

**SISTEM REKOMENDASI MOBIL PADA GAME MENGGUNAKAN METODE  
*CONTENT-BASED FILTERING***

**SKRIPSI**

**Oleh :**  
**IHWAN TAMAMI**  
**NIM. 210605110047**



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

**SISTEM REKOMENDASI MOBIL PADA *GAME* MENGGUNAKAN  
METODE *CONTENT-BASED FILTERING***

**SKRIPSI**

Diajukan kepada:  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang  
Untuk memenuhi Salah Satu Persyaratan dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :  
**IHWAN TAMAMI**  
**NIM. 210605110047**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2025**

## HALAMAN PERSETUJUAN

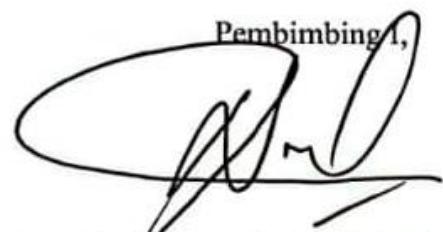
### SISTEM REKOMENDASI MOBIL PADA GAME MENGGUNAKAN METODE **CONTENT-BASED FILTERING**

#### SKRIPSI

Oleh :  
IHWAN TAMAMI  
NIM. 210605110047

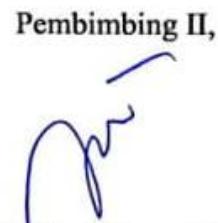
Telah Diperiksa dan Disetujui untuk Diuji:  
Tanggal: 03 Desember 2025

Pembimbing I,



Dr. Ir. Fresy Nugroho, M.T., IPM, ASEAN Eng  
NIP. 19710722 201101 1 001

Pembimbing II,



Ahmad Pahmi Karami, M.Kom  
NIP. 19870909202012 1 001

Mengetahui,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Supriyono, M.Kom  
NIP. 19841010 201903 1 012

## HALAMAN PENGESAHAN

### SISTEM REKOMENDASI MOBIL PADA GAME MENGGUNAKAN METODE *CONTENT-BASED FILTERING*

#### SKRIPSI

Oleh :  
IHWAN TAMAMI  
NIM. 210605110047

Diajukan kepada:  
Telah Dipertahankan di Depan Dewan Pengaji Skripsi dan  
Dinyatakan Diterima Sebagai Salah Satu Persyaratan  
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Tanggal: 03 Desember 2025

#### Susunan Dewan Pengaji

Ketua Pengaji : Dr. Ir. Yunifa Miftachul Arif, S.ST., M.T  
NIP. 19830616 201101 1 004

Anggota Pengaji I : Hani Nurhayati, M.T  
NIP. 19780625 200801 2 006

Anggota Pengaji II : Dr. Ir. Fresy Nugroho, M.T., IPM, ASEAN Eng  
NIP. 19710722 201101 1 001

Anggota Pengaji III : Ahmad Fahmi Karami, M.Kom  
NIP. 19870909202012 1 001

Mengetahui dan Mengesahkan,  
Ketua Program Studi Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Supriyono, M.Kom  
NIP. 19841010 201903 1 012

## **PERNYATAAN KEASLIAN TULISAN**

Saya yang bertanda tangan di bawah ini:

Nama : Ihwan Tamami  
NIM : 210605110047  
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika  
Judul Skripsi : Sistem Rekomendasi Mobil Pada Game Menggunakan Metode *Content-Based Filtering*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 03 Desember 2025  
Yang membuat pernyataan,



Ihwan Tamami  
NIM. 210605110047

## **MOTTO**

*“Khoirun nas anfa’uhum linnas”*

*"Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia lainnya".*

## **HALAMAN PERSEMBAHAN**

Segala puji bagi Allah SWT yang telah melimpahkan rahmat, hidayah, dan kemudahan-Nya, sehingga saya dapat menyelesaikan skripsi ini dengan baik. Terima kasih yang tak terhingga atas segala nikmat dan karunia-Nya yang selalu menyertai setiap langkah dan usaha saya.

Skripsi ini saya persembahkan untuk kedua orang tua saya yang sangat saya cintai, Ayahanda Imron Mahmudi dan Ibunda Suparti, S.Pd., yang menjadi sumber kekuatan dan inspirasi dalam hidup saya. Dengan penuh cinta, kesabaran, dan doa yang tak pernah putus, mereka selalu mendukung dan mendampingi saya dalam setiap perjalanan. Segala pencapaian yang saya raih adalah hasil dari pengorbanan dan kasih sayang mereka yang tiada henti.

Ucapan terima kasih yang tulus saya sampaikan kepada Bapak Dr. Ir. Fresy Nugroho, M.T., IPM, ASEAN Eng dan Bapak Ahmad Fahmi Karami, M.Kom, selaku dosen pembimbing yang dengan sabar telah memberikan bimbingan, arahan, serta ilmu yang sangat berharga selama proses penyusunan skripsi ini. Terima kasih juga kepada Dr. Ir. Yunifa Miftachul Arif, S.ST., M.T dan Hani Nurhayati, M.T selaku dosen penguji, yang telah memberikan kritik dan saran yang membangun dalam penyempurnaan karya ini.

Saya juga mengucapkan terima kasih kepada sahabat saya, Nazzil Akfa Said Fiddaraini, yang telah menjadi rekan seperjuangan dalam pengembangan project *game Perfect Car*. Terima kasih atas kerja sama, dukungan, dan semangat yang diberikan selama proses penelitian ini berlangsung.

Rasa terima kasih yang mendalam juga saya sampaikan kepada keluarga besar Lembaga Tinggi Pesantren Luhur Malang, yang telah menjadi tempat bernaung penuh doa, ketenangan, dan kebersamaan selama masa perkuliahan. Ucapan terima kasih juga untuk sahabat-sahabat terbaik saya, teman-teman Discord, dan rekan seperjuangan Teknik Informatika UIN Malang, yang selalu memberikan semangat, ide, serta dukungan moral di setiap langkah perjuangan.

Dan terakhir, saya persembahkan karya ini untuk diri saya sendiri, Ihwan Tamami, yang telah berjuang dan bertahan melalui setiap tantangan, meskipun sering merasa lelah dan ingin menyerah. Terima kasih telah memilih untuk terus melangkah, berusaha, dan menyelesaikan perjalanan ini dengan sebaik mungkin. Semoga segala proses ini menjadi langkah menuju keberkahan dan kesuksesan yang diridhai Allah SWT.

*Amin ya Rabbal 'Alamin.*

## KATA PENGANTAR

*Assalamu'alaikum Warahmatullahi Wabarakatuh.*

Segala puji dan syukur penulis haturkan kehadirat Allah SWT atas berkah, rahmat, dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul "Sistem Rekomendasi Mobil Pada *Game* Menggunakan Metode *Content-Based Filtering*" sebagai salah satu persyaratan untuk menyelesaikan Sarjana Komputer (S.Kom) pada Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Shalawat serta salam semoga senantiasa tercurahkan kepada Nabi Muhammad SAW yang telah mengantarkan umat manusia dari zaman kegelapan ke zaman yang terang benderang ini.

Penulis menyadari banyak pihak yang terlibat dalam pemberian dukungan dan bantuan selama menyelesaikan studi dan penulisan tugas akhir ini, baik moril maupun materil. Oleh karena itu, pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Prof. Dr. Hj. Ilfi Nur Diana, M.Si., selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Dr. Agus Mulyono, M.Kes., selaku Dekan Fakultas Sains dan Teknologi UIN Maulana Malik Ibrahim Malang.
3. Supriyono, M. Kom, selaku Ketua Program Studi Teknik Informatika UIN Maulana Malik Ibrahim Malang.

4. Dr. Ir. Fresy Nugroho, M.T., IPM, ASEAN Eng, selaku Dosen Pembimbing I yang telah memberi bimbingan, arahan, dan semangat dalam proses penulisan skripsi.
5. Ahmad Fahmi Karami, M.Kom, selaku Dosen Pembimbing II yang telah memberikan bimbingan, motivasi, serta bantuan yang berarti dalam proses penulisan skripsi.
6. Dr. Ir. Yunifa Miftachul Arif, S.ST., M.T selaku Dosen Ketua Penguji dan Hani Nurhayati, M.T, selaku Dosen Anggota Penguji I yang telah menguji serta memberikan masukan sehingga penulis dapat menghasilkan skripsi yang baik.
7. Nia Faricha S, Si., selaku admin Program Studi Teknik Informatika yang selalu sabar memberikan informasi, membantu, dan memberikan arahan selama perkuliahan dan proses penulisan skripsi ini.
8. Seluruh Dosen, Laboran dan Civitas Akademik Program Studi Teknik Informatika yang telah memberikan ilmu pengetahuan, dan dukungan selama studi.
9. Kedua orang tua penulis, Ayah Imron Mahmudi dan Bunda Suparti S.Pd, Kakak-Adik dan keluarga besar saya yang selalu memberikan semangat dan doa yang terus dipanjatkan sehingga menjadikan proses penyelesaian skripsi menjadi lebih lancar.
10. Sahabat saya, Nazzil Akfa Said Fiddaraini, yang telah menjadi rekan berjuang dalam pengembangan project *game Perfect Car* dan selalu membantu dan mendukung dalam proses penelitian ini.

11. Sahabat-sahabat terbaik, teman-teman Discord, dan seluruh rekan seperjuangan di Teknik Informatika UIN Malang, atas semangat, kebersamaan, dan dukungan selama proses penggerjaan skripsi ini.
12. Keluarga besar Lembaga Tinggi Pesantren Luhur Malang, atas doa, bimbingan spiritual, semangat, dan kebersamaan yang menenangkan di setiap proses perjuangan akademik.
13. Seluruh pihak yang telah terlibat secara langsung maupun tidak langsung yang tidak bisa saya sebutkan satu persatu dalam penyelesaian skripsi ini.
14. And *lastly*, kepada diri saya sendiri, Ihwan Tamami. Terima kasih atas keteguhan untuk terus berjuang sampai saat ini. Terima kasih sudah bertahan dan berusaha, meskipun sering merasa lelah merasa putus asa dengan apa yang sudah diusahakan, namun tetap menjadi manusia yang mau berusaha dan bangkit dari keterpurukan. Terimakasih untuk tidak menyerah dalam proses penyusunan skripsi ini dan berusaha menyelesaikannya semaksimal mungkin.

Penulis menyadari dalam penulisan skripsi ini tidak luput dari kesalahan yang jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun sehingga skripsi ini dapat lebih dikembangkan.

*Wassalamu 'alaikum, Wr. Wb*

Malang, 25 November 2025

Penulis

## DAFTAR ISI

<b>HALAMAN PERSETUJUAN.....</b>	<b>iii</b>
<b>HALAMAN PENGESAHAN.....</b>	<b>iv</b>
<b>PERNYATAAN KEASLIAN TULISAN .....</b>	<b>v</b>
<b>MOTTO.....</b>	<b>vi</b>
<b>HALAMAN PERSEMBAHAN .....</b>	<b>vii</b>
<b>KATA PENGANTAR.....</b>	<b>ix</b>
<b>DAFTAR ISI .....</b>	<b>xii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiv</b>
<b>DAFTAR TABEL.....</b>	<b>xv</b>
<b>ABSTRAK.....</b>	<b>xvi</b>
<b>ABSTRACT .....</b>	<b>xvii</b>
<b>مستخلص البحث.....</b>	<b>xviii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>19</b>
1.1 Latar Belakang.....	19
1.2 Rumusan Masalah.....	23
1.3 Batasan Masalah .....	23
1.4 Tujuan Penelitian .....	24
1.5 Manfaat Penelitian .....	24
<b>BAB II STUDI PUSTAKA .....</b>	<b>25</b>
2.1 Penelitian Terkait .....	25
2.2 <i>Game</i> .....	32
2.3 Sistem Rekomendasi .....	34
2.4 <i>Content-Based Filtering (CBF)</i> .....	36
2.4.1 Tahapan <i>Content-Based Filtering</i> .....	37
2.5 <i>Cosine Similarity</i> .....	38
2.6 <i>System Usability Scale (SUS)</i> .....	40
<b>BAB III DESAIN DAN IMPLEMENTASI.....</b>	<b>41</b>
3.1 Analisis dan Perancangan <i>Game</i> .....	41
3.1.1 Analisis <i>Game</i> .....	41
3.1.2 Perancangan <i>Game</i> .....	42
3.1.3 Rancangan Antarmuka .....	44
3.2 Desain Sistem .....	47
3.3 Rancangan Perhitungan <i>Content-Based Filtering</i> .....	50
3.3.1 Item .....	51
3.3.2 Fitur .....	52
3.3.3 <i>Filtering</i> Kendaraan per Mode.....	56
3.3.4 Matriks Fitur Kendaraan ( <i>Item-Fitur Matrix</i> ) .....	58
3.3.5 Normalisasi Matriks Fitur Kendaraan ( <i>Item-Fitur Matrix</i> ).....	59
3.3.6 Perhitungan <i>Similarity</i> (Kemiripan) .....	61
3.4 Desain Pengujian Sistem.....	65
<b>BAB IV HASIL DAN PEMBAHASAN.....</b>	<b>68</b>
4.1 Implementasi Sistem.....	68
4.2 Implementasi Antarmuka <i>Game</i> .....	69

4.3 Implementasi Sistem Rekomendasi .....	73
4.2.1 Tahapan Metode Content-Based <i>Filtering</i> .....	75
4.2.2 Hasil Implementasi Sistem Rekomendasi .....	81
4.2.2.1 Hasil Implementasi pada Mode Balap .....	82
4.2.2.2 Hasil Implementasi pada Mode <i>Offroad</i> .....	84
4.2.2.3 Hasil Implementasi pada Mode Penumpang.....	86
4.2.2.4 Perbandingan Hasil Rekomendasi Tiap Mode .....	89
4.3 Pengujian Sistem .....	91
4.3.1 Pengujian Kesesuaian Rekomendasi.....	92
4.3.2 Pengujian <i>System Usability Scale</i> (SUS) .....	96
4.3.3 Analisa Aspek <i>Usability</i> .....	102
4.4 Integrasi Islam .....	106
4.4.1 <i>Mu'amalah Ma'a Allah</i> .....	107
4.4.2 <i>Mu'amalah Ma'a An-Nas</i> .....	109
4.4.3 <i>Mu'amalah Ma'a Al-Alam</i> .....	111
<b>BAB V KESIMPULAN DAN SARAN .....</b>	<b>115</b>
5.1 Kesimpulan.....	115
5.2 Saran.....	116
<b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	

## DAFTAR GAMBAR

Gambar 3.1 Menu Diagram.....	43
Gambar 3.2 Diagram <i>Finite State Machine</i> (FSM) Sistem Game .....	48
Gambar 3.3 Diagram <i>Finite State Machine</i> (FSM) Sistem Game .....	49
Gambar 4.1 Tampilan Awal <i>Game</i> .....	69
Gambar 4.2 Tampilan Main Menu .....	70
Gambar 4.3 Tampilan Pilih Mode .....	71
Gambar 4.4 Tampilan Pilih Mobil.....	72
Gambar 4.5 Tampilan <i>Gameplay</i> .....	73
Gambar 4.6 Tampilan Hasil Rekomendasi Mode Balap .....	83
Gambar 4.7 Tampilan Hasil Console log Nilai <i>Similarity</i> Mode Balap.....	83
Gambar 4.8 Tampilan Hasil Rekomendasi Mode <i>Offroad</i> .....	85
Gambar 4.9 Tampilan Console log Nilai <i>Similarity</i> Mode <i>Offroad</i> .....	85
Gambar 4.10 Tampilan Hasil Rekomendasi Mode Kurir .....	88
Gambar 4.11 Tampilan Console log Nilai <i>Similarity</i> Mode Penumpang .....	88
Gambar 4.12 Grafik Distribusi Tingkat Akurasi Rekomendasi .....	95
Gambar 4.13 Hasil Akhir Skor SUS.....	102

## DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu .....	29
Tabel 3.1 <i>Storyboard Game Perfect Car</i> .....	45
Tabel 3.2 Tabel Item .....	51
Tabel 3.3 Tabel Fitur.....	52
Tabel 3.4 Interval dan Kategori Performa Akselerasi Kendaraan .....	54
Tabel 3.5 Interval dan Kategori Performa Kecepatan Maksimal .....	54
Tabel 3.6 Interval dan Kategori Stabilitas <i>Handling</i> Kendaraan.....	54
Tabel 3.7 Interval dan Kategori Performa Suspensi Kendaraan .....	55
Tabel 3.8 Interval dan Kategori Kapasitas Penumpang Kendaraan .....	55
Tabel 3.9 Pemetaan Kendaraan .....	57
Tabel 3.10 Matriks Fitur Kendaraan.....	59
Tabel 3.11 Normalisasi Matriks Fitur Kendaraan .....	60
Tabel 3.12 Daftar Pernyataan Kuesioner <i>System Usability Scale</i> .....	66
Tabel 4.1 Hasil Perhitungan <i>Similarity</i> Mobil Lamborghini Mode Balap .....	82
Tabel 4.2 Hasil Perhitungan <i>Cosine Similarity</i> pada Mode Balap .....	84
Tabel 4.3 Hasil Perhitungan <i>Similarity</i> Mobil Ford Everest Mode <i>Offroad</i> .....	84
Tabel 4.4 Hasil Perhitungan <i>Cosine Similarity</i> pada Mode <i>Offroad</i> .....	86
Tabel 4.5 Hasil Perhitungan <i>Similarity</i> Mobil Mode Penumpang.....	87
Tabel 4.6 Hasil Perhitungan <i>Cosine Similarity</i> pada Mode Penumpang.....	88
Tabel 4.7 Perbandingan Hasil Rekomendasi Berdasarkan Mode Permainan .....	89
Tabel 4.8 Data Pengujian Kesesuaian Rekomendasi.....	93
Tabel 4.9 Rekapitulasi Tingkat Akurasi Rekomendasi.....	94
Tabel 4.10 Pernyataan <i>System Usability Scale</i> (SUS).....	96
Tabel 4.11 Nilai <i>System Usability Scale</i> (SUS).....	97
Tabel 4.12 Skor Asli Responden SUS .....	99
Tabel 4.13 Perhitungan Responden 1 .....	99
Tabel 4.14 Hasil Perhitungan Skor SUS .....	100

## ABSTRAK

Tamami, Ihwan. 2025. **Sistem Rekomendasi Mobil Pada Game Menggunakan Metode Content-Based Filtering.** Skripsi, Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Ir. Fresy Nugroho, ST., MT, IPM (II) Ahmad Fahmi Karami, M.Kom.

**Kata Kunci:** Sistem Rekomendasi, *Game* Simulasi, *Content-Based Filtering*, *Cosine Similarity*, *System Usability Scale* (SUS).

Industri *game* simulasi kendaraan menuntut pemain untuk memilih kendaraan yang tepat agar dapat menyelesaikan tantangan di berbagai kondisi lintasan. Permasalahan yang sering terjadi adalah pemain kesulitan menentukan mobil yang paling sesuai dengan gaya bermain dan kebutuhan mode permainan (Balap, *Offroad*, dan Penumpang) karena banyaknya variasi spesifikasi teknis kendaraan. Untuk mengatasi permasalahan tersebut, penelitian ini menerapkan metode *Content-Based Filtering* (CBF) dalam *game* "Perfect Car" guna memberikan rekomendasi mobil secara otomatis. Metode ini dipilih karena mampu menganalisis kemiripan spesifikasi kendaraan berdasarkan fitur akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas menggunakan algoritma *Cosine Similarity*. Sistem akan merekomendasikan mobil yang memiliki karakteristik paling mirip dengan kendaraan yang digunakan pemain sebelumnya. Hasil pengujian kesesuaian rekomendasi menunjukkan tingkat relevansi sebesar 86,67%. Selanjutnya, pengujian kelayakan menggunakan *System Usability Scale* (SUS) memperoleh skor rata-rata 78,56 yang termasuk kategori baik (*Acceptable*). Temuan ini menunjukkan bahwa sistem yang dikembangkan efektif dalam membantu pemain memilih kendaraan yang tepat dan meningkatkan pengalaman bermain secara keseluruhan.

## ABSTRACT

Tamami, Ihwan. 2025. **Car Recommendation System in Game Using Content-Based Filtering Method.** Thesis, Department of Informatics Engineering, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University of Malang. Supervisors: (I) Dr. Ir. Fresy Nugroho, ST., MT, IPM (II) Ahmad Fahmi Karami, M.Kom.

**Keywords:** Recommendation System, Simulation *Game*, Content-Based *Filtering*, *Cosine Similarity*, *System Usability Scale* (SUS).

The vehicle simulation *game* industry requires players to choose the right vehicle to complete challenges in various track conditions. The problem that often occurs is that players struggle to determine the car that best suits their playing style and *game* mode needs (Racing, *Offroad*, and Passenger) due to the wide variety of vehicle technical specifications. To overcome these problems, this research applies the Content-Based *Filtering* (CBF) method in the "Perfect Car" *game* to provide automatic car recommendations. This method was chosen because it is able to analyze the *similarity* of vehicle specifications based on acceleration, maximum speed, *Handling*, suspension, and capacity features using the *Cosine Similarity* algorithm. The system recommends cars that have the most similar characteristics to the vehicle used by the player previously. The results of the recommendation suitability test showed a relevance rate of 86.67%. Furthermore, feasibility testing using the *System Usability Scale* (SUS) obtained an average score of 78.56 which is in the good category (*Acceptable*). These findings indicate that the developed system is effective in helping players choose the right vehicle and improving the overall gaming experience.

## مستخلص البحث

تمامي، إخوان. ٢٠٢٥. نظام توصية السيارات في اللعبة باستخدام طريقة التصفية القائمة على المحتوى (*Content-Based Filtering*). البحث الجامعي، قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف: (الأول) الدكتور فريسي نوغروهو الماجستير، (الثانى) أحمد فهمي كرامي الماجستير.

الكلمات المفتاحية: نظام التوصية، لعبة المحاكاة، التصفية القائمة على المحتوى (*Content-Based Filtering*)، التشابه بجibb التمام (*Cosine Similarity*)، مقياس قابلية استخدام النظام (SUS).

تتطلب صناعة ألعاب محاكاة المركبات من اللاعبين اختيار المركبة المناسبة لإكمال التحديات في ظروف المسار المختلفة. المشكلة التي تحدث غالباً هي أن اللاعبين يجدون صعوبة في تحديد السيارة التي تناسب أسلوب لعبهم واحتياجات وضع اللعبة (السباق، والطرق الوعرة، والسائق). نظراً للتنوع الكبير في الموصفات الفنية للمركبة. للتغلب على هذه المشكلة، يطبق هذا البحث طريقة التصفية القائمة على المحتوى (CBF) في لعبة "فُرْقِيچَتْ چَار" لتقديم توصيات تلقائية للسيارات. تم اختيار هذه الطريقة لقدرها على تحليل تشابه مواصفات المركبة بناءً على ميزات التسارع، والسرعة القصوى، والتحكم، والتعليق، والسرعة باستخدام خوارزمية التشابه بجibb التمام (*Cosine Similarity*). يقوم النظام بتوصية السيارات التي لها خصائص مشابهة جداً للمركبة التي استخدمها اللاعب سابقاً. أظهرت نتائج اختبار ملاءمة التوصية معدل صلة بنسبة 86.67%. علاوة على ذلك، حصل اختبار المجدوى باستخدام مقياس قابلية استخدام النظام (SUS) على متوسط درجة 78.56 ما يضعه في الفئة الجيدة (مقبول). تشير هذه النتائج إلى أن النظام المطور فعال في مساعدة اللاعبين على اختيار المركبة المناسبة وتحسين تجربة اللعب بشكل عام.

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Industri *game* telah menjadi salah satu media hiburan yang sangat populer di berbagai kalangan, mulai dari anak-anak hingga orang dewasa. Dengan perkembangan teknologi yang pesat, *game* kini dapat dimainkan di berbagai platform seperti komputer, konsol, hingga smartphone, yang memberikan kemudahan akses bagi para pemainnya. Tidak hanya sekadar sarana hiburan, *game* juga memiliki potensi besar dalam berbagai bidang, termasuk pendidikan, pelatihan, ekonomi, serta pengembangan keterampilan kognitif dan sosial. Menurut penelitian yang dilakukan oleh (Mulachela, 2020), industri *game* Indonesia memberikan kontribusi yang berarti terhadap pertumbuhan ekonomi nasional, hal ini didorong oleh faktor-faktor seperti populasi yang besar dan juga akses internet dan teknologi yang semakin mudah, terutama di kalangan anak-anak dan remaja. Menurut (Yuwono, 2021) *game* modern tidak hanya berfungsi sebagai hiburan, tetapi juga dapat dimanfaatkan dalam berbagai aspek kehidupan, seperti peningkatan keterampilan berpikir kritis, kolaborasi, dan kreativitas.

Perkembangan industri *game* global juga menunjukkan pertumbuhan yang signifikan dengan pendapatan mencapai 81 miliar USD pada tahun 2014 dan diproyeksikan meningkat hingga 86 miliar USD pada tahun 2016 (Hasan dkk., 2021). Di Indonesia sendiri, industri *game* mengalami peningkatan dengan pendapatan mencapai 190 juta USD pada tahun 2013, meningkat 30% dibandingkan tahun sebelumnya (Yuwono, 2021). Dengan angka pertumbuhan yang menjanjikan

ini, pengembang *game* terus berinovasi untuk menciptakan pengalaman bermain yang lebih interaktif dan sesuai dengan preferensi pengguna.

Salah satu genre yang populer adalah *game* balap, di mana pemain dapat mengendalikan berbagai jenis kendaraan dalam berbagai lintasan. Genre *game* balap menawarkan simulasi kendaraan dalam berbagai lintasan, yang mengharuskan pemain memilih kendaraan yang sesuai dengan kondisi medan untuk meningkatkan performa bermain. Menurut penelitian yang dilakukan oleh (Herlambang Yudha dkk., 2019) kepopuleran *game* balap terjadi karena beberapa faktor utama, yaitu sensasi kecepatan yang memberikan pengalaman bermain yang mengasyikkan, variasi lintasan yang menarik, grafis yang semakin realistik, serta perkembangan kecerdasan buatan yang meningkatkan tantangan dalam permainan. *Game* ini tidak hanya menawarkan banyak pilihan kendaraan, tetapi juga variasi lintasan seperti jalan raya, sirkuit balap, serta medan *off-road* yang menantang (Alfiani, 2023). Setiap lintasan memiliki karakteristik unik yang mengharuskan pemain memilih kendaraan yang sesuai dengan kondisi medan untuk mendapatkan pengalaman bermain yang optimal.

Namun, permasalahan utama dalam *game* simulasi kendaraan adalah sistem pemilihan kendaraan yang masih bersifat acak (*randomized selection*) tanpa mempertimbangkan kemiripan spesifikasi kendaraan (Mulachela, 2020). Pemain sering kali kesulitan dalam menentukan kendaraan yang paling sesuai dengan akselerasi, kecepatan maksimal, *Handling*, suspensi dan kapasitas mobil yang mereka butuhkan. Tanpa adanya sistem rekomendasi yang cerdas, proses pemilihan kendaraan menjadi kurang efisien dan dapat mengurangi kepuasan bermain. Untuk

mengatasi hal ini, diperlukan sistem rekomendasi yang lebih adaptif yang dapat memberikan rekomendasi kendaraan yang lebih personal dan relevan.

Salah satu pendekatan yang dapat digunakan dalam membangun sistem rekomendasi adalah metode *Content-Based Filtering* (CBF). Metode ini bekerja dengan menganalisis karakteristik kendaraan dalam *game* dan mencocokkannya dengan preferensi pengguna berdasarkan riwayat interaksi mereka. Menurut penelitian yang dilakukan oleh (Putri dkk., 2022), penerapan metode *Content-Based Filtering* dalam sistem rekomendasi video *game* dapat meningkatkan akurasi rekomendasi hingga 87,75%, menunjukkan potensi signifikan dalam meningkatkan pengalaman pengguna. Dalam industri digital, metode ini telah banyak diterapkan dalam layanan *e-commerce* dan platform *streaming* untuk memberikan rekomendasi yang dipersonalisasi kepada pengguna (Farhan dkk., 2024). Namun, penerapannya dalam konteks *game* simulasi kendaraan masih sangat terbatas dan belum banyak diteliti secara mendalam.

Dalam perspektif Islam, penggunaan teknologi yang bertujuan untuk memberikan manfaat, kemudahan dan efisiensi dalam pengambilan keputusan merupakan bagian dari prinsip hikmah yang diberikan kepada manusia. Sebagaimana firman Allah Subhanahu Wa Ta'ala mengenai memberi manfaat kepada sesama pada QS. Al-Qashas ayat 84:

مَنْ جَاءَ بِالْحُسْنَةِ فَلَهُ خَيْرٌ مَّنْهَا وَمَنْ جَاءَ بِالسَّيْئَةِ فَلَا يُجْزِي اللَّذِينَ عَمِلُوا السَّيِّئَاتِ إِلَّا مَا كَانُوا يَعْمَلُونَ

“Siapa yang datang dengan (membawa) kebaikan, baginya (*pahala*) yang lebih baik daripada kebaikannya itu. Siapa yang datang dengan (membawa) kejahatan, maka orang-orang yang telah mengerjakan kejahatan itu hanya diberi balasan (*seimbang*) dengan apa yang selalu mereka kerjakan.” (QS. Al-Qashas:84)

Menurut Ibnu Katsir dalam Tafsir al-Qur'an al-'Azhim, QS. Al-Qashas ayat 84 menjelaskan bahwa Allah SWT menyediakan kenikmatan akhirat yang kekal bagi hamba-Nya yang bertaqwa dan rendah hati. Dan barangsiapa yang tidak bersikap angkuh, tidak sombong, tidak bertindak sewenang-wenang, serta tidak menimbulkan kerusakan akan memperoleh balasan yang lebih baik dari kebaikan yang mereka lakukan. Sebaliknya, setiap perbuatan buruk yang dilakukan manusia akan mendapatkan balasan yang setimpal sesuai dengan amal yang mereka perbuat. Konsep ini menegaskan bahwa setiap tindakan memiliki konsekuensi, baik di dunia maupun di akhirat.

Masalah utama yang diidentifikasi adalah kurangnya sistem rekomendasi yang secara efektif menyesuaikan saran kendaraan berdasarkan preferensi individu pemain dalam *game* bergenre simulasi kendaraan. Sebagian besar penelitian sebelumnya berfokus pada penerapan metode *Content-Based Filtering* dalam konteks umum, seperti rekomendasi produk atau konten media, tanpa memberikan perhatian khusus pada dinamika dan kebutuhan unik dalam *game* simulasi kendaraan yang sesuai dengan kebutuhan pengguna. Hal ini menunjukkan adanya kesenjangan penelitian yang perlu diisi untuk meningkatkan personalisasi dan relevansi rekomendasi dalam konteks tersebut.

Penelitian ini bertujuan untuk mengisi gap tersebut dengan mengembangkan dan mengintegrasikan sistem rekomendasi mobil yang menggunakan metode *Content-Based Filtering*, yang mempertimbangkan kemiripan spesifikasi kendaraan berdasarkan perilaku mereka dalam *game* "Perfect Car". Dengan demikian, sistem diharapkan dapat memberikan rekomendasi mobil yang lebih

akurat dan sesuai dengan speksifikasi kendaraan sesuai masing-masing pemain, meningkatkan kepuasan dan performa mereka dalam permainan serta memberikan kontribusi dalam pengembangan ilmu pengetahuan dan industri *game* secara keseluruhan (Miftachul dkk., 2022).

## 1.2 Rumusan Masalah

Bagaimana menghasilkan rekomendasi mobil yang tepat pada *game* menggunakan metode *Content-Based Filtering* dengan mempertimbangkan kemiripan spesifikasi kendaraan?

## 1.3 Batasan Masalah

Adapun batasan masalah yang diberikan pada penelitian ini sebagai berikut:

- a. Sistem rekomendasi menggunakan 5 variabel fitur, yaitu :
  - Akselerasi (F1), Kecepatan Maksimal (F2), *Handling* (F3), Suspensi (F4), Kapasitas (F5).
- b. Variabel item yang digunakan berjumlah 24 mobil yaitu:
  - Toyota Innova, Daihatsu Sigra, Toyota Fortuner, Toyota Vios, Mazda 3, Honda Civic, MG 5 GT, Daihatsu Ayla, Toyota Rush, Daihatsu GranMax, Toyota Hiace, Mercedes Benz Sprinte, DFSK Gelora, Ford Everest, Ford ranger Raptor, Toyota Avanza, Honda BRV, Audi R8, Toyota GR Supra, Nissan GT-R, Lamborghini Aventador, Jeep Wrengler, Suzuki Jimny, Toyota Hilux.
- c. Penelitian ini hanya berfokus pada preferensi pemain dalam memilih mobil berdasarkan kemiripan spesifikasi kendaraan.

- d. Faktor lain seperti desain grafis, kualitas visual, narasi permainan, maupun interaksi sosial antar pemain tidak termasuk dalam lingkup penelitian.

#### **1.4 Tujuan Penelitian**

Menerapkan sistem rekomendasi dalam *game "Perfect Car"* yang dapat merekomendasikan mobil berdasarkan metode *Content-Based Filtering* sesuai dengan Kemiripan Spesifikasi Kendaraan.

#### **1.5 Manfaat Penelitian**

Penelitian ini diharapkan dapat memberikan beberapa manfaat sebagai berikut:

- a. Meningkatkan ketepatan dan kemudahan pemilihan kendaraan melalui penerapan metode *Content-Based Filtering* (CBF), sehingga sistem mampu memberikan rekomendasi kendaraan yang sesuai dengan gaya bermain dan kondisi lintasan dalam *game Perfect Car*.
- b. Meningkatkan efisiensi proses pengambilan keputusan dengan mengurangi kebutuhan percobaan berulang (*trial and error*) dalam menentukan kendaraan yang tepat pada setiap mode permainan.

## **BAB II**

### **STUDI PUSTAKA**

#### **2.1 Penelitian Terkait**

Penelitian yang dilakukan oleh Ridwansyah dkk. (2024) berfokus pada penerapan metode *Content-Based Filtering* pada aplikasi pemesanan produk untuk meningkatkan omzet penjualan dan pemasaran. Dalam penelitian ini melibatkan *Term Frequency-Inverse Document Frequency* (TF-IDF) untuk membuat matriks kata menjadi vektor dan *Cosine Similarity* untuk menghitung nilai kemiripan antar kata pada vektor. Hasil penelitian menunjukkan bahwa sistem rekomendasi yang dibangun memiliki akurasi mencapai 73%. Peneliti menyimpulkan bahwa metode ini efektif dalam membantu pelanggan memilih rekomendasi produk yang dibutuhkan dengan lebih cepat dan efisien.

Penelitian yang dilakukan oleh Kurniaji & Santi (2023) mengimplementasikan metode *Content-Based Filtering* dalam sistem rekomendasi komik digital. Penelitian ini memanfaatkan TF-IDF untuk pembobotan kata dan *Cosine Similarity* untuk menghitung kemiripan antar komik berdasarkan sinopsis, genre, dan nama komikus. Pengujian dilakukan dengan menggunakan komik *Ao Ashi* sebagai preferensi utama. Hasil penelitian menunjukkan bahwa nilai *Cosine Similarity* tertinggi sebesar 0.13901 dengan persentase kemiripan 100%, sedangkan nilai terendah pada peringkat kesepuluh sebesar 0.09046 dengan persentase kemiripan 65.08%. Rata-rata persentase kemiripan yang diperoleh adalah 76.38%, yang menunjukkan bahwa sistem mampu memberikan rekomendasi yang sesuai dengan preferensi pengguna.

penelitian oleh Azri Saputra dkk. (2024) membahas penerapan metode *Content-Based Filtering* dalam sistem rekomendasi film pada platform *streaming*. Dalam penelitian ini, TF-IDF digunakan untuk pembobotan kata dalam deskripsi film, sedangkan *Cosine Similarity* digunakan untuk menghitung kemiripan antarfilm. Evaluasi sistem dilakukan dengan menggunakan metrix *Precision@K* (*P@K*), *Average Precision@K* (*AP@K*), dan *Mean Average Precision@K* (*MAP@K*). Hasil penelitian menunjukkan bahwa sistem rekomendasi memiliki nilai *MAP@K* sebesar 0.7022 untuk rekomendasi film tipe *movie* dan 0.7973 untuk tipe *show*, yang berarti sistem menunjukkan efektivitas metode yang diterapkan.

Penelitian yang dilakukan oleh Utomo dkk. (2024) berfokus pada penerapan metode *Content-Based Filtering* pada sistem rekomendasi pemilihan buku dengan referensi Rumah Belajar Pancasila. Evaluasi sistem yang digunakan oleh peneliti dilakukan dengan menggunakan metrix *precision* dan *recall*. Hasil penelitian menunjukkan bahwa sistem rekomendasi yang dibangun memiliki rata-rata *precision* sebesar 0,90 (90%) dan *recall* sebesar 1(100%), yang menunjukkan bahwa metode yang dipakai oleh sistem ini efektif dalam memberikan rekomendasi buku yang sesuai dengan preferensi pengguna.

Selain itu, penelitian yang dilakukan oleh Putra & Santika (2020) membahas implementasi *machine learning* dalam sistem rekomendasi musik dengan metode *Content-Based Filtering*. Penelitian ini bertujuan untuk memberikan rekomendasi musik sesuai dengan preferensi pengguna guna meningkatkan pengalaman pengguna. Dalam penelitian ini, digunakan *Cosine Similarity* untuk mengukur tingkat kemiripan antara lagu berdasarkan atribut yang dimiliki. Evaluasi dilakukan

menggunakan *precision* dan *recall*. Hasil penelitian menunjukkan bahwa sistem rekomendasi yang dibangun menghasilkan nilai *similarity* rata-rata sebesar 0.6684, *precision* sebesar 0.125, dan *recall* sebesar 0.200. Hasil pengujian menunjukkan bahwa sistem rekomendasi mampu bekerja dengan baik dengan rata-rata *response time* sebesar 3,5 detik. Namun, penelitian ini juga menyarankan bahwa untuk menangani data dalam jumlah yang lebih besar, diperlukan algoritma yang lebih efisien.

Penelitian yang dilakukan oleh Viljanen dkk. (2020) mengembangkan model berbasis konten untuk rekomendasi *game* dalam mengatasi masalah cold start. Dalam penelitian ini, *Cosine Similarity* digunakan untuk mengukur kemiripan antara *game* berdasarkan tag dan fitur yang ada pada *game*, serta preferensi pemain yang diambil dari data survei. Model ini terbukti lebih unggul daripada metode Collaborative *Filtering* dalam situasi di mana tidak ada interaksi historis antara pengguna dan *game*, yaitu dalam kasus permainan baru atau pemain baru. Peneliti menunjukkan bahwa *Cosine Similarity* dapat diandalkan dalam memberikan rekomendasi yang tepat meskipun data pengguna terbatas.

Penelitian yang dilakukan oleh Pragusma dkk. (2023) mengimplementasikan metode *Content-Based Filtering* (CBF) dalam sistem rekomendasi *game*. Dalam penelitian ini, peneliti menggunakan *Cosine Similarity* untuk menghitung kemiripan antara *game* berdasarkan atribut yang dimiliki seperti kategori, genre, dan pengembang. Pengguna memberikan input berupa *game* yang telah dimainkan atau diminati sebelumnya, yang kemudian digunakan untuk membangun profil pengguna. Hasil dari pengujian menunjukkan bahwa sistem

rekomendasi yang dibangun menggunakan metode CBF ini berhasil mencapai akurasi 82%, yang membuktikan efektivitas metode ini dalam memberikan rekomendasi *game* yang relevan dengan preferensi pengguna. Penelitian ini mengemukakan bahwa metode ini dapat digunakan untuk membantu pengguna menemukan *game* yang sesuai dengan minat mereka tanpa bergantung pada interaksi data pengguna lainnya.

Penelitian oleh Xu (2021) juga menggunakan metode *Content-Based Filtering* (CBF) untuk platform lelang mobil online. Dalam penelitian ini, *Cosine Similarity* digunakan untuk mengukur kemiripan antara mobil berdasarkan atribut-atribut produk seperti jenis mobil, harga, dan tahun pembuatan. Data preferensi pengguna seperti mobil yang telah dilihat atau dibeli sebelumnya digunakan untuk membangun profil pengguna. Dalam penelitian ini, menemukan bahwa sistem berbasis konten ini berhasil memberikan rekomendasi mobil yang sesuai dengan preferensi pengguna dan dapat mengatasi masalah sparsity serta cold start, yang sering muncul dalam Collaborative *Filtering*.

Di sisi lain, penelitian oleh Yuen dkk. (2023) mengembangkan sistem rekomendasi *game* yang menggunakan kombinasi *Content-Based Filtering* dan Collaborative *Filtering*. Sistem ini mengadopsi teknik web scraping untuk mengumpulkan data dari platform Steam, yang kemudian dianalisis menggunakan algoritma rekomendasi untuk memberikan hasil yang lebih relevan dengan preferensi pengguna. Selain itu, penggunaan visualisasi data dalam sistem ini meningkatkan pengalaman pengguna dengan menampilkan data secara interaktif, yang memungkinkan pemain untuk memilih *game* berdasarkan genre dan ulasan

yang sesuai dengan minat mereka. Pendekatan ini memastikan bahwa rekomendasi yang diberikan lebih personal dan sesuai dengan gaya bermain pemain, menjadikan pengalaman bermain lebih menyenangkan dan memuaskan.

Tabel 2.1 Penelitian Terdahulu

No	Nama Peneliti	Judul Penelitian	Metode	Hasil Penelitian
1	(Ridwansyah dkk., 2024)	Penerapan Metode <i>Content-Based Filtering</i> pada Sistem Rekomendasi	<i>Content-Based Filtering</i>	Menggunakan TF-IDF untuk membuat matriks kata menjadi vektor dan <i>Cosine Similarity</i> untuk menghitung nilai kemiripan antar kata pada vektor. Akurasi sistem rekomendasi mencapai 73%.
2	(Kurniaji & Santi, 2023)	Implementasi Metode Content Based <i>Filtering</i> Pada Pemilihan Komik	<i>Content-Based Filtering</i>	Menggunakan metode TF-IDF untuk pembobotan kata dan <i>Cosine Similarity</i> untuk menghitung kemiripan antar komik berdasarkan sinopsis, genre, dan nama komikus. nilai <i>Cosine Similarity</i> tertinggi sebesar 0.13901 dengan persentase kemiripan 100%, sedangkan nilai terendah pada peringkat kesepuluh sebesar 0.09046 dengan persentase kemiripan 65.08%. Rata-rata persentase kemiripan yang diperoleh adalah 76.38%.

No	Nama Peneliti	Judul Penelitian	Metode	Hasil Penelitian
3	(Azri Saputra dkk., 2024)	Sistem Rekomendasi Film pada Platform Streaming Menggunakan Metode <i>Content-Based Filtering</i>	<i>Content-Based Filtering</i>	Menggunakan TF-IDF digunakan untuk pembobotan kata dalam deskripsi film, sedangkan <i>Cosine Similarity</i> digunakan untuk menghitung kemiripan antarfilm. nilai MAP@K sebesar 0.7022 untuk rekomendasi film tipe <i>movie</i> dan 0.7973 untuk tipe <i>show</i> .
4	(Utomo dkk., 2024)	Penerapan Metode Content Based <i>Filtering</i> Pada Sistem Rekomendasi Pemilihan Buku Referensi Rumah Belajar Pancasila	<i>Content-Based Filtering</i>	Menggunakan metrix <i>precision</i> dan <i>recall</i> . rata-rata <i>precision</i> sebesar 0,90 (90%) dan <i>recall</i> sebesar 1(100%).
5	(Putra & Santika, 2020)	Implementasi Machine Learning dalam Penentuan Rekomendasi Musik dengan Metode <i>Content-Based Filtering</i>	<i>Content-Based Filtering</i>	Menggunakan <i>Cosine Similarity</i> untuk mengukur tingkat kemiripan antara lagu berdasarkan atribut yang dimiliki. nilai <i>similarity</i> rata-rata sebesar 0.6684, <i>precision</i> sebesar 0.125, dan <i>recall</i> sebesar 0.200. Rata-rata <i>response time</i> sebesar 3,5 detik, tetapi peneliti menyarankan jumlah data yang lebih besar agar algoritma menjadi efisien.
6.	(Viljanen dkk., 2020)	Content-Based Player and <i>Game</i> Interaction Model for <i>Game</i> Recommendation in the Cold Start Setting	<i>Content-Based Filtering</i>	Menggunakan <i>Cosine Similarity</i> untuk mengukur kemiripan antara <i>game</i> dan pemain berdasarkan atribut <i>game</i> dan preferensi pemain. Ditemukan bahwa model ini dapat mengatasi masalah cold start dengan lebih baik daripada Collaborative <i>Filtering</i> dan memberikan rekomendasi yang relevan meskipun data interaksi terbatas.

No	Nama Peneliti	Judul Penelitian	Metode	Hasil Penelitian
7.	(Pragusma dkk., 2023)	<i>Game Recommendation Using Content-based Algorithm</i>	<i>Content-Based Filtering</i>	Menggunakan <i>Cosine Similarity</i> untuk menghitung kemiripan antara <i>game</i> yang dipilih pengguna dan <i>game</i> yang ada dalam database berdasarkan atribut genre, kategori, dan developer. Sistem mencapai akurasi 82% dalam memberikan rekomendasi yang relevan dengan preferensi pengguna.
8.	(Xu, 2021)	A Hybrid Recommendation System for Online Car Auction Platform	<i>Hybrid (Collaborative &amp; Content-Based)</i>	Menggunakan <i>Cosine Similarity</i> untuk mengukur kesamaan antara produk (mobil) yang diinginkan oleh pengguna dengan produk lainnya di platform lelang mobil. Penelitian ini menggabungkan <i>Content-Based Filtering</i> dengan <i>Collaborative Filtering</i> untuk menghasilkan rekomendasi yang lebih personalisasi dan mengatasi masalah sparsity dalam dataset.
9.	(Chythanya dkk., 2019)	A Novel Video Game Recommender System using Content Based Filtering	<i>Content-Based Filtering</i>	Menggunakan <i>Cosine Similarity</i> dan TF-IDF untuk mengukur kemiripan antar <i>game</i> berdasarkan atribut seperti genre, plot, dan elemen <i>gameplay</i> . Peneliti menyimpulkan bahwa pendekatan ini sangat efektif untuk memberikan rekomendasi <i>game</i> yang relevan dengan preferensi pengguna berdasarkan analisis konten <i>game</i> tersebut.

No	Nama Peneliti	Judul Penelitian	Metode	Hasil Penelitian
10.	(Yuen dkk., 2023)	<i>Game Recommendation System</i>	<i>Content-Based Filtering &amp; Collaborative Filtering</i>	Sistem rekomendasi <i>game</i> yang menggunakan kombinasi <i>Content-Based Filtering</i> dan <i>Collaborative Filtering</i> . Penggunaan teknik web scraping dan visualisasi data membantu dalam meningkatkan pengalaman pengguna, memberikan rekomendasi yang lebih relevan dan personalisasi sesuai preferensi pemain.

## 2.2 *Game*

*Game*, atau dalam bahasa Indonesia disebut "permainan", adalah aktivitas terstruktur yang dilakukan untuk hiburan, pendidikan, atau tujuan lainnya. *Game* atau dalam Bahasa indonesia yaitu “gim” yang menurut KBBI yang berarti permainan. Permainan adalah sesuatu yang digunakan untuk bermain, barang atau sesuatu yang dipermainkan, atau mainan (Iskandar & Cahyadi, 2021). *Game* merupakan kegiatan atau aktivitas yang diperuntukkan untuk bersenang-senang, dan juga digunakan untuk mengisi waktu luang, berefreshing atau bisa juga untuk berolahraga ringan (Husein dkk., 2019).

Menurut (Hadisaputra, 2022) *game* didefinisikan sebagai sebuah aktivitas yang memiliki aturan tertentu yang memungkinkan pemainnya untuk berkompetisi dalam mencapai kemenangan atau sekadar bersenang-senang sebagai bentuk hiburan. Permainan ini tidak hanya dirancang untuk mengisi waktu luang, tetapi juga berfungsi sebagai alat untuk melatih strategi dan keterampilan berpikir, baik dalam permainan individu maupun yang melibatkan banyak pemain. Dari

perspektif pendidikan, *game* juga dapat menjadi media pembelajaran yang efektif dengan menghadirkan skenario yang memungkinkan pemain memahami konsep atau materi tertentu secara interaktif. Dengan demikian, *game* memiliki peran yang luas, baik sebagai hiburan, sarana pelatihan, hingga media edukasi yang memberikan manfaat kognitif bagi pemainnya.

Seiring dengan perkembangan teknologi digital, industri *game* telah mengalami evolusi pesat, menciptakan berbagai genre yang disesuaikan dengan kebutuhan seperti mencakup *game* edukasi, simulasi, arcade, strategi, dan RPG (Marzian & Qamal, 2017). *Game* edukasi berkembang sebagai alat pembelajaran interaktif yang menggabungkan elemen permainan dengan materi akademik atau keterampilan tertentu. Menurut penelitian dari (Erfan & Maulyda, 2020), *game* edukasi memiliki dampak positif dalam meningkatkan pemahaman dan kesadaran lingkungan, serta dapat digunakan sebagai media pembelajaran alternatif. Salah satu contohnya adalah *game* edukasi yang memperkenalkan nama-nama kendaraan kepada anak-anak, membantu mereka mengenali berbagai jenis transportasi dengan cara yang menyenangkan.

Selain itu, *game* simulasi juga mengalami pertumbuhan yang signifikan, terutama dalam bidang otomotif. *Game* simulasi mobil, seperti "VR: Parking Simulator", dirancang untuk memberikan pengalaman realistik dalam mengemudikan dan memarkir kendaraan, dengan mempertimbangkan aspek fisika, kondisi jalan, serta sistem mekanik yang mendekati dunia nyata. Simulasi ini tidak hanya berfungsi sebagai hiburan, tetapi juga sebagai sarana pelatihan bagi

pengemudi pemula untuk memahami teknik parkir yang benar (Gamadya Wiratamtama dkk., 2020).

Sejalan dengan itu, *game* simulasi edukasi yang dikembangkan oleh (Bagus dkk., 2015) mengajarkan mengenai rambu-rambu lalu lintas juga untuk membantu pengendara memahami dan menghafal arti dari berbagai rambu yang ada. *Game* semacam ini menggunakan metode interaktif untuk meningkatkan pengetahuan pengguna tentang peraturan lalu lintas dengan cara yang lebih menarik. Dengan demikian, baik *game* edukasi maupun simulasi memiliki peran penting dalam memberikan manfaat edukatif dan praktis bagi pemainnya, terutama dalam konteks pembelajaran tentang kendaraan dan keselamatan berkendara.

*Game* simulasi memiliki potensi besar untuk menjadi alat pembelajaran yang efektif karena mampu menghadirkan pengalaman interaktif yang menarik dan meningkatkan keterlibatan pengguna dalam memahami konsep tertentu. Dengan pendekatan berbasis permainan, *game* edukasi dapat merangsang motivasi belajar, meningkatkan daya ingat, serta mengembangkan keterampilan kognitif dan pemecahan masalah. Selain itu, *game* edukasi memungkinkan penyampaian materi yang kompleks menjadi lebih mudah dipahami melalui simulasi, animasi, dan skenario interaktif yang menyerupai situasi dunia nyata.

### 2.3 Sistem Rekomendasi

Sistem rekomendasi adalah teknologi yang dirancang untuk membantu pengguna untuk menyaring dan menemukan item atau informasi yang relevan di tengah banyaknya pilihan yang tersedia berdasarkan analisis data perilaku dan preferensi pengguna sebelumnya. Menurut ('Alim dkk., 2020) sistem rekomendasi

adalah sebuah teknologi yang dirancang untuk mengolah informasi pengguna dan memberikan saran yang relevan berdasarkan karakteristik pengguna tersebut. Sistem ini sering diterapkan dalam berbagai bidang yang memiliki jumlah data besar dan terus bertambah, seperti pencarian lowongan kerja, rekomendasi film, rekomendasi buku, dan e-learning. Tujuan utamanya adalah membantu pengguna menemukan item yang sesuai dengan preferensi atau kebutuhan mereka, seperti produk, layanan, atau konten digital.

Ada beberapa metode dalam sistem rekomendasi, diantaranya adalah *Collaborative filtering* dan *Content-Based Filtering*. *Collaborative filtering* adalah salah satu teknik yang paling umum digunakan dalam sistem rekomendasi untuk memberikan saran yang dipersonalisasi kepada pengguna. Teknik ini bekerja dengan memprediksi minat pengguna berdasarkan pola preferensi yang serupa dari pengguna lain. Sedangkan *Content-Based Filtering* bekerja dengan menganalisis karakteristik dari item yang disukai oleh pengguna sebelumnya, kemudian merekomendasikan item lain yang memiliki karakteristik serupa (Didi Riswan dkk., 2024).

Dalam operasionalnya, sistem rekomendasi mengumpulkan data dari aktivitas pengguna, seperti riwayat pencarian, penilaian, atau pembelian. Data ini kemudian dianalisis untuk memahami preferensi dan minat pengguna. Berdasarkan analisis tersebut, sistem akan menyusun daftar rekomendasi yang sesuai dengan kebutuhan dan keinginan pengguna. Sebagai contoh, dalam platform e-commerce, sistem rekomendasi dapat menyarankan produk yang kemungkinan besar diminati

oleh pengguna berdasarkan pembelian sebelumnya atau perilaku pengguna lain yang serupa (Didi Riswan dkk., 2024).

#### **2.4 Content-Based Filtering (CBF)**

*Content-Based Filtering (CBF)* adalah salah satu metode dalam sistem rekomendasi yang bekerja dengan menganalisis karakteristik atau atribut dari item yang telah digunakan oleh pengguna, kemudian mencocokkannya dengan item lain yang memiliki tingkat kemiripan yang tinggi. Pendekatan ini berfokus pada preferensi spesifik yang dimiliki oleh pengguna berdasarkan pola interaksi mereka terhadap item tertentu. Menurut (Putri dkk., 2022), sistem ini bekerja dengan membangun profil pengguna berdasarkan atribut dari item yang mereka pilih dan memberikan rekomendasi yang relevan.

Dalam konteks implementasi metode *Content-Based Filtering* pada game "Perfect Car", sistem rekomendasi ini bekerja dengan menilai pola perilaku pemain dalam menggunakan mobil tertentu selama permainan berlangsung. Sebagai contoh, apabila seorang pemain menggunakan mobil dengan akselerasi yang tinggi dan *Handling* yang stabil, maka sistem secara otomatis akan mengidentifikasi karakteristik tersebut sebagai preferensi spesifik pemain. Berdasarkan analisis ini, sistem akan merekomendasikan mobil lain yang memiliki karakteristik serupa untuk meningkatkan pengalaman bermain pemain dengan memberikan opsi kendaraan yang paling sesuai dengan gaya bermain mereka. Dengan demikian, sistem rekomendasi yang diterapkan dalam game ini mampu memberikan saran yang lebih personal dan disesuaikan dengan kebutuhan individu setiap pemain, sehingga meningkatkan kepuasan mereka dalam bermain.

### 2.4.1 Tahapan *Content-Based Filtering*

Dalam penerapannya, metode *Content-Based Filtering* terdiri dari beberapa tahapan utama yang harus dilakukan agar sistem rekomendasi dapat berfungsi secara optimal. Tahapan-tahapan tersebut mencakup:

**a. Ekstraksi Fitur**

Pada tahap ini, sistem mengidentifikasi berbagai karakteristik utama dari setiap mobil yang tersedia dalam *game*. Fitur-fitur yang dianalisis dapat mencakup akselerasi, kecepatan maksimal, *Handling*, suspensi dan kapasitas. Data fitur ini kemudian disimpan dan digunakan untuk membandingkan kemiripan antara satu mobil dengan mobil lainnya.

**b. Pencocokan Item**

Sistem akan melakukan pencocokan antara mobil yang telah digunakan dengan mobil lain yang memiliki fitur serupa. Perhitungan kemiripan ini biasanya dilakukan menggunakan teknik seperti *Cosine Similarity* yang memungkinkan sistem untuk menemukan mobil dengan spesifikasi teknis yang paling mirip dan sesuai dengan gaya bermain pengguna.

Menurut (Putri dkk., 2022) metode *Content-Based Filtering* terbukti efektif dalam memberikan rekomendasi yang lebih akurat karena sistem ini secara langsung menganalisis spesifikasi kendaraan yang digunakan oleh pemain, dan mencocokkannya dengan kendaraan lain yang memiliki kemiripan spesifikasi teknis. Dengan pendekatan ini, rekomendasi mobil didasarkan pada kecocokan atribut seperti akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas kendaraan, yang lebih relevan dengan kebutuhan dan preferensi pengguna. Hal ini

menjadikan metode ini lebih fleksibel dan adaptif, terutama dalam situasi di mana jumlah pengguna belum terlalu besar atau data dari pengguna lain belum cukup tersedia untuk digunakan dalam proses rekomendasi.

Dalam implementasinya pada game "*Perfect Car*", sistem rekomendasi berbasis *Content-Based Filtering* akan bekerja secara dinamis dengan terus memperbarui profil pengguna berdasarkan interaksi mereka selama bermain. Dengan menggunakan teknik perhitungan kemiripan seperti *Cosine Similarity* atau sistem dapat secara efektif mengidentifikasi dan merekomendasikan mobil yang paling sesuai dengan pola penggunaan pemain.

## 2.5 *Cosine Similarity*

*Cosine Similarity* adalah salah satu pendekatan atau teknik dalam *content-based similarity* yang digunakan untuk mengukur tingkat kemiripan antara dua objek berdasarkan sudut kosinus dari dua vektor dalam ruang dimensi. Menurut (Putri et al., 2022), metode *Cosine Similarity* banyak diterapkan dalam berbagai sistem rekomendasi karena kemampuannya dalam mengukur kemiripan antara dua vektor tanpa dipengaruhi oleh skala atau perbedaan absolut antar nilai fitur yang digunakan dalam analisis. Dalam sistem rekomendasi berbasis *Content-Based Filtering*, *Cosine Similarity* sangat efektif karena dapat membandingkan karakteristik suatu item dengan item lain berdasarkan sudut kosinus antara vektor fitur yang merepresentasikan masing-masing item tersebut. Hal ini menjadikannya metode yang handal dalam menemukan item yang paling relevan dengan preferensi pengguna, tanpa terpengaruh oleh perbedaan satuan atau besaran nilai fitur yang digunakan.

$$\text{Cosine Similarity } (A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} \quad (2.1)$$

Keterangan:

- $A$  dan  $B$  adalah vektor fitur dari dua item yang dibandingkan.
- $A \cdot B$  adalah hasil perkalian dot *product* dari kedua vektor.
- $\|A\| \times \|B\|$  adalah panjang (norma) masing-masing vektor.

Pemilihan metode *Cosine Similarity* dalam penelitian ini didasarkan pada beberapa pertimbangan. Pertama, *Cosine Similarity* mampu mengukur tingkat kemiripan antar item tanpa dipengaruhi oleh perbedaan skala atau besar nilai fitur, sehingga lebih objektif dibanding metode berbasis jarak Euclidean yang sensitif terhadap perbedaan magnitudo data (Samuel dkk., 2018). Kedua, metode ini banyak digunakan dalam penelitian sistem rekomendasi berbasis konten karena perhitungannya sederhana, efisien, serta memberikan hasil yang konsisten pada data berdimensi tinggi (Putri dkk., 2022). Ketiga, dalam konteks *game* “*Perfect Car*”, setiap mobil direpresentasikan sebagai vektor fitur (akselerasi, kecepatan, *Handling*, suspensi, dan kapasitas). *Cosine Similarity* memungkinkan sistem menemukan mobil yang paling mirip secara spesifikasi tanpa terpengaruh perbedaan unit ukuran antar fitur. Dengan alasan tersebut, *Cosine Similarity* dinilai sebagai pendekatan yang paling tepat dan relevan untuk mendukung metode *Content-Based Filtering* dalam penelitian ini. Nilai *Cosine Similarity* biasanya berada dalam rentang 0 hingga 1, karena vektor umumnya bernilai positif (Samuel et al., 2018). Jika dua mobil memiliki nilai *Cosine Similarity* yang mendekati 1, maka dapat disimpulkan bahwa kedua mobil memiliki karakteristik spesifikasi yang mirip dan dapat direkomendasikan kepada pengguna berdasarkan preferensi mereka.

## 2.6 *System Usability Scale (SUS)*

*System Usability Scale (SUS)* merupakan instrumen yang digunakan untuk mengukur seberapa efektif, efisien, dan menyenangkan sebuah sistem atau aplikasi dari sudut pandang penggunanya (Sri Handayani & Adelin, 2021). SUS terdiri dari sepuluh pertanyaan yang dirancang untuk menangkap pengalaman pengguna mengenai kemudahan penggunaan dan kualitas antarmuka sistem. Pertanyaan tersebut dibagi menjadi dua jenis, yakni lima pernyataan positif yang menggambarkan kemudahan penggunaan dan lima pernyataan negatif yang menggambarkan kesulitan atau ketidaknyamanan pengguna. Setiap responden memberikan penilaian untuk setiap pernyataan menggunakan skala Likert yang terdiri dari lima pilihan, mulai dari "sangat tidak setuju" hingga "sangat setuju" (Sri Handayani & Adelin, 2021). Berikut merupakan rumus perhitungan skor SUS.

$$\bar{x} = \frac{\Sigma x}{n} \times 2.5 \quad (2.2)$$

Dimana:

- $\bar{x}$  adalah skor rata-rata
- $\Sigma x$  adalah jumlah skor SUS
- $n$  adalah jumlah responden

## **BAB III**

### **DESAIN DAN IMPLEMENTASI**

#### **3.1 Analisis dan Perancangan *Game***

##### **3.1.1 Analisis *Game***

*Game Perfect Car* merupakan *game* 3D simulasi kendaraan yang dimainkan dalam mode *single player*. Pemain memulai permainan dengan memilih mobil yang akan digunakan dan harus menyelesaikan mode permainan yang menuntut keterampilan mengemudi dan strategi pemilihan kendaraan yang tepat. *Game* ini memiliki beberapa mode seperti mode Balap, mode Penumpang dan mode *Offroad*. Setiap mode permainan memiliki tantangan unik yang mengharuskan pemain beradaptasi dengan kondisi lintasan dan kendaraan yang tersedia. Mode Balap akan dilakukan pada lintasan balap dengan tujuan mencapai garis *finish* secepat mungkin. Mode *Offroad* disini akan menguji pemain dengan ditempatkan pada lintasan yg menantang dan menguji kemampuan kendaraan di medan sulit seperti bebatuan, tanah dan jalan yang tidak rata untuk menuju ke tempat tujuan dengan waktu yang telah di tentukan. Mode Penumpang menuntut pemain mengantarkan penumpang ke lokasi tujuan sesuai waktu yang ditentukan. Jumlah penumpang yang harus diantar sama pada setiap misi, namun kapasitas tiap kendaraan berbeda sehingga pemain harus memilih mobil yang sesuai agar proses pengantaran lebih efisien.

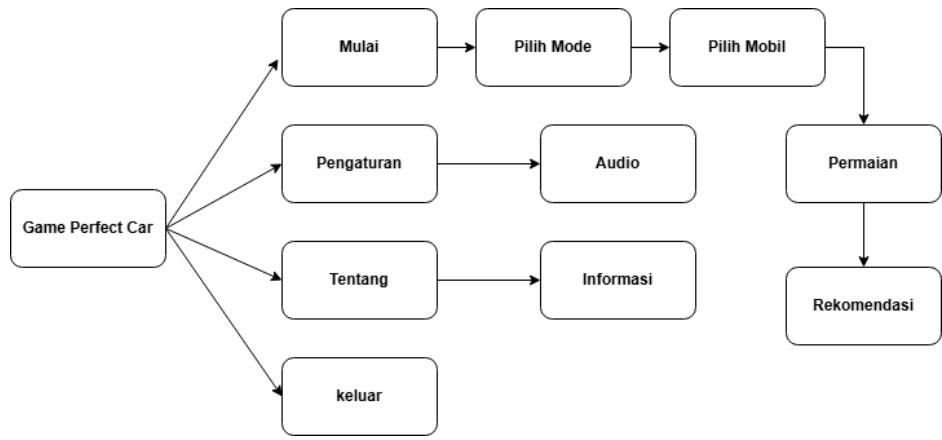
Pemain dapat mengontrol kendaraan menggunakan tombol “WASD” untuk pergerakan dan tombol “SPACE” untuk penggereman. Selama permainan berlangsung, sistem secara otomatis mencatat pola berkendara pemain, seperti

kecepatan rata-rata, gaya akselerasi, dan efektivitas pengereman. Berdasarkan data ini, sistem rekomendasi yang berbasis *Content-Based Filtering (CBF)* akan menilai kendaraan mana yang paling cocok untuk pemain berdasarkan spesifikasi atribut seperti akselerasi, kecepatan maksimal, *Handling*, suspensi dan kapasitas.

Dalam implementasi *game Perfect Car*, setiap mode hanya menampilkan kendaraan yang telah difilter berdasarkan kecocokan karakteristiknya. Mobil yang tidak sesuai dengan kebutuhan mode tidak akan muncul pada daftar pemilihan kendaraan. Karena setiap mode telah membatasi pilihan kendaraan melalui proses *filtering*, pemain tidak dapat memilih mobil yang tidak relevan dengan kondisi lintasan. Sistem rekomendasi kemudian bekerja hanya pada daftar kendaraan yang tersedia untuk mode tersebut, sehingga hasil rekomendasi lebih fokus dan akurat. Sebaliknya, sistem akan tetap menyesuaikan rekomendasi dengan spesifikasi mobil yang telah dipilih pemain, sehingga rekomendasi yang muncul adalah kendaraan dengan karakteristik yang hampir mirip (misalnya sesama sedan dengan nilai akselerasi dan *Handling* serupa). Dengan cara ini, sistem rekomendasi mempertahankan konsistensi preferensi pemain sekaligus memberikan arahan kendaraan alternatif yang relevan berdasarkan kemiripan spesifikasi.

### 3.1.2 Perancangan Game

*Game Perfect Car* dirancang dengan alur permainan yang dimulai dari halaman utama yang menampilkan beberapa opsi tombol menu utama, yaitu mulai, pengaturan, tentang, dan keluar. Dapat digambarkan dalam gambar 3.1.

Gambar 3.1 *Menu* Diagram

Berdasarkan gambar 3.1, dimana pemain memilih opsi tombol mulai, mereka akan diberikan menu pemilihan mode permainan. Setelah pemain memilih salah satu mode permainan yang tersedia, mereka kemudian dapat memilih mobil yang ingin mereka gunakan untuk menjalankan mode tersebut. Setiap mode memiliki tantangan yang berbeda, termasuk variasi lintasan, rintangan, serta kondisi medan yang mempengaruhi performa kendaraan. Pemain akan mengendalikan mobil menggunakan tombol WASD untuk bergerak, tombol P untuk pause *game* dan tombol SPACE untuk melakukan pengereman. Setelah menyelesaikan mode permainan, sistem akan menganalisis pola permainan pemain untuk memberikan rekomendasi mobil yang paling sesuai spesifikasi fitur. Dalam menu pengaturan, pemain dapat menyesuaikan aspek dalam *game* seperti menaikkan atau menurunkan volume musik. Menu tentang berisi informasi mengenai pengembang serta ucapan terima kasih kepada pemain yang telah memainkan *game* ini. Sementara itu, jika pemain memilih Keluar, maka permainan akan berakhir dan langsung kembali ke desktop.

Sistem rekomendasi dalam *Perfect Car* menggunakan metode Content Based *Filtering* (CBF) untuk memberikan rekomendasi kendaraan yang paling sesuai dengan preferensi spesifikasi mobil yang dipakai. Proses ini dimulai dengan pengumpulan data penggunaan mobil, di mana sistem mencatat mobil yang sering digunakan pemain untuk berkendara mereka selama menjalankan mode permainan. Data ini dianalisis berdasarkan beberapa atribut utama kendaraan seperti akselerasi, kecepatan maksimal, *Handling*, suspensi dan kapasitas. Setelah melakukan analisis, sistem menghitung kesamaan kendaraan dengan menggunakan *Cosine Similarity* untuk menentukan fitur spesifikasi mobil yang dipilih pemain. Ketika sistem menemukan mobil dengan tingkat kesamaan tertinggi terhadap pola bermain pemain, kendaraan tersebut akan direkomendasikan sebagai pilihan optimal. Dengan sistem ini, pemain mendapatkan rekomendasi kendaraan yang lebih sesuai dengan preferensi mereka, memungkinkan mereka untuk meningkatkan performa dalam menjalankan mode permainan. *Game* akan berakhir setelah pemain menerima rekomendasi kendaraan terbaik, menandakan bahwa sistem telah berhasil membantu pemain menemukan kendaraan yang paling sesuai dengan gaya bermain mereka.

### 3.1.3 Rancangan Antarmuka

Pada tahap perancangan ini, antarmuka dan alur permainan *game Perfect Car* disusun dalam bentuk *storyboard* untuk menggambarkan rancangan tampilan serta interaksi yang akan dialami oleh pemain. *Storyboard* digunakan sebagai pedoman visual sebelum proses implementasi dilakukan, sehingga setiap elemen yang disajikan masih berupa sketsa dan belum menampilkan tampilan akhir dari

game. Melalui rancangan ini, alur navigasi, posisi tombol, serta struktur informasi dapat divisualisasikan secara lebih jelas dan sistematis. Berikut merupakan rancangan *storyboard* dari game *Perfect Car*.

Tabel 3.1 *Storyboard Game Perfect Car*

No	Gambar	Keterangan
1.	<p>TEXT JUDUL GAME "PERFECT CAR" SIZE JUDUL GAME BESAR</p> <p>ILUSTRASI MOBIL</p> <p>TOMBOL KLIK UNTUK MULAI</p>	Tampilan awal game <i>Perfect Car</i> yang memungkinkan pemain mengetuk bagian mana pun pada layar untuk melanjutkan ke menu utama.
2.	<p>ILUSTRASI MOBIL</p> <p>TOMBOL MULAI</p> <p>TOMBOL PENGATURAN</p> <p>TOMBOL TENTANG</p> <p>TOMBOL KELUAR</p> <p>MOBIL YG SAMA DENGAN DI TAMPAKAN AWAL</p>	Menu utama menampilkan empat pilihan yaitu <i>Mulai</i> , <i>Pengaturan</i> , <i>Tentang</i> , dan <i>Keluar</i> sebagai navigasi awal pemain.
3.	<p>TOMBOL KEMBALI</p> <p>BACKGROUND ILUSTRASI MOBIL</p> <p>TEXT MENU PILIH MODE</p> <p>TEXT BALAP</p> <p>ILUSTRASI MOBIL BALAP</p> <p>TOMBOL PILIH</p> <p>TEXT OFFROAD</p> <p>ILUSTRASI MOBIL OFFROAD</p> <p>TOMBOL PILIH</p> <p>TEXT PENUMPANG</p> <p>ILUSTRASI MOBIL SEPERTI TRAVEL</p> <p>TOMBOL PILIH</p> <p>BACKGROUND ILUSTRASI MOBIL</p>	Pada halaman ini pemain memilih salah satu dari tiga mode permainan, yaitu <i>Mode Balap</i> , <i>Mode Offroad</i> , dan <i>Mode Penumpang</i> .
4.	<p>TOMBOL KEMBALI</p> <p>NAMA MOBIL</p> <p>SPESIFIKASI MOBIL</p> <p>LATAR CARAS</p> <p>TOMBOL PILIH</p>	Pemain memilih mobil berdasarkan preferensi dan spesifikasi yang diinginkan sebagai persiapan sebelum memasuki permainan.

No	Gambar	Keterangan
5.		Mode Penumpang berlangsung di lingkungan perkotaan, di mana pemain bertugas menyelesaikan misi dengan mengantarkan penumpang ke tujuan.
6.		Mode Balap berlangsung di lintasan sirkuit, dan pemain harus berusaha memenangkan perlombaan dengan mengalahkan lawan.

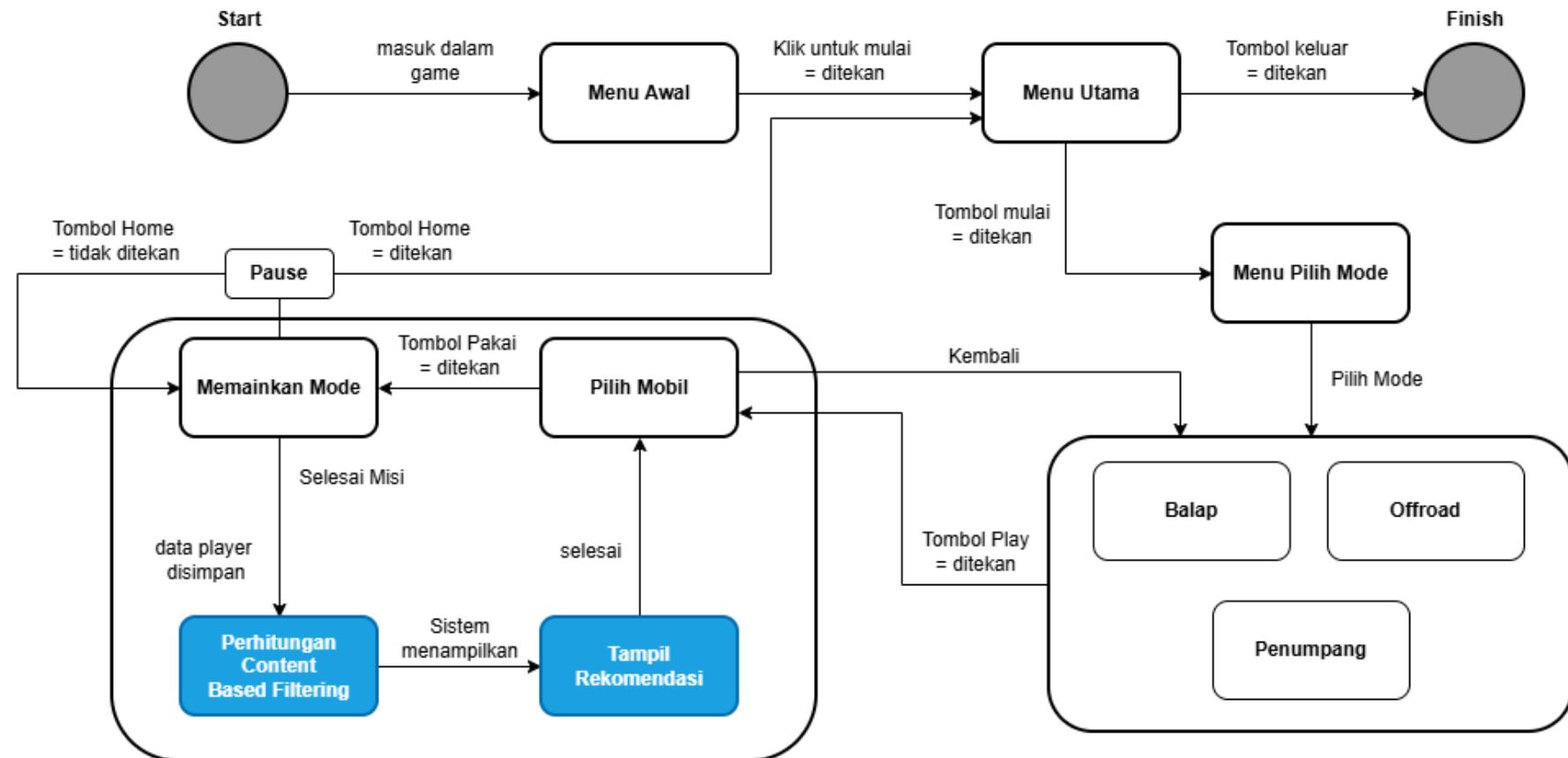
Rancangan antarmuka dan alur permainan yang telah disusun melalui storyboard ini menjadi dasar dalam proses pengembangan *game Perfect Car*. Seluruh tampilan direpresentasikan dalam bentuk wireframe untuk menggambarkan struktur, navigasi, serta interaksi utama yang akan dialami oleh pemain sebelum tahap implementasi dilakukan. Dengan adanya rancangan ini, pengembangan sistem dapat berjalan lebih terarah karena setiap elemen visual, fungsi, dan alur permainan telah ditetapkan secara sistematis. Rancangan tersebut selanjutnya diimplementasikan pada tahap pengembangan sehingga menghasilkan tampilan dan pengalaman bermain yang sesuai dengan konsep awal yang telah direncanakan.

### 3.2 Desain Sistem

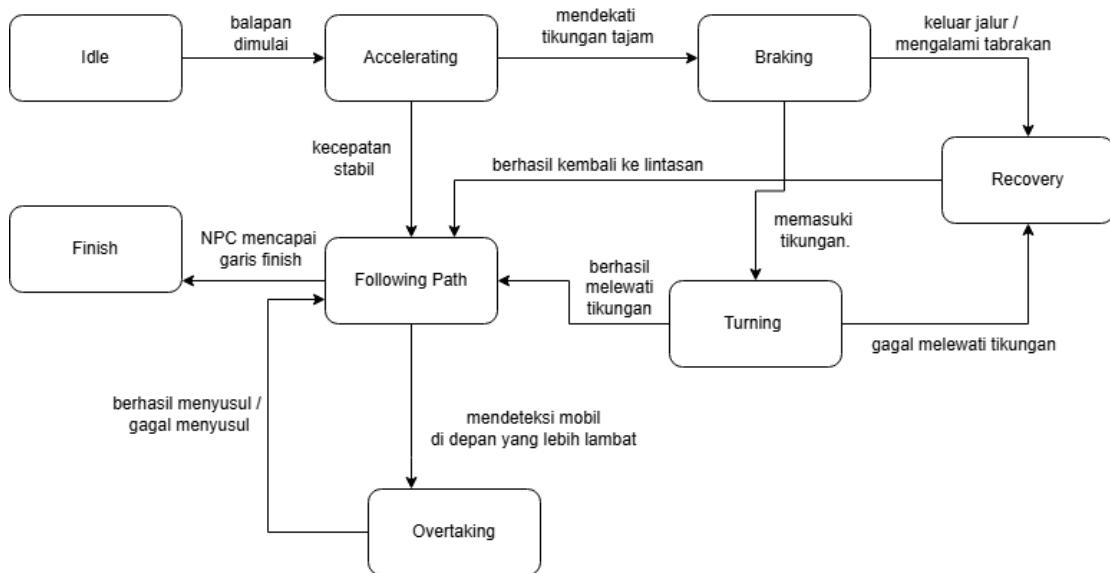
Desain sistem adalah proses merancang sistem yang digunakan untuk mendukung penelitian ini. Proses desain sistem dalam *game* “*Perfect Car*” ini mencakup pengembangan arsitektur, antarmuka, dan komponen sistem yang dirancang untuk mengintegrasikan metode *Content-Based Filtering* dalam rekomendasi kendaraan bagi pemain. Tahapan ini dilakukan setelah analisis sistem untuk memastikan metode yang digunakan dapat memberikan rekomendasi kendaraan yang relevan dan sesuai dengan Kemiripan Spesifikasi Kendaraan.

Untuk memudahkan alur kerja sistem rekomendasi dalam *game* “*Perfect Car*”, diagram alur sistem dirancang menggunakan *Finite State Machine* (FSM). Diagram ini menggambarkan setiap tahapan interaksi pemain, mulai dari pemilihan kendaraan hingga pemberian rekomendasi mobil yang paling sesuai, baik secara manual maupun melalui sistem rekomendasi otomatis. Dengan menggunakan FSM, alur pemrosesan rekomendasi dapat divisualisasikan secara sistematis, memberikan pemahaman yang lebih jelas mengenai bagaimana pemain berinteraksi dengan fitur rekomendasi dalam *game*. Diagram alur sistem ini dapat dilihat pada Gambar 3.2.

Selain FSM yang digunakan untuk menggambarkan sistem rekomendasi, terdapat juga FSM yang mengatur perilaku *Non-Player Character* (NPC) dalam permainan. NPC memiliki peran penting dalam balapan, seperti mempercepat kendaraan, mengerem saat menghadapi tikungan tajam, melakukan manuver untuk menyalip kendaraan lain, dan memulihkan posisi setelah keluar lintasan atau mengalami tabrakan. Diagram FSM NPC yang menggambarkan transisi antara berbagai keadaan NPC dapat dilihat pada Gambar 3.3.



Gambar 3.2 Diagram Finite State Machine (FSM) Sistem Game



Gambar 3.3 Diagram Sistem Finite State Machine (FSM) NPC

Gambar 3.2 menggambarkan aliran interaksi pemain dengan sistem dalam permainan. Sistem dimulai dengan pemain masuk ke dalam *game*, dan akan melalui berbagai menu untuk memilih mode permainan yang ingin dimainkan. Pada Menu Awal, pemain memilih apakah mereka ingin memulai permainan atau keluar. Jika memilih untuk memulai permainan, pemain akan diarahkan ke Menu Utama dan kemudian ke Menu Pilih Mode, di mana mereka bisa memilih mode permainan, seperti Balap, *Offroad* dan Penumpang.

Setelah memilih mode permainan, pemain akan memilih mobil yang akan digunakan dalam balapan. Setelah memilih mobil, pemain akan memulai balapan dengan menekan tombol *Play* dan kemudian memasuki tahap Memainkan Mode. Selama balapan, pemain akan melewati beberapa tahapan, termasuk Perhitungan *Content-Based Filtering* untuk memberikan rekomendasi mobil lebih lanjut

berdasarkan kinerja pemain, dan Tampil Rekomendasi yang menampilkan saran kendaraan yang mungkin lebih sesuai setelah balapan selesai. Jika pemain ingin menghentikan permainan, mereka bisa kembali ke Menu Utama dengan menekan tombol *Home*.

Gambar 3.3 menggambarkan perilaku NPC (*Non-Player Character*) selama balapan. Perilaku NPC dimulai dengan *Idle*, yaitu keadaan awal di mana NPC tidak melakukan apa-apa sebelum balapan dimulai. Ketika balapan dimulai, NPC berpindah ke keadaan *Accelerating*, mempercepat mobil hingga mencapai kecepatan yang stabil. Jika NPC mendekati tikungan tajam, mereka akan berpindah ke *Braking* untuk mengurangi kecepatan dan menghindari kecelakaan.

Setelah melewati tikungan, NPC beralih ke *Following Path*, yaitu mengikuti jalur balapan dengan kecepatan stabil. Jika NPC berhasil melewati kendaraan lain yang lebih lambat, mereka akan memasuki keadaan *Overtaking* untuk menyalip. Jika NPC gagal menyalip, mereka kembali ke keadaan *Following Path*. Jika NPC mendekati tikungan atau hambatan lainnya, mereka akan beralih ke *Turning* dan mencoba mengatasi tikungan. Jika NPC berhasil, mereka kembali ke *Following Path*. Namun, jika NPC gagal melewati tikungan atau keluar dari lintasan, mereka akan masuk ke keadaan *Recovery*, di mana NPC memulihkan posisi mereka untuk kembali ke lintasan.

### 3.3 Rancangan Perhitungan *Content-Based Filtering*

Bagian ini akan membahas langkah-langkah perhitungan manual *Content-Based Filtering* untuk mendapatkan rekomendasi mobil yang sesuai dengan

pengguna. Kita akan menggunakan contoh data yang sesuai dengan table item dan Fitur.

### 3.3.1 Item

Dalam game “*Perfect Car*”, pemain menggunakan berbagai jenis kendaraan sebagai alat utama untuk menyelesaikan mode permainan yang tersedia di berbagai jenis lintasan. Setiap kendaraan dirancang dengan karakteristik unik yang memengaruhi performa pemain dalam menyelesaikan tantangan, seperti lintasan jalanan perkotaan, pegunungan, sirkuit balap, atau medan *off-road*. Pemilihan kendaraan yang tepat sangat penting untuk meningkatkan efektivitas dalam menjalankan mode permainan, di mana fitur seperti kecepatan, akselerasi, *Handling*, suspensi, dan kapasitas berperan dalam menentukan kesuksesan pemain.

Tabel 3.2 Tabel Item

Simbol	Arti
A1	Toyota Vios
A2	Mazda 3
A3	Honda Civic Type R
A4	MG 5 GT
A5	Audi R8
A6	Toyota GR Supra
A7	Lamborghini Aventador
A8	Nissan GT-R
A9	Toyota Fortuner
A10	Toyota Rush
A11	Ford Everest
A12	Ford Ranger Raptor
A13	Honda BR-V
A14	Jeep Wrangler
A15	Suzuki Jimny
A16	Toyota Hilux
A17	Daihatsu Gran Max
A18	Toyota HiAce
A19	Mercedes-Benz Sprinter
A20	DFSK Gelora
A21	Toyota Innova
A22	Toyota Avanza
A23	Daihatsu Sigra
A24	Daihatsu Ayla

### 3.3.2 Fitur

Fitur digunakan untuk menilai efektivitas setiap kendaraan dalam sistem rekomendasi. Fitur - fitur ini menjadi dasar penilaian dalam *Content-Based Filtering* (CBF) dan mencakup berbagai faktor seperti kecepatan, akselerasi, *Handling*, suspensi, dan kapasitas kendaraan. Fitur ini digunakan untuk mengevaluasi bagaimana setiap kendaraan dapat membantu pemain dalam menghadapi berbagai tantangan di dalam *game*, seperti lintasan jalanan perkotaan, sirkuit balap atau medan *off-road*.

Tabel 3.3 Tabel Fitur

Simbol	Nama Fitur
F1	Akselerasi
F2	Kecepatan Maksimal
F3	<i>Handling</i>
F4	Suspensi
F5	Kapasitas

Fitur di atas akan digunakan untuk memberikan penilaian berdasarkan situasi yang dihadapi oleh pemain selama permainan. Sebagai contoh, Akselerasi (F1) mengacu pada seberapa cepat kendaraan dapat mencapai kecepatan optimal, yang sangat penting dalam skenario seperti mulai balapan atau melewati tikungan dengan cepat. Kecepatan Maksimal (F2) menentukan batas tertinggi kecepatan kendaraan, yang menjadi faktor utama dalam lintasan lurus atau sirkuit balap yang membutuhkan kecepatan tinggi. *Handling* (F3) menggambarkan kemampuan kendaraan dalam bermanuver dan stabilitas saat melewati tikungan tajam atau

jalanan sempit, sehingga kendaraan dengan *Handling* yang baik akan lebih mudah dikendalikan dalam tantangan teknis. Suspensi (F4) menilai kemampuan kendaraan dalam menyerap guncangan saat melewati berbagai medan, seperti jalanan berlubang, trek *off-road*, atau jalur yang tidak rata, yang dapat memengaruhi kestabilan dan kenyamanan kendaraan. Kapasitas (F5) mengacu pada jumlah penumpang atau kapasitas kargo yang dapat diangkut oleh kendaraan, yang menjadi faktor penting dalam mode permainan yang memerlukan transportasi atau perjalanan dengan lebih dari satu karakter. Fitur ini akan memastikan bahwa kendaraan yang direkomendasikan sesuai dengan kebutuhan pemain berdasarkan tantangan yang dihadapi di dalam *game*. Penentuan interval kategori untuk setiap fitur di bawah ini (Tabel 3.3 hingga Tabel 3.7) disusun berdasarkan distribusi data spesifikasi dari 24 kendaraan yang tersedia di dalam *game*. Rentang nilai dari setiap fitur dipetakan berdasarkan nilai minimum dan maksimum kendaraan, kemudian dikelompokkan secara proporsional. Pendekatan ini dilakukan untuk menjaga keseimbangan permainan (*game balancing*) dan memastikan distribusi klasifikasi kendaraan yang merata dalam simulasi.

a. Akselerasi (F1)

Akselerasi diukur berdasarkan durasi yang dibutuhkan kendaraan untuk mencapai kecepatan 0-100 km/jam. Semakin singkat waktu yang diperlukan, semakin baik performa akselerasi kendaraan tersebut. Penilaian akselerasi digunakan untuk mengidentifikasi kendaraan yang lebih responsif dalam percepatan, yang dapat memberikan pengalaman berkendara yang lebih optimal.

Tabel 3.4 Interval dan Kategori Performa Akselerasi Kendaraan

<b>Fitur</b>	<b>Interval Akselerasi (detik)</b>	<b>Contoh Mobil</b>
Akselerasi	$\leq 4.9$	Nissan GT-R, Lamborghini Aventador
	5.0 – 8.9	Audi R8, Honda Civic Type R
	9.0 – 11.9	Toyota Innova, Toyota Fortuner
	12.0 – 13.9	Toyota Rush, Daihatsu Sigra
	$\geq 14.0$	Daihatsu Gran Max, DFSK Gelora

### b. Kecepatan Maksimal (F2)

Kecepatan maksimal mengacu pada batas tertinggi kecepatan yang dapat dicapai oleh kendaraan dalam kondisi optimal. Semakin tinggi nilai kecepatan maksimal, semakin baik performa kendaraan dalam mencapai kecepatan puncak di lintasan lurus. Parameter ini digunakan untuk menilai seberapa cepat kendaraan dapat melaju tanpa hambatan, yang berpengaruh terhadap pemilihan kendaraan dalam mode permainan tertentu.

Tabel 3.5 Interval dan Kategori Performa Kecepatan Maksimal

<b>Fitur</b>	<b>Interval Kecepatan Maksimal (km/jam)</b>	<b>Contoh Mobil</b>
Kecepatan Maksimal	$\geq 300$	Lamborghini Aventador, Nissan GT-R
	220 – 299	Toyota GR Supra, Audi R8
	180 – 219	Mazda 3, Ford Everest
	160 – 179	Toyota Avanza, Toyota HiAce
	$\leq 159$	Daihatsu Gran Max, Suzuki Jimny

### c. Handling (F3)

Tabel 3.6 Interval dan Kategori Stabilitas Handling Kendaraan

<b>Fitur</b>	<b>Interval Handling (Manuver &amp; Stabilitas)</b>	<b>Contoh Mobil</b>
Handling	9	Lamborghini Aventador, Audi R8
	8	Honda Civic Type R, Mazda 3
	7	Toyota Vios, Ford Ranger Raptor
	6	Toyota Innova, Toyota Avanza
	$\leq 5$	Daihatsu Gran Max, Mercedes-Benz Sprinter

*Handling* mengacu pada kemampuan kendaraan dalam bermanuver, menjaga stabilitas, dan mempertahankan kendali saat melaju di berbagai kondisi jalan. Faktor ini dipengaruhi oleh desain suspensi, distribusi berat, serta sistem kemudi kendaraan. Semakin baik *Handling* sebuah kendaraan, semakin stabil dan responsif kendaraan saat berbelok atau menghadapi tikungan tajam.

#### d. Suspensi (F4)

Suspensi berperan dalam menyerap guncangan dan menjaga stabilitas kendaraan saat melintasi berbagai jenis medan, baik di jalan raya maupun *off-road*. Sistem suspensi yang baik memungkinkan kendaraan mempertahankan keseimbangan dan kenyamanan berkendara, terutama di permukaan yang tidak rata. Penilaian suspensi dalam sistem rekomendasi didasarkan pada kombinasi suspensi depan dan belakang, yang berkontribusi terhadap ground clearance kendaraan sebagai indikator kemampuan mobil dalam menghadapi medan kasar.

Tabel 3.7 Interval dan Kategori Performa Suspensi Kendaraan

Fitur	Interval Suspensi	Contoh Mobil
Suspensi	$\geq 215$	Jeep Wrangler, Ford Ranger Raptor
	200 – 214	Toyota Fortuner, Toyota HiAce
	180 – 199	Toyota Innova, Toyota GR Supra
	160 – 179	Honda Civic Type R, Mazda 3
	$\leq 159$	Toyota Vios, MG 5 GT

#### e. Kapasitas (F5)

Tabel 3.8 Interval dan Kategori Kapasitas Penumpang Kendaraan

Fitur	Interval Kapasitas	Contoh Mobil
Kapasitas	$\geq 15$	Mercedes-Benz Sprinter, Toyota HiAce
	11 – 14	Daihatsu Gran Max, DFSK Gelora
	6 – 10	Toyota Innova, Toyota Fortuner
	4 – 5	Honda Civic, Mazda 3
	$\leq 2$	Lamborghini Aventador, Toyota GR Supra

Kapasitas mengacu pada jumlah penumpang yang dapat ditampung oleh kendaraan secara optimal sesuai dengan desain dan fungsinya. Faktor ini berpengaruh pada kenyamanan, fungsi kendaraan, serta kemampuannya dalam membawa penumpang. Kendaraan dengan kapasitas lebih besar umumnya dirancang untuk kebutuhan keluarga atau angkutan komersial, sementara kendaraan dengan kapasitas kecil lebih cocok untuk penggunaan pribadi atau performa yang lebih lincah.

### 3.3.3 *Filtering* Kendaraan per Mode

Dalam *game Perfect Car*, setiap mode permainan memiliki karakteristik tantangan yang berbeda, seperti kebutuhan kecepatan pada mode Balap, kemampuan menghadapi medan tidak rata pada mode *Offroad*, dan kapasitas penumpang pada mode Penumpang. Oleh karena itu, tidak semua kendaraan dapat digunakan pada setiap mode. Untuk memastikan pemilihan kendaraan yang realistik dan sesuai konteks *gameplay*, diterapkan proses *filtering* kendaraan sebelum tahap perhitungan kesamaan dilakukan.

*Filtering* dilakukan dengan mengelompokkan kendaraan berdasarkan kesesuaian spesifikasi utama terhadap kebutuhan masing-masing mode permainan. Hanya kendaraan yang memenuhi karakteristik minimal mode tersebut yang akan ditampilkan kepada pemain, sementara kendaraan lain tidak akan muncul pada daftar pemilihan. Dengan cara ini, sistem secara otomatis membatasi ruang perhitungan *similarity* hanya pada kendaraan yang relevan, sehingga rekomendasi yang dihasilkan menjadi lebih fokus dan sesuai konteks.

Proses *filtering* dilakukan berdasarkan pertimbangan teknis sebagai berikut:

### 1. Mode Balap

Mode ini menuntut akselerasi dan kecepatan maksimal yang tinggi.

Kendaraan yang termasuk dalam kategori sedan, sport, dan supercar dimasukkan dalam mode ini karena memiliki performa yang sesuai untuk lintasan berkecepatan tinggi.

### 2. Mode *Offroad*

Mode ini membutuhkan stabilitas suspensi dan kemampuan medan kasar. Kendaraan seperti SUV, Jeep, Pickup, dan kendaraan berkemampuan *Offroad* lain dimasukkan ke mode ini karena memiliki ground clearance dan suspensi yang mendukung medan tidak rata.

### 3. Mode Penumpang

Mode ini memerlukan kapasitas penumpang yang memadai. Kendaraan MPV, minibus, serta kendaraan komersial penumpang termasuk pada mode ini karena mampu mengangkut jumlah penumpang sesuai kebutuhan misi.

Pemetaan kendaraan ke masing-masing mode ditunjukkan dalam Tabel 3.8 berikut.

Tabel 3.9 Pemetaan Kendaraan

<b>Mode</b>	<b>Mobil yang Digunakan</b>	<b>Alasan Pemilihan</b>
<b>Balap</b>	Toyota Vios, Mazda 3, Honda Civic, MG 5 GT, Audi R8, Toyota GR Supra, Lamborghini Aventador, Nissan GT-R	Kendaraan dengan akselerasi cepat, kecepatan maksimal tinggi, serta <i>Handling</i> yang responsif sehingga ideal digunakan pada lintasan berkecepatan tinggi.
<b>Offroad</b>	Toyota Fortuner, Toyota Rush, Ford Everest, Ford Ranger Raptor, Jeep Wrangler, Suzuki Jimny, Toyota Hilux	Kendaraan bertipe SUV, Jeep, dan Pickup yang memiliki suspensi kuat, ground clearance tinggi, serta stabilitas yang baik untuk medan berbatu, tanah, dan jalur tidak rata.

<b>Mode</b>	<b>Mobil yang Digunakan</b>	<b>Alasan Pemilihan</b>
<b>Penumpang</b>	Daihatsu Gran Max, Toyota HiAce, Mercedes-Benz Sprinter, DFSK Gelora, Toyota Innova, Toyota Avanza, Daihatsu Sigra, Toyota Ayla, Honda BR-V	Kendaraan MPV, minibus, dan kendaraan penumpang dengan kapasitas yang sesuai untuk misi pengantaran sehingga stabil digunakan di lingkungan perkotaan.

Dengan diterapkannya *filtering* ini, setiap mode hanya berisi daftar kendaraan yang telah sesuai karakteristiknya. Tahap ini membuat proses rekomendasi lebih akurat dan cocok, sebab kendaraan yang dihitung *similarity*-nya sudah memiliki konteks fungsional yang sama. *Filtering* juga memastikan bahwa hasil rekomendasi tidak akan menyarankan kendaraan yang tidak sesuai dengan mode permainan, misalnya sedan direkomendasikan pada tantangan *Offroad* atau supercar direkomendasikan untuk misi mengantar penumpang.

### 3.3.4 Matriks Fitur Kendaraan (*Item-Fitur Matrix*)

*Perfect Car*, setiap kendaraan direpresentasikan sebagai sebuah item dengan beberapa fitur utama. Matriks fitur kendaraan ini digunakan untuk menyusun profil kendaraan berdasarkan karakteristik fitur. Fitur tersebut meliputi akselerasi (detik 0–100 km/jam), kecepatan maksimal (km/jam), *Handling* (skor), suspensi atau *ground clearance* (mm), serta kapasitas penumpang (jumlah orang). Data mentah ini disusun dalam bentuk matriks fitur kendaraan sehingga dapat digunakan untuk menyusun profil setiap kendaraan. Data spesifikasi teknis kendaraan (seperti akselerasi, kecepatan maksimal dan kapasitas penumpang) dikumpulkan dan divalidasi berdasarkan basis data dari portal otomotif Oto.com dan CarsGuide (2024), yang kemudian disesuaikan untuk kebutuhan simulasi.

Tabel berikut menunjukkan matriks fitur kendaraan yang digunakan dalam penelitian ini:

Tabel 3.10 Matriks Fitur Kendaraan

Item (A)	Fitur (F)				
	(F1)	(F2)	(F3)	(F4)	(F5)
A1	11.2	170	7	150	5
A2	8.5	210	8	160	5
A3	7.5	220	8	165	5
A4	8.0	210	7	160	5
A5	6.9	240	9	175	5
A6	4.3	250	9	180	2
A7	2.9	350	9	185	2
A8	2.7	328	9	185	4
A9	11.5	190	6	200	7
A10	12.0	175	6	190	7
A11	10.5	190	6	205	7
A12	7.5	210	7	220	5
A13	10.2	185	7	185	7
A14	6.5	180	6	225	5
A15	12.3	145	7	215	4
A16	10.8	185	6	210	5
A17	14.8	140	4	190	11
A18	14.0	160	4	200	15
A19	13.5	170	5	205	15
A20	14.3	150	4	195	11
A21	11.0	180	6	185	7
A22	12.5	175	6	180	7
A23	13.5	165	5	175	7
A24	13.0	165	6	170	5

Matriks fitur kendaraan pada Tabel 3.8 di atas merupakan data mentah yang digunakan sebagai dasar perhitungan kesamaan antar kendaraan. Sebelum dilakukan perhitungan menggunakan metode *Cosine Similarity*, seluruh nilai fitur pada tabel tersebut terlebih dahulu dinormalisasi dengan metode Min-Max Normalization agar memiliki skala yang sebanding (0–1).

### 3.3.5 Normalisasi Matriks Fitur Kendaaraan (*Item-Fitur Matrix*)

Setiap fitur kendaraan memiliki satuan pengukuran yang berbeda, misalnya akselerasi dalam detik, kecepatan dalam km/jam, suspensi dalam milimeter, dan

kapasitas dalam jumlah kapasitas. Perbedaan skala ini menyebabkan perlunya proses normalisasi agar nilai setiap fitur dapat dibandingkan secara proporsional. Normalisasi dilakukan untuk menyamakan skala antar fitur kendaraan sehingga perhitungan *Cosine Similarity* tidak dipengaruhi oleh perbedaan skala unit dari setiap fitur. Metode yang digunakan adalah Min-Max Normalization yang mengubah nilai data ke dalam rentang 0 hingga 1 berdasarkan nilai minimum dan maksimum dalam dataset (Chamidah et al., 2012). Dengan demikian, semua fitur kendaraan akan berada pada skala yang sebanding. Persamaan normalisasi ditunjukkan pada Persamaan (3.1):

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

Keterangan:

- $X$  = nilai asli dari fitur kendaraan.
- $X'$  = nilai hasil normalisasi.
- $X_{max}$  = nilai maksimum dalam fitur tersebut.
- $X_{min}$  = nilai minimum dalam fitur tersebut.

Melalui persamaan tersebut, setiap fitur kendaraan diubah menjadi nilai antara 0 hingga 1 tanpa mengubah perbandingan relatif antar item. Normalisasi ini juga mendukung proses *mode-based filtering*, karena kendaraan yang berada dalam mode permainan yang sama tetap mempertahankan karakteristik relatifnya setelah distandarisasi. Hasil proses normalisasi ditunjukkan pada Tabel 3.9.

Tabel 3.11 Normalisasi Matriks Fitur Kendaraan

Item (A)	Fitur (F)				
	(F1)	(F2)	(F3)	(F4)	(F5)
A1	0.2975	0.1429	0.6	0.0000	0.2308
A2	0.5207	0.3333	0.8	0.1333	0.2308
A3	0.6033	0.3810	0.8	0.2000	0.2308
A4	0.5620	0.3333	0.6	0.1333	0.2308
A5	0.6529	0.4762	1.0	0.3333	0.2308
A6	0.8678	0.5238	1.0	0.4000	0.0000
A7	0.9835	1.0000	1.0	0.4667	0.0000

Item (A)	Fitur (F)				
	(F1)	(F2)	(F3)	(F4)	(F5)
A8	1.0000	0.8952	1.0	0.4667	0.1538
A9	0.2727	0.2381	0.4	0.6667	0.3846
A10	0.2314	0.1667	0.4	0.5333	0.3846
A11	0.3554	0.2381	0.4	0.7333	0.3846
A12	0.6033	0.3333	0.6	0.9333	0.2308
A13	0.3802	0.2143	0.6	0.4667	0.3846
A14	0.6860	0.1905	0.4	1.0000	0.2308
A15	0.2066	0.0238	0.6	0.8667	0.1538
A16	0.3306	0.2143	0.4	0.8000	0.2308
A17	0.0000	0.0000	0.0	0.5333	0.6923
A18	0.0661	0.0952	0.0	0.6667	0.9231
A19	0.1074	0.1429	0.2	0.7333	1.0000
A20	0.0413	0.0476	0.0	0.6000	0.6923
A21	0.3140	0.1905	0.4	0.4667	0.3846
A22	0.1901	0.1667	0.4	0.4000	0.3846
A23	0.1074	0.1190	0.2	0.3333	0.3846
A24	0.1488	0.1190	0.4	0.2667	0.2308

### 3.3.6 Perhitungan *Similarity* (Kemiripan)

Tahap berikutnya dalam sistem rekomendasi adalah menghitung *similarity* (kemiripan) antar kendaraan. Perhitungan ini dilakukan untuk menemukan kendaraan yang memiliki spesifikasi paling mirip berdasarkan fitur yang telah dianalisis. Dalam sistem rekomendasi berbasis *Content-Based Filtering* (CBF), kendaraan yang memiliki kemiripan tinggi cenderung memiliki karakteristik yang hampir sama dalam hal performa, sehingga lebih mungkin untuk direkomendasikan kepada pemain setelah menyelesaikan mode permainan. Salah satu metode yang digunakan dalam sistem rekomendasi ini adalah *Cosine Similarity*, yang mengukur derajat kemiripan antara dua kendaraan berdasarkan fitur seperti kecepatan maksimal, akselerasi, *Handling*, suspensi, dan kapasitas.

$$\text{Cosine Similarity } (A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} \quad (3.2)$$

- $A \cdot B$  adalah hasil perkalian dot *product* antara vektor fitur kendaraan A dan kendaraan B.
- $\|A\|$  adalah norma (*magnitude*) dari vektor kendaraan A, dihitung sebagai:

$$\|A\| = \sqrt{\sum_{i=1}^n A_i^2} \quad (3.3)$$

- $\|B\|$  adalah norma dari vektor kendaraan B, dihitung sebagai:

$$\|B\| = \sqrt{\sum_{i=1}^n B_i^2} \quad (3.4)$$

Contoh Perhitungan *Cosine Similarity* pada Mode Balap

Sebagai contoh, akan dihitung nilai *similarity* antara Mazda 3 (A2) dan Honda Civic (A3) pada mode Balap. Nilai vektor fitur yang digunakan adalah nilai hasil normalisasi dari Tabel 3.9, yaitu:

Item a (A2) :  $A [0,5207, 0,3333, 0,8000, 0,1333, 0,2308]$

Item b (A3) :  $B [0,6033, 0,3810, 0,8000, 0,2000, 0,2308]$

Hitung dot product  $A \cdot B$  :

$$\begin{aligned} A \cdot B &= (0,5207 \times 0,6033) + (0,3333 \times 0,3810) + (0,8000 \times 0,8000) + (0,1333 \times \\ &\quad 0,2000) + (0,2308 \times 0,2308) \\ &= 1,1610 \end{aligned}$$

$$\begin{aligned} \|A\| &= \sqrt{(0,5207)^2 + (0,3333)^2 + (0,8000)^2 + (0,1333)^2 + (0,2308)^2} \\ &= 1,046 \end{aligned}$$

$$\|B\| = \sqrt{(0,6033)^2 + (0,3810)^2 + (0,8000)^2 + (0,2000)^2 + (0,2308)^2}$$

$$= 1,115$$

$$\text{Cosine Similarity } (A2, A3) = \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}|| \times ||\mathbf{B}||} = \frac{1,1610}{1,046 \times 1,115} = 0,9962$$

Perhitungan Cosine Similarity item a (A9) dan item b (A11) pada mode *Offroad*

Item a (A9) :  $A [0.2727, 0.2381, 0.4, 0.6667, 0.3846]$

Item b (A11) :  $B [0.3544, 0.2381, 0.4, 0.7333, 0.3846]$

$$\begin{aligned} \mathbf{A} \cdot \mathbf{B} &= (0.2727 \times 0.3544) + (0.2381 \times 0.2381) + (0.4 \times 0.4) + (0.6667 \times 0.7333) \\ &\quad + (0.3846 \times 0.3846) \\ &= 0.0969 + 0.0567 + 0.16 + 0.4889 + 0.1479 \\ &= 0.9504 \end{aligned}$$

$$\begin{aligned} \|\mathbf{A}\| &= \sqrt{(0.2727)^2 + (0.2381)^2 + (0.4)^2 + (0.6667)^2 + (0.3846)^2} \\ &= \sqrt{(0.0743) + (0.0567) + (0.16) + (0.4444) + (0.1479)} \\ &= \sqrt{0.8832} = 0.9390 \end{aligned}$$

$$\begin{aligned} \|\mathbf{B}\| &= \sqrt{(0.3544)^2 + (0.2381)^2 + (0.4)^2 + (0.7333)^2 + (0.3846)^2} \\ &= \sqrt{(0.1263) + (0.0567) + (0.16) + (0.5377) + 0.1473} \\ &= \sqrt{1.0286} = 1.0142 \end{aligned}$$

$$\text{CosSim}(A9, A11) = \frac{\mathbf{A} \cdot \mathbf{B}}{||\mathbf{A}|| \times ||\mathbf{B}||} = \frac{0.9504}{0.9390 \times 1.0142} = \frac{0.9504}{0.9523} = 0.9979$$

Perhitungan Cosine Similarity item a (A18) dan item b (A19) pada mode Penumpang

Item a (A18) :  $A [0.0661, 0.0952, 0.2, 0.6667, 0.9231]$

Item b (A19) :  $B [0.1074, 0.1429, 0.2, 0.7333, 1.0]$

$$\begin{aligned} \mathbf{A} \cdot \mathbf{B} &= (0.0661 \times 0.1074) + (0.0952 \times 0.1429) + (0.2 \times 0.2) + (0.6667 \times 0.7333) \\ &\quad + (0.9231 \times 1.0) \end{aligned}$$

$$= 1.4727$$

$$\|A\| = \sqrt{(0.0661)^2 + (0.0952)^2 + (0.2)^2 + (0.6667)^2 + (0.9231)^2}$$

$$= \sqrt{1.3499} = 1.1625$$

$$\|B\| = \sqrt{(0.1074)^2 + (0.1429)^2 + (0.2)^2 + (0.7333)^2 + (1.0)^2}$$

$$= \sqrt{1.6096} = 1.2687$$

$$\textbf{CosSim}(A18, A19) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \times \|\mathbf{B}\|} = \frac{1.4727}{1.1625 \times 1.2687} = \frac{1.4727}{01.4741} = \mathbf{0.999}$$

Hasil perhitungan *Cosine Similarity* antar kendaraan dalam *game Perfect Car* menunjukkan sejauh mana setiap mobil memiliki kemiripan spesifikasi berdasarkan lima fitur utama: akselerasi (F1), kecepatan maksimal (F2), *Handling* (F3), suspensi (F4), dan kapasitas (F5). Berdasarkan hasil perhitungan *similarity* menggunakan metode *Cosine Similarity* pada ketiga mode permainan, terlihat bahwa pasangan kendaraan dalam setiap mode memiliki tingkat kemiripan yang sangat tinggi. Nilai *Cosine Similarity* berada pada rentang 0 hingga 1, di mana nilai mendekati 1 menunjukkan kemiripan yang tinggi dan nilai mendekati 0 menandakan perbedaan signifikan.

Pada mode Balap, Mazda 3 (A2) dan Honda Civic (A3) memiliki nilai *similarity* sebesar 0.9962, menunjukkan bahwa kedua kendaraan tersebut memiliki profil performa yang hampir identik, terutama pada fitur akselerasi, kecepatan maksimal, dan *Handling*. Kondisi ini menggambarkan bahwa kedua kendaraan memang sama-sama cocok digunakan pada lintasan berkecepatan tinggi.

Pada mode *Offroad*, Toyota Fortuner (A9) dan Ford Everest (A11) memperoleh nilai *similarity* sebesar 0.9979, yang menunjukkan tingkat kesamaan

yang sangat tinggi. Kedua kendaraan tersebut sama-sama memiliki suspensi yang kuat, ground clearance yang memadai, serta kapasitas yang serupa sehingga mampu melintasi medan tidak rata dengan stabil.

Sementara itu, pada mode Penumpang, Toyota HiAce (A18) dan Mercedes-Benz Sprinter (A19) memiliki nilai *similarity* sebesar 0.9990, yang merupakan nilai tertinggi di antara semua pasangan yang diuji. Hal ini terjadi karena keduanya memiliki kapasitas penumpang yang besar, stabilitas yang baik, serta nilai suspensi dan *Handling* yang berdekatan, sehingga keduanya merupakan pilihan ideal untuk misi pengangkutan penumpang.

Secara keseluruhan, hasil perhitungan *similarity* menunjukkan bahwa sistem rekomendasi mampu mengidentifikasi kendaraan dengan karakteristik paling mirip terhadap kendaraan yang digunakan pemain. Proses *filtering* yang dilakukan sebelum perhitungan *similarity* berperan penting dalam meningkatkan akurasi rekomendasi, karena kendaraan yang dibandingkan sudah sesuai konteks mode permainan. Dengan demikian, rekomendasi yang diberikan tidak hanya relevan secara spesifikasi, tetapi juga selaras dengan kebutuhan mode permainan yang sedang dijalankan.

### 3.4 Desain Pengujian Sistem

Untuk mengevaluasi efektivitas sistem rekomendasi kendaraan dalam *game Perfect Car*, dilakukan uji coba yang melibatkan sejumlah pengguna sebagai responden. Pengujian ini bertujuan untuk menilai sejauh mana sistem rekomendasi berbasis *Cosine Similarity* dapat membantu pemain dalam memilih kendaraan yang sesuai dengan kebutuhan mereka. Penelitian ini juga menerapkan pendekatan User-

Centric Evaluation untuk mengukur tingkat akurasi rekomendasi (Knijnenburg et al., 2012). Metode ini berfokus pada persepsi subjektif pengguna terhadap relevansi kendaraan yang direkomendasikan sistem. Akurasi dihitung berdasarkan persentase kesesuaian antara rekomendasi sistem dengan penilaian responden (Gaudioso, 2008). Metode evaluasi yang digunakan adalah *System Usability Scale (SUS)*. Metode *System Usability Scale (SUS)* digunakan dalam penelitian ini untuk mengevaluasi tingkat kegunaan (*Usability*) dari sistem rekomendasi kendaraan pada *game Perfect Car*. SUS pertama kali diperkenalkan oleh John Brooke (1986) dan hingga kini menjadi salah satu instrumen standar yang banyak digunakan karena sederhana, reliabel, serta efektif untuk mengukur persepsi pengguna terhadap sistem interaktif (Lewis, 2018). Metode *System Usability Scale (SUS)* adalah sebuah standar evaluasi yang mengukur tingkat kegunaan suatu sistem berdasarkan respons pengguna melalui kuesioner (Ulfa, 2021).

Tabel 3.12 Daftar Pernyataan Kuesioner *System Usability Scale*

No	Jenis	Pernyataan
1	(+)	Saya berpikir akan sering menggunakan sistem ini dalam memilih mobil saat bermain.
2	(-)	Saya merasa sistem rekomendasi ini terlalu rumit untuk digunakan.
3	(+)	Saya merasa sistem rekomendasi ini mudah digunakan.
4	(-)	Saya membutuhkan bantuan teknis atau orang lain untuk menggunakan sistem ini.
5	(+)	Saya merasa sistem rekomendasi ini berjalan dengan baik.
6	(-)	Saya menilai banyak inkonsistensi dalam fitur rekomendasi ini.
7	(+)	Saya merasa kebanyakan orang akan mudah menggunakan fitur rekomendasi ini dengan cepat.
8	(-)	Saya menilai fitur rekomendasi ini sangat rumit untuk digunakan.
9	(+)	Saya merasa sangat percaya diri menggunakan fitur rekomendasi ini.
10	(-)	Saya perlu belajar terlebih dahulu sebelum dapat memahami fitur rekomendasi ini dengan baik.

Instrumen SUS terdiri dari 10 pertanyaan yang disusun dalam bentuk skala Likert 1 sampai 5, di mana nilai 1 menunjukkan sangat tidak setuju dan nilai 5

menunjukkan sangat setuju. Pertanyaan dalam kuesioner SUS terdiri dari lima pernyataan positif (+) dan lima pernyataan negatif (-) yang disusun secara bergantian. Tujuan dari pola ini adalah untuk mengurangi bias jawaban sehingga responden memberikan penilaian yang lebih objektif terhadap sistem yang diuji (Bangor dkk., 2008).

Dalam implementasinya, responden diminta untuk memainkan *game Perfect Car* sesuai skenario yang telah ditentukan, kemudian mengisi kuesioner SUS setelah menyelesaikan permainan. Setiap jawaban responden akan diubah menjadi skor dengan aturan sebagai berikut:

- Untuk pertanyaan positif (+), skor dihitung dengan rumus jawaban - 1.
- Untuk pertanyaan negatif (-), skor dihitung dengan rumus 5 - jawaban.
- Setiap pertanyaan memiliki skor antara 0 hingga 4.
- Jumlah skor dari semua pertanyaan dikalikan 2,5 untuk mendapatkan skor akhir SUS dalam rentang 0–100.

Interpretasi nilai SUS merujuk pada penelitian Bangor & Miller (2008), yaitu: skor di atas 70 menunjukkan sistem memiliki tingkat kegunaan yang baik, skor di atas 80 menunjukkan sistem sangat baik, sedangkan skor di bawah 60 menunjukkan sistem masih memerlukan perbaikan signifikan. Dengan demikian, melalui metode SUS ini, penelitian dapat memperoleh gambaran yang komprehensif mengenai sejauh mana sistem rekomendasi dalam *game Perfect Car* dianggap mudah digunakan, konsisten, dan memuaskan dari perspektif pemain.

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Implementasi Sistem**

Tahap implementasi merupakan proses penerapan metode *Content-Based Filtering* pada game *Perfect Car* sebagai sistem rekomendasi kendaraan yang mampu memberikan saran mobil paling sesuai dengan preferensi pemain. Sistem ini menganalisis kemiripan spesifikasi antar kendaraan menggunakan perhitungan *Cosine Similarity* berdasarkan lima fitur utama, yaitu akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas penumpang. Seluruh data kendaraan diolah dalam bentuk matriks fitur yang telah dinormalisasi agar setiap atribut memiliki skala yang seimbang. Setelah pemain memilih mode permainan dan menentukan preferensi fitur, sistem menghitung nilai kesamaan antar mobil untuk memperoleh hasil rekomendasi dengan nilai *similarity* tertinggi.

Implementasi sistem rekomendasi ini diintegrasikan langsung ke dalam mekanisme permainan yang dikembangkan menggunakan Unity Engine dengan bahasa pemrograman C#. Hasil perhitungan *similarity* secara keseluruhan ditampilkan melalui *console log* dalam bentuk peringkat kendaraan dari yang tertinggi hingga terendah, namun pada tampilan game hanya satu mobil dengan nilai tertinggi yang direkomendasikan kepada pemain. Mobil tersebut dianggap paling sesuai dengan gaya bermain dan kondisi mode yang dipilih. Dengan demikian, penerapan metode *Content-Based Filtering* pada game *Perfect Car* mampu menghasilkan rekomendasi kendaraan yang efisien, adaptif, dan meningkatkan pengalaman bermain secara interaktif.

## 4.2 Implementasi Antarmuka *Game*

Bagian ini menjelaskan hasil implementasi antarmuka *game* *Perfect Car* yang telah dikembangkan berdasarkan rancangan storyboard. Setiap tampilan yang disajikan merupakan bentuk akhir dari proses pengembangan, mulai dari tampilan awal permainan hingga tampilan *gameplay* pada masing-masing mode. Implementasi ini menunjukkan bagaimana rancangan sebelumnya direalisasikan ke dalam antarmuka visual dan interaksi yang dapat digunakan oleh pemain selama memainkan *game*.

### a. Tampilan Awal *Game*

Ketika pemain pertama kali masuk ke dalam *game*, pemain akan diberikan tampilan awal yang terdiri dari tombol mulai tengah bawah, terdapat teks "*Klik Untuk Mulai*" yang berfungsi sebagai tombol awal untuk memulai permainan. Saat pemain mengetuk area ini, mereka akan diarahkan ke menu utama pada *game*.



Gambar 4.1 Tampilan Awal *Game*

b. Tampilan *Main Menu*



Gambar 4.2 Tampilan *Main Menu*

Tampilan main menu ini menampilkan menu utama dalam *game* yang terdiri dari beberapa tombol seperti tombol mulai, tombol pengaturan, tombol tentang dan juga tombol keluar. Tombol ini memiliki fungsi yang berbeda, dimana tombol mulai berfungsi Ketika ditekan akan menuju menu pilih mode permainan dan dapat memilih mode permainan sesuai dengan preferensi mereka. Tombol pengaturan dapat mengakses pengaturan seperti menaikkan dan menurunkan volume sesuai dengan keingunan pemain. Tombol tentang ketika ditekan akan berisi informasi mengenai pengembang serta ucapan terima kasih kepada pemain yang telah memainkan *game Perfect Car*. Dan yang terakhir ada tombol keluar ketika ditekan maka permainan akan berakhir dan menuju desktop.

c. Tampilan Pilih Mode permainan

Menu ini akan muncul setelah pemain menekan tombol Mulai. Pemain akan memilih mode permainan yang ingin diselesaikan terlebih dahulu. Setelah

memilih mode, daftar kendaraan yang muncul telah difilter sesuai dengan karakteristik mode tersebut. Pemain hanya dapat memilih kendaraan yang kompatibel dengan mode, sehingga pemilihan menjadi lebih realistik dan sesuai konteks misi. Dalam menu ini pemain akan diberikan keleluasaan untuk memilih mode permainan yg mereka inginkan yaitu ada mode balap , *Offroad* dan Penumpang yang memiliki keunikan masing masing pada setiap lintasan.



Gambar 4.3 Tampilan Pilih Mode

#### d. Tampilan Pilih Mobil

Tampilan pilih mobil merupakan halaman yang digunakan pemain untuk menentukan kendaraan yang akan digunakan dalam mode permainan yang dipilih. Pada implementasinya, halaman ini menampilkan daftar mobil beserta spesifikasi utama seperti akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas. Pemain dapat melihat detail setiap mobil dan memilih kendaraan yang paling sesuai dengan preferensi maupun kebutuhan mode permainan. Tampilan ini merupakan realisasi dari rancangan antarmuka pemilihan mobil pada tahap perancangan, di mana fokus utamanya adalah memberikan

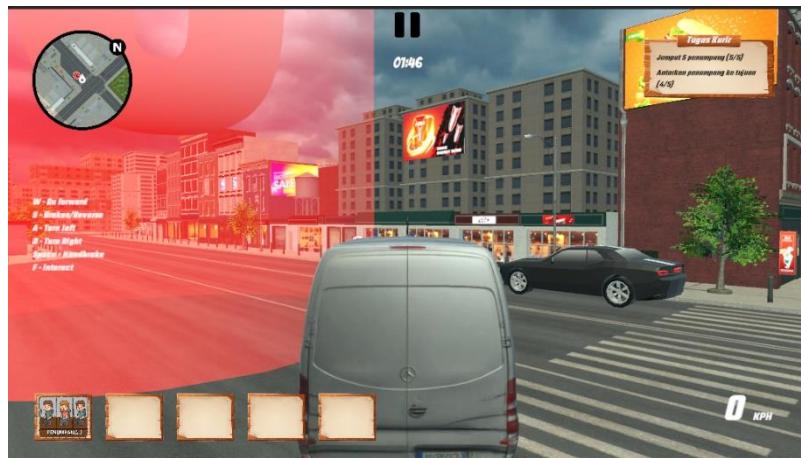
kemudahan bagi pemain dalam membandingkan mobil dan melakukan pemilihan secara intuitif.



Gambar 4.4 Tampilan Pilih Mobil

e. *Gameplay*

Tampilan permainan yang berlangsung adalah seperti berikut, pada gambar 4.5 menampilkan pemandangan dalam permainan, yang memperlihatkan lingkungan dalam *game*. Saat permainan dimulai, layar utama akan menampilkan kendaraan yang telah dipilih sebelumnya oleh pemain untuk menyelesaikan mode permainan yang sedang berlangsung. Lingkungan permainan akan menyesuaikan dengan jenis mode permainan yang dipilih, termasuk variasi medan seperti jalan raya, perkotaan, atau jalur *off-road*.



Gambar 4.5 Tampilan Gameplay

### 4.3 Implementasi Sistem Rekomendasi

Implementasi sistem rekomendasi pada *game Perfect Car* merupakan tahap penerapan metode *Content-Based Filtering* (CBF) yang telah dirancang pada bab sebelumnya. Tujuan utama dari implementasi ini adalah untuk menghasilkan sistem yang mampu memberikan rekomendasi kendaraan berdasarkan kemiripan spesifikasi antar mobil menggunakan perhitungan *Cosine Similarity*. Seluruh proses dilakukan secara otomatis tanpa memerlukan input preferensi dari pemain, sehingga sistem dapat menampilkan mobil yang paling sesuai dengan karakteristik kendaraan yang telah digunakan pada sesi permainan sebelumnya.

Pada tahap implementasi, sistem rekomendasi bekerja berdasarkan tiga proses utama, yaitu *filtering* kendaraan sesuai mode permainan, normalisasi fitur kendaraan menggunakan metode Min-Max Normalization, dan perhitungan kemiripan antar kendaraan menggunakan *Cosine Similarity*. Pendekatan ini dipilih karena setiap mode permainan memiliki karakteristik kendaraan yang berbeda, sehingga proses *filtering* pada masing-masing mode sudah cukup

merepresentasikan kebutuhan performa kendaraan dalam konteks *gameplay*. Ketika pemain memilih suatu mode permainan, sistem secara otomatis menampilkan daftar kendaraan yang relevan dengan karakteristik mode tersebut. *Filtering* ini memastikan bahwa kendaraan yang tidak sesuai dengan konteks mode permainan tidak diikutsertakan dalam proses perhitungan kemiripan, sehingga rekomendasi yang dihasilkan lebih terarah dan akurat.

Nilai fitur kendaraan yang meliputi akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas kemudian dinormalisasi menggunakan metode Min-Max Normalization untuk menyeragamkan skala antar fitur. Proses normalisasi ini diperlukan agar setiap fitur memiliki kontribusi yang seimbang dalam perhitungan *Cosine Similarity*. Setelah pemain menyelesaikan permainan dengan menggunakan salah satu kendaraan, sistem mengambil vektor fitur kendaraan tersebut sebagai profil acuan. Profil ini selanjutnya dibandingkan dengan seluruh kendaraan lain dalam mode yang sama untuk memperoleh nilai kemiripan menggunakan rumus *Cosine Similarity*. Kendaraan dengan nilai kemiripan tertinggi dianggap sebagai mobil yang paling mendekati karakteristik kendaraan yang digunakan pemain dan ditampilkan sebagai rekomendasi.

Hasil perhitungan *Cosine Similarity* disajikan dalam dua bentuk, yaitu daftar peringkat kemiripan pada *console log* Unity sebagai verifikasi internal dan satu rekomendasi utama yang ditampilkan pada antarmuka *game* setelah pemain menyelesaikan mode permainan. Integrasi proses ini dengan antarmuka *game* memastikan bahwa rekomendasi dapat ditampilkan secara real-time tanpa memerlukan interaksi tambahan dari pemain. Secara keseluruhan, implementasi

sistem rekomendasi berhasil menggabungkan mekanisme *filtering*, normalisasi, dan perhitungan kemiripan menjadi sebuah alur kerja yang konsisten dan mampu memberikan rekomendasi kendaraan yang sesuai dengan kondisi permainan dan spesifikasi kendaraan yang digunakan pemain.

#### 4.2.1 Tahapan Metode Content-Based *Filtering*

Metode *Content-Based Filtering* yang diterapkan dalam penelitian ini terdiri dari beberapa langkah utama yang dilakukan secara berurutan. Tahapan-tahapan tersebut dijelaskan melalui pseudocode berikut.

*Pseudocode 4.1 Penyusunan Matriks Fitur Kendaraan*

```

PROSEDUR BangunMatriksFitur()
    // Daftar mobil: N item
    // Fitur: F1=Akselerasi, F2=KecepatanMaks, F3=Handling,
    F4=Suspensi, F5=Kapasitas
    deklarasi matriksFitur : array[N][5]

    UNTUK i ← 0 S/D N-1 LAKUKAN
        matriksFitur[i][0] ← AmbilAkselerasi(mobil[i])
        matriksFitur[i][1] ← AmbilKecepatanMaks(mobil[i])
        matriksFitur[i][2] ← AmbilHandling(mobil[i])
        matriksFitur[i][3] ← AmbilSuspensi(mobil[i])
        matriksFitur[i][4] ← AmbilKapasitas(mobil[i])
    SELESAI

    KEMBALIKAN matriksFitur
END PROSEDUR

```

*Pseudocode 4.1* di atas merupakan tahap awal dari metode *Content-Based Filtering* yang digunakan pada sistem rekomendasi *Perfect Car*. Pada tahap ini, seluruh data kendaraan yang terdapat dalam *game* disusun dalam bentuk Item–Fitur Matrix yang berisi lima atribut utama: akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas penumpang. Setiap kendaraan direpresentasikan sebagai vektor numerik berdasarkan nilai dari masing-masing fitur, sehingga dapat dilakukan perhitungan kesamaan antar mobil secara matematis.

*Pseudocode 4.2 Filtering Kendaraan Berdasarkan Mode Permainan*

```

PROSEDUR FilterKendaraanBerdasarkanMode (modePermainan)
// Input : modePermainan (Balap / Offroad / Penumpang)
// Output : daftarKandidat = daftar mobil yang sesuai mode
    deklarasi daftarKandidat : list

    UNTUK i ← 0 S/D jumlahMobil-1 LAKUKAN
        JIKA Kategori(mobil[i]) = modePermainan MAKA
            TambahKeDaftar(daftarKandidat, mobil[i])
        ENDIF
    SELESAI

    KEMBALIKAN daftarKandidat
END PROSEDUR

    KEMBALIKAN matriksFitur
END PROSEDUR

```

Pseudocode 4.2 menggambarkan proses *filtering* kendaraan berdasarkan mode permainan yang dipilih oleh pemain. Setelah mode permainan ditentukan, misalnya Balap, *Offroad*, atau Penumpang, sistem hanya akan memilih kendaraan-kendaraan yang kategori atau tipenya sesuai dengan mode tersebut. Proses ini dilakukan dengan memeriksa atribut kategori setiap mobil melalui fungsi *Kategori(mobil[i])* dan menambahkan mobil yang sesuai ke dalam *daftarKandidat*. Dengan adanya *filtering* ini, kendaraan yang tidak relevan, seperti mobil sport pada mode *Offroad* atau mobil penumpang besar pada mode Balap, tidak diikutsertakan dalam proses perhitungan kemiripan. Hal ini membuat rekomendasi yang dihasilkan menjadi lebih kontekstual terhadap kondisi lintasan dan tujuan permainan pada mode yang sedang dimainkan. Selain itu, *filtering* juga mengurangi jumlah kendaraan yang harus dibandingkan sehingga proses perhitungan menjadi lebih efisien..

*Pseudocode 4.3 Normalisasi Nilai Fitur dengan Metode Min–Max*

```

PROSEDUR NormalisasiMinMax(matriksFitur)
// Input : matriksFitur berukuran N × 5
// Output : matriksFiturTernormalisasi berukuran N × 5
deklarasi matriksFiturTernormalisasi : array[N][5]

UNTUK k ← 0 S/D 4 LAKUKAN
    minK ← NILAI_MINIMUM(matriksFitur[0..N-1][k])
    maxK ← NILAI_MAKSIMUM(matriksFitur[0..N-1][k])

    UNTUK i ← 0 S/D N-1 LAKUKAN
        JIKA (maxK - minK) ≠ 0 MAKA
            matriksFiturTernormalisasi[i][k] ←
                (matriksFitur[i][k] - minK) / (maxK - minK)
        LAIN
            matriksFiturTernormalisasi[i][k] ← 0
        ENDIF
    SELESAI
SELESAI

KEMBALIKAN matriksFiturTernormalisasi
END PROSEDUR

```

*Pseudocode 4.3* menjelaskan proses normalisasi nilai fitur kendaraan menggunakan metode *Min–Max Normalization*. Setiap kolom fitur pada *Item–Fitur Matrix* memiliki rentang nilai yang berbeda-beda, misalnya kecepatan maksimal dapat memiliki rentang puluhan hingga ratusan, sedangkan kapasitas hanya bernilai beberapa orang. Jika perbedaan rentang ini dibiarkan, fitur dengan skala yang lebih besar berpotensi mendominasi hasil perhitungan kemiripan. Oleh karena itu, pada tahap ini sistem terlebih dahulu mencari nilai minimum dan maksimum untuk setiap fitur menggunakan fungsi `NILAI_MINIMUM` dan `NILAI_MAKSIMUM`. Selanjutnya, setiap nilai fitur dihitung ulang dengan rumus  $(\text{nilai} - \text{minK}) / (\text{maxK} - \text{minK})$  sehingga seluruh nilai fitur berada pada rentang 0 hingga 1. Kondisi pengecekan  $(\text{maxK} - \text{minK}) \neq 0$  digunakan untuk menghindari pembagian dengan nol apabila suatu fitur memiliki nilai yang sama untuk semua kendaraan. Hasil akhirnya adalah

matriks fitur ternormalisasi yang lebih seimbang dan siap digunakan pada tahap perhitungan *Cosine Similarity*.

*Pseudocode 4.4 Perhitungan Nilai Cosine Similarity*

```

FUNGSI HitungCosineSimilarity(vektorA, vektorB)
// Input : vektorA dan vektorB berukuran 5 elemen
// Output : nilaiCosine (0 ≤ nilaiCosine ≤ 1)
    deklarasi dotProduct : float ← 0
    deklarasi magnitudeA : float ← 0
    deklarasi magnitudeB : float ← 0

    UNTUK k ← 0 S/D 4 LAKUKAN
        dotProduct ← dotProduct + (vektorA[k] × vektorB[k])
        magnitudeA ← magnitudeA + (vektorA[k] × vektorA[k])
        magnitudeB ← magnitudeB + (vektorB[k] × vektorB[k])
    SELESAI

    magnitudeA ← akar(magnitudeA)
    magnitudeB ← akar(magnitudeB)

    JIKA magnitudeA = 0 ATAU magnitudeB = 0 MAKA
        KEMBALIKAN 0
    LAIN
        KEMBALIKAN dotProduct / (magnitudeA × magnitudeB)
    ENDIF
END FUNGSI

```

*Pseudocode 4.4* menggambarkan implementasi rumus *Cosine Similarity* yang digunakan untuk menghitung tingkat kemiripan antara dua kendaraan. Setiap kendaraan direpresentasikan oleh sebuah vektor fitur hasil normalisasi berukuran lima elemen, kemudian sistem menghitung nilai `dotProduct` sebagai penjumlahan dari hasil perkalian masing-masing pasangan elemen vektor. Selain itu, sistem juga menghitung `magnitudeA` dan `magnitudeB` yang merepresentasikan panjang vektor A dan B melalui penjumlahan kuadrat setiap elemen dan kemudian diakar.

Nilai *Cosine Similarity* diperoleh dengan membagi `dotProduct` dengan hasil perkalian `magnitudeA` dan `magnitudeB`, sehingga menghasilkan nilai di antara 0 dan 1, di mana nilai mendekati 1 menunjukkan bahwa kedua vektor

memiliki arah yang sangat mirip. Pengecekan terhadap kondisi `magnitudeA = 0` atau `magnitudeB = 0` dilakukan untuk mencegah terjadinya pembagian dengan nol jika salah satu vektor tidak memiliki komponen yang valid. Dengan cara ini, fungsi `HitungCosineSimilarity` menjadi inti dari perhitungan kemiripan spesifikasi teknis antar kendaraan pada sistem rekomendasi.

*Pseudocode 4.5 Penyusunan Peringkat Nilai Similarity*

```

PROSEDUR SusunPeringkatSimilarity(daftarKandidat,
matriksTernormalisasi, indeksMobilDipakai)
// Input : daftarKandidat = daftar mobil hasil filter mode
//          matriksTernormalisasi = matriks fitur hasil
normalisasi
//          indeksMobilDipakai = indeks mobil yang digunakan
pemain
// Output : array nilaiSimilarity yang sudah berisi nilai
kemiripan untuk setiap kandidat

    deklarasi nilaiSimilarity : array[ jumlahKandidat ]
    vektorAcuan ← matriksTernormalisasi[indeksMobilDipakai]

    UNTUK j ← 0 S/D jumlahKandidat-1 LAKUKAN
        indeksMobilKandidat ← IndeksGlobal(daftarKandidat[j])
        vektorKandidat ←
        matriksTernormalisasi[indeksMobilKandidat]

        JIKA indeksMobilKandidat ≠ indeksMobilDipakai MAKA
            nilaiSimilarity[j] ←
            HitungCosineSimilarity(vektorAcuan, vektorKandidat)
        LAIN
            nilaiSimilarity[j] ← 0
        ENDIF
    SELESAI

    KEMBALIKAN nilaiSimilarity
END PROSEDUR

```

Pseudocode 4.5 menggambarkan tahap penyusunan nilai kemiripan antara kendaraan yang digunakan pemain dengan seluruh kendaraan lain yang sesuai dengan mode permainan. Sistem pertama-tama mengambil vektor fitur dari kendaraan yang digunakan pemain sebagai vektor acuan. Kemudian, untuk setiap kendaraan kandidat, sistem mengambil vektor fitur kendaraan tersebut dari matriks

ternormalisasi dan menghitung tingkat kemiripan menggunakan fungsi *Cosine Similarity*. Jika kendaraan kandidat merupakan kendaraan yang sama dengan yang dipakai pemain, maka nilai *similarity*-nya diatur menjadi nol agar tidak dipertimbangkan sebagai rekomendasi.

Semua nilai *similarity* yang dihasilkan disimpan dalam array nilai *Similarity*. Pada tahap ini belum dilakukan pemilahan atau pemilihan mobil terbaik, melainkan hanya menyusun nilai kemiripan sebagai dasar untuk proses perankingan dan rekomendasi pada pseudocode berikutnya. Tahap ini sangat penting karena menentukan hubungan kedekatan antar kendaraan secara numerik berdasarkan kesamaan nilai fitur hasil normalisasi.

*Pseudocode 4.6 Penentuan Rekomendasi Utama*

```

PROSEDUR PilihRekomendasiTerbaik(daftarKandidat,
nilaiSimilarity)
// Input : daftarKandidat = daftar mobil yang sesuai mode
//          nilaiSimilarity = daftar nilai similarity yang
telah dihitung
// Output : mobilRekomendasi = kendaraan dengan similarity
tertinggi

    deklarasi similarityTertinggi : float ← 0
    deklarasi indeksTerbaik : integer ← -1

    UNTUK j ← 0 S/D jumlahKandidat-1 LAKUKAN
        JIKA nilaiSimilarity[j] > similarityTertinggi MAKA
            similarityTertinggi ← nilaiSimilarity[j]
            indeksTerbaik ← j
        ENDIF
    SELESAI

    JIKA indeksTerbaik ≠ -1 MAKA
        mobilRekomendasi ← daftarKandidat[indeksTerbaik]
    LAIN
        mobilRekomendasi ← NIL
    ENDIF

    KEMBALIKAN mobilRekomendasi
END PROSEDUR

```

Pseudocode 4.6 menjelaskan proses pemilihan satu kendaraan terbaik berdasarkan nilai *similarity* yang telah dihitung pada tahap sebelumnya. Sistem melakukan iterasi terhadap seluruh nilai *similarity* dan secara bertahap membandingkannya untuk menemukan nilai tertinggi. Variabel *similarityTertinggi* digunakan untuk menyimpan nilai kemiripan terbesar sementara, sedangkan *indeksTerbaik* menyimpan indeks kendaraan kandidat yang memiliki nilai *similarity* tersebut. Jika selama proses iterasi ditemukan nilai *similarity* yang lebih tinggi, kedua variabel tersebut akan diperbarui.

Setelah seluruh kandidat selesai dievaluasi, sistem memeriksa apakah terdapat indeks terbaik yang valid. Jika ada, kendaraan pada posisi tersebut diambil dari *daftarKandidat* dan ditetapkan sebagai *mobilRekomendasi*. Jika tidak ditemukan kandidat yang memiliki *similarity* valid, maka sistem mengembalikan nilai *NIL*. Tahap ini menghasilkan satu mobil yang dianggap paling mirip secara matematis dengan kendaraan yang digunakan pemain, sehingga menjadi rekomendasi utama yang ditampilkan pada antarmuka permainan.

#### 4.2.2 Hasil Implementasi Sistem Rekomendasi

Hasil implementasi menunjukkan bahwa sistem rekomendasi kendaraan pada *game Perfect Car* telah berjalan sesuai dengan rancangan yang dibuat. Sistem menggunakan metode *Content-Based Filtering* dengan perhitungan *Cosine Similarity* untuk menentukan kendaraan yang paling mirip berdasarkan lima atribut utama, yaitu akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas penumpang. Proses perhitungan dilakukan secara otomatis tanpa melibatkan input

preferensi dari pemain, sehingga sistem dapat memberikan hasil rekomendasi secara efisien dan konsisten.

Sistem menampilkan hasil rekomendasi dalam dua bentuk, yaitu tampilan visual di antarmuka permainan dan hasil keluaran pada *console log* Unity untuk keperluan analisis. Hasil rekomendasi utama yang ditampilkan pada antarmuka *game* berupa satu mobil dengan nilai kemiripan tertinggi berdasarkan hasil perhitungan *Cosine Similarity*. Tampilan ini muncul setelah pemain menyelesaikan sesi permainan pada setiap mode.

#### **4.2.2.1 Hasil Implementasi pada Mode Balap**

Pada mode Balap, kendaraan acuan yang digunakan dalam pengujian implementasi adalah Lamborghini Aventador. Setelah pemain menyelesaikan permainan menggunakan kendaraan tersebut, sistem secara otomatis menghitung nilai *Cosine Similarity* terhadap seluruh kendaraan lain yang berada pada kategori Balap. Perhitungan dilakukan berdasarkan data fitur kendaraan yang telah dinormalisasi, yang kemudian dibandingkan dengan vektor fitur kendaraan acuan.

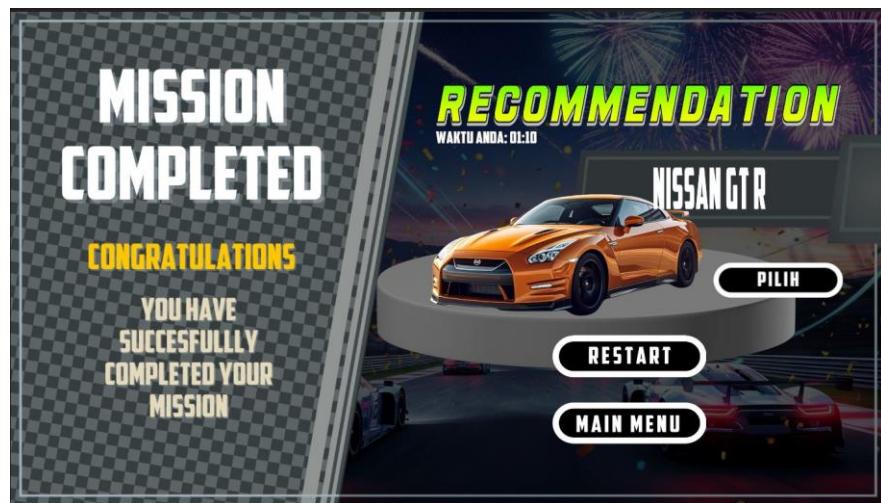
Hasil perhitungan *similarity* ditampilkan pada Tabel 4.1.

Tabel 4.1 Hasil Perhitungan *Similarity* Mobil Lamborghini Mode Balap

<b>Peringkat</b>	<b>Mobil</b>	<b>Nilai <i>Similarity</i></b>
1	Nissan GT-R	0.9947
2	Toyota GR Supra	0.9714
3	Audi R8	0.9451
4	Honda Civic Type R	0.9389
5	MG 5 GT	0.9366
6	Mazda 3	0.9151
7	Toyota Vios	0.8029

Berdasarkan Tabel 4.1, diketahui bahwa Nissan GT-R memiliki nilai *similarity* tertinggi, yaitu 0.9947, sehingga kendaraan tersebut direkomendasikan

sebagai mobil yang paling mirip dengan Lamborghini Aventador untuk mode Balap. Nilai *similarity* yang tinggi ini menunjukkan bahwa GT-R memiliki karakter akselerasi, kecepatan maksimal, dan *Handling* yang paling mendekati Aventador setelah dinormalisasi. Tampilan rekomendasi ini ditampilkan pada panel *Mission Completed* setelah pemain menyelesaikan sesi permainan, sedangkan urutan lengkap nilai *similarity* dapat dilihat melalui console log Unity sebagai bagian dari proses verifikasi hasil perhitungan.



Gambar 4.6 Tampilan Hasil Rekomendasi Mode Balap

Selain Tampilan rekomendasi di antarmuka, sistem juga menampilkan nilai *similarity* tertinggi melalui console log Unity, seperti ditunjukkan pada Gambar 4.2.

```
--- HASIL REKOMENDASI (SERUPA) UNTUK 'Lamborghini Aventador' PADA MODE 'Balap' ---
#1: Nissan GT R (Skor: 0.994654)
#2: Toyota GR Supra (Skor: 0.971366)
#3: Audi R8 (Skor: 0.945108)
#4: Honda Civic Type R (Skor: 0.938886)
#5: MG 5 GT (Skor: 0.936572)
#6: Mazda 3 (Skor: 0.915103)
#7: Toyota Vios (Skor: 0.802915)
```

Gambar 4.7 Tampilan Hasil *Console log* Nilai *Similarity* Mode Balap

Untuk memberikan gambaran kemiripan kendaraan secara menyeluruh, penelitian ini juga menyajikan hasil perhitungan *similarity* untuk seluruh kendaraan

pada mode Balap. Tabel berikut menampilkan nilai *Cosine Similarity* lengkap antar kendaraan pada mode Balap.

Tabel 4.2 Hasil Perhitungan *Cosine Similarity* pada Mode Balap

Balap	A1	A2	A3	A4	A5	A6	A7	A8
A1	1	0,9738	0,9525	0,9384	0,9398	0,8730	0,8029	0,8416
A2	0,9738	1	0,9962	0,9858	0,9906	0,9569	0,9151	0,9413
A3	0,9525	0,9962	1	0,9937	0,9951	0,9724	0,9389	0,9634
A4	0,9384	0,9858	0,9937	1	0,9797	0,9603	0,9366	0,9643
A5	0,9398	0,9906	0,9951	0,9797	1	0,9777	0,9451	0,9654
A6	0,8730	0,9569	0,9724	0,9603	0,9777	1	0,9714	0,9781
A7	0,8029	0,9151	0,9389	0,9366	0,9451	0,9714	1	0,9947
A8	0,8416	0,9413	0,9634	0,9643	0,9654	0,9781	0,9947	1

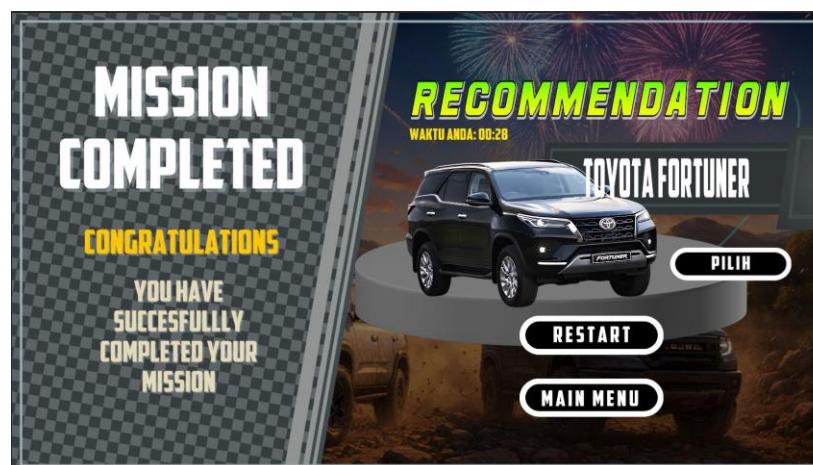
#### 4.2.2.2 Hasil Implementasi pada Mode *Offroad*

Pada mode *Offroad*, kendaraan acuan yang digunakan untuk pengujian adalah Ford Everest. Setelah pemain menyelesaikan permainan menggunakan kendaraan tersebut, sistem menjalankan proses perhitungan *Cosine Similarity* terhadap seluruh kendaraan lain yang termasuk dalam kategori *Offroad*. *Filtering* mode memastikan bahwa hanya kendaraan dengan karakteristik *Offroad* seperti suspensi kuat dan *Handling* stabil yang dimasukkan ke dalam proses perbandingan. Proses ini menghasilkan nilai similaritas yang mencerminkan kesesuaian performa kendaraan lain terhadap profil Ford Everest setelah dinormalisasi.

Tabel 4.3 Hasil Perhitungan *Similarity* Mobil Ford Everest Mode *Offroad*

Peringkat	Kendaraan	Nilai <i>Similarity</i>
1	Toyota Fortuner	0.9970
2	Toyota Rush	0.9862
3	Toyota Hilux	0.9857
4	Ford Ranger Raptor	0.9715
5	Jeep Wrangler	0.9557
6	Honda BR-V	0.9439
7	Suzuki Jimny	0.9209

Berdasarkan Tabel 4.3, kendaraan dengan nilai *similarity* tertinggi terhadap Ford Everest adalah Toyota Fortuner dengan nilai 0.9970. Nilai *similarity* yang sangat tinggi ini menunjukkan bahwa Toyota Fortuner memiliki karakteristik performa yang paling mendekati Ford Everest, terutama pada aspek suspensi dan stabilitas yang menjadi ciri khas kendaraan *Offroad*. Oleh karena itu, Toyota Fortuner ditampilkan sebagai kendaraan rekomendasi utama pada antarmuka permainan setelah sesi mode *Offroad* berakhir. Tampilan hasil rekomendasi ini dapat dilihat pada Gambar 4.3.



Gambar 4.8 Tampilan Hasil Rekomendasi Mode *Offroad*

Selain menampilkan hasil rekomendasi pada antarmuka permainan, sistem juga mencatat seluruh nilai *similarity* yang dihitung pada mode *Offroad* secara lengkap melalui *console log* Unity, sebagaimana ditunjukkan pada Gambar 4.4 berikut:

```
--- HASIL REKOMENDASI (SERUPA) UNTUK 'Ford Everest' PADA MODE 'Offroad' ---
#1: Toyota Fortuner (Skor: 0.996982)
#2: Toyota Rush (Skor: 0.986155)
#3: Toyota Hilux (Skor: 0.985673)
#4: Ford Ranger Raptor (Skor: 0.971490)
#5: Jeep Wrangler (Skor: 0.955667)
#6: Honda BRV (Skor: 0.943859)
#7: Suzuki Jimny (Skor: 0.920939)
```

Gambar 4.9 Tampilan *Console log* Nilai *Similarity* Mode *Offroad*

Untuk melihat hubungan kemiripan antar kendaraan secara lengkap, disajikan pula hasil perhitungan *similarity* untuk seluruh kendaraan pada mode *Offroad*. Tabel berikut menunjukkan nilai *Cosine Similarity* pada kategori kendaraan tersebut.

Tabel 4.4 Hasil Perhitungan *Cosine Similarity* pada Mode *Offroad*

<b><i>Offroad</i></b>	<b>A9</b>	<b>A10</b>	<b>A11</b>	<b>A12</b>	<b>A13</b>	<b>A14</b>	<b>A15</b>	<b>A16</b>
<b>A9</b>	1	0,9933	0,9970	0,9584	0,9490	0,9312	0,9204	0,9782
<b>A10</b>	0,9933	1	0,9862	0,9384	0,9680	0,9035	0,9128	0,9556
<b>A11</b>	0,9970	0,9862	1	0,9715	0,9439	0,9557	0,9209	0,9857
<b>A12</b>	0,9584	0,9384	0,9715	1	0,9335	0,9795	0,9287	0,9840
<b>A13</b>	0,9490	0,9680	0,9439	0,9335	1	0,8739	0,8733	0,9090
<b>A14</b>	0,9312	0,9035	0,9557	0,9795	0,8739	1	0,9051	0,9725
<b>A15</b>	0,9204	0,9128	0,9209	0,9287	0,8733	0,9051	1	0,9562
<b>A16</b>	0,9782	0,9556	0,9857	0,9840	0,9090	0,9725	0,9562	1

#### 4.2.2.3 Hasil Implementasi pada Mode Penumpang

Pada mode Penumpang, kendaraan acuan yang digunakan dalam pengujian adalah Mercedes-Benz Sprinter, yaitu kendaraan van berkapasitas besar yang biasa digunakan untuk angkut penumpang. Setelah pemain menyelesaikan permainan menggunakan kendaraan ini, sistem secara otomatis menghitung nilai *Cosine Similarity* terhadap kendaraan lain yang termasuk dalam kategori kendaraan penumpang dan niaga ringan. *Filtering* mode memastikan bahwa kendaraan yang dibandingkan memiliki karakteristik fungsi yang serupa, seperti kapasitas penumpang yang besar, kenyamanan suspensi, serta stabilitas ketika membawa muatan.

Perhitungan dilakukan menggunakan nilai fitur yang telah dinormalisasi, kemudian diurutkan berdasarkan *similarity* tertinggi hingga terendah. Hasil rinci

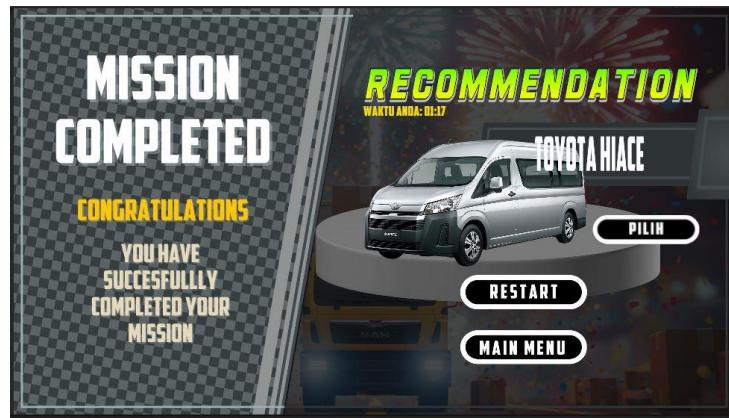
perhitungan *similarity* untuk mode Penumpang ditunjukkan dalam Tabel 4.5 berikut.

Tabel 4.5 Hasil Perhitungan *Similarity* Mobil Mode Penumpang

<b>Peringkat</b>	<b>Kendaraan</b>	<b>Nilai Similarity</b>
1	Toyota HiAce	0.9866
2	DFSK Gelora	0.9815
3	Daihatsu Gran Max	0.9771
4	Daihatsu Sigra	0.9648
5	Toyota Avanza	0.8669
6	Toyota Innova	0.8415
7	Toyota Ayla	0.7506

Berdasarkan Tabel 4.5, kendaraan dengan tingkat kemiripan tertinggi terhadap Mercedes-Benz Sprinter adalah Toyota Hiace dengan nilai *similarity* sebesar 0.9866. Tingginya nilai *similarity* ini menunjukkan bahwa Hiace memiliki kesamaan signifikan dari sisi kapasitas penumpang, karakteristik suspensi, dan kenyamanan berkendara, sehingga menjadi kendaraan yang paling mendekati Sprinter secara matematis. Oleh karena itu, sistem menampilkan Toyota Hiace sebagai kendaraan rekomendasi utama pada panel *Mission Completed* setelah pemain menyelesaikan mode Penumpang.

Tampilan rekomendasi tersebut dapat dilihat pada Gambar 4.10 yang menunjukkan hasil antarmuka panel *Mission Completed*.



Gambar 4.10 Tampilan Hasil Rekomendasi Mode Kurir

Untuk memberikan transparansi perhitungan serta memastikan bahwa rekomendasi yang ditampilkan merupakan kendaraan dengan nilai *similarity* tertinggi, sistem juga menampilkan seluruh hasil urutan perhitungan *similarity* melalui console log Unity, sebagaimana ditunjukkan pada Gambar 4.11.

```
--- HASIL REKOMENDASI (SERUPA) UNTUK 'Mercedes Benz Sprinter' PADA MODE 'Kurir' ---
#1: Toyota Hiace (Skor: 0.986602)
#2: DFSK Gelora (Skor: 0.981509)
#3: Daihatsu GranMax (Skor: 0.977120)
#4: Daihatsu Sigra (Skor: 0.964855)
#5: Toyota Avanza (Skor: 0.866957)
#6: Toyota Innova (Skor: 0.841520)
#7: Daihatsu Ayla (Skor: 0.750683)
```

Gambar 4.11 Tampilan Console log Nilai *Similarity* Mode Penumpang

Tabel 4.6 Hasil Perhitungan *Cosine Similarity* pada Mode Penumpang

Penumpang	A17	A18	A19	A20	A21	A22	A23	A24
<b>A17</b>	1	0,9944	0,9771	0,9960	0,7253	0,7525	0,8916	0,6103
<b>A18</b>	0,9944	1	0,9866	0,9955	0,7579	0,7789	0,9133	0,6357
<b>A19</b>	0,9771	0,9866	1	0,9815	0,8415	0,8670	0,9649	0,7507
<b>A20</b>	0,9960	0,9955	0,9815	1	0,7614	0,7795	0,9103	0,6377
<b>A21</b>	0,7253	0,7579	0,8415	0,7614	1	0,9887	0,9497	0,9616
<b>A22</b>	0,7525	0,7789	0,8670	0,7795	0,9887	1	0,9661	0,9772
<b>A23</b>	0,8916	0,9133	0,9649	0,9103	0,9497	0,9661	1	0,8919
<b>A24</b>	0,6103	0,6357	0,7507	0,6377	0,9616	0,9772	0,8919	1

#### 4.2.2.4 Perbandingan Hasil Rekomendasi Tiap Mode

Setelah dilakukan pengujian sistem rekomendasi pada ketiga mode permainan, diperoleh hasil rekomendasi yang berbeda-beda untuk setiap mode. Hal ini disebabkan oleh proses *filtering* mode yang memastikan bahwa perhitungan *Cosine Similarity* hanya dilakukan pada kelompok kendaraan yang relevan dengan karakteristik tantangan pada masing-masing mode permainan. Dengan demikian, setiap mode menghasilkan satu rekomendasi utama berdasarkan kendaraan yang paling mirip secara matematis dengan kendaraan acuan yang digunakan pemain.

Perbandingan hasil rekomendasi pada ketiga mode permainan dapat dilihat pada Tabel 4.7 berikut.

Tabel 4.7 Perbandingan Hasil Rekomendasi Berdasarkan Mode Permainan

Mode Permainan	Mobil Acuan (Player)	Mobil Rekomendasi	Nilai <i>Similarity</i> Tertinggi
Balap	Lamborghini Aventador	Nissan GT-R	0.9947
Offroad	Ford Everest	Toyota Fortuner	0.9970
Penumpang	Mercedes Benz Sprinter	Toyota Hiace	0.9866

Dari tabel tersebut terlihat bahwa sistem rekomendasi menghasilkan kendaraan yang berbeda untuk setiap mode permainan. Pada mode Balap, Nissan GT-R muncul sebagai kendaraan paling mirip dengan Lamborghini Aventador karena memiliki karakteristik performa tinggi yang serupa. Pada mode *Offroad*, Toyota Fortuner direkomendasikan karena memiliki nilai kemiripan tertinggi terhadap Ford Everest, terutama pada aspek suspensi dan stabilitas lintasan. Sementara itu, pada mode Penumpang, sistem merekomendasikan Toyota Hiace sebagai kendaraan paling santarmuka permainan dan keluaran console log, seluruh

proses perhitungan telah berjalan dengan konsisten dan menghasilkan rekomendasi yang logis pada setiap mode permainan.

Pada mode Balap, Lamborghini Aventador digunakan sebagai kendaraan acuan. Hasil implementasi menunjukkan bahwa Nissan GT-R memiliki nilai kemiripan tertinggi dibandingkan kendaraan lain dalam kategori yang sama. Rekomendasi ini sesuai dengan karakteristik GT-R yang memiliki performa akselerasi, kecepatan maksimal, dan *Handling* yang sekelas dengan Aventador. Perbedaan nilai *similarity* antara kendaraan satu dengan lainnya juga menunjukkan bahwa sistem berhasil mengidentifikasi pola fitur yang paling mendekati kendaraan acuan, meskipun semua kendaraan balap memiliki performa yang relatif tinggi.

Pada mode *Offroad*, Ford Everest menghasilkan rekomendasi Toyota Fortuner sebagai kendaraan yang memiliki nilai kemiripan tertinggi. Hasil tersebut dapat diterima karena Fortuner dan Everest memiliki karakteristik teknis yang serupa, terutama pada aspek suspensi dan stabilitas kendaraan ketika digunakan pada medan berbatu atau tidak rata. Nilai *similarity* pada mode ini memiliki variasi yang lebih terlihat, menunjukkan bahwa fitur-fitur kendaraan *Offroad* memiliki rentang karakteristik yang lebih beragam sehingga proses pemeringkatan menjadi lebih jelas.

Pada mode Penumpang, Mercedes-Benz Sprinter digunakan sebagai kendaraan acuan dan menghasilkan Toyota Hiace sebagai rekomendasi utama. Rekomendasi ini sangat relevan karena Sprinter dan Hiace memiliki kapasitas penumpang besar dan fungsi yang sama sebagai kendaraan angkut penumpang. Hasil *similarity* menunjukkan bahwa Hiace memiliki pola fitur yang paling

mendekati Sprinter setelah melalui proses normalisasi, terutama pada kapasitas dan kenyamanan suspensi.

Secara keseluruhan, hasil implementasi pada ketiga mode permainan menunjukkan bahwa sistem rekomendasi bekerja secara konsisten dan menghasilkan rekomendasi yang relevan dengan karakteristik kendaraan dalam setiap mode. *Filtering* mode memastikan bahwa perbandingan dilakukan pada kelompok kendaraan yang sesuai, sedangkan proses normalisasi menjaga agar setiap fitur memiliki kontribusi seimbang dalam perhitungan. Kendaraan dengan nilai *similarity* tertinggi selalu ditampilkan pada panel *Mission Completed* sebagai rekomendasi utama dan sesuai dengan hasil pemeringkatan yang ditampilkan pada console log. Hal ini menunjukkan bahwa sistem berhasil memberikan rekomendasi yang akurat, stabil, dan bermanfaat bagi pemain dalam memilih kendaraan alternatif yang paling mendekati kendaraan yang digunakan.

### 4.3 Pengujian Sistem

Pengujian sistem dilakukan untuk menilai kinerja serta kualitas rekomendasi yang dihasilkan oleh sistem rekomendasi kendaraan pada *game Perfect Car*. Pengujian difokuskan pada dua aspek utama, yaitu pengujian kesesuaian rekomendasi dan pengujian *System Usability Scale* (SUS). Pengujian kesesuaian rekomendasi digunakan untuk mengevaluasi sejauh mana kendaraan yang direkomendasikan sistem dianggap mirip atau relevan oleh pengguna berdasarkan pengalaman penggunaan pada setiap mode permainan. Sementara itu, pengujian SUS digunakan untuk mengukur tingkat *Usability* sistem dari sudut pandang pengguna, meliputi aspek kemudahan penggunaan, kenyamanan, dan

tingkat penerimaan sistem secara keseluruhan. Kedua pengujian ini diharapkan dapat memberikan gambaran yang komprehensif mengenai keakuratan rekomendasi serta kualitas pengalaman pengguna terhadap sistem yang telah dikembangkan.

#### **4.3.1 Pengujian Kesesuaian Rekomendasi**

Pengujian kesesuaian rekomendasi dilakukan untuk mengevaluasi apakah kendaraan yang direkomendasikan sistem dianggap sesuai oleh pengguna setelah mencoba kendaraan tersebut di dalam permainan. Pengujian ini dilakukan menggunakan pendekatan *User-Centric Evaluation*, di mana keberhasilan sistem diukur berdasarkan persepsi dan kepuasan pengguna terhadap relevansi rekomendasi yang diberikan. Pada proses pengujian ini, setiap responden terlebih dahulu memainkan satu mode permainan menggunakan kendaraan acuan, kemudian sistem memberikan satu kendaraan rekomendasi berdasarkan hasil perhitungan *Cosine Similarity*. Responden kemudian mencoba kendaraan rekomendasi tersebut dan memberikan penilaian langsung berupa “Sesuai” atau “Tidak Sesuai” berdasarkan pengalaman mereka saat menggunakan kedua kendaraan tersebut.

Selain memberikan penilaian kesesuaian, waktu penyelesaian misi pada kendaraan acuan dan kendaraan rekomendasi juga dicatat sebagai data pendukung. Data waktu penyelesaian misi pada kendaraan acuan dan kendaraan rekomendasi turut dicatat sebagai parameter pendukung untuk memvalidasi efektivitas rekomendasi. Perbandingan waktu ini digunakan untuk menganalisis apakah kendaraan yang direkomendasikan mampu memberikan performa yang kompetitif

atau lebih efisien dibandingkan kendaraan sebelumnya. Rekomendasi dinilai Sesuai apabila responden menyatakan bahwa kendaraan rekomendasi terasa mirip atau relevan digunakan pada mode permainan tersebut, sedangkan penilaian Tidak Sesuai diberikan apabila responden merasa kendaraan rekomendasi tidak memiliki karakteristik yang mendekati kendaraan acuan.

Tabel 4.8 Data Pengujian Kesesuaian Rekomendasi

Responden	Mode	Acuan	Waktu	Rekomendasi	Waktu	Penilaian (S/T)
R1	Balap	A5	78	A3	76	Sesuai
R1	Offroad	A13	28	A10	31	Sesuai
R1	Penumpang	A20	95	A18	71	Sesuai
R2	Balap	A8	70	A7	57	Sesuai
R2	Offroad	A15	28	A12	23	Sesuai
R2	Penumpang	A20	103	A19	89	Sesuai
R3	Balap	A4	81	A3	64	Sesuai
R3	Offroad	A15	31	A12	32	Sesuai
R3	Penumpang	A17	65	A18	83	Tidak Sesuai
R4	Balap	A2	83	A3	75	Sesuai
R4	Offroad	A14	19	A12	19	Sesuai
R4	Penumpang	A17	75	A18	77	Sesuai
R5	Balap	A7	59	A8	58	Sesuai
R5	Offroad	A15	35	A12	28	Sesuai
R5	Penumpang	A19	75	A18	88	Sesuai
R6	Balap	A2	88	A3	85	Sesuai
R6	Offroad	A11	26	A9	24	Sesuai
R6	Penumpang	A17	95	A18	92	Sesuai
R7	Balap	A7	65	A8	62	Sesuai
R7	Offroad	A16	34	A12	29	Sesuai
R7	Penumpang	A19	72	A18	98	Tidak Sesuai
...	...	...	...	...	...	...
R15	Balap	A5	75	A3	73	Sesuai
R15	Offroad	A16	26	A11	24	Sesuai
R15	Penumpang	A17	95	A20	93	Sesuai

Setelah dilakukan pengujian terhadap 15 responden pada tiga mode permainan (Balap, Offroad, dan Penumpang), diperoleh total 45 penilaian kesesuaian rekomendasi. Setiap penilaian diklasifikasikan ke dalam dua kategori, yaitu “Sesuai” apabila responden merasa kendaraan rekomendasi memiliki

kemiripan yang relevan dengan kendaraan acuan, dan “Tidak Sesuai” apabila kendaraan rekomendasi dirasa kurang mendekati karakteristik kendaraan acuan.

Perhitungan akurasi dilakukan menggunakan rumus berikut:

$$Akurasi = \frac{\text{Total Pengujian Sesuai}}{\text{Total Pengujian}} \times 100\% \quad (4.1)$$

$$Akurasi = \frac{39}{45} \times 100\%$$

$$Akurasi = 86,67\%$$

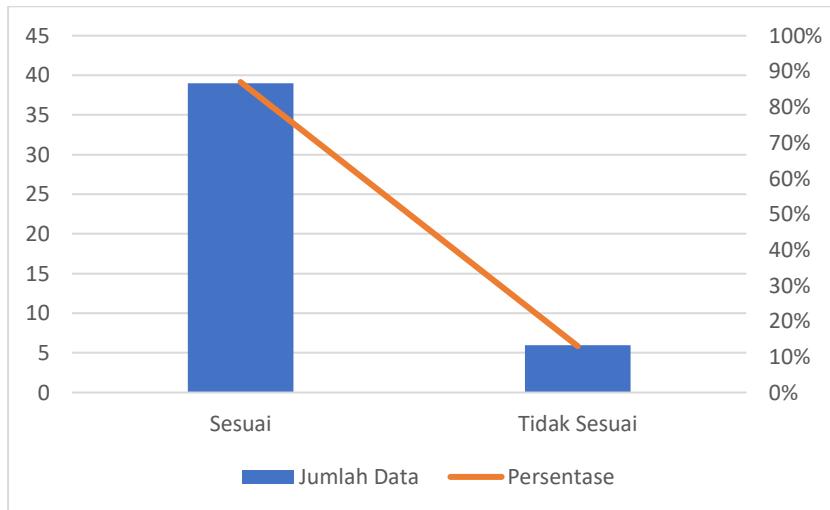
Tabel 4.9 Rekapitulasi Tingkat Akurasi Rekomendasi

Keterangan	Jumlah
Total Penilaian	45
Total “Sesuai”	39
Total “Tidak Sesuai”	6
Akurasi	86,67%

Berdasarkan Tabel 4.9, dari total 45 penilaian yang diberikan oleh 15 responden pada tiga mode permainan, diperoleh 39 penilaian “Sesuai” dan 6 penilaian “Tidak Sesuai”. Hasil akurasi sebesar 86,67% menunjukkan bahwa sistem rekomendasi kendaraan telah bekerja secara efektif dalam mencocokkan kendaraan berdasarkan kemiripan fitur. Tingginya tingkat kesesuaian ini sejalan dengan konsep dasar Content-Based *Filtering*, yang mengasumsikan bahwa item dengan karakteristik serupa akan cenderung dianggap relevan oleh pengguna. Dengan demikian, kendaraan rekomendasi yang dipilih melalui perhitungan *Cosine Similarity* terbukti mampu menggambarkan kedekatan fitur antara kendaraan acuan dan kendaraan rekomendasi.

Selain itu, adanya enam penilaian “Tidak Sesuai” memberikan indikasi bahwa faktor subjektivitas pengguna tetap memengaruhi penilaian akhir terhadap rekomendasi. Hal ini merupakan hal yang lumrah, karena pengalaman bermain,

gaya mengemudi, serta preferensi individu terhadap tipe kendaraan tertentu dapat berbeda dari satu pemain ke pemain lainnya. Meski demikian, dominasi penilaian “Sesuai” mengonfirmasi bahwa algoritma rekomendasi yang digunakan telah mampu menangkap pola kemiripan fitur kendaraan secara tepat.



Gambar 4.12 Grafik Distribusi Tingkat Akurasi Rekomendasi

Untuk memperjelas distribusi hasil pengujian tersebut, visualisasi tingkat akurasi dapat dilihat pada Gambar 4.12. Grafik tersebut menunjukkan perbandingan yang signifikan antara rekomendasi yang dinilai sesuai dan tidak sesuai. Garis persentase memperlihatkan bahwa tingkat penerimaan pengguna mencapai 87% (39 data), yang mendominasi proporsi keseluruhan pengujian. Hal ini mengindikasikan bahwa algoritma *Content-Based Filtering* yang diterapkan telah bekerja secara efektif dalam memberikan rekomendasi yang relevan menurut persepsi mayoritas pemain.

Secara keseluruhan, hasil ini menunjukkan bahwa sistem tidak hanya bekerja secara matematis, tetapi juga mampu memberikan rekomendasi yang dipersepsikan relevan oleh mayoritas pemain. Dengan demikian, sistem

rekomendasi yang diterapkan dapat dianggap stabil, konsisten, dan layak digunakan sebagai fitur pendukung pemilihan kendaraan dalam *game Perfect Car*.

#### **4.3.2 Pengujian System Usability Scale (SUS)**

Pengujian sistem dilakukan untuk menilai sejauh mana sistem rekomendasi kendaraan pada *game Perfect Car* dapat diterima dan digunakan dengan baik oleh pengguna. Pengujian ini tidak berfokus pada aspek fungsionalitas internal seperti pada uji *black-box*, melainkan berorientasi pada penilaian pengalaman pengguna terhadap sistem yang telah dikembangkan. Oleh karena itu, metode yang digunakan dalam tahap ini adalah *System Usability Scale* (SUS). Metode *System Usability Scale* (SUS) merupakan instrumen pengujian standar yang dikembangkan oleh John Brooke pada tahun 1986 untuk mengukur tingkat kegunaan (*Usability*) suatu sistem interaktif berdasarkan persepsi pengguna.

Tabel 4.10 Pernyataan *System Usability Scale* (SUS)

No	Pernyataan	Komponen	(+) / (-)
Q1	Saya berpikir akan sering menggunakan sistem ini dalam memilih mobil saat bermain.	<i>Satisfaction</i>	(+)
Q2	Saya merasa sistem rekomendasi ini terlalu rumit untuk digunakan.	<i>Memorability</i>	( - )
Q3	Saya merasa sistem rekomendasi ini mudah digunakan.	<i>Learnability</i>	( + )
Q4	Saya membutuhkan bantuan teknis atau orang lain untuk menggunakan sistem ini.	<i>Error Handling</i>	( - )
Q5	Saya merasa sistem rekomendasi ini berjalan dengan baik.	<i>Efficiency</i>	( + )
Q6	Saya menilai banyak inkonsistensi dalam fitur rekomendasi ini.	<i>Error Handling</i>	( - )
Q7	Saya merasa kebanyakan orang akan mudah menggunakan fitur rekomendasi ini dengan cepat.	<i>Efficiency</i>	( + )
Q8	Saya menilai fitur rekomendasi ini sangat rumit untuk digunakan.	<i>Memorability</i>	( - )
Q9	Saya merasa sangat percaya diri menggunakan fitur rekomendasi ini.	<i>Satisfaction</i>	( + )
Q10	Saya perlu belajar terlebih dahulu sebelum dapat memahami fitur rekomendasi ini dengan baik.	<i>Learnability</i>	( - )

Berdasarkan Tabel 4.10, setiap pernyataan pada kuesioner *System Usability Scale* (SUS) mewakili lima komponen utama *Usability*, yaitu *learnability*, *efficiency*, *error Handling*, *satisfaction*, dan *memorability*. Komponen *learnability* menggambarkan sejauh mana pengguna dapat memahami dan mempelajari cara kerja sistem rekomendasi kendaraan dengan mudah. *Efficiency* berkaitan dengan efektivitas dan kelancaran pengguna dalam menyelesaikan tugas menggunakan sistem. Sementara itu, *error Handling* menilai kemampuan sistem dalam meminimalkan kebingungan serta kebutuhan bantuan teknis selama penggunaan. Komponen *satisfaction* mencerminkan tingkat kenyamanan dan rasa percaya diri pengguna dalam berinteraksi dengan sistem. Adapun *memorability* berkaitan dengan kemudahan pengguna dalam mengingat kembali cara penggunaan sistem setelah tidak mengaksesnya dalam jangka waktu tertentu. Kelima komponen ini menjadi dasar untuk menilai tingkat kegunaan sistem rekomendasi pada *game Perfect Car* secara menyeluruh.

Tabel 4.11 Nilai *System Usability Scale* (SUS)

No	Jawaban	Nilai
1.	Sangat Tidak Setuju	1
2.	Tidak Setuju	2
3.	Netral	3
4.	Setuju	4
5.	Sangat Setuju	5

Kuesioner ini diberikan kepada 52 responden setelah mereka mencoba langsung fitur sistem rekomendasi kendaraan dalam *game Perfect Car*. Kesepuluh pernyataan tersebut disusun secara bergantian antara pernyataan positif (+) dan negatif (-) dengan tujuan untuk mengurangi bias jawaban responden. Masing-masing pernyataan dijawab menggunakan skala Likert 1–5, di mana nilai 1

menunjukkan Sangat Tidak Setuju dan nilai 5 menunjukkan Sangat Setuju. Pola pernyataan positif dan negatif ini memungkinkan pengukuran persepsi pengguna terhadap kemudahan penggunaan, efisiensi sistem, dan tingkat kepercayaan diri dalam berinteraksi dengan fitur rekomendasi kendaraan. Setelah seluruh data kuesioner dikumpulkan, hasil jawaban responden kemudian diolah menggunakan rumus perhitungan SUS untuk mendapatkan skor rata-rata serta kategori tingkat kegunaan sistem.

Dalam pelaksanaannya, pertanyaan-pertanyaan pada kuesioner *System Usability Scale* (SUS) diklasifikasikan ke dalam dua kategori, yaitu pernyataan positif dan pernyataan negatif. Masing-masing kategori memiliki rumus perhitungan skor yang berbeda agar nilai total yang diperoleh mencerminkan tingkat kepuasan dan kemudahan penggunaan sistem secara akurat.

- a. Untuk pertanyaan bermomor ganjil (Q1, Q3, Q5, Q7, Q9) yang bersifat positif, nilai skor dihitung dengan rumus:

$$\text{Skor} = (\text{Nilai} - 1) \quad (4.1)$$

- b. Untuk pertanyaan bermomor genap (Q2, Q4, Q6, Q8, Q10) yang bersifat negatif, nilai skor dihitung dengan rumus:

$$\text{Skor} = (5 - \text{Nilai}) \quad (4.2)$$

Setelah seluruh skor untuk masing-masing pernyataan dihitung menggunakan rumus di atas, maka nilai total dari setiap responden dijumlahkan. Hasil penjumlahan tersebut kemudian dikalikan dengan faktor 2,5 untuk

mendapatkan skor akhir *System Usability Scale* (SUS) dengan rentang antara 0 hingga 100.

$$\bar{x} = \frac{\Sigma x}{n} \times 2.5 \quad (4.3)$$

Skor ini menggambarkan tingkat kegunaan (*Usability*) sistem berdasarkan persepsi pengguna, di mana semakin tinggi nilai SUS yang diperoleh, maka semakin tinggi pula tingkat kemudahan penggunaan dan kepuasan terhadap sistem rekomendasi kendaraan pada game *Perfect Car*. Tabel di bawah ini menyajikan hasil skor mentah dari setiap responden terhadap 10 pernyataan SUS yang digunakan dalam evaluasi game.

Tabel 4.12 Skor Asli Responden SUS

No	Responden	Skor Asli									
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	R1	2	1	5	1	4	1	5	1	5	2
2	R2	3	1	5	1	3	2	5	1	5	1
3	R3	5	4	4	4	4	4	4	4	5	5
4	R4	4	1	5	1	4	2	4	2	5	2
5	R5	5	1	5	3	5	3	5	1	5	3
6	R6	3	1	5	1	4	1	5	1	4	1
7	R7	4	1	5	2	5	3	5	1	5	2
8	R8	5	1	5	1	5	1	5	1	5	1
...	...	...	...	...	...	...	...	...	...	...	...
52	R52	4	2	5	2	5	2	5	1	3	2

Untuk mempermudah proses analisis, pada penjelasan berikut digunakan contoh data dari salah satu responden yaitu Responden 1 (R1). Data hasil jawaban responden R1 terhadap 10 pernyataan ditunjukkan sebagai berikut:

Tabel 4.13 Perhitungan Responden 1

Kode	Jenis	Nilai Jawaban	Rumus Perhitungan	Skor
Q1	Positif	2	(2 - 1)	1
Q2	Negatif	1	(5 - 1)	4

Kode	Jenis	Nilai Jawaban	Rumus Perhitungan	Skor
Q3	Positif	5	(5 – 1)	4
Q4	Negatif	1	(5 – 1)	4
Q5	Positif	4	(4 – 1)	3
Q6	Negatif	1	(5 – 1)	4
Q7	Positif	5	(5 – 1)	4
Q8	Negatif	1	(5 – 1)	4
Q9	Positif	5	(5 – 1)	4
Q10	Negatif	2	(5 – 2)	3

Total skor responden R<sub>1</sub> diperoleh dari penjumlahan seluruh skor:

$$\text{Total Skor } R_1 = 1 + 4 + 4 + 4 + 3 + 4 + 4 + 4 + 4 + 3 = 35$$

Nilai akhir *System Usability Scale* (SUS) dihitung dengan mengalikan total skor dengan faktor 2,5 :

$$\text{SUS } R_1 = 37 \times 2,5 = 92,5$$

Seluruh hasil perhitungan yang telah dilakukan kemudian dikalikan dengan faktor 2,5 untuk mendapatkan nilai akhir yang lebih representatif. Hasil perhitungan tersebut disajikan dalam Tabel 4.6 berikut :

Tabel 4.14 Hasil Perhitungan Skor SUS

No.	Responden	Jumlah	SUS Skor
1.	R1	35	87,5
2.	R2	35	87,5
3.	R3	21	52,5
4.	R4	34	85
5.	R5	34	85
6.	R6	36	90
7.	R7	35	87,5
8.	R8	40	100
9.	R9	35	87,5
...	...	...	...
51.	R51	27	67,5
52.	R52	33	82,5
Total Skor			4085

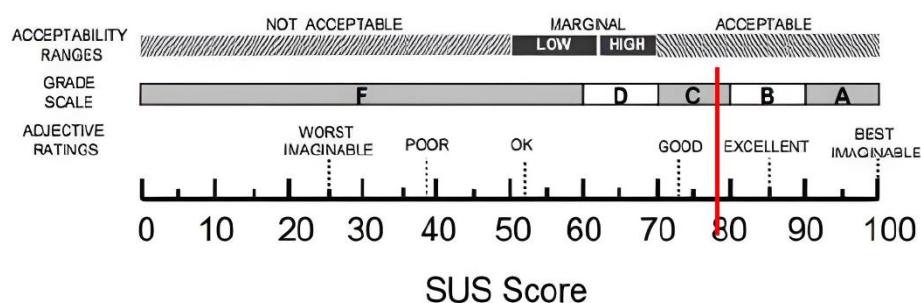
Berdasarkan hasil perhitungan pada Tabel 4.6, dapat diketahui bahwa setiap responden memiliki nilai *System Usability Scale (SUS)* yang bervariasi dengan rentang antara 52,5 hingga 100. Nilai tersebut diperoleh dari hasil konversi jawaban kuesioner yang kemudian dijumlahkan dan dikalikan dengan faktor 2,5 untuk menghasilkan skor akhir dengan rentang 0–100. Dari total 52 responden, jumlah keseluruhan skor SUS yang diperoleh adalah 4085, sehingga rata-rata nilai SUS dapat dihitung sebagai berikut:

$$\text{SUS} = \frac{4085}{52} = 78,56$$

Berdasarkan hasil perhitungan yang diperoleh dari 52 responden, didapatkan nilai rata-rata skor *System Usability Scale (SUS)* sebesar 78,56. Nilai ini menunjukkan bahwa *game Perfect Car* memiliki tingkat kegunaan (*Usability*) yang baik dan dapat diterima oleh pengguna menurut standar pengukuran SUS. Mengacu pada pedoman interpretasi yang dikemukakan oleh John Brooke (1996) dan diperkuat oleh Bangor dkk. (2008), skor SUS dengan nilai di atas 68 sudah dikategorikan memenuhi kriteria kegunaan yang baik (*Good Usability*). Sedangkan skor yang mendekati atau melebihi 80 menandakan bahwa sistem memiliki tingkat kegunaan yang sangat baik serta mampu memberikan pengalaman pengguna yang menyenangkan dan efisien.

Dari hasil analisis kuesioner, mayoritas responden memberikan penilaian tinggi pada pernyataan yang berkaitan dengan kemudahan penggunaan dan kepercayaan diri pengguna dalam mengoperasikan sistem rekomendasi, seperti pada pernyataan “Saya merasa sangat percaya diri menggunakan sistem rekomendasi ini” (Q9) dan “Saya merasa sistem rekomendasi ini mudah

digunakan” (Q3). Hal ini mengindikasikan bahwa pengguna dapat memahami mekanisme rekomendasi kendaraan tanpa mengalami kesulitan yang berarti. Selain itu, aspek integrasi fitur dan keandalan sistem juga memperoleh skor positif, menunjukkan bahwa sistem telah berjalan secara konsisten dan stabil selama pengujian.



Gambar 4.13 Hasil Akhir Skor SUS

Hasil pengujian menunjukkan nilai rata-rata 78,56 menempatkan sistem rekomendasi kendaraan pada *game Perfect Car* dalam kategori “*Acceptable (C)*” atau “*Good*” dalam aspek kegunaan. Artinya, sistem ini sudah memenuhi standar *Usability* yang memuaskan dan dapat diterima oleh pengguna secara umum, meskipun pengembang masih memiliki peluang untuk melakukan penyempurnaan pada sisi desain antarmuka dan interaksi pengguna agar sistem dapat mencapai kategori “*Excellent*” di masa mendatang.

#### 4.3.3 Analisa Aspek *Usability*

Analisis aspek *Usability* dilakukan untuk memahami sejauh mana sistem rekomendasi mobil pada *game Perfect Car* mampu memberikan pengalaman penggunaan yang efektif, efisien, dan memuaskan bagi pengguna. Evaluasi ini

didasarkan pada instrumen *System Usability Scale* (SUS) yang mengukur persepsi pengguna melalui sepuluh pernyataan yang kemudian dikelompokkan ke dalam lima aspek utama, yaitu *learnability*, *efficiency*, *error Handling*, *satisfaction*, dan *Memorability*. Setiap aspek dianalisis untuk mengidentifikasi keunggulan serta potensi perbaikan pada sistem dari sudut pandang pengguna. Berikut merupakan uraian hasil analisis tiap aspek *Usability* berdasarkan data kuesioner yang telah diperoleh.

a. Aspek *Learnability*

*Learnability* merupakan aspek yang menilai seberapa mudah pengguna memahami dan mulai menggunakan sistem sejak pertama kali berinteraksi. Sistem dengan tingkat learnability yang baik memungkinkan pengguna mengoperasikan fitur tanpa membutuhkan proses pembelajaran yang panjang. Berdasarkan hasil kuesioner, pada pernyataan mengenai kemudahan penggunaan (Q3), mayoritas responden memilih skor 5 (59,6%), yang menunjukkan bahwa pengguna merasa sistem sangat mudah untuk dipahami. Hal ini didukung oleh pernyataan Q10 yang bersifat negatif, di mana pilihan terbanyak berada pada skor 2 (42,3%) dan 1 (28,8%), menunjukkan bahwa responden tidak merasa memerlukan banyak pembelajaran sebelum dapat menggunakan fitur rekomendasi. Dengan demikian, sistem menunjukkan tingkat kemudahan belajar yang baik sehingga pengguna dapat memahami fungsionalitasnya secara cepat.

b. Aspek *Efficiency*

*Efficiency* berfokus pada kemampuan sistem dalam membantu pengguna menyelesaikan tugas secara cepat, efektif, dan tanpa hambatan. Sistem yang efisien

akan berfungsi secara lancar dan mendukung kelancaran aktivitas pengguna di dalam *game*. Hasil kuesioner menunjukkan bahwa responden menilai fitur rekomendasi pada *game Perfect Car* bekerja dengan baik. Pada pernyataan Q5, jawaban terbanyak berada pada skor 4 (55,8%), sedangkan pada Q7 mayoritas memilih skor 5 (61,5%). Temuan ini menunjukkan bahwa pengguna merasa sistem memiliki integrasi fungsi yang baik serta dapat dipahami oleh orang lain dengan cepat. Hal tersebut mencerminkan tingkat efisiensi yang tinggi dalam penggunaan sistem rekomendasi.

c. Aspek *Error Handling*

*Error Handling* menilai sejauh mana sistem dapat meminimalkan kebingungan dan menghindari kesalahan yang dapat mengganggu pengalaman pengguna. Sistem yang baik seharusnya memberikan interaksi yang konsisten dan tidak memerlukan bantuan tambahan untuk digunakan. Berdasarkan jawaban responden, pada pernyataan Q4, sebagian besar memilih skor 1 (48,1%), yang mengindikasikan bahwa pengguna tidak memerlukan bantuan teknis atau orang lain untuk mengoperasikan sistem. Pada pernyataan mengenai inkonsistensi sistem (Q6), skor terbanyak berada pada nilai 2 (40,3%), menunjukkan bahwa pengguna tidak menemukan banyak ketidaksesuaian dalam fitur rekomendasi. Dengan demikian, sistem dinilai cukup stabil dan tidak menimbulkan *error* yang signifikan.

d. Aspek *Satisfaction*

*Satisfaction* menggambarkan persepsi pengguna mengenai tingkat kenyamanan dan kepuasan dalam menggunakan sistem, termasuk keinginan untuk menggunakan kembali sistem tersebut. Pada pernyataan Q1, jawaban terbanyak

berada pada skor 4 (48,1%), menunjukkan bahwa pengguna memiliki kecenderungan positif untuk menggunakan fitur rekomendasi dalam permainan. Hal ini diperkuat oleh pernyataan Q9, di mana mayoritas responden memilih skor 5 (42,3%) dan 4 (40,3%), menunjukkan bahwa pengguna merasa percaya diri saat menggunakan fitur rekomendasi yang disediakan. Secara keseluruhan, responden memberikan penilaian yang baik terhadap kenyamanan dan kepuasan penggunaan sistem.

e. Aspek *Memorability*

*Memorability* mengukur sejauh mana pengguna dapat mengingat kembali cara menggunakan sistem setelah tidak mengaksesnya dalam jangka waktu tertentu. Sistem dengan *Memorability* yang baik tidak memerlukan proses belajar ulang ketika digunakan kembali. Pada pernyataan mengenai kerumitan sistem (Q2), jawaban terbanyak berada pada skor 1 (53,8%), yang menunjukkan bahwa pengguna tidak menganggap sistem sulit atau kompleks. Hal ini diperkuat oleh pernyataan Q8, di mana skor 1 (51,9%) juga menjadi pilihan terbanyak, menunjukkan bahwa pengguna tidak merasa kebingungan dalam menggunakan fitur rekomendasi. Dengan demikian, sistem memiliki tingkat *Memorability* yang baik karena pengguna dapat dengan mudah mengingat cara menggunakannya.

Berdasarkan hasil evaluasi terhadap seluruh aspek *Usability learnability, efficiency, error Handling, satisfaction, dan Memorability* dapat disimpulkan bahwa pengguna memberikan penilaian yang dominan positif terhadap fitur rekomendasi dalam *game Perfect Car*. Hal ini tercermin dari tingginya persentase responden yang memilih skor 4 dan 5 pada sebagian besar pernyataan SUS. Secara

keseluruhan, sistem memperoleh skor SUS sebesar 78,56, yang termasuk dalam kategori *Good* dengan tingkat penerimaan *Grade C (Acceptable)*. Hasil ini menunjukkan bahwa sistem rekomendasi yang dikembangkan memenuhi standar kelayakan *Usability* dan dapat diterima oleh pengguna dengan baik.

#### 4.4 Integrasi Islam

Integrasi nilai-nilai Islam dalam ilmu pengetahuan dan teknologi merupakan upaya untuk memadukan antara aspek rasional dan spiritual dalam setiap aktivitas manusia. Islam tidak memisahkan antara agama dan ilmu, karena keduanya saling melengkapi dalam membentuk peradaban yang berkeadaban dan berkeilmuan. Penguasaan ilmu pengetahuan yang tidak disertai nilai-nilai moral dapat menyebabkan penyalahgunaan ilmu, sedangkan nilai-nilai agama tanpa penerapan ilmu akan membatasi kemajuan dan inovasi manusia. Oleh karena itu, integrasi nilai Islam menjadi landasan etis yang menuntun setiap karya ilmiah agar memiliki arah yang benar dan memberi manfaat bagi kehidupan.

Dalam pandangan Islam, manusia dipandang sebagai khalifah di muka bumi yang memiliki tanggung jawab untuk memakmurkan dan menjaga keseimbangan ciptaan Allah Swt. Segala bentuk aktivitas, termasuk pengembangan ilmu dan teknologi, seyogianya didasarkan pada nilai ibadah dan tanggung jawab moral kepada Allah, sesama manusia, dan alam semesta. Prinsip ini tercermin dalam konsep mu‘amalah, yaitu bentuk interaksi manusia yang tidak hanya terbatas pada aspek sosial, tetapi juga mencakup hubungan *vertikal* dan *horizontal* dalam kehidupan.

Para ulama kemudian mengelompokkan mu'amalah menjadi tiga dimensi utama, yaitu: *mu'amalah ma'a Allah*, yang menekankan hubungan dan penghambaan manusia kepada Sang Pencipta; *mu'amalah ma'a an-Nas*, yang mengatur hubungan manusia dengan sesamanya berdasarkan prinsip keadilan, kejujuran, dan kasih sayang; serta *mu'amalah ma'a al-'Alam*, yang menegaskan tanggung jawab manusia terhadap alam dan seluruh ciptaan Allah. Ketiga dimensi tersebut menjadi fondasi integrasi nilai Islam dalam setiap aktivitas manusia, termasuk dalam bidang ilmu pengetahuan dan teknologi.

#### **4.4.1 *Mu'amalah Ma'a Allah***

*Mu'amalah ma'a Allah* merupakan bentuk hubungan vertikal antara manusia dengan Allah Swt. yang menjadi dasar dari seluruh aktivitas kehidupan seorang Muslim. Hubungan ini menuntut kesadaran bahwa segala bentuk usaha manusia baik yang bersifat ibadah maupun aktivitas duniawi pada hakikatnya merupakan bagian dari pengabdian kepada Allah Swt. Nilai-nilai yang terkandung dalam hubungan ini meliputi keikhlasan (ikhlas), ketaatan (tha'ah), dan tanggung jawab (amanah) dalam melaksanakan setiap amanah ilmu dan pekerjaan. Dalam Islam, segala bentuk ilmu dan teknologi harus dikembangkan dengan niat yang lurus dan tujuan yang mulia, karena setiap amal akan dimintai pertanggungjawaban di hadapan Allah Swt.

Pada dimensi hubungan dengan Allah, pengembang dan pengguna teknologi hendaknya meniatkan seluruh aktivitas sebagai bagian dari ibadah dan ketaatan kepada-Nya. Allah SWT menegaskan tujuan penciptaan manusia adalah untuk beribadah kepada-Nya, sebagaimana firman-Nya:

مَا خَلَقْتُ الْجِنَّ وَالْإِنْسَ إِلَّا لِيَعْبُدُونِ

*“Dan aku tidak menciptakan jin dan manusia melainkan supaya mereka mengabdi kepada-Ku.”* (QS. Adz-Dzariyat [51]: 56)

Ayat ini menegaskan bahwa segala aspek kehidupan manusia termasuk inovasi ilmu pengetahuan dan teknologi semestinya diarahkan untuk pengabdian kepada Allah. Menurut Ibnu Katsir dalam Tafsir al-Qur'an al-'Azhim, ayat ini menunjukkan bahwa Allah menciptakan jin dan manusia bukan karena kebutuhan terhadap mereka, melainkan agar mereka mengesakan Allah dan menyembah-Nya dengan penuh keikhlasan. Lafadz “*liyabudūn*” bermakna *liyuwahhidūn* agar mereka mentauhidkan Allah. Ibadah yang dimaksud bukan hanya ritual, tetapi segala aktivitas yang diniatkan karena Allah dan sesuai syariat (kathir, 1419 H).

Quraish Shihab dalam Tafsir al-Misbah menambahkan bahwa ayat ini menegaskan tujuan penciptaan manusia sebagai bentuk pengabdian yang menyeluruh meliputi seluruh aktivitas hidup, termasuk kerja intelektual dan inovasi teknologi selama diarahkan untuk kebaikan dan sesuai nilai moral Islam (Shihab M. Q., 1419 H).

Dalam konteks pengembangan sistem rekomendasi kendaraan berbasis *Content-Based Filtering* pada game *Perfect Car*, nilai *mu'amalah ma'a Allah* tercermin melalui keikhlasan niat dan tanggung jawab ilmiah pengembang dalam merancang sistem yang bermanfaat. Usaha memastikan akurasi algoritma, kejujuran hasil rekomendasi, dan profesionalisme kerja merupakan implementasi nilai ihsan sebagaimana sabda Nabi SAW:

إِنَّ اللَّهَ يُحِبُّ إِذَا عَمِلْتُمْ أَحَدُكُمْ عَمَلاً أَنْ يُتْقِنَهُ

“Sesungguhnya Allah mencintai apabila seseorang di antara kalian melakukan suatu pekerjaan, hendaklah ia melaksanakannya dengan itqan (sempurna).” (Al-Bayhaqi, 1423 H).

Nilai *itqan* ini mengajarkan bahwa kesungguhan dan kualitas dalam bekerja termasuk dalam mengembangkan sistem digital merupakan bagian dari ibadah.

#### **4.4.2 Mu'amalah Ma'a An-Nas**

Dimensi kedua menitikberatkan pada hubungan dan kemaslahatan bagi sesama manusia. Teknologi dalam Islam harus berperan sebagai alat untuk membantu dan memberi manfaat kepada orang lain. Prinsip ini sejalan dengan sabda Nabi Muhammad SAW:

خَيْرُ النَّاسِ أَنْفَعُهُمْ لِلنَّاسِ

“Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia lainnya” (hambal, 1995 H).

Dalam konteks sistem rekomendasi *game*, nilai ini tercermin pada upaya pengembang menghadirkan fitur yang memudahkan pemain (sesama manusia) dalam mengambil keputusan yang tepat. Sistem rekomendasi mobil yang akurat akan membantu pemain memilih kendaraan optimal sesuai kebutuhan, sehingga meningkatkan pengalaman bermain mereka. Memberikan kemudahan dan manfaat semacam ini merupakan perwujudan nyata dari perintah Islam untuk *ta'awun* (saling tolong-menolong dalam kebaikan). Al-Qur'an menegaskan konsekuensi positif bagi siapa saja yang berbuat kebaikan kepada sesama. Firman Allah SWT:

مَنْ جَاءَ بِالْخُسْنَةِ فَلَهُ خَيْرٌ مِّنْهَا وَمَنْ جَاءَ بِالسَّيِّئَةِ فَلَا يُجْزِي الَّذِينَ عَمِلُوا السَّيِّئَاتِ إِلَّا مَا كَانُوا يَعْمَلُونَ

*“Siapa yang datang dengan (membawa) kebaikan, baginya (pahala) yang lebih baik daripada kebaikannya itu. Siapa yang datang dengan (membawa) kejahanatan, maka orang-orang yang telah mengerjakan kejahanatan itu hanya diberi balasan (seimbang) dengan apa yang selalu mereka kerjakan.”* (QS. Al-Qashas:84)

Ayat di atas menggarisbawahi prinsip keadilan Allah: perbuatan baik akan dibalas dengan kebaikan yang lebih besar, sedangkan perbuatan buruk akan dibalas setimpal. Tafsir Ibnu Katsir menjelaskan bahwa Allah menyediakan kenikmatan akhirat yang kekal bagi hamba-hamba-Nya yang beriman, beramal saleh, dan rendah hati. Siapa yang tidak sompong, tidak berbuat sewenang-wenang, dan tidak menimbulkan kerusakan terhadap orang lain akan memperoleh balasan lebih baik dari kebaikan yang ia lakukan (Kathir, 1419 H). Sebaliknya, setiap keburukan yang diperbuat akan dibalas sepadan. Konsep ini menegaskan bahwa setiap tindakan manusia terhadap sesama membawa konsekuensi moral. Penerapannya dalam teknologi adalah keharusan memastikan bahwa sistem yang dibangun mendatangkan maslahat (kebaikan) dan menghindarkan mudarat (kerugian) bagi pengguna.

Dalam pengembangan sistem rekomendasi *Content-Based Filtering* untuk game, pengintegrasian nilai *ma'a an-naas* berarti sistem dirancang dengan memperhatikan keadilan dan kemanfaatan bagi seluruh pemain. Data dan algoritma diperlakukan secara etis, misalnya tidak memihak atau mendiskriminasi pengguna tertentu. Selain itu, transparansi dan akuntabilitas dalam hasil rekomendasi akan meningkatkan kepercayaan pengguna, menciptakan hubungan yang baik antara teknologi dan manusia. Hal ini sesuai dengan etika Islam yang melarang penipuan

(*gharar*) dan kecurangan, serta menganjurkan transparansi dan kejujuran. Ketika sistem rekomendasi mampu membantu pemain meningkatkan performa permainan tanpa merugikan pihak lain, maka teknologi tersebut telah menjadi sarana amal jariyah kebaikan yang mengalir pahalanya. Pengetahuan dan inovasi yang bermanfaat bagi orang banyak tergolong ilmu yang bermanfaat, yang pahalanya terus berlanjut meskipun sang inovator mungkin tak lagi terlibat langsung. Dengan demikian, integrasi nilai Islam pada hubungan sesama manusia mendorong teknologi yang adil, bermanfaat, serta mempererat solidaritas di antara pengguna.

#### **4.4.3 Mu'amalah Ma'a Al-Alam**

Dimensi ketiga menekankan tanggung jawab manusia terhadap alam dan lingkungan sekitarnya. Islam memandang alam sebagai amanah (titipan) dari Allah yang harus dijaga dan dimanfaatkan secara bijaksana. Manusia diberi kedudukan istimewa sebagai khalifah *fil-ardh* (pengelola di bumi) dengan dua peran: sebagai hamba Allah yang tunduk pada ketentuan-Nya, dan sebagai pengelola bumi yang bebas berkreasi namun dibebani tanggung jawab moral. Kebebasan manusia dalam memanfaatkan sumber daya alam bukanlah tanpa batas, melainkan dibatasi oleh prinsip tidak merusak dan menjaga keseimbangan. Allah SWT berfirman memperingatkan tentang kerusakan lingkungan akibat ulah manusia:

ظَهَرَ الْفَسَادُ فِي الْأَرْضِ وَالْبَحْرِ إِمَّا كَسَبَتْ أَيْدِي النَّاسِ لِيُنْدِرُوهُمْ بَعْضَ الَّذِي عَمِلُوا لَعَلَّهُمْ يَرْجِعُونَ

*“Telah tampak kerusakan di darat dan di laut disebabkan karena perbuatan tangan manusia, sehingga Allah menimpakan kepada mereka sebagian dari akibat perbuatan mereka, agar mereka kembali (ke jalan yang benar).”* (QS. Ar-Rum [30]: 41)

Ayat di atas menjelaskan bahwa bencana dan kerusakan alam yang muncul merupakan konsekuensi dari perbuatan manusia yang menyimpang. Ibnu Katsir menafsirkan bahwa kerusakan (*fasād*) yang dimaksud meliputi bencana alam, kekeringan, ketimpangan sosial, dan kehancuran moral akibat dosa manusia. Allah memperingatkan agar manusia menyadari kesalahan dan kembali menjaga keseimbangan bumi (Kathir, 1419 H).

Quraish Shihab dalam *Tafsir al-Misbah* menegaskan bahwa ayat ini menampilkan korelasi langsung antara perilaku moral manusia dan kelestarian lingkungan; kerusakan alam bukan hanya akibat teknis, tetapi juga akibat moralitas yang rusak (Shihab M. Q., 1419 H). Senada dengan itu, para ulama klasik menafsirkan “kerusakan” di sini mencakup bencana alam maupun kekacauan sosial sebagai akibat kemaksiatan. Diriwayatkan, Abu al-‘Aliyah berkata: “Barangsiapa bermaksiat kepada Allah di muka bumi, berarti ia telah berbuat kerusakan di bumi; karena baiknya kondisi bumi dan langit bergantung pada ketaatan kepada Allah.”. Pandangan ini menegaskan keterkaitan erat antara moralitas manusia dan kelestarian lingkungan: kerusakan ekologis pada hakikatnya berakar dari pelanggaran manusia terhadap aturan Allah (misalnya sikap serakah, boros, atau eksplorasi berlebihan).

Dalam konteks pembangunan teknologi dan *game*, integrasi nilai ma’ā al-alam menuntut perhatian terhadap dampak lingkungan dan prinsip keberlanjutan. Pengembang sistem rekomendasi perlu mempertimbangkan efisiensi dan *resource optimization* agar teknologi yang diciptakan tidak menjadi beban lingkungan (misalnya dengan meminimalkan penggunaan daya komputasi yang berlebihan).

Prinsip Islami mlarang *israf* (pemborosan) dan perusakan, sehingga dalam pengembangan *game* pun hendaknya dihindari fitur atau konten yang mendorong sikap eksploratif terhadap alam. Sebaliknya, *game* dapat dijadikan media edukasi nilai lingkungan, misalnya dengan menghadirkan skenario yang mengajak pemain menghargai alam atau memilih opsi kendaraan yang ramah lingkungan di dalam permainan. Penerapan teknologi yang sejalan dengan kepedulian lingkungan ini merupakan wujud aktual dari tugas kekhilafahan. Manusia sebagai khalifah wajib menjaga keseimbangan ekosistem demi generasi sekarang dan yang akan datang. Oleh karena itu, sistem rekomendasi mobil dalam *game* pun idealnya dikembangkan dengan mempertimbangkan aspek keberlanjutan, misalnya mendorong pilihan kendaraan virtual yang efisien atau membatasi elemen permainan yang mencerminkan pemborosan sumber daya.

Secara keseluruhan, integrasi nilai Islam melalui tiga dimensi relasi tersebut akan mewarnai setiap tahap pengembangan sistem dengan kesadaran spiritual dan etis. Nilai *ma'a Allah* memastikan teknologi dikembangkan dengan niat ibadah, syukur, dan kepatuhan pada aturan Ilahi. Nilai *ma'a an-naas* menjamin hasil teknologi berorientasi pada kemaslahatan manusia secara adil dan beradab. Sementara itu, nilai *ma'a al-alam* meneguhkan komitmen untuk menjaga harmoni dengan alam melalui praktik teknologi yang bertanggung jawab. Dengan menggabungkan ketiga dimensi ini, sistem rekomendasi mobil dalam *game* “*Perfect Car*” yang dibangun tidak hanya unggul dalam kinerja, tetapi juga membawa misi moral sebagai kontribusi pada peradaban yang *rahmatan lil 'alamin* (rahmat bagi semesta alam). Teknologi semacam ini diharapkan menjadi contoh

bahwa inovasi modern dapat sejalan dengan nilai-nilai religius, sehingga memberikan manfaat optimal bagi pengguna sekaligus mendekatkan kita kepada ridha Allah SWT.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Implementasi metode *Content-Based Filtering* dengan algoritma *Cosine Similarity* pada game "Perfect Car" berhasil memberikan rekomendasi kendaraan secara otomatis dan relevan. Melalui analisis kemiripan pada lima fitur utama (akselerasi, kecepatan maksimal, *Handling*, suspensi, dan kapasitas), sistem mampu menyajikan saran mobil yang sesuai dengan karakteristik setiap mode permainan (Balap, *Offroad*, dan Penumpang). Hal ini terbukti efektif meningkatkan efisiensi pemain dalam menentukan kendaraan optimal secara konsisten tanpa memerlukan input manual.

Berdasarkan pengujian kesesuaian rekomendasi yang dilakukan, diperoleh tingkat kesesuaian akurasi sebesar 86,67%, yang menunjukkan bahwa sebagian besar responden menilai kendaraan rekomendasi sebagai kendaraan yang mirip atau relevan dengan kendaraan acuan. Hasil ini membuktikan bahwa sistem yang dikembangkan mampu memberikan rekomendasi yang dapat diterima oleh pengguna dalam konteks permainan. Berdasarkan hasil pengujian menggunakan metode *System Usability Scale* (SUS) yang melibatkan 52 responden, diperoleh skor rata-rata sebesar 78,56, yang termasuk dalam kategori "Baik". Nilai ini menunjukkan bahwa sistem rekomendasi pada game *Perfect Car* mudah digunakan, mudah dipelajari, serta memberikan tingkat kepuasan yang tinggi bagi pengguna. Dengan demikian, penelitian ini membuktikan bahwa penerapan metode *Content-Based Filtering* dengan algoritma *Cosine Similarity* efektif digunakan

dalam sistem rekomendasi berbasis *game* simulasi, serta dapat meningkatkan pengalaman bermain yang interaktif, efisien, dan kontekstual.

## 5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan mengenai penerapan sistem rekomendasi mobil menggunakan metode *Content-Based Filtering* pada game "*Perfect Car*", penulis menyadari bahwa meskipun sistem telah berjalan dengan baik, masih terdapat berbagai aspek yang memiliki potensi besar untuk dikembangkan lebih lanjut. Demi penyempurnaan sistem dan peningkatan kualitas pengalaman pengguna di masa mendatang, penulis merumuskan beberapa rekomendasi konstruktif yang dapat dijadikan bahan pertimbangan bagi penelitian selanjutnya, antara lain:

1. Penambahan Data dan Fitur Kendaraan Disarankan untuk memperbanyak jumlah data mobil dan parameter teknis (seperti torsi atau jenis penggerak) agar sistem dapat memberikan rekomendasi yang jauh lebih spesifik dan variatif.
2. Penerapan Metode *Hybrid Filtering* Pengembangan selanjutnya dapat menggabungkan metode *Content-Based* dengan *Collaborative Filtering* agar sistem tidak hanya melihat kemiripan mobil, tetapi juga dapat mempelajari pola preferensi unik dari setiap pemain.
3. Penggunaan Instrumen Evaluasi Tambahan Selain SUS, disarankan menggunakan metode seperti *User Experience Questionnaire* (UEQ) untuk mengukur aspek kenyamanan dan kepuasan emosional pengguna secara lebih mendalam.

## DAFTAR PUSTAKA

- 'Alim, A., Solichin, A., & Painem, P. (2020). Sistem Rekomendasi Keterampilan Teknologi Informasi Dengan Metode User-Based Collaborative *Filtering* dan Log-Likelihood *Similarity*. *CogITO Smart Journal*, 6(2), 141–154. <https://doi.org/10.31154/cogito.v6i2.234.141-154>
- Al-Bayhaqi, A. B. (2003). *Syu'abul Iman*. (Terjemahan). Riyadh: Maktabah Al-Rusyd.
- Alfiani, W. N. (2023). Rekomendasi Pemilihan Motor Pada *Game* “Blar Rising” Menggunakan Metode Topsis Berbasis Bonferroni Mean. *Universitas Islam Negeri Maulana Malik Ibrahim Malang*, VIII(I), 1–19.
- Azri Saputra, J. M., Huizen, L. M., & Arianto, D. B. (2024). Sistem Rekomendasi Film pada Platform Streaming Menggunakan Metode Content-Based *Filtering*. *Jurnal Transformatika*, 22(1), 10. <https://doi.org/10.26623/transformatika.v22i1.7041>
- Bagus, I., Adi, K., Buana, P. W., & W, A. A. K. A. C. (2015). *Game* Edukasi Rambu Lalu Lintas Berbasis Android. *Merpati*, 3(3), 190–201.
- Bangor, A., Kortum, P. T., & Miller, J. T. (2008). An empirical evaluation of the *System Usability Scale*. *International Journal of Human-Computer Interaction*, 24(6), 574–594. <https://doi.org/10.1080/10447310802205776>
- CarsGuide. (2024). *Car Reviews, News & Advice / Car Specifications*. Diakses pada 09 April 2025, dari <https://www.carsguide.com.au/>
- Chamidah, N., Jurusan Teknik Informatika, W., Salamah, U., & Kunci Backpropagasi, K. (2012). Pengaruh Normalisasi Data pada Jaringan Syaraf Tiruan Backpropagasi Gradient Descent Adaptive Gain (BPGDAG) untuk Klasifikasi. *Jurnal Itsmart*, 1(1), 28–33. <http://archive.ics.uci.edu/ml/>.
- Chythanya N, K., Bhargavi Y, K., & Rohan, B. (2019). A Novel Video *Game* Recommender System using Content Based *Filtering* -Vidya. *International Journal of Recent Technology and Engineering (IJRTE)*, 8(4), 536–541. <https://doi.org/10.35940/ijrte.d7251.118419>
- Didi Riswan, Heri Eko Rahmadi Putra, & Risfan Nazar Saputra. (2024). Pengembangan Sistem Rekomendasi Berbasis Kecerdasan Buatan Untuk Meningkatkan Pengalaman Pengguna Di Platform E-Commerce. *Jurnal Komputer Teknologi Informasi Dan Sistem Informasi (JUKTISI)*, 2(3), 572–580. <https://doi.org/10.62712/juktisi.v2i3.145>
- Erfan, M., & Maulyda, M. A. (2020). Meningkatkan Pemahaman Konsep Bangun Ruang Mahasiswa Calon Guru Sekolah Dasar Menggunakan *Game* Android. *Palapa*, 8(2), 418–427. <https://doi.org/10.36088/palapa.v8i2.925>
- Farhan, M., Andreawan, M. R., Antonius, R., & Ulhag, N. D. (2024). *Evaluasi Dan Pengembangan Sistem Rekomendasi Game Berbasis Content-Based Filtering Dengan TF-IDF Dan Cosine Similarity*. 2(4), 400–408.
- Gamadya Wiratamtama, R., Pragantha, J., & Haris, D. A. (2020). Pembuatan *Game* VR Parking Simulator Dengan Unity. *Journal Komputer Dan Informatika*, 15(1), 334–340.
- Gaudioso, E. (2008). *Evaluation of recommender systems : A new approach*. 35,

- 790–804. <https://doi.org/10.1016/j.eswa.2007.07.047>
- Hadisaputra. (2022). Strategi Pemanfaatan *Game* Online Dalam Mendidik Anak Usia Dini. *NANAEKE: Indonesian Journal of Early Childhood Education*, 5(1), 1–14. <https://doi.org/10.24252/nananeke.v5i1.26721>
- Hanbal, A. (1995). *Musnad Ahmad bin Hanbal*. Kairo: Dar al-Hadits.
- Hasan, A., Rahmat, A., & Napu, Y. (2021). Dampak *Game* Online Mobile Legends Terhadap Perilaku Sosial Remaja. *Student Journal of Community Education*, 1, 1–13. <https://doi.org/10.37411/sjce.v1i1.830>
- Herlambang Yudha, P., Muhammad Aminul, A., & Issa, A. (2019). Penerapan Neural Network untuk NPC Braking Decision pada Racing *Game*. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 3(5), 4148–4154.
- Husein, S. M., Savitri, P., Studi, P., Informatika, T., Teknik, F., Buana, U. S., & Sunda, B. (2019). Pembangunan *Game* Edukasi Belajar Bahasa Sunda Berbasis Android Menggunakan Construct2 Dan Adobe Phonegap. *Infotronik : Jurnal Teknologi Informasi Dan Elektronika*, 4(2), 64–70. <https://doi.org/10.32897/infotronik.2019.4.2.4>
- Ibnu Katsir, I. (1999). *Tafsir Al-Qur'an Al-'Azhim*. (Terjemahan). Jakarta: Pustaka Imam Asy-Syafi'i.
- Iskandar, M. A., & Cahyadi, D. (2021). Desain Wadah Permainan Scrabble Dan Ludo Untuk Anak -Anak Usia 7-12 Tahun. *Jurnal Kreatif: Desain Produk Industri Dan Arsitektur*, 9(2), 8. <https://doi.org/10.46964/jkdpia.v9i2.183>
- Knijnenburg, B. P., Willemsen, M. C., Gantner, Z., Soncu, H., & Newell, C. (2012). *Explaining the user experience of recommender systems*. 441–504. <https://doi.org/10.1007/s11257-011-9118-4>
- Kurniaji, A., & Santi, R. C. N. (2023). Implementasi Metode Content Based Filtering Pada Pemilihan Komik. *Jurnal Informatika*, 10(2), 109–117. <https://doi.org/10.31294/inf.v10i2.16113>
- Lewis, J. R. (2018). The *System Usability Scale*: Past, Present, and Future. *International Journal of Human-Computer Interaction*, 34(7), 577–590. <https://doi.org/10.1080/10447318.2018.1455307>
- Marzian, F., & Qamal, M. (2017). *Game* Rpg “the Royal Sword” Berbasis Desktop Dengan Menggunakan Metode Finite State Machine (Fsm). *Sisfo: Jurnal Ilmiah Sistem Informasi*, 1(2), 61–96. <https://doi.org/10.29103/sisfo.v1i2.244>
- Mulachela, A. (2020). Analisis Perkembangan Industri *Game* di Indonesia Melalui Pendekatan Rantai Nilai Global (Global Value Chain). *Indonesian Journal of Global Discourse*, 2(2), 32–51. <https://doi.org/10.29303/ijgd.v2i2.17>
- Oto.com. (2024). *Portal Spesifikasi dan Harga Mobil Baru di Indonesia*. Diakses pada 09 April 2025, dari <https://www.oto.com/>
- Pragusma, Z. B., Kwandinata, V., Natthannael, J., Meiliana, & Achmad, S. (2023). *Game* Recommendation Using Content-based Algorithm Title. *2023 International Conference on Informatics, Multimedia, Cyber and Information Systems, ICIMCIS 2023*, November 2023, 313–317. <https://doi.org/10.1109/ICIMCIS60089.2023.10349063>
- Putra, A. I., & Santika, R. R. (2020). Implementasi Machine Learning dalam Penentuan Rekomendasi Musik dengan Metode Content-Based *Filtering*. *Eduomatic : Jurnal Pendidikan Informatika*, 4(1), 121–130.

<https://doi.org/10.29408/edumatic.v4i1.2162>

- Putri, D. A., Pramesti, D., I, D., & Santiyasa, W. (2022). Penerapan Metode Content-Based *Filtering* dalam Sistem Rekomendasi Video *Game*. *Jnativa*, 1(1), 229–234.
- Ridwansyah, T., Subartini, B., & Sylviani, S. (2024). Penerapan Metode Content-Based *Filtering* pada Sistem Rekomendasi. *Mathematical Sciences and Applications Journal*, 4(2), 70–77.
- Samuel, R., Natan, R., & Syafiqoh, U. (2018). Application of *Cosine Similarity* and K-Nearest Neighbor (K-NN) in Classification and Book Search. *Journal of Big Data Analytic and Artificial Intelligence*, 1(1), 9–14.
- Shihab, M. Q. (2002). *Tafsir Al-Misbah: Pesan, Kesan dan Keserasian Al-Qur'an*. Jakarta: Lentera Hati.
- Sri Handayani, F., & Adelin, A. (2021). Interpretasi Pengujian Usabilitas Wibatara Menggunakan *System Usability Scale* Interpretation of Wibatara *Usability Testing* Using *System Usability Scale*. *Techno Com Jurnal Teknologi Informasi*, 18(4), 340–347.
- Ulfa, R. (2021). *Mengukur Kepuasan Pengguna Sistem Informasi Bimbingan Konseling (E-BK) Menggunakan System Usability Scale (SUS) Di SMK Negeri 1 Banda Aceh*. 1–77.
- Utomo, D. C., Atina, V., & Widyaningsih, P. (2024). Penerapan Metode Content Based *Filtering* Pada Sistem Rekomendasi Pemilihan Buku Referensi Rumah Belajar Pancasila. *Infotech: Journal of Technology Information*, 10(1), 121–128. <https://doi.org/10.37365/jti.v10i1.262>
- Viljanen, M., Vahlo, J., Koponen, A., & Pahikkala, T. (2020). *Content Based Player and Game Interaction Model for Game Recommendation in the Cold Start setting*. <http://arxiv.org/abs/2009.08947>
- Xu, R. (2021). A Hybrid Recommendation System For Online Car Auction Platform, And Then Product Review. *Pharmacognosy Magazine*, 75(17), 399–405.
- Yuen, M. C., Yung, C. W., Cheng, W. F., Tsang, H. P., Kwan, C. H., Chan, C. L., & Li, P. Y. (2023). *Game Recommendation System*. *Frontiers in Artificial Intelligence and Applications*, 378, 843–857. <https://doi.org/10.3233/FAIA231096>
- Yuwono, A. I. (2021). Eksistensi Developer *Game* Independen Indonesia (Studi Kasus Eksistensi Developer *Game* Independen Agate Studio, Creacle Studio, dan Digital Happiness Dalam Perspektif Ekonomi Politik Komunikasi). *Jurnal Media Dan Komunikasi Indonesia*, 2(1), 22. <https://doi.org/10.22146/jmki.63054>

# **LAMPIRAN**

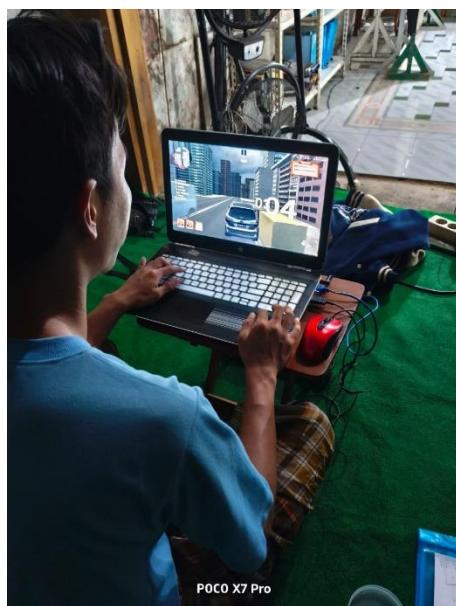
Lampiran A. Dokumentasi Uji Coba Game Perfect Car



Uji Coba Responden1



Uji Coba Responden2



Uji Coba Responden5



Uji Coba Responden4

Lampiran B. Hasil Kuisioner SUS

No	Responden	Skor Asli									
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
1	R1	2	1	5	1	4	1	5	1	5	2
2	R2	3	1	5	1	3	2	5	1	5	1
3	R3	5	4	4	4	4	4	4	4	5	5
4	R4	4	1	5	1	4	2	4	2	5	2
5	R5	5	1	5	3	5	3	5	1	5	3
6	R6	3	1	5	1	4	1	5	1	4	1
7	R7	4	1	5	2	5	3	5	1	5	2
8	R8	5	1	5	1	5	1	5	1	5	1
9	R9	3	1	5	1	4	2	5	1	5	2
10	R10	3	2	3	3	3	3	3	3	3	4
11	R11	3	2	4	1	5	2	5	1	4	2
12	R12	3	1	4	1	5	2	4	2	5	2
13	R13	4	1	4	2	4	1	4	1	4	2
14	R14	3	1	5	1	5	2	5	1	4	1
15	R15	4	2	5	2	4	2	5	2	4	3
16	R16	1	5	3	4	3	3	3	4	1	5
17	R17	2	1	5	1	3	4	3	2	3	2
18	R18	5	1	5	1	4	1	5	1	5	2
19	R19	2	2	4	3	2	2	5	2	1	1
20	R20	4	1	5	1	4	1	5	1	5	1
21	R21	3	2	4	2	3	1	3	2	4	2
22	R22	4	4	4	4	4	4	4	4	4	4
23	R23	4	2	2	4	4	4	4	2	4	4
24	R24	4	1	4	1	5	1	5	2	3	2
25	R25	5	1	5	1	5	1	5	1	4	1
26	R26	4	1	5	1	4	1	5	1	5	2
27	R27	4	3	3	2	3	2	5	2	4	3
28	R28	4	2	4	2	5	2	5	2	4	2
29	R29	4	1	5	2	5	2	5	2	5	2
30	R30	5	2	4	2	4	1	5	1	4	3
31	R31	5	1	5	1	5	1	5	1	5	1
32	R32	4	2	5	1	4	2	4	2	4	2
33	R33	4	2	5	4	4	2	4	2	5	2
34	R34	5	2	5	2	4	1	5	1	5	1
35	R35	5	1	5	1	4	2	5	1	5	2
36	R36	4	2	5	1	4	2	5	1	5	2
37	R37	4	1	5	1	4	2	5	1	4	2
38	R38	4	2	2	3	3	4	4	3	4	4
39	R39	4	1	5	2	5	1	5	2	2	3
40	R40	4	3	4	2	4	2	4	1	4	1
41	R41	3	1	4	2	4	4	5	1	5	3
42	R42	2	2	2	1	3	4	4	2	4	3
43	R43	5	2	5	3	4	2	4	2	3	1
44	R44	4	1	5	1	4	3	5	1	4	1
45	R45	4	1	5	2	4	1	4	2	5	1
46	R46	3	3	3	2	4	2	4	3	4	4
47	R47	5	1	5	1	5	1	5	1	5	1
48	R48	4	1	5	1	4	3	5	1	4	2
49	R49	4	1	5	1	4	3	5	1	5	1
50	R50	2	4	4	4	4	4	4	2	4	2

No	Responden	Skor Asli									
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
51	R51	4	2	4	3	4	2	4	2	3	3
52	R52	4	2	5	2	5	2	5	1	3	2

Lampiran F. Skor Hasil Perhitungan SUS

Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Jumlah	SUS Skor
R1	1	4	4	4	3	4	4	4	4	3	35	87,5
R2	2	4	4	4	2	3	4	4	4	4	35	87,5
R3	4	1	3	1	3	1	3	1	4	0	21	52,5
R4	3	4	4	4	3	3	3	3	4	3	34	85
R5	4	4	4	2	4	2	4	4	4	2	34	85
R6	2	4	4	4	3	4	4	4	3	4	36	90
R7	3	4	4	3	4	2	4	4	4	3	35	87,5
R8	4	4	4	4	4	4	4	4	4	4	40	100
R9	2	4	4	4	3	3	4	4	4	3	35	87,5
R10	2	3	2	2	2	2	2	2	2	1	20	50
R11	2	3	3	4	4	3	4	4	3	3	33	82,5
R12	2	4	3	4	4	3	3	3	4	3	33	82,5
R13	3	4	3	3	3	4	3	4	3	3	33	82,5
R14	2	4	4	4	4	3	4	4	3	4	36	90
R15	3	3	4	3	3	3	4	3	3	2	31	77,5
R16	0	0	2	1	2	2	2	1	0	0	10	25
R17	1	4	4	4	2	1	2	3	2	3	26	65
R18	4	4	4	4	3	4	4	4	4	3	38	95
R19	1	3	3	2	1	3	4	3	0	4	24	60
R20	3	4	4	4	3	4	4	4	4	4	38	95
R21	2	3	3	3	2	4	2	3	3	3	28	70
R22	3	1	3	1	3	1	3	1	3	1	20	50
R23	3	3	1	1	3	1	3	3	3	1	22	55
R24	3	4	3	4	4	4	4	3	2	3	34	85
R25	4	4	4	4	4	4	4	4	3	4	39	97,5
R26	3	4	4	4	3	4	4	4	4	3	37	92,5
R27	3	2	2	3	2	3	4	3	3	2	27	67,5
R28	3	3	3	3	4	3	4	3	3	3	32	80
R29	3	4	4	3	4	3	4	3	4	3	35	87,5
R30	4	3	3	3	3	4	4	4	3	2	33	82,5
R31	4	4	4	4	4	4	4	4	4	4	40	100
R32	3	3	4	4	3	3	3	3	3	3	32	80
R33	3	3	4	1	3	3	3	3	4	3	30	75
R34	4	3	4	3	3	4	4	4	4	4	37	92,5
R35	4	4	4	4	3	3	4	4	4	3	37	92,5
R36	3	3	4	4	3	3	4	4	4	3	35	87,5
R37	3	4	4	4	3	3	4	4	3	3	35	87,5
R38	3	3	1	2	2	1	3	2	3	1	21	52,5
R39	3	4	4	3	4	4	4	3	1	2	32	80
R40	3	2	3	3	3	3	3	4	3	4	31	77,5
R41	2	4	3	3	3	1	4	4	4	2	30	75
R42	1	3	1	4	2	1	3	3	3	2	23	57,5
R43	4	3	4	2	3	3	3	3	2	4	31	77,5
R44	3	4	4	4	3	2	4	4	3	4	35	87,5

<b>Responden</b>	<b>Q1</b>	<b>Q2</b>	<b>Q3</b>	<b>Q4</b>	<b>Q5</b>	<b>Q6</b>	<b>Q7</b>	<b>Q8</b>	<b>Q9</b>	<b>Q10</b>	<b>Jumlah</b>	<b>SUS Skor</b>
R45	3	4	4	3	3	4	3	3	4	4	35	87,5
R46	2	2	2	3	3	3	3	2	3	1	24	60
R47	4	4	4	4	4	4	4	4	4	4	40	100
R48	3	4	4	4	3	2	4	4	3	3	34	85
R49	3	4	4	4	3	2	4	4	4	4	36	90
R50	1	1	3	1	3	1	3	3	3	3	22	55
R51	3	3	3	2	3	3	3	3	2	2	27	67,5
R52	3	3	4	3	4	3	4	4	2	3	33	82,5
Jumlah skor SUS												4085
Skor Rata-rata SUS												78,56

#### Lampiran D. Data Pengujian Akurasi

<b>Responden</b>	<b>Mode</b>	<b>Acuan</b>	<b>Waktu</b>	<b>Rekomendasi</b>	<b>Waktu</b>	<b>Penilaian (S/T)</b>
<b>R1</b>	Balap	A5	78	A3	76	Sesuai
<b>R1</b>	<i>Offroad</i>	A13	28	A10	31	Sesuai
<b>R1</b>	Penumpang	A20	95	A17	71	Sesuai
<b>R2</b>	Balap	A8	70	A7	57	Sesuai
<b>R2</b>	<i>Offroad</i>	A15	28	A16	23	Sesuai
<b>R2</b>	Penumpang	A20	103	A17	89	Sesuai
<b>R3</b>	Balap	A4	81	A3	64	Sesuai
<b>R3</b>	<i>Offroad</i>	A15	31	A16	32	Sesuai
<b>R3</b>	Penumpang	A17	65	A20	83	Tidak Sesuai
<b>R4</b>	Balap	A2	83	A3	75	Sesuai
<b>R4</b>	<i>Offroad</i>	A14	19	A12	19	Sesuai
<b>R4</b>	Penumpang	A17	75	A20	77	Sesuai
<b>R5</b>	Balap	A7	59	A8	58	Sesuai
<b>R5</b>	<i>Offroad</i>	A15	35	A16	28	Sesuai
<b>R5</b>	Penumpang	A19	75	A18	88	Sesuai
<b>R6</b>	Balap	A2	88	A3	85	Sesuai
<b>R6</b>	<i>Offroad</i>	A16	26	A11	24	Sesuai
<b>R6</b>	Penumpang	A17	95	A20	92	Sesuai
<b>R7</b>	Balap	A7	65	A8	62	Sesuai
<b>R7</b>	<i>Offroad</i>	A10	34	A9	29	Sesuai
<b>R7</b>	Penumpang	A18	72	A20	98	Tidak Sesuai
<b>R8</b>	Balap	A8	60	A7	56	Sesuai
<b>R8</b>	<i>Offroad</i>	A16	24	A11	22	Sesuai
<b>R8</b>	Penumpang	A19	70	A18	71	Sesuai
<b>R9</b>	Balap	A4	95	A3	90	Sesuai
<b>R9</b>	<i>Offroad</i>	A14	28	A12	26	Sesuai
<b>R9</b>	Penumpang	A24	135	A22	120	Sesuai
<b>R10</b>	Balap	A2	84	A3	80	Sesuai
<b>R10</b>	<i>Offroad</i>	A16	27	A11	25	Sesuai
<b>R10</b>	Penumpang	A17	74	A20	90	Tidak Sesuai
<b>R11</b>	Balap	A4	89	A3	85	Sesuai
<b>R11</b>	<i>Offroad</i>	A10	36	A9	30	Sesuai
<b>R11</b>	Penumpang	A22	115	A21	110	Sesuai
<b>R12</b>	Balap	A5	72	A3	76	Tidak Sesuai
<b>R12</b>	<i>Offroad</i>	A14	22	A12	21	Sesuai
<b>R12</b>	Penumpang	A21	96	A22	112	Tidak Sesuai
<b>R13</b>	Balap	A1	92	A2	88	Sesuai

<b>Responden</b>	<b>Mode</b>	<b>Acuan</b>	<b>Waktu</b>	<b>Rekomendasi</b>	<b>Waktu</b>	<b>Penilaian</b>
<b>R13</b>	<i>Offroad</i>	A13	38	A10	36	Sesuai
<b>R13</b>	Penumpang	A18	80	A20	105	Sesuai
<b>R14</b>	Balap	A4	86	A3	82	Sesuai
<b>R14</b>	<i>Offroad</i>	A15	35	A16	42	Tidak Sesuai
<b>R14</b>	Penumpang	A24	140	A22	125	Sesuai
<b>R15</b>	Balap	A5	75	A3	73	Sesuai
<b>R15</b>	<i>Offroad</i>	A16	26	A11	24	Sesuai
<b>R15</b>	Penumpang	A17	95	A20	93	Sesuai