting, dan Dynamic Speed Control. *Waypoint Following Algorithm* digunakan untuk mengarahkan kendaraan mengikuti lintasan yang telah ditentukan melalui serangkaian titik tujuan (*waypoint*). Selanjutnya, algoritma Obstacle Detection memanfaatkan teknik *raycasting* untuk mendeteksi keberadaan kendaraan lain di depan sehingga dapat menghindari tabrakan. Berdasarkan hasil deteksi tersebut, algoritma Dynamic Speed Control akan menyesuaikan kecepatan kendaraan secara dinamis sesuai jarak terhadap kendaraan di depannya. Kombinasi ketiga algoritma tersebut menghasilkan pergerakan kendaraan yang lebih halus, responsif, dan menyerupai kondisi lalu lintas nyata.

Proses pergerakan kendaraan diawali dengan menentukan *waypoint* tujuan yang harus dicapai oleh setiap kendaraan. Selama kendaraan bergerak menuju *waypoint* tersebut, sistem secara berkala melakukan deteksi menggunakan *raycasting* untuk mengetahui keberadaan kendaraan lain di depan. Apabila jalur dalam kondisi aman, kendaraan akan melanjutkan perjalanan dengan kecepatan normal menuju *waypoint* berikutnya. Sebaliknya, apabila terdeteksi hambatan, sistem akan mengurangi kecepatan kendaraan secara bertahap atau menghentikan kendaraan apabila jarak terhadap hambatan berada di bawah batas yang telah ditentukan. Setelah jalur kembali kosong, kendaraan akan kembali bergerak dan melanjutkan perjalanan menuju tujuan berikutnya hingga mencapai titik akhir lintasan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
Transform target = waypoints[currentWaypoint];
if (target == null)
{
    currentWaypoint++;
    return;
}

float finalSpeed = currentSpeed;
if (currentLane != null)
{
    finalSpeed *=
        currentLane.currentSpeedMultiplier;
}

Vector3 newPosition = Vector3.MoveTowards(
    transform.position,
    target.position,
    finalSpeed * Time.deltaTime
);

if (!isBlocked || distanceToObstacle > stoppingDistance + 0.5f)
{
    transform.position = newPosition;
}

Vector3 direction = (target.position - transform.position).normalized;
if (direction != Vector3.zero)
{
    Quaternion rot = Quaternion.LookRotation(direction);
    transform.rotation = Quaternion.Slerp(transform.rotation, rot, 5f *
Time.deltaTime);
}

if (Vector3.Distance(transform.position, target.position) < 0.5f)
{
    currentWaypoint++;

    if (currentWaypoint >= waypoints.Length)
    {
        if (IsInQueue)
        {
            IsInQueue = false;
        }

        if (currentMode == CarMode.Highway)
        {
            GiveReward();
        }

        DestroyCar();
    }
}
```

Gambar 4.17 *Script Waypoint Following Algorithm*

Gambar 4.17 menunjukkan implementasi Waypoint Following Algorithm pada sistem pergerakan kendaraan. Sistem terlebih dahulu menentukan waypoint tujuan melalui variabel target, kemudian menghitung kecepatan akhir kendaraan menggunakan nilai currentSpeed yang dapat dipengaruhi oleh currentSpeedMultiplier pada jalur aktif. Selanjutnya, kendaraan bergerak menuju waypoint menggunakan fungsi Vector3.MoveTowards(). Selama proses tersebut, arah hadap

kendaraan disesuaikan secara bertahap menggunakan Quaternion.Slerp() agar rotasi terlihat lebih halus ketika mengikuti lintasan. Setelah kendaraan mencapai waypoint dengan jarak kurang dari 0,5 meter, sistem akan meningkatkan indeks waypoint (currentWaypoint) untuk mengarahkan kendaraan ke titik berikutnya. Apabila seluruh waypoint telah dilalui, kendaraan akan menyelesaikan proses perjalanannya sesuai mode permainan yang sedang digunakan.

```
bool isBlocked = false;
float distanceToObstacle = detectionDistance;

Vector3 rayOrigin = transform.position + transform.forward * raycastOffset + Vector3.up * 0.5f;

RaycastHit hit;
if (Physics.Raycast(rayOrigin, transform.forward, out hit, detectionDistance))
{
    if (hit.collider.CompareTag("Car") && hit.collider.gameObject != gameObject)
    {
        isBlocked = true;
        distanceToObstacle = hit.distance;
    }
}
```

Gambar 4.18 *Script Obstacle Detection*

Gambar 4.18 menunjukkan implementasi algoritma Obstacle Detection menggunakan metode Raycasting. Sistem terlebih dahulu menentukan titik awal ray (rayOrigin) yang berada di depan kendaraan untuk menghindari deteksi terhadap objek kendaraan itu sendiri. Selanjutnya, fungsi Physics.Raycast() digunakan untuk mendeteksi keberadaan objek pada arah pergerakan kendaraan sejauh detectionDistance. Apabila objek yang terdeteksi memiliki tag Car dan bukan kendaraan itu sendiri, maka variabel isBlocked akan diubah menjadi true serta jarak kendaraan terhadap hambatan disimpan pada variabel distanceToObstacle. Informasi tersebut kemudian digunakan sebagai dasar bagi sistem untuk menentukan penyesuaian kecepatan kendaraan pada proses pergerakan berikutnya sehingga risiko tabrakan antar kendaraan dapat diminimalkan.

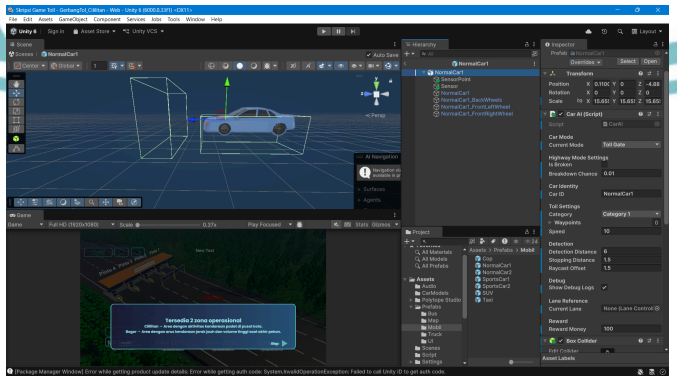
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
if (isBlocked)
{
    if (distanceToObstacle <= stoppingDistance)
    {
        currentSpeed = 0f;
    }
    else if (distanceToObstacle < detectionDistance)
    {
        float speedMultiplier = (distanceToObstacle - stoppingDistance) / (detectionDistance - stoppingDistance);
        currentSpeed = Mathf.Lerp(0f, speed, speedMultiplier);
    }
    else
    {
        currentSpeed = speed;
    }
}
else
{
    currentSpeed = Mathf.Min(currentSpeed + Time.deltaTime * 10f, speed);
}
```

Gambar 4.19 *Script Dynamic Speed Control*

Gambar 4.19 menunjukkan implementasi algoritma Dynamic Speed Control yang digunakan untuk mengatur kecepatan kendaraan berdasarkan hasil deteksi hambatan di depan. Apabila kendaraan berada pada jarak yang lebih kecil atau sama dengan stoppingDistance, sistem akan menghentikan kendaraan dengan mengatur nilai currentSpeed menjadi 0f. Jika hambatan masih berada dalam area deteksi tetapi belum mencapai batas berhenti, sistem menghitung nilai speedMultiplier berdasarkan jarak terhadap hambatan, kemudian menggunakan fungsi Mathf.Lerp() untuk menyesuaikan kecepatan kendaraan secara bertahap. Sebaliknya, apabila tidak terdapat hambatan di depan, sistem akan meningkatkan kecepatan kendaraan secara perlahan hingga kembali ke kecepatan normal menggunakan fungsi Mathf.Min(). Pendekatan ini menghasilkan perubahan kecepatan yang lebih halus sehingga pergerakan kendaraan terlihat lebih alami dan mampu mengurangi potensi tabrakan selama simulasi berlangsung.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.20 Implementasi Sistem Pergerakan Kendaraan pada Unity

Pada tahap awal pegembangan, sistem pergerakan kendaraan menggunakan nilai kecepatan dan jarak deteksi yang bersifat tetap sehingga kendaraan cenderung berhenti secara mendadak ketika terdapat kendaraan lain di depannya. Kondisi tersebut menyebabkan pergerakan kendaraan terlihat kurang alami dan beberapa kali menimbulkan tabrakan akibat respons yang terlambat terhadap hambatan.

Berdasarkan hasil pengujian, sistem kemudian dikembangkan dengan menerapkan mekanisme Dynamic Speed Control sehingga kecepatan kendaraan dapat menyesuaikan secara bertahap berdasarkan jarak terhadap kendaraan di depannya. Selain itu, parameter detectionDistance, stoppingDistance, dan currentSpeedMultiplier juga disesuaikan melalui beberapa tahap pengujian untuk memperoleh jarak deteksi dan kecepatan yang optimal pada setiap kondisi lalu lintas.

Hasil iterasi menunjukkan bahwa kombinasi algoritma Waypoint Following, Obstacle Detection, dan Dynamic Speed Control mampu menghasilkan pergerakan kendaraan yang lebih halus, responsif, serta mengurangi potensi tabrakan antar kendaraan. Dengan penyesuaian parameter tersebut, simulasi lalu lintas pada permainan *Toll Tycoon* menjadi lebih stabil dan memberikan pengalaman bermain yang lebih realistis.

4.3.4.2 Implementasi Sistem Antrean Kendaraan

Sistem antrean kendaraan merupakan mekanisme yang mengatur kendaraan ketika memasuki area gerbang tol agar proses pelayanan dapat berlangsung secara teratur. Setiap kendaraan yang tiba di area antrean akan dideteksi dan ditempatkan pada urutan antrean sesuai kedatangannya. Selama proses tersebut, sistem mengendalikan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

pergerakan kendaraan sehingga hanya kendaraan yang berada di posisi terdepan yang dapat memasuki gerbang tol, sedangkan kendaraan lainnya akan menunggu hingga giliran pelayanan berikutnya. Implementasi sistem ini bertujuan menciptakan simulasi antrean yang lebih realistis, mencegah terjadinya tabrakan antar kendaraan, serta memastikan proses pelayanan kendaraan berlangsung secara tertib.

Implementasi sistem antrean kendaraan menggunakan kombinasi dua algoritma utama, yaitu Queue Detection Algorithm dan First In First Out (FIFO) Queue Management. Queue Detection Algorithm digunakan untuk mendeteksi kendaraan yang memasuki area antrean dan mencatat status kendaraan sebagai bagian dari antrean. Selanjutnya, algoritma First In First Out (FIFO) Queue Management mengatur urutan pelayanan kendaraan berdasarkan urutan kedatangannya sehingga kendaraan yang lebih dahulu memasuki antrean akan diproses terlebih dahulu. Kombinasi kedua algoritma tersebut menghasilkan proses antrean yang lebih teratur, mencegah kendaraan saling mendahului, serta menjaga urutan pelayanan agar menyerupai mekanisme antrean pada gerbang tol sebenarnya.

Proses antrean kendaraan diawali ketika kendaraan memasuki area deteksi antrean. Sistem akan mendeteksi kendaraan tersebut kemudian menambahkan kendaraan ke dalam struktur data antrean. Selanjutnya, kendaraan diproses berdasarkan urutan kedatangannya menggunakan mekanisme First In First Out (FIFO) sehingga kendaraan yang berada pada posisi paling depan memperoleh prioritas pelayanan terlebih dahulu. Selama proses pelayanan berlangsung, kendaraan lain tetap mempertahankan urutan antreannya hingga kendaraan terdepan selesai diproses. Setelah proses pelayanan selesai, kendaraan pertama dikeluarkan dari antrean dan sistem akan memproses kendaraan berikutnya secara

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

otomatis. Mekanisme tersebut berlangsung secara berulang hingga seluruh kendaraan dalam antrean berhasil dilayani.

```
    private void OnTriggerEnter(Collider other)
    {
        if (!other.CompareTag("Car")) return;

        CarAI car = other.GetComponentInParent<CarAI>();

        if (car == null) return;
        if (car.HasReachedDestination()) return;
        if (queuedCars.Contains(car)) return;
        if (queuedCars.Count >= maxQueueSize) return;

        queuedCars.Add(car);
        car.IsInQueue = true;

        if (showDebugLogs && Time.time - lastDebugLogTime >
debugLogInterval)
        {
            // HAPUS queueNumber dari sini
            Debug.Log($"🚗 MASUK ANTRIAN: {car.carID} | Total:
{queuedCars.Count}/{maxQueueSize}");
            lastDebugLogTime = Time.time;
        }

        OnQueueChanged?.Invoke(queuedCars.Count);
        OnCarEntered?.Invoke(car);
    }

    private void OnTriggerExit(Collider other)
    {
        if (!other.CompareTag("Car")) return;

        CarAI car = other.GetComponentInParent<CarAI>();

        if (car == null) return;

        if (queuedCars.Contains(car))
        {
            queuedCars.Remove(car);
            car.IsInQueue = false;

            if (showDebugLogs && Time.time - lastDebugLogTime >
debugLogInterval)
            {
                Debug.Log($"🚗 KELUAR ANTRIAN: {car.carID} | Sisa:
{queuedCars.Count}/{maxQueueSize}");
                lastDebugLogTime = Time.time;
            }

            OnQueueChanged?.Invoke(queuedCars.Count);
            OnCarExited?.Invoke(car);
        }
    }
```

Gambar 4.21 *Script Queue Detection Algorithm*

Gambar 4.21 menunjukkan implementasi Queue Detection Algorithm yang digunakan untuk mendeteksi kendaraan yang memasuki area antrean gerbang tol. Sistem memanfaatkan metode OnTriggerEnter() untuk mendeteksi kendaraan yang memasuki area *trigger*. Apabila objek yang terdeteksi memiliki komponen CarAI dan belum berada di dalam antrean, kendaraan akan ditambahkan ke dalam struktur data antrean menggunakan fungsi

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Enqueue() serta status IsInQueue diubah menjadi aktif. Selanjutnya, ketika kendaraan keluar dari area antrean, metode OnTriggerExit() akan dijalankan untuk mengubah kembali status IsInQueue menjadi tidak aktif. Mekanisme ini memastikan sistem selalu mengetahui kendaraan yang sedang berada di dalam area antrean sehingga proses pengelolaan antrean dapat dilakukan dengan tepat.

```
private void OnTriggerEnter(Collider other)
{
    if (!isUnlocked) return;

    if (other.CompareTag("Car"))
    {
        CarAI car = other.GetComponentInParent<CarAI>();

        if (car != null && !carQueue.Contains(car))
        {
            car.IsInQueue = true;
            car.StartPaying();
            carQueue.Enqueue(car);

            if (IsAuto())
            {
                TryProcessNextCar();
            }
        }
    }
}

void TryProcessNextCar()
{
    if (!isProcessing && carQueue.Count > 0)
        StartCoroutine(ProcessCar());
}

IEnumerator ProcessCar()
{
    isProcessing = true;

    CarAI car = carQueue.Dequeue();
    car.IsInQueue = false;
```

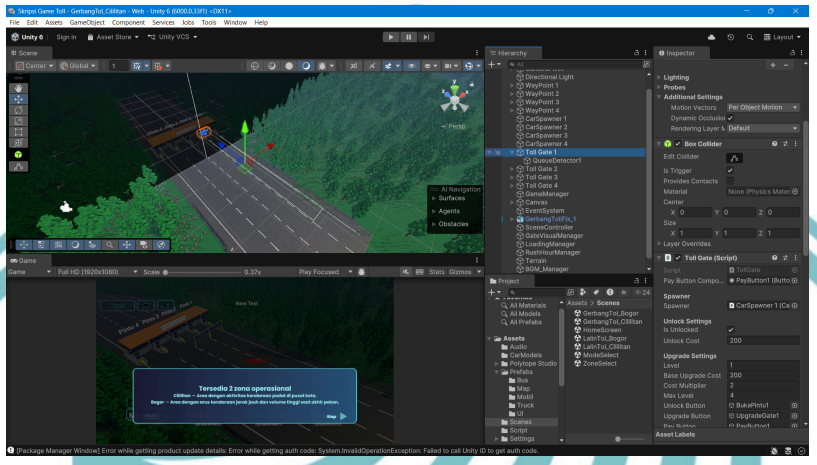
Gambar 4.22 *Script First In First Out*

Gambar 4.22 menunjukkan implementasi algoritma First In First Out (FIFO) Queue Management pada sistem antrean kendaraan. Ketika kendaraan memasuki area gerbang tol, sistem akan menambahkan kendaraan ke dalam struktur data antrean menggunakan fungsi Enqueue(). Selanjutnya, fungsi TryProcessNextCar() memeriksa apakah masih terdapat kendaraan di dalam antrean dan memastikan tidak ada kendaraan lain yang sedang diproses. Apabila kondisi tersebut terpenuhi, sistem menjalankan ProcessCar() untuk mengambil kendaraan pertama

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

menggunakan fungsi Dequeue(). Dengan mekanisme ini, kendaraan yang pertama kali memasuki antrean akan menjadi kendaraan pertama yang diproses sehingga urutan pelayanan tetap mengikuti prinsip First In First Out (FIFO).



Gambar 4.23 Implementasi Sistem Antrean Kendaraan pada Unity

Pada tahap awal pengembangan, sistem antrean kendaraan masih memungkinkan terjadinya konflik ketika beberapa kendaraan memasuki area gerbang tol secara bersamaan. Kondisi tersebut menyebabkan kendaraan berpotensi diproses secara bersamaan atau urutan pelayanan menjadi tidak konsisten sehingga mengurangi realisme simulasi antrean. Berdasarkan hasil pengujian, sistem kemudian dikembangkan dengan menerapkan mekanisme Queue Detection untuk mendeteksi setiap kendaraan yang memasuki area antrean serta memanfaatkan struktur data Queue yang menerapkan prinsip First In First Out (FIFO) sehingga urutan pelayanan kendaraan selalu mengikuti urutan kedatangannya.

Selain itu, dilakukan penyesuaian pada mekanisme pemrosesan antrean melalui penggunaan variabel isProcessing sebagai pengunci (*locking mechanism*) agar hanya satu kendaraan yang dapat diproses pada satu waktu. Pengujian juga dilakukan pada berbagai kondisi kepadatan kendaraan untuk memastikan fungsi TryProcessNextCar() hanya memanggil proses pelayanan setelah

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kendaraan sebelumnya selesai diproses. Hasil iterasi menunjukkan bahwa kombinasi Queue Detection, FIFO Queue Management, serta mekanisme penguncian proses mampu menjaga urutan pelayanan kendaraan tetap konsisten, mencegah konflik pemrosesan, dan menghasilkan sistem antrean yang lebih stabil serta menyerupai kondisi antrean pada gerbang tol sebenarnya.

4.3.5 Implementasi Sistem Pendapatan Gerbang Tol

Sistem pendapatan gerbang tol merupakan komponen yang mengelola proses transaksi kendaraan selama permainan berlangsung. Setiap kendaraan yang berhasil melewati gerbang tol akan menghasilkan pendapatan sesuai dengan tarif kendaraan yang diproses. Selain menambahkan saldo pemain, sistem juga memberikan umpan balik berupa efek visual dan suara untuk menunjukkan bahwa transaksi telah berhasil dilakukan. Implementasi sistem ini bertujuan menciptakan mekanisme ekonomi yang terintegrasi dengan aktivitas pemain sehingga setiap keputusan dalam mengelola gerbang tol memberikan dampak langsung terhadap perkembangan permainan.

4.3.5.1 Implementasi Sistem Transaksi Kendaraan

Sistem transaksi kendaraan merupakan mekanisme yang mengelola proses pembayaran setiap kendaraan yang berhasil melewati gerbang tol. Setiap kendaraan memiliki tarif sesuai dengan jenis kendaraannya yang akan dikonversi menjadi pendapatan pemain setelah proses transaksi selesai dilakukan. Selain melakukan penambahan saldo, sistem juga menampilkan umpan balik berupa teks pendapatan dan efek suara sebagai konfirmasi bahwa transaksi telah berhasil diproses. Implementasi sistem ini bertujuan menciptakan mekanisme transaksi yang terintegrasi dengan sistem ekonomi permainan sehingga setiap kendaraan yang dilayani memberikan kontribusi terhadap perkembangan permainan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Implementasi sistem transaksi kendaraan menggunakan kombinasi dua algoritma utama, yaitu Transaction Processing Algorithm dan Accumulator Pattern. *Transaction Processing Algorithm* digunakan untuk mengelola tahapan transaksi mulai dari pengambilan nilai tarif kendaraan, pemrosesan pembayaran, hingga penyelesaian transaksi sebelum kendaraan melanjutkan perjalanan. Selanjutnya, *Accumulator Pattern* digunakan untuk menambahkan nilai pendapatan setiap transaksi ke dalam total saldo pemain secara kumulatif. Kombinasi kedua algoritma tersebut menghasilkan proses transaksi yang konsisten, menjaga akumulasi pendapatan tetap akurat, serta memberikan umpan balik kepada pemain setiap kali transaksi berhasil dilakukan.

Proses transaksi kendaraan diawali ketika kendaraan yang berada pada posisi terdepan antrean selesai diproses oleh sistem gerbang tol. Sistem kemudian mengambil nilai tarif kendaraan berdasarkan jenis kendaraan yang sedang dilayani. Nilai tersebut selanjutnya ditambahkan ke dalam total saldo pemain melalui sistem pengelolaan pendapatan. Setelah proses transaksi berhasil dilakukan, sistem menampilkan umpan balik berupa teks pendapatan (*floating text*) dan efek suara sebagai indikator keberhasilan transaksi. Selanjutnya kendaraan keluar dari area gerbang tol, kemudian sistem akan memproses kendaraan berikutnya apabila masih terdapat kendaraan di dalam antrean. Mekanisme tersebut berlangsung secara berulang selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
int money = car.GetPrice();

if (MoneyManager.instance ≠ null)
    MoneyManager.instance.AddMoney(money);

if (audioSource ≠ null && moneySound ≠ null)
    audioSource.PlayOneShot(
        moneySound,
        0.5f * BGMManager.SFXVolume
    );
```

Gambar 4.24 *Script Transaction Processing Algorithm*

Gambar 4.24 menunjukkan implementasi *Transaction Processing Algorithm* pada proses perhitungan tarif kendaraan. Sistem terlebih dahulu mengambil nilai tarif kendaraan melalui pemanggilan fungsi GetPrice() yang terdapat pada objek CarAI. Fungsi tersebut mengembalikan nilai tarif berdasarkan jenis kendaraan yang sedang diproses pada gerbang tol. Nilai tarif kemudian disimpan ke dalam variabel money sebagai dasar perhitungan pendapatan sebelum diteruskan ke sistem pengelolaan saldo pemain. Pendekatan ini memungkinkan setiap jenis kendaraan memberikan nilai pendapatan yang berbeda sesuai dengan karakteristik kendaraan yang telah ditentukan pada permainan.

```
2 references
public void AddMoney(int amount)
{
    money += amount;
    PlayerPrefs.SetInt("Money", money);
}
```

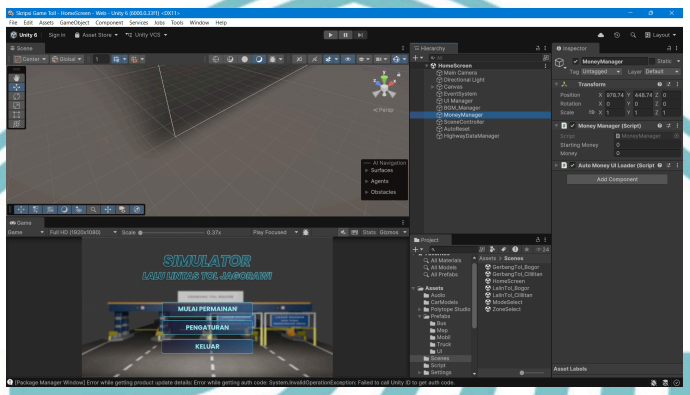
Gambar 4.25 *Script Accumulator Pattern*

Gambar 4.25 menunjukkan implementasi *Accumulator Pattern* pada proses penambahan pendapatan pemain. Setelah nilai tarif kendaraan diperoleh, sistem memanggil fungsi AddMoney() pada kelas MoneyManager untuk menambahkan nilai transaksi ke dalam total saldo pemain secara kumulatif. Di dalam fungsi tersebut, nilai transaksi dijumlahkan dengan saldo yang telah dimiliki

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

sebelumnya, kemudian hasilnya disimpan menggunakan PlayerPrefs agar data pendapatan tetap tersedia meskipun permainan ditutup atau berpindah scene. Pendekatan ini memastikan proses akumulasi pendapatan berlangsung secara konsisten selama permainan.



Gambar 4.26 Implementasi Sistem Transaksi Kendaraan pada Unity

Pada tahap awal pengembangan, sistem transaksi kendaraan hanya berfokus pada penambahan saldo pemain setelah kendaraan berhasil melewati gerbang tol. Namun, hasil pengujian menunjukkan bahwa mekanisme tersebut belum memberikan informasi yang cukup kepada pemain mengenai keberhasilan transaksi serta belum menjamin konsistensi data pendapatan ketika permainan berpindah scene atau ditutup. Berdasarkan hasil evaluasi, sistem kemudian dikembangkan dengan memisahkan pengelolaan saldo ke dalam kelas MoneyManager sehingga seluruh proses penambahan maupun pengurangan pendapatan dikelola secara terpusat. Selain itu, data saldo disimpan menggunakan PlayerPrefs agar nilai pendapatan tetap dapat dipertahankan selama permainan berlangsung.

Selanjutnya, dilakukan penambahan mekanisme *feedback* berupa *floating text* dan efek suara setelah transaksi berhasil diproses sehingga pemain memperoleh konfirmasi secara visual maupun audio terhadap setiap pendapatan yang diterima. Pengujian juga dilakukan pada berbagai jenis kendaraan untuk memastikan nilai tarif

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

yang diperoleh melalui fungsi GetPrice() dapat ditambahkan dengan benar ke dalam saldo pemain. Hasil iterasi menunjukkan bahwa mekanisme transaksi mampu mengelola pendapatan secara konsisten, memberikan umpan balik yang informatif kepada pemain, serta menjaga sinkronisasi data pendapatan selama permainan berlangsung.

4.3.5.2 Implementasi Sistem Balancing Ekonomi

Sistem balancing ekonomi digunakan untuk menjaga keseimbangan antara tingkat pendapatan pemain, biaya operasional, dan tingkat perkembangan permainan selama simulasi berlangsung. Implementasi balancing dilakukan agar pemain tetap memperoleh tantangan dalam mengelola gerbang tol tanpa menyebabkan permainan menjadi terlalu mudah maupun terlalu sulit.

Pada penelitian ini, proses balancing ekonomi dilakukan menggunakan konsep konversi waktu. Konsep ini digunakan untuk menghitung estimasi waktu yang dibutuhkan pemain dalam memperoleh pendapatan untuk melakukan upgrade maupun membuka gerbang tol baru berdasarkan performa operasional gerbang tol selama permainan berlangsung.

Implementasi balancing dilakukan dengan menghubungkan beberapa parameter utama permainan, seperti waktu transaksi kendaraan pada gerbang tol, interval *spawn* kendaraan, kategori kendaraan, nilai pendapatan kendaraan, biaya pembukaan gerbang tol, serta biaya upgrade gerbang tol.

Pada penelitian ini terdapat empat gerbang tol dengan biaya pembukaan (*unlock cost*) dan kategori kendaraan yang berbeda. Gerbang 1 terbuka sejak awal permainan dengan biaya upgrade awal sebesar 200. Gerbang 2 memiliki biaya pembukaan sebesar

250 dengan kendaraan kategori 1 bernilai 15. Sementara itu, Gerbang 3 dan Gerbang 4 memiliki biaya pembukaan lebih tinggi karena melayani kendaraan kategori 2 dan kategori 5 dengan nilai pendapatan lebih besar, yaitu 30 dan 40.

Selain itu, setiap level gerbang tol memiliki waktu transaksi kendaraan yang sama, yaitu:

1. Level Bayar Tunai = 1,5 detik
2. Level E-Money = 1 detik
3. Level RFID = 0,5 detik
4. Level MLFF = 0,1 detik

Sistem *spawn* kendaraan juga berubah berdasarkan level gerbang tol. Semakin tinggi level gerbang, maka interval *spawn* kendaraan menjadi lebih cepat sehingga jumlah kendaraan yang masuk menjadi lebih banyak.

Untuk menghitung estimasi perkembangan ekonomi permainan, digunakan rumus berikut:

$$Jumlah\ Kendaraan\ per\ Menit = \frac{60}{Rata - rata\ spawn\ interval}$$

Keterangan:

1. 60 = jumlah detik dalam satu menit
2. Spawn Interval = rata-rata waktu kemunculan kendaraan

Setelah diperoleh jumlah kendaraan per menit, maka estimasi pendapatan pemain dihitung menggunakan rumus:

$$Pendapatan\ per\ Menit = Jumlah\ Kendaraan\ per\ Menit \times Reward\ Kendaraan$$

Keterangan:

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

1. Jumlah Kendaraan per Menit = total kendaraan yang berhasil diproses dalam satu menit
2. Reward Kendaraan = nilai uang dari kategori kendaraan

Selanjutnya, estimasi waktu yang dibutuhkan pemain untuk melakukan upgrade dihitung menggunakan rumus:

$$Waktu\ Upgrade = \frac{Biaya\ Upgrade}{Pendapatan\ Per\ Menit}$$

Keterangan:

1. Biaya Upgrade = biaya peningkatan level gerbang tol
2. Pendapatan per Menit = estimasi total pendapatan pemain setiap menit

Sebagai contoh, pada Gerbang 1 level 1:

A. Reward kendaraan = 15
B. Rata-rata *spawn interval* = 3 detik
C. Delay transaksi = 1,5 detik
D. Biaya upgrade = 200

Maka jumlah kendaraan yang dapat diproses dalam satu menit adalah:

$$\frac{60}{3} = 20$$

Sehingga estimasi pendapatan pemain menjadi:

$$20 \times 15 = 300$$

Dengan biaya upgrade sebesar 200, maka estimasi waktu yang dibutuhkan pemain untuk memperoleh upgrade pertama adalah:

$$\frac{200}{300} = 0,67\ Menit$$

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Hasil tersebut menunjukkan bahwa pemain membutuhkan waktu sekitar 40 detik untuk memperoleh upgrade pertama pada kondisi lalu lintas normal.

Ketika pemain berhasil meningkatkan level gerbang tol menjadi level 2, waktu transaksi berubah dari 1,5 detik menjadi 1 detik sehingga kendaraan dapat diproses lebih cepat. Selain itu, interval *spawn* kendaraan juga berubah menjadi 2,0–3,0 detik sehingga volume kendaraan yang masuk menjadi lebih tinggi dibanding level sebelumnya.

Tabel 4.16 menunjukkan implementasi balancing ekonomi pada setiap gerbang tol.

Tabel 4.9 Implementasi balancing ekonomi setiap gerbang

Gate	Unlock Cost	Reward Kendaraan	Level Gate	Delay Transaksi	Spawn Interval	Estimasi Kendaraan / Menit	Estimasi Pendapatan / Menit
Gate 1	Terbuka awal	15	Level 1	1,5 Detik	1–4 detik	±24	±300
Gate 2	250	15	Level 1	1,5 detik	1–4 detik	±24	±360
Gate 3	450	30–40	Level 1	1,5 detik	1–3 detik	±30	±900–1200
Gate 4	500	30–40	Level 1	1,5 detik	1–3 detik	±30	±900–1200

Selain biaya pembukaan gerbang tol, sistem juga menerapkan peningkatan biaya upgrade menggunakan parameter *Cost*

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Multiplier sebesar 2. Hal tersebut menyebabkan biaya upgrade meningkat dua kali lipat pada setiap level sehingga perkembangan ekonomi permainan menjadi lebih bertahap dan tidak terlalu cepat.

Tabel 4.17 menunjukkan simulasi perkembangan ekonomi pada Gerbang 1 berdasarkan perubahan level gerbang tol.

Tabel 4.10 Perkembangan Ekonomi Pada Gerbang 1

Level Gate	Biaya Upgrade	Delay Transaksi	Spawn Interval	Estimasi Kendaraan / Menit	Reward Kendaraan	Estimasi Pendapatan / Menit	Estimasi Waktu Upgrade
Level Bayar Tunai	200	1,5 detik	1–4 detik	±24	15	±360	±34 detik
Level E-Money	400	1 detik	1–3 detik	±30	15	±450	±53 detik
Level RFID	800	0,5 detik	1–2 detik	±40	15	±600	±80 detik
Level 4	Maksimal	0,1 detik	1 detik	±60	15	±900	-

Berdasarkan hasil pengujian internal, parameter tersebut dinilai mampu menghasilkan progres permainan yang tidak terlalu cepat maupun terlalu lambat. Pada pengujian awal, biaya upgrade yang terlalu murah menyebabkan pemain dapat membuka seluruh gerbang tol dalam waktu singkat sehingga tantangan permainan menjadi rendah. Sebaliknya, biaya upgrade yang terlalu tinggi

menyebabkan pemain membutuhkan waktu terlalu lama untuk berkembang sehingga permainan terasa monoton.

Estimasi pendapatan pada setiap level gerbang tol menunjukkan potensi pendapatan pemain setelah pemain berhasil melakukan upgrade pada level tersebut. Dengan demikian, peningkatan level gerbang tol akan mengubah parameter operasional seperti kecepatan transaksi dan interval *spawn* kendaraan sehingga jumlah pendapatan pemain per menit juga meningkat.

Melalui implementasi balancing tersebut, pemain diharapkan dapat mempertimbangkan strategi operasional yang akan digunakan selama permainan berlangsung, seperti memilih untuk melakukan upgrade gerbang tol terlebih dahulu guna meningkatkan kecepatan transaksi kendaraan, atau membuka gerbang tol baru untuk meningkatkan kapasitas pelayanan kendaraan dan mengurangi antrean lalu lintas. Dengan demikian, proses pengambilan keputusan pemain menjadi bagian penting dalam mekanik simulasi pengelolaan gerbang tol yang dikembangkan pada penelitian ini.

4.3.6 Implementasi Sistem Pendapatan Lalu Lintas Tol

Sistem pendapatan lalu lintas tol digunakan untuk mengatur mekanisme pendapatan pemain berdasarkan kendaraan yang berhasil melewati ruas jalan tol selama permainan berlangsung. Sistem ini dirancang agar kondisi lalu lintas dan performa kendaraan memiliki pengaruh langsung terhadap jumlah pendapatan yang diperoleh pemain.

Implementasi sistem dilakukan menggunakan beberapa komponen utama seperti *HighwayCarSpawner*, *LaneController*, dan sistem kendaraan pada ruas jalan tol. Ketiga sistem tersebut saling terhubung untuk mengatur kemunculan kendaraan, kondisi jalur lalu lintas, serta proses perhitungan pendapatan pemain selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Pada gameplay lalu lintas tol, pemain memperoleh pendapatan ketika kendaraan berhasil mencapai ujung ruas jalan tol. Selain itu, pemain juga perlu menjaga kondisi jalur lalu lintas agar kendaraan dapat bergerak dengan optimal dan tidak mengalami hambatan yang dapat memengaruhi jumlah pendapatan permainan.

Implementasi sistem pendapatan lalu lintas pada penelitian ini dibagi menjadi tiga bagian utama, yaitu implementasi sistem *spawn* dan distribusi kendaraan, implementasi sistem maintenance jalur, dan implementasi sistem balancing lalu lintas. Ketiga sistem tersebut saling terhubung untuk menciptakan simulasi pengelolaan lalu lintas yang dinamis selama permainan berlangsung.

4.3.6.1 Implementasi Sistem Spawn dan Distribusi Kendaraan

Sistem spawn dan distribusi kendaraan pada ruas jalan tol berfungsi untuk menghasilkan kendaraan secara otomatis pada area jalan utama selama permainan berlangsung. Berbeda dengan sistem spawn pada gerbang tol, mekanisme ini berfokus pada pembentukan arus lalu lintas di sepanjang ruas jalan sehingga kendaraan dapat bergerak menuju tujuan masing-masing melalui jalur yang telah ditentukan. Implementasi sistem ini bertujuan menciptakan distribusi kendaraan yang bervariasi, menjaga kelancaran arus lalu lintas, serta mendukung simulasi kondisi jalan tol yang lebih dinamis.

Implementasi sistem spawn dan distribusi kendaraan menggunakan kombinasi tiga algoritma utama, yaitu Random Vehicle Selection Algorithm, Spawn Safety Detection, dan Waypoint Distribution. *Random Vehicle Selection Algorithm* digunakan untuk memilih jenis kendaraan yang akan dihasilkan secara acak dari kumpulan prefab yang tersedia sehingga variasi kendaraan pada ruas jalan menjadi lebih beragam. Selanjutnya, *Spawn Safety Detection* memastikan proses spawn hanya dilakukan apabila area spawn

berada dalam kondisi aman dan tidak terdapat kendaraan lain di sekitarnya. Setelah kendaraan berhasil dihasilkan, algoritma *Waypoint Distribution* mengatur jalur pergerakan kendaraan dengan menetapkan waypoint dan lane yang akan diikuti selama perjalanan. Kombinasi ketiga algoritma tersebut menghasilkan distribusi kendaraan yang lebih teratur, aman, dan menyerupai kondisi lalu lintas pada jalan tol sebenarnya.

Proses spawn kendaraan diawali dengan menjalankan proses spawn secara berkala menggunakan interval waktu yang dipilih secara acak. Sebelum kendaraan dihasilkan, sistem terlebih dahulu memeriksa kondisi area spawn untuk memastikan tidak terdapat kendaraan lain yang berada dalam jarak aman. Apabila area spawn dinyatakan aman, sistem akan memilih salah satu jenis kendaraan secara acak kemudian membuat objek kendaraan pada titik spawn yang telah ditentukan. Selanjutnya, kendaraan diberikan jalur pergerakan melalui waypoint serta informasi lane yang akan digunakan selama perjalanan. Proses tersebut berlangsung secara berulang selama sistem spawn dalam keadaan aktif sehingga arus kendaraan pada ruas jalan tetap terjaga secara dinamis.

```
int index = Random.Range(0, carPrefabs.Length);
GameObject car = Instantiate(carPrefabs[index], spawnPoint.position, Quaternion.identity);
CarAI ai = car.GetComponent<CarAI>();
```

Gambar 4.27 *Script Random Vehicle Selection Algorithm*

Gambar 4.27 menunjukkan implementasi Random Vehicle Selection Algorithm pada sistem spawn kendaraan ruas jalan tol. Sistem menggunakan fungsi Random.Range() untuk memilih salah satu prefab kendaraan secara acak dari kumpulan carPrefabs yang tersedia. Nilai indeks hasil pemilihan tersebut kemudian digunakan pada fungsi Instantiate() untuk menghasilkan objek kendaraan baru pada titik spawnPoint. Pendekatan ini memungkinkan variasi jenis kendaraan yang muncul pada ruas jalan sehingga distribusi lalu

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

lintas menjadi lebih beragam dan tidak bersifat monoton selama permainan berlangsung.

```
1 reference
bool IsSafeToSpawn()
{
    if (spawnPoint == null) return false;

    Collider[] hits = Physics.OverlapSphere(
        spawnPoint.position,
        detectionDistance,
        carLayer
    );

    return hits.Length == 0;
}
```

Gambar 4.28 *Script Spawn Safety Detection*

Gambar 4.28 menunjukkan implementasi Spawn Safety Detection yang digunakan untuk memastikan proses spawn kendaraan hanya dilakukan ketika area spawn berada dalam kondisi aman. Sistem memanfaatkan fungsi Physics.OverlapSphere() untuk mendeteksi keberadaan kendaraan lain di sekitar titik spawn berdasarkan nilai detectionDistance. Apabila tidak ditemukan objek kendaraan pada area tersebut, fungsi akan mengembalikan nilai true sehingga proses spawn dapat dilanjutkan. Sebaliknya, apabila masih terdapat kendaraan pada area spawn, sistem akan menunda proses spawn hingga kondisi kembali aman. Mekanisme ini membantu mengurangi potensi kendaraan saling bertumpuk maupun bertabrakan saat pertama kali muncul.

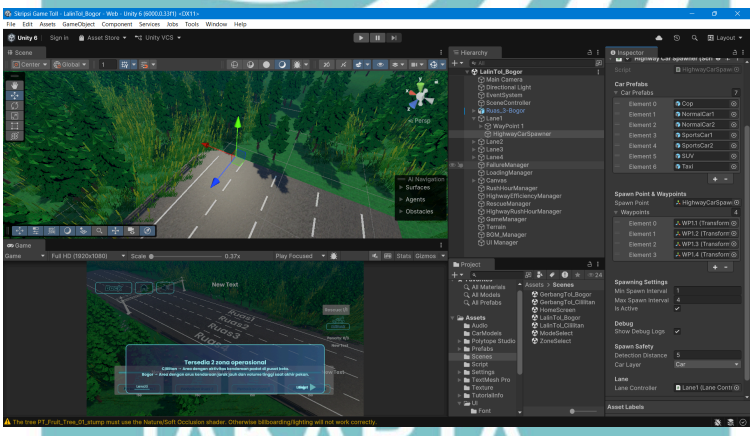
```
CarAI ai = car.GetComponent<CarAI>();
ai.currentLane = laneController;
if (ai != null)
{
    ai.waypoints = waypoints;
    ai.currentMode = CarAI.CarMode.Highway;

    spawnedCars.Add(car);

    if (showDebugLogs)
        Debug.Log($"🚗 Car spawned: {ai.carID}");
}
```


© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.29 *Script Waypoint Distribution*

Gambar 4.29 menunjukkan implementasi Waypoint Distribution pada sistem distribusi kendaraan ruas jalan tol. Setelah kendaraan berhasil dihasilkan, sistem mengambil komponen CarAI kemudian menetapkan jalur yang akan digunakan melalui variabel waypoints serta menentukan lane aktif menggunakan laneController. Selain itu, mode kendaraan diubah menjadi Highway agar kendaraan mengikuti perilaku pergerakan yang sesuai dengan kondisi ruas jalan tol. Selanjutnya, objek kendaraan ditambahkan ke dalam daftar spawnedCars sehingga seluruh kendaraan yang telah dihasilkan dapat dikelola selama permainan berlangsung. Implementasi ini memastikan setiap kendaraan memperoleh jalur pergerakan yang sesuai sejak proses spawn selesai dilakukan.



Gambar 4.30 Implementasi Sistem Spawn dan Distribusi Kendaraan

Pada tahap awal pengembangan, sistem spawn kendaraan pada ruas jalan menggunakan interval spawn yang relatif tetap sehingga kemunculan kendaraan terlihat kurang bervariasi dan pada beberapa kondisi menyebabkan penumpukan kendaraan di area spawn. Selain itu, proses spawn yang tidak mempertimbangkan kondisi sekitar berpotensi menghasilkan kendaraan pada posisi yang terlalu dekat dengan kendaraan lain. Berdasarkan hasil pengujian, sistem kemudian dikembangkan dengan menerapkan interval spawn secara acak serta mekanisme *Spawn Safety*

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Detection untuk memastikan kendaraan hanya dihasilkan ketika area spawn berada dalam kondisi aman. Pengujian juga dilakukan terhadap variasi jenis kendaraan dan distribusi waypoint agar setiap kendaraan memperoleh jalur pergerakan yang sesuai sejak proses spawn berlangsung. Hasil iterasi menunjukkan bahwa kombinasi mekanisme tersebut mampu menghasilkan distribusi kendaraan yang lebih beragam, mengurangi potensi tabrakan pada area spawn, serta menciptakan arus lalu lintas yang lebih stabil dan realistis selama permainan berlangsung.

4.3.6.2 Implementasi Sistem Maintenance Jalur

Sistem maintenance jalur merupakan mekanisme yang digunakan untuk menjaga kondisi ruas jalan tol agar tetap berada dalam keadaan optimal selama permainan berlangsung. Setiap jalur memiliki tingkat kondisi (*health*) yang akan berkurang secara bertahap seiring berjalannya waktu sehingga pemain perlu melakukan maintenance secara berkala. Kondisi jalan yang menurun akan memengaruhi kelancaran lalu lintas, seperti berkurangnya kecepatan kendaraan dan meningkatnya risiko gangguan kendaraan. Implementasi sistem ini bertujuan mendorong pemain untuk mengelola kondisi jalan secara aktif sehingga operasional jalan tol dapat berjalan dengan lebih efektif dan efisien.

Implementasi sistem maintenance jalur menggunakan kombinasi tiga algoritma utama, yaitu Health Decay Algorithm, Threshold-Based Condition Evaluation, dan Maintenance Recovery Algorithm. *Health Decay Algorithm* digunakan untuk mengurangi kondisi jalan secara bertahap berdasarkan waktu sehingga setiap jalur mengalami penurunan kualitas selama permainan berlangsung. Selanjutnya, *Threshold-Based Condition Evaluation* mengevaluasi kondisi jalan berdasarkan nilai *health* untuk

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

menentukan pengaruhnya terhadap kecepatan kendaraan dan kemungkinan terjadinya gangguan kendaraan. Apabila pemain melakukan maintenance, *Maintenance Recovery Algorithm* akan memulihkan kondisi jalan sesuai nilai perbaikan yang telah ditentukan. Kombinasi ketiga algoritma tersebut menghasilkan mekanisme pemeliharaan jalan yang dinamis dan memberikan dampak langsung terhadap kondisi lalu lintas dalam permainan.

Proses maintenance jalur diawali dengan pengurangan nilai *health* secara berkala selama permainan berlangsung. Setelah nilai *health* berubah, sistem akan mengevaluasi kondisi jalan untuk menentukan efek yang diberikan terhadap kendaraan yang melintasi jalur tersebut. Semakin rendah kondisi jalan, semakin besar pengaruhnya terhadap kecepatan kendaraan dan kemungkinan terjadinya gangguan. Ketika pemain melakukan maintenance, sistem akan memeriksa kecukupan saldo, kemudian mengurangi biaya maintenance serta menambahkan nilai *health* sesuai jumlah perbaikan yang ditentukan. Selanjutnya kondisi jalan diperbarui dan disimpan sehingga perubahan tersebut dapat langsung memengaruhi jalannya simulasi lalu lintas.

```
1 reference
void DecayHealth()
{
    decayTimer += Time.deltaTime;

    if (decayTimer ≥ 1f)
    {
        decayTimer = 0f;

        currentHealth -= healthDecayRate;

        currentHealth =
            Mathf.Clamp(currentHealth, 0, maxHealth);

        SaveLaneHealth();
    }
}
```

Gambar 4.31 *Script Health Decay Algorithm*

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.31 menunjukkan implementasi Health Decay Algorithm yang digunakan untuk mengurangi kondisi jalan secara bertahap selama permainan berlangsung. Sistem memanfaatkan variabel decayTimer untuk menghitung waktu, kemudian setiap satu detik nilai currentHealth akan dikurangi sebesar healthDecayRate. Selanjutnya, fungsi Mathf.Clamp() digunakan untuk membatasi nilai health agar tetap berada pada rentang 0 hingga maxHealth. Setelah kondisi jalan diperbarui, sistem memanggil fungsi SaveLaneHealth() untuk menyimpan perubahan kondisi jalan sehingga dapat dipertahankan selama permainan berlangsung.

```
1 reference
public void RepairLane()
{
    if (MoneyManager.instance == null) return;

    if (MoneyManager.instance.money < maintenanceCost)
    {
        Debug.Log("❌ Uang maintenance tidak cukup!");
        return;
    }

    MoneyManager.instance.SpendMoney(maintenanceCost);

    currentHealth += repairAmount;

    currentHealth =
        Mathf.Clamp(currentHealth, 0, maxHealth);
        SaveLaneHealth();

    Debug.Log($"🛠 {laneName} repaired!");
}
```

Gambar 4.32 *Script Maintenance Recovery Algorithm*

Gambar 4.32 menunjukkan implementasi Maintenance Recovery Algorithm pada proses perbaikan kondisi jalan. Sistem terlebih dahulu memeriksa ketersediaan saldo pemain sebelum maintenance dilakukan. Apabila saldo mencukupi, sistem akan mengurangi biaya maintenance menggunakan fungsi SpendMoney(), kemudian menambahkan nilai repairAmount ke dalam currentHealth. Selanjutnya, fungsi Mathf.Clamp() memastikan nilai *health* tidak melebihi batas maksimum, kemudian kondisi jalan disimpan melalui fungsi SaveLaneHealth(). Mekanisme ini memastikan proses maintenance hanya dapat dilakukan apabila pemain

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

memiliki sumber daya yang cukup serta menjaga konsistensi data kondisi jalan.

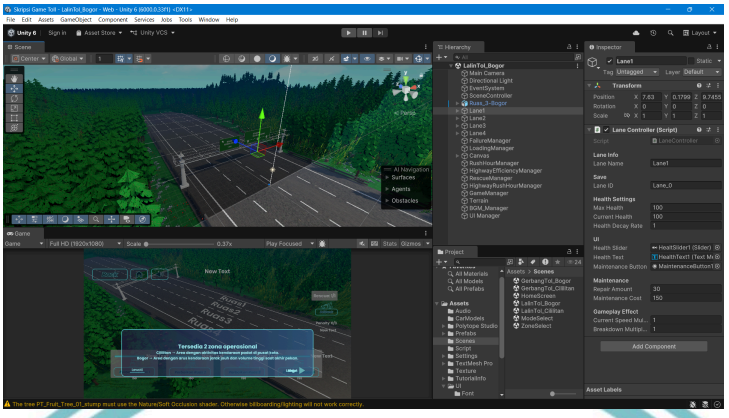
```
1 reference
void UpdateGameplayEffect()
{
    float healthPercent =
        currentHealth / maxHealth;

    // HEALTH BAGUS
    if (healthPercent > 0.7f)
    {
        currentSpeedMultiplier = 1f;
        breakdownMultiplier = 1f;
    }

    // HEALTH SEDANG
    else if (healthPercent > 0.4f)
    {
        currentSpeedMultiplier = 0.8f;
        breakdownMultiplier = 2f;
    }

    // HEALTH KRITIS
    else
    {
        currentSpeedMultiplier = 0.5f;
        breakdownMultiplier = 4f;
    }
}
```

Gambar 4.33 *Script Maintenance Recovery Algorithm*

Gambar 4.33 menunjukkan implementasi Threshold-Based Condition Evaluation yang digunakan untuk menentukan pengaruh kondisi jalan terhadap permainan. Sistem menghitung persentase kondisi jalan melalui perbandingan antara currentHealth dan maxHealth, kemudian membagi kondisi jalan ke dalam tiga kategori, yaitu baik, sedang, dan kritis. Setiap kategori memiliki nilai currentSpeedMultiplier dan breakdownMultiplier yang berbeda sehingga kondisi jalan secara langsung memengaruhi kecepatan kendaraan serta kemungkinan terjadinya gangguan kendaraan. Pendekatan ini memungkinkan perubahan kondisi jalan memberikan dampak yang nyata terhadap dinamika lalu lintas selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.34 Implementasi Sistem Maintenance Jalur pada Unity

Pada tahap awal pengembangan, kondisi jalan mengalami penurunan dengan laju yang relatif tinggi sehingga pemain harus melakukan maintenance terlalu sering dan biaya operasional menjadi kurang seimbang dengan pendapatan yang diperoleh. Selain itu, pengaruh kondisi jalan terhadap kecepatan kendaraan dan kemungkinan terjadinya gangguan belum memberikan perbedaan yang signifikan pada setiap tingkat kerusakan jalan. Berdasarkan hasil pengujian, dilakukan penyesuaian terhadap nilai healthDecayRate, repairAmount, serta batas persentase kondisi jalan pada setiap kategori agar penurunan kualitas jalan berlangsung secara bertahap dan maintenance menjadi keputusan strategis bagi pemain. Hasil iterasi menunjukkan bahwa mekanisme maintenance mampu menjaga keseimbangan antara biaya operasional dan kondisi lalu lintas, di mana jalan dengan kondisi baik dapat mempertahankan kelancaran arus kendaraan, sedangkan jalan yang tidak dirawat akan menurunkan kecepatan kendaraan dan meningkatkan risiko gangguan. Dengan penyesuaian tersebut, sistem maintenance berhasil memberikan dampak yang nyata terhadap pengelolaan lalu lintas sekaligus mendorong pemain untuk melakukan pemeliharaan jalan secara berkala.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3.6.3 Implementasi Sistem Balancing Lalu Lintas

Sistem balancing lalu lintas digunakan untuk menjaga keseimbangan antara kondisi jalur, performa kendaraan, tingkat pendapatan pemain, dan tingkat kesulitan permainan selama simulasi berlangsung. Implementasi balancing dilakukan agar kondisi lalu lintas tetap memberikan tantangan kepada pemain tanpa menyebabkan permainan menjadi terlalu mudah maupun terlalu sulit.

Pada penelitian ini, proses balancing lalu lintas dilakukan menggunakan konsep konversi waktu seperti pada balancing gerbang tol. Konsep tersebut digunakan untuk menghitung hubungan antara tingkat kerusakan jalur, biaya maintenance, performa kendaraan, dan pengaruhnya terhadap pendapatan pemain selama permainan berlangsung.

Implementasi balancing dilakukan menggunakan beberapa parameter utama, seperti *health decay rate*, *maintenance cost*, *repair amount*, interval *spawn* kendaraan, dan nilai pendapatan kendaraan.

Untuk menghitung estimasi waktu kerusakan jalur, digunakan rumus berikut:

$$Waktu\ Kerusakan = \frac{MaxHealth}{HealthDecayRate}$$

Keterangan:

1. MaxHealth = kondisi kesehatan maksimum jalur
2. HealthDecayRate = tingkat penurunan kesehatan jalur setiap detik

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Sebagai contoh, Lane 1 memiliki nilai *health decay rate* sebesar 1 dengan kondisi kesehatan maksimum sebesar 100. Maka estimasi waktu hingga kondisi jalur habis adalah:

$$\frac{100}{1} = 100\ detik$$

Sementara itu, Lane 4 memiliki nilai *health decay rate* sebesar 2 sehingga waktu kerusakan jalur menjadi lebih cepat:

$$\frac{100}{2} = 50\ detik$$

Hasil tersebut menunjukkan bahwa Lane 4 mengalami kerusakan dua kali lebih cepat dibanding Lane 1 karena digunakan oleh kendaraan kategori 2 dan kategori 3 dengan beban operasional yang lebih tinggi.

Selain itu, sistem maintenance juga memengaruhi balancing lalu lintas. Pada penelitian ini, setiap maintenance memulihkan kondisi jalur sebesar 30 dengan biaya operasional sebesar 150. Untuk menghitung estimasi durasi efek maintenance, digunakan rumus berikut:

$$Durasi\ Maintenance = \frac{RepairAmount}{HealthDecayRate}$$

Keterangan:

1. RepairAmount = jumlah pemulihan kesehatan jalur
2. HealthDecayRate = tingkat penurunan kesehatan jalur

Sebagai contoh, pada Lane 1:

$$\frac{30}{1} = 30\ detik$$

Sedangkan pada Lane 4:

$$\frac{30}{2} = 15\ detik$$

Hasil tersebut menunjukkan bahwa maintenance pada Lane 4 memiliki durasi efektivitas yang lebih singkat dibanding Lane 1 sehingga pemain perlu melakukan maintenance lebih sering pada jalur dengan tingkat kerusakan yang lebih tinggi.

Selain memengaruhi kondisi jalur, sistem kesehatan jalur juga memengaruhi performa kendaraan selama permainan berlangsung. Semakin rendah kondisi kesehatan jalur, maka kendaraan akan bergerak lebih lambat dan risiko kendaraan mogok menjadi lebih tinggi. Kondisi tersebut menyebabkan jumlah kendaraan yang berhasil mencapai tujuan menjadi lebih sedikit sehingga pendapatan pemain ikut menurun.

Tabel 4.18 menunjukkan implementasi balancing kondisi jalur lalu lintas pada penelitian ini.

Tabel 4.11 Balancing kondisi jalur

Lane	Kategori Kendaraan	Health Decay Rate	Estimasi Waktu Kerusakan
Lane 1	Kategori 1	1	100 detik
Lane 2	Kategori 1	1	100 detik
Lane 3	Kategori 2 dan 3	1,8	56 detik
Lane 4	Kategori 2 dan 3	2	50 detik

Tabel 4.19 menunjukkan implementasi balancing maintenance jalur.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 4.12 Implementasi balancing maintenance

Lane	Repair Amount	Maintenance Cost	Estimasi Durasi Maintenance
Lane 1	30	150	30 detik
Lane 2	30	150	30 detik
Lane 3	30	150	17 detik
Lane 4	30	150	15 detik

Berdasarkan hasil *playtesting* internal, parameter tersebut dinilai mampu menciptakan tekanan gameplay yang cukup seimbang. Pada pengujian awal, nilai *health decay rate* yang terlalu tinggi menyebabkan jalur mengalami kerusakan terlalu cepat sehingga pemain kesulitan mempertahankan kondisi lalu lintas dan pendapatan permainan menurun secara drastis. Sebaliknya, nilai *decay* yang terlalu rendah menyebabkan sistem maintenance menjadi jarang digunakan sehingga gameplay terasa kurang menantang.

Selain itu, balancing biaya maintenance juga dilakukan agar pemain perlu mempertimbangkan penggunaan uang operasional secara strategis. Maintenance yang terlalu murah menyebabkan pemain dapat mempertahankan seluruh jalur tanpa kesulitan, sedangkan biaya maintenance yang terlalu tinggi menyebabkan

pemain kesulitan menjaga kestabilan lalu lintas selama permainan berlangsung.

Melalui implementasi balancing tersebut, pemain diharapkan dapat mempertimbangkan strategi operasional yang digunakan selama permainan berlangsung, seperti menentukan jalur yang perlu diprioritaskan untuk maintenance, menjaga kestabilan pendapatan lalu lintas, serta mempertahankan kondisi operasional jalan tol agar kendaraan tetap dapat bergerak secara optimal.

4.3.6.4 Implementasi Sistem Rescue Kendaraan

Sistem rescue kendaraan merupakan mekanisme yang digunakan untuk menangani kendaraan yang mengalami gangguan selama melintasi ruas jalan tol. Ketika sebuah kendaraan mengalami kerusakan, kendaraan tersebut akan berhenti sementara dan memunculkan proses penyelamatan yang dapat dilakukan oleh pemain menggunakan unit rescue yang tersedia. Apabila kendaraan berhasil diselamatkan sebelum batas waktu berakhir, kendaraan dapat kembali beroperasi tanpa memberikan penalti kepada pemain. Sebaliknya, apabila proses rescue tidak dilakukan hingga waktu habis, sistem akan memberikan penalti yang memengaruhi kondisi permainan. Implementasi sistem ini bertujuan memberikan tantangan tambahan dalam pengelolaan lalu lintas sekaligus mendorong pemain untuk mengatur penggunaan unit rescue secara efektif.

Implementasi sistem rescue kendaraan menggunakan kombinasi tiga algoritma utama, yaitu Breakdown Countdown Algorithm, Rescue Process Management, dan Rescue Resource Management. *Breakdown Countdown Algorithm* digunakan untuk menghitung batas waktu penanganan kendaraan yang mengalami kerusakan sebelum sistem memberikan penalti secara otomatis. Selanjutnya, *Rescue Process Management* mengatur jalannya proses

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

penyelamatan kendaraan mulai dari penugasan unit rescue hingga kendaraan kembali beroperasi. Untuk mendukung pengelolaan sumber daya, *Rescue Resource Management* mengatur jumlah unit rescue yang tersedia, status penggunaannya, serta mekanisme pembelian unit rescue tambahan oleh pemain. Kombinasi ketiga algoritma tersebut menghasilkan proses penyelamatan kendaraan yang lebih terstruktur, responsif, dan terintegrasi dengan sistem manajemen lalu lintas dalam permainan.

Proses rescue kendaraan diawali ketika sebuah kendaraan mengalami kerusakan sehingga sistem mengaktifkan penghitung waktu (*countdown*) sebagai batas maksimal penanganan. Selama proses tersebut berlangsung, pemain dapat menggunakan unit rescue yang masih tersedia untuk melakukan penyelamatan terhadap kendaraan yang mengalami gangguan. Setelah proses rescue dimulai, kendaraan akan berada dalam status penyelamatan hingga waktu rescue selesai, kemudian kendaraan kembali beroperasi dan status gangguan dihapus dari sistem. Apabila tidak terdapat unit rescue yang tersedia atau pemain tidak melakukan penyelamatan hingga waktu habis, sistem akan memberikan penalti sesuai ketentuan permainan. Selain itu, pemain juga dapat menambah jumlah unit rescue melalui mekanisme pembelian sehingga kapasitas penanganan kendaraan dapat ditingkatkan sesuai kebutuhan.

```
void Update()
{
    if (isResolved) return;

    if (!isBeingRescued)
    {
        countdown -= Time.deltaTime;
    }

    UpdateCountdownUI();

    if (countdown <= 0)
    {
        AutoRecover();
    }
}
```


© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.35 *Script Breakdown Countdown Algorithm*

Gambar 4.35 menunjukkan implementasi Breakdown Countdown Algorithm yang digunakan untuk mengatur batas waktu penanganan kendaraan yang mengalami kerusakan. Selama kendaraan belum berhasil diselamatkan (isResolved) dan belum berada dalam proses rescue (isBeingRescued), sistem akan mengurangi nilai countdown secara bertahap menggunakan Time.deltaTime. Selanjutnya, fungsi UpdateCountdownUI() digunakan untuk memperbarui tampilan penghitung waktu pada layar. Apabila nilai countdown mencapai nol, sistem akan menjalankan fungsi AutoRecover() untuk mengakhiri status kerusakan kendaraan sekaligus memberikan konsekuensi sesuai mekanisme permainan. Pendekatan ini memastikan setiap kendaraan yang mengalami gangguan memiliki batas waktu penanganan yang jelas sehingga pemain dituntut mengambil keputusan secara cepat.

```
1 reference
public void StartRescue(BrokenCar targetCar)
{
    if (!CanRescue()) return;

    StartCoroutine(RescueProcess(targetCar));
}

1 reference
IEnumerator RescueProcess(BrokenCar targetCar)
{
    busyRescue++;
    Debug.Log($"🚒 Available Rescue: {GetAvailableRescue()}");

    Debug.Log($"🚒 Rescue started! Busy: {busyRescue}");

    yield return new WaitForSeconds(rescueDuration);

    if (targetCar ≠ null)
    {
        targetCar.FinishRescue();
    }

    busyRescue--;

    Debug.Log($"✅ Rescue finished! Available: {GetAvailableRescue()}");
}
```

Gambar 4.36 *Script Rescue Process Management*

Gambar 4.36 menunjukkan implementasi Rescue Process Management yang mengatur proses penyelamatan kendaraan. Sistem terlebih dahulu memeriksa ketersediaan unit rescue melalui

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

fungsi CanRescue(). Apabila masih terdapat unit yang tersedia, sistem menjalankan *coroutine* RescueProcess() untuk memulai proses penyelamatan. Selama proses berlangsung, jumlah unit rescue yang sedang digunakan (busyRescue) akan bertambah sehingga tidak dapat digunakan oleh kendaraan lain. Setelah waktu penyelamatan (rescueDuration) selesai, fungsi FinishRescue() dipanggil untuk mengembalikan kendaraan ke kondisi normal, kemudian jumlah unit rescue yang sedang digunakan dikurangi sehingga dapat digunakan kembali pada proses penyelamatan berikutnya.

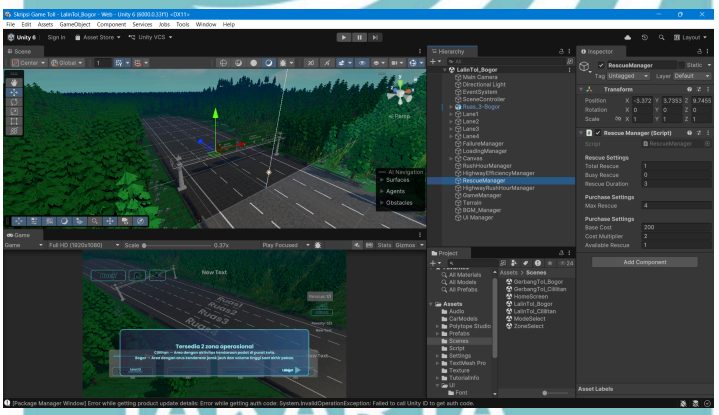
```
public bool BuyRescue()
{
    if (availableRescue >= maxRescue)
    {
        Debug.Log("❌ Rescue sudah max!");
        return false;
    }

    int cost = GetRescueCost();

    if (MoneyManager.instance == null)
        return false;

    if (MoneyManager.instance.money < cost)
    {
        Debug.Log($"❌ Uang tidak cukup! Need: {cost}");
        return false;
    }

    MoneyManager.instance.SpendMoney(cost);

    availableRescue++;
    totalRescue++;
    purchaseCount++;

    PlayerPrefs.SetInt(
        GetRescueKey(),
        availableRescue
    );
    PlayerPrefs.SetInt(
        GetPurchaseKey(),
        purchaseCount
    );

    PlayerPrefs.Save();
    Debug.Log($"✅ Rescue purchased! Total: {totalRescue}");
    return true;
}
```

Gambar 4.37 *Script Rescue Resource Management*

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.37 menunjukkan implementasi Rescue Resource Management yang digunakan untuk mengelola jumlah unit rescue yang dimiliki pemain. Sistem terlebih dahulu memeriksa apakah jumlah rescue telah mencapai batas maksimum (maxRescue) serta memastikan saldo pemain mencukupi untuk melakukan pembelian. Apabila kedua kondisi tersebut terpenuhi, sistem akan mengurangi saldo pemain melalui fungsi SpendMoney(), kemudian menambahkan jumlah unit rescue (availableRescue dan totalRescue) serta memperbarui jumlah pembelian (purchaseCount). Seluruh perubahan tersebut selanjutnya disimpan menggunakan PlayerPrefs sehingga kapasitas rescue tetap dipertahankan pada sesi permainan berikutnya. Mekanisme ini memungkinkan pemain meningkatkan kemampuan penanganan kendaraan sesuai perkembangan permainan.



Gambar 4.38 Implementasi Sistem Rescue Kendaraan pada Unity

Pada tahap awal pengembangan, sistem rescue kendaraan menggunakan jumlah unit rescue yang terbatas serta waktu penyelamatan yang relatif singkat sehingga beberapa kendaraan sering tidak tertangani sebelum batas waktu berakhir. Kondisi tersebut menyebabkan penalti terjadi terlalu sering dan mengurangi keseimbangan permainan. Berdasarkan hasil pengujian, dilakukan penyesuaian terhadap jumlah maksimum unit rescue, durasi proses penyelamatan, serta biaya pembelian rescue tambahan agar pemain

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

memiliki pilihan strategi dalam menangani kendaraan yang mengalami gangguan. Selain itu, mekanisme peningkatan biaya pembelian rescue diterapkan secara bertahap sehingga pemain tidak dapat memperoleh terlalu banyak unit rescue dalam waktu singkat. Hasil iterasi menunjukkan bahwa kombinasi batas jumlah rescue, durasi penyelamatan, dan skala biaya pembelian mampu menciptakan keseimbangan antara tingkat kesulitan permainan dan kemampuan pemain dalam mengelola gangguan kendaraan, sehingga sistem rescue memberikan tantangan yang tetap adil sekaligus mendukung kelancaran lalu lintas selama permainan berlangsung.

4.3.6.5 Implementasi Sistem Efisiensi Jalan Tol

Sistem efisiensi jalan tol merupakan mekanisme yang digunakan untuk mengevaluasi tingkat kinerja operasional jalan tol secara keseluruhan selama permainan berlangsung. Nilai efisiensi dihitung berdasarkan kondisi setiap jalur, jumlah kendaraan yang mengalami gangguan, serta jumlah penalti yang telah diterima pemain. Informasi tersebut kemudian disajikan dalam bentuk persentase sebagai indikator performa pengelolaan jalan tol. Implementasi sistem ini bertujuan memberikan umpan balik kepada pemain mengenai efektivitas strategi yang diterapkan dalam menjaga kelancaran lalu lintas dan kondisi infrastruktur jalan.

Implementasi sistem efisiensi jalan tol menggunakan dua algoritma utama, yaitu Weighted Efficiency Calculation Algorithm dan Threshold-Based Efficiency Classification. *Weighted Efficiency Calculation Algorithm* digunakan untuk menghitung nilai efisiensi berdasarkan rata-rata kondisi seluruh jalur (*lane health*) yang kemudian dikurangi dengan pengaruh jumlah kendaraan yang mengalami gangguan (*breakdown*) serta jumlah penalti (*strike*)

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

yang diterima pemain. Selanjutnya, *Threshold-Based Efficiency Classification* mengelompokkan nilai efisiensi ke dalam beberapa kategori berdasarkan batas persentase tertentu sehingga kondisi operasional jalan tol dapat ditampilkan secara lebih informatif kepada pemain. Kombinasi kedua algoritma tersebut menghasilkan indikator efisiensi yang mampu merepresentasikan kondisi lalu lintas secara menyeluruh selama permainan berlangsung.

```
1 reference
void CalculateEfficiency()
{
    float totalHealth = 0f;

    foreach (LaneController lane in lanes)
    {
        if (lane != null)
        {
            totalHealth += lane.currentHealth;
        }
    }

    float averageHealth =
        totalHealth / lanes.Length;

    int activeBreakdowns =
        FindObjectsByType<BrokenCar>(
            FindObjectsSortMode.None
        ).Length;

    int strikeCount = 0;

    if (FailureManager.Instance != null)
    {
        strikeCount =
            FailureManager.Instance.currentStrike;
    }

    currentEfficiency =
        averageHealth
        - (activeBreakdowns * 5f)
        - (strikeCount * 15f);

    currentEfficiency =
        Mathf.Clamp(currentEfficiency, 0, 100);
}
```

Gambar 4.39 *Script Weighted Efficiency Calculation Algorithm*

Gambar 4.39 menunjukkan implementasi Weighted Efficiency Calculation Algorithm yang digunakan untuk menghitung tingkat efisiensi operasional jalan tol. Sistem terlebih dahulu menghitung total kondisi seluruh jalur (lane) kemudian memperoleh nilai rata-ratanya melalui variabel averageHealth. Selanjutnya, sistem menghitung jumlah kendaraan yang sedang mengalami gangguan (activeBreakdowns) serta jumlah penalti (strikeCount) yang

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

dimiliki pemain. Ketiga komponen tersebut dikombinasikan dalam perhitungan currentEfficiency, di mana setiap kendaraan yang mengalami gangguan mengurangi nilai efisiensi sebesar 5%, sedangkan setiap penalti mengurangi nilai efisiensi sebesar 15%. Setelah proses perhitungan selesai, fungsi Mathf.Clamp() digunakan untuk membatasi nilai efisiensi pada rentang 0 hingga 100 persen sehingga indikator efisiensi tetap berada dalam batas yang valid.

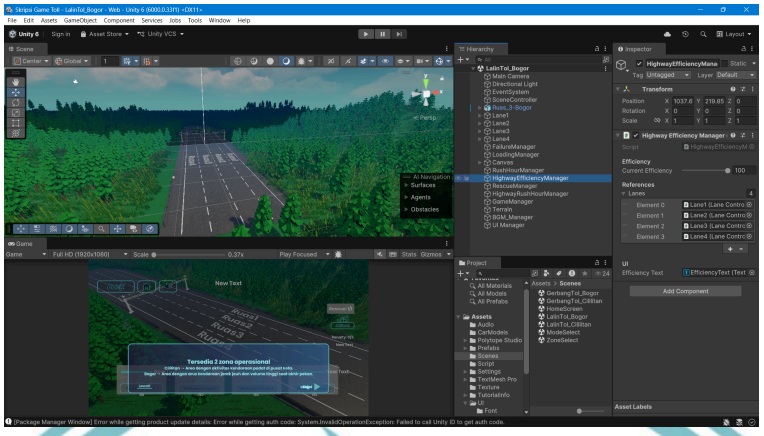
```
1 reference
void UpdateUI()
{
    if (efficiencyText == null) return;

    efficiencyText.text =
        $"Efficiency: {Mathf.RoundToInt(currentEfficiency)}%";

    if (currentEfficiency ≥ 90)
    {
        efficiencyText.color = Color.green;
    }
    else if (currentEfficiency ≥ 70)
    {
        efficiencyText.color = Color.yellow;
    }
    else
    {
        efficiencyText.color = Color.red;
    }
}
```

Gambar 4.40 *Script Threshold-Based Efficiency Classification*

Gambar 4.40 menunjukkan implementasi Threshold-Based Efficiency Classification yang digunakan untuk mengelompokkan nilai efisiensi jalan tol ke dalam beberapa kategori. Setelah nilai currentEfficiency diperoleh, sistem menampilkan persentase efisiensi pada antarmuka permainan menggunakan komponen efficiencyText. Selanjutnya, sistem menentukan kategori efisiensi berdasarkan nilai ambang (threshold), yaitu hijau apabila efisiensi mencapai 90% atau lebih, kuning apabila berada pada rentang 70% hingga kurang dari 90%, dan merah apabila berada di bawah 70%. Pendekatan ini memudahkan pemain dalam memantau kondisi operasional jalan tol secara cepat melalui indikator visual yang ditampilkan selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



Gambar 4.41 Implementasi Sistem Efisiensi Jalan Tol pada Unity

Pada tahap awal pengembangan, sistem efisiensi jalan tol hanya mempertimbangkan kondisi jalan sebagai indikator utama sehingga perubahan kondisi lalu lintas belum sepenuhnya tercermin pada nilai efisiensi yang ditampilkan. Akibatnya, keberadaan kendaraan yang mengalami gangguan maupun penalti yang diterima pemain belum memberikan pengaruh yang signifikan terhadap hasil evaluasi sistem. Berdasarkan hasil pengujian, mekanisme perhitungan kemudian dikembangkan dengan menerapkan pembobotan terhadap rata-rata kondisi jalan, jumlah kendaraan yang mengalami gangguan (*breakdown*), serta jumlah penalti (*strike*) yang diperoleh pemain. Selain itu, batas klasifikasi efisiensi juga disesuaikan menjadi tiga kategori agar perubahan kondisi operasional jalan tol dapat terlihat dengan lebih jelas melalui indikator visual. Hasil iterasi menunjukkan bahwa sistem efisiensi mampu merepresentasikan kondisi operasional jalan tol secara lebih akurat, sehingga pemain dapat mengevaluasi efektivitas strategi pengelolaan lalu lintas dan menentukan tindakan yang diperlukan untuk mempertahankan kinerja jalan tol selama permainan berlangsung.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.3.7 Implementasi Sistem Penyimpanan Data

Pada tahap implementasi awal permainan, seluruh kondisi permainan akan kembali ke kondisi awal setiap kali pemain berpindah scene maupun membuka kembali permainan. Kondisi tersebut menyebabkan progres permainan seperti jumlah uang pemain, kondisi jalur lalu lintas, level gerbang tol, serta antrean kendaraan tidak dapat dipertahankan selama simulasi berlangsung. Permasalahan tersebut menyebabkan proses simulasi manajemen operasional menjadi kurang optimal karena pemain harus memulai ulang kondisi permainan dari awal.

Berdasarkan permasalahan tersebut, pada penelitian ini diterapkan sistem penyimpanan data menggunakan *PlayerPrefs* pada Unity. Metode ini digunakan untuk menyimpan data permainan secara lokal sehingga kondisi permainan dapat dimuat kembali ketika permainan dijalankan ulang maupun ketika pemain berpindah scene.

Implementasi sistem penyimpanan data bertujuan untuk mempertahankan data operasional permainan seperti jumlah uang pemain, kondisi jalur lalu lintas, dan perkembangan gerbang tol selama simulasi berlangsung. Sistem ini juga terhubung dengan berbagai mekanik utama permainan seperti sistem pendapatan, maintenance jalur, dan upgrade gerbang tol sehingga setiap perubahan kondisi permainan dapat disimpan secara otomatis.

Dengan diterapkannya sistem penyimpanan data menggunakan *PlayerPrefs*, pemain dapat melanjutkan permainan sesuai kondisi terakhir tanpa kehilangan progres yang telah dicapai selama permainan berlangsung.

Pada penelitian ini, implementasi sistem penyimpanan data dibagi menjadi tiga bagian utama, yaitu implementasi penyimpanan pendapatan pemain, implementasi penyimpanan kondisi jalur lalu lintas, dan implementasi penyimpanan progres gerbang tol. Ketiga sistem tersebut saling terhubung

untuk menjaga konsistensi kondisi permainan selama simulasi berlangsung.

4.3.7.1 Implementasi Penyimpanan Pendapatan

Sistem penyimpanan pendapatan merupakan mekanisme yang digunakan untuk menyimpan dan memulihkan jumlah uang yang dimiliki pemain selama permainan berlangsung. Setiap perubahan pendapatan akibat transaksi, baik berupa penambahan maupun pengurangan uang, akan disimpan secara otomatis sehingga data tetap tersedia ketika permainan ditutup maupun dijalankan kembali. Implementasi sistem ini bertujuan menjaga konsistensi data ekonomi pemain serta memastikan perkembangan permainan dapat dilanjutkan tanpa kehilangan informasi pendapatan yang telah diperoleh.

Implementasi sistem penyimpanan pendapatan menggunakan dua algoritma utama, yaitu Persistent Money Storage Algorithm dan Money Transaction Management Algorithm. *Persistent Money Storage Algorithm* digunakan untuk menyimpan serta memuat kembali data pendapatan pemain menggunakan media penyimpanan lokal (*PlayerPrefs*) sehingga nilai uang tetap dipertahankan pada setiap sesi permainan. Selanjutnya, *Money Transaction Management Algorithm* mengatur proses penambahan maupun pengurangan uang berdasarkan aktivitas pemain selama permainan berlangsung. Kombinasi kedua algoritma tersebut memastikan seluruh transaksi ekonomi dapat diproses secara konsisten sekaligus disimpan secara otomatis.

Proses penyimpanan pendapatan diawali ketika permainan dijalankan dengan memuat jumlah uang terakhir yang tersimpan pada media penyimpanan lokal. Selanjutnya, setiap kali pemain memperoleh pendapatan ataupun mengeluarkan biaya operasional, sistem akan memperbarui nilai uang sesuai transaksi yang terjadi.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Setelah nilai uang berubah, sistem secara otomatis menyimpan data terbaru ke dalam penyimpanan lokal sehingga perkembangan ekonomi pemain tetap tersimpan. Dengan mekanisme tersebut, jumlah pendapatan pemain dapat dipulihkan kembali ketika permainan dijalankan pada sesi berikutnya tanpa memerlukan proses penyimpanan secara manual.

```
0 references | Unity Message
void Start()
{
    money = PlayerPrefs.GetInt("Money", startingMoney);
}
3 references
public void ResetMoney()
{
    money = startingMoney;

    PlayerPrefs.SetInt("Money", money);

    Debug.Log($"💰 Money di-reset ke {money}");
}
```

Gambar 4.43 *Script Persistent Money Storage Algorithm*

Gambar 4.43 menunjukkan implementasi Persistent Money Storage Algorithm yang digunakan untuk memuat dan menyimpan data pendapatan pemain pada media penyimpanan lokal menggunakan PlayerPrefs. Ketika permainan dimulai, sistem memanggil fungsi PlayerPrefs.GetInt() untuk mengambil nilai uang terakhir yang tersimpan dengan menggunakan kunci "Money". Apabila data belum tersedia, sistem akan menggunakan nilai awal yang tersimpan pada variabel startingMoney. Selain itu, sistem juga menyediakan fungsi ResetMoney() untuk mengembalikan jumlah uang ke nilai awal, kemudian menyimpan kembali nilai tersebut melalui fungsi PlayerPrefs.SetInt(). Dengan mekanisme ini, data pendapatan pemain dapat dipertahankan maupun dikembalikan ke kondisi awal sesuai kebutuhan permainan.

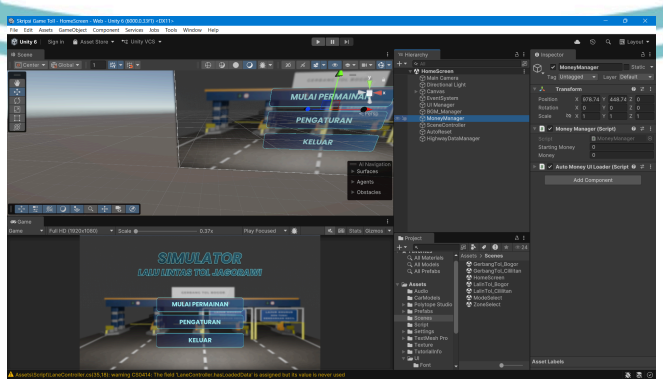
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
2 references
public void AddMoney(int amount)
{
    money += amount;
    PlayerPrefs.SetInt("Money", money);
}

5 references
public void SpendMoney(int amount)
{
    money -= amount;
    PlayerPrefs.SetInt("Money", money);

    if (money < 0 && GameManager.instance != null)
        GameManager.instance.GameOver();
}
```

Gambar 4.44 *Script Persistent Money Storage Algorithm*

Gambar 4.44 menunjukkan implementasi Money Transaction Management Algorithm yang digunakan untuk mengelola transaksi pendapatan selama permainan berlangsung. Ketika pemain memperoleh pendapatan, fungsi AddMoney() akan menambahkan nilai uang sesuai parameter amount kemudian langsung menyimpannya menggunakan PlayerPrefs.SetInt(). Sebaliknya, ketika pemain mengeluarkan biaya operasional, fungsi SpendMoney() akan mengurangi jumlah uang dan kembali memperbarui data yang tersimpan pada PlayerPrefs. Selain melakukan penyimpanan secara otomatis, sistem juga memeriksa apakah nilai uang menjadi kurang dari nol. Apabila kondisi tersebut terjadi, fungsi GameOver() akan dipanggil untuk mengakhiri permainan. Pendekatan ini memastikan setiap transaksi ekonomi langsung tercatat sekaligus menjaga konsistensi data pendapatan selama permainan berlangsung.



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.45 Implementasi Penyimpanan Pendapatan pada Unity

Pada tahap awal pengembangan, sistem penyimpanan pendapatan hanya memperbarui nilai uang selama permainan berlangsung tanpa memastikan data selalu tersimpan ketika terjadi perubahan transaksi. Kondisi tersebut berpotensi menyebabkan kehilangan data apabila permainan ditutup secara tiba-tiba atau berpindah ke sesi permainan berikutnya. Berdasarkan hasil pengujian, mekanisme penyimpanan kemudian dikembangkan dengan menerapkan proses penyimpanan otomatis setiap kali terjadi penambahan maupun pengurangan pendapatan menggunakan *PlayerPrefs*. Selain itu, sistem juga dilengkapi dengan mekanisme pemuatan data saat permainan dimulai serta fungsi untuk mengembalikan nilai pendapatan ke kondisi awal ketika permainan direset. Hasil iterasi menunjukkan bahwa sistem mampu menjaga konsistensi data pendapatan pada setiap sesi permainan, sehingga perkembangan ekonomi pemain dapat dipertahankan dengan baik tanpa memerlukan proses penyimpanan secara manual.

4.3.7.2 Implementasi Penyimpanan Kondisi Jalur

Sistem penyimpanan kondisi jalur merupakan mekanisme yang digunakan untuk menyimpan serta memulihkan kondisi setiap jalur jalan tol selama permainan berlangsung. Nilai kondisi jalur akan berubah seiring penggunaan maupun proses perawatan (*maintenance*), sehingga perubahan tersebut perlu dipertahankan meskipun permainan ditutup atau pemain berpindah sesi permainan. Implementasi sistem ini bertujuan menjaga konsistensi kondisi infrastruktur jalan sehingga perkembangan permainan dapat dilanjutkan tanpa mengulang kondisi jalur dari awal.

Implementasi sistem penyimpanan kondisi jalur menggunakan dua algoritma utama, yaitu Lane State Saving Algorithm dan Lane State Restoration Algorithm. *Lane State Saving Algorithm*

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

digunakan untuk menyimpan nilai kondisi (*health*) setiap jalur ke media penyimpanan lokal menggunakan *PlayerPrefs* setiap kali terjadi perubahan kondisi. Selanjutnya, *Lane State Restoration Algorithm* digunakan untuk memuat kembali data kondisi jalur yang telah tersimpan ketika permainan dijalankan sehingga setiap jalur dapat melanjutkan kondisi terakhir yang dimiliki sebelum permainan ditutup. Kombinasi kedua algoritma tersebut memastikan kondisi seluruh jalur tetap konsisten pada setiap sesi permainan.

Proses penyimpanan kondisi jalur diawali ketika nilai kondisi (*health*) suatu jalur mengalami perubahan akibat penggunaan jalan maupun proses *maintenance*. Setelah nilai kondisi diperbarui, sistem akan menyimpan data tersebut ke media penyimpanan lokal menggunakan identitas unik setiap jalur. Ketika permainan dijalankan kembali, sistem akan membaca data yang telah tersimpan dan mengembalikan nilai kondisi masing-masing jalur ke keadaan terakhir. Dengan mekanisme tersebut, seluruh perubahan kondisi jalan dapat dipertahankan secara otomatis sehingga perkembangan permainan tetap berlanjut tanpa kehilangan data kondisi jalur.

```
2 references
void SaveLaneHealth()
{
    PlayerPrefs.SetFloat(
        GetSaveKey() + "_health",
        currentHealth
    );

    PlayerPrefs.Save();
}
```

Gambar 4.46 *Script Lane State Saving*

Gambar 4.46 menunjukkan implementasi Lane State Saving Algorithm yang digunakan untuk menyimpan kondisi setiap jalur jalan tol ke media penyimpanan lokal. Sistem memanfaatkan fungsi PlayerPrefs.SetFloat() untuk menyimpan nilai kondisi jalur

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

(currentHealth) dengan menggunakan key yang dibentuk melalui fungsi GetSaveKey() sehingga setiap jalur memiliki identitas penyimpanan yang unik. Setelah nilai kondisi diperbarui, fungsi PlayerPrefs.Save() dipanggil untuk memastikan seluruh data tersimpan secara permanen pada media penyimpanan. Dengan mekanisme tersebut, setiap perubahan kondisi jalur akibat penurunan kualitas jalan maupun proses maintenance akan langsung direkam sehingga kondisi terakhir jalur dapat dipulihkan ketika permainan dijalankan kembali.

```
0 references | Unity Message
void Start()
{
    currentHealth =
        PlayerPrefs.GetFloat(
            GetSaveKey() + "_health",
            maxHealth
        );

    UpdateUI();
    hasLoadedData = true;

    if (maintenanceButton != null)
    {
        maintenanceButton.onClick.AddListener(RepairLane);
    }
}
```

Gambar 4.47 *Script Lane State Restoration Algorithm*

Gambar 4.47 menunjukkan implementasi Lane State Restoration Algorithm yang digunakan untuk memulihkan kondisi jalur ketika permainan dijalankan. Pada awal permainan, sistem memanggil fungsi PlayerPrefs.GetFloat() untuk mengambil nilai kondisi jalur (currentHealth) yang sebelumnya telah disimpan dengan menggunakan key yang dihasilkan oleh fungsi GetSaveKey(). Apabila data penyimpanan belum tersedia, sistem akan menggunakan nilai maxHealth sebagai kondisi awal jalur. Setelah proses pemuatan selesai, sistem memperbarui tampilan antarmuka melalui fungsi UpdateUI() serta mengaktifkan status bahwa data telah berhasil dimuat. Selanjutnya, sistem menambahkan listener pada tombol maintenance agar pemain dapat melakukan perbaikan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kondisi jalur selama permainan berlangsung. Dengan mekanisme tersebut, kondisi setiap jalur dapat dipulihkan secara otomatis sesuai data terakhir yang tersimpan sehingga progres permainan tetap terjaga.

4.3.7.3 Implementasi Penyimpanan Progress Gerbang Tol

Sistem penyimpanan *progress* gerbang tol merupakan mekanisme yang digunakan untuk menyimpan seluruh perkembangan gerbang selama permainan berlangsung. Informasi yang disimpan meliputi tingkat (*level*) gerbang, status pembukaan (*unlock*), jumlah penalti, serta kondisi antrean kendaraan yang masih berada di sekitar gerbang. Implementasi sistem ini bertujuan memastikan seluruh perkembangan gerbang dapat dipertahankan ketika permainan dihentikan maupun dijalankan kembali sehingga pemain dapat melanjutkan permainan dari kondisi terakhir tanpa kehilangan progres yang telah dicapai

Implementasi sistem penyimpanan *progress* gerbang tol menggunakan tiga algoritma utama, yaitu Gate Progress Saving Algorithm, Gate Progress Restoration Algorithm, dan Queue Restoration Algorithm. *Gate Progress Saving Algorithm* digunakan untuk menyimpan informasi perkembangan gerbang, seperti level, status pembukaan, jumlah penalti, serta jumlah kendaraan yang berada di area antrean ke media penyimpanan lokal menggunakan *PlayerPrefs*. Selanjutnya, *Gate Progress Restoration Algorithm* digunakan untuk memuat kembali seluruh data tersebut ketika permainan dijalankan sehingga kondisi gerbang dapat dikembalikan sesuai sesi permainan sebelumnya. Selain itu, sistem juga menerapkan *Queue Restoration Algorithm* untuk membangun kembali kondisi antrean kendaraan berdasarkan jumlah antrean yang telah disimpan sehingga simulasi lalu lintas dapat dilanjutkan secara lebih konsisten. Kombinasi ketiga algoritma tersebut

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

memastikan perkembangan setiap gerbang tol tetap terjaga pada setiap sesi permainan.

Proses penyimpanan *progress* gerbang diawali ketika terjadi perubahan kondisi gerbang, seperti peningkatan level, pembukaan gerbang, perubahan jumlah penalti, maupun perubahan jumlah kendaraan yang berada pada area antrean. Seluruh informasi tersebut kemudian disimpan menggunakan identitas unik setiap gerbang sebagai *key* pada media penyimpanan lokal. Ketika permainan dijalankan kembali, sistem terlebih dahulu memuat seluruh data perkembangan gerbang yang telah tersimpan. Apabila masih terdapat data antrean kendaraan, sistem akan menjalankan proses pemulihan antrean secara bertahap hingga kondisi lalu lintas kembali sesuai dengan keadaan sebelum permainan dihentikan. Setelah proses pemulihan selesai, sistem kembali mengaktifkan mekanisme permainan seperti proses *spawning* kendaraan dan *Rush Hour*. Dengan mekanisme tersebut, seluruh perkembangan gerbang tol dapat dipertahankan secara konsisten tanpa mengurangi kontinuitas permainan.

```
7 references
void SaveProgressSimple()
{
    if (string.IsNullOrEmpty(gateID)) return;

    // Simpan data dasar
    PlayerPrefs.SetInt(GetSaveKey() + "_level", level);
    PlayerPrefs.SetInt(
        GetSaveKey() + "_unlock",
        isUnlocked ? 1 : 0
    );
    PlayerPrefs.SetInt(
        GetSaveKey() + "_penaltyCount",
        currentPenaltyCount
    );
    // SIMPLE: Simpan jumlah mobil di queue area
    int queueCount = GetQueueCount();
    PlayerPrefs.SetInt(
        GetSaveKey() + "_queueCount",
        queueCount
    );
    if (MoneyManager.instance != null)
    {
        PlayerPrefs.SetInt("TotalMoney", MoneyManager.instance.money);
    }

    PlayerPrefs.Save();

    Debug.Log($"💾 SAVE SIMPLE {gateID} | Level: {level} | Queue Count: {queueCount}");
}
```

Gambar 4.48 *Script Gate Progress Saving Algorithm*

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gambar 4.48 menunjukkan implementasi Gate Progress Saving Algorithm yang digunakan untuk menyimpan perkembangan setiap gerbang tol ke media penyimpanan lokal. Sistem terlebih dahulu memeriksa apakah gerbang memiliki identitas (gateID) yang valid sebelum proses penyimpanan dilakukan. Selanjutnya, sistem menyimpan berbagai informasi penting, meliputi tingkat (*level*) gerbang, status pembukaan (*unlock*), jumlah penalti, serta jumlah kendaraan yang berada pada area antrean menggunakan fungsi PlayerPrefs.SetInt(). Selain itu, sistem juga menyimpan jumlah uang pemain agar kondisi ekonomi tetap sinkron dengan perkembangan gerbang. Setelah seluruh data berhasil diperbarui, fungsi PlayerPrefs.Save() dipanggil untuk memastikan seluruh informasi tersimpan secara permanen. Dengan mekanisme tersebut, perkembangan setiap gerbang tol dapat dipertahankan dan digunakan kembali pada sesi permainan berikutnya.

```
void LoadProgress()
{
    if (string.IsNullOrEmpty(gateID)) return;

    level =
        PlayerPrefs.GetInt(
            GetSaveKey() + "_level",
            level
        );
    isUnlocked =
        PlayerPrefs.GetInt(
            GetSaveKey() + "_unlock",
            isUnlocked ? 1 : 0
        ) == 1;
    currentPenaltyCount =
        PlayerPrefs.GetInt(
            GetSaveKey() + "_penaltyCount",
            0
        );
    // SIMPLE: Load jumlah queue
    savedQueueCount =
        PlayerPrefs.GetInt(
            GetSaveKey() + "_queueCount",
            0
        );
    Debug.Log($"📥 LOAD SIMPLE {gateID} | Level: {level} | Saved Queue Count: {savedQueueCount}");
}
```

Gambar 4.49 *Script Gate Progress Restoration Algorithm*

Gambar 4.49 menunjukkan implementasi Gate Progress Restoration Algorithm yang digunakan untuk memulihkan perkembangan gerbang tol ketika permainan dijalankan kembali. Sistem terlebih dahulu memeriksa keberadaan identitas gerbang (gateID) sebelum melakukan proses pemuatan data. Selanjutnya, fungsi PlayerPrefs.GetInt() digunakan untuk mengambil informasi

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

yang sebelumnya telah disimpan, meliputi tingkat (*level*) gerbang, status pembukaan (*unlock*), jumlah penalti, serta jumlah kendaraan yang masih berada pada area antrean (savedQueueCount). Setiap data yang tidak ditemukan akan menggunakan nilai bawaan (*default value*) sehingga sistem tetap dapat berjalan dengan baik. Setelah seluruh data berhasil dimuat, sistem menampilkan informasi proses pemulihan melalui Debug.Log() sebagai indikator bahwa kondisi gerbang telah berhasil dikembalikan. Dengan mekanisme tersebut, setiap gerbang tol dapat melanjutkan kondisi terakhir yang dimiliki sebelum permainan dihentikan sehingga kontinuitas permainan tetap terjaga.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
IEnumerator RestoreQueueSimple()
{
    isRestoringTraffic = true;

    if (savedQueueCount <= 0)
    {
        Debug.Log($"✅ Tidak ada queue untuk direstore");
        StartRushHourSystemIfReady();
        yield break;
    }

    IncreaseAllExistingCarsSpeed(restoreSpeedMultiplier);

    if (spawner != null)
        spawner.SetActive(false);

    int successfullySpawned = 0;
    int retryCount = 0;
    int maxRetries = savedQueueCount * 3;

    while (successfullySpawned < savedQueueCount && retryCount < maxRetries)
    {
        if (spawner != null)
        {
            bool spawnSuccess = spawner.SpawnRandomCarFast(
                restoreYOffset,
                restoreSpeedMultiplier
            );

            if (spawnSuccess)
            {
                successfullySpawned++;
                retryCount = 0;
            }
            else
            {
                retryCount++;
            }
        }

        yield return new WaitForSeconds(0.2f);
    }

    if (spawner != null)
    {
        spawner.SetActive(true);
        spawner.ClearLastSpawnedCar();
    }

    if (isUnlocked && spawner != null)
    {
        spawner.StartSpawner();
    }

    PlayerPrefs.SetInt(
        GetSaveKey() + "_queueCount",
        0
    );
    PlayerPrefs.Save();

    isRestoringTraffic = false;

    StartRushHourSystemIfReady();
}
```

Gambar 4.50 *Script Queue Restoration Algorithm*

Gambar 4.50 menunjukkan implementasi Queue Restoration Algorithm yang digunakan untuk membangun kembali kondisi antrean kendaraan setelah permainan dijalankan kembali. Proses diawali dengan memeriksa jumlah kendaraan yang sebelumnya tersimpan pada variabel savedQueueCount. Apabila tidak terdapat

antrean yang perlu dipulihkan, sistem langsung mengaktifkan kembali mekanisme *Rush Hour*. Sebaliknya, apabila masih terdapat data antrean, sistem akan menonaktifkan sementara *spawner* kendaraan agar proses pemulihan tidak terganggu. Selanjutnya, kendaraan dipanggil kembali secara bertahap menggunakan fungsi SpawnRandomCarFast() hingga jumlah kendaraan yang berhasil dipulihkan sesuai dengan data yang tersimpan. Setelah seluruh proses selesai, sistem mengaktifkan kembali *spawner*, menghapus data antrean yang telah dipulihkan dari *PlayerPrefs*, kemudian menjalankan kembali sistem *Rush Hour*. Pendekatan ini memungkinkan kondisi lalu lintas sebelum permainan dihentikan dapat direkonstruksi secara bertahap sehingga transisi permainan menjadi lebih alami dan kontinuitas simulasi tetap terjaga.

Pada tahap awal pengembangan, sistem penyimpanan gerbang tol hanya menyimpan sebagian data perkembangan gerbang sehingga ketika permainan dijalankan kembali, beberapa kondisi seperti level gerbang, status pembukaan, dan jumlah kendaraan pada area antrean tidak dapat dipulihkan secara konsisten. Selain itu, antrean kendaraan yang tersimpan sebelumnya akan hilang dan menyebabkan kondisi lalu lintas setelah proses load berbeda dengan keadaan sebelum permainan dihentikan. Berdasarkan hasil pengujian, sistem kemudian dikembangkan dengan menyimpan seluruh informasi penting gerbang, meliputi level, status unlock, jumlah penalti, dan jumlah kendaraan pada area antrean menggunakan PlayerPrefs dengan identitas gerbang yang unik. Selanjutnya, ditambahkan mekanisme Queue Restoration Algorithm untuk membangun kembali kondisi antrean kendaraan secara bertahap sebelum sistem spawning dan Rush Hour diaktifkan kembali. Hasil iterasi menunjukkan bahwa perkembangan setiap gerbang tol dapat dipulihkan secara lebih konsisten dan kondisi lalu lintas setelah proses load menjadi lebih

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

mendekati keadaan sebelum permainan dihentikan, sehingga kontinuitas permainan tetap terjaga dengan baik.

4.3.8 Implementasi Sistem Game Over

Sistem *Game Over* merupakan mekanisme yang digunakan untuk mengakhiri permainan ketika pemain tidak lagi memenuhi kondisi untuk melanjutkan permainan. Pada permainan *Toll Tycoon*, kondisi *Game Over* dapat terjadi akibat beberapa faktor, seperti jumlah penalti yang telah mencapai batas maksimum atau kondisi keuangan pemain yang tidak lagi mencukupi untuk menjalankan operasional jalan tol. Implementasi sistem ini bertujuan memberikan batasan permainan sekaligus menjaga keseimbangan simulasi sehingga pemain dituntut untuk mengelola pendapatan, kondisi jalan, dan kelancaran lalu lintas secara optimal.

Implementasi sistem *Game Over* menggunakan dua algoritma utama, yaitu Game Over Detection Algorithm dan Game State Reset Algorithm. *Game Over Detection Algorithm* digunakan untuk mendeteksi berbagai kondisi yang menyebabkan permainan berakhir, seperti jumlah penalti yang mencapai batas maksimum maupun kondisi keuangan yang tidak mencukupi. Setelah kondisi tersebut terpenuhi, sistem menjalankan Game State Reset Algorithm untuk menghentikan permainan, menampilkan antarmuka *Game Over*, serta melakukan proses reset terhadap seluruh data permainan agar pemain dapat memulai sesi permainan baru. Kombinasi kedua algoritma tersebut memastikan proses transisi menuju kondisi *Game Over* berlangsung secara konsisten dan seluruh data permainan dapat dikembalikan ke kondisi awal.

Proses *Game Over* diawali ketika sistem mendeteksi salah satu kondisi kegagalan selama permainan berlangsung. Setelah kondisi tersebut terpenuhi, sistem akan memanggil fungsi *GameOver()* untuk menghentikan seluruh aktivitas permainan serta menampilkan antarmuka *Game Over*. Selanjutnya, sistem menjalankan proses reset terhadap data permainan yang meliputi jumlah uang pemain, perkembangan gerbang tol,

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kondisi jalur, dan data permainan lainnya sehingga seluruh progres kembali ke kondisi awal. Setelah proses reset selesai, pemain dapat memulai permainan kembali melalui menu yang telah disediakan. Mekanisme tersebut memastikan setiap sesi permainan dimulai dengan kondisi yang konsisten tanpa dipengaruhi oleh data dari permainan sebelumnya.

```
5 references
public void GameOver()
{
    Debug.Log("GAME OVER!");
    Time.timeScale = 0f;

    if (gameOverPanel ≠ null)
        gameOverPanel.SetActive(true);

    if (finalMoneyText ≠ null && MoneyManager.instance ≠ null)
        finalMoneyText.text = "Money: " + MoneyManager.instance.money;

    if (gameOverText ≠ null)
        gameOverText.localScale = Vector3.zero;

    StartCoroutine(FadeInGameOver());
}
```

Gambar 4.51 *Script Game Over Detection Algorithm*

Gambar 4.51 menunjukkan implementasi Game Over Detection Algorithm yang digunakan untuk menangani kondisi ketika permainan dinyatakan berakhir. Setelah fungsi GameOver() dipanggil oleh sistem, permainan akan dihentikan sementara dengan mengatur nilai Time.timeScale menjadi 0f sehingga seluruh aktivitas permainan berhenti. Selanjutnya, sistem menampilkan antarmuka Game Over melalui objek gameOverPanel serta menampilkan jumlah pendapatan terakhir pemain pada komponen finalMoneyText. Sebelum animasi ditampilkan, skala teks Game Over diinisialisasi menjadi nol agar efek animasi dapat dijalankan dari kondisi awal. Tahap terakhir dilakukan dengan memanggil coroutine FadeInGameOver() untuk menampilkan panel Game Over secara bertahap. Mekanisme tersebut memastikan proses transisi menuju kondisi permainan berakhir berlangsung dengan jelas serta memberikan informasi akhir permainan kepada pemain.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
0 references
public void RestartGame()
{
    Debug.Log("FULL GAME RESET");

    Time.timeScale = 1f;
    PlayerPrefs.DeleteAll();
    PlayerPrefs.Save();

    if (MoneyManager.instance != null)
        MoneyManager.instance.ResetMoney();

    if (HighwayDataManager.Instance != null)
    {
        HighwayDataManager.Instance.ResetAllData();
    }

    if (gameOverPanel != null)
        gameOverPanel.SetActive(false);

    SceneManager.LoadScene(SceneManager.GetActiveScene().buildIndex);
}
```

Gambar 4.52 *Script Game State Reset Algorithm*

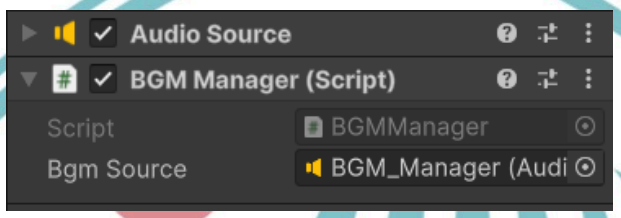
Gambar 4.52 menunjukkan implementasi Game State Reset Algorithm yang digunakan untuk mengembalikan permainan ke kondisi awal setelah pemain memilih untuk memulai permainan kembali. Proses diawali dengan mengaktifkan kembali waktu permainan melalui Time.timeScale = 1f agar seluruh aktivitas permainan dapat berjalan normal. Selanjutnya, sistem menghapus seluruh data penyimpanan menggunakan PlayerPrefs.DeleteAll() dan memastikan proses penghapusan tersimpan melalui PlayerPrefs.Save(). Setelah itu, sistem mengembalikan jumlah uang pemain ke nilai awal menggunakan fungsi ResetMoney() serta menghapus seluruh data perkembangan jalan tol melalui ResetAllData(). Sebelum permainan dimulai kembali, panel Game Over ditutup agar antarmuka kembali ke kondisi normal. Tahap terakhir dilakukan dengan memuat ulang scene yang sedang aktif menggunakan SceneManager.LoadScene(), sehingga seluruh objek permainan diinisialisasi kembali dan pemain dapat memulai permainan dari kondisi awal.

4.3.9 Implementasi Sistem Audio

Sistem audio digunakan untuk memberikan umpan balik suara dan meningkatkan pengalaman bermain selama simulasi berlangsung. Implementasi sistem audio bertujuan untuk membantu pemain memahami kondisi permainan melalui efek suara, meningkatkan suasana permainan,

serta memberikan respon interaktif terhadap setiap tindakan yang dilakukan pemain.

Pada penelitian ini, implementasi sistem audio dibagi menjadi dua bagian utama, yaitu *background music* (BGM) dan *sound effect* (SFX). Sistem audio diimplementasikan menggunakan *AudioSource* pada Unity yang dikontrol melalui script *BGMManager* dan *SettingsUI*.



Gambar 4.53 Implementasi Musik Pada Unity

Implementasi *background music* digunakan untuk memutar musik permainan secara terus menerus selama permainan berlangsung.

```
0 references | Unity Message
void Start()
{
    float volume = PlayerPrefs.GetFloat("BGM", 1f);
    bgmSource.volume = volume;

    SFXVolume = PlayerPrefs.GetFloat("SFX", 1f);
    bgmSource.loop = true;
    bgmSource.Play();
}
```

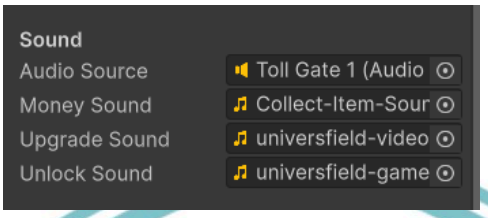
Gambar 4.54 Potongan Kode Backgroun Musik

Potongan kode tersebut digunakan untuk memutar *background music* secara otomatis ketika permainan dimulai. Sistem juga memuat kembali pengaturan volume yang sebelumnya telah disimpan menggunakan *PlayerPrefs* sehingga pengaturan audio pemain tetap dapat dipertahankan ketika permainan dijalankan kembali.

Selain *background music*, sistem juga menerapkan *sound effect* sebagai respon terhadap berbagai tindakan pemain selama permainan berlangsung

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

seperti transaksi kendaraan, upgrade gerbang tol, dan pembukaan gerbang tol.



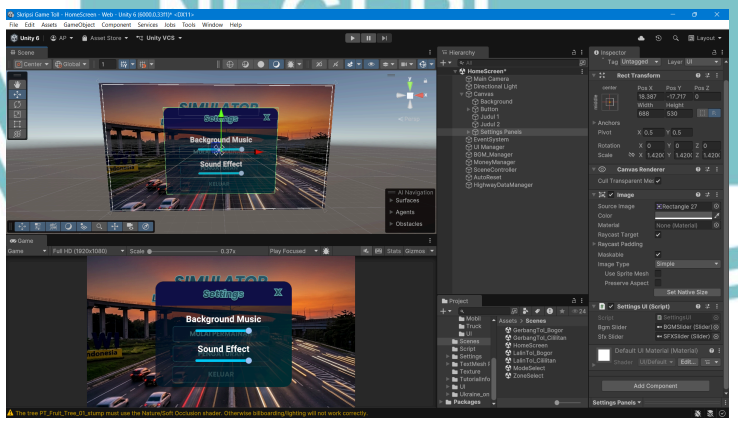
Gambar 4.55 Implementasi Efek Suara Pada Unity

```
if (audioSource != null && upgradeSound != null)
    audioSource.PlayOneShot(
        upgradeSound,
        BGMManager.SFXVolume
    );
```

Gambar 4.56 Potongan Kode Efek Suara

Potongan kode tersebut digunakan untuk menjalankan efek suara ketika pemain melakukan tindakan tertentu pada permainan. Implementasi *sound effect* digunakan untuk memberikan umpan balik interaktif sehingga setiap aksi pemain dapat terasa lebih responsif selama simulasi berlangsung.

Pada penelitian ini, pemain juga dapat mengatur volume *background music* dan *sound effect* menggunakan menu pengaturan permainan.



Gambar 4.57 Pengaturan Suara

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
0 references | Unity Message
void Start()
{
    // Load saved values
    bgmSlider.value =
        PlayerPrefs.GetFloat("BGM", 1f);

    sfxSlider.value =
        PlayerPrefs.GetFloat("SFX", 1f);

    // Register listeners
    bgmSlider.onValueChanged.AddListener(SetBGMVolume);
    sfxSlider.onValueChanged.AddListener(SetSFXVolume);
}
```

Gambar 4.58 Potongan Kode Pengaturan Suara

Potongan kode tersebut digunakan untuk menghubungkan *slider* pengaturan audio dengan sistem volume permainan sehingga pemain dapat mengatur tingkat volume sesuai kebutuhan selama permainan berlangsung.

Implementasi sistem audio memiliki peran penting dalam meningkatkan kenyamanan bermain dan memperkuat suasana simulasi pengelolaan jalan tol. Melalui penerapan *background music* dan *sound effect*, pemain dapat memperoleh umpan balik audio secara langsung terhadap kondisi permainan dan tindakan operasional yang dilakukan selama simulasi berlangsung.

4.3.10 Implementasi Sistem Loading

Sistem *Loading* merupakan mekanisme yang digunakan untuk menampilkan proses pemuatan data sebelum permainan dapat dijalankan sepenuhnya. Pada permainan *Toll Tycoon*, sistem ini berfungsi menampilkan antarmuka *Loading* ketika proses pemulihan data permainan sedang berlangsung, seperti pemulihan kondisi gerbang tol, antrean kendaraan, maupun data pendukung lainnya. Selain memberikan informasi mengenai proses pemuatan, sistem juga menampilkan indikator perkembangan (*progress*) sehingga pemain mengetahui bahwa proses *loading* masih berlangsung. Implementasi sistem ini bertujuan memberikan transisi permainan yang lebih halus sekaligus memastikan

seluruh data berhasil dipulihkan sebelum mekanisme permainan kembali dijalankan.

Implementasi sistem *Loading* menggunakan tiga algoritma utama, yaitu Loading Screen Management Algorithm, Loading Progress Update Algorithm, dan Loading Animation Algorithm. *Loading Screen Management Algorithm* digunakan untuk mengatur kapan panel *Loading* ditampilkan maupun disembunyikan berdasarkan jumlah proses pemuatan yang sedang berlangsung. Selanjutnya, *Loading Progress Update Algorithm* digunakan untuk memperbarui nilai progres pemuatan sehingga pemain dapat memantau perkembangan proses *loading*. Untuk meningkatkan pengalaman pengguna, sistem juga menerapkan *Loading Animation Algorithm* yang menampilkan animasi titik bergerak pada teks *Loading* sehingga antarmuka terlihat lebih dinamis selama proses pemuatan berlangsung. Kombinasi ketiga algoritma tersebut menghasilkan proses *loading* yang lebih informatif, responsif, dan nyaman bagi pemain.

Proses *Loading* diawali ketika sistem menerima permintaan untuk memuat data permainan. Sistem akan menampilkan panel *Loading* beserta indikator progres dan memulai animasi teks sebagai penanda bahwa proses pemuatan sedang berlangsung. Selama data dipulihkan, sistem secara berkala memperbarui nilai progres sesuai perkembangan proses *loading*. Setelah seluruh data berhasil dimuat, sistem akan menghentikan animasi, menyembunyikan panel *Loading*, kemudian mengaktifkan kembali seluruh mekanisme permainan sehingga pemain dapat melanjutkan permainan dari kondisi terakhir yang tersimpan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
1 reference
public void ShowLoading(string message = "Loading")
{
    activeRequests++;

    if (loadingPanel != null && !isShowing)
    {
        loadingPanel.SetActive(true);
        isShowing = true;
        showTime = Time.time;
        currentBaseMessage = message;

        // Reset UI
        if (loadingSlider != null)
            loadingSlider.value = 0f;

        targetProgress = 0f;
        if (percentText != null)
            percentText.text = "0%";

        // Mulai animasi titik
        if (dotAnimationCoroutine != null)
            StopCoroutine(dotAnimationCoroutine);
        dotAnimationCoroutine = StartCoroutine(AnimateDots());

        Debug.Log($"✅ Loading panel ditampilkan (Requests: {activeRequests})");
    }
    else if (isShowing)
    {
        // Update message jika sudah showing
        UpdateMessage(message);
    }
}
```

Gambar 4.59 *Script Loading Screen Management*

Gambar 4.59 menunjukkan implementasi Loading Screen Management Algorithm yang digunakan untuk mengatur tampilan panel *Loading* selama proses pemuatan data berlangsung. Ketika fungsi ShowLoading() dipanggil, sistem terlebih dahulu menambahkan jumlah permintaan *loading* melalui variabel activeRequests. Apabila panel *Loading* belum ditampilkan, sistem akan mengaktifkan panel, mencatat waktu dimulainya proses *loading*, serta menginisialisasi seluruh komponen antarmuka, seperti *slider* progres dan persentase pemuatan ke nilai awal. Selanjutnya, sistem menjalankan coroutine AnimateDots() untuk menampilkan animasi pada teks *Loading*. Apabila panel *Loading* telah aktif, sistem hanya memperbarui pesan yang ditampilkan melalui fungsi UpdateMessage(). Dengan mekanisme tersebut, panel *Loading* hanya ditampilkan satu kali meskipun terdapat beberapa proses pemuatan yang berlangsung secara bersamaan, sehingga antarmuka permainan tetap konsisten dan mudah dipahami oleh pemain.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
// Method untuk update progress (0 - 1)
1 reference
public void UpdateProgress(float progress)
{
    if (!isShowing) return;

    progress = Mathf.Clamp01(progress);

    // Simpan target progress
    targetProgress = progress;

    // Update persen text
    if (percentText != null)
    {
        int percentValue = Mathf.RoundToInt(progress * 100f);
        percentText.text = percentValue + "%";
    }
}
0 references | Unity Message
void Update()
{
    if (loadingSlider != null)
    {
        loadingSlider.value = Mathf.Lerp(
            loadingSlider.value,
            targetProgress,
            Time.deltaTime * 5f
        );
    }
}
```

Gambar 4.60 *Script Loading Progress Update Algorithm*

Gambar 4.60 menunjukkan implementasi Loading Progress Update Algorithm yang digunakan untuk memperbarui indikator perkembangan proses *loading*. Ketika fungsi UpdateProgress() dipanggil, sistem terlebih dahulu memastikan bahwa panel *Loading* sedang aktif, kemudian membatasi nilai progres pada rentang 0 hingga 1 menggunakan fungsi Mathf.Clamp01(). Nilai tersebut selanjutnya disimpan pada variabel targetProgress dan dikonversi menjadi persentase yang ditampilkan pada komponen percentText. Pada setiap *frame*, fungsi Update() akan memperbarui nilai *slider* secara bertahap menuju targetProgress menggunakan fungsi Mathf.Lerp(). Pendekatan ini menghasilkan perubahan indikator progres yang lebih halus sehingga pergerakan *slider* tidak terlihat meloncat secara tiba-tiba selama proses pemuatan berlangsung.

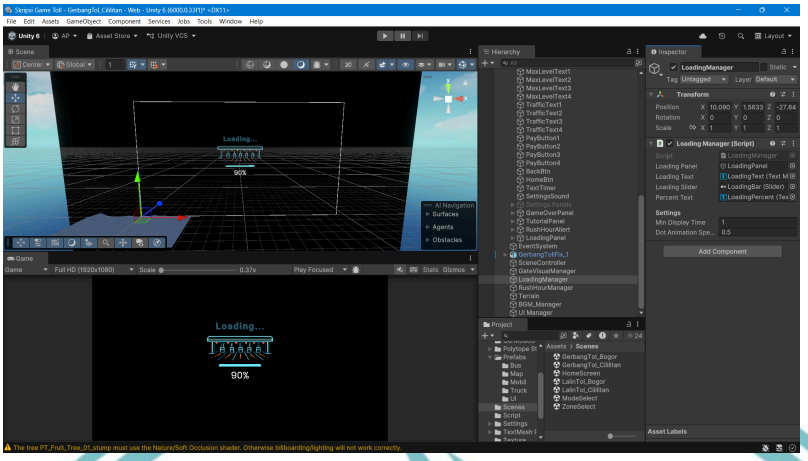
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

```
1 reference
private IEnumerator AnimateDots()
{
    int dotCount = 0;

    while (isShowing)
    {
        if (loadingText != null)
        {
            string dots = new string('.', dotCount);
            loadingText.text = currentBaseMessage + dots;
        }

        dotCount = (dotCount + 1) % 4; // 0,1,2,3 lalu ke 0 lagi

        yield return new WaitForSeconds(dotAnimationSpeed);
    }
}
```

Gambar 4.61 *Script Loading Animation Algorithm*

Gambar 4.61 menunjukkan implementasi Loading Animation Algorithm yang digunakan untuk memberikan animasi pada teks *Loading* selama proses pemuatan berlangsung. Sistem menggunakan coroutine AnimateDots() yang dijalankan selama panel *Loading* masih aktif (isShowing). Pada setiap iterasi, sistem membentuk sejumlah karakter titik (.) berdasarkan nilai dotCount, kemudian menggabungkannya dengan pesan dasar (currentBaseMessage) untuk ditampilkan pada komponen loadingText. Setelah teks diperbarui, nilai dotCount akan bertambah secara siklik dari 0 hingga 3 menggunakan operasi modulo sehingga menghasilkan animasi titik yang berulang. Selanjutnya, coroutine menunggu selama dotAnimationSpeed sebelum mengulangi proses berikutnya. Mekanisme ini memberikan indikasi visual bahwa proses pemuatan masih berlangsung sehingga antarmuka permainan terlihat lebih dinamis dan informatif.



Gambar 4.62 Implementasi Loading pada Unity

Berdasarkan hasil pengujian, sistem kemudian dikembangkan dengan menambahkan indikator persentase pemuatan (*loading progress*), *slider* progres, serta animasi titik pada teks *Loading* sebagai indikator visual bahwa proses pemuatan masih berlangsung. Selain itu, diterapkan mekanisme *active request* untuk mengelola beberapa permintaan *loading* secara bersamaan sehingga panel *Loading* tidak ditampilkan maupun disembunyikan secara berulang. Sistem juga menerapkan waktu tampil minimum (*minimum display time*) agar panel *Loading* tidak menghilang terlalu cepat ketika proses pemuatan selesai dalam waktu singkat, sehingga transisi antarscene menjadi lebih halus.

Hasil iterasi menunjukkan bahwa penambahan indikator progres, animasi *Loading*, serta pengelolaan permintaan pemuatan mampu meningkatkan pengalaman pengguna selama proses pemuatan berlangsung. Informasi mengenai perkembangan *loading* menjadi lebih jelas, perpindahan antarscene terasa lebih mulus, dan proses pemulihan data permainan dapat berlangsung secara lebih stabil tanpa menimbulkan perubahan tampilan yang terjadi secara tiba-tiba.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4.4 Pengujian

4.4.1 Deskripsi Pengujian

Pengujian sistem dilakukan untuk memastikan seluruh fitur dan mekanisme permainan dapat berjalan sesuai dengan rancangan yang telah dibuat pada tahap implementasi sistem. Selain itu, pengujian juga bertujuan untuk mengevaluasi kestabilan gameplay, keseimbangan sistem permainan, serta kenyamanan pemain selama simulasi berlangsung.

Pada penelitian ini, pengujian dilakukan menggunakan dua metode utama yaitu *alpha testing* dan *beta testing*. *Alpha testing* dilakukan secara internal oleh pengembang untuk menguji fungsi sistem, mendeteksi kesalahan sistem, serta memastikan seluruh fitur permainan dapat berjalan dengan baik sebelum permainan diuji oleh pengguna lain. Sementara itu, *beta testing* dilakukan untuk memperoleh masukan dan evaluasi langsung dari *expert game* serta pengguna umum terkait gameplay, tingkat kesulitan permainan, kenyamanan bermain, dan kualitas simulasi yang dihasilkan.

Selain pengujian fungsional sistem, penelitian ini juga melakukan evaluasi terhadap balancing gameplay seperti balancing ekonomi gerbang tol, balancing lalu lintas kendaraan, tingkat kepadatan kendaraan saat *rush hour*, dan efisiensi jalan tol. Pengujian tersebut dilakukan melalui proses *playtesting* secara berulang untuk memperoleh keseimbangan permainan yang sesuai antara tingkat tantangan dan kenyamanan bermain.

Hasil dari seluruh proses pengujian kemudian digunakan sebagai dasar evaluasi dan penyempurnaan sistem permainan agar simulasi pengelolaan jalan tol yang dikembangkan dapat berjalan sesuai dengan tujuan penelitian

4.4.2 Prosedur Pengujian

Prosedur pengujian dilakukan untuk memastikan seluruh sistem permainan dapat berjalan sesuai dengan rancangan yang telah dibuat pada tahap

implementasi. Pengujian dilakukan secara bertahap mulai dari pengujian internal oleh pengembang hingga pengujian langsung oleh *expert game* dan pengguna umum.

Pada penelitian ini, proses pengujian dibagi menjadi dua tahap utama yaitu *alpha testing* dan *beta testing*. *Alpha testing* dilakukan untuk menguji fungsi utama sistem permainan seperti sistem spawn kendaraan, transaksi gerbang tol, sistem antrean kendaraan, maintenance jalur, rescue kendaraan, save system, dan sistem *game over*. Pengujian tersebut dilakukan secara internal menggunakan metode pengujian fungsional untuk memastikan seluruh fitur permainan dapat berjalan dengan baik dan sesuai dengan kebutuhan sistem.

Setelah seluruh fungsi utama permainan berjalan dengan baik, proses pengujian dilanjutkan ke tahap *beta testing*. Pengujian ini dilakukan dengan melibatkan *expert game* dan pengguna umum untuk memperoleh evaluasi terhadap gameplay, tingkat kesulitan permainan, kenyamanan bermain, kualitas antarmuka, serta keseimbangan sistem permainan selama simulasi berlangsung.

Pada proses *beta testing*, data pengujian dikumpulkan menggunakan kuesioner yang disebarkan kepada responden setelah memainkan permainan. Hasil pengujian kemudian dianalisis untuk mengetahui tingkat keberhasilan sistem dan mengevaluasi pengalaman bermain pengguna terhadap simulasi pengelolaan jalan tol yang dikembangkan pada penelitian ini.

4.4.2.1 Alpha Testing

Pada tahap *Alpha Testing*, pengujian dilakukan secara internal oleh penulis dan satu rekan tim untuk memastikan seluruh fitur dan mekanisme permainan dapat berjalan sesuai dengan rancangan sistem yang telah dibuat. Pengujian ini bertujuan untuk memverifikasi fungsi utama permainan seperti sistem menu utama,

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

tutorial, gameplay, spawn kendaraan, AI kendaraan, gerbang tol, sistem pendapatan, maintenance, rescue, upgrade, antarmuka permainan (*user interface*), hingga sistem *game over*.

Proses pengujian dilakukan menggunakan metode pengujian fungsional (*functional testing*) dengan cara menjalankan setiap fitur permainan secara langsung pada perangkat komputer menggunakan *game engine* Unity. Setiap fitur diuji berdasarkan skenario penggunaan untuk memastikan sistem dapat berjalan dengan baik tanpa mengalami kesalahan fungsi selama permainan berlangsung.

Selain menguji fungsi utama permainan, tahap *Alpha Testing* juga dilakukan untuk mendeteksi kemungkinan terjadinya bug, kesalahan antarmuka, maupun ketidaksesuaian mekanisme gameplay terhadap rancangan awal permainan. Jika ditemukan kesalahan pada sistem, maka dilakukan proses analisis dan evaluasi untuk memperbaiki sistem sebelum permainan memasuki tahap *Beta Testing* oleh *expert game* dan pengguna umum.

Tabel 4.20 menunjukkan daftar pengujian fungsional yang dilakukan pada tahap *Alpha Testing*.

Tabel 4.13 Pertanyaan Alpha testing

No	Fitur	Pertanyaan
1.	Menu Utama	Apakah tombol Mulai Game berfungsi dengan baik?
2.	Menu Utama	Apakah menu Pengaturan berfungsi dengan baik?
3.	Tutorial	Apakah tutorial tampil dengan jelas dan dapat digunakan dengan baik?
4.	Gameplay	Apakah permainan berjalan dengan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

		kondisi awal yang sesuai?
5.	Gameplay	Apakah timer permainan berjalan dengan baik?
6.	Spawn Kendaraan	Apakah kendaraan muncul dengan normal?
7.	AI Kendaraan	Apakah kendaraan dapat bergerak dan mengikuti jalur dengan benar?
8.	AI Kendaraan	Apakah kendaraan dapat berhenti dan bergerak kembali saat antrean terjadi?
9.	AI Kendaraan	Apakah kendaraan keluar area setelah melewati gerbang tol?
10.	Gerbang Tol	Apakah kendaraan dapat memasuki gerbang yang sesuai?
11.	Gerbang Tol	Apakah proses transaksi kendaraan berjalan dengan baik?
12.	Gerbang Tol	Apakah sistem antrean kendaraan berjalan dengan baik?
13.	Gerbang Tol	Apakah detail visual pada aset Gerbang Tol Jagorawi mudah dikenali oleh pengguna ?
14.	Gerbang Tol	Apakah proporsi ukuran antara gerbang tol, gardu tol, dan jalur kendaraan terlihat sesuai ?
15.	Gerbang Tol	Apakah Material dan tekstur yang digunakan pada aset Gerbang Tol

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

POLITEKNIK NEGERI JAKARTA

		Jagorawi sudah sesuai dengan konsep visual yang dirancang ?
16.	Ruas Jalan Tol	Apakah Bentuk dan struktur ruas jalan tol telah sesuai dengan karakteristik jalan tol pada umumnya ?
17.	Ruas Jalan Tol	Apakah marka jalan yang ditampilkan mudah dikenali dan dipahami?
18.	Ruas Jalan Tol	Apakah Pembatas jalan dan elemen pendukung lainnya ditempatkan dengan tepat?
19.	Ruas Jalan Tol	Apakah proporsi lebar jalan dan jalur kendaraan terlihat realistis ?
20.	Ruas Jalan Tol	Apakah Material dan tekstur pada ruas jalan memberikan tampilan yang sesuai dengan kondisi jalan tol ?
21.	Rescue	Apakah fitur rescue dapat digunakan dengan baik?
22.	Rescue	Apakah sistem harga rescue berjalan dengan benar?
23.	Upgrade	Apakah fitur upgrade gerbang berfungsi dengan baik?
24.	Upgrade	Apakah sistem biaya dan level upgrade berjalan dengan benar?
25.	Upgrade	Apakah upgrade memberikan perubahan pada gameplay?

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

26.	UI	Apakah informasi saldo, kondisi, dan level gerbang tampil dengan jelas?
27.	UI	Apakah tombol, teks, dan ikon dapat digunakan dan dibaca dengan baik?
28.	UI	Apakah tampilan antarmuka berjalan dengan rapi tanpa tumpang tindih?
29.	Game Over	Apakah kondisi game over muncul dengan benar?
30.	Game Over	Apakah informasi hasil permainan ditampilkan dengan baik?
31.	Game Over	Apakah tombol kembali ke menu utama berfungsi dengan baik?

4.4.2.2 Beta Testing

Pada tahap *Beta Testing*, pengujian melibatkan pengguna eksternal yaitu *expert game* dan pengguna umum untuk mengevaluasi kualitas gameplay, balancing permainan, sistem pengambilan keputusan, serta kenyamanan bermain pada game *Toll Tycoon*. Pengujian ini dilakukan untuk mengetahui apakah seluruh mekanisme permainan telah berjalan sesuai dengan tujuan penelitian serta memberikan pengalaman simulasi pengelolaan gerbang tol yang baik bagi pemain.

Pengujian *Beta Testing* dilakukan dengan cara memainkan game secara langsung pada perangkat komputer, kemudian responden diminta untuk mengisi kuesioner berdasarkan pengalaman bermain yang telah dilakukan. Data hasil pengujian kemudian digunakan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

sebagai bahan evaluasi terhadap gameplay, balancing sistem, kualitas antarmuka, dan pengalaman bermain secara keseluruhan.

1. **Beta Testing Pengguna Umum**

Pada pengujian pengguna umum, pengujian dilakukan untuk mengetahui tingkat pemahaman pemain terhadap gameplay, kemudahan penggunaan antarmuka, kualitas visual permainan, serta tingkat kenyamanan bermain game *Toll Tycoon*. Pengujian ini melibatkan pengguna umum sebagai pemain utama untuk mengetahui apakah gameplay dan sistem simulasi yang diterapkan dapat dipahami dengan baik oleh pemain.

Pada kuesioner yang diberikan kepada pengguna umum terdapat lima belas (15) pertanyaan yang digunakan untuk mengevaluasi pengalaman bermain pengguna seperti pada Tabel 4.21.

Tabel 4.14 Pertanyaan Beta Testing Umum

No	Pertanyaan
1.	Game Toll Tycoon memberikan pengalaman bermain yang menyenangkan sebagai media hiburan.
2.	Gameplay pada game ini membuat saya tertarik untuk terus bermain hingga permainan selesai.
3.	Tantangan dalam mengelola gerbang tol membuat permainan terasa seru dan tidak membosankan.
4.	Fitur Maintenance, Rescue, dan Upgrade membuat permainan menjadi lebih menarik.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

5.	Visual lingkungan Jalan Tol Jagorawi membuat pengalaman bermain terasa lebih imersif.
6.	Animasi kendaraan dan kondisi lalu lintas membuat permainan terasa lebih hidup.
7.	Permainan memberikan tingkat tantangan yang sesuai sehingga tetap menyenangkan untuk dimainkan.
8.	Saya merasa puas dengan pengalaman bermain game Toll Tycoon.
9.	Saya tertarik untuk memainkan game Toll Tycoon kembali di lain waktu.
10.	Secara keseluruhan, game Toll Tycoon berhasil memberikan pengalaman bermain yang menghibur.
11.	Game ini memberikan gambaran mengenai operasional pelayanan di gerbang tol.
12.	Setelah memainkan game ini, saya memperoleh gambaran mengenai pentingnya kelancaran pelayanan kendaraan di gerbang tol.
13.	Fitur Maintenance, Rescue, dan Upgrade memberikan gambaran mengenai upaya menjaga kualitas pelayanan jalan tol.
14.	Simulasi antrean kendaraan membantu saya memahami pentingnya pengelolaan lalu lintas di gerbang tol.
15.	Secara keseluruhan, game Toll Tycoon mampu memberikan hiburan sekaligus memberikan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

	informasi mengenai operasional dan pelayanan jalan tol.

2. Beta Testing Expert Game

Pada pengujian *expert game*, pengujian dilakukan untuk mengevaluasi mekanisme gameplay, balancing ekonomi permainan, progression gameplay, sistem pengambilan keputusan, serta kualitas simulasi pengelolaan gerbang tol yang diterapkan pada game *Toll Tycoon*. Pengujian ini dilakukan oleh *expert game* untuk memperoleh evaluasi terhadap kualitas sistem permainan dari sisi desain gameplay dan simulasi manajemen permainan.

Data pengujian dikumpulkan melalui pengisian kuesioner yang terdiri dari beberapa pertanyaan terkait gameplay, balancing permainan, sistem ekonomi, progression permainan, dan pengalaman bermain secara keseluruhan seperti pada Tabel 4.22.

Tabel 4.15 Pertanyaan Beta Testing Expert Game

No	Pertanyaan
1.	Mekanisme inti permainan mudah dipahami dan memiliki tujuan yang jelas.
2.	Sistem pengelolaan gerbang tol memberikan tantangan yang sesuai kepada pemain.
3.	Mekanisme antrean kendaraan berjalan secara logis dan mendukung gameplay simulasi.
4.	Variasi jenis kendaraan memberikan pengaruh terhadap strategi permainan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

5.	Sistem gameplay mendorong pemain untuk mengambil keputusan selama permainan berlangsung.
6.	Pemain perlu menentukan prioritas tindakan untuk mengelola kondisi permainan.
7.	Sistem maintenance memberikan dampak yang signifikan terhadap jalannya permainan.
8.	Fitur rescue memberikan solusi yang relevan terhadap kondisi tertentu dalam permainan.
9.	Mekanisme upgrade memberikan perkembangan gameplay yang terasa jelas.
10.	Sistem upgrade mendorong pemain untuk terus meningkatkan performa gerbang tol.
11.	Sistem pendapatan dalam game mampu memotivasi pemain untuk terus bermain.
12.	Pendapatan yang diperoleh pemain terasa seimbang dengan tantangan permainan.
13.	Peningkatan level gerbang memberikan pengaruh terhadap peningkatan pendapatan pemain.
14.	Sistem biaya maintenance dan rescue sudah seimbang dengan sistem pendapatan game.
15.	Sistem ekonomi dalam permainan mendorong pemain untuk membuat keputusan strategis.
16.	Mekanisme pengeluaran dan pemasukan dalam game terasa realistis untuk simulasi gerbang tol.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

17.	Progress permainan terasa berkembang seiring bertambahnya pendapatan pemain.
18.	Game memberikan situasi yang mengharuskan pemain mengambil keputusan secara cepat.
19.	Pemain dapat menentukan strategi pengelolaan gerbang sesuai kondisi permainan.
20.	Pilihan penggunaan uang dalam game memengaruhi jalannya permainan.
21.	Pemain perlu mempertimbangkan kapan harus melakukan maintenance, rescue, atau upgrade.
22.	Sistem permainan memberikan konsekuensi terhadap keputusan yang diambil pemain.
23.	Gameplay yang disajikan mampu melatih kemampuan pengambilan keputusan pemain.
24	Tingkat kesulitan permainan meningkat secara bertahap dan seimbang.
25.	Progress permainan terasa konsisten dari awal hingga akhir permainan.
26.	Sistem upgrade dan peningkatan tantangan sudah memiliki balancing yang baik.
27.	Tidak terdapat mekanisme yang terasa terlalu mudah atau terlalu sulit dalam permainan.
28.	Gameplay memiliki alur progression yang dapat mempertahankan minat pemain.

Kritik dan Saran

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 4.16 Pertanyaan Beta Testing Kritik dan Saran Expert Game

No	Pertanyaan
1.	Kritik dan saran terhadap mekanisme permainan
2.	Kritik dan saran terhadap balancing gameplay
3.	Kritik dan saran terhadap sistem ekonomi dan progression
4.	Kritik dan saran terhadap pengalaman bermain secara keseluruhan
5.	Saran pengembangan fitur untuk game Toll Tycoon

4.4.3 Data Hasil Pengujian

Data hasil pengujian diperoleh berdasarkan proses *Alpha Testing* dan *Beta Testing* yang dilakukan pada game *Toll Tycoon* untuk mengetahui tingkat keberhasilan sistem, kestabilan gameplay, kualitas simulasi pengelolaan gerbang tol, serta pengalaman bermain pengguna selama permainan berlangsung. Hasil pengujian diperoleh melalui pengujian fungsional sistem dan pengisian kuesioner oleh *expert game* maupun pengguna umum setelah memainkan permainan. Berikut merupakan data hasil pengujian yang diperoleh dari proses pengujian game *Toll Tycoon*.

4.4.3.1 Hasil Alpha Testing

Hasil *Alpha Testing* diperoleh berdasarkan pengujian fungsional yang dilakukan terhadap seluruh fitur utama pada game *Toll Tycoon*. Pengujian dilakukan secara internal oleh penulis dan satu rekan tim dengan menjalankan setiap fitur permainan secara langsung untuk memastikan seluruh sistem dapat berjalan sesuai dengan fungsi yang telah dirancang.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Pengujian dilakukan terhadap fitur menu utama, tutorial, gameplay, spawn kendaraan, AI kendaraan, gerbang tol, ruas jalan tol, rescue, upgrade, antarmuka permainan (*user interface*), hingga sistem *game over*. Berdasarkan hasil pengujian yang telah dilakukan, seluruh fitur utama pada game *Toll Tycoon* berhasil berjalan dengan baik tanpa ditemukan kesalahan fungsi yang mengganggu jalannya permainan.

Tabel 4.17 Hasil Alpha Testing

No	Fitur	Pertanyaan	Hasil
1.	Menu Utama	Apakah tombol Mulai Game berfungsi dengan baik?	Berhasil
2.	Menu Utama	Apakah menu Pengaturan berfungsi dengan baik?	Berhasil
3.	Tutorial	Apakah tutorial tampil dengan jelas dan dapat digunakan dengan baik?	Berhasil
4.	Gameplay	Apakah permainan berjalan dengan kondisi awal yang sesuai?	Berhasil
5.	Gameplay	Apakah timer permainan berjalan dengan baik?	Berhasil
6.	Spawn Kendaraan	Apakah kendaraan muncul dengan normal?	Berhasil
7.	AI Kendaraan	Apakah kendaraan dapat bergerak dan mengikuti jalur dengan benar?	Berhasil
8.	AI	Apakah kendaraan dapat berhenti	Berhasil

	Kendaraan	dan bergerak kembali saat antrean terjadi?	
9.	AI Kendaraan	Apakah kendaraan keluar area setelah melewati gerbang tol?	Berhasil
10.	Gerbang Tol	Apakah kendaraan dapat memasuki gerbang yang sesuai?	Berhasil
11.	Gerbang Tol	Apakah proses transaksi kendaraan berjalan dengan baik?	Berhasil
12.	Gerbang Tol	Apakah sistem antrean kendaraan berjalan dengan baik?	Berhasil
13.	Gerbang Tol	Apakah detail visual pada aset Gerbang Tol Jagorawi mudah dikenali oleh pengguna ?	Berhasil
14.	Gerbang Tol	Apakah proporsi ukuran antara gerbang tol, gardu tol, dan jalur kendaraan terlihat sesuai ?	Berhasil
15.	Gerbang Tol	Apakah Material dan tekstur yang digunakan pada aset Gerbang Tol Jagorawi sudah sesuai dengan konsep visual yang dirancang ?	Berhasil
16.	Ruas Jalan Tol	Apakah Bentuk dan struktur ruas jalan tol telah sesuai dengan karakteristik jalan tol pada umumnya ?	Berhasil
17.	Ruas	Apakah marka jalan yang	Berhasil

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

	Jalan Tol	ditampilkan mudah dikenali dan dipahami?	
18.	Ruas Jalan Tol	Apakah Pembatas jalan dan elemen pendukung lainnya ditempatkan dengan tepat?	Berhasil
19.	Ruas Jalan Tol	Apakah proporsi lebar jalan dan jalur kendaraan terlihat realistis ?	Berhasil
20.	Ruas Jalan Tol	Apakah Material dan tekstur pada ruas jalan memberikan tampilan yang sesuai dengan kondisi jalan tol ?	Berhasil
21.	Rescue	Apakah fitur rescue dapat digunakan dengan baik?	Berhasil
22.	Rescue	Apakah sistem harga rescue berjalan dengan benar?	Berhasil
23.	Upgrade	Apakah fitur upgrade gerbang berfungsi dengan baik?	Berhasil
24.	Upgrade	Apakah sistem biaya dan level upgrade berjalan dengan benar?	Berhasil
25.	Upgrade	Apakah upgrade memberikan perubahan pada gameplay?	Berhasil
26.	UI	Apakah informasi saldo, kondisi, dan level gerbang tampil dengan jelas?	Berhasil
27.	UI	Apakah tombol, teks, dan ikon dapat digunakan dan dibaca dengan baik?	Berhasil

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian , penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

28.	UI	Apakah tampilan antarmuka berjalan dengan rapi tanpa tumpang tindih?	Berhasil
29.	Game Over	Apakah kondisi game over muncul dengan benar?	Berhasil
30.	Game Over	Apakah informasi hasil permainan ditampilkan dengan baik?	Berhasil
31.	Game Over	Apakah tombol kembali ke menu utama berfungsi dengan baik?	Berhasil

4.4.3.2 Hasil Beta Testing Umum

Pengujian Beta Testing oleh pengguna umum dilakukan untuk mengetahui tingkat penerimaan pengguna terhadap game Toll Tycoon sebagai media hiburan sekaligus media edukasi mengenai operasional pelayanan jalan tol. Pengujian dilakukan melalui proses memainkan game secara langsung, kemudian responden diminta mengisi kuesioner menggunakan skala Likert untuk memberikan penilaian terhadap aspek gameplay, visual, tantangan permainan, serta manfaat edukatif yang diperoleh setelah memainkan permainan.

Pada penelitian ini, pengujian dilakukan terhadap 42 responden pengguna umum yang telah memainkan game *Toll Tycoon*. Hasil lengkap pengujian *Beta Testing* oleh pengguna umum dapat dilihat pada Tabel 4.18.

Tabel 4.18 Hasil Beta Testing Pengguna Umum

No	Pertanyaan	Jawaban				
		STS	TS	C/N	S	SS
1.	Pengalaman bermain	0	0	3	18	21

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

	pada game Toll Tycoon terasa menyenangkan.					
2.	Gameplay pada game ini membuat saya tertarik untuk terus bermain hingga permainan selesai.	0	1	5	16	21
3.	Tantangan dalam mengelola gerbang tol terasa seru dan tidak membosankan.	0	2	6	12	20
4.	Fitur Maintenance, Rescue, dan Upgrade membuat permainan menjadi lebih menarik.	0	0	2	15	25
5.	Visual lingkungan Jalan Tol Jagorawi membuat pengalaman bermain terasa lebih imersif.	0	0	2	14	26
6.	Animasi dan pergerakan kendaraan terlihat hidup sehingga permainan terasa lebih realistis.	0	0	3	16	23

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

7.	Tingkat kesulitan permainan sudah sesuai dan memberikan tantangan yang seimbang.	0	1	5	13	23
8.	Saya merasa puas setelah memainkan game Toll Tycoon.	0	1	4	15	22
9.	Saya tertarik untuk memainkan game Toll Tycoon kembali di lain waktu.	0	2	4	15	21
10.	Secara keseluruhan game Toll Tycoon memberikan pengalaman bermain yang menghibur.	0	0	4	16	22
11.	Game ini memberikan gambaran mengenai operasional pelayanan di gerbang tol.	0	0	1	11	30
12.	Saya menjadi lebih memahami pentingnya menjaga kelancaran pelayanan kendaraan	0	0	2	18	22

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

	di gerbang tol.					
13.	Fitur Maintenance, Rescue, dan Upgrade membantu saya memahami pengelolaan jalan tol.	0	0	2	14	26
14.	Simulasi antrean kendaraan pada game membantu saya memahami proses pelayanan di gerbang tol.	0	0	2	14	26
15.	Secara keseluruhan, game Toll Tycoon mampu memberikan hiburan sekaligus memberikan informasi mengenai operasional dan pelayanan jalan tol.	0	0	1	13	28
Jumlah		0	5	55	207	363

Untuk menganalisis secara kuantitatif data umpan balik dari pengguna umum, setiap jawaban pada skala Likert dikonversi menjadi skor numerik, di mana Sangat Setuju (SS) bernilai 5 poin, Setuju (S) bernilai 4 poin, Cukup/Netral (C/N) bernilai 3 poin, Tidak Setuju (TS) bernilai 2 poin, dan Sangat Tidak Setuju (STS) bernilai 1 poin.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Berdasarkan hasil pengujian yang telah dilakukan oleh 42 responden, diperoleh total skor aktual sebesar 2818 poin. Nilai tersebut diperoleh dari hasil penjumlahan seluruh skor jawaban responden terhadap 15 butir pertanyaan menggunakan skala Likert. Rekapitulasi perhitungan skor aktual dapat dilihat pada Tabel 4.19.

Tabel 4.19 Hasil Skor Aktual

Kategori Jawaban	Frekuensi	Skor	Total Skor
Sangat Tidak Setuju (STS)	0	× 1	0
Tidak Setuju (TS)	5	× 2	10
Cukup/Netral (C/N)	55	× 3	165
Setuju (S)	207	× 4	828
Sangat Setuju (SS)	363	× 5	1815
Total	**630**	**2818**	

Sehingga total skor aktual diperoleh melalui perhitungan sebagai berikut.

$$(0 \times 1) + (5 \times 2) + (55 \times 3) + (207 \times 4) + (363 \times 5) = 2818$$

Adapun total skor maksimum yang dapat diperoleh apabila seluruh responden memberikan penilaian Sangat Setuju pada seluruh pertanyaan adalah sebagai berikut.

$$42 \times 15 \times 5 = 3150$$

Adapun total skor maksimum yang dapat diperoleh apabila seluruh responden memberikan penilaian Sangat Setuju pada seluruh pertanyaan adalah sebagai berikut.

$$\frac{2818}{3150} \times 100\% = 89,46\%$$

Berdasarkan kategori persentase skala Likert yang digunakan pada penelitian ini, hasil sebesar 89,46% termasuk ke dalam kategori "Sangat Setuju". Hasil tersebut menunjukkan bahwa pengguna umum memberikan penilaian yang sangat positif terhadap game Toll Tycoon. Mayoritas responden menilai bahwa permainan mampu memberikan pengalaman bermain yang menyenangkan, menghadirkan tantangan yang menarik melalui mekanisme pengelolaan gerbang tol, serta didukung oleh visual lingkungan dan animasi kendaraan yang baik sehingga meningkatkan pengalaman bermain. Selain itu, aspek edukasi yang diimplementasikan melalui simulasi operasional gerbang tol, pengelolaan lalu lintas, serta fitur Maintenance, Rescue, dan Upgrade juga dinilai mampu memberikan pemahaman kepada pemain mengenai proses pelayanan dan pengelolaan jalan tol. Dengan demikian, hasil Beta Testing menunjukkan bahwa Toll Tycoon telah berhasil memenuhi tujuan pengembangannya sebagai permainan yang menggabungkan unsur hiburan (*entertainment*) dan edukasi (*educational game*) dengan tingkat penerimaan pengguna yang sangat baik.

4.4.3.3 Hasil Beta Testing Expert Game

Pengujian *Beta Testing* oleh *expert game* dilakukan untuk mengevaluasi mekanisme gameplay, balancing permainan, sistem ekonomi, progression gameplay, serta sistem pengambilan keputusan pada game *Toll Tycoon*. Pengujian dilakukan melalui proses memainkan game secara langsung kemudian dilanjutkan dengan pengisian kuesioner menggunakan skala Likert.

Hasil lengkap pengujian *Beta Testing* oleh *expert game* dapat dilihat pada Tabel 4.20.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Tabel 4.20 Hasil Beta Testing Expert Game

No	Pertanyaan	Skor
1.	Mekanisme inti permainan mudah dipahami dan memiliki tujuan yang jelas.	4
2.	Sistem pengelolaan gerbang tol memberikan tantangan yang sesuai kepada pemain.	3
3.	Mekanisme antrean kendaraan berjalan secara logis dan mendukung gameplay simulasi.	3
4.	Variasi jenis kendaraan memberikan pengaruh terhadap strategi permainan.	3
5.	Sistem gameplay mendorong pemain untuk mengambil keputusan selama permainan berlangsung.	4
6.	Pemain perlu menentukan prioritas tindakan untuk mengelola kondisi permainan.	3
7.	Sistem maintenance memberikan dampak yang signifikan terhadap jalannya permainan.	3
8.	Fitur rescue memberikan solusi yang relevan terhadap kondisi tertentu dalam permainan.	2
9.	Mekanisme upgrade memberikan perkembangan gameplay yang terasa jelas.	5
10.	Sistem upgrade mendorong pemain untuk terus meningkatkan performa gerbang tol.	5
11.	Sistem pendapatan dalam game mampu memotivasi pemain untuk terus bermain.	3

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

12.	Pendapatan yang diperoleh pemain terasa seimbang dengan tantangan permainan.	2
13.	Peningkatan level gerbang memberikan pengaruh terhadap peningkatan pendapatan pemain.	5
14.	Sistem biaya maintenance dan rescue sudah seimbang dengan sistem pendapatan game.	2
15.	Sistem ekonomi dalam permainan mendorong pemain untuk membuat keputusan strategis.	3
16.	Mekanisme pengeluaran dan pemasukan dalam game terasa realistis untuk simulasi gerbang tol.	3
17.	Progress permainan terasa berkembang seiring bertambahnya pendapatan pemain.	4
18.	Game memberikan situasi yang mengharuskan pemain mengambil keputusan secara cepat.	4
19.	Pemain dapat menentukan strategi pengelolaan gerbang sesuai kondisi permainan.	4
20.	Pilihan penggunaan uang dalam game memengaruhi jalannya permainan.	4
21.	Pemain perlu mempertimbangkan kapan harus melakukan maintenance, rescue, atau upgrade.	2
22.	Sistem permainan memberikan konsekuensi terhadap keputusan yang diambil pemain.	3
23.	Gameplay yang disajikan mampu melatih kemampuan pengambilan keputusan pemain.	3
24	Tingkat kesulitan permainan meningkat secara	3

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

	bertahap dan seimbang.	
25.	Progress permainan terasa konsisten dari awal hingga akhir permainan.	3
26.	Sistem upgrade dan peningkatan tantangan sudah memiliki balancing yang baik.	3
27.	Tidak terdapat mekanisme yang terasa terlalu mudah atau terlalu sulit dalam permainan.	2
28.	Gameplay memiliki alur progression yang dapat mempertahankan minat pemain.	2

Untuk menganalisis secara kuantitatif data umpan balik dari *expert game*, setiap jawaban pada skala Likert dikonversi menjadi skor numerik, di mana "Sangat Setuju" bernilai 5 poin, "Setuju" bernilai 4 poin, "Cukup/Netral" bernilai 3 poin, "Tidak Setuju" bernilai 2 poin, dan "Sangat Tidak Setuju" bernilai 1 poin.

Berdasarkan hasil pengujian yang telah dilakukan, diperoleh total skor aktual sebesar 90 poin dari total 28 pertanyaan. Adapun total skor maksimum yang dapat diperoleh apabila seluruh pertanyaan mendapatkan penilaian "Sangat Setuju" adalah:

$$28 \times 5 = 140$$

Selanjutnya, untuk memperoleh persentase penilaian keseluruhan dilakukan perhitungan sebagai berikut:

$$\frac{90}{140} \times 100\% = 64,28\ \%$$

Berdasarkan kategori persentase skala Likert yang digunakan pada penelitian ini, hasil sebesar 64,28% termasuk ke dalam kategori "Setuju". Hasil tersebut menunjukkan bahwa *expert game* menilai

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

mekanisme gameplay, balancing permainan, sistem ekonomi, progression gameplay, dan sistem pengambilan keputusan pada game *Toll Tycoon* telah berjalan dengan cukup baik sesuai dengan tujuan simulasi pengelolaan gerbang tol yang dirancang pada penelitian ini.

Selain memberikan penilaian menggunakan skala Likert, *expert game* juga memberikan beberapa kritik dan saran terkait balancing gameplay, sistem ekonomi, progression permainan, dan pengalaman bermain secara keseluruhan sebagai bahan evaluasi dan pengembangan sistem permainan selanjutnya.

4.4.4 Analisis Data / Evaluasi Pengujian

Analisis data dan evaluasi pengujian dilakukan berdasarkan hasil *Alpha Testing* dan *Beta Testing* yang telah dilakukan pada game *Toll Tycoon*. Tahap ini bertujuan untuk mengevaluasi keberhasilan sistem permainan, kualitas gameplay, balancing permainan, serta pengalaman bermain pengguna selama simulasi berlangsung. Selain itu, proses evaluasi juga dilakukan untuk mengetahui apakah seluruh sistem yang telah diimplementasikan dapat berjalan sesuai dengan tujuan penelitian yang telah dirancang sebelumnya.

Berikut merupakan hasil analisis dan evaluasi dari proses pengujian game *Toll Tycoon*.

4.4.4.1 Analisis Alpha Testing

Berdasarkan hasil *Alpha Testing* yang telah dilakukan, seluruh fitur utama pada game *Toll Tycoon* dapat berjalan sesuai dengan fungsi yang telah dirancang pada tahap implementasi sistem. Pengujian menunjukkan bahwa sistem menu utama, tutorial, gameplay, spawn kendaraan, AI kendaraan, sistem gerbang tol, rescue, upgrade, antarmuka permainan (*user interface*), hingga sistem *game over*

dapat berjalan dengan baik tanpa ditemukan kesalahan fungsi yang mengganggu jalannya permainan secara keseluruhan.

Hasil pengujian juga menunjukkan bahwa sistem kendaraan mampu bergerak sesuai jalur yang telah ditentukan, membentuk antrean kendaraan secara otomatis, serta menjalankan proses transaksi pada gerbang tol dengan baik. Selain itu, sistem maintenance, rescue, dan upgrade juga dapat memberikan perubahan terhadap kondisi gameplay sesuai dengan rancangan permainan yang telah dibuat.

Pada aspek visual dan antarmuka permainan, hasil pengujian menunjukkan bahwa tampilan gerbang tol, ruas jalan tol, ikon, teks, serta informasi gameplay dapat ditampilkan dengan jelas dan mudah dipahami oleh pengguna. Sistem antarmuka juga dapat berjalan dengan rapi tanpa terjadi tumpang tindih tampilan selama permainan berlangsung.

Berdasarkan hasil pengujian fungsional tersebut, dapat disimpulkan bahwa seluruh sistem utama pada game *Toll Tycoon* telah berjalan secara stabil dan siap digunakan pada tahap *Beta Testing* oleh *expert game* dan pengguna umum.

4.4.4.2 Analisis Beta Testing oleh Pengguna Umum

Berdasarkan hasil Beta Testing yang dilakukan oleh pengguna umum, diperoleh persentase penilaian sebesar 89,46% yang termasuk ke dalam kategori "Sangat Setuju". Hasil tersebut menunjukkan bahwa game Toll Tycoon mampu memberikan pengalaman bermain yang sangat baik sekaligus memberikan informasi mengenai operasional pelayanan jalan tol kepada pengguna. Nilai tersebut mengindikasikan bahwa tujuan pengembangan permainan sebagai media hiburan (*entertainment*)

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

dan media edukasi (*educational game*) telah berhasil dicapai dengan tingkat penerimaan yang sangat baik dari pengguna umum.

Pada aspek hiburan, sebagian besar responden memberikan penilaian positif terhadap pengalaman bermain yang dihadirkan oleh game Toll Tycoon. Gameplay yang mengharuskan pemain mengelola operasional gerbang tol, mengatur kondisi lalu lintas, serta mengambil keputusan melalui fitur Maintenance, Rescue, dan Upgrade dinilai mampu memberikan tantangan yang menarik dan membuat permainan terasa lebih dinamis. Selain itu, visual lingkungan Jalan Tol Jagorawi, animasi kendaraan, serta kondisi lalu lintas yang disimulasikan secara dinamis juga dinilai mampu meningkatkan pengalaman bermain sehingga permainan terasa lebih imersif dan tidak membosankan.

Pada aspek edukasi, hasil pengujian menunjukkan bahwa mayoritas responden merasa memperoleh pemahaman mengenai operasional pelayanan di gerbang tol setelah memainkan permainan. Simulasi antrean kendaraan, pengelolaan kondisi jalan, serta mekanisme pelayanan kendaraan di gerbang tol mampu memberikan gambaran mengenai pentingnya menjaga kelancaran lalu lintas dan kualitas pelayanan jalan tol. Fitur Maintenance, Rescue, dan Upgrade juga dinilai berhasil memperkenalkan konsep dasar pengelolaan infrastruktur jalan tol melalui mekanisme permainan yang interaktif.

Hasil tersebut menunjukkan bahwa pendekatan simulasi yang diterapkan pada game Toll Tycoon tidak hanya mampu memberikan hiburan kepada pemain, tetapi juga menyampaikan informasi mengenai operasional dan pelayanan jalan tol secara lebih menarik dibandingkan penyampaian materi secara konvensional. Kombinasi antara unsur hiburan dan edukasi membuat pemain dapat memahami konsep pengelolaan jalan tol melalui pengalaman bermain secara langsung *(learning by doing)*.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Berdasarkan hasil evaluasi tersebut, dapat disimpulkan bahwa game Toll Tycoon telah berhasil memenuhi tujuan penelitian sebagai permainan simulasi yang menggabungkan aspek hiburan dan edukasi. Tingginya tingkat penerimaan pengguna yang ditunjukkan melalui persentase penilaian sebesar 89,46% menunjukkan bahwa mekanisme permainan, visual, serta materi edukasi yang disampaikan telah berjalan dengan baik dan mampu memberikan pengalaman bermain yang positif bagi pengguna umum.

4.4.4.3 Analisis Beta Testing Expert Game

Berdasarkan hasil *Beta Testing* yang dilakukan oleh *expert game*, diperoleh persentase penilaian sebesar 64,28% yang termasuk ke dalam kategori "Setuju". Hasil tersebut menunjukkan bahwa mekanisme gameplay, balancing permainan, sistem ekonomi, progression gameplay, dan sistem pengambilan keputusan pada game *Toll Tycoon* telah berjalan dengan cukup baik sesuai dengan konsep simulasi pengelolaan gerbang tol yang dirancang pada penelitian ini.

Hasil pengujian menunjukkan bahwa mekanisme inti permainan, sistem antrean kendaraan, sistem upgrade gerbang tol, dan sistem pengambilan keputusan mampu memberikan pengalaman simulasi manajemen yang cukup baik kepada pemain. Selain itu, gameplay dinilai mampu mendorong pemain untuk menentukan prioritas tindakan seperti melakukan maintenance, rescue, maupun upgrade gerbang tol sesuai kondisi permainan yang sedang berlangsung.

Pada aspek progression gameplay, sistem peningkatan level gerbang tol dinilai mampu memberikan perkembangan permainan yang cukup jelas dan memotivasi pemain untuk terus meningkatkan performa gerbang tol selama permainan berlangsung. Sistem ekonomi permainan juga dinilai telah mampu

menciptakan hubungan antara pemasukan dan pengeluaran sehingga pemain perlu mempertimbangkan penggunaan uang secara strategis selama simulasi berlangsung.

Meskipun demikian, berdasarkan hasil evaluasi dari *expert game*, masih terdapat beberapa aspek gameplay yang perlu dikembangkan lebih lanjut terutama pada balancing sistem rescue, balancing ekonomi permainan, dan progression tingkat kesulitan permainan. Beberapa mekanisme permainan dinilai masih dapat dikembangkan agar tantangan gameplay terasa lebih konsisten dan seimbang dari awal hingga akhir permainan.

Selain memberikan penilaian terhadap gameplay, *expert game* juga memberikan beberapa kritik dan saran terhadap balancing gameplay, sistem ekonomi, progression permainan, dan pengalaman bermain secara keseluruhan sebagai bahan evaluasi dan pengembangan sistem permainan selanjutnya.

Berdasarkan hasil evaluasi tersebut, dapat disimpulkan bahwa game *Toll Tycoon* telah mampu memberikan simulasi pengelolaan gerbang tol yang cukup baik serta berhasil menerapkan mekanisme gameplay berbasis pengambilan keputusan sesuai dengan tujuan penelitian yang telah dirancang

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

BAB V

PENUTUP

5.1 Simpulan

Berdasarkan hasil penelitian yang telah dilakukan, penelitian ini berhasil merancang dan membangun mekanik operasional pada game simulasi manajemen *Toll Tycoon* berbasis WebGL menggunakan Unity. Mekanik permainan yang berhasil diimplementasikan meliputi sistem antrean kendaraan, pengelolaan gerbang tol, sistem pendapatan, maintenance, rescue, upgrade gerbang tol, dan monitoring kemacetan yang direpresentasikan ke dalam bentuk gameplay interaktif berbasis pengambilan keputusan pemain.

Berdasarkan hasil pengujian yang telah dilakukan, seluruh sistem utama permainan berhasil berjalan dengan baik pada tahap *Alpha Testing*. Selain itu, hasil *Beta Testing* oleh *expert game* memperoleh persentase sebesar 64,28% dengan kategori "Setuju", sedangkan pengujian oleh pengguna umum memperoleh persentase sebesar 89% dengan kategori "Sangat Setuju". Hasil tersebut menunjukkan bahwa game *Toll Tycoon* telah mampu memberikan simulasi pengelolaan gerbang tol yang cukup baik serta memberikan pengalaman bermain yang positif bagi pengguna.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat digunakan untuk pengembangan penelitian maupun game *Toll Tycoon* selanjutnya. Pengembangan berikutnya dapat menambahkan variasi sistem lalu lintas yang lebih kompleks seperti perbedaan kepadatan kendaraan pada setiap gerbang tol, sistem cuaca dinamis, maupun kondisi lalu lintas secara *real-time* agar simulasi permainan menjadi lebih variatif dan interaktif.

Selain itu, pengembangan selanjutnya juga dapat meningkatkan sistem kecerdasan kendaraan (*vehicle AI*), balancing gameplay, serta variasi fitur operasional agar proses pengambilan keputusan pemain menjadi lebih menantang. Berdasarkan hasil evaluasi dari *expert game*, pengembangan berikutnya juga disarankan untuk menambahkan sistem *progression* dan *end condition* yang lebih jelas agar pemain tetap memiliki motivasi untuk melanjutkan permainan setelah seluruh upgrade gerbang tol terbuka.

Pengembangan berikutnya juga dapat meningkatkan visualisasi dan implementasi mekanik operasional seperti sistem rescue, kendaraan mogok, dan indikator *efficiency* agar fungsi setiap fitur lebih mudah dipahami oleh pemain selama permainan berlangsung. Selain itu, sistem penyimpanan data permainan juga dapat dikembangkan lebih lanjut agar progres pemain tetap tersimpan ketika permainan selesai atau mengalami *game over*.

Penelitian selanjutnya juga dapat mengembangkan integrasi antara sistem gerbang tol dan ruas jalan tol ke dalam satu gameplay scene agar gameplay loop yang dihasilkan menjadi lebih terintegrasi dan interaktif.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

DAFTAR PUSTAKA

Alyamani, H., Alharbi, N., Roboey, A. and Kavakli, M., 2023. The Impact of Gamifications and Serious Games on Driving under Unfamiliar Traffic Regulations. Applied Sciences, 13(5), p.3262. Available at: https://doi.org/10.3390/app13053262.

Amalia, A., et al., 2024. AI-Driven Traffic Simulation using Unity: Implementing Finite State Machines for Adaptive NPC Behaviour. Available at: https://www.researchgate.net/publication/385143681_AI-Driven_Traffic_Simulation_using_Unity_Implementing_Finite_State_Machines_for_Adaptive_NPC_Behaviour.

Arief, A., Muhammad, M. and Amin, F., 2023. Pengembangan Game Edukasi dengan Metode GDLC: Studi Kasus Mata Kuliah Algoritma dan Struktur Data. IJIS Indonesian Journal on Information System, 8(2), pp.120–128.

Arifudin, D., Suliswaningsih, Pramesti, D. and Heryanti, L., 2022. Implementasi Game Design Document Pada Perancangan Game-based Learning. Cogito Smart Journal, 8(2), pp.385–397.

Caroux, L. and Pujol, F., 2024. Player Enjoyment in Video Games: A Systematic Review and Meta-analysis of the Effects of Game Design Choices. International Journal of Human-Computer Interaction, 40(16), pp.4227–4238. Available at: https://doi.org/10.1080/10447318.2023.2210880.

Eshuis, J., et al., 2023. Player Experience and Enjoyment: A Preliminary Examination of Differences in Video Game Genre. Simulation & Gaming, 54(2), pp.209–220. Available at: https://doi.org/10.1177/10468781231158818.

Fu, K., Li, Y., He, J., Yuan, Y. and Mai, E., 2022. Innovative Learning and Training Approach in Business: A Systematic Business Games and

Literature Review. In: Proceedings of the 2022 6th International Conference on Education and E-Learning (ICEEL 2022). Yamanashi, Japan: ACM, pp.50–58. Available at: https://doi.org/10.1145/3578837.3578845.

Geetha, V., Gomathy, C.K., Vijayasimha and Vyshnavi, 2024. Optimal Decision Making in Games. International Journal of Scientific Research in Engineering and Management (IJSREM), 8(2). Available at: https://doi.org/10.55041/IJSREM28889.

Mehanna, N. and Rudametkin, W., 2023. Caught in the Game: On the History and Evolution of Web Browser Gaming. In: Companion Proceedings of the ACM Web Conference 2023. Austin, Texas, USA: ACM, pp.1–9. Available at: https://doi.org/10.1145/3543873.3585572.

Nova, S., Triasanti, D. and Hartiwati, E.N., 2025. Perancangan dan Implementasi Game Berbasis Android Menggunakan Unity 3D. Jurnal Minfo Polgan, 14(2). Available at: https://doi.org/10.33395/jmp.v14i2.15557.

Parreiras, M., Marques, P. and Xexéo, G., 2024. A Taxonomy of Game Balancing Elements. Available at: https://doi.org/10.5753/sbgames_estendido.2024.236999.

Raghothama, J., Hauge, J.B. and Meijer, S., 2022. Curating Player Experience Through Simulations in City Games. Urban Planning, 7(2), pp.253–263. Available at: https://doi.org/10.17645/up.v7i2.5031.

Rhamadan, M.D., 2025. Implementasi Finite State Machine untuk Dynamic Story dalam Game Simulasi Kampus. Techno Creative, 3(2), pp.91–102.

Rodrigues, P., et al., 2022. Player Engagement Analysis of a Business Simulation Game from Physiological, Psychological and Behavioral Perspectives: A Case Study. Applied Sciences, 12(19), p.10143. Available at: https://doi.org/10.3390/app121910143.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Rogers, K., Karaosmanoglu, S., Altmeyer, M., Suarez, A. and Nacke, L.E., 2022. Much Realistic, Such Wow! A Systematic Literature Review of Realism in Digital Games. In: CHI Conference on Human Factors in Computing Systems. New Orleans, LA, USA: ACM, pp.1–21. Available at: https://doi.org/10.1145/3491102.3501875.

Sewagegn, A., 2022. Optimization of Toll Services Using Queuing Theory in the Case of Ethiopia. Perner's Contacts, 17(2), pp.1–13. Available at: https://doi.org/10.46585/pc.2022.2.2400.

Suparno, A. and Utami, D., 2025. Mechanics of Serious Simulation Game for Karakuri Concept towards Low-Cost Automation. Greener Journal of Environmental Management and Public Safety, 13(1), pp.108–114. Available at: https://doi.org/10.15580/gjemps.2025.1.032725057.

Syarif, S., Hasanuddin, T. and Hasnawi, M., 2022. Perancangan Game Puzzle Labirin Menggunakan Metode Game Development Life Cycle (GDLC) Berbasis Unreal Engine. Buletin Sistem Informasi dan Teknologi Islam, 3(1), pp.34–41.

Universitas Islam Bandung, 2022. Simulasi Antrian Kendaraan Pada Gerbang Tol Pasteur Kota Bandung Menggunakan SimEvents MATLAB. Matematika: Jurnal Teori dan Terapan Matematika. Available at: https://journals.unisba.ac.id/index.php/matematika/article/view/1555.

Zhou, Y., 2024. Game Study on Morning Peak Road Congestion Problem. Proceedings of the 3rd International Conference on International Law and Legal Policy.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 1. Game Design Document TollTyconn

Toll Tycoon!

Game Design Document

Version 1.0

Team

(Name the people who are part of your development team, their roles and primary emails)

Name	Roles	Email-Mobile
Angga Yusma Prasetia	Programming	angga.yusma.prasetia.tik22@mhsw.pnj.ac.id
Yovan Athaillah Supriyadi	3D Artist	yovan.athaillah.supriyadi.tik22@mhsw.pnj.ac.id

Introduction

Toll Tycoon merupakan game simulasi manajemen jalan tol yang berfokus pada pengelolaan lalu lintas, operasional gerbang tol, serta penanganan kondisi darurat di ruas jalan secara real-time. Game ini terinspirasi dari kondisi lalu lintas jalan tol di Indonesia yang sering mengalami kepadatan kendaraan, kecelakaan ringan, kendaraan mogok, serta hambatan operasional lainnya yang dapat memengaruhi kelancaran arus lalu lintas.

Dalam permainan ini, pemain berperan sebagai manajer operasional jalan tol yang bertanggung jawab menjaga kelancaran kendaraan di seluruh area jalan tol. Pemain harus mengatur gerbang tol, meningkatkan kapasitas jalan, mengembangkan teknologi pembayaran, serta menangani berbagai permasalahan lalu lintas yang muncul secara dinamis. Selain mengelola kepadatan kendaraan, pemain juga harus menangani kendaraan bermasalah menggunakan sistem rescue serta melakukan maintenance pada ruas jalan yang dapat memengaruhi kecepatan kendaraan dan kondisi lalu lintas secara keseluruhan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Gameplay Toll Tycoon mengusung konsep simulation management dengan sistem progresi berbasis pengembangan infrastruktur dan pengelolaan sumber daya. Pemain akan memulai permainan dengan fasilitas terbatas, kemudian secara bertahap dapat membuka gerbang baru, meningkatkan pelayanan tol, memperluas ruas jalan, serta meningkatkan kemampuan operasional untuk menghadapi volume kendaraan yang semakin tinggi. Permainan juga menghadirkan kondisi dinamis seperti rush hour, peningkatan jumlah kendaraan, kendaraan mogok, dan penurunan kondisi jalan yang membuat pemain harus terus mengambil keputusan secara cepat dan strategis.

Tantangan utama dalam permainan adalah menjaga keseimbangan antara kapasitas jalan, kecepatan pelayanan gerbang tol, kondisi infrastruktur, dan volume kendaraan agar tidak terjadi kemacetan berlebihan. Jika kendaraan menumpuk terlalu lama atau gangguan lalu lintas tidak segera ditangani, maka kondisi jalan akan semakin padat dan dapat menyebabkan permainan berakhir (game over).

Melalui pendekatan simulasi ini, Toll Tycoon tidak hanya memberikan pengalaman bermain yang menantang dan interaktif, tetapi juga memperkenalkan konsep dasar pengelolaan transportasi, penanganan lalu lintas, dan pengambilan keputusan dalam kondisi operasional yang dinamis. Dengan visual 3D yang sederhana serta mekanik permainan yang mudah dipahami, Toll tycoon diharapkan mampu memberikan pengalaman bermain yang menarik sekaligus edukatif bagi pemain.

Target Audience

Toll Tycoon ditujukan untuk remaja hingga dewasa dengan rentang usia sekitar 17 hingga 35 tahun, khususnya pemain yang telah mengenal penggunaan kendaraan pribadi dan memahami aktivitas lalu lintas jalan raya maupun jalan tol. Game ini menargetkan pemain yang menyukai genre simulasi, manajemen, dan strategi, terutama yang tertarik pada pengelolaan transportasi serta pengambilan keputusan dalam kondisi dinamis. Namun, dengan mekanik permainan yang sederhana dan mudah dipahami, Toll Tycoon tetap dapat dimainkan oleh berbagai kalangan pemain kasual maupun pemain umum yang mencari pengalaman bermain yang menantang, interaktif, dan edukatif.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Genre

- Simulation
- Management
- Tycoon

Platform

Toll Tycoon dikembangkan untuk platform mobile web berbasis WebGL sehingga permainan dapat dimainkan secara langsung melalui browser pada perangkat smartphone tanpa perlu melakukan instalasi aplikasi tambahan. Game ini direncanakan untuk dipublikasikan melalui itch.io agar mudah diakses dan dimainkan pada berbagai perangkat yang mendukung browser modern. Platform berbasis web dipilih untuk memberikan kemudahan akses, fleksibilitas bermain, serta proses distribusi dan pembaruan game yang lebih efisien.

Game Structure

Goals

Primary Goals

Tujuan utama *Toll Tycoon* adalah memberikan pengalaman simulasi manajemen jalan tol yang menantang dan interaktif melalui pengelolaan arus lalu lintas, pengembangan infrastruktur, serta penanganan berbagai kondisi operasional secara real-time. Pemain dituntut untuk menjaga kelancaran kendaraan, mengatasi kemacetan, menangani kendaraan bermasalah menggunakan sistem *rescue*, serta melakukan *maintenance* jalan agar operasional jalan tol tetap berjalan dengan optimal.

Secondary Goals

Tujuan sekunder *Toll Tycoon* adalah memberikan pemahaman dasar mengenai sistem pengelolaan transportasi dan lalu lintas modern melalui gameplay yang edukatif dan mudah dipahami. Selain itu, game ini juga bertujuan melatih kemampuan pemain dalam berpikir strategis, mengambil keputusan cepat, mengatur prioritas pengembangan, serta mengelola sumber daya dalam

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

menghadapi kondisi lalu lintas yang dinamis seperti peningkatan volume kendaraan saat *rush hour*.

Interaction

1. Player Interaction

Interaksi pemain dalam *Toll Tycoon* berfokus pada pengambilan keputusan strategis melalui kontrol sentuhan (*touch input*) pada perangkat mobile. Pemain dapat membuka dan meningkatkan gerbang tol, memperluas ruas jalan, meningkatkan teknologi pembayaran, mengatur sistem *rescue* untuk menangani kendaraan bermasalah, serta melakukan *maintenance* pada jalan yang mengalami penurunan kondisi. Setiap tindakan pemain akan secara langsung memengaruhi kelancaran lalu lintas, kecepatan kendaraan, tingkat kemacetan, dan pendapatan operasional jalan tol. .

2. Interaction dengan Lingkungan (Environment Interaction)

Pemain dapat berinteraksi dengan berbagai elemen lingkungan dalam permainan seperti gerbang tol, ruas jalan, kendaraan, dan sistem lalu lintas yang berjalan secara real-time. Pemain harus memantau kepadatan kendaraan, kondisi jalan, serta menangani kendaraan mogok atau gangguan lalu lintas lainnya yang dapat memperlambat arus kendaraan. Selain itu, permainan memiliki kondisi dinamis seperti *rush hour* yang meningkatkan volume kendaraan dan membuat kondisi lalu lintas menjadi lebih sulit dikendalikan.

3. User Interface Interaction

Toll Tycoon menggunakan antarmuka yang dirancang sederhana dan responsif untuk platform mobile berbasis web. Pemain dapat berinteraksi dengan berbagai elemen UI seperti panel informasi untuk melihat jumlah koin, kondisi lalu lintas, status gerbang tol, serta kondisi jalan. Selain itu terdapat menu upgrade untuk meningkatkan fasilitas operasional, notifikasi sistem untuk memberikan peringatan saat terjadi kemacetan atau kendaraan bermasalah, serta indikator dan timer untuk membantu pemain memantau kondisi permainan secara real-time.

4. Challenge & Decision Interaction

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Pemain harus menghadapi berbagai tantangan dinamis selama permainan berlangsung, seperti lonjakan kendaraan saat *rush hour*, kendaraan mogok, penurunan kondisi jalan, serta keterbatasan sumber daya untuk melakukan upgrade dan perbaikan. Sistem ini menuntut pemain untuk menentukan prioritas tindakan, mengambil keputusan dengan cepat, serta mengelola strategi operasional secara efektif agar kondisi lalu lintas tetap terkendali dan tidak menyebabkan *game over*.

5. Feedback & Response System

Setiap interaksi pemain akan menghasilkan umpan balik visual dan audio untuk membantu pemain memahami dampak dari keputusan yang diambil. Umpan balik tersebut meliputi perubahan kecepatan kendaraan saat terjadi kemacetan, indikator kondisi lalu lintas, efek visual pada jalan yang mengalami kerusakan, animasi kendaraan bermasalah, serta efek suara peringatan ketika kondisi jalan mulai kritis. Sistem ini dirancang agar pemain dapat memahami situasi permainan dengan cepat dan menciptakan pengalaman bermain yang lebih interaktif serta responsif.

Obstacles

1. Infrastructure Limitations

Pada awal permainan, pemain memiliki infrastruktur jalan tol yang terbatas, seperti jumlah gerbang tol yang sedikit, kapasitas jalan yang sempit, serta sistem pembayaran yang masih lambat. Keterbatasan ini membuat arus kendaraan lebih mudah mengalami antrean dan kemacetan sehingga pemain harus menentukan prioritas pengembangan infrastruktur secara strategis.

2. Resource Limitations

Koin menjadi sumber daya utama dalam permainan yang digunakan untuk melakukan upgrade gerbang tol, perluasan jalan, peningkatan teknologi pembayaran, sistem rescue, dan maintenance jalan. Karena jumlah koin terbatas, pemain harus memilih prioritas pengeluaran secara efektif. Kesalahan dalam pengelolaan sumber daya dapat menyebabkan operasional jalan tol menjadi tidak optimal dan meningkatkan risiko kemacetan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

3. Rush Hour

Pada waktu tertentu, permainan akan memasuki kondisi rush hour yang menyebabkan volume kendaraan meningkat secara drastis dalam waktu singkat. Situasi ini membuat antrean kendaraan bertambah lebih cepat, meningkatkan tekanan terhadap sistem jalan tol, serta memaksa pemain untuk mengambil keputusan dengan cepat agar kondisi lalu lintas tetap terkendali.

4. Traffic Congestion

Kemacetan merupakan hambatan utama dalam permainan dan terjadi ketika jumlah kendaraan melebihi kapasitas jalan atau kemampuan pelayanan gerbang tol. Kemacetan dapat disebabkan oleh tingginya volume kendaraan, kapasitas jalan yang tidak memadai, sistem pembayaran yang lambat, maupun kendaraan bermasalah yang menghambat jalur lalu lintas. Jika tidak segera ditangani, antrean kendaraan akan terus meningkat dan mengurangi efisiensi operasional jalan tol.

5. Vehicle Breakdown & Emergency Situation

Permainan menghadirkan kondisi dinamis berupa kendaraan bermasalah atau mogok yang dapat muncul pada ruas jalan tertentu. Kendaraan tersebut akan memperlambat arus kendaraan lain dan berpotensi menyebabkan kemacetan apabila tidak segera ditangani menggunakan sistem rescue. Situasi darurat ini menambah tantangan bagi pemain dalam menjaga kelancaran lalu lintas secara real-time.

6. Road Condition & Maintenance

Kondisi ruas jalan akan menurun seiring berjalannya permainan apabila tidak dilakukan maintenance secara berkala. Jalan yang rusak atau memiliki kondisi buruk akan memengaruhi kecepatan kendaraan dan menyebabkan arus lalu lintas menjadi lebih lambat. Pemain harus mengelola perawatan jalan dengan baik agar performa operasional jalan tol tetap optimal.

7. System Pressure & Game Over

Permainan memiliki sistem tekanan berupa batas toleransi kemacetan. Jika antrean kendaraan terlalu panjang dan kondisi lalu lintas tidak dapat dikendalikan dalam waktu tertentu, maka permainan akan berakhir (game over). Sistem ini

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

dirancang untuk menciptakan ketegangan dan meningkatkan tantangan selama permainan berlangsung.

8. Difficulty Scaling

Seiring progres permainan, tingkat kesulitan akan meningkat melalui penambahan volume kendaraan, kondisi lalu lintas yang lebih kompleks, peningkatan frekuensi kendaraan bermasalah, serta kebutuhan maintenance yang lebih tinggi. Hal ini membuat pemain harus terus menyesuaikan strategi dan meningkatkan kemampuan pengelolaan operasional jalan tol.

Rules

1. Player Actions

Pemain dapat melakukan berbagai tindakan untuk menjaga kelancaran operasional jalan tol, seperti membuka dan meningkatkan gerbang tol, memperluas ruas jalan, meningkatkan teknologi pembayaran, melakukan *maintenance* jalan, serta mengaktifkan sistem *rescue* untuk menangani kendaraan bermasalah. Pemain juga dapat menggunakan koin yang diperoleh untuk melakukan upgrade infrastruktur dan meningkatkan efisiensi lalu lintas.

2. Resource Usage Rules

Setiap tindakan upgrade dan operasional membutuhkan sejumlah koin sebagai sumber daya utama. Pemain tidak dapat melakukan seluruh upgrade secara bersamaan apabila jumlah koin tidak mencukupi, sehingga pemain harus menentukan prioritas pengeluaran dan strategi pengembangan secara efektif.

3. Traffic Management Rules

Pemain harus menjaga agar arus kendaraan tetap berjalan lancar dan tidak melebihi kapasitas jalan maupun gerbang tol. Kendaraan akan terus masuk secara otomatis selama permainan berlangsung, dan pemain tidak dapat menghentikan aliran kendaraan secara manual. Jika antrean kendaraan terlalu panjang, tingkat kemacetan akan meningkat dan dapat menyebabkan kondisi *game over*.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

4. Rescue & Maintenance Rules

Kendaraan bermasalah yang muncul di jalan harus ditangani menggunakan sistem *rescue* agar tidak menghambat lalu lintas terlalu lama. Selain itu, pemain juga harus melakukan *maintenance* pada ruas jalan secara berkala untuk menjaga kondisi jalan dan kecepatan kendaraan tetap optimal. Pemain tidak dapat mengabaikan kondisi kendaraan bermasalah maupun kerusakan jalan karena keduanya akan berdampak langsung terhadap kelancaran lalu lintas.

5. Rush Hour Rules

Pada kondisi *rush hour*, jumlah kendaraan akan meningkat secara signifikan dan pemain harus tetap mengelola lalu lintas dalam kondisi tekanan tinggi. Pemain tidak dapat menghentikan atau melewati fase *rush hour*, sehingga diperlukan persiapan infrastruktur dan strategi yang baik sebelum kondisi tersebut terjadi.

6. Game Over Conditions

Permainan akan berakhir apabila tingkat kemacetan melebihi batas toleransi yang telah ditentukan atau ketika kondisi lalu lintas tidak dapat dikendalikan dalam waktu tertentu. Pemain harus terus menjaga keseimbangan antara kapasitas jalan, kecepatan pelayanan gerbang, kondisi jalan, dan volume kendaraan agar permainan dapat terus berlangsung.

Story

1. Beginning of the Game

- Pemain memulai permainan sebagai manajer operasional jalan tol.
- Infrastruktur awal masih terbatas, seperti jumlah gerbang tol yang sedikit dan kapasitas jalan yang kecil.
- Kendaraan mulai memasuki jalan tol secara otomatis dengan volume lalu lintas yang masih rendah.
- Pemain diberikan tujuan untuk menjaga kelancaran lalu lintas dan mengembangkan sistem operasional jalan tol.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

2. Core Gameplay Loop

Selama permainan berlangsung, pemain akan melakukan berbagai aktivitas utama, seperti:

- Mengelola arus kendaraan agar tidak terjadi kemacetan.
- Membuka dan meningkatkan gerbang tol.
- Memperluas kapasitas ruas jalan.
- Meningkatkan teknologi pembayaran untuk mempercepat pelayanan kendaraan.
- Menangani kendaraan bermasalah menggunakan sistem *rescue*.
- Melakukan *maintenance* jalan untuk menjaga kondisi dan kecepatan kendaraan.
- Mengelola penggunaan koin untuk upgrade dan pengembangan infrastruktur.

3. Dynamic Events & Challenges

Seiring permainan berjalan, pemain akan menghadapi berbagai kondisi dinamis, seperti:

- Peningkatan volume kendaraan.
- Kondisi *rush hour* yang menyebabkan lonjakan lalu lintas.
- Kendaraan mogok atau bermasalah di jalan.
- Penurunan kondisi jalan yang memengaruhi kecepatan kendaraan.
- Tingkat kesulitan yang terus meningkat seiring progres permainan.

4. Progression System

- Pemain memperoleh koin dari kendaraan yang berhasil melewati jalan tol.
- Koin digunakan untuk melakukan upgrade infrastruktur dan meningkatkan efisiensi operasional.
- Semakin berkembang jalan tol yang dikelola, semakin besar tantangan lalu lintas yang harus dihadapi pemain.

5. End of the Game

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Permainan akan berakhir (*game over*) apabila:

- Tingkat kemacetan melebihi batas toleransi.
- Antrean kendaraan terlalu panjang dan tidak dapat dikendalikan.
- Pemain gagal menangani kondisi lalu lintas dalam waktu tertentu.

Permainan dapat terus berlanjut selama pemain mampu menjaga kelancaran operasional jalan tol dan mengatasi berbagai tantangan yang muncul.

Level Design

1. Early Game

Pada tahap awal permainan, pemain memulai dengan infrastruktur jalan tol yang sederhana dan volume kendaraan yang masih rendah. Pemain diperkenalkan dengan mekanik dasar permainan seperti mengelola gerbang tol, memantau arus kendaraan, dan menggunakan koin untuk melakukan upgrade awal. Tantangan pada tahap ini masih relatif mudah agar pemain dapat memahami sistem permainan secara bertahap.

Fitur utama pada tahap awal:

- Jumlah gerbang tol terbatas.
- Volume kendaraan rendah.
- Belum banyak kendaraan bermasalah.
- Sistem *maintenance* dan *rescue* masih sederhana.

2. Mid Game

Pada tahap pertengahan permainan, volume kendaraan mulai meningkat dan kondisi lalu lintas menjadi lebih kompleks. Pemain mulai menghadapi kondisi *rush hour*, antrean kendaraan yang lebih panjang, serta kebutuhan upgrade infrastruktur yang lebih besar. Pada tahap ini pemain harus mulai mengatur prioritas penggunaan koin dan mengambil keputusan strategis agar lalu lintas tetap berjalan lancar.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Fitur utama pada tahap pertengahan:

- Volume kendaraan meningkat secara signifikan.
- Kendaraan bermasalah mulai muncul di ruas jalan.
- Sistem *rescue* mulai menjadi bagian penting gameplay.
- Pemain mulai melakukan perluasan jalan dan peningkatan teknologi pembayaran.

3. Late Game

Pada tahap akhir permainan, tingkat kesulitan meningkat secara drastis dengan volume kendaraan yang sangat tinggi dan kondisi lalu lintas yang lebih dinamis. Pemain harus mengelola berbagai sistem secara bersamaan, mulai dari pengaturan gerbang tol, penanganan kendaraan bermasalah, hingga *maintenance* jalan yang lebih intensif. Kesalahan kecil dapat menyebabkan kemacetan besar dan mempercepat kondisi *game over*.

Fitur utama pada tahap akhir:

- Kepadatan kendaraan sangat tinggi.
- Frekuensi *rush hour* meningkat.
- Kendaraan bermasalah muncul lebih sering.
- Kondisi jalan lebih cepat mengalami penurunan.
- Pemain membutuhkan strategi upgrade yang lebih efektif dan cepat.

4. Difficulty Progression

Tingkat kesulitan permainan meningkat secara bertahap seiring progres pemain. Peningkatan kesulitan dilakukan melalui:

- Penambahan jumlah kendaraan.
- Peningkatan kecepatan kedatangan kendaraan.
- Penambahan frekuensi kendaraan bermasalah.
- Penurunan kondisi jalan yang lebih cepat.
- Tekanan waktu yang semakin tinggi saat kondisi lalu lintas mulai padat.

Sistem progresi ini dirancang agar permainan tetap menantang dan mendorong pemain untuk terus mengembangkan strategi pengelolaan jalan tol.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

5. Level Flow

Setiap level dalam *Toll Tycoon* memiliki pola permainan yang serupa namun dengan tingkat kompleksitas yang berbeda. Pemain akan:

1. Memulai dengan kondisi lalu lintas normal.
2. Mengelola kendaraan dan mengembangkan infrastruktur.
3. Menghadapi peningkatan volume kendaraan dan kondisi dinamis.
4. Menangani berbagai hambatan seperti kemacetan, kendaraan mogok, dan kerusakan jalan.
5. Bertahan selama mungkin tanpa melewati batas toleransi kemacetan.

Items and Characters

1. Vehicles

Kendaraan merupakan elemen utama yang terus bergerak di dalam permainan dan menjadi sumber pendapatan pemain ketika berhasil melewati jalan tol. Setiap kendaraan memiliki kecepatan dan perilaku pergerakan yang dapat dipengaruhi oleh kondisi lalu lintas, kapasitas jalan, serta kondisi ruas jalan.

Pengaruh dalam permainan:

- Menambah pendapatan pemain saat melewati gerbang tol.
- Menyebabkan antrean dan kemacetan apabila jumlah kendaraan terlalu banyak.
- Menjadi indikator utama tingkat kepadatan lalu lintas.

2. Toll Gates

Gerbang tol berfungsi sebagai titik pelayanan kendaraan untuk memasuki atau keluar dari jalan tol. Pemain dapat membuka dan meningkatkan gerbang tol untuk meningkatkan kapasitas pelayanan kendaraan.

Pengaruh dalam permainan:

- Menentukan kecepatan antrean kendaraan.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- Memengaruhi jumlah kendaraan yang dapat diproses.
- Upgrade gerbang dapat mengurangi kemacetan dan meningkatkan efisiensi lalu lintas.

3. Road Infrastructure

Ruas jalan merupakan jalur utama kendaraan bergerak selama permainan berlangsung. Pemain dapat memperluas kapasitas jalan dan melakukan *maintenance* untuk menjaga kondisi jalan tetap optimal.

Pengaruh dalam permainan:

- Memengaruhi kecepatan kendaraan.
- Kondisi jalan yang buruk dapat memperlambat arus kendaraan.

4. Rescue Unit

Sistem *rescue* digunakan untuk menangani kendaraan bermasalah atau mogok yang muncul di ruas jalan. Unit ini akan membantu membersihkan jalur lalu lintas agar kendaraan lain dapat kembali bergerak dengan normal.

Pengaruh dalam permainan:

- Mengurangi dampak kendaraan mogok terhadap lalu lintas.
- Membantu menjaga kelancaran arus kendaraan.
- Menjadi sistem penting pada tingkat kesulitan tinggi.

5. Maintenance System

Sistem *maintenance* digunakan untuk memperbaiki kondisi jalan yang mengalami penurunan kualitas akibat penggunaan terus-menerus selama permainan berlangsung.

Pengaruh dalam permainan:

- Menjaga performa jalan tetap optimal.
- Meningkatkan kecepatan kendaraan.
- Mengurangi risiko kemacetan akibat kondisi jalan yang buruk.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

6. Coin

Koin merupakan sumber daya utama dalam permainan yang diperoleh dari kendaraan yang berhasil melewati jalan tol. Koin digunakan untuk melakukan berbagai upgrade dan pengembangan infrastruktur.

Pengaruh dalam permainan:

- Digunakan untuk membeli upgrade.
- Menentukan perkembangan sistem jalan tol.
- Menjadi faktor utama dalam strategi pengelolaan permainan.

7. Traffic System

Sistem lalu lintas mengatur pergerakan kendaraan secara real-time, termasuk kepadatan kendaraan, antrean, dan kondisi *rush hour*.

Pengaruh dalam permainan:

- Menentukan tingkat kesulitan permainan.
- Memengaruhi kecepatan dan kelancaran kendaraan.
- Menjadi indikator utama keberhasilan pemain dalam mengelola jalan tol.

8. User Interface Elements

Elemen antarmuka seperti panel informasi, indikator lalu lintas, notifikasi, dan timer membantu pemain memantau kondisi permainan secara real-time.

Pengaruh dalam permainan:

- Memberikan informasi penting kepada pemain.
- Membantu pemain mengambil keputusan lebih cepat.
- Menampilkan kondisi kritis seperti kemacetan dan *game over*.

Gameplay

1. Main Menu

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Saat permainan dimulai, pemain akan masuk ke halaman *Main Menu* yang berfungsi sebagai pusat navigasi utama permainan. Pada menu ini pemain dapat:

- Memulai permainan (*Play*).
- Mengatur suara permainan melalui menu *Settings*.
- Keluar dari permainan (*Exit*).

Menu utama dirancang sederhana dan mudah dipahami agar pemain dapat langsung mengakses permainan dengan cepat.

2. Core Gameplay

Pemain memainkan *Toll Tycoon* dengan mengelola operasional jalan tol secara real-time menggunakan kontrol sentuhan (*touch input*) pada perangkat mobile. Selama permainan berlangsung, kendaraan akan terus memasuki area jalan tol dan pemain harus menjaga agar lalu lintas tetap lancar.

Aktivitas utama pemain meliputi:

- Membuka dan meningkatkan gerbang tol.
- Memperluas kapasitas jalan.
- Meningkatkan teknologi pembayaran.
- Mengatur sistem *rescue* untuk kendaraan bermasalah.
- Melakukan *maintenance* jalan.
- Mengelola penggunaan koin untuk upgrade infrastruktur.
- Mengatasi kemacetan dan kondisi *rush hour*.

Gameplay berfokus pada pengambilan keputusan cepat dan strategi pengelolaan lalu lintas agar kendaraan dapat terus bergerak dengan efisien.

3. In-Game UI Elements

Selama permainan berlangsung, pemain akan menggunakan berbagai elemen antarmuka (*User Interface*) yang memengaruhi gameplay secara langsung, antara lain:

Coin Indicator

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Menampilkan jumlah koin yang dimiliki pemain dan digunakan untuk melakukan upgrade.

Traffic Indicator

Menampilkan kondisi lalu lintas saat ini, mulai dari lancar, padat, hingga macet.

Rush Hour Timer

Menampilkan durasi kondisi *rush hour* yang sedang berlangsung.

Upgrade Button

Digunakan untuk membuka menu peningkatan gerbang tol, jalan, teknologi pembayaran, *rescue*, dan *maintenance*.

Road Condition Indicator

Menampilkan kondisi ruas jalan yang memengaruhi kecepatan kendaraan.

Rescue Notification

Memberikan peringatan ketika terdapat kendaraan bermasalah yang harus segera ditangani.

Pause Menu

Memungkinkan pemain untuk menghentikan permainan sementara, melanjutkan permainan, atau kembali ke menu utama.

4. Upgrade System

Pemain dapat meningkatkan berbagai fasilitas jalan tol menggunakan koin yang diperoleh selama permainan berlangsung. Sistem upgrade memungkinkan pemain meningkatkan kapasitas dan efisiensi operasional untuk menghadapi tingkat kesulitan yang semakin tinggi.

Upgrade meliputi:

- Gerbang tol.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

- Teknologi pembayaran.
- Sistem *rescue*.
- Sistem *maintenance*.

5. Game Over Screen

Permainan akan berakhir apabila tingkat kemacetan melebihi batas toleransi yang ditentukan. Saat kondisi *game over* terjadi, pemain akan melihat informasi hasil permainan seperti total koin yang diperoleh dan pilihan untuk mengulangi permainan atau kembali ke menu utama.

References

Game Referensi (Gameplay & Mekanik)

- *Mini Metro*
  Referensi utama untuk manajemen lalu lintas, visual sederhana, dan sistem tekanan waktu.

- *Cities: Skylines*
  Inspirasi sistem simulasi transportasi, kepadatan lalu lintas, dan manajemen infrastruktur.

- *Traffic Run!* dan *Idle Traffic Tycoon*
  Referensi gameplay casual berbasis pengaturan arus kendaraan dan progresi bertahap.

- *SimCity Series*
  Acuan sistem ekonomi, balancing, dan konsekuensi keputusan pemain.

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 2. Transkrip Wawancara

[00:00] Oke, jadi sebelum pertanyaannya dimulai, saya mau ngejelasin Pak ada beberapa poin besar pertanyaan Pak terkait manajemen operasional jalan tol yang pertama itu ada di identifikasi permasalahan utama melalui sudut pandang jasa marga lalu ada analisis pola kepadatan dan waktu kritis lalu ada sumber gangguan operasional di lapangan [00:29] Lalu ada prosedur penanganan dan pengambilan keputusan Nah itu aja Pak Ada 3 poin berarti ya? Ada 3 poin [01:09] Hai dari akhirnya percaya Hai kata-kata Hai harusnya enggak Hai kita harusnya berpengaruh di jauhnya Hai Oke untuk pertanyaan pertama [01:32] Apakah Bapak bisa menjelaskan secara singkat peran dan tanggung jawab jasa marga dalam pengelolaan lalu lintas jalan tol? Tanggung jawab secara spesifik ya Tentang pengelolaan lalu lintasnya ya, karena lalu lintas nih Secara spesifik Jadi untuk pengelolaan lalu lintasnya [02:02] Semua untuk menjamin kelancaran pengguna jalan Jadi intinya pelayanan lalu lintas di jalan tol itu melayani pengguna jalan secara lancar aman-nyaman Berarti jasa marga ini bisa dibilang sebagai perusahaan pelayanan Tul [02:37] Menurut Bapak, permasalahan utama apa yang paling sering menyebabkan kemacetan di tol Jagorawi Pak? Baik di pintu maupun di ruas jalan tolnya Pak Kalau di Jagorawi memang tipe jalannya dan tipe lalu lintasnya berbeda dengan ruas lain [03:04] Seperti contohnya di dalam kota Kalau di dalam kota kan Pintu keluar Akses pintu keluar tolnya Bersinggungan langsung dengan Jalan arteri terlalu dekat Tapi kalau Jaburawi Masih Tergolong Aman Karena jauh dari jalur arteri Kita bersinggungan dengan arterinya Terlalu Lebih jauh dari Ruas lain [03:33] Artinya begini, kalau di arteri biasanya kan ada lampu merah, ada persimpangan jalan. Nah ini yang potensi membuat kepadatan lalu lintas. Namun kalau di Jagorawi, untuk hal-hal tersebut kayak perpotongan di arteri dan lampu merah di arteri, [04:02] dibilang hanya sebagian hanya sedikit jadi untuk lamu merah atau persinggungan dengan arteri itu hanya beberapa titik contoh kalau di jagorawi yang sering terjadi kepadatan akibat [04:23] Persinggungan dengan arteri itu di keluar Bogor, Meranangsiang. Terus di Ciawi. Keluarnya berdekatan dengan Lahu Merah yang di situ notabene ada pasar. Nah ini yang bikin kepadatan. Tapi untuk yang lainnya seperti di keluar Gunung Putri misalkan, keluar Citrep, keluar Sentul, itu hampir [04:53] hanya sebagian kecil

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

akibat dari parteri. Nah, yang jadi permasalahan utama kalau kepadatan itu biasanya terutama di hari-hari libur panjang atau weekend. Kalau Jagorawi termasuknya jalur wisata. Nah, contohnya di [05:22] Ciawi, Ciawi itu kan arah ke Puncak ini sudah rutinitas disitu pasti di pagi hari, hari Sabtu itu pasti terjadi kepadatan yang mengarah ke Puncak nah disitu kita berkoordinasi dengan kepolisian wilayah Kabupaten Bugur karena Puncak ini wilayahnya Kabupaten Bugur Polres Kabupaten Bugur disitu [05:53] dari pihak lantasnya diadakan rekayasa lalu lintas. Terus dari kita juga sedikit banyak akan apa ya, dari rekayasa itu akan berpengaruh besar. [06:20] Karena lalu lintas yang dari tol ini adalah memang penyumbang terbesar arah ke puncak. Jadi awal kalau pagi hari Sabtu itu awal diperlakukan ganjil genap untuk sortir kendaraan biar nggak terlalu padat arah puncak. Nah nanti biasanya itu diperlakukan dari jam 6 sampai jam 8 atau 9 pagi. Nanti setelah jam 8 rekayasinya diubah bukan ganjil genap lagi. [06:50] dia dibikin one way satu arah-arah puncak jadi yang dari tol dibikin satu arah yang dari atas sudah ditutup nih yang dari puncaknya ditutup yang dari tol di lost ke atas semua nah ini upaya dari pepolisian wilayah untuk memperlancar menarik antrian yang dari tol bisa langsung ke puncak nah ini biasanya jam 9 jam 8-9 pagi udah lancar nih ke atas nah nanti [07:20] jam 12 ini dibalik jam 12 pengunjung puncak ini mulai turun mulai turun dibalik one way nya dibalik ke arah Jakarta yang ke arah puncaknya ditutup dari tol gadok ditutup one way prioritas turun ini ini bermasalahnya yang [07:50] Setiap air pekan Jagurawi yang menimbulkan kepadatan diadakan rekayasa untuk melancarkan pengunjung puncak ini keluar dan masuk Ciawi. [08:07] Tidak ada Andriang Kalau dari sisi pengendara gitu Jadi langsung kayak Akibat Pengendara mobil A, mobil B Jalanan jadi macet itu Ada masalah itu gak Pak? Kalau Dari Pemakai jalan Biasanya yang bikin macet itu Kalau terjadi [08:35] gangguan perjalanan salah mogok Oh baik kendaraan besar apalagi ya karena Jagorawi kan tidak ada pembatasan kendaraan besar seperti dalam kota ada pembatasan nih siang nggak ada truk masuk kalau Jagorawi nggak ada pembatasan artinya 24 jam kendaraan besar melintas Jagorawi nah ini yang jadi kendala biasanya [09:01] ada satu kendaraan besar apabila mengalami gangguan apalagi di jalur utama ya itu akan menghambat kendaraan yang lain akhirnya itu akan jadi kepadatan jadi upaya kita [09:21] Ya

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kita percepatan penanganan kendaraan gangguan tersebut Karena kita mengacu ke SPM, Standar Pelayanan Minimum Jalan Kau Jadi untuk satu ruas itu dihitung visi rasionya Tidak boleh lebih dari 0,8 Kalau sudah 0,8 [09:48] Kita harus ambil tindakan untuk rekayasa lalu lintas. Atau kalau itu kontraflow, atau pengalihan arus. Nah ini tentunya nanti kita koordinasi dengan PJR. Karena yang punya izin, punya wawenang diskresi untuk mengizinkan rekayasi itu dari PJR. Berarti tetap ada kolaborasi dengan pihak PJR ya Pak ya? [10:16] Berarti terkait pengelolaan seperti kontraflow, one way dan lainnya itu ada indikator juga ya Pak? Ada, ada. Tadi apa Pak namanya indikatornya? Visi ratio. Visi ratio. Itu dari perhitungan visi ratio. Volume kendaraan dibagi dengan kapasitas jalan. Nanti ketemunya kalau di atas 0,8 artinya itu udah macet. [10:44] Itu harus kita rekayasa Kalau masih di bawah 0,8 itu masih Padat tapi masih wajar Ini masih memenuhi SPM [11:01] Dari pengalaman Bapak di Operasional Jalan Tol bagian mana yang paling krusial dalam memicu kemacetan? Pintu tol, ruas jalan tol atau kombinasi dari keduanya Pak? Kalau yang paling pengaruh macetan itu biasanya [11:29] Saya jelaskan tadi, jadi pintu keluar yang arah ke tempat-tempat wisata. Karena disitu pasti ada puncak, puncak lalu lintasnya naik. Suruh kesini aja, Bu. Oke, baik. [11:55] Selanjutnya, apakah dari jasa marga pernah melakukan evaluasi khusus terkait kemacetan di tol Jagorawi? Dan apa masalah yang paling sulit dikendalikan? Kalau evaluasi itu sudah pasti ya. Kita tiap bulan itu kita kan membuat laporan untuk kebanyakan. [12:21] situasi maupun laporan kondisi lalu lintas yang ada di Jalan Tor khususnya Jagorawi. Jadi kita tiap bulan itu ada pelaporan untuk ke pusat di Samarga. Itu nanti laporannya untuk lalu lintasnya banyak terkait nanti ada kepadatan lalu lintas, ada titik-titik black spot kecelakaan yang sering terjadi kecelakaan, terus [12:52] jumlah korban kecelakaannya ada berapa. Nah ini tugas kita evaluasi nanti. Bagaimana caranya kita bisa menurunkan akal kecelakaan itu. Ini penyebabnya apa? Kita evaluasi penyebabnya apa? Apakah dari kondisi jalan? Nanti Pak. Kondisi jalan, ataukah dari kendaraan, faktor kendaraan, atau dari faktor pengemudi. Suruh di dalam aja. Di dalam aja tuh Pak. Situ ada bangku. Maaf dulu Pak. [13:23] enggak saya potong saya potong ya saya potong ya enggak apa-apa silahkan bentar oke bisa dilanjut lagi ya Pak ya oke tadi kan Bapak udah ngomong kepadatan waktu di apa ya tempat wisata ya biasanya kan

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Sabtu Minggu weekend lah Pak ya [13:51] Kalau di weekday gitu Pak, apakah ada waktu tertentu pola kepadatan lalu lintasnya gitu Pak? Jika hari-hari kerja ya, hari biasa, kepadatan itu biasanya puncaknya, kalau pagi di jam berangkat kerja. Sore pulang kerja, itu udah rutinitas. Jadi upaya kita apa? Di situ untuk mengurangi kepadatan, [14:20] Sama, jadi kita punya solusi Kita tentunya koordinasi sama pihak PCR Kita mengalami rekayasa lalu lintas Untuk melancarkan arus lalu lintas Kalau pagi Itu mayoritas lalu lintas dari Bogor, dari arah Bogor, arah Jakarta Karena banyak bekerja ke Jakarta Ini misalnya ada kewadatan nih [14:51] Nah, kita dari Jagorawi khususnya itu tiap hari melakukan rekayasa kontraflow. Kita melakukan kontraflow meraih dari kilometer 17, itu sekitaran Cimanggis, itu dibuka sampai kilometer 8, daerah Pasarbu. [15:19] situs dan rutinitas setiap pagi jadi memang kontravo ini satu-satunya upaya kita rekayasa yang bisa mengurangi kepadatan lalu lintas yang arah Jakarta Oke selanjutnya tuh untuk lonjakan volume kendaraan tuh biasanya terjadi secara bertahap atau mendadak gitu Pak [15:49] Kalau pagi ya kita biasanya di jam-jam puncaknya itu jam sekitar jam jam teman lah mereka berangkat kerja itu ada kepadataan biasanya arah Jakarta [16:13] Itu pusat kepadatannya dari mulai masuk tol dalam kota itu udah padat. Berarti imbas dari kepadatannya itu sampai Jagorawi. Jadi ini kepadatan di Jagorawi ini sebagai imbas dari dalam kota sebenarnya. Bukan dari ruas Jagorawi sendiri. [16:30] Berarti saling berkaitan juga sebenarnya? Berkaitan, karena di dalam kotanya sudah padat, masuk gerbang sililitan padat, atau arah ke uki nya padat, imbatnya ke Jagorawi. Oh baik, betul. Berarti bisa dibilang lonjakan volume itu bertahap juga ya? Bertahap. Dari luas tol lain, terus masuk ke Jagorawi. Betul. [16:54] Oke, tadi kan kita sudah membahas ini Pak, beberapa gangguan untuk kemacetan ya Bapak? Nah sekarang bisa Bapak urutkan dari gangguannya yang menurut Bapak paling mengganggu untuk yang memicu kemacetan yang sangat signifikan di Tol Jagorawi Pak Yang pertama A. Kecelakaan lalu lintas B. Kendaraan mogok [17:23] C. Penyempitan jalur dikarenakan maintenance D. Faktor cuaca Kalau yang yang pasti ini ya penyebab kepadatan yang memerlukan waktu yang lama pengurahannya untuk waktu yang lama biasanya saat maintenance [17:52] maintenance ini kenapa maintenance ini yang sering kita alami kan rekonstruksi [18:03] Jalan dibobok, sudah berlubang,

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

sudah tidak bisa diapa-apakan Itu kan tidak bisa dilewati Ini yang akan penyebab kepadatan yang sudah ibaratnya sudah permanen Karena rekayasa apapun sudah tidak bisa Butuh waktu yang lama juga Nunggu artinya nunggu proyek itu selesai baru [18:30] Kalau kecelakaan kan tergantung Seberapa berat kejadiannya Kalau kejadiannya melibatkan kendaraan besar Ada korban atau ada material yang kumpah dari kendaraan angkutan Ini mungkin perlu waktu yang lama juga Apalagi [18:54] muatan jenis muatannya mungkin bahan yang berbahaya ini kita kan nggak bisa langsung evakuasi ini harus menunggu pihak yang beronak dulu panggil yang punya barang instansi terkait kalau bahan B3 kan kita harus antisipasi bahaya apa yang terjadi baik buat kita maupun penggunaan jalan yang lain ini yang bikin lama tapi kan itu bisa bisa [19:24] diukur, enggak sampai berhari-hari. Karena kita punya SOP, kalau ada kecelakaan, percepatan penanganan. Sebagaimana cepat kita singkirkan kendaraan, bisa dilewatin, bisa lancar lagi. Jadi enggak makan waktu lama. Baik, baik. [19:53] Menurut bapak gangguan mana yang paling sulit diprediksi dan dikendalikan? Kalau gangguan perjalanan ya, kalau main tanan udah jelas itu terencana. Jadi kita udah bisa mapping, kita bisa menentukan. [20:13] mulai kerjaannya hari apa selesainya terhari apa terus kita bisa prediksi ini puncak kepadatan lalu lintasnya nanti jam-jam berapa itu kita bisa antisipasi tapi kalau bangguan perjalanan misalnya kendaraan pokok kendaraan rusak spare partnya ini yang enggak bisa kita prediksi tiba-tiba mobil truk tronton [20:41] ada di lajur 1 atau lajur 2 baut rodanya lepas beres ya? ini yang kita prediksi jadi kejadian itu bisa cara tiba-tiba kita kan gak tahu mobil itu layak jalan atau enggak muatannya overload atau enggak [21:11] Kebanyakan kendaraan besar yang rusak bagian roda itu biasanya overload Baut rodanya patah, rodanya kan lepas, itu kan perlu penanganan lama Nah ini [21:31] nanti disitu makanya dengan adanya sering terjadinya kejadian gangguan seperti itu kita punya ide kita punya solusi alat apa nih harus kita pakai kita punya inovasi untuk mempercepat evakuasi kendaraan yang penting [21:53] Di saat ada kejadian, kita percepatkan penanganan. Kendaraan itu ke pinggir, tidak mengganggu pengguna jalan yang lain. Baik Pak. Selanjutnya ada di pertanyaan prosedur penanganan dan pengambilan keputusan Pak. Oke. Ketika antrian di pintu tol atau ruas jalan mulai memanjang, langkah operasional apa yang pertama

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

kali dilakukan? Oke. Kalau antrian di... [22:23] itu tol, itu kan sudah pasti itu dari banyaknya volume kendaraan dan itu tidak kalau di Jagorawi ya tidak setiap hari atau setiap saat jadi waktu waktu tertentu nah ini kalau Jagorawi saya bilang tadi karena jalur wisata biasanya di hari-hari weekend itu pasti ada kepadatan [22:52] antisipasi apa yang kita untuk menghadapi antrian gerbang? yang pertama kita yang sudah kita lakukan itu namanya pakai sistem mobil reader jadi kita jemput bola dari antrian yang panjang ada petugas yang membawa reader emani tempel dia jalan ke depan jadi antrian lebih cepat tuh [23:23] itu sudah terbukti ya, itu untuk mengurangi kepadatan yang kedua penambahan gardu satelit jadi pas kalian lihat di belakang gardu utama itu ada gardu yang dibangun kecil-kecil gardu yang miring itu suatu upaya untuk mengurangi kepadatan itu juga sudah terbukti itu bisa mengurangi antrian [23:53] Jadi itu akhirnya sudah kita jalankan, gak cuma di Jagora semua ruas kita berlakukan seperti itu. Menurut Bapak, apakah terdapat standar waktu toleransi antrean sebelum dilakukan tindakan tertentu? Kita kembali ke SOP. SOP untuk yang diberbang itu. [24:22] kalau gak salah nih saya ya karena saya traffic nih saya gak paham banget dengan di gerbang untuk waktu transaksi itu maksimal 2 detik 2 detik itu maksimal ya Pak maksimal 2 detik jadi untuk antisipasi panjangnya antrean itu SOP ya jadi kalau memang sekarang kan ngetap doang dulu iya kalau sudah 2 detik berarti udah [24:51] dia artinya SOP udah dijalankan enggak tidak menyebabkan padatan yang panjang hanya sekarang dikembangkan di gerbang-gerbang itu nir sentuh iya nir sentuh flow gitu kan jadi kita lewat langsung buka nggak perlu berhenti dan tidak perlu membuka kaca di SOP ya oke [25:24] Sambil rokok apa ya? Ya, silahkan Pak. Menurut Bapak, apakah teknologi gerbang tol itu sangat berpengaruh terhadap jumlah antrean, Pak? Yang sudah kita jalankan sangat berpengaruh. Sangat berpengaruh. Dulu pertama ada tol, [25:52] tiap gardu itu ada orang, iya kan itu kan perlu waktu orang bayar cash belum kalau uang gede ada kembaliannya berapa menit prosesnya sekarang akhirnya kegulungan langganan langganan gak usah pakai kembalian jadi [26:18] bayarnya sebulan sekali langganan Oh gitu sekarang udah ada itu ya Sekarang berkembang lagi emane emane emane ngetep doang berkembang lagi bro ini nanti rencana kedepannya infonya kan pakai MLFF multi-line [26:43] MLF Free Flow berarti dia tanpa berhenti langsung tanpa

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

berhenti alat itu adanya di tengah jalur tapi masih uji coba nanti mas kalau lewat Cimanggis kilometer 18 itu udah dipasang terus di cekir sini luar bambu hapus ada di situ ada kamera itu nyorot ke nomor polisi kendaraan jenis kendaraan [27:11] sama gulungan kendaraan. Nanti itu mungkin di kendaraannya harus membeli alat-alat perangkat yang connect ke situ. Cuman dari evaluasi kita sih kayaknya Indonesia belum siap ya Pak. Belum siap. [27:40] Kalau dari Bapak sendiri untuk staff-staff di lapangan yang terjun langsung itu pembelajarannya itu melalui simulasi atau praktek atau gimana Pak? Kalau saya, pengalaman saya ya, saya kan dari lapangan juga, awal dari lapangan. Jadi setiap karyawan terutama yang di traffic [28:10] Itu sebelum ke lapangan kita ada pelatihan dulu Pertama kita rekrutmen baru biasanya ada pelatihan Zaman dulu saya masuk masih ada namanya penataran P4 kan Mungkin sekarang masih sudah enggak ada Kita ada penataran P4 jadi kita dari segi mental dimasukin [28:37] Penataran P4, nah disitu nanti bisa digabung nih sekalian ada pelatihan operasional loh di lapangan nanti seperti apa SOB-nya seperti apa, jadi sebelum kita terjun ke lapangan kita dibekali ada pelatihannya, nanti secara periodik tiap tahun atau dua tahun sekali atau tiga tahun sekali kita ada pelatihan lagi [29:07] Jadi untuk refresh juga, untuk menambah teori juga, evaluasi pekerjaan yang sudah kita jalankan, jadi bisa digabung di situ. Berarti lebih ke praktek ya Pak? Langsung ya? Kita ke praktek. Oke. Oke. [29:33] Nah, sebenernya kan saya tuh mau ngebuat yang namanya sistem jadi bikin game gitu pak Nah, gamenya itu tuh tentang manajemen operasional jalan tol Jadi game ini merepresentasikan operasional gitu loh bisa dibilang game simulasi Nah, menurut bapak sendiri hal apa yang harus ada atau bisa dibilang merepresentasikan jalan tol gitu pak [30:02] yang harus ada gitu kalau jelatol harus ada ini, harus ada itu dan lainnya sekarang kan terutama di traffic ya kita kan punya SOP sendiri di traffic sendiri di gerbang-gerbang itu ada SOP nya sendiri kalau secara SOP di traffic itu karena kita pelayanan yang pertama yang pertama harus ada orang petugasnya harus ada kita gak bisa mengandalkan namanya [30:32] di jalur kita nggak bisa mengandalkan robot nggak ada robot orang kecelakaan robotnya nangani nggak bisa harus ada orang kedua harus ada sarana pendukung kendaraan karena kan kita mobil mobil harus menyelusuri per kilometer jadi kita menyiapkan jalur itu memang harus bersih

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

[31:01] dari gangguan dan dari hal-hal yang bisa membahayakan penggunaan jalan. Itu kan harus pakai mobil muternya kan. Yang kedua, dari keselamatannya. Di saat penggunaan jalan melewati jalan tol tidak menutup penggunaan, ada kecelakaan. Atau orang yang terjatuh, atau orang yang ketabrak. Di situ ada unit penyelamat. [31:30] penyelamatan, keselamatan, kita harus ada ambulan. Bisa di segi hukum, di saat ada kecelakaan, pasti di situ ada unsur menuntut dan ditutup. Kecelakaan yang ada kontranya pasti satu menuntut, satu ditutup. Di situ harus ada [31:56] piranti alat penegak hukumnya, siapa PCR. Nah terus untuk perawatan, perawatan juga kita harus menyediakan alat-alat kendaraan terutama ya untuk melakukan perawatan kerusakan yang ada di Jalan Tol. Jadi secara SPM semua sudah diatur di situ. [32:25] Misalkan kendaraan mogok harus dibantu sama derek SPM-nya derek dalam suatu area, suatu ruas itu minimal 15 kilo itu ada satu derek per 15 kilo satu derek secara SPM ada derek, ada ambulan [32:55] ada patroli desa marganya, ada PCR-nya, ada kendaraan reskyunya. Jadi, itu syarat pelayanan ini Jalang Tol. Harus ada. Harus ada itu ya Pak? Harus ada SPM. [33:17] Kalau menurut Bapak, seberapa berpengaruh jalan tol Jagorawi bagi mobilitas masyarakat, Pak? Khususnya di Metropolitan ini, Pak? Nah, kota Bogor, ini kan Jagorawi ada di Bogor ya, dan sekitarnya lah. Ini kan penyangga dari DKI Jakarta. Hampir, mungkin hampir 60% itu penduduk [33:46] yang kerja di Jakarta itu dari Bugur kalau enggak ada jalan tol bisa bayangkan mereka lewat Transyogi ke Jakarta macetnya udah seperti apa ada tol aja masih macet loh Transyogi ini makanya dengan arjaya Jagorawi itu pengguna jalannya juga merasa sangat-sangat diuntungkan sangat terbantu [34:16] Perjalanan bisa dicobalah dari Bogor lewat jalan luar sama lewat tol. Bedanya berapa jam. Bisa dipersingkatlah Pak E. Iya. [34:44] Menurut Bapak, kendala apa yang sering muncul dalam proses pengambilan keputusan di lapangan? Memang kita ini kan pelayanan 24 jam.

Lampiran 3. Dokumentasi Wawancara

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 4. CV Ahli Game

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Contact
almukhlishiddin@gmail.com
www.linkedin.com/in/muhammad-al-mukhlishiddin-0632b941 (LinkedIn)

Top Skills
Guest Lecturing
3D Modeling
Game Design

Muhammad Al Mukhlishiddin

Program Manager at Agate International
West Java, Indonesia

Experience

Agate International
6 years 9 months

Program Manager
January 2020 - Present (6 years 5 months)
Bandung

Job Desk

1) managing game development training program with various scope and duration

a. small scope program: Agate Game Course regular program, Agate Game Course for Teachers

b. medium scope program: DTS VSGA Game Creation, BPPMPV KPTK Upskilling & Reskilling (Artistik Gim & Metamesta), DILo Game Academy, Indigo Game Academy

c. large scope program: Studi Independen Pengembangan Game Agate - Kampus Merdeka (Batch 1, 5, 6, 7)

2) organizing offline and online event with various scope and duration

3) managing academic partnership with various stakeholder

a. securing academic partnership deal (MoU & training program)

4) developing curriculum for training program

a. developing basic game programming, basic 3D game art, basic game design for various programs

5) teaching various basic practical aspect of game development

a. teach basic game design in Agate Game Course - Game Development 101 course and various one-off session

b. teach basic 3D game art to vocational school teachers in BPPMPV KPTK upskilling & reskilling program

6) creating course materials for various aspect of game development

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

7) supervising course materials creation

8) designing game & managing project for Game Development Profesional Program

Management Trainee
September 2019 - December 2019 (4 months)

Politeknik Negeri Lhokseumawe
Guest Lecturer
September 2024 - December 2024 (4 months)
Teach various game development topics in Project-based Learning Class at Multimedia Technology Engineering Program
Oversee project management for 3 group projects

Politeknik Negeri Jakarta
Guest Lecturer
September 2024 - December 2024 (4 months)
Teach various game development topics in Project-based Learning Class at Digital Multimedia Engineering Program
Oversee project management for 3 group projects

Asosiasi Game Indonesia
2 years 3 months

Drafting Committee Member
April 2023 - October 2024 (1 year 7 months)
Drafting various formal documents regarding game development occupations and qualification with drafting committee and facilitation from Kemenkominfo, Kemenparekraf, Kemenaker:
SKKNI (Standar Kompetensi Kerja Nasional Indonesia) Game Development
KKNI (Kerangka Kualifikasi Nasional Indonesia) Game Development
Peta Okupasi Game Development
Skema Okupasi Game Development

Talent Development Associate
August 2022 - August 2023 (1 year 1 month)

Universitas Katolik Parahyangan (UNPAR)
Guest Lecturer
September 2023 - July 2024 (11 months)

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Bandung, West Java, Indonesia

Teach Bussiness Administration students about basic game design in Enrichment Class full semester (collaboration program between Agate & Unpar)

Freelance
Freelance Translator
2017 - January 2020 (3 years)

Past projects:
Translation:
-Orang Gagal - Osamu Dazai (Penerbit Basabasi, 2020)
-Paris yang Tak Berkesudahan - Ernest Hemingway (Penerbit Basabasi, 2020)
-Delapan Pemandangan dari Tokyo - Osamu Dazai (Penerbit Trubadur, 2018)
-Finks: Bagaimana CIA Mengelabui Para Sastrawan Besar Dunia - Joel Whitney (Jungkir Balik Pustaka, 2018)
-Nick Drake: Sebuah Biografi - Patrick Humphries (Jungkir Balik Pustaka, 2017)
-various scientific journals

Writing:
Buku Pendukung Pembelajaran BIPA: Memahami Indonesia Melalui Sastra - Co-author with M. Adji (Badan Pembinaan dan Pengembangan Bahasa, Kementerian Pendidikan dan Kebudayaan, 2018)

Gimkavenesia
Content Creator
2019 - 2020 (1 year)

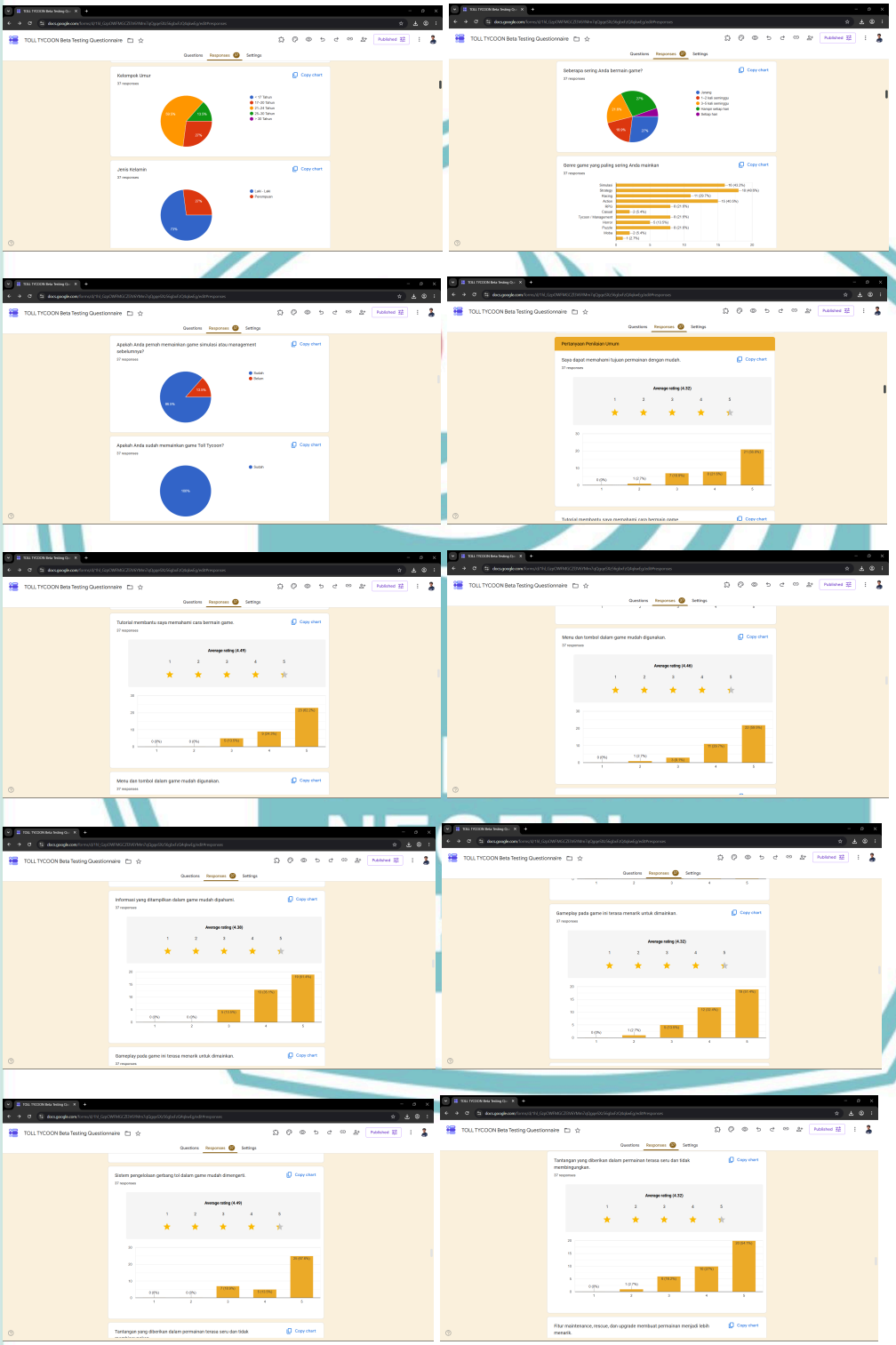
Gimkavenesia is an Instagram account that publish various information about Indonesian game development industry.
https://www.instagram.com/gimkavenesia/?hl=id

Education

Universitas Padjadjaran
Sarjana Humaniora, Sastra Indonesia · (2012 - 2017)

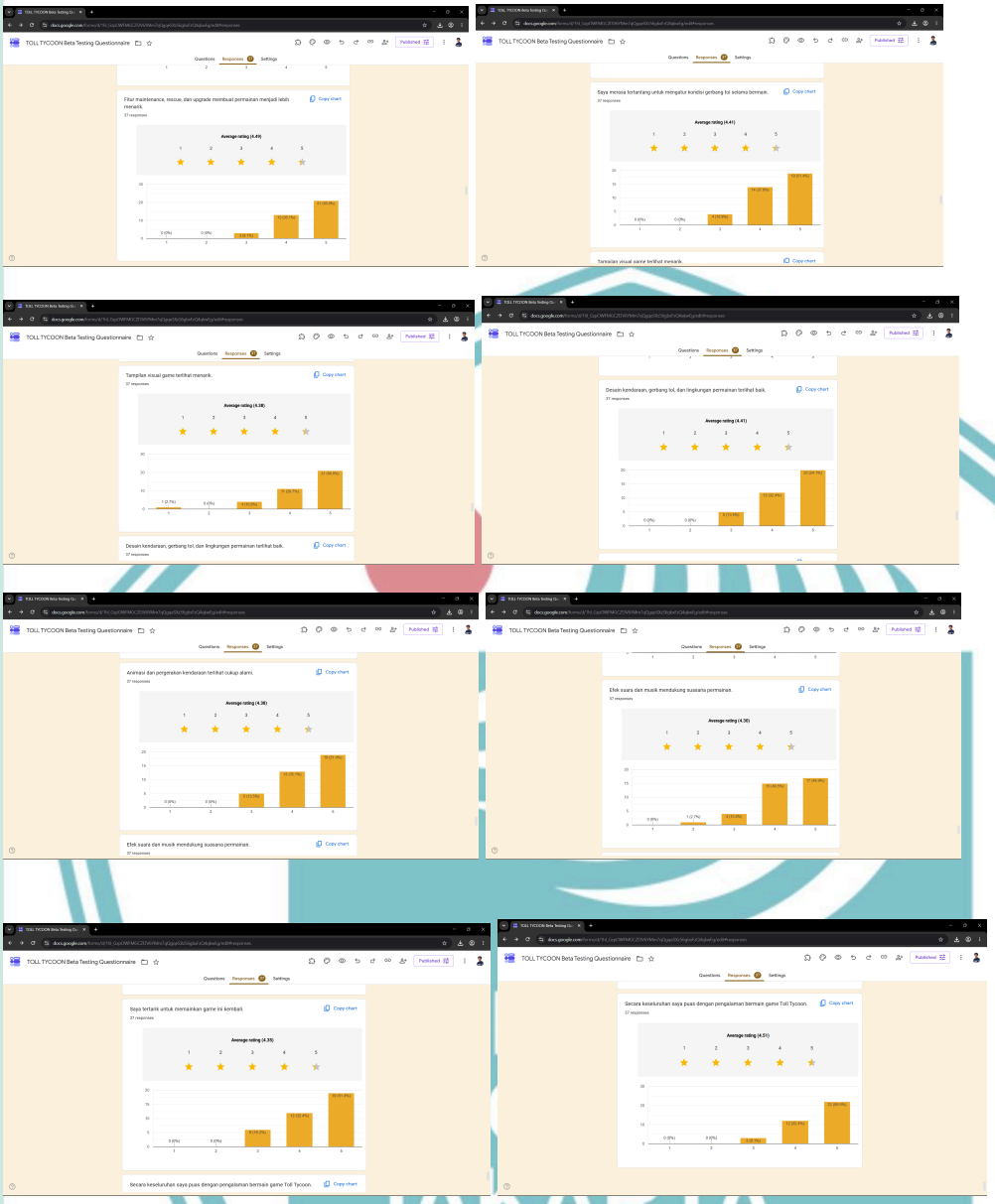
Page 3 of 3

Lampiran 5. Hasil Responden Pengguna Umum



© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

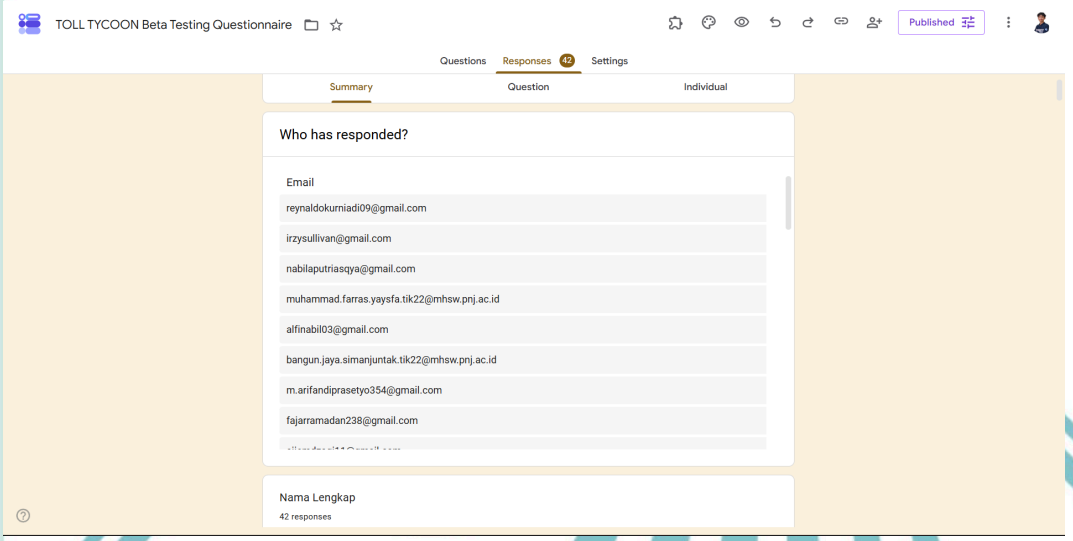
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

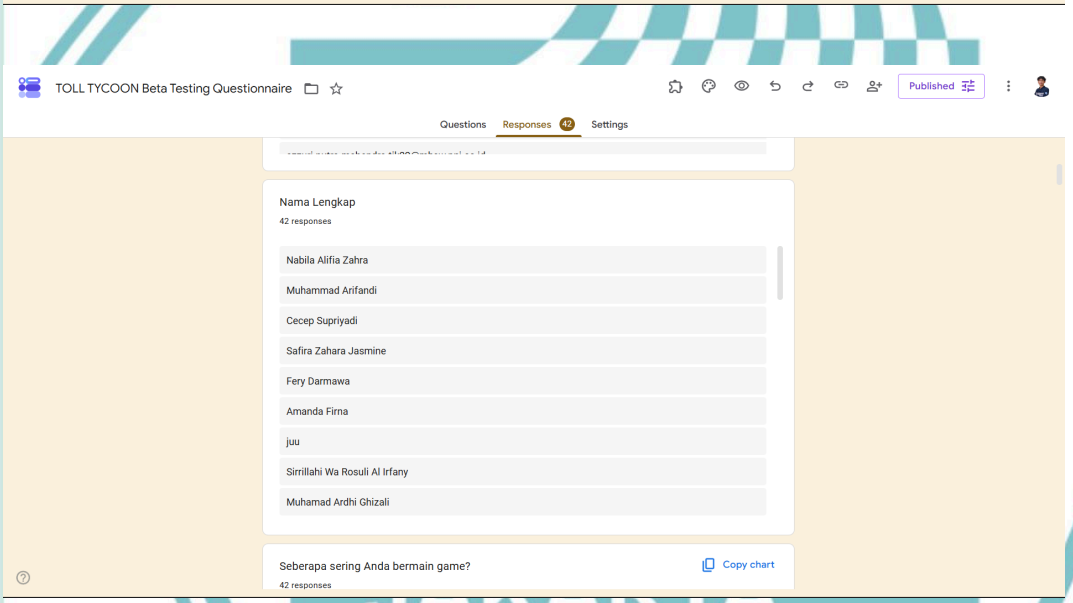


© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 6. Hasil Beta Testing Expert Game





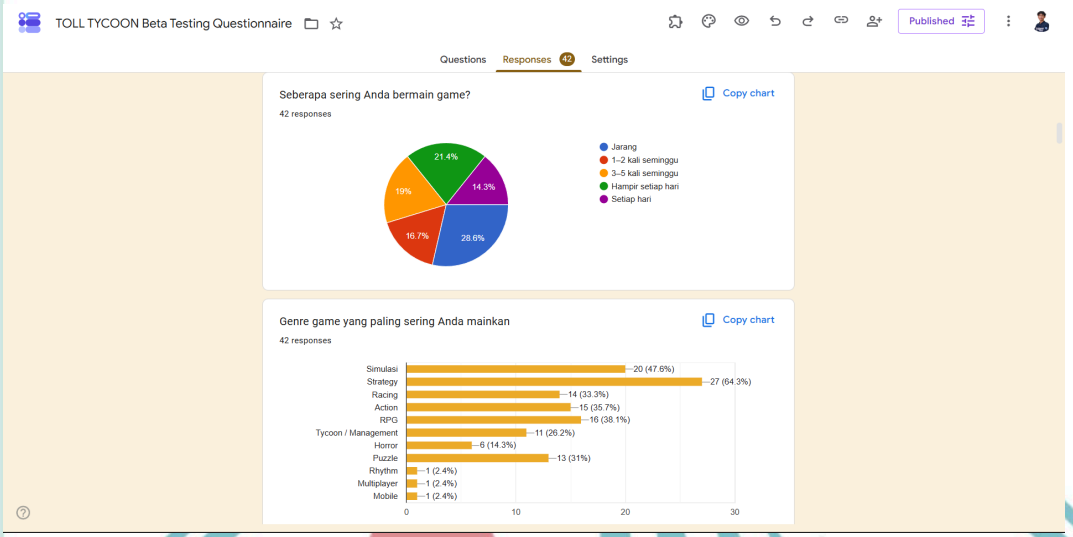
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

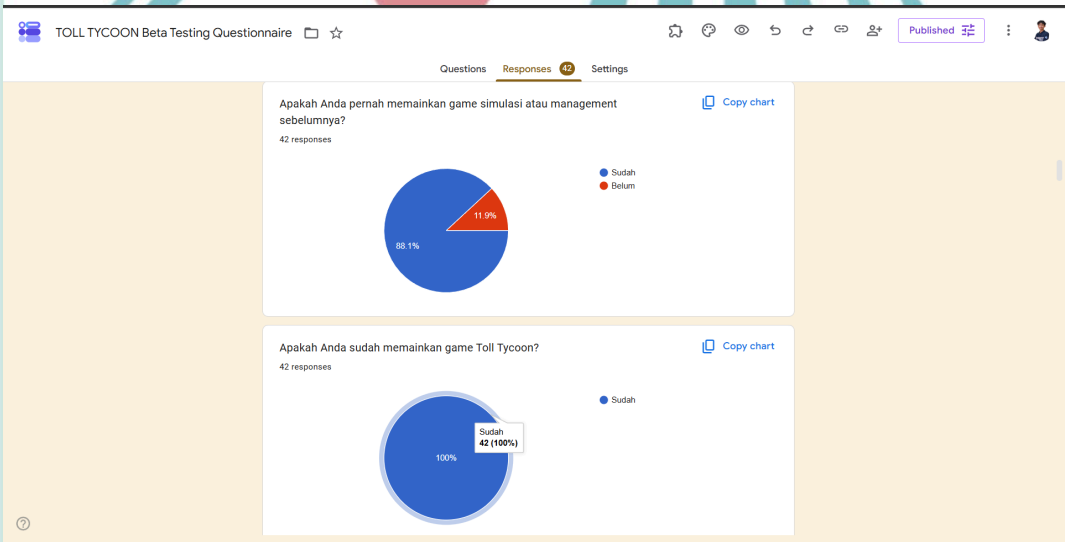
Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

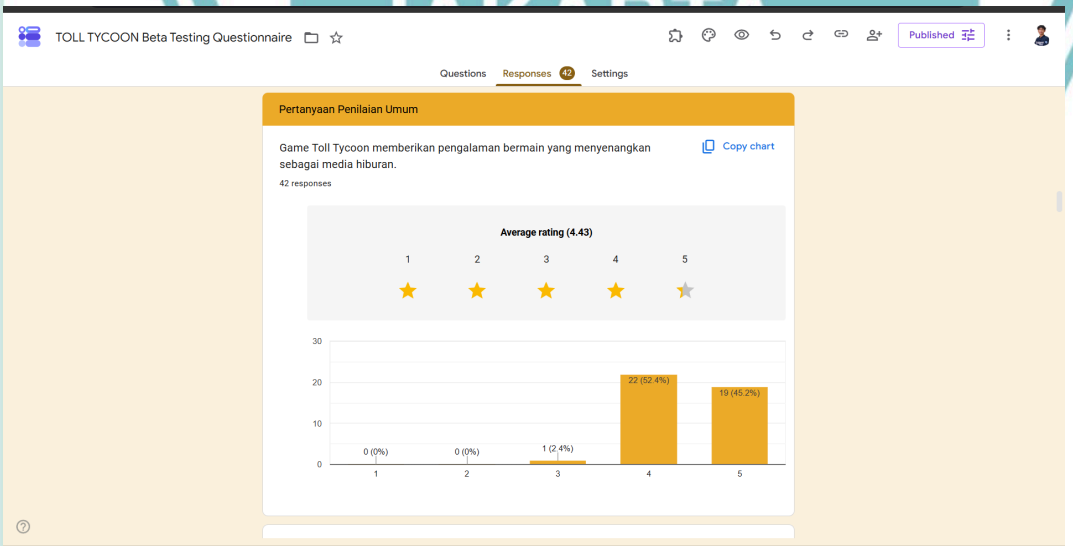
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



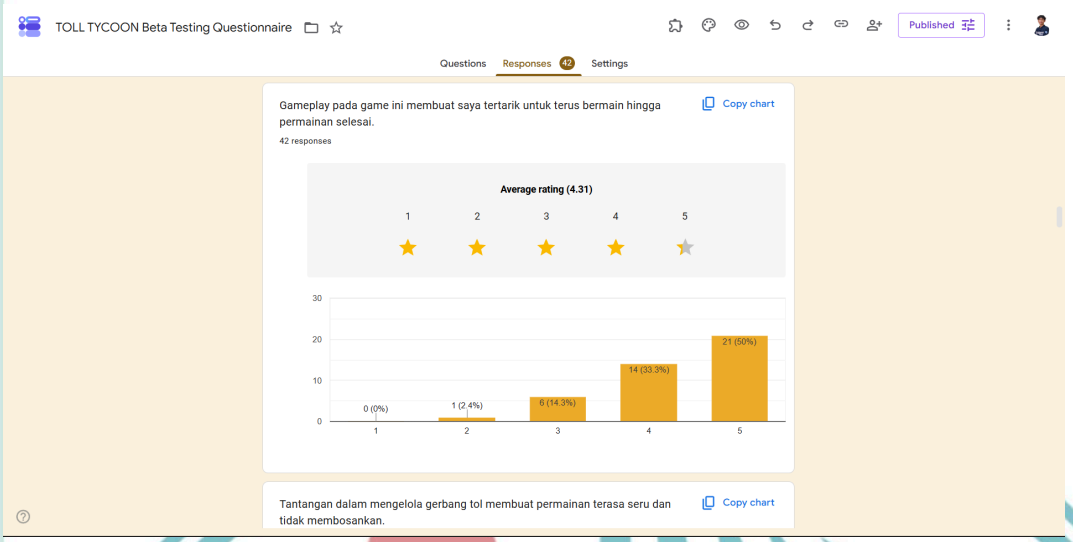


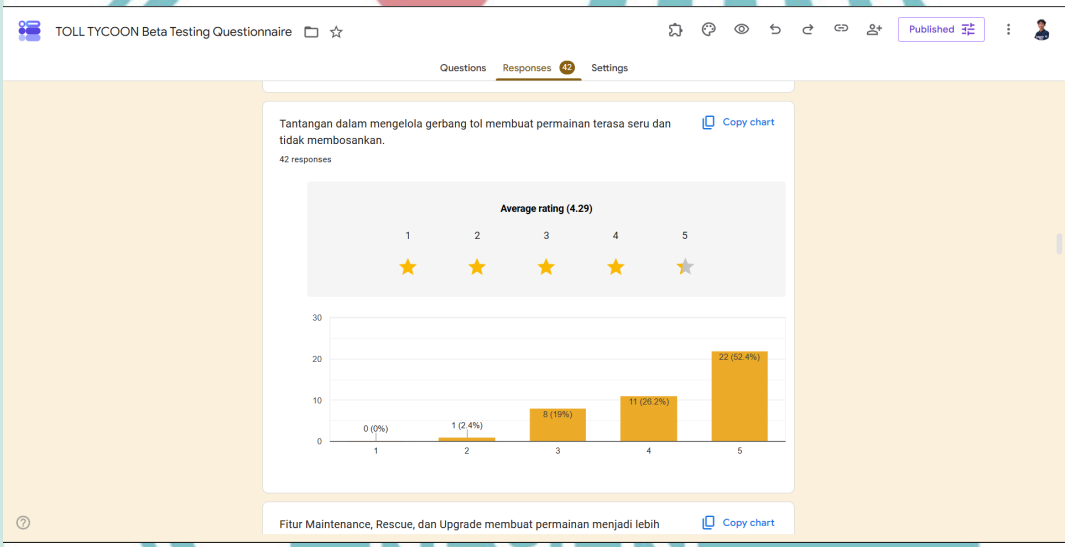


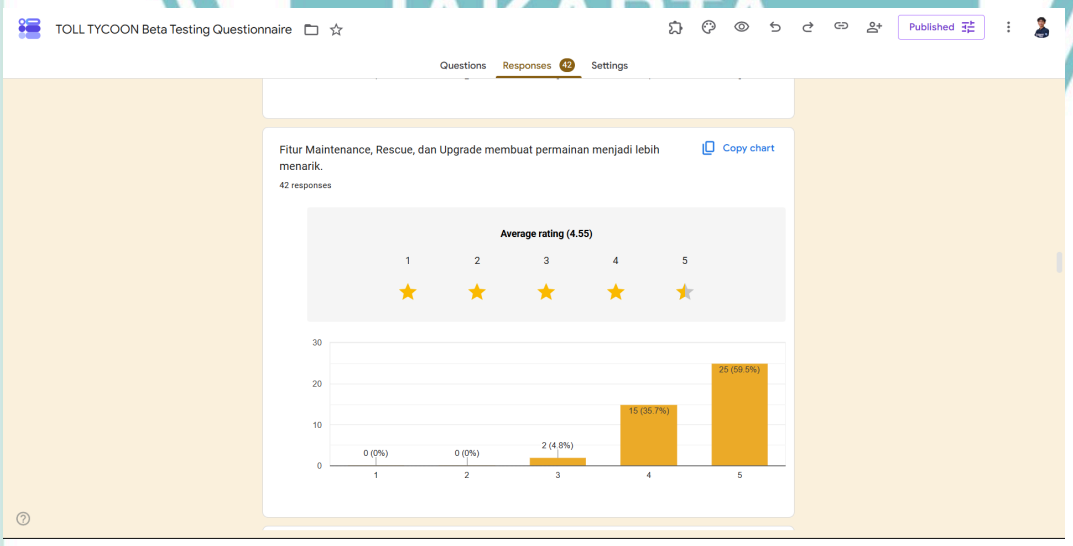
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

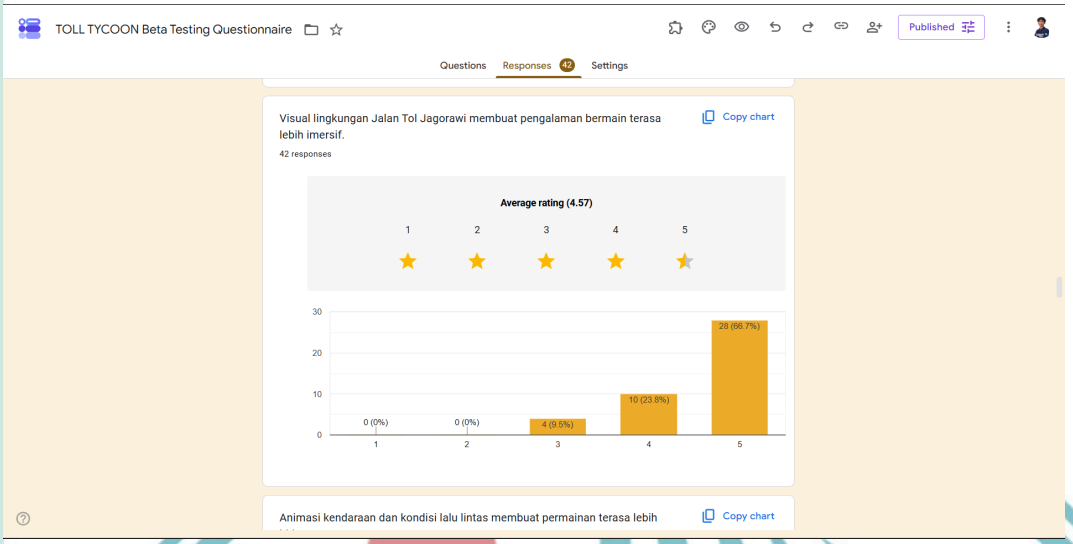
Hak Cipta :

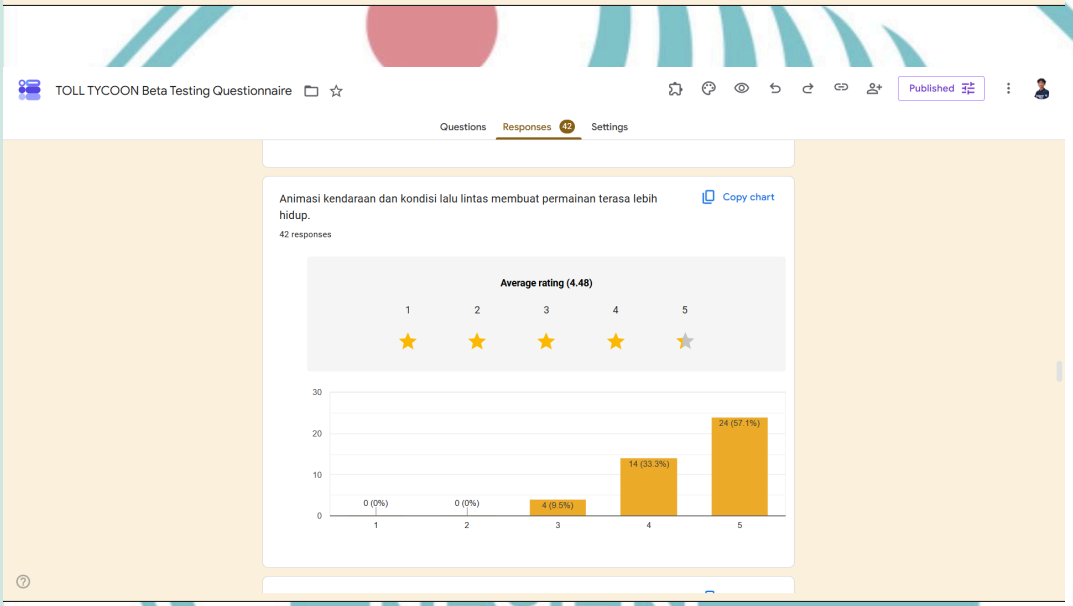
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

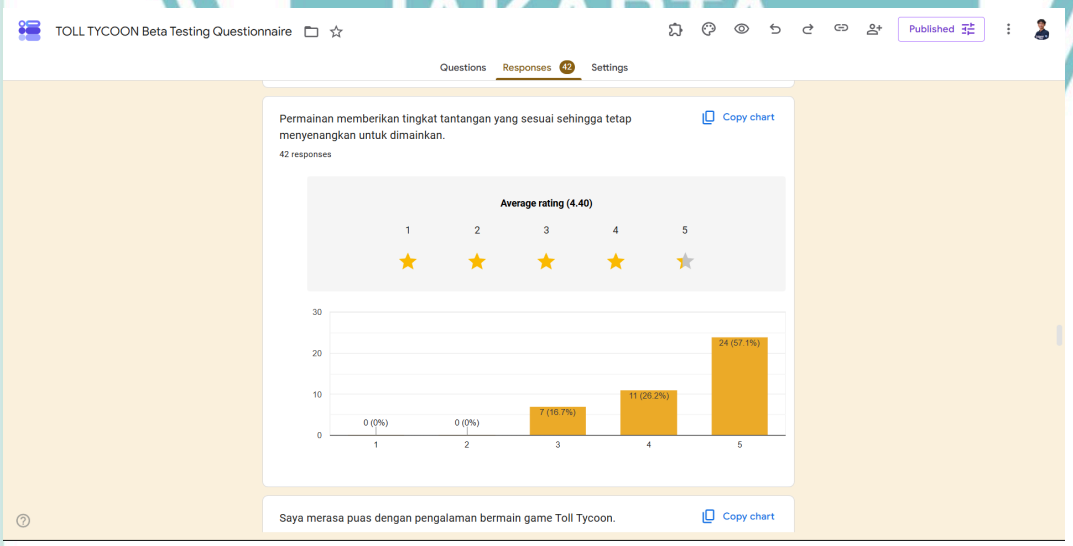












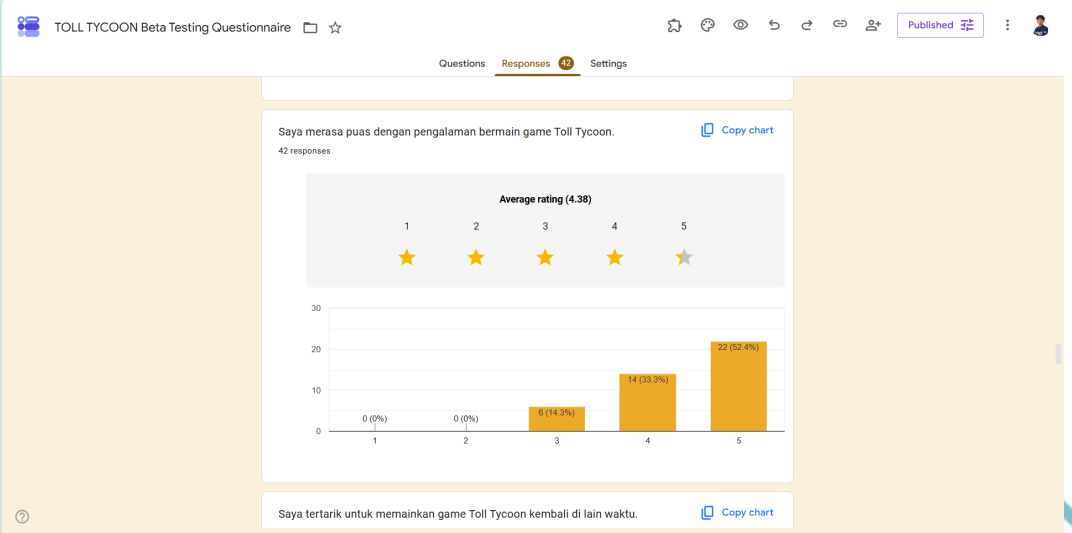
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

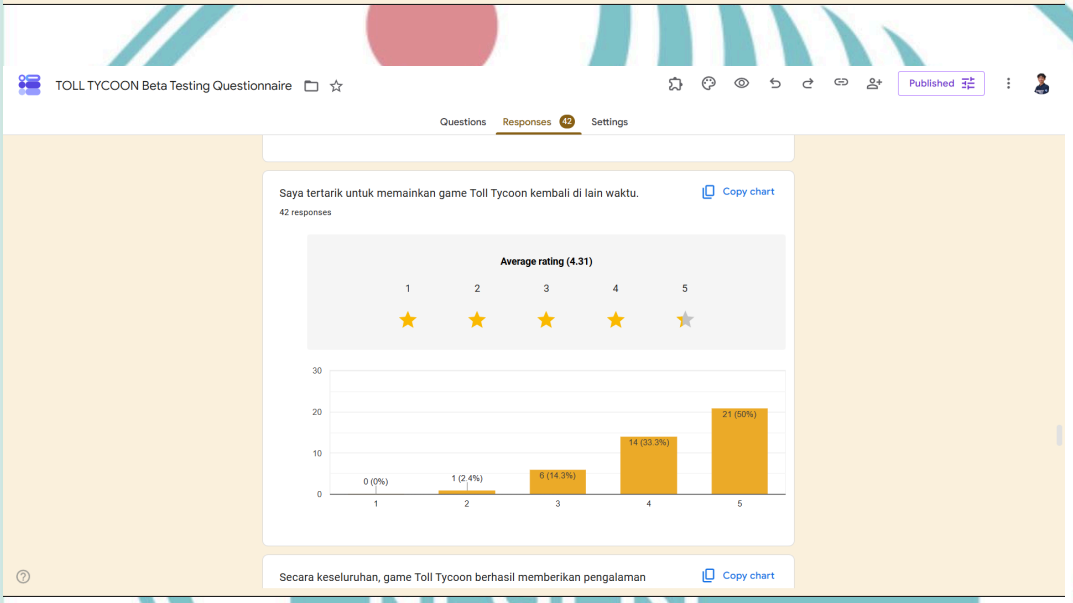
Hak Cipta :

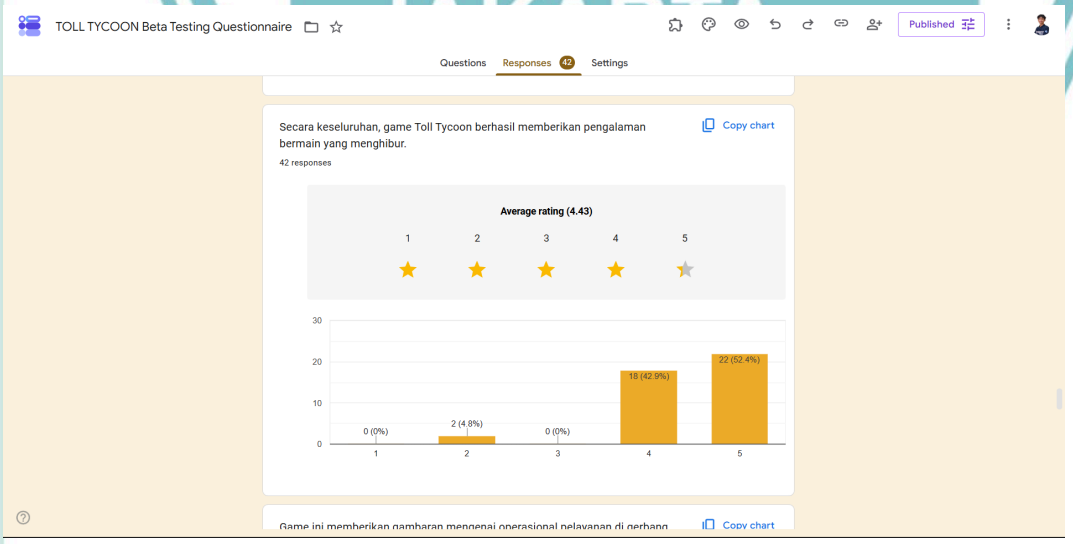
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta



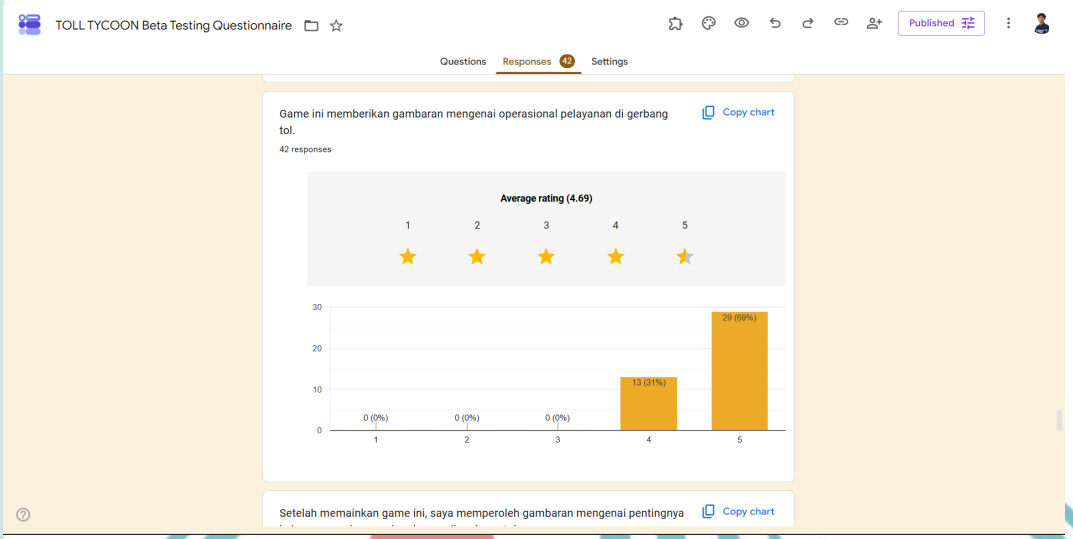


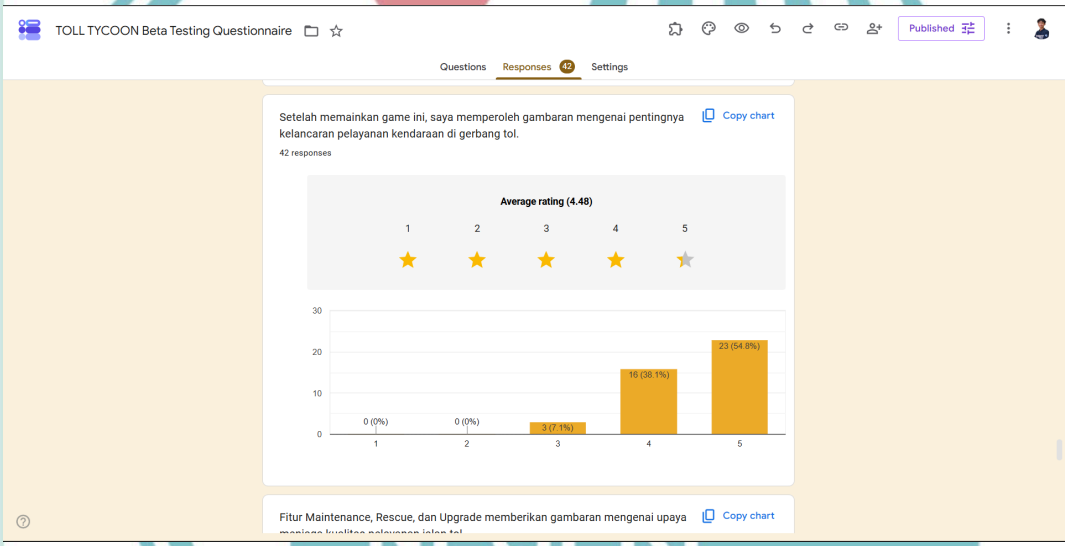


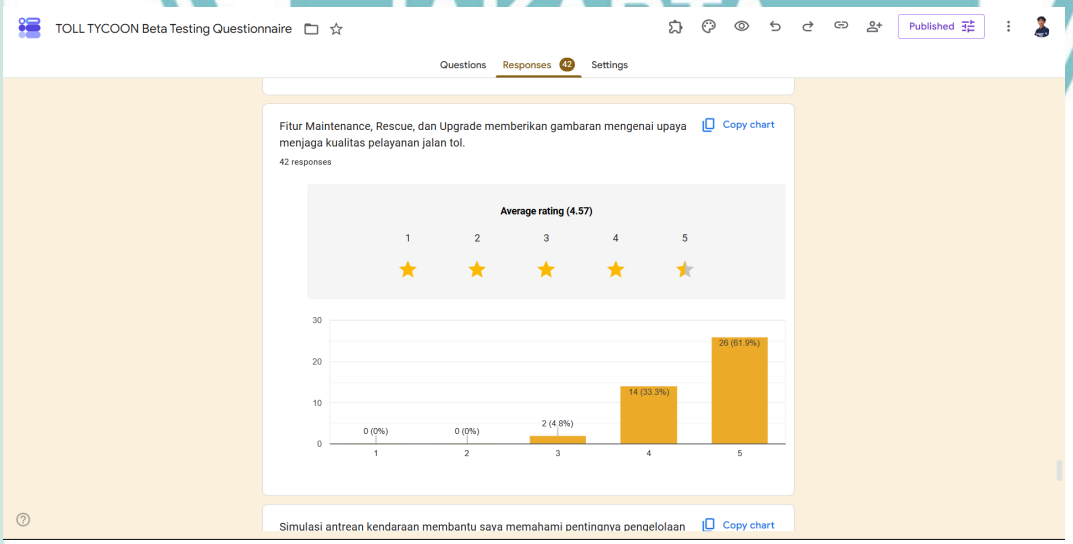
© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

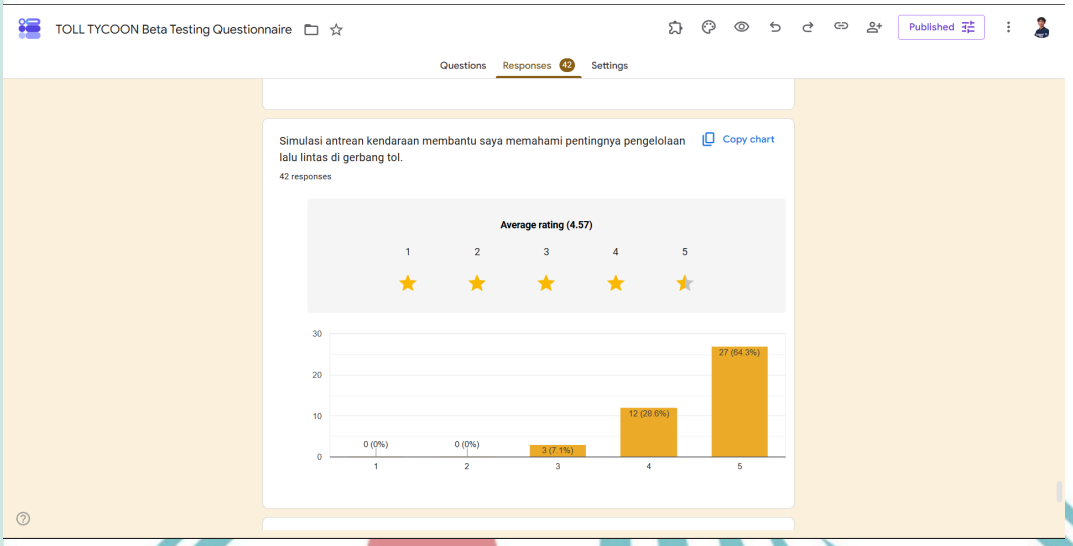
Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta









© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
   a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
   b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

© Hak Cipta milik Jurusan TIK Politeknik Negeri Jakarta

Hak Cipta :
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber :
a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penulisan laporan, penulisan kritik atau tinjauan suatu masalah.
b. Pengutipan tidak merugikan kepentingan yang wajar Politeknik Negeri Jakarta
2. Dilarang mengumukan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin dari Jurusan TIK Politeknik Negeri Jakarta

Lampiran 7. Daftar Riwayat Hidup Penulis

Angga Yusma Prasetia lahir di Jakarta pada tanggal 26 September 2003. Menempuh pendidikan menengah kejuruan di SMKN 58 Jakarta dengan jurusan Desain Komunikasi Visual. Setelah menyelesaikan pendidikan menengah, melanjutkan pendidikan di Politeknik Negeri Jakarta pada Program Studi Teknik Multimedia Digital. Selama menjalani masa perkuliahan, aktif dalam kegiatan organisasi kemahasiswaan, khususnya di Himpunan Mahasiswa Teknik Informatika dan Komputer (HIMATIK) selama tiga tahun berturut-turut. Pada tahun pertama menjabat sebagai Staff Departemen Pendidikan dan Teknologi. Kemudian pada tahun kedua dipercaya menjadi Kepala Departemen Pendidikan dan Teknologi. Selanjutnya, pada tahun terakhir kepengurusan menjabat sebagai Badan Pengurus Harian Satuan Pengamatan Internal Departemen Penelitian dan Pengembangan. Selain aktif dalam organisasi, juga pernah menjadi Ketua Divisi Komisi Disiplin pada acara EXPECTIK serta mengikuti kegiatan pengabdian masyarakat bersama dosen sebagai bentuk kontribusi dalam kegiatan akademik dan sosial. Pengalaman organisasi, kepanitiaan, dan kegiatan pengabdian masyarakat tersebut memberikan banyak pembelajaran dalam hal kepemimpinan, tanggung jawab, komunikasi, serta kerja sama tim.