

**PEMILIHAN TINGKAT KESULITAN PADA *ENDLESS RUNNER GAME*
'AVOID NAJASA' MENGGUNAKAN METODE AHP-VIKOR
BERDASARKAN KEMAMPUAN PEMAIN**

SKRIPSI

**Oleh :
R. ABDURROSYID
NIM. 220605110042**



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

**PEMILIHAN TINGKAT KESULITAN PADA *ENDLESS RUNNER GAME*
'AVOID NAJASA' MENGGUNAKAN METODE AHP-VIKOR
BERDASARKAN KEMAMPUAN PEMAIN**

SKRIPSI

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

Oleh :
R. ABDURROSYID
NIM. 220605110042

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2026**

HALAMAN PERSETUJUAN

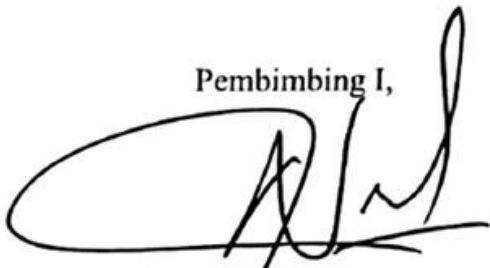
PEMILIHAN TINGKAT KESULITAN PADA *ENDLESS RUNNER GAME* 'AVOID NAJASA' MENGGUNAKAN METODE AHP-VIKOR BERDASARKAN KEMAMPUAN PEMAIN

SKRIPSI

Oleh :
R. ABDURROSYID
NIM. 220605110042

Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 22 Juni 2026

Pembimbing I,



Dr. Ir. Fresy Nugroho, ST., MT, IPM
NIP. 19710722 201101 1 001

Pembimbing II,



Nur Fitriyah Ayu Tunjung Sari, M.Cs
NIP. 19911226 202012 2 001

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



Supriyono, M.Kom
NIP. 19841010 201903 1 012

HALAMAN PENGESAHAN

PEMILIHAN TINGKAT KESULITAN PADA *ENDLESS RUNNER GAME* 'AVOID NAJASA' MENGGUNAKAN METODE AHP-VIKOR BERDASARKAN KEMAMPUAN PEMAIN

SKRIPSI

Oleh :
R. ABDURROSYID
NIM. 220605110042

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 22 Juni 2026

Susunan Dewan Penguji

Ketua Penguji	: <u>Fajar Rohman Hariri, M.Kom</u> NIP. 19890515 201801 1 001	()
Anggota Penguji I	: <u>Ahmad Fahmi Karami, M.Kom</u> NIP. 19870909 202012 1 001	()
Anggota Penguji II	: <u>Dr. Ir. Fresy Nugroho, M.T</u> NIP. 19710722 201101 1 001	()
Anggota Penguji III	: <u>Nur Fitriyah Ayu Tunjung Sari, M. Cs</u> NIP. 19911226 202012 2 001	()

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



Suherman, M.Kom
NIP. 19841010 201903 1 012

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : R. Abdurrosyid
NIM : 220605110042
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : Pemilihan Tingkat Kesulitan Pada *Endless Runner*
Game 'Avoid Najasa' Menggunakan Metode
AHP-VIKOR Berdasarkan Kemampuan Pemain

Dengan ini saya menyatakan bahwa skripsi yang saya susun merupakan hasil karya saya sendiri dan tidak mengandung unsur plagiarisme berupa pengambilan atau penggunaan data, tulisan, maupun gagasan pihak lain yang diakui sebagai karya saya, kecuali bagian-bagian yang telah dicantumkan sumber rujukannya secara jelas dalam daftar pustaka.

Apabila di kemudian hari terbukti bahwa skripsi ini mengandung unsur penjiplakan atau pelanggaran terhadap ketentuan akademik terkait keaslian karya ilmiah, maka saya bersedia menerima segala sanksi yang berlaku atas tindakan tersebut.

Malang, 22 Juni 2026
Yang membuat pernyataan,



R. Abdurrosyid
NIM.220605110042

MOTTO

Semua Pasti Ada Hikmahnya

HALAMAN PERSEMBAHAN

Alhamdulillah rabbil ‘alamin, segala puji dan syukur penulis panjatkan ke hadirat Allah *Subhānahu wa Ta ‘ālā* atas segala rahmat, karunia, dan petunjuk-Nya sehingga skripsi ini dapat diselesaikan dengan baik. Shalawat serta salam senantiasa tercurah kepada Rasulullah *Ṣallallāhu ‘Alaihi Wasallam* yang telah membimbing umat manusia dari masa kegelapan menuju cahaya Islam. Skripsi ini penulis dedikasikan kepada seluruh pihak yang telah memberikan dukungan, bantuan, serta kontribusi dalam proses penyusunan dan pelaksanaan penelitian ini.

Skripsi ini penulis persembahkan kepada kedua orang tua tercinta sebagai bentuk rasa hormat, cinta, dan terima kasih yang mendalam atas segala doa, kasih sayang, pengorbanan, serta dukungan yang senantiasa diberikan. Kepercayaan dan keyakinan yang mereka tanamkan menjadi sumber kekuatan bagi penulis dalam menempuh setiap langkah untuk meraih cita-cita. Penulis juga mempersembahkan karya ini kepada kakak-kakak dan adik-adik tercinta yang selalu memberikan semangat, motivasi, serta dukungan yang berarti selama proses penyusunan skripsi hingga dapat terselesaikan dengan baik.

Terakhir, penulis juga mempersembahkan kepada sahabat-sahabat seperjuangan skripsi dan teman-teman angkatan 2022 yang sudah memberi dukungan, semangat, motivasi, serta menemani dalam setiap proses, baik dalam suka maupun duka, hingga skripsi ini dapat terselesaikan.

Semoga segala bentuk bantuan, dukungan, dan kebaikan yang telah diberikan memperoleh balasan yang berlipat ganda dari Allah *Subhānahu wa Ta ‘ālā*. *Aamiin*.

KATA PENGANTAR

Alhamdulillah *rabbil ‘alamin*, segala puji dan syukur penulis panjatkan ke hadirat Allah *Subhānahu wa Ta ‘ālā* atas segala rahmat, nikmat, serta kasih sayang-Nya yang senantiasa menyertai sehingga penulis dapat menyelesaikan skripsi yang berjudul “PEMILIHAN TINGKAT KESULITAN PADA ENDLESS RUNNER GAME ‘AVOID NAJASA’ MENGGUNAKAN METODE AHP-VIKOR BERDASARKAN KEMAMPUAN PEMAIN” dengan baik. Shalawat serta salam semoga senantiasa tercurah kepada Nabi Muhammad *Ṣallallāhu ‘Alaihi Wasallam*, beserta keluarga, sahabat, dan seluruh pengikut beliau hingga akhir zaman. Semoga kita semua memperoleh syafaat beliau pada hari kiamat kelak. *Aamiin ya Rabbal ‘Alamin*. Dengan penuh rasa syukur, penulis menyampaikan terima kasih yang sebesar-besarnya kepada seluruh pihak yang telah memberikan bantuan, dukungan, bimbingan, doa, serta motivasi sehingga proses penyusunan skripsi ini dapat terselesaikan. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada:

1. Prof. Dr. Hj. Ilfi Nurdiana, M.Si., CAHRM., CRMP, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang, yang telah memberikan kesempatan serta menyediakan berbagai fasilitas yang mendukung penulis dalam menempuh pendidikan di universitas ini.
2. Dr. Agus Mulyono, M.Kes, selaku Dekan Fakultas Saintek Universitas Islam Negeri Malang, yang senantiasa memberikan dukungan, arahan, dan kebijakan yang menunjang kelancaran proses studi penulis.

3. Supriyono, M. Kom, selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang, atas motivasi dan arahan kepada mahasiswa-mahasiswanya untuk menyelesaikan tugas akhir.
4. Dr. Ir. Fresy Nugroho, M.T, IPM, ASEAN Eng selaku dosen pembimbing I atas segala arahan, bimbingan, dan solusi dalam melakukan penelitian dan kepenulisan skripsi ini.
5. Nur Fitriyah Ayu Tunjung Sari, M.Cs selaku dosen pembimbing II yang telah meluangkan waktu, memberikan bimbingan, arahan, serta masukan yang sangat berharga kepada penulis selama proses penyusunan skripsi hingga dapat terselesaikan dengan baik.
6. Fajar Rohman Hariri, M.Kom selaku Ketua Dewan Penguji dan Ahmad Fahmi Karami, M.Kom selaku dosen penguji I yang telah menguji serta memberikan masukan sehingga penulis dapat menuntaskan skripsi dengan baik.
7. Seluruh dosen, tenaga administrasi, laboran, dan mahasiswa Program Studi Teknik Informatika yang telah memberikan dukungan, bantuan, serta berbagai ilmu dan pengalaman yang bermanfaat selama proses perkuliahan maupun penyusunan skripsi ini.
8. Kedua orang tua dan seluruh keluarga penulis yang senantiasa memberikan doa, kasih sayang, dukungan, serta motivasi tanpa henti, sehingga penulis dapat terus bersemangat dan berusaha menyelesaikan skripsi ini dengan lancar dan sebaik-baiknya.

DAFTAR ISI

HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	v
MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xii
DAFTAR TABEL	xiii
ABSTRAK	xiv
ABSTRACT	xv
مستخلص البحث.....	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian	5
BAB II STUDI PUSTAKA	7
2.1 Penelitian Terkait	7
2.2 <i>Game</i>	10
2.3 <i>Multi-Criteria Decision Making (MCDM)</i>	13
2.4 <i>Analytical Hierarchy Process (AHP)</i>	15
2.5 <i>VIšekriterijumsko Kompromisno Rangiranje (VIKOR)</i>	19
2.6 <i>System Usability Scale (SUS)</i>	22
BAB III DESAIN DAN IMPLEMENTASI	26
3.1 Analisis dan Perancangan <i>Game</i>	26
3.1.1 Analisis <i>Endless Runner Game</i>	26
3.1.2 Perancangan <i>Endless Runner Game</i>	26
3.1.3 Rancangan Antarmuka	27
3.2 Desain Sistem.....	31
3.3 Data Alternatif.....	32
3.4 Data Kriteria.....	35
3.5 Pembobotan Kriteria menggunakan AHP	38
3.5.1 Penyusunan Matriks Perbandingan Berpasangan oleh Para Pengembang <i>Game</i>	39
3.5.2 Agregasi Penilaian Pengembang <i>Game</i> Menggunakan <i>Geometric Mean (AIJ)</i>	40
3.5.3 Matriks Perbandingan Berpasangan Kelompok	40
3.5.4 Menghitung Nilai Normalisasi Matriks Keputusan.....	41
3.5.5 Nilai <i>Eigen</i> / Bobot Kriteria.....	42
3.5.6 Nilai Konsistensi.....	42

3.6 Penentuan Alternatif terbaik menggunakan VIKOR	44
3.6.1 Matriks Keputusan.....	45
3.6.2 Melakukan Normalisasi Matriks Keputusan	46
3.6.3 Menghitung Nilai S dan R.....	47
3.6.4 Menghitung Nilai Indeks Kompromi	48
3.6.5 Melakukan Pengurutan Alternatif	50
3.7 Simulasi Uji Coba Penentuan Tingkat Kesulitan	51
3.8 Desain Pengujian Sistem.....	53
BAB IV HASIL DAN PEMBAHASAN	56
4.1 Implementasi Tampilan <i>User Interface</i>	56
4.2 Implementasi Perhitungan AHP.....	61
4.3 Implementasi Perhitungan VIKOR.....	63
4.4 Uji Coba Penentuan Tingkat Kesulitan.....	66
4.5 Hasil Pengujian	73
4.5.1 <i>Blackbox Testing</i>	73
4.5.2 <i>System Usability Scale (SUS)</i>	77
4.6 Integrasi dalam Islam	81
BAB V KESIMPULAN DAN SARAN	84
5.1 Kesimpulan	84
5.2 Saran.....	85
DAFTAR PUSTAKA	

DAFTAR GAMBAR

Gambar 2.1 Struktur Hierarki AHP	16
Gambar 3. 1 Diagram Menu.....	26
Gambar 3. 2 Rancangan Menu.....	27
Gambar 3. 3 Rancangan Pemilihan.....	27
Gambar 3. 4 Rancangan Opsi.....	28
Gambar 3. 5 Rancangan Kredit.....	28
Gambar 3. 6 Rancangan Jeda	29
Gambar 3. 7 Rancangan Tampilan <i>Gameplay</i>	29
Gambar 3. 8 Rancangan Skor.....	30
Gambar 3. 9 Diagram <i>Finite State Machine</i>	31
Gambar 4. 1 Tampilan Menu	57
Gambar 4. 2 Tampilan Pemilihan	57
Gambar 4. 3 Tampilan Opsi	58
Gambar 4. 4 Tampilan Kredit	58
Gambar 4. 5 Tampilan Jeda	59
Gambar 4. 6 Tampilan <i>Gameplay</i>	59
Gambar 4. 7 UI pada <i>Gameplay</i>	60
Gambar 4. 8 <i>Item</i> dalam <i>Gameplay</i>	60
Gambar 4. 9 Tampilan Skor	61
Gambar 4. 10 Data Pemain pada Percobaan Pertama	67
Gambar 4. 11 Hasil Sistem pada Percobaan Pertama	68
Gambar 4. 12 Data Pemain pada Percobaan Kedua.....	69
Gambar 4. 13 Hasil Sistem pada Percobaan Kedua	70
Gambar 4. 14 Data Pemain pada Percobaan Ketiga	71
Gambar 4. 15 Hasil Sistem pada Percobaan Ketiga.....	72
Gambar 4. 16 Statistik Jawaban SUS.....	78
Gambar 4. 17 Grafik Hasil SUS.....	81

DAFTAR TABEL

Tabel 2. 1 Penelitian terkait	9
Tabel 3. 1 Alternatif Tingkat Kesulitan	34
Tabel 3. 2 Data Kriteria Penentu Tingkat Kesulitan	35
Tabel 3. 3 Skala Penilaian Kriteria	36
Tabel 3. 4 Skala Penilaian Perbandingan Berpasangan	38
Tabel 3. 5 Matriks Perbandingan Berpasangan Pengembang <i>Game</i> 1	39
Tabel 3. 6 Matriks Perbandingan Berpasangan Pengembang <i>Game</i> 2	39
Tabel 3. 7 Matriks Perbandingan Berpasangan Pengembang <i>Game</i> 3	40
Tabel 3. 8 Matriks Perbandingan Berpasangan antar Kriteria	41
Tabel 3. 9 Matriks Normalisasi	41
Tabel 3. 10 Bobot kriteria	42
Tabel 3. 11 Perhitungan $\lambda_{\max}$	43
Tabel 3. 12 Matriks Keputusan Alternatif Tingkat Kesulitan	45
Tabel 3. 13 Normalisasi Matrik Keputusan Alternatif Tingkat Kesulitan	46
Tabel 3. 14 Hasil nilai R	47
Tabel 3. 15 Hasil nilai S	48
Tabel 3. 16 Nilai Max, Min dari Nilai S dan R	48
Tabel 3. 17 Nilai Indeks Kompromi (Q)	49
Tabel 3. 18 Pengurutan Alternatif	50
Tabel 3. 19 Daftar pernyataan Kuesioner SUS	54
Tabel 4. 1 Data Nilai Q Percobaan Pertama	68
Tabel 4. 2 Data Nilai Q Percobaan Kedua	70
Tabel 4. 3 Data Nilai Q Percobaan Pertama	72
Tabel 4. 4 Pengujian Halaman <i>Menu</i>	74
Tabel 4. 5 Pengujian Halaman Pemilihan	74
Tabel 4. 6 Pengujian Halaman <i>Gameplay</i>	75
Tabel 4. 7 Pengujian Panel Jeda	76
Tabel 4. 8 Pengujian Panel Skor	76
Tabel 4. 9 Daftar Pertanyaan SUS	77
Tabel 4. 10 Hasil Skor Asli SUS	78
Tabel 4. 11 Hasil Akhir Skor SUS	80

ABSTRAK

Abdurrosyid, R. 2026. **Pemilihan Tingkat Kesulitan Pada *Endless Runner Game* ‘Avoid Najasa’ Menggunakan Metode AHP-VIKOR Berdasarkan Kemampuan Pemain.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Ir. Fresy Nugroho, MT., IPM (II) Nur Fitriyah Ayu Tunjung Sari, M. Cs.

Kata kunci: *Endless Runner*, *Adaptive Difficulty*, *System Usability Scale*, AHP, VIKOR, Unity.

Penelitian ini bertujuan mengembangkan *game endless runner* berjudul ‘Avoid Najasa’ yang mampu menentukan tingkat kesulitan secara adaptif berdasarkan kemampuan pemain menggunakan metode *Analytical Hierarchy Process* (AHP) dan *VlseKriterijumska Optimizacija I Kompromisno Resenje* (VIKOR). Riset dilakukan dengan merancang dan mengimplementasikan *game* memakai Unity serta bahasa pemrograman C#, dengan AHP dipakai dalam menentukan bobot delapan kriteria performa pemain, yaitu jarak tempuh, durasi bertahan, persentase koin yang diperoleh, jumlah terkena najis, jumlah air yang dikumpulkan, persentase koin yang terlewat, jumlah najis yang berhasil dihindari, dan jumlah air yang terlewat, sedangkan VIKOR digunakan untuk menentukan alternatif tingkat kesulitan yang paling sesuai berdasarkan nilai kompromi. Hasil riset menunjukkan bahwasanya sistem berhasil menyesuaikan tingkat kesulitan permainan secara dinamis sesuai performa pemain sehingga menghasilkan pengalaman bermain yang lebih seimbang, personal, dan adaptif. Pengujian menggunakan *System Usability Scale* (SUS) pada 20 responden mendapat skor rata-rata bernilai 82 yang tergolong kategori sangat baik, sehingga mengindikasikan bahwasanya *game* yang dikembangkan dapat diterima dengan baik oleh pengguna dan penerapan metode AHP-VIKOR berhasil mencapai tujuan penelitian dalam menciptakan sistem permainan adaptif yang efektif.

ABSTRACT

Abdurrosyid, R. 2026. **Selection of Difficulty Levels in the Endless Runner Game ‘Avoid Najasa’ Using the AHP-VIKOR Method Based on Player Ability.** Undergraduate Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Advisors: (I) Dr. Ir. Fresy Nugroho, MT., IPM (II) Nur Fitriyah Ayu Tunjung Sari, M. Cs.

Key words: Endless Runner, Adaptive Difficulty, System Usability Scale, AHP, VIKOR, Unity.

This research aims to develop an endless runner *game* entitled Avoid Najasa capable of determining difficulty levels adaptively based on player abilities using the Analytical Hierarchy Process (AHP) and VlseKriterijumska Optimizacija I Kompromisno Resenje (VIKOR) methods. The research was conducted by designing and implementing the *game* using Unity and the C# programming language, where AHP was used to determine the weights of eight player performance criteria, namely travel distance, survival duration, percentage of collected coins, number of najasa collisions, amount of collected water, percentage of missed coins, number of successfully avoided najasa, and amount of missed water, while VIKOR was used to determine the most appropriate difficulty level alternative based on the compromise value. The results show that the system successfully adjusts the *game* difficulty dynamically according to player performance, resulting in a more balanced, personalized, and adaptive gaming experience. Testing using the System Usability Scale (SUS) involving 20 respondents obtained an average score of 82, which is categorized as excellent, indicating that the developed *game* was well accepted by users and that the implementation of the AHP-VIKOR methods successfully achieved the research objective of creating an effective adaptive gaming system.

مستخلص البحث

عبد الرشيد، ر. ٢٠٢٦. اختيار مستوى الصعوبة في ألعاب الجري اللاهوائي "Avoid Najasa" باستخدام طريقة-AHP VIKOR على أساس قدرة اللاعب. البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف الأول: د. فريسي نوغروهو، الماجستير؛ المشرف الثاني: نور فطرية أيو تونجونج ساري، الماجستير.

الكلمات الرئيسية: جري لانهوائي، صعوبة تكتيفية، مقياس سهولة استخدام نظام، AHP، VIKOR، Unity.

" قدرة على تحديد مستوى الصعوبة بشكل تكتيفي Avoid Najasa يهدف هذا البحث إلى تطوير ألعاب الجري اللاهوائي بعنوان " (تم VIKOR) وتحسين المعايير المتعددة والحلول التسوية (AHP بناءً على قدرة اللاعب باستخدام طريقتي عملية التحليل الهرمي) لتحديد وزن ثمانية AHP، حيث تُستخدم C# ولغة البرمجة Unity إجراء البحث من خلال تصميم وتنفيذ اللعبة باستخدام معايير لأداء اللاعب، وهي المسافة المقطوعة، مدة البقاء، نسبة العملات التي تم جمعها، عدد مرات التعرض للنجاسة، كمية الماء التي تم جمعها، نسبة العملات التي فُقدت، عدد النجاسات التي تم تجنبها بنجاح، وكمية الماء التي فُقدت، بينما تُستخدم VIKOR لتحديد بدائل مستوى الصعوبة الأنسب وفقًا لقيمة التسوية. أظهرت نتائج البحث أن النظام نجح في تعديل مستوى صعوبة اللعبة بشكل ديناميكي وفق أداء اللاعب، مما أدى إلى تجربة لعب أكثر توازنًا وشخصية وتكيفًا. وأجريت اختبارات باستخدام مقياس قابلية (على ٢٠ مشاركًا، حيث حصلت الدرجة المتوسطة على 82، مما يصنف ضمن فئة ممتازة، مما يدل على أن SUS استخدام النظام) نجح في تحقيق أهداف البحث في إنشاء AHP-VIKOR اللعبة المطورة يمكن قبولها جيدًا من قبل المستخدمين وأن تطبيق طريقة نظام لعبة تكتيفي فعال

BAB I

PENDAHULUAN

1.1 Latar Belakang

Game telah menjadi bagian integral dari kehidupan modern dan budaya masyarakat masa kini. Perkembangan teknologi mendorong peningkatan kompleksitas serta daya tarik *game* yang kini mencakup berbagai *genre*, dari permainan sederhana hingga aksi dan petualangan yang menantang. Sebagai bentuk hiburan, *game* memadukan unsur ilmiah, teknologi, dan kreativitas dalam satu kesatuan (Fencott et al., 2012).

Satu diantara faktor yang sangat berpengaruh pada kualitas pengalaman bermain yaitu tingkat kesulitan permainan. Tingkat kesulitan yang terlalu mudah bisa menyebabkan pemain merasa bosan karena kurangnya tantangan, sedangkan tingkat kesulitan yang sangat tinggi bisa menjadikan frustrasi dan mengurangi motivasi untuk melanjutkan permainan (Karavidas et al., 2022). Oleh sebab itu, penyesuaian tingkat kesulitan yang sesuai dengan kemampuan pemain menjadi aspek penting pada pengembangan *game*. Ketidaksesuaian antara kemampuan pemain dan tantangan yang diberikan dapat mengurangi keterlibatan pemain serta menurunkan kualitas pengalaman bermain (Nugroho et al., 2023).

Dalam perspektif Al-Qur'an, konsep “kesulitan” mempunyai arti bahwa Allah *Subhanahu wa Ta'ala* tak akan memberi ujian di luar batas kesanggupan makhluk-Nya. Hal ini dijelaskan dalam Surah Al-Baqarah ayat 286 yang

menyatakan bahwa setiap beban yang ditanggung manusia sudah disesuaikan dengan kemampuannya.

...لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya...”
(Q.S Al-Baqarah: 286)

Berdasarkan penafsiran dalam Tafsir *Al-Wajiz*, ayat ini menjelaskan bahwasanya Allah memberi beban, kewajiban, dan ujian kepada manusia sesuai dengan kesanggupannya. Setiap perintah dan larangan yang ditetapkan tidak akan melampaui batas kesanggupan manusia, sehingga tak ada alasan untuk berputus asa dalam menjalani ujian hidup. Selain itu, ayat ini juga memiliki arti bahwasanya manusia tetap bertanggung jawab atas apa ia usahakan, baik berupa hal baik maupun hal buruk.

Dengan demikian, konsep kesulitan dalam ayat ini bukan hanya menunjukkan adanya batas kemampuan manusia, tetapi juga menegaskan kasih sayang serta keadilan Allah dalam menetapkan setiap ujian. Hal ini memberi dorongan untuk manusia agar tetap berusaha, bersabar, dan bertawakal dalam menghadapi berbagai tantangan kehidupan.

Perbedaan kemampuan, pengalaman, dan gaya bermain menyebabkan setiap pemain memiliki kebutuhan tingkat kesulitan yang berbeda. Oleh karena itu, banyak penelitian mengembangkan pendekatan *Dynamic Difficulty Adjustment* (DDA), yaitu mekanisme yang memungkinkan tingkat kesulitan permainan menyesuaikan diri secara dinamis sesuai dengan performa pemain. *Genre Endless Runner* dipilih sebagai objek penelitian karena memiliki mekanisme permainan

yang sederhana, mudah dipahami, serta memungkinkan peningkatan tantangan secara berkelanjutan. Karakteristik permainan yang tidak memiliki batas akhir (*infinite gameplay*) menjadikan genre ini sesuai untuk penerapan sistem penyesuaian tingkat kesulitan secara dinamis berdasarkan kondisi pemain (Fahrullazi et al., 2019).

Guna menetapkan tingkat kesukaran yang paling cocok dengan kondisi pemain, riset ini mengombinasikan metode *Analytic Hierarchy Process* (AHP) dan *VlseKriterijumska Optimizacija I Kompromisno Resenje* (VIKOR). AHP dimanfaatkan dalam menghitung bobot setiap parameter performa pemain, sedangkan VIKOR digunakan untuk memilih tingkat kesulitan yang paling sesuai berdasarkan nilai kompromi dari berbagai kriteria yang dipertimbangkan. Pendekatan ini memungkinkan penyesuaian tingkat kesulitan dilakukan secara lebih objektif dan adaptif (Armonitha & Rahmansyah, 2024). Metode AHP–VIKOR banyak dimanfaatkan untuk pengambilan keputusan multikriteria sebab kemampuannya menghasilkan solusi kompromi yang stabil dalam kondisi konflik antar kriteria. Dibandingkan metode seperti SAW dan TOPSIS, VIKOR lebih menekankan pada pencapaian solusi kompromi terbaik, sementara TOPSIS dan SAW memiliki sensitivitas yang lebih tinggi terhadap perubahan data dan proses normalisasi (Opricovic & Tzeng, 2004; Mardani et al., 2015).

Dalam penelitian ini, metode tersebut diterapkan pada *game Endless Runner* berjudul ‘Avoid Najasa’, yang dikembangkan memakai Unity dengan bahasa pemrograman C#. Sistem pada *game* ini dirancang menerapkan *Dynamic Difficulty Adjustment* (DDA) berbasis AHP–VIKOR agar tingkat kesulitan dapat beradaptasi

secara proporsional terhadap performa pemain. Dengan demikian, permainan diharapkan mampu memberi pengalaman bermain yang lebih seimbang, menantang, serta menarik bagi berbagai tingkat kemampuan pemain.

1.2 Rumusan Masalah

Mengacu pada apa yang telah dipaparkan dalam latar belakang, maka dapat diambil rumusan masalah penelitian ini yaitu bagaimana membuat *game* yang dapat menentukan tingkat kesulitan yang sesuai dengan kemampuan player menggunakan metode AHP-VIKOR ?

1.3 Batasan Masalah

Agar riset ini terarah serta tak melebar dari tujuan utama, maka diberlakukan batasan-batasan masalah seperti berikut:

1. Permainan yang dikembangkan merupakan *game single player* bertipe *endless runner* berjudul ‘Avoid Najasa’, yang berfokus pada penghindaran rintangan untuk memperoleh skor tertinggi.
2. Mekanisme pergerakan karakter terbatas pada tiga aksi utama, yaitu bergerak ke kanan, ke kiri, dan melompat.
3. Penentuan tingkat kesulitan permainan didasarkan pada kemampuan pemain yang diukur melalui beberapa indikator performa, yaitu Jarak, Durasi Bertahan, Koin yang diperoleh, Jumlah menabrak rintangan, Jumlah air yang diperoleh, Koin yang terlewat, Jumlah rintangan yang berhasil dihindari, dan Jumlah air yang berhasil diambil.

4. Durasi setiap *level* maksimal hanya 2 menit untuk mengukur performa pemain.
5. Metode penentuan tingkat kesulitan adaptif menggunakan kombinasi *Analytical Hierarchy Process* (AHP) guna menetapkan bobot kriteria serta VIKOR untuk memilih alternatif tingkat kesulitan yang paling sesuai.
6. Penelitian ini tidak membahas aspek grafis, animasi, maupun desain artistik secara mendalam, tetapi berfokus pada mekanisme adaptasi tingkat kesulitan permainan.
7. Implementasi sistem dengan memakai *game engine Unity* dengan bahasa pemrograman C#, dengan pengujian dilakukan pada platform *desktop*.

1.4 Tujuan Penelitian

Adapun tujuan dari riset yang dijalankan yaitu mengembangkan *game* yang dapat menentukan tingkat kesulitan berdasarkan kemampuan pemain menggunakan metode AHP-VIKOR.

1.5 Manfaat Penelitian

Riset ini harapannya bisa memberi kontribusi yang bermanfaat, baik dari aspek teoretis maupun praktis, yang dijabarkan seperti berikut:

1. Manfaat Teoretis
 - a. Berkontribusi terhadap pengembangan ilmu pengetahuan di bidang *game development* serta sistem pendukung keputusan, terutama pada implementasi metode AHP-VIKOR pada penentuan tingkat kesulitan adaptif.

- b. Menjadi acuan untuk penelitian berikutnya yang ingin mengembangkan sistem *adaptive difficulty* berbasis kemampuan pemain dengan pendekatan multi-kriteria.
- c. Memperkuat kajian mengenai integrasi metode *Analytical Hierarchy Process* (AHP) dan VIKOR.

2. Manfaat Praktis

- a. Membantu pengembang *game* dalam merancang sistem kesulitan yang bisa menyesuaikan diri dengan kemampuan pemain dengan otomatis, sehingga meningkatkan pengalaman bermain (*player experience*).
- b. Menjadi acuan bagi peneliti atau mahasiswa lain dalam mengimplementasikan metode pengambilan keputusan multi-kriteria pada konteks hiburan digital atau sistem adaptif lainnya.
- c. Memberikan dasar bagi pengembangan *game* edukatif maupun komersial yang lebih inklusif, menantang, dan sesuai dengan tingkat kemampuan masing-masing pemain.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terkait

Berbagai riset terdahulu sudah dilakukan untuk mengembangkan sistem pengambilan keputusan dalam penentuan tingkat kesulitan maupun penyesuaian elemen permainan berdasarkan kemampuan pemain. Masing-masing riset memakai teknik pengambilan keputusan multikriteria (*Multi-Criteria Decision Making/MCDM*) yang berbeda, dengan tujuan meningkatkan pengalaman bermain (*player experience*) secara adaptif dan dinamis.

Penelitian yang dilakukan oleh Ardianto pada tahun 2024, menerapkan metode *Complex Proportional Assessment* (COPRAS) untuk menentukan tingkat kesulitan dalam sebuah permainan edukatif berbasis nilai-nilai religius. Kriteria yang digunakan cukup kompleks, meliputi aspek fisik, psikologis, hingga perilaku pemain seperti jarak tempuh, durasi bertahan, rasa malas, hingga interaksi dengan objek seperti Al-Qur'an dan koin. Hasil penelitian tersebut menunjukkan adanya rentang tingkat kesulitan yang berbeda di antara tiga golongan usia pemain, yang menegaskan bahwa metode pengambilan keputusan proporsional dapat digunakan untuk menilai performa pemain secara kontekstual.

Selanjutnya, penelitian oleh Hartanti pada tahun 2025, memakai metode VIKOR guna menyesuaikan skenario permainan berdasarkan performa pemain. Kriteria yang digunakan mencakup jumlah tabrakan, karung beras yang diperoleh, total skor, serta total rintangan yang berhasil dilewati. Hasil riset mengindikasikan

bahwasanya sistem berbasis VIKOR bisa menyesuaikan tingkat kesulitan secara otomatis, sehingga permainan menjadi lebih seimbang dan sesuai dengan kemampuan pemain. Penelitian ini memperlihatkan bahwa pendekatan kompromi dalam MCDM efektif untuk sistem adaptif dalam konteks permainan.

Penelitian oleh Nugroho et al. (2025) mengimplementasikan metode MOORA (*Multi-Objective Optimization on the basis of Ratio Analysis*) dalam menganalisis keterlibatan pemain dalam lingkungan pembelajaran virtual. Kriteria yang digunakan meliputi waktu bermain, poin yang diperoleh, jarak tempuh, dan akurasi jawaban. Hasil riset membuktikan bahwasanya metode ini bisa menciptakan pengalaman belajar yang lebih interaktif dan emosional untuk pemain. Meskipun fokusnya pada konteks edukatif, pendekatan MOORA menunjukkan potensi untuk menilai performa pemain secara objektif dalam *game* berbasis simulasi.

Sementara itu, Prakasa pada tahun 2024 memakai metode ORESTE (*Order Preference by Similarity to Ideal Solution*) pada pemilihan karakter *game* berdasarkan berbagai kriteria seperti gerakan, stamina, pertahanan, umur, dan kekuatan. Penelitian ini membuktikan bahwa ORESTE efektif dimanfaatkan dalam pengambilan keputusan yang mempertimbangkan berbagai kriteria dengan bobot yang dinamis. Pendekatan ini relevan bagi penelitian yang melibatkan proses seleksi berbasis banyak faktor dalam lingkungan permainan.

Adapun penelitian oleh Hidayat pada tahun 2024 menerapkan metode VIKOR untuk memberikan rekomendasi terhadap atribut dalam permainan, yaitu pemilihan sepatu yang sesuai dengan preferensi pemain berdasarkan bonus koin,

peningkatan skor, kecepatan, dan durasi power-up. Hasil riset mengindikasikan bahwasanya VIKOR mampu memberikan hasil rekomendasi yang optimal. Kombinasi kedua metode ini dinilai efektif untuk menyelesaikan permasalahan pengambilan keputusan yang membutuhkan pertimbangan rasional dan kompromi yang seimbang.

Mengacu pada berbagai penelitian tersebut, bisa ditarik kesimpulan bahwasanya metode-metode MCDM, VIKOR, MOORA, ORESTE, dan COPRAS, mempunyai potensi besar untuk diimplementasikan dalam konteks pengambilan keputusan adaptif di dunia permainan. Namun, belum banyak riset yang secara spesifik menerapkan kombinasi AHP-VIKOR untuk menentukan tingkat kesulitan adaptif pada *game* bergenre *endless runner*. Oleh sebab itu, riset ini bertujuan untuk mengisi celah tersebut dengan mengimplementasikan metode AHP-VIKOR dalam *game* 'Avoid Najasa', sehingga sistem bisa menyesuaikan tingkat kesulitan permainan berdasarkan kemampuan aktual pemain.

Adapun ringkasan beberapa penelitian terkait ditampilkan pada Tabel 2.1 berikut.

Tabel 2. 1 Penelitian terkait

No	Referensi	Acuan Kriteria	Metode	Hasil
1	(Ardianto, 2024)	Jarak, Durasi Bertahan, Nyawa, Rasa Malas, Al-Qur'an, Koin, Menabrak, Haus, Minuman, Melewati Al-Qur'an, Melewati Koin	<i>Complex Proportional Assessment</i> (CORPAS)	Hasil uji menghasilkan adanya rentang distribusi tingkat kesulitan yang berbeda pada ketiga golongan usia yang diteliti.
2	(Hartanti, 2025)	jumlah tabrakan dengan pencuri, jumlah karung beras yang	VIKOR	Sistem dapat melakukan penyesuaian

No	Referensi	Acuan Kriteria	Metode	Hasil
		dikumpulkan, jumlah skor, jumlah rintangan yang dilewati, dan jumlah pencuri yang dihindari.		skenario dengan otomatis sehingga menghasilkan pengalaman bermain yang lebih proporsional serta selaras dengan keterampilan setiap pemain.
3	(Nugroho et al., 2025)	Playing Time, Point Player, Distance Traveled, Total Correct, Total Wrong, Remaining HP	MOORA	Membuktikan bahwasanya pemain merasa terlibat secara emosional dan hadir di dunia pembelajaran virtual.
4	(Prakasa, 2024)	Tinggi Badan, Nyawa, Gerakan, Berat, Pertahanan, Stamina, Kelincahan, Umur, Gemuk, Kekuatan, Keyakinan	ORESTE	Metode ORESTE efektif digunakan untuk pemilihan karakter dalam <i>game</i> dengan mempertimbangkan kriteria ganda dan variasi bobot waktu.
5	(Hidayat, 2024)	Bonus koin, Peningkatan skor, Peningkatan jarak, Durasi <i>power-up</i> , Kekuatan lompat, Kekuatan memunduk, Kekuatan berpindah, Kecepatan	VIKOR	Hasil riset mengindikasikan bahwasanya metode VIKOR efektif untuk memberi rekomendasi sepatu yang cocok dengan keinginan pemain.

2.2 Game

Game merupakan salah satu bentuk media interaktif yang menggabungkan aspek hiburan, simulasi, dan pembelajaran melalui keterlibatan aktif pemain dalam suatu sistem yang memiliki aturan tertentu. Menurut Tekinbas dan Zimmerman

(2003), *game* bisa diartikan sebagai sistem di mana pemain terlibat dalam masalah buatan yang diatur oleh aturan dan memperoleh hasil yang bisa diukur. Sementara itu, menurut Juul (2011), *game* merupakan sistem berbasis aturan dengan hasil yang bisa berubah serta terkalkulasi, di mana pemain berusaha memengaruhi hasil tersebut melalui keterlibatan aktif dalam permainan. Dari beberapa pengertian tersebut, bisa ditarik kesimpulan bahwasanya *game* bukan hanya sarana rekreasi, namun juga media kompleks yang melibatkan interaksi antara sistem, aturan, dan pemain, yang dapat digunakan untuk mengembangkan kemampuan kognitif maupun emosional.

Dalam pengembangannya, *game* adalah sistem yang tersusun dari berbagai elemen yang saling berinteraksi serta berpengaruh terhadap kualitas pengalaman bermain. Menurut Schell (2008), desain *game* mencakup berbagai aspek seperti mekanika permainan (*mechanics*), aturan (*rules*), serta pengalaman bermain yang dihasilkan dari interaksi pemain dengan sistem *game* (*gameplay experience*). Mekanika permainan mengacu pada sistem dan prosedur yang mengatur bagaimana *game* berjalan, termasuk sistem skor, pergerakan karakter, dan respons terhadap input pemain. Aturan berfungsi sebagai batasan yang mengarahkan tindakan pemain dalam konteks permainan. Sementara itu, *gameplay* merupakan pengalaman yang muncul dari interaksi antara pemain dan sistem permainan secara keseluruhan. Elemen-elemen tersebut saling berkaitan dan perlu dirancang secara seimbang agar *game* mampu memberikan pengalaman yang menarik, menantang, dan bermakna.

Salah satu genre yang banyak dikembangkan dalam konteks hiburan maupun edukasi adalah *endless runner*. Genre ini menekankan pada permainan yang tidak memiliki akhir tetap, di mana pemain mengendalikan karakter yang terus bergerak maju dan berusaha bertahan selama mungkin dengan menghindari berbagai rintangan. Karakteristik utama *game endless runner* adalah menuntut kemampuan reaksi cepat dan refleks pemain dalam merespons rintangan yang muncul secara terus-menerus (Arfiningtyas et al., 2017). Selain itu, genre ini menerapkan peningkatan tingkat kesulitan secara bertahap seiring progres permainan atau durasi bertahan pemain. Contoh populer dari genre ini adalah ‘Temple Run’ dan ‘Subway Surfers’, yang keduanya menuntut pemain untuk mempertahankan fokus dan ketepatan gerak. Dalam konteks penelitian ini, konsep *endless runner* digunakan karena sifatnya yang adaptif dan dinamis, sehingga sangat relevan untuk menguji mekanisme penyesuaian tingkat kesulitan berdasarkan performa pemain.

Dalam kaitannya dengan penelitian ini, *game* ‘Avoid Najasa’ dirancang sebagai *endless runner* yang tak hanya berfungsi menjadi sarana hiburan, namun juga sebagai media dalam menerapkan sistem adaptif berbasis kemampuan pemain. Melalui penerapan metode AHP-VIKOR, tingkat kesulitan permainan dapat disesuaikan dengan kinerja pemain secara objektif berdasarkan kriteria tertentu, seperti kecepatan, refleks, dan ketepatan tindakan. Pendekatan ini diharapkan dapat meningkatkan keterlibatan pemain serta menjaga keseimbangan antara tantangan dan kemampuan, sesuai dengan prinsip *flow theory* dalam desain *game* (Csikszentmihalyi, 2009). Dengan demikian, teori-teori tentang *game* tidak hanya

menjadi landasan konseptual, tetapi juga berperan langsung dalam mendukung perancangan sistem adaptif yang diterapkan dalam ‘Avoid Najasa’.

2.3 Multi-Criteria Decision Making (MCDM)

Multi-Criteria Decision Making (MCDM) adalah metode dalam pengambilan keputusan yang dipakai dalam mengevaluasi alternatif mengacu pada lebih dari satu kriteria yang saling bertentangan guna memperoleh alternatif terbaik (Keeney & Raiffa, 1993; Mardani et al., 2015). Pendekatan ini berperan penting dalam menangani permasalahan kompleks yang tidak dapat diselesaikan hanya dengan satu ukuran atau dimensi penilaian. Sementara itu, menurut Triantaphyllou (2000), MCDM dipakai dalam menuntaskan permasalahan pengambilan keputusan yang terdiri dari beberapa kriteria, baik kuantitatif maupun kualitatif, melalui proses pembobotan dan pemeringkatan alternatif sehingga diperoleh hasil keputusan yang lebih sistematis dan terukur. Dengan demikian, MCDM tak hanya membantu menentukan alternatif terbaik, namun juga menjelaskan sejauh mana setiap alternatif memenuhi tujuan yang diinginkan.

Dalam praktiknya, terdapat berbagai metode yang dikembangkan di bawah payung MCDM, masing-masing dengan karakteristik dan keunggulannya sendiri. Metode Analytic Hierarchy Process (AHP) yang diperkenalkan oleh Saaty (2008) menekankan pada perbandingan berpasangan untuk menentukan bobot relatif antar kriteria. Metode VIKOR (*Vlsekriterijumska Optimizacija I Kompromisno Resenje*) yang diperkenalkan oleh Opricovic dan Tzeng (2004) berfokus pada pencarian solusi kompromi terbaik berdasarkan jarak relatif setiap alternatif pada solusi ideal. Selain itu, ada metode lain seperti *Technique for Order Preference by Similarity to*

Ideal Solution (TOPSIS) yang menilai kedekatan alternatif pada solusi ideal negatif dan positif, serta MOORA (*Multi-Objective Optimization on the Basis of Ratio Analysis*) yang menggunakan rasio normalisasi dalam menentukan urutan preferensi. Metode COPRAS (*Complex Proportional Assessment*) juga banyak digunakan untuk mengevaluasi alternatif secara proporsional berdasarkan tingkat kepentingannya. Beragamnya metode ini mengindikasikan bahwasanya MCDM mempunyai fleksibilitas tinggi dalam menyesuaikan pendekatan dengan konteks dan kebutuhan permasalahan yang dihadapi.

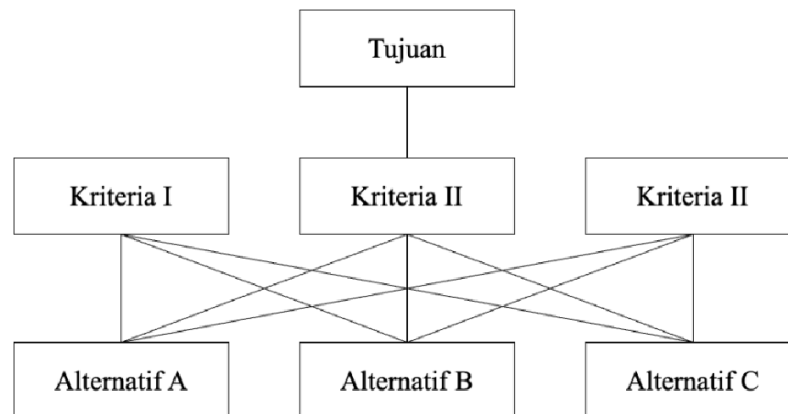
Kelebihan utama metode MCDM terletak pada keandalannya dalam mengakomodasi kompleksitas situasi pengambilan keputusan yang terdiri dari banyak kriteria dengan bobot berbeda. Menurut Hwang dan Yoon (1981), MCDM dipakai dalam memudahkan pengambilan keputusan mengacu pada beberapa kriteria yang dinilai secara sistematis, sehingga memperoleh alternatif terbaik berdasarkan pendekatan matematis yang terstruktur. Selain itu, MCDM memberikan transparansi dalam proses penilaian dan hasil akhir yang dapat dijustifikasi secara logis, sehingga sangat berguna dalam lingkungan yang menuntut keputusan berbasis data dan analisis. Dalam konteks sistem cerdas, pendekatan ini juga mendukung penerapan algoritma adaptif yang dapat mengubah keputusan berdasarkan kondisi atau masukan baru. Oleh karena itu, MCDM banyak diterapkan dalam bidang manajemen, rekayasa, pendidikan, hingga pengembangan sistem berbasis *game* yang memerlukan pengambilan keputusan dinamis.

Dalam riset ini, pendekatan MCDM dipakai sebagai dasar dalam menentukan tingkat kesulitan pada *game* 'Avoid Najasa' berdasarkan kemampuan

pemain. Pemilihan kombinasi metode AHP dan VIKOR dilakukan karena keduanya saling melengkapi, AHP berfungsi dalam menghitung bobot kepentingan setiap kriteria yang mencerminkan faktor kemampuan pemain, sedangkan VIKOR digunakan untuk menilai alternatif tingkat kesulitan berdasarkan bobot tersebut dan menghasilkan solusi kompromi terbaik. Melalui integrasi kedua metode ini, sistem dapat dengan adaptif menyesuaikan tingkat kesulitan permainan supaya seimbang dengan performa pemain. Dengan demikian, penerapan MCDM dalam konteks ini tak hanya meningkatkan pengalaman bermain, tetapi juga mendukung terciptanya *game* yang lebih responsif dan inklusif terhadap kemampuan individu pemain.

2.4 Analytical Hierarchy Process (AHP)

Analytical Hierarchy Process (AHP) termasuk salah satu metode pengambilan keputusan multikriteria yang diperkenalkan oleh Thomas L. Saaty pada tahun 1980. Metode ini didesain guna memudahkan pengambil keputusan dalam memecahkan permasalahan kompleks yang terdiri dari banyak kriteria dengan cara menyusunnya ke dalam bentuk hierarki. Menurut Saaty (2008), AHP bekerja berdasarkan prinsip perbandingan berpasangan (*pairwise comparison*) yang memungkinkan pengambil keputusan untuk menilai tingkat kepentingan relatif antar kriteria atau alternatif dengan skala tertentu. Melalui pendekatan ini, masalah yang bersifat kualitatif bisa diubah menjadi bentuk kuantitatif sehingga mempermudah tahap analisis dan penentuan prioritas. Struktur hierarki pada metode AHP tersusun atas tiga tingkatan utama, yakni tujuan sebagai level tertinggi, kriteria dan subkriteria pada level tengah, serta alternatif keputusan yang berada pada level paling bawah.



Gambar 2.1 Struktur Hierarki AHP

Dalam konteks riset ini, langkah-langkah AHP dipakai dalam menghitung bobot kepentingan dari berbagai aspek kemampuan pemain yang menjadi dasar penentuan tingkat kesulitan dalam *game* ‘Avoid Najasa’.

1. Menentukan Tujuan dan Kriteria Keputusan

Langkah awal dalam metode AHP yaitu mengidentifikasi tujuan utama dari proses pengambilan keputusan dan menetapkan kriteria serta alternatif yang relevan. Tujuan dalam riset ini yaitu guna menyesuaikan level kesukaran yang sesuai dengan keterampilan pemain. Kriteria yang digunakan menggambarkan faktor-faktor performa pemain seperti kecepatan reaksi, tingkat ketepatan, waktu bertahan, serta perolehan skor. Semua kriteria tersebut disusun dalam bentuk hierarki, di mana tujuan utama berada pada tingkat teratas, diikuti oleh kriteria di tingkat menengah, dan alternatif tingkat kesulitan pada tingkat terbawah.

2. Menyusun Matriks Perbandingan Berpasangan

Tahap selanjutnya yaitu menyusun matriks perbandingan berpasangan guna mengetahui tingkat kepentingan relatif antar kriteria. Setiap pasangan kriteria dibandingkan memakai skala penilaian 1 sampai 9 sebagaimana dikemukakan oleh

Saaty (2008), di mana nilai 1 mengindikasikan kepentingan yang sama, sedangkan nilai 9 mengindikasikan bahwa satu kriteria jauh lebih penting dibanding yang lain.

Matriks perbandingan berpasangan bisa dituliskan sebagai berikut:

$$A = \begin{bmatrix} 1 & a_{12} & a_{13} & \dots & a_{1n} \\ \frac{1}{a_{12}} & 1 & a_{23} & \dots & a_{2n} \\ \frac{1}{a_{13}} & \frac{1}{a_{23}} & 1 & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_{1n}} & \frac{1}{a_{2n}} & \frac{1}{a_{3n}} & \dots & 1 \end{bmatrix} \quad (2.1)$$

Keterangan:

A : Matriks perbandingan berpasangan dengan dimensi $n \times n$
 a_{ij} : Tingkat kepentingan kriteria ke- i terhadap kriteria ke- j
 n : Jumlah total kriteria
 $\frac{1}{a_{ij}}$: Hubungan timbal balik berlaku

3. Menghitung Bobot Prioritas dan Nilai Eigen

Setelah matriks perbandingan berpasangan terbentuk, dilakukan perhitungan bobot prioritas setiap kriteria melalui proses normalisasi matriks. Normalisasi dilakukan dengan membagi setiap nilai pada kolom terhadap total nilai kolom yang bersangkutan. Bobot prioritas lalu diperoleh dengan mengkalkulasi rata-rata nilai pada setiap baris hasil normalisasi menggunakan rumus berikut:

$$w_i = \frac{1}{n} \sum_{j=1}^n \frac{a_{ij}}{\sum_{i=1}^n a_{ij}} \quad (2.2)$$

Keterangan:

w_i : bobot prioritas kriteria ke- i
 a_{ij} : nilai perbandingan antara kriteria ke- i dan ke- j
 n : jumlah kriteria yang dibandingkan

Selain itu, vektor bobot juga dapat diperoleh dari persamaan eigen berikut:

$$Aw = \lambda_{max} w \quad (2.3)$$

Keterangan:

A : matriks perbandingan berpasangan
 w : vektor bobot prioritas yang dicari

λ_{max} : nilai eigen maksimum dari matriks A

4. Menghitung Rasio Konsistensi (CR dan CI)

Tahap penting berikutnya adalah menguji tingkat konsistensi dari penilaian yang sudah dilakukan. Hal ini memiliki tujuan guna mengetahui bahwasanya perbandingan antar kriteria tidak bersifat kontradiktif. Nilai indeks konsistensi (*Consistency Index*, CI) dikalkulasi memakai rumus:

$$CI = \frac{\lambda_{max} - n}{n - 1} \quad (2.4)$$

Selanjutnya skor rasio konsistensi (CR) dihitung dengan membandingkan nilai CI dengan *Random Index* (RI) menggunakan rumus:

$$CR = \frac{CI}{RI} \quad (2.5)$$

Keterangan:

λ_{max} : nilai eigen maksimum dari matriks perbandingan

n : jumlah kriteria

RI : *Random Index* yang nilainya berdasarkan n

Jika $CR \leq 0,1$ maka tingkat konsistensi dianggap bisa disetujui.

5. Evaluasi Hasil dan Interpretasi Bobot

Tahap terakhir adalah evaluasi dan interpretasi hasil perhitungan bobot. Bobot yang sudah didapat menunjukkan tingkat kepentingan relatif setiap kriteria terhadap tujuan pengambilan keputusan. Dalam konteks penelitian ini, bobot yang diperoleh dari metode AHP akan dipakai sebagai input pada metode VIKOR guna menetapkan alternatif tingkat kesulitan yang paling cocok dengan keterampilan pemain. Dengan demikian, AHP berfungsi sebagai dasar pembobotan yang memastikan bahwa keputusan akhir mengenai tingkat kesulitan dalam *game* ‘Avoid Najasa’ dilakukan secara objektif, rasional, dan konsisten berdasarkan analisis matematis yang terukur.

2.5 *VIšekriterijumsko Kompromisno Rangiranje (VIKOR)*

Metode VIKOR (*VIšekriterijumsko KOmpromisno Rangiranje*) dipaparkan oleh Opricovic dan Tzeng (2004) sebagai satu di antara pendekatan dalam pengambilan keputusan multi-kriteria (*Multi-Criteria Decision Making/MCDM*) yang berfokus pada pencarian solusi kompromi. VIKOR berupaya menentukan alternatif terbaik dari sejumlah alternatif yang saling bersaing berdasarkan kriteria-kriteria yang saling bertentangan. Pendekatan ini menekankan konsep kompromi, yaitu memilih solusi yang paling mendekati ideal, bukan hanya yang unggul pada satu aspek, tetapi memberikan keseimbangan antar seluruh kriteria yang dipertimbangkan (Opricovic dan Tzeng, 2004). Dalam konteks riset ini, metode VIKOR dimanfaatkan dalam menentukan tingkat kesulitan permainan berdasarkan hasil pembobotan dari metode AHP, sehingga setiap tingkat kesulitan mencerminkan kompromi antara berbagai kemampuan pemain.

Berikut langkah-langkah utama dalam metode VIKOR:

1. Menentukan Solusi Ideal Positif dan Solusi Ideal Negatif

Langkah pertama dalam metode VIKOR yaitu menetapkan solusi ideal positif f_i^+ serta solusi ideal negatif f_i^- pada setiap kriteria. Solusi ideal positif merepresentasikan nilai terbaik dari suatu kriteria, sedangkan solusi ideal negatif merepresentasikan nilai terburuk dari kriteria yang sama. Rumus penentuannya adalah seperti berikut:

$$f_i^+ = \max_j f_{ij} \quad (2.6)$$

$$f_i^- = \min_j f_{ij} \quad (2.7)$$

Keterangan:

f_i^+ : nilai ideal positif (terbaik) dari kriteria ke- i
 f_i^- : nilai ideal negatif (terburuk) dari kriteria ke- i
 f_{ij} : nilai kinerja alternatif ke- j terhadap kriteria ke- i

Langkah ini memiliki tujuan guna menentukan batas referensi yang akan digunakan dalam penilaian jarak setiap alternatif terhadap kondisi ideal, sehingga bisa diketahui seberapa jauh sebuah alternatif dari hasil terbaik dan terburuk.

2. Menghitung Jarak pada Solusi Ideal Positif dan Negatif

Setelah solusi ideal positif serta negatif ditetapkan, langkah selanjutnya yaitu mengkalkulasi jarak antara setiap alternatif pada kedua solusi tersebut. Tahap ini dilakukan dengan menghitung selisih tertimbang antara nilai kriteria dari suatu alternatif dengan nilai ideal positif serta negatif. Secara matematis, jarak relatif setiap alternatif pada solusi ideal dihitung sebagai:

$$d_{ij} = \frac{f_i^+ - f_{ij}}{f_i^+ - f_i^-} \quad (2.8)$$

Keterangan:

d_{ij} : jarak relatif alternatif ke- j terhadap kriteria ke- i
 f_i^+ : nilai ideal positif dari kriteria ke- i
 f_i^- : nilai ideal negatif dari kriteria ke- i
 f_{ij} : nilai kinerja alternatif ke- j terhadap kriteria ke- i

Langkah ini penting untuk menormalkan nilai kriteria yang mungkin memiliki satuan dan skala berbeda, sehingga hasil evaluasi antar kriteria dapat dibandingkan secara proporsional.

3. Menghitung Nilai Utilitas dan Nilai *Regret* Individu

Langkah ketiga adalah menghitung dua ukuran utama dalam metode VIKOR, yaitu nilai utilitas total (S_j) dan nilai *individual regret* (R_j). Nilai utilitas menggambarkan kepuasan kolektif terhadap semua kriteria, sedangkan nilai *regret*

menunjukkan ketidakpuasan maksimum terhadap satu kriteria yang paling tidak menguntungkan. Rumusnya adalah sebagai berikut:

$$S_j = \sum_{i=1}^n w_i \frac{f_i^+ - f_{ij}}{f_i^+ - f_i^-}, \quad R_j = \max_i \left[w_i \frac{f_i^+ - f_{ij}}{f_i^+ - f_i^-} \right] \quad (2.9)$$

Keterangan:

- S_j : Nilai utilitas total alternatif ke- j
- R_j : Nilai *individual regret* alternatif ke- j
- w_i : Bobot kriteria ke- i
- $f_i^+ f_i^-$: Nilai ideal positif dan negatif dari kriteria ke- i
- f_{ij} : nilai kinerja alternatif ke- j terhadap kriteria ke- i

Dalam konteks penelitian ini, (S_j) mencerminkan seberapa baik tingkat kesulitan tertentu memenuhi keseluruhan kriteria kemampuan pemain, sedangkan (R_j) menggambarkan sejauh mana tingkat kesulitan tersebut menghindari kinerja terburuk pada kriteria tertentu.

4. Menghitung Indeks Kompromi (Q)

Langkah keempat adalah menentukan indeks kompromi (Q_j) untuk setiap alternatif. Nilai (Q_j) mengindikasikan sejauh mana sebuah alternatif merupakan kompromi antara kepuasan kelompok (S_j) dan kepuasan individu (R_j). Rumus perhitungannya adalah:

$$Q_j = v \frac{S_j - S^+}{S^- - S^+} + (1 - v) \frac{R_j - R^+}{R^- - R^+} \quad (2.10)$$

Keterangan:

- Q_j : nilai indeks kompromi alternatif ke- j
- $S^- S^+$: nilai minimum dan maksimum dari S_j
- $R^- R^+$: skor minimum dan maksimum dari R_j
- v : parameter strategi kepuasan (biasanya $v = 0,5$) yang menunjukkan keseimbangan antara kepuasan kelompok dan individu.

Alternatif dengan nilai Q_j terkecil diartikan sebagai solusi kompromi terbaik, karena mempunyai jarak terpendek dari kondisi ideal berdasarkan keseimbangan dua perspektif tersebut.

5. Menentukan Peringkat dan Evaluasi Kompromi

Langkah terakhir yaitu menetapkan peringkat akhir alternatif menurut nilai Q_j yang sudah dikalkulasi. Alternatif dengan nilai Q_j terkecil menempati posisi tertinggi dalam peringkat keputusan. Selain itu, perlu dilakukan evaluasi kondisi penerimaan kompromi, yaitu memastikan bahwa alternatif terbaik memiliki perbedaan yang cukup signifikan dibandingkan alternatif berikutnya serta memenuhi kondisi konsistensi preferensi. Evaluasi ini menentukan bahwasanya hasil yang diperoleh bukan hanya secara matematis unggul, namun juga secara logis dapat diterima sebagai solusi kompromi yang adil.

2.6 *System Usability Scale (SUS)*

SUS merupakan instrumen evaluasi yang dipaparkan oleh John Brooke pada tahun 1986 sebagai alat untuk menilai tingkat *usability* atau kemudahan penggunaan suatu sistem, produk, atau antarmuka pengguna secara menyeluruh. SUS didesain guna memberikan pengukuran yang sederhana, cepat, namun tetap reliabel pada persepsi pengguna mengenai sistem yang digunakan. Metode ini bersifat *technology-independent*, artinya dapat diterapkan pada berbagai jenis sistem, aplikasi seluler, situs web, baik perangkat lunak, ataupun produk interaktif lainnya. Keunggulan SUS berada pada keandalannya dalam mengubah persepsi subjektif pengguna menjadi nilai numerik yang terstandardisasi, sehingga

memungkinkan analisis yang konsisten dan perbandingan antarproduk atau sistem dengan efisien (Lewis, 2018).

Metode SUS terdiri atas sepuluh butir pernyataan yang disusun berbentuk kuesioner memakai skala Likert lima poin, mulai dari “sangat tidak setuju” hingga “sangat setuju”. Pernyataan-pernyataan tersebut meliputi aspek kepuasan, kompleksitas, konsistensi, dan tingkat kepercayaan pengguna pada sistem yang diuji. Setiap butir pernyataan bergantian antara pernyataan positif dan negatif untuk mengurangi bias dalam penilaian. Menurut Bangor, Kortum, dan Miller (2008), SUS memiliki reliabilitas dan validitas tinggi, bahkan dengan jumlah responden yang relatif kecil, sehingga menjadi satu diantara metode terbanyak yang digunakan pada riset *usability* modern.

Langkah-langkah dalam pengukuran *System Usability Scale* dapat dijelaskan sebagai berikut.

1. Penyusunan Kuesioner SUS

Tahap awal melibatkan penyusunan kuesioner yang terdiri atas sepuluh pernyataan standar. Lima di antaranya bersifat positif, dan lima sisanya bersifat negatif. Responden diarahkan untuk menilai pada setiap pernyataan dengan memakai skala Likert 1 hingga 5, di mana nilai 1 mengindikasikan ketidaksetujuan yang tinggi dan nilai 5 mengindikasikan persetujuan yang tinggi pada pernyataan yang diajukan. Penyusunan kuesioner ini bertujuan guna mendapat persepsi langsung dari pengguna mengenai konsistensi, kemudahan penggunaan, dan keefektifan sistem.

2. Pengumpulan dan Penyesuaian Skor

Setelah kuesioner diisi oleh responden, langkah berikutnya adalah menyesuaikan nilai setiap item berdasarkan jenis pernyataannya. Untuk item positif (nomor ganjil), skor penyesuaian dikalkulasi dengan mengurangi nilai 1 dari skor asli. Sementara untuk item negatif (nomor genap), skor penyesuaian dihitung dengan mengurangi nilai tersebut dari 5. Rumus yang digunakan adalah:

$$S_i = \begin{cases} x_i - 1, & \text{untuk item positif} \\ 5 - x_i, & \text{untuk item negatif} \end{cases} \quad (2.11)$$

Keterangan:

S_i : skor penyesuaian untuk *item* ke- i
 x_i : skor asli dari responden pada *item* ke- i

Langkah ini memastikan bahwa nilai yang lebih tinggi selalu merepresentasikan persepsi *usability* yang lebih baik dari pengguna terhadap sistem yang dievaluasi.

3. Perhitungan Total Skor SUS

Setelah semua skor item disesuaikan, nilai total SUS dihitung dengan menjumlahkan seluruh skor item, kemudian mengalikannya dengan faktor konversi sebesar 2,5. Rumus perhitungan dituliskan sebagai berikut:

$$SUS = (\sum_{i=1}^{10} S_i) \times 2.5 \quad (2.12)$$

Keterangan:

SUS : skor total *System Usability Scale*
 S_i : skor penyesuaian *item* ke- i

Hasil akhir memiliki rentang nilai antara 0 hingga 100. Nilai ini tidak mewakili persentase, melainkan skor relatif yang menggambarkan tingkat *usability* sistem secara keseluruhan.

4. Interpretasi Hasil Pengukuran

Menurut penelitian oleh Bangor, Kortum, dan Miller (2008), interpretasi skor SUS dapat diklasifikasikan ke dalam beberapa kategori kualitas. Skor di bawah 50 menunjukkan tingkat *usability* yang buruk (*Poor*), skor antara 50 hingga 70 dianggap cukup (OK), skor antara 70 hingga 85 tergolong baik (*Good*), dan skor di atas 85 dikategorikan sangat baik (*Excellent*). Klasifikasi ini membantu peneliti untuk menentukan apakah sistem atau aplikasi yang dikembangkan sudah memenuhi ekspektasi pengguna dari aspek kemudahan penggunaan dan pengalaman interaktif yang memuaskan.

Dalam konteks penelitian ini, metode *System Usability Scale* dimanfaatkan dalam menguji sisi *usability* dari *game* ‘Avoid Najasa’, yang dikembangkan dengan sistem penyesuaian tingkat kesulitan berdasarkan kemampuan pemain menggunakan metode AHP–VIKOR. SUS menjadi alat yang penting untuk menilai sejauh mana hasil penerapan metode tersebut berpengaruh terhadap pengalaman pengguna dalam memainkan *game*. Melalui evaluasi ini, peneliti bisa memastikan bahwasanya penyesuaian tingkat kesulitan yang diperoleh tak hanya akurat secara algoritmik, namun juga memberikan kenyamanan, kejelasan kontrol, dan kepuasan bagi pemain. Dengan demikian, hasil pengujian SUS berperan dalam menilai keberhasilan sistem dalam menciptakan keseimbangan antara tantangan permainan dan kenyamanan pengguna.

BAB III

DESAIN DAN IMPLEMENTASI

3.1 Analisis dan Perancangan *Game*

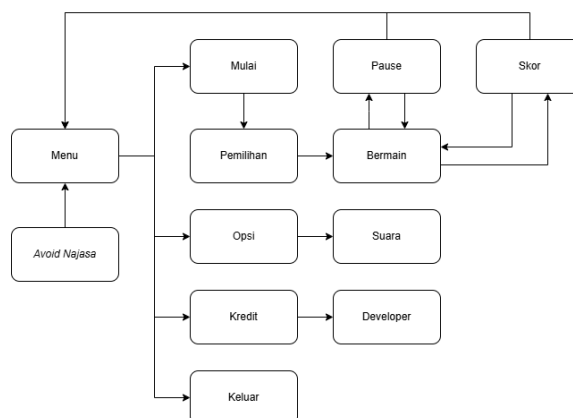
Analisis dan perancangan yang akan dilakukan untuk mengembangkan permainan ‘Avoid Najasa’ meliputi analisis *game*, desain *game*, serta desain antarmuka.

3.1.1 Analisis *Endless Runner Game*

Game ‘Avoid Najasa’ adalah *game* tiga dimensi dengan genre *endless runner* yang memiliki platform *desktop*. *Game* ini menantang pemain untuk berlari tanpa batas untuk mendapatkan koin di sepanjang platform tanpa menabrak rintangan.

3.1.2 Perancangan *Endless Runner Game*

Perancangan permainan ‘Avoid Najasa’ diawali dengan tampilan menu utama yang muncul pada halaman awal yang terdiri dari Mulai, Opsi, Kredit dan Keluar. Seperti ditampilkan pada Gambar 3.1 Berikut:

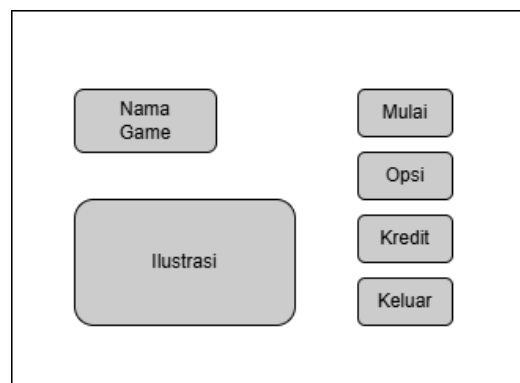


Gambar 3. 1 Diagram Menu

Pada Gambar 3.1 tampak diagram menu permainan dimana alurnya dimulai dari menu utama, lalu dapat masuk ke tiap menu lain yaitu kredit, opsi, dan main untuk masuk ke halaman pemilihan. Sebelum memulai permainan, pemain diminta untuk memilih karakter terlebih dahulu di bagian pemiliha, kemudian permainan berlangsung hingga selesai dan menampilkan skor atau ketika permainan dijeda.

3.1.3 Rancangan Antarmuka

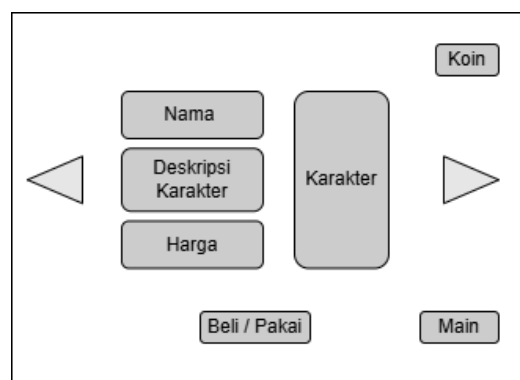
1. Halaman Menu



Gambar 3. 2 Rancangan Menu

Seperti tampak pada Gambar 3.2, halaman awal berupa menu dengan pilihan Mulai, Opsi, Kredit, dan Keluar.

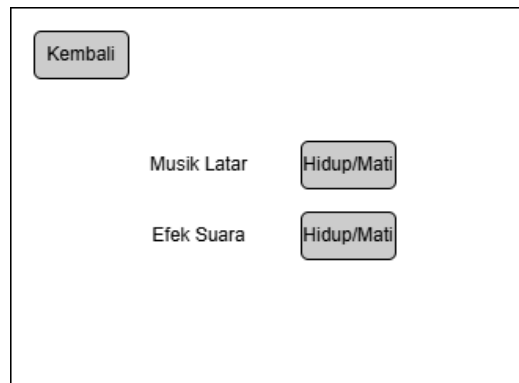
2. Tampilan Pemilihan



Gambar 3. 3 Rancangan Pemilihan

Gambar 3.3 merupakan rancangan halaman pemilihan untuk memilih karakter yang akan digunakan.

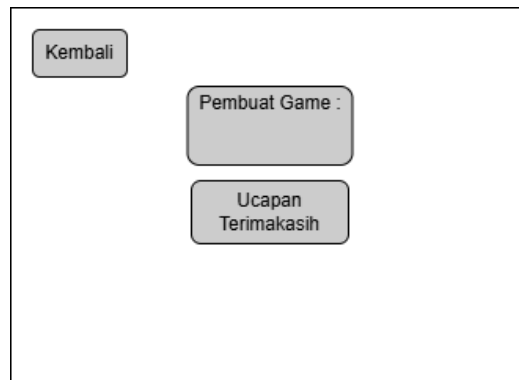
3. Tampilan Opsi



Gambar 3. 4 Rancangan Opsi

Gambar 3.4 menunjukkan desain halaman Opsi, untuk menonaktifkan atau mengaktifkan musik maupun efek suara.

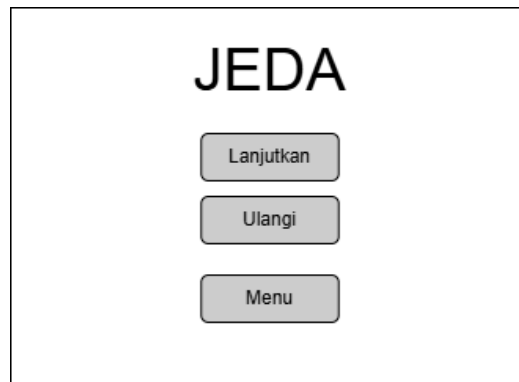
4. Tampilan Kredit



Gambar 3. 5 Rancangan Kredit

Gambar 3.5 merupakan rancangan kredit, yang berisi informasi mengenai pengembang *game* dan juga ucapan terimakasih.

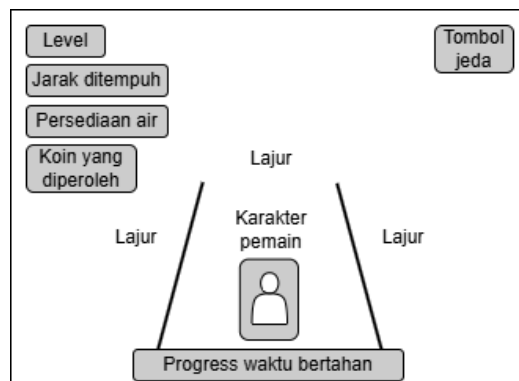
5. Tampilan Jeda



Gambar 3. 6 Rancangan Jeda

Gambar 3.6, tampilan awal berupa menu dengan pilihan Mulai, Jeda, Kredit, dan Keluar.

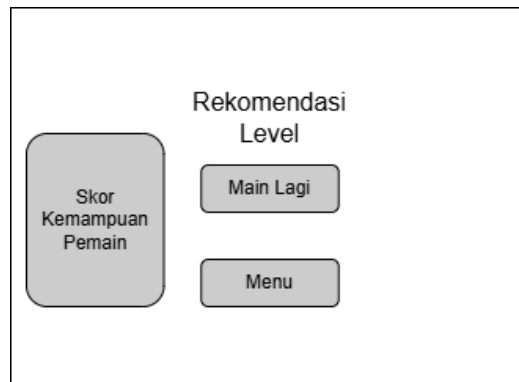
6. Tampilan *gameplay*



Gambar 3. 7 Rancangan Tampilan *Gameplay*

Gambar 3.7 menunjukkan rancangan tampilan *gameplay*, dimana terdapat karakter pemain yang akan berlari dan dapat berpindah ke kiri dan ke kanan antar lajur maupun melompat untuk menghindari dari najis maupun mengambil koin/air.

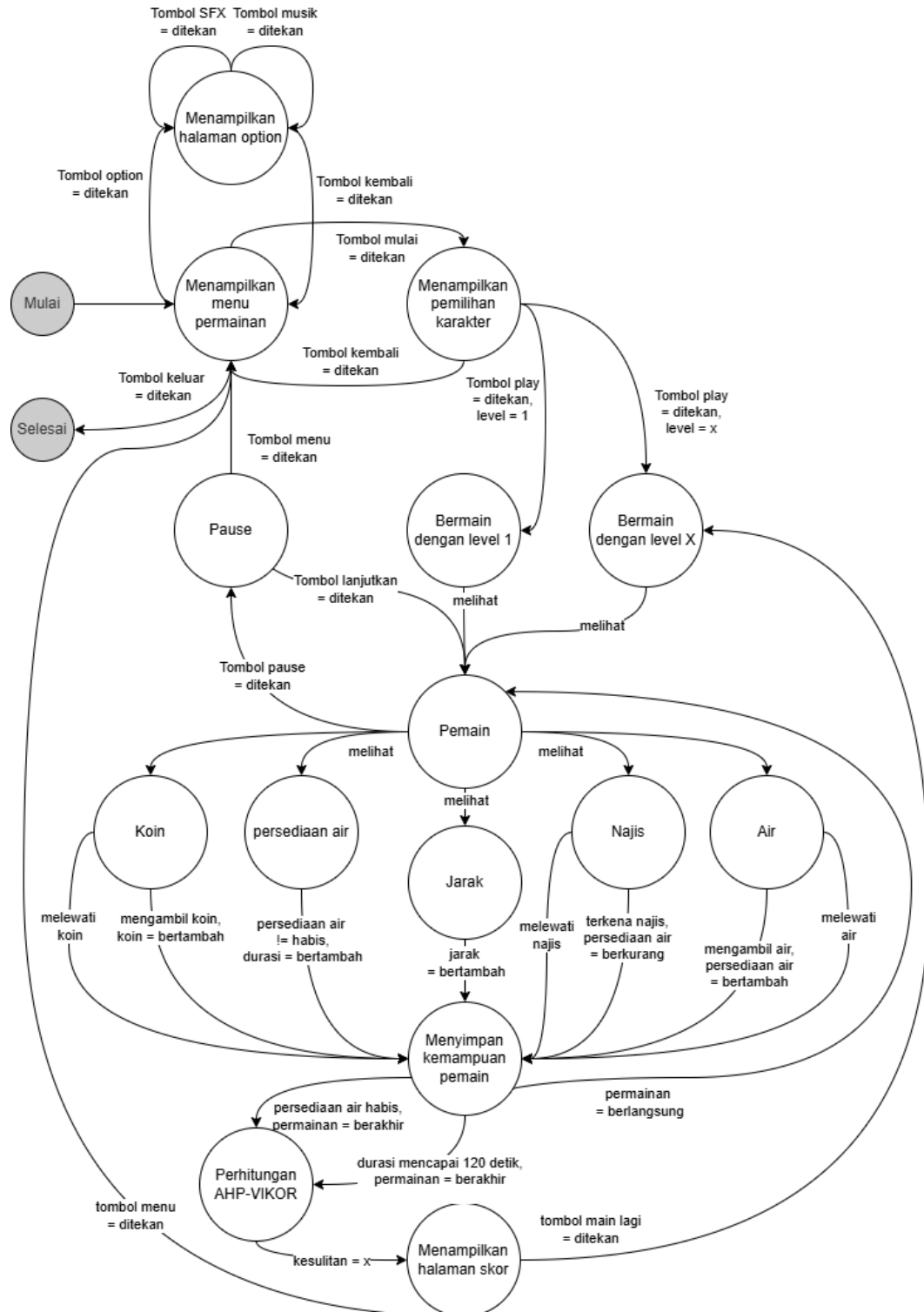
7. Tampilan Skor



Gambar 3. 8 Rancangan Skor

Gambar 3.8 menunjukkan rancangan tampilan skor yang akan ditampilkan ketika permainan berakhir, pemain bisa memilih untuk kembali bermain maupun keluar menuju menu.

3.2 Desain Sistem



Gambar 3. 9 Diagram *Finite State Machine*

Pada Gambar 3.9 menunjukkan *Diagram Finite State Machine* dimulai dari keadaan menu utama, di mana pemain dapat memilih opsi, mulai, atau keluar. Saat pemain menekan mulai, sistem menampilkan pemilihan karakter, kemudian berlanjut ke keadaan bermain dengan level 1 atau level X sesuai tingkat kesulitan yang telah ditentukan.

Dalam keadaan bermain, pemain berinteraksi dengan beberapa elemen seperti koin, air, najis, durasi bertahan dan jarak tempuh. Pemain dapat mengambil koin untuk menambah skor, mengambil air untuk menambah durasi bermain, dan menghindari najis karena kontak dengannya akan mengurangi persediaan air. Ketika persediaan air habis atau durasi permainan mencapai 120 detik, permainan berakhir dan data kemampuan pemain disimpan untuk proses perhitungan AHP-VIKOR, yang menentukan tingkat kesulitan permainan selanjutnya secara adaptif. Setelah hasil diperoleh, sistem menampilkan halaman skor, dan pemain bisa memilih untuk bermain kembali atau keluar ke menu utama.

3.3 Data Alternatif

Menurut Saaty (2008), alternatif merupakan objek evaluasi dalam proses pengambilan keputusan multi-kriteria yang mewakili berbagai pilihan solusi untuk mencapai tujuan tertentu. Dalam konteks metode AHP–VIKOR, alternatif berperan sebagai opsi yang akan dibandingkan dan diperingkat menurut kriteria yang sudah ditetapkan guna memperoleh keputusan yang paling optimal. Pada penelitian ini, alternatif yang digunakan berupa sembilan tingkat kesulitan permainan pada *game* ‘Avoid Najasa’, yaitu Level 1 hingga Level 15. Setiap level mewakili variasi kompleksitas permainan yang berbeda, baik dari sisi kecepatan karakter, jumlah

dan frekuensi kemunculan najis. Seluruh alternatif ini menjadi dasar evaluasi dalam menentukan tingkat kesukaran permainan yang paling cocok dengan keterampilan pemain secara terukur.

Permainan ‘Avoid Najasa’ dirancang dengan sistem adaptif di mana sesi pertama permainan selalu dimulai dari Level 1. Setelah pemain menyelesaikan atau gagal dalam satu sesi permainan, sistem akan mengumpulkan data performa seperti jumlah koin yang diperoleh, jarak tempuh, durasi bertahan, jumlah terkena najis, jumlah air yang berhasil diperoleh, jumlah koin yang terlewat, jumlah najis yang berhasil dihindari dan jumlah air yang terlewat. Data tersebut direkam sebagai indikator kemampuan pemain yang selanjutnya diolah menggunakan metode VIKOR yang sebelumnya sudah memperoleh bobot kepentingan setiap kriteria yang dihasilkan dengan metode AHP. Kombinasi tersebut untuk menentukan alternatif terbaik berdasarkan nilai kompromi antara kepuasan kelompok (nilai S_i) dan individu (nilai R_i) sebagaimana dijelaskan oleh Opricovic dan Tzeng (2004). Hasil perhitungan tersebut digunakan untuk menentukan level kesulitan pada permainan berikutnya, apakah meningkat atau menurun, sesuai dengan kemampuan pemain. Dengan begitu, sistem ini tak hanya bersifat adaptif tetapi juga objektif karena pengambilan keputusannya dilakukan secara kuantitatif. Struktur data alternatif ini menjadi landasan penting bagi tahapan selanjutnya, yaitu pembobotan kriteria menggunakan AHP dan penentuan alternatif terbaik menggunakan VIKOR, guna menciptakan pengalaman bermain yang seimbang antara tantangan dan kemampuan pemain.

Tabel 3. 1 Alternatif Tingkat Kesulitan

Tingkat Kesulitan	Kecepatan Permainan (UU/s)	Kemunculan Najis (Detik)	Jumlah maksimal Najis (Lajur)
<i>Level 1</i>	12-15	4	2
<i>Level 2</i>	12-15	4	3
<i>Level 3</i>	16-19	3,5	2
<i>Level 4</i>	16-19	3,5	3
<i>Level 5</i>	20-23	3	2
<i>Level 6</i>	20-23	3	3
<i>Level 7</i>	24-27	2,5	2
<i>Level 8</i>	24-27	2,5	3
<i>Level 9</i>	28-31	2	2
<i>Level 10</i>	28-31	2	3
<i>Level 11</i>	32-35	1,5	2
<i>Level 12</i>	32-35	1,5	3
<i>Level 13</i>	36-40	1	2
<i>Level 14</i>	36-40	1	3
<i>Level 15</i>	36-40	0,75	3

Pada Tabel 3.1 menunjukkan parameter setiap level, dimana parameter-parameter level diperoleh dengan mengembangkan parameter level pada riset yang telah dilakukan oleh Ardianto pada tahun 2024. Parameter kecepatan permainan dinyatakan dalam satuan *Unity Unit* per detik (UU/s), dimana *Unity Unit* merupakan satuan jarak dalam lingkungan *Unity*. Parameter kemunculan najis dinyatakan dalam satuan detik (s) yang menunjukkan interval waktu munculnya objek najis. Adapun jumlah maksimal najis dinyatakan dalam satuan lajur, yaitu banyaknya lajur yang dapat ditempati objek najis secara bersamaan pada suatu waktu. Pada permainan ini tersedia tiga lajur yang digunakan sebagai jalur pergerakan karakter dan objek najis.

3.4 Data Kriteria

Dalam metode pengambilan keputusan multikriteria, kriteria merupakan aspek atau parameter yang dijadikan dasar dalam mengevaluasi alternatif untuk mendapat keputusan yang paling sesuai dengan tujuan penelitian (Saaty, 2008). Dalam konteks penelitian ini, kriteria berfungsi sebagai ukuran performa pemain dalam permainan Avoid Najasa yang akan menentukan tingkat kesulitan berikutnya secara adaptif. Melalui kombinasi metode AHP–VIKOR, setiap kriteria akan mendapat bobot sesuai tingkat kepentingannya, yang berikutnya dipakai dalam proses perhitungan guna menetapkan tingkat kesukaran yang paling sesuai berdasarkan kemampuan pemain. Adapun kriteria yang dipakai pada riset ini ditampilkan pada Tabel 3.2.

Tabel 3. 2 Data Kriteria Penentu Tingkat Kesulitan

Simbol	Kriteria	Tipe	Deskripsi
<i>C1</i>	Jarak/Skor(m)	<i>Benefit</i>	Total jarak atau skor yang berhasil ditempuh oleh pemain selama permainan berlangsung
<i>C2</i>	Durasi bertahan(s)	<i>Benefit</i>	Mengukur lamanya waktu pemain mampu bertahan sampai persediaan air habis
<i>C3</i>	Koin	<i>Benefit</i>	Mengindikasikan jumlah koin yang berhasil dikumpulkan pemain
<i>C4</i>	Terkena Najis/ <i>Obstacle</i>	<i>Cost</i>	Frekuensi pemain menabrak atau terkena Najis.
<i>C5</i>	Air	<i>Benefit</i>	Jumlah pemain mengambil air untuk menambah persediaan air
<i>C6</i>	Melewati Koin	<i>Cost</i>	Jumlah koin yang terlewat atau tidak dikumpulkan selama permainan berlangsung
<i>C7</i>	Melewati Najis	<i>Benefit</i>	Angka keberhasilan pemain menghindari najis
<i>C8</i>	Melewati Air	<i>Cost</i>	Jumlah pemain gagal mengambil air yang seharusnya diambil untuk menambah persediaan air

Untuk mengonversi data mentah hasil permainan ke dalam bentuk skala yang dapat dibandingkan, setiap kriteria diberikan rentang nilai 1-5. Nilai 5 menunjukkan kondisi terbaik untuk kriteria bertipe *benefit*, sedangkan 1 menunjukkan kondisi terbaik untuk kriteria bertipe *cost*. Skala penilaian ini disusun berdasarkan asumsi performa pemain dalam permainan ‘Avoid Najasa’.

Tabel 3. 3 Skala Penilaian Kriteria

Simbol	Kriteria	Sub Kriteria	Skala Nilai
C1	Jarak/Skor	< 721 m	1
		721 – 820 m	2
		821 – 920 m	3
		921 – 1020 m	4
		> 1020 m	5
C2	Durasi bertahan	< 30 s	1
		30 – 59 s	2
		60 – 89 s	3
		90 – 119 s	4
		120 s	5
C3	Koin yang didapat	≤ 20%	1
		21 – 40%	2
		41 – 60%	3
		61 – 80%	4
		> 80%	5
C4	Jumlah terkena Najis	> 12	1
		10 – 12	2
		7 – 9	3
		4 – 6	4
		< 4	5
C5	Jumlah Air yang didapat	< 4	1
		4 – 6	2
		7 – 9	3
		10 – 12	4
		> 12	5
C6	Jumlah koin yang terlewat	> 80%	1
		61 – 80%	2
		41 – 60%	3
		21 – 40%	4
		≤ 20%	5
C7		< 6	1

Simbol	Kriteria	Sub Kriteria	Skala Nilai
	Jumlah najis berhasil dihindari	6 – 10	2
		11 – 15	3
		16 – 20	4
		> 20	5
C8	Jumlah air yang terlewat	> 12	1
		10 – 12	2
		7 – 9	3
		4 – 6	4
		< 4	5

Penjelasan:

1. Jarak menunjukkan seberapa jauh pemain mampu bertahan tanpa menabrak rintangan. Pemain dengan jarak tempuh lebih jauh menandakan kemampuan refleks dan fokus yang lebih baik.
2. Durasi menunjukkan daya tahan pemain terhadap rintangan dan kemampuan mempertahankan performa dalam waktu lama.
3. Semakin banyak koin yang didapat, menandakan performa pemain lebih baik.
4. Semakin sedikit pemain menabrak najis, semakin baik kemampuan menghindari rintangan. Karena termasuk kriteria *cost*, nilai rendah menunjukkan performa buruk
5. Air berperan penting untuk memperpanjang waktu bermain. Semakin banyak air yang dikumpulkan, semakin baik daya adaptasi pemain.
6. Koin terlewat mengindikasikan kurangnya presisi gerak pemain. Karena bertipe *cost*, semakin sedikit jumlahnya semakin baik.
7. Semakin banyak najis yang berhasil dihindari, semakin baik kemampuan pemain dalam membaca pola permainan dan menjaga kelincahan.

8. Air terlewat mencerminkan tingkat kecepatan dan ketepatan pemain saat mengambil item. Karena bertipe *cost*, semakin sedikit jumlahnya semakin baik.

Kombinasi antara kriteria bertipe *benefit* dan *cost* memungkinkan sistem menilai performa pemain secara lebih seimbang dan menyeluruh (Opricovic & Tzeng, 2004). Kriteria-kriteria ini menjadi fondasi bagi proses pembobotan menggunakan metode AHP pada subbab berikutnya dan seleksi alternatif terbaik menggunakan metode VIKOR, sehingga hasil penentuan tingkat kesulitan dalam permainan Avoid Najasa bersifat objektif dan adaptif terhadap kemampuan pemain.

3.5 Pembobotan Kriteria menggunakan AHP

Metode *Analytic Hierarchy Process* (AHP) dikembangkan oleh Thomas L. Saaty (2008) dan dipakai dalam menetapkan tingkat kepentingan relatif antar kriteria dalam pengambilan keputusan multikriteria. Prinsip dasar AHP yaitu melakukan perbandingan berpasangan (*pairwise comparison*) terhadap setiap kriteria berdasarkan tingkat kepentingannya dalam mencapai tujuan, kemudian menghasilkan bobot prioritas yang merepresentasikan tingkat pengaruh masing-masing kriteria terhadap hasil akhir.

Dalam penelitian ini, pembobotan kriteria dilakukan untuk menentukan kontribusi relatif setiap aspek terhadap penentuan tingkat kesulitan permainan 'Avoid Najasa'. Menurut Saaty pada tahun 2008, perbandingan berpasangan dilakukan dengan skala 1-9, sebagaimana ditampilkan pada Tabel 3.4.

Tabel 3. 4 Skala Penilaian Perbandingan Berpasangan

Nilai	Keterangan
1	Kedua elemen sama penting
3	Elemen yang satu sedikit lebih penting
5	Elemen yang satu jelas lebih penting

Nilai	Keterangan
7	Elemen yang satu sangat lebih penting
9	Elemen yang satu mutlak lebih penting
2,4,6,8	Nilai kompromi di antara dua tingkat kepentingan yang berdekatan

3.5.1 Penyusunan Matriks Perbandingan Berpasangan oleh Para Pengembang *Game*

Tahapan awal dalam metode *Analytical Hierarchy Process* (AHP) yaitu penyusunan matriks perbandingan berpasangan antar kriteria. Pada penelitian ini, penilaian dilakukan oleh tiga orang pengembang *game*, yang diminta untuk membandingkan tingkat kepentingan setiap kriteria secara berpasangan memakai skala Saaty.

Setiap pengembang *game* menghasilkan satu matriks perbandingan berpasangan antar kriteria. Matriks perbandingan berpasangan dari masing-masing pengembang *game* disajikan pada Tabel 3.5-3.7 berikut.

Tabel 3. 5 Matriks Perbandingan Berpasangan Pengembang *Game* 1

Kriteria	C1	C2	C3	C4	C5	C6	C7	C8
C1	1	5	1/3	3	5	7	9	3
C2	1/5	1	2	2	3	4	2	2
C3	3	1/2	1	2	3	3	5	3
C4	1/3	1/2	1/2	1	2	2	1/3	1/3
C5	1/5	1/3	1/3	1/2	1	4	2	2
C6	1/7	1/4	1/3	1/2	1/4	1	3	2
C7	1/9	1/2	1/5	3	1/2	1/3	1	2
C8	1/3	1/2	1/3	3	1/2	1/2	1/2	1

Tabel 3. 6 Matriks Perbandingan Berpasangan Pengembang *Game* 2

Kriteria	C1	C2	C3	C4	C5	C6	C7	C8
C1	1	1	1/5	1/7	5	5	1/5	7
C2	1	1	3	1/5	3	3	3	5
C3	5	1/3	1	1/3	5	5	5	7
C4	7	5	3	1	7	5	9	7

Kriteria	C1	C2	C3	C4	C5	C6	C7	C8
C5	1/5	1/3	1/5	1/7	1	3	5	5
C6	1/5	1/3	1/5	1/5	1/3	1	1	1
C7	5	1/3	1/5	1/9	1/5	1	1	1
C8	1/7	1/5	1/7	1/7	1/5	1	1	1

Tabel 3. 7 Matriks Perbandingan Berpasangan Pengembang *Game 3*

Kriteria	C1	C2	C3	C4	C5	C6	C7	C8
C1	1	9	9	9	9	9	9	9
C2	1/9	1	1/9	9	9	9	9	9
C3	1/9	9	1	9	9	9	9	9
C4	1/9	1/9	1/9	1	4	4	4	4
C5	1/9	1/9	1/9	1/4	1	4	4	4
C6	1/9	1/9	1/9	1/4	1/4	1	1/9	1/9
C7	1/9	1/9	1/9	1/4	1/4	9	1	9
C8	1/9	1/9	1/9	1/4	1/4	9	1/9	1

3.5.2 Agregasi Penilaian Pengembang *Game* Menggunakan *Geometric Mean* (AIJ)

Karena riset ini melibatkan lebih dari satu pengembang *game*, maka diperlukan proses penggabungan penilaian untuk memperoleh satu matriks perbandingan berpasangan kelompok. Penggabungan penilaian dilakukan menggunakan pendekatan *Aggregation of Individual Judgments* (AIJ) dengan metode *geometric mean*, sebagaimana direkomendasikan dalam metode AHP (Forman & Peniwati, 1998). Sebagai contoh, nilai perbandingan antara kriteria C1 dan C2 diperoleh dari hasil *geometric mean* penilaian tiga pengembang *game*.

3.5.3 Matriks Perbandingan Berpasangan Kelompok

Hasil agregasi penilaian dari ketiga pengembang *game* menghasilkan satu matriks perbandingan berpasangan kelompok, yang dipakai sebagai dasar dalam

perhitungan bobot kriteria. Matriks perbandingan berpasangan kelompok ditunjukkan pada Tabel 3.8.

Tabel 3. 8 Matriks Perbandingan Berpasangan antar Kriteria

Kriteria	C1	C2	C3	C4	C5	C6	C7	C8
C1	1	3,557	0,843	1,568	6,082	6,804	2,53	5,739
C2	0,281	1	0,874	1,533	4,327	4,762	3,78	4,481
C3	1,186	1,145	1	1,817	5,13	5,13	6,082	5,739
C4	0,638	0,652	0,55	1	3,826	3,42	2,289	2,105
C5	0,164	0,231	0,195	0,261	1	3,634	3,42	3,42
C6	0,147	0,21	0,195	0,292	0,275	1	0,693	0,606
C7	0,395	0,265	0,164	0,437	0,292	1,442	1	2,621
C8	0,174	0,223	0,174	0,475	0,292	1,651	0,382	1
Total	3,985	7,283	3,996	7,384	21,225	27,844	20,177	25,711

3.5.4 Menghitung Nilai Normalisasi Matriks Keputusan

Langkah selanjutnya yakni menormalisasi matriks perbandingan berpasangan kelompok dengan cara membagi setiap elemen matriks dengan total kolom masing-masing. Proses ini bertujuan untuk mendapat matriks ternormalisasi yang digunakan dalam perhitungan bobot kriteria. Matriks ternormalisasi bisa dilihat dalam tabel 3.9 berikut.

Tabel 3. 9 Matriks Normalisasi

[illegible]

3.5.5 Nilai *Eigen* / Bobot Kriteria

Nilai eigen atau bobot dari setiap kriteria diperoleh dengan mengkalkulasi rata-rata dari nilai kriteria pada setiap kolom. Perhitungan dilakukan dengan menjumlahkan semua nilai pada satu kolom kriteria, kemudian membaginya dengan jumlah keseluruhan kriteria yang digunakan dalam riset ini, yaitu sebanyak delapan kriteria. Nilai eigen tersebut dihasilkan berdasarkan matriks yang sudah dinormalisasi. Untuk hasil kalkulasi nilai eigen pada setiap kriteria tampak pada Tabel 3.10 berikut.

Tabel 3. 10 Bobot kriteria

Kriteria	C1	C2	C3	C4	C5	C6	C7	C8	Bobot
C1	0,251	0,488	0,211	0,212	0,287	0,244	0,125	0,223	0,255
C2	0,071	0,137	0,219	0,208	0,204	0,171	0,187	0,174	0,171
C3	0,298	0,157	0,250	0,246	0,242	0,184	0,301	0,223	0,238
C4	0,160	0,090	0,138	0,135	0,180	0,123	0,113	0,082	0,128
C5	0,041	0,032	0,049	0,035	0,047	0,131	0,170	0,133	0,080
C6	0,037	0,029	0,049	0,040	0,013	0,036	0,034	0,024	0,033
C7	0,099	0,036	0,041	0,059	0,014	0,052	0,050	0,102	0,057
C8	0,044	0,031	0,044	0,064	0,014	0,059	0,019	0,039	0,039

3.5.6 Nilai Konsistensi

Pada tahap perhitungan Nilai Konsistensi (*Consistency Ratio/CR*), diperlukan tahapan-tahapan perhitungan guna memperoleh nilai konsistensi akhir. Nilai-nilai yang dibutuhkan dalam proses ini meliputi nilai uji konsistensi, nilai lambda maksimum, indeks konsistensi (*Consistency Index/CI*), serta nilai indeks acak (*Random Index/RI*). Setelah seluruh nilai tersebut diperoleh, barulah dapat dihitung nilai konsistensinya (*CR*).

Untuk memperoleh nilai eigen maksimum ($\lambda_{\max}$), langkah yang dilakukan adalah mengalikan matriks perbandingan berpasangan dengan vektor bobot kriteria,

kemudian hasil perkalian setiap elemen dibagi dengan bobot kriteria yang bersangkutan. Nilai rata-rata dari hasil pembagian tersebut digunakan sebagai nilai $\lambda_{\max}$.

Baris ke 1

$$\frac{Aw_1}{w_1} = \frac{1 \times 0,255 + 3,557 \times 0,171 + 0,843 \times 0,238 + 1,568 \times 0,128 + 6,082 \times 0,08 + 6,804 \times 0,033 + 2,53 \times 0,57 + 5,739 \times 0,039}{0,255} \quad (3.1)$$

$$= 9,165$$

Dengan melakukan perhitungan pada setiap baris, jika perhitungannya benar akan menghasilkan nilai seperti pada tabel 3.11 berikut.

Tabel 3. 11 Perhitungan $\lambda_{\max}$

Kriteria	Bobot	Hasil kali
C1	0,307	9,165
C2	0,206	8,964
C3	0,145	8,891
C4	0,053	9,098
C5	0,119	8,623
C6	0,035	8,422
C7	0,1	8,313
C8	0,034	8,242
$\lambda_{\max}$		8,715

Didapatkan nilai $\lambda_{\max} = 8,715$, sesudah memperoleh skor $\lambda_{\max}$ berikutnya menghitung Indeks Konsistensi (CI).

$$CI = \frac{\lambda_{\max} - n}{n - 1} = \frac{8,715 - 8}{8 - 1} = 0,102 \quad (3.2)$$

Setelah didapatkan indeks konsistensi, selanjutnya menghitung Rasio Konsistensi (CR). Dalam penelitian ini jumlah kriterianya adalah 8, maka Random Indeksnya adalah 1,141.

$$CR = \frac{CI}{RI} = \frac{0,102}{1,141} = 0,072 \quad (3.3)$$

Dari hasil perhitungan CR diatas, diperoleh CR dibawah 0,1 maka matriks perbandingan berpasangan termasuk konsisten serta penilaian bisa diterima.

3.6 Penentuan Alternatif terbaik menggunakan VIKOR

Metode VIKOR (*VlseKriterijumska Optimizacija I Kompromisno Resenje*) dipakai dalam menentukan tingkat kesulitan permainan yang paling sesuai berdasarkan hasil pembobotan kriteria yang telah diperoleh melalui metode AHP. Metode ini berfokus pada pemilihan alternatif terbaik dengan pendekatan kompromi, yakni mencari solusi yang paling mendekati kondisi ideal berdasarkan keseimbangan antara maksimalisasi keuntungan (*benefit*) dan minimalisasi kerugian (*cost*).

Dalam konteks penelitian ini, setiap alternatif merepresentasikan level kesulitan permainan (Level 1–15), sedangkan kriteria mencerminkan berbagai aspek performa pemain selama bermain, seperti jarak tempuh, durasi bertahan, jumlah koin yang diperoleh, jumlah najis yang ditabrak, air yang dikumpulkan, koin yang terlewat, najis yang dihindari, serta air yang terlewat.

Proses perhitungan dengan metode VIKOR melibatkan beberapa langkah utama, yaitu:

1. Menyusun matriks keputusan berdasarkan data performa setiap alternatif terhadap kriteria.
2. Melakukan normalisasi terhadap nilai matriks keputusan.
3. Menghitung nilai utilitas (S) dan nilai penyesalan (R).

4. Menghitung indeks VIKOR (Q).
5. Menentukan alternatif terbaik, yaitu level kesulitan dengan nilai Q terendah sebagai tingkat yang paling sesuai dengan kemampuan pemain.

3.6.1 Matriks Keputusan

Tahap awal dalam metode VIKOR yaitu membentuk matriks keputusan (*decision matrix*), yang berisi nilai kinerja dari masing-masing alternatif (*level kesulitan*) terhadap seluruh kriteria. Nilai pada matriks ini menggunakan skala penilaian 1–5 seperti yang sudah ditetapkan pada bagian sebelumnya, di mana skala 5 menunjukkan performa terbaik untuk kriteria *benefit*, dan skala 1 menunjukkan performa terbaik untuk kriteria *cost*.

Tabel 3.12 menunjukkan contoh representasi matriks keputusan yang digunakan dalam penelitian.

Tabel 3. 12 Matriks Keputusan Alternatif Tingkat Kesulitan

Alt	C1	C2	C3	C4	C5	C6	C7	C8
L1	1	1	1	1	1	1	1	1
L2	1	1	1	2	1	2	1	2
L3	2	2	1	2	2	2	2	2
L4	2	2	2	3	2	3	2	3
L5	2	3	2	3	2	3	3	3
L6	3	3	2	3	3	3	3	3
L7	3	3	3	4	3	4	3	4
L8	3	4	3	4	3	4	4	4
L9	4	4	3	4	4	4	4	4
L10	4	4	4	4	4	4	4	4
L11	4	4	4	5	4	5	4	5
L12	4	5	4	5	4	5	5	5
L13	5	5	4	5	5	5	5	5
L14	5	5	5	5	5	5	4	5
L15	5	5	5	5	5	5	5	5
LX	5	4	5	2	3	5	5	5

Baris terbawah menampilkan nilai evaluasi pemain yang telah diubah ke dalam lima kriteria dengan skala yang sama seperti alternatif yang lain. Nilai pemain tersebut tak diperhitungkan sebagai alternatif solusi, namun berfungsi sebagai alternatif X yang menjadi acuan pembandingan pada alternatif yang tersedia guna menentukan tingkat kesulitan yang paling mendekati menurut nilai Q.

3.6.2 Melakukan Normalisasi Matriks Keputusan

Dengan proses normalisasi, seluruh nilai kinerja alternatif akan dikonversi menjadi rentang 0 sampai 1, di mana skor dengan performa terbaik adalah yang paling mendekati 0, sementara nilai semakin mendekati 1 menunjukkan performa terendah. Tabel 3.13 menggambarkan hasil normalisasi dari matriks keputusan sebelumnya:

Tabel 3. 13 Normalisasi Matrik Keputusan Alternatif Tingkat Kesulitan

Label	C1	C2	C3	C4	C5	C6	C7	C8
L1	1	1	1	1	1	1	1	1
L2	1	1	1	0,75	1	0,75	1	0,75
L3	0,75	0,75	1	0,75	0,75	0,75	0,75	0,75
L4	0,75	0,75	0,75	0,5	0,75	0,5	0,75	0,5
L5	0,75	0,5	0,75	0,5	0,75	0,5	0,5	0,5
L6	0,5	0,5	0,75	0,5	0,5	0,5	0,5	0,5
L7	0,5	0,5	0,5	0,25	0,5	0,25	0,5	0,25
L8	0,5	0,25	0,5	0,25	0,5	0,25	0,25	0,25
L9	0,25	0,25	0,5	0,25	0,25	0,25	0,25	0,25
L10	0,25	0,25	0,25	0,25	0,25	0,25	0,25	0,25
L11	0,25	0,25	0,25	0	0,25	0	0,25	0
L12	0,25	0	0,25	0	0,25	0	0	0
L13	0	0	0,25	0	0	0	0	0
L14	0	0	0	0	0	0	0,25	0
L15	0	0	0	0	0	0	0	0
LX	0	0,25	0	0,75	0,5	0	0	0

3.6.3 Menghitung Nilai S dan R

Kolom R Max menunjukkan skor tertinggi dari masing-masing alternatif. Nilai ini merepresentasikan kriteria dengan performa paling rendah pada alternatif tersebut. Selanjutnya, seluruh skor R *Max* yang sudah dihimpun diurut dari yang terendah hingga yang terbesar guna menentukan nilai R minimum dan R maksimum. Nilai R ditampilkan pada Tabel 3.14.

Tabel 3. 14 Hasil nilai R

Alt	C1	C2	C3	C4	C5	C6	C7	C8	R
L1	0,26	0,17	0,24	0,13	0,08	0,03	0,06	0,04	0,26
L2	0,26	0,17	0,24	0,10	0,08	0,02	0,06	0,03	0,26
L3	0,19	0,13	0,24	0,10	0,06	0,02	0,04	0,03	0,24
L4	0,19	0,13	0,18	0,06	0,06	0,02	0,04	0,02	0,19
L5	0,19	0,09	0,18	0,06	0,06	0,02	0,03	0,02	0,19
L6	0,13	0,09	0,18	0,06	0,04	0,02	0,03	0,02	0,18
L7	0,13	0,09	0,12	0,03	0,04	0,01	0,03	0,01	0,13
L8	0,13	0,04	0,12	0,03	0,04	0,01	0,01	0,01	0,13
L9	0,06	0,04	0,12	0,03	0,02	0,01	0,01	0,01	0,12
L10	0,06	0,04	0,06	0,03	0,02	0,01	0,01	0,01	0,06
L11	0,06	0,04	0,06	0,00	0,02	0,00	0,01	0,00	0,06
L12	0,06	0,00	0,06	0,00	0,02	0,00	0,00	0,00	0,06
L13	0,00	0,00	0,06	0,00	0,00	0,00	0,00	0,00	0,06
L14	0,00	0,00	0,00	0,00	0,00	0,00	0,01	0,00	0,01
L15	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0
LX	0,00	0,04	0,00	0,10	0,04	0,00	0,00	0,00	0,10
Bobot	0,26	0,17	0,24	0,13	0,08	0,03	0,06	0,04	

Pada Tabel 3.14, skor R didapat dari elemen tertinggi pada matriks normalisasi yang sudah dikali dengan bobot kriteria, sedangkan nilai S dihitung dengan menjumlah seluruh nilai yang ada pada setiap baris matriks tersebut. Berikut contoh beberapa kalkulasi Nilai S.

$$S_1 = 0,26 + 0,17 + 0,24 + 0,13 + 0,08 + 0,03 + 0,06 + 0,04 = 1$$

$$S_2 = 0,26 + 0,17 + 0,24 + 0,1 + 0,08 + 0,02 + 0,06 + 0,03 = 0,95$$

$$S_{15} = 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 = 0$$

$$S_x = 0 + 0,04 + 0 + 0,10 + 0,04 + 0 + 0 + 0 = 0,18$$

Nilai S merepresentasikan total jarak setiap alternatif terhadap keadaan ideal dengan keseluruhan. Hasil perhitungan nilai S bisa dilihat dengan rinci dalam Tabel 3.15.

Tabel 3. 15 Hasil nilai S

Alt	S
L1	1
L2	0,95
L3	0,81
L4	0,70
L5	0,64
L6	0,56
L7	0,45
L8	0,39
L9	0,31
L10	0,25
L11	0,20
L12	0,14
L13	0,06
L14	0,01
L15	0
LX	0,18

Sesudah dilakukan perhitungan maka bisa ditetapkan skor *Max*, *Min* dari R serta S tampak dalam Tabel 3.16.

Tabel 3. 16 Nilai Max, Min dari Nilai S dan R

Level	Nilai S	Nilai R
Min	0	0
Max	1	0,26

3.6.4 Menghitung Nilai Indeks Kompromi

Setelah diperoleh nilai langkah berikutnya adalah menentukan nilai indeks kompromi yang akan menjadi acuan pada pengambilan keputusan tahap

selanjutnya. Perhitungan indeks kompromi (Q) dilakukan menggunakan persamaan berikut.

$$Q_j = v \frac{S_j - S^+}{S^- - S^+} + (1 - v) \frac{R_j - R^+}{R^- - R^+}$$

Hasil nilai indeks kompromi untuk masing-masing alternatif ditampilkan secara rinci, dengan penetapan parameter V sebesar 0,5 agar memberikan keseimbangan.

Tabel 3. 17 Nilai Indeks Kompromi (Q)

Alternatif	v	Q
L1	0,5	1,000
L2		0,975
L3		0,870
L4		0,725
L5		0,697
L6		0,629
L7		0,475
L8		0,447
L9		0,387
L10		0,250
L11		0,225
L12		0,197
L13		0,146
L14		0,035
L15		0,000
LX		0,277

Tabel 3.17 menunjukkan hasil akhir perhitungan vikor yaitu berupa nilai Q , dimana semakin kecil nilai Q berarti semakin baik. Nilai yang diperoleh player nantinya akan di ranking dengan seluruh nilai Q pada level lain sehingga diperoleh level dengan nilai Q yang paling mendekati nilai Q pemain.

3.6.5 Melakukan Pengurutan Alternatif

Sesudah proses kalkulasi dilakukan terhadap skor indeks kompromi (Q), hasil lengkapnya ditampilkan pada Tabel 3.18. Berdasarkan hasil tersebut, dilakukan pengurutan alternatif dengan ketentuan bahwa nilai Q terkecil berada pada peringkat pertama, sedangkan nilai Q terbesar berada pada peringkat terakhir.

Tabel 3. 18 Pengurutan Alternatif

Alternatif	v	Q	Rank
L1	0,5	1,000	16
L2		0,975	15
L3		0,870	14
L4		0,725	13
L5		0,697	12
L6		0,629	11
L7		0,475	10
L8		0,447	9
L9		0,387	8
L10		0,250	6
L11		0,225	5
L12		0,197	4
L13		0,146	3
L14		0,035	2
L15		0,000	1
LX		0,277	7

Dari hasil kalkulasi nilai Q, diketahui bahwa pemain berada pada **posisi ke-7** dengan skor Q bernilai **0,277**. Untuk menetapkan level kesukaran yang paling cocok dengan keterampilan pemain, dilakukan perbandingan terhadap dua alternatif yang posisinya paling berdekatan, yaitu alternatif peringkat **ke-6 (L10)** dan peringkat **ke-8 (L9)**. Nilai Q dari L10 sebesar **0,250**, sedangkan dari L9 sebesar **0,387**.

Langkah berikutnya adalah menghitung selisih absolut antara skor Q pemain dengan alternatif keduanya, yaitu:

- Selisih dengan L10 = $|0,250 - 0,277| = \mathbf{0,027}$
- Selisih dengan L9 = $|0,387 - 0,277| = \mathbf{0,110}$

Dikarenakan jarak terdekat berada pada L10, maka **alternatif L10** dianggap terdekat dengan performa pemain. Dengan demikian, sistem menetapkan kesulitan dari **alternatif L10** sebagai kesulitan lanjutan yang akan digunakan oleh pemain pada sesi berikutnya.

3.7 Simulasi Uji Coba Penentuan Tingkat Kesulitan

Simulasi pengujian dilakukan untuk mengevaluasi sistem penentuan tingkat kesulitan berbasis metode VIKOR yang telah diimplementasikan pada *game* 'Avoid Najasa'. Simulasi tersebut menggambarkan kemampuan pemain pada tiga kategori kemampuan yang berbeda yang bertujuan untuk memverifikasi keandalan sistem dalam memilih level kesukaran permainan menurut hasil performa pemain pada sesi sebelumnya.

Proses simulasi dilakukan dengan memasukkan data hasil permainan yang sudah diubah ke dalam skala penilaian (1–5) untuk setiap kriteria (C1–C8). Hasil tersebut kemudian dihitung memakai metode VIKOR guna menetapkan tingkat kesulitan yang paling sesuai. Simulasi dijalankan dalam tiga skenario percobaan berikut:

Percobaan Pertama – Kemampuan Rendah

Percobaan pertama menggambarkan keadaan saat pemain mempunyai kemampuan yang masih rendah. Dalam kondisi ini, pemain hanya mampu menempuh jarak pendek dengan waktu bermain singkat, serta mengumpulkan

sedikit koin dan air. Selain itu, pemain juga sering menabrak najis dan gagal menghindari rintangan dengan baik. Jumlah koin dan air yang terlewat juga cukup banyak.

Tujuan dari pengujian ini adalah guna melihat kemampuan sistem dalam menetapkan level kesukaran rendah, sehingga pemain memiliki kesempatan untuk beradaptasi dengan mekanisme permainan, memahami pola yang ada, serta mengembangkan kemampuannya secara bertahap.

Percobaan Kedua – Kemampuan Sedang

Pada uji coba kedua, kemampuan pemain digambarkan berada pada level sedang. Pemain mampu bertahan lebih lama, menempuh jarak yang cukup jauh, dan mengumpulkan jumlah koin serta air yang mendekati target. Namun, masih terdapat beberapa kesalahan seperti sesekali menabrak najis atau gagal mengambil koin dan air.

Tujuan percobaan ini yaitu guna memastikan apakah sistem bisa menentukan tingkat kesulitan menengah, yaitu memberikan tantangan yang sedikit lebih tinggi dari level sebelumnya tanpa membuat permainan menjadi terlalu sulit.

Percobaan Ketiga – Kemampuan Tinggi

Percobaan ketiga menggambarkan pemain dengan performa sangat baik. Dalam simulasi ini, pemain mampu bertahan lama, menempuh jarak jauh, mengumpulkan banyak koin, serta menghindari najis hampir secara sempurna. Jumlah air dan koin yang terlewat sangat sedikit, menandakan kemampuan adaptasi dan ketepatan tinggi.

Tujuan dari percobaan ini adalah untuk memverifikasi kemampuan sistem dalam menetapkan tingkat kesulitan tinggi yang sesuai dengan kapasitas pemain, sehingga pengalaman bermain tetap menantang dan tidak menimbulkan kejenuhan.

Setiap skenario pengujian diuji dengan menggunakan data kemampuan pemain yang sudah diubah menjadi nilai kriteria dengan skala 1–5 berdasarkan ketentuan yang tercantum pada tabel skala penilaian. Selanjutnya, sistem akan menghitung nilai S, R, dan Q sesuai rumus metode VIKOR, kemudian menentukan alternatif tingkat kesulitan dengan nilai Q terkecil sebagai hasil rekomendasi.

Proses ini diulang untuk tiga kondisi performa berbeda guna memastikan konsistensi, validitas, dan kemampuan adaptif sistem untuk menyesuaikan level kesulitan permainan menurut kemampuan aktual pemain.

Dengan pendekatan simulasi ini, diharapkan sistem ‘Avoid Najasa’ dapat menjaga keseimbangan antara tantangan dan kenyamanan bermain, sehingga pengalaman permainan menjadi optimal, adaptif, dan menyenangkan bagi pemain dengan berbagai tingkat kemampuan.

3.8 Desain Pengujian Sistem

Pengujian sistem dijalankan guna menilai apakah *game* ‘Avoid Najasa’ bisa berfungsi dengan baik dan mampu memberi pengalaman bermain yang menarik serta mudah dipahami oleh pengguna. Untuk mencapai tujuan tersebut, digunakan dua metode pengujian, yaitu *System Usability Scale* (SUS) dan *blackbox testing*.

Metode *blackbox testing* diterapkan untuk memverifikasi bahwa seluruh fitur dan fungsi dalam *game* berjalan sesuai dengan spesifikasi yang telah ditetapkan. Pengujian ini dilakukan dengan mengevaluasi hubungan antara

masukan (*input*) yang diberikan pemain dan keluaran (*output*) yang dihasilkan sistem tanpa meninjau kode program secara langsung. Aspek yang diuji meliputi fungsi tombol navigasi, mekanisme perhitungan skor, sistem permainan, deteksi tabrakan antar objek, serta perpindahan antar scene.

Selain itu, metode *System Usability Scale* (SUS) dipakai dalam mengukur aspek kemudahan penggunaan *game* berdasarkan persepsi pengguna. Instrumen ini memudahkan peneliti dalam mengevaluasi apakah sistem navigasi, antarmuka, dan berbagai fitur yang tersedia sudah memenuhi kebutuhan pengguna dan mampu memberikan pengalaman interaksi yang baik. Pengujian SUS melibatkan 20 responden yang berasal dari kalangan mahasiswa. Daftar pernyataan yang dipakai dalam kuesioner SUS disajikan pada Tabel 3.19.

Tabel 3. 19 Daftar pernyataan Kuesioner SUS

No	Pernyataan	Jenis Pernyataan
1	Saya merasa tingkat kesulitan <i>game</i> ini sesuai dengan kemampuan saya saat bermain.	Positif
2	Saya merasa tingkat kesulitan dalam <i>game</i> ini sering tidak sesuai dengan kemampuan saya.	Negatif
3	Saya merasa perubahan tingkat kesulitan antar permainan terasa wajar dan masuk akal.	Positif
4	Saya merasa perubahan tingkat kesulitan dalam <i>game</i> ini membingungkan.	Negatif
5	Saya merasa sistem dalam <i>game</i> ini mampu menyesuaikan tantangan berdasarkan performa saya.	Positif
6	Saya merasa tantangan yang diberikan <i>game</i> ini tidak pas.	Negatif
7	Saya merasa sistem penentuan tingkat kesulitan membantu saya bermain lebih nyaman.	Positif
8	Saya merasa sistem penentuan tingkat kesulitan justru mengganggu pengalaman bermain.	Negatif
9	Saya merasa lebih percaya diri bermain karena tingkat kesulitan disesuaikan dengan kemampuan saya.	Positif
10	Saya perlu menyesuaikan diri terlalu lama karena tingkat kesulitan sering berubah-ubah.	Negatif

Kuesioner *System Usability Scale* (SUS) terdiri atas 10 pernyataan yang disusun secara bergantian antara pernyataan positif dan negatif. Setiap pernyataan dinilai menggunakan skala Likert 1–5 yang merepresentasikan tingkat persetujuan responden terhadap pernyataan yang diberikan.

Perhitungan skor SUS dilakukan dengan mengurangi nilai jawaban pada pernyataan bernomor ganjil dengan angka 1, sedangkan untuk pernyataan bernomor genap diperoleh dengan mengurangi nilai jawaban dari angka 5. Seluruh nilai yang telah disesuaikan kemudian dijumlahkan dan hasilnya dikalikan dengan faktor 2,5 sehingga menghasilkan skor akhir pada rentang 0 hingga 100.

Interpretasi nilai berdasarkan metode yang diperkenalkan oleh Brooke (1996), di mana semakin tinggi nilai yang didapat, semakin bagus tingkat kegunaan (*usability*) sistem serta semakin tinggi tingkat penerimaannya oleh pengguna.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Tampilan *User Interface*

Pada bagian implementasi, rancangan antarmuka pengguna (*User Interface/UI*) yang sudah dibuat sebelumnya direalisasikan menjadi bentuk halaman visual serta elemen interaktif pada *game* ‘Avoid Najasa’. Implementasi ini meliputi semua halaman, panel, serta menu yang dipakai pemain selama proses permainan, mulai dari tampilan awal sampai akhir permainan. Setiap komponen antarmuka dirancang dengan memperhatikan aspek kemudahan penggunaan, navigasi yang intuitif, serta kesesuaian dengan tema dan konsep permainan yang diusung. Berikut disajikan hasil implementasi antarmuka pengguna pada *game* ‘Avoid Najasa’.

1. Tampilan Menu

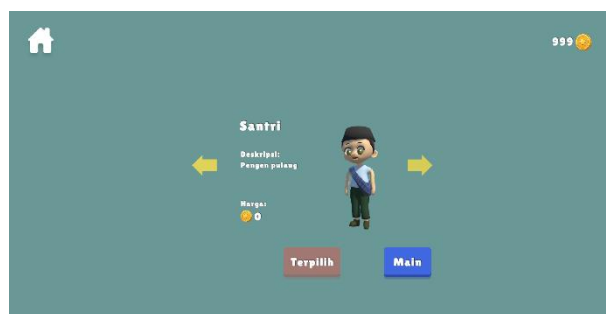
Tampilan Menu muncul ketika pemain membuka *game*. Berdasarkan Gambar 4.1, tampilan menu utama memiliki empat tombol utama yang terletak di bagian kanan, yaitu tombol Main, Opsi, Kredit, dan Keluar. Keempat tombol tersebut mempunyai fungsi untuk sarana navigasi utama yang digunakan pemain untuk mengakses berbagai fitur sebelum memulai permainan. Tampilan ini didesain dengan latar belakang lanskap perjalanan menuju masjid dengan detail seperti koin, najis, dan air, logo pada bagian kiri atas nama permainan.



Gambar 4. 1 Tampilan Menu

2. Tampilan Pemilihan

Halaman Pemilihan pada Gambar 4.2 akan tampil ketika pemain memulai permainan dengan menekan tombol 'Main'. Pemain dapat memilih karakter yang diinginkan sebelum lanjut ke *gameplay*.



Gambar 4. 2 Tampilan Pemilihan

3. Tampilan Opsi

Bagian ini ditampilkan ketika pemain menekan tombol 'Opsi' pada Menu. Pada bagian ini pemain dapat memilih untuk menonaktifkan musik maupun efek suara, seperti yang ditampilkan pada Gambar 4.3.



Gambar 4. 3 Tampilan Opsi

4. Tampilan Kredit

Panel Kredit berisi informasi terkait pengembang *game* serta ungkapan terima kasih untuk berbagai pihak yang sudah mendukung selama tahap pengembangan *game* ini. Tampilan tersebut bersifat statis dan dapat diakses oleh pemain melalui tombol Kredit yang terdapat pada menu utama, seperti tampak pada Gambar 4.4.



Gambar 4. 4 Tampilan Kredit

5. Tampilan Jeda

Ketika pemain memencet tombol Jeda, permainan akan dijeda serta menampilkan Jeda seperti pada Gambar 4.5. Panel ini mempunyai tiga opsi yaitu meneruskan permainan, mengulang kembali permainan, atau keluar ke Menu.



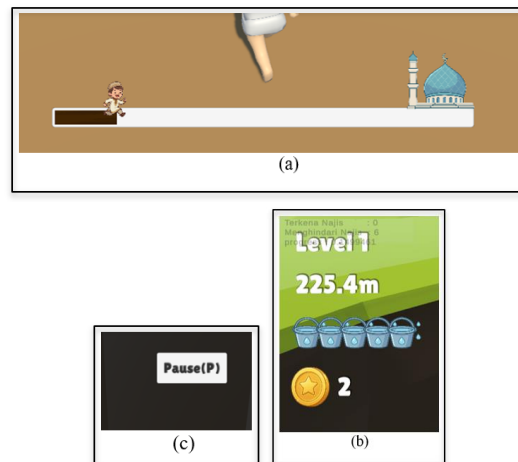
Gambar 4. 5 Tampilan Jeda

6. Tampilan *Gameplay*

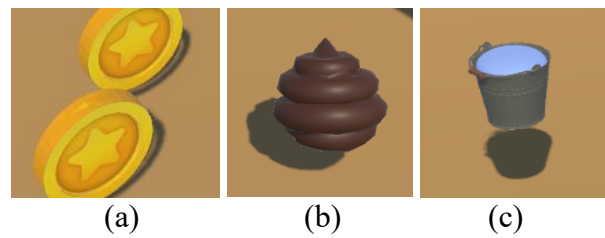
Tampilan *Gameplay* yang disajikan pada Gambar 4.6 adalah antarmuka utama yang digunakan selama permainan berlangsung. Pada tampilan ini, karakter bergerak berlari secara otomatis, sementara pemain bertugas menghindari objek najis serta mengumpulkan air dan koin yang tersedia di sepanjang permainan.

Gambar 4. 6 Tampilan *Gameplay*

Di bagian bawah layar tampak UI yang menunjukkan *progress bar*, dimana *game* akan berakhir ketika bar penuh, yaitu selama 120 detik, seperti pada Gambar 4.7 (a), pada bagian kiri berurutan dari atas ke bawah terdapat tampilan Level saat ini, Jarak ditempuh, Sisa air, Koin didapatkan pada gambar 4.7 (b), di bagian pojok kanan atas tampak tombol jeda seperti pada Gambar 4.7 (c).

Gambar 4. 7 UI pada *Gameplay*

Sepanjang jalan, pemain akan menjumpai item koin pada Gambar 4.8 (a), najis pada Gambar 4.8 (b), serta air pada Gambar 4.8 (c).

Gambar 4. 8 *Item dalam Gameplay*

7. Tampilan Skor

progress bar di bagian bawah penuh, atau ketika air habis, menjadi tanda permainan berakhir yang kemudian memunculkan panel Skor seperti pada gambar 4.9. Tampilan Skor memuat informasi mengenai hasil performa pemain dan hasil konversi menjadi skala satu sampai lima, meliputi jarak(C1), waktu(C2), persentase koin diperoleh(C3), jumlah terkena najis(C4), air didapat(C5), persentase koin terlewat(C6), jumlah menghindari najis(C7), jumlah air terlewat(C8). Kemudian terdapat rekomendasi level berikutnya berdasarkan hasil

perhitungan performa pemain, di bawahnya terdapat tombol ‘Main Lagi’ dan tombol ‘Menu’.



Gambar 4. 9 Tampilan Skor

4.2 Implementasi Perhitungan AHP

Metode AHP dipakai dalam *game* ‘Avoid Najasa’ guna menghitung bobot kriteria berdasarkan pendapat 3 pengembang *game*. Proses implementasi dilakukan dengan beberapa langkah seperti berikut:

1. Inisialisasi Matriks Perbandingan Berpasangan

Pada tahap awal, sistem menerima *input* berupa matriks perbandingan berpasangan dari beberapa pengembang *game* (*expert*). Dalam penelitian ini digunakan 3 *expert* sehingga terbentuk *array* 3 dimensi.

Pseudocode 4.1 Inisialisasi Matriks Perbandingan Berpasangan

```

Algoritma AHP
// inisialisasi matriks perbandingan berpasangan 3 ahli
Deklarasi nExpert = 3
Deklarasi nKriteria = 8
Inisialisasi matriksAhli[nExpert][nKriteria][nKriteria]

```

2. Perhitungan *Geometric Mean* Antar Pengembang *Game*

Pada langkah ini dilakukan penggabungan penilaian dari beberapa *expert* menggunakan metode *geometric mean*.

Pseudocode 4.2 Perhitungan Matriks *Geometric Mean* Antar Pengembang *Game*

```

// inisialisasi matriks geometric mean antar ahli
Inisialisasi geometricMeanMatriks[nBarisKolom][nBarisKolom]

```

```

Untuk setiap nKriteria i:
    Untuk setiap nKriteria j:
        inisialisasi produk = 1
    Untuk setiap nExpert k:
        produk = produk * matriksAhli[i][j][k]
    geometricMeanMatriks[i][j] = produk^(1/3)

```

3. Perhitungan *Geometric Mean* per Baris

Setelah diperoleh matriks gabungan, langkah selanjutnya adalah menghitung *geometric mean* untuk setiap baris.

Pseudocode 4.3 Perhitungan Matriks Geometric Mean per Baris

```

// inisialisasi matriks geometric mean per baris
Deklarasi barisGeometricMean[nKriteria]
Untuk setiap nKriteria i:
    Inisialisasi produk = 1
    Untuk setiap nKriteria j:
        produk = produk * geometricMeanMatriks[i][j]
    barisGeometricMean[i] = produk^(1/nKriteria)

```

4. Normalisasi Bobot

Nilai `barisGeometricMean` yang diperoleh belum dalam bentuk bobot akhir, sehingga perlu dinormalisasi.

Pseudocode 4.4 Perhitungan Normalisasi Bobot

```

Deklarasi bobot[nKriteria]
Inisialisasi sum = 0
Untuk setiap nKriteria i:
    sum = sum + barisGeometricMean[i]

Untuk setiap nKriteria i:
    bobot[i] = barisGeometricMean[i]/sum

```

5. *Output* Bobot Kriteria

Hasil akhir dari proses AHP adalah *array* bobot, yang merepresentasikan tingkat kepentingan masing-masing kriteria.

Pseudocode 4.5 Fungsi getWeights()

```

Fungsi getWeights()
    Return bobot

```

4.3 Implementasi Perhitungan VIKOR

Metode VIKOR diterapkan pada *game* ‘Avoid Najasa’ guna menghasilkan saran level yang sesuai dengan kemampuan pemain. Tahap penerapannya dihitung melalui beberapa langkah yang dijelaskan seperti berikut:

1. Transformasi Data Performa(*Scaling*)

Sebelum masuk ke proses VIKOR, data mentah performa pemain diubah menjadi skala diskrit 1-5 menggunakan fungsi *Scaling*.

Pseudocode 4.6 Transformasi Data Performa Pemain

```

Algoritma Scaling
IF jarak < 721 set c1 = 1
ELSE IF jarak ≤ 820 set c1 = 2
ELSE IF jarak ≤ 920 set c1 = 3
ELSE IF jarak ≤ 1020 set c1 = 4
ELSE set c1 = 5
IF waktuBertahan < 30 set c2 = 1
ELSE IF waktuBertahan ≤ 59 set c2 = 2
ELSE IF waktuBertahan ≤ 89 set c2 = 3
ELSE IF waktuBertahan ≤ 119 set c2 = 4
ELSE set c2 = 5
IF koinPersen ≤ 20 set c3 = 1
ELSE IF koinPersen ≤ 40 set c3 = 2
ELSE IF koinPersen ≤ 60 set c3 = 3
ELSE IF koinPersen ≤ 80 set c3 = 4
ELSE set c3 = 5
IF najisHit > 12 set c4 = 1
ELSE IF najisHit ≥ 10 set c4 = 2
ELSE IF najisHit ≥ 7 set c4 = 3
ELSE IF najisHit ≥ 4 set c4 = 4
ELSE set c4 = 5
IF water < 4 set c5 = 1
ELSE IF water ≤ 6 set c5 = 2
ELSE IF water ≤ 9 set c5 = 3
ELSE IF water ≤ 12 set c5 = 4
ELSE set c5 = 5
IF koinMissPercent > 80 set c6 = 1
ELSE IF koinMissPercent ≥ 61 set c6 = 2
ELSE IF koinMissPercent ≥ 41 set c6 = 3
ELSE IF koinMissPercent ≥ 21 set c6 = 4
ELSE set c6 = 5
IF najisAvoided < 6 set c7 = 1
ELSE IF najisAvoided ≤ 10 set c7 = 2
ELSE IF najisAvoided ≤ 15 set c7 = 3
ELSE IF najisAvoided ≤ 20 set c7 = 4
ELSE set c7 = 5
IF waterMiss > 12 set c8 = 1

```

```

ELSE IF waterMiss  $\geq$  10 set c8 = 2
ELSE IF waterMiss  $\geq$  7 set c8 = 3
ELSE IF waterMiss  $\geq$  4 set c8 = 4
ELSE set c8 = 5

```

2. Penentuan Nilai Terbaik dan Terburuk

Tahap pertama pada metode VIKOR yaitu menetapkan nilai maksimum serta minimum pada seluruh kriteria dari seluruh alternatif yang ada. Skor terbaik (f^+) merepresentasikan kemampuan tertinggi, sedangkan skor terburuk (f^-) menunjukkan kemampuan terendah.

Pseudocode 4.7 Nilai Terbaik dan Terburuk

```

Algoritma VIKOR
Inisialisasi bobot[]
Inisialisasi v = 0,5
Inisialisasi alternatifMatriks[][]
Inisialisasi baris = panjang alternatifMatriks
Inisialisasi kolom = lebar alternatifMatriks
Deklarasi S[] = baris
Deklarasi R[] = baris

Inisialisasi fstar[] = 5 sebanyak kolom
Inisialisasi fminus[] = 1 sebanyak kolom

```

3. Perhitungan Nilai S dan R

Untuk setiap alternatif, dihitung dua nilai yaitu nilai S (*Utility Measure*) dan nilai R (*Regret Measure*).

Pseudocode 4.8 Perhitungan Nilai S dan R

```

Untuk setiap baris i:
  Inisialisasi sTotal = 0
  Inisialisasi rMax =  $-\infty$ 

  Untuk setiap kolom j:
    Inisialisasi value = (fstar[j] - alternatifMatriks[i][j]) / (fstar[j] - fminus[j])
    Inisialisasi wv = bobot * value
    sTotal = sTotal + wv
    Jika wv > rMax
      rMax = wv

  S[i] = sTotal
  R[i] = rMax

```


Nilai S dan R kemudian dinormalisasi menggunakan nilai minimum dan maksimum.

Pseudocode 4.9 Nilai Minimum dan Maksimum S dan R

```
Inisialisasi Smin = MIN(S)
Inisialisasi Smax = MAX(S)
Inisialisasi Rmin = MIN(R)
Inisialisasi Rmax = MAX(R)
```

4. Perhitungan Nilai Q (Indeks VIKOR)

Nilai Q merupakan indeks kompromi antara kepuasan keseluruhan (S) dan penyesalan maksimum (R).

Pseudocode 4.10 Perhitungan Nilai Q

```
Deklarasi Q[] = baris
Untuk setiap baris i:
    Q[i] = v * (S[i] - Smin) / (Smax - Smin) + (1 - v) * (R[i] - Rmin) / (Rmax - Rmin)
```

5. Perhitungan Nilai Q Pemain

Sistem kemudian menghitung nilai Q untuk performa pemain.

Pseudocode 4.11 Perhitungan Nilai Q Pemain

```
Inisialisasi LX[] // Data performa pemain
Inisialisasi sTotalP = 0
Inisialisasi rMaxP = -∞
Untuk setiap kriteria j dalam LX:
    jarakP = (5 - LX[j]) / 4
    wvP = bobot[j] * jarakP
    sTotalP = sTotalP + wvP
    Jika wvP > rMaxP
        rMaxP = wvP
//Normalisasi Q Pemain
QPlayer = v*(sTotalP - Smin)/(Smax - Smin) + (1 - V)*(rMaxP - Rmin)/(Rmax - Rmin)
```

6. Pemilihan Level Berdasarkan Kedekatan Q

Kemudian mencari alternatif dengan nilai Q yang paling mendekati nilai Q pemain.

Pseudocode 4.12 Pemilihan Level

```

Inisialisasi levelTerpilih = 0
Inisialisasi selisihTerkecil = ∞
Untuk setiap baris i:
    Inisialisasi diff = ABS(Q[i] - qPlayer)
    Jika diff < selisihTerkecil
        selisihTerkecil = diff
        levelTerpilih = i

```

7. Output Level Kesulitan

Hasil akhir berupa level yang diperoleh dari nilai Q dengan selisih terkecil dengan nilai Q pemain.

Pseudocode 4.13 Hasil Level

```

Return levelTerpilih + 1

```

4.4 Uji Coba Penentuan Tingkat Kesulitan

Setelah sistem pemilihan tingkat kesulitan dengan metode AHP–VIKOR selesai diterapkan, terdapat rangkaian pengujian guna memastikan kemampuan sistem untuk memilih tingkat kesulitan permainan menurut kemampuan pemain. Pengujian ini bertujuan guna mengetahui bahwasanya sistem dapat menentukan kemampuan pemain secara tepat serta menentukan tingkat kesulitan yang cocok menurut skor Q yang diperoleh dari kalkulasi AHP–VIKOR. Metode pengujian mengacu pada penelitian yang dilakukan oleh Hartanti (2025). Pengujian dilakukan melalui simulasi performa pemain pada tiga skenario yang berbeda, yaitu performa buruk, sedang, dan baik. Setiap simulasi dimulai pada level 1 yang berperan sebagai tingkat kesulitan awal (*default*) saat permainan dimulai. Simulasi ini dirancang untuk merepresentasikan variasi kemampuan pemain sehingga dapat digunakan untuk memastikan keandalan sistem dalam menentukan level kesulitan permainan secara adaptif menurut hasil yang dicapai pada permainan sebelumnya. Melalui

simulasi tersebut, sistem diharapkan mampu menghasilkan rekomendasi tingkat kesulitan yang sesuai dengan kondisi pemain, sehingga tingkat tantangan yang diberikan tetap seimbang dan dapat meningkatkan pengalaman bermain secara keseluruhan.

Proses simulasi dilakukan dengan memasukkan data hasil permainan yang sudah dikonversi menjadi skala penilaian (1–5) untuk setiap kriteria (C1–C8). Hasil tersebut kemudian diproses dengan metode VIKOR guna memilih tingkat kesulitan yang paling sesuai. Dimana dalam perhitungan VIKOR menggunakan bobot yang telah dihasilkan oleh perhitungan AHP sebelumnya.

Percobaan Pertama (Performa Buruk)

Pada percobaan pertama, pemain mendapat hasil performa seperti pada Gambar 4.10 berikut.



Gambar 4. 10 Data Pemain pada Percobaan Pertama

Pada percobaan ini, pemain dengan kemampuan rendah, dengan jarak yang dicapai hanya 200m, dan hanya bertahan selama 15 detik, koin yang didapat hanya 3 dan koin yang terlewat berjumlah 5, air yang berhasil diperoleh hanya 1, kemudian jumlah menghindari najis 9 kali, dan terkena najis 6 kali, yang kemudian

di transformasi menjadi skala 1-5 seperti pada Gambar 4.10. Dari hasil penilaian tersebut kemudian dihitung nilai Q-nya, dan diperoleh rekomendasi level 3 seperti pada Gambar 4.11.



Gambar 4. 11 Hasil Sistem pada Percobaan Pertama

Hasil perhitungan Q pemain memperoleh nilai 0,892, dimana nilai Q terdekat pada alternatif Level 3 dengan nilai Q 0,871.

Tabel 4. 1 Data Nilai Q Percobaan Pertama

Alternatif	Nilai Q	Ranking
L1	1,000	16
L2	0,975	15
L3	0,870	13
L4	0,725	12
L5	0,697	11
L6	0,629	10
L7	0,475	9
L8	0,447	8
L9	0,387	7
L10	0,250	6
L11	0,225	5

L12	0,197	4
L13	0,146	3
L14	0,035	2
L15	0,000	1
LX	0,892	14

Validasi dilakukan melalui perhitungan manual dengan Ms. Excel guna memastikan bahwa hasil yang didapat sistem sesuai dengan perhitungan yang seharusnya. Pada Tabel 4.1 tampak bahwasanya skor *Q Player* berada pada urutan ke-14 dengan selisih terdekat bernilai 0,022 pada Alternatif L3 sehingga Alternatif L3 menjadi tingkat kesulitan yang direkomendasikan oleh sistem.

Percobaan Kedua (Performa Sedang)

Pada percobaan kedua, pemain mendapat skor seperti pada Gambar 4.12.



Gambar 4. 12 Data Pemain pada Percobaan Kedua

Pada percobaan ini, pemain dengan kemampuan menengah, dengan jarak yang dicapai 1303,2 m, bertahan selama 91,3 detik, koin yang didapat sebanyak 63 koin, air yang berhasil diperoleh 9 dan yang terlewat hanya 1, kemudian jumlah menghindari najis 48 kali, dan terkena najis 12 kali, yang kemudian di transformasi menjadi skala 1-5 seperti pada Gambar 4.12. Dari hasil penilaian tersebut kemudian

dihitung nilai Q-nya, dan diperoleh rekomendasi level 10 tampak pada Gambar 4.13.



Gambar 4. 13 Hasil Sistem pada Percobaan Kedua

Hasil perhitungan Q pemain memperoleh nilai 0,278, dimana nilai Q terdekat pada alternatif Level 10 dengan nilai Q 0,250.

Tabel 4. 2 Data Nilai Q Percobaan Kedua

Alternatif	Nilai Q	Ranking
L1	1,000	16
L2	0,975	15
L3	0,870	14
L4	0,725	13
L5	0,697	12
L6	0,629	11
L7	0,475	10
L8	0,447	9
L9	0,387	8
L10	0,250	6
L11	0,225	5
L12	0,197	4

Alternatif	Nilai Q	Ranking
L13	0,146	3
L14	0,035	2
L15	0,000	1
LX	0,277	7

Kalkulasi manual menggunakan Ms. Excel dilakukan guna menguji kecocokan hasil perhitungan sistem. Pada Tabel 4.2 tampak bahwasanya skor Q *Player* berada pada peringkat ke-7 dengan selisih terkecil bernilai 0,027 pada Alternatif L10 sehingga Alternatif L10 menjadi tingkat kesulitan yang direkomendasikan oleh sistem.

Percobaan Ketiga (Performa Bagus)

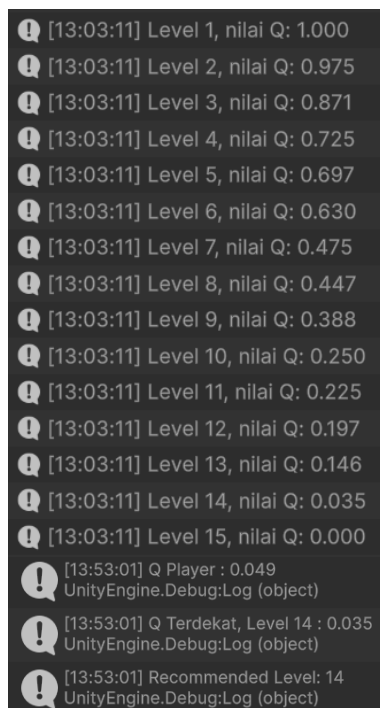
Pada percobaan ketiga, pemain mendapat skor seperti pada Gambar 4.14.



Gambar 4. 14 Data Pemain pada Percobaan Ketiga

Pada percobaan ini, pemain dengan kemampuan bagus, dengan jarak yang dicapai 1765,6 m , dan bertahan hingga 120 detik, koin yang didapat 80 dan koin yang terlewat berjumlah 4, air yang berhasil diperoleh hingga 12 dan air yang terlewat hanya 2, kemudian jumlah menghindari najis 86 kali, dan terkena najis hanya 2 kali, yang kemudian di transformasi menjadi skala 1-5 tampak pada

Gambar 4.14. Dari hasil penilaian tersebut kemudian dihitung nilai Q-nya, dan diperoleh rekomendasi level 14 seperti pada Gambar 4.15.



Gambar 4. 15 Hasil Sistem pada Percobaan Ketiga

Hasil perhitungan Q pemain memperoleh nilai 0,049, dimana nilai Q terdekat pada alternatif Level 14 dengan nilai Q 0,035.

Tabel 4. 3 Data Nilai Q Percobaan Pertama

Alternatif	Nilai Q	Ranking
L1	1,000	16
L2	0,975	15
L3	0,870	14
L4	0,725	13
L5	0,697	12
L6	0,629	11
L7	0,475	10
L8	0,447	9
L9	0,387	8
L10	0,250	7
L11	0,225	6
L12	0,197	5
L13	0,146	4

Alternatif	Nilai Q	Ranking
L14	0,035	2
L15	0,000	1
LX	0,049	3

Proses perhitungan manual dilakukan dengan Ms. Excel sebagai langkah validasi terhadap implementasi sistem guna memastikan akurasi hasil perhitungan yang dihasilkan. Pada Tabel 4.3 tampak bahwasanya skor Q *Player* berada pada peringkat ke-3 dengan perbedaan terkecil bernilai 0,014 pada Alternatif L14 sehingga Alternatif L14 menjadi tingkat kesulitan yang direkomendasikan oleh sistem.

4.5 Hasil Pengujian

Pengujian dilakukan dengan melibatkan 20 mahasiswa/i yang terbagi menjadi dua jenis, yaitu 10 mahasiswa yang mengerti *game development*, dan 10 mahasiswa umum.

4.5.1 Blackbox Testing

Pengujian *blackbox* dilakukan sebagai tahap awal untuk memastikan bahwa sistem berfungsi sesuai dengan kebutuhan yang telah ditetapkan tanpa memperhatikan struktur atau kode program yang digunakan. Pengujian ini berfokus pada pemeriksaan hubungan antara masukan (*input*) dan keluaran (*output*) yang dihasilkan sistem. Seluruh fitur pada *game* 'Avoid Najasa' diuji berdasarkan skenario yang telah ditentukan guna memastikan setiap fungsi dapat berjalan dengan baik dan memberikan hasil yang sesuai dengan yang diharapkan.

1. Pengujian Halaman *Menu*

Tabel 4. 4 Pengujian Halaman *Menu*

No.	Fitur	Skenario	Input	Output yang diharapkan	Status
1	Tombol Main	Menekan tombol Main	Klik tombol Main	Berpindah ke halaman Pemilihan	Berhasil
2	Tombol Opsi	Menekan tombol Opsi	Klik tombol Opsi	Berpindah ke halaman Opsi	Berhasil
3	Tombol Kredit	Menekan tombol Kredit	Klik tombol Kredit	Berpindah ke halaman Kredit	Berhasil
4	Tombol Keluar	Menekan tombol Keluar	Klik tombol Keluar	Menutup Permainan	Berhasil

Pada Tabel 4.4, pengujian tampilan *Menu* memperlihatkan bahwasanya seluruh tombol navigasi utama sudah berfungsi seperti yang diharapkan. Pengguna bisa berpindah ke Halaman lain, hingga keluar dari permainan tanpa error menunjukkan bahwa sistem navigasi utama sudah berfungsi dengan baik.

2. Pengujian Halaman Pemilihan

Tabel 4. 5 Pengujian Halaman Pemilihan

No.	Fitur	Skenario	Input	Output yang diharapkan	Status
1	Tombol Panah Kanan	Menekan tombol Panah kanan	Klik tombol Panah kanan	Menampilkan karakter berikutnya	Berhasil
2	Tombol Panah Kiri	Menekan tombol Panah kiri	Klik tombol Panah kiri	Menampilkan karakter sebelumnya	Berhasil
3	Tombol Pilih	Menekan tombol Pilih	Klik tombol Pilih	Tombol Pilih menjadi Terpilih dan nonaktif	Berhasil
4	Tombol Main	Menekan tombol Main	Klik tombol Main	Berpindah ke halaman <i>Gameplay</i>	Berhasil
5	Tombol <i>Menu</i>	Menekan tombol <i>Menu</i>	Klik tombol <i>Menu</i>	Kembali ke halaman <i>Menu</i>	Berhasil

Pada Tabel 4.5, Tampak hasil uji pada Halaman pemilihan, seluruh tombol mulai dari tombol panah kanan dan kiri untuk berganti karakter, tombol pilih,

hingga tombol main sudah berfungsi seperti yang diharapkan, ini menunjukkan pada halaman pemilihan sudah bekerja dengan baik.

3. Pengujian Halaman *Gameplay*

Tabel 4. 6 Pengujian Halaman *Gameplay*

No.	Fitur	Skenario	Input	Output yang diharapkan	Status
1	Tombol Mengerti	Menekan tombol Mengerti setelah membaca tutorial	Klik tombol Mengerti	<i>Game</i> dimulai	Berhasil
2	Mengambil Air	Mengambil Air	Menyentuh <i>item</i> Air	<i>Counter</i> Air bertambah	Berhasil
3	Mengambil Koin	Mengambil Koin	Menyentuh <i>item</i> Koin	<i>Counter</i> Koin bertambah	Berhasil
4	Terkena Najis	Terkena Najis	Menyentuh <i>item</i> Najis	<i>Counter</i> Air berkurang 1	Berhasil
5	Tombol Jeda	Menekan tombol Jeda atau P pada <i>keyboard</i>	Klik tombol Jeda atau P pada <i>keyboard</i>	Permainan berhenti dan Memunculkan panel Jeda	Berhasil
6	Permainan berakhir	Kehabisan persediaan Air	Menyentuh <i>item</i> najis hingga persediaan Air habis	Memunculkan panel Skor	Berhasil
7	Permainan berakhir	Bertahan selama 120 detik yaitu hingga <i>progress bar</i> penuh tanpa kehabisan Air	Waktu bertahan mencapai 120 detik	Memunculkan panel Skor	Berhasil

Pada Tabel 4.6, hasil pengujian pada halaman *Gameplay*, mulai dari tombol jeda, setiap item yang berinteraksi dengan pemain, hingga pemicu halaman Skor yang menandakan permainan berakhir, semuanya sudah berfungsi seperti yang diharapkan, ini menunjukkan bahwa halaman *Gameplay* sudah berfungsi seperti yang diharapkan.

4. Pengujian Panel Jeda

Tabel 4. 7 Pengujian Panel Jeda

No.	Fitur	Skenario	Input	Output yang diharapkan	Status
1	Tombol Lanjut	Menekan tombol Lanjut atau P	Klik tombol Lanjut atau P pada <i>keyboard</i>	Menutup panel Jeda dan permainan berlanjut	Berhasil
2	Tombol <i>Restart</i>	Menekan tombol <i>Restart</i> atau R	Klik tombol <i>Restart</i> atau R pada <i>keyboard</i>	Menutup panel Jeda dan mengulang permainan	Berhasil
3	Tombol Keluar	Menekan Keluar atau Q	Klik Keluar atau Q pada <i>keyboard</i>	Berpindah ke halaman <i>Menu</i>	Berhasil

Pada Tabel 4.7, memperlihatkan hasil validasi pada panel Jeda, semua tombol tombol sudah berfungsi seperti yang diharapkan, mulai dari melanjutkan permainan, memulai kembali permainan, dan keluar dari permainan ke halaman Menu.

5. Pengujian Panel Skor

Tabel 4. 8 Pengujian Panel Skor

No.	Fitur	Skenario	Input	Output yang diharapkan	Status
1	Tombol Main Lagi	Menekan tombol Main Lagi	Klik tombol Main Lagi	Bermain kembali dengan level yang direkomendasikan	Berhasil
2	Tombol Menu	Menekan tombol Menu	Klik tombol Menu	Berpindah ke halaman Menu	Berhasil

Pada Tabel 4.8, memperlihatkan hasil pengujian pada panel Skor. Pada panel skor hanya ada dua tombol, yakni tombol Menu serta Main Lagi, pada tombol Main Lagi sudah berfungsi yaitu permainan dimulai kembali dengan tingkat kesulitan yang sudah direkomendasikan, dan tombol Menu sudah berhasil mengalihkan ke halaman *Menu*.

4.5.2 System Usability Scale (SUS)

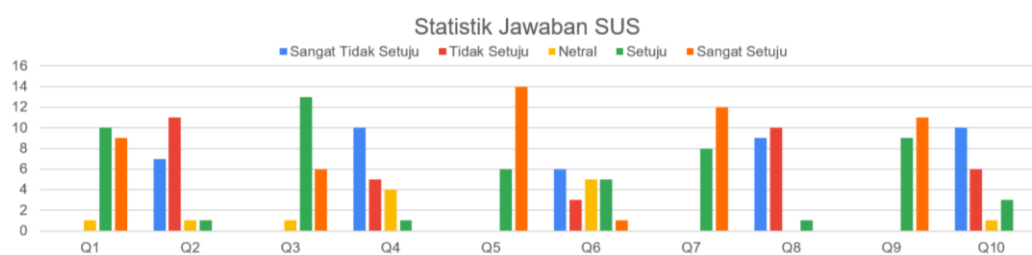
Metode SUS dipakai dalam mengevaluasi aspek kegunaan dari sistem atau *game* yang sedang dikembangkan (Hartanti, 2025). Pada riset ini, melibatkan 20 partisipan yang terdiri atas 10 Mahasiswa yang memahami *game development* dan 10 Mahasiswa Umum. Pertanyaan pada kuesioner SUS dalam penelitian ini disesuaikan dengan pembahasan pada *game* yang dikembangkan, yaitu pada penentuan tingkat kesulitannya. Terdapat 10 pernyataan yang terdiri dari 5 pernyataan positif dan 5 pernyataan negatif. Kategori tersebut ditampilkan pada Tabel 4.9 berikut.

Tabel 4. 9 Daftar Pertanyaan SUS

No.	Pertanyaan	Kategori
1	Saya merasa tingkat kesulitan <i>game</i> ini sesuai dengan kemampuan saya saat bermain.	Positif
2	Saya merasa tingkat kesulitan dalam <i>game</i> ini sering tidak sesuai dengan kemampuan saya.	Negatif
3	Saya merasa perubahan tingkat kesulitan antar permainan terasa wajar dan masuk akal.	Positif
4	Saya merasa perubahan tingkat kesulitan dalam <i>game</i> ini membingungkan.	Negatif
5	Saya merasa sistem dalam <i>game</i> ini mampu menyesuaikan tantangan berdasarkan performa saya.	Positif
6	Saya merasa tantangan yang diberikan <i>game</i> ini tidak pas.	Negatif
7	Saya merasa sistem penentuan tingkat kesulitan membantu saya bermain lebih nyaman.	Positif
8	Saya merasa sistem penentuan tingkat kesulitan justru mengganggu pengalaman bermain.	Negatif
9	Saya merasa lebih percaya diri bermain karena tingkat kesulitan disesuaikan dengan kemampuan saya.	Positif
10	Saya perlu menyesuaikan diri terlalu lama karena tingkat kesulitan sering berubah-ubah.	Negatif

Setelah mencoba *game* yang telah dikembangkan, responden diminta untuk mengisi kuesioner SUS yang terdiri atas 10 pernyataan dengan skala Likert, mulai

dari 1 (Sangat Tidak Setuju) hingga 5 (Sangat Setuju). Untuk memberikan gambaran yang lebih jelas mengenai tanggapan responden terhadap setiap pernyataan, distribusi jawaban responden dapat dilihat melalui diagram statistik pada masing-masing item SUS.



Gambar 4. 16 Statistik Jawaban SUS

Berdasarkan Figur 4.16, terlihat sebaran respon partisipan pada setiap pernyataan SUS. Pada pernyataan positif (Q1, Q3, Q5, Q7, dan Q9), mayoritas partisipan memberi respon setuju dan sangat setuju. Hasil ini mengindikasikan bahwa responden memiliki pandangan yang baik pada mekanisme penentuan tingkat kesulitan yang diterapkan dalam game. Sebaliknya, pada pernyataan negatif (Q2, Q4, Q6, Q8, dan Q10), sebagian besar partisipan memberi respon tak setuju dan sangat tak setuju. Temuan tersebut mengindikasikan bahwa responden menilai sistem penentuan tingkat kesulitan pada game yang dikembangkan telah berjalan dengan baik dan sesuai dengan harapan pengguna.

Tabel 4. 10 Hasil Skor Asli SUS

No.	Pertanyaan									
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R1	3	4	4	3	5	4	5	2	5	4
R2	5	1	5	1	5	4	5	2	4	1
R3	4	2	5	1	5	3	5	1	5	2
R4	4	2	4	1	4	4	4	2	5	2
R5	5	1	4	2	5	4	5	1	5	1
R6	4	1	4	1	5	1	5	1	5	1

No.	Pertanyaan									
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10
R7	5	1	5	1	5	1	5	1	4	2
R8	5	2	5	3	5	5	4	1	5	4
R9	4	2	5	1	4	2	4	1	4	1
R10	4	1	4	3	5	3	5	1	4	1
R11	5	2	5	1	5	1	5	1	5	2
R12	4	2	4	4	4	3	5	2	5	1
R13	4	1	4	2	5	2	4	4	4	4
R14	4	2	4	3	4	3	4	2	4	3
R15	4	2	3	2	4	3	4	2	5	2
R16	5	1	4	1	5	1	4	1	4	1
R17	5	3	4	1	5	1	4	2	4	1
R18	5	2	4	1	4	1	5	2	5	1
R19	4	2	4	2	5	2	5	2	4	2
R20	5	2	4	2	5	4	5	2	5	1

Tabel 4.10 menampilkan skor mentah (*raw score*) SUS yang diperoleh dari setiap pernyataan pada masing-masing responden. Data tersebut menggambarkan persepsi awal pengguna terhadap sistem tanpa melalui proses perhitungan atau transformasi skor, sehingga dapat digunakan untuk melihat tanggapan langsung responden mengenai mekanisme penentuan tingkat kesulitan pada *game* yang dievaluasi.

Sebelum dilakukan analisis, nilai pada kuesioner SUS terlebih dahulu dikonversi sesuai dengan aturan perhitungan SUS. Untuk pernyataan positif, skor konversi diperoleh dengan mengurangi nilai *jawaban* dengan 1 ($\text{jawaban} - 1$), sedangkan untuk pernyataan negatif digunakan rumus 5 dikurangi nilai *jawaban* ($5 - \text{jawaban}$). Selanjutnya, seluruh skor hasil konversi dijumlahkan dan hasilnya dikalikan dengan 2,5 untuk mendapatkan skor SUS masing-masing responden. Berikut disajikan contoh perhitungan SUS pada salah satu responden, yaitu R2:

$$Q1 = 5 - 1 = 4$$

$$Q2 = 5 - 1 = 4$$

$$Q3 = 5 - 1 = 4$$

$$Q4 = 5 - 1 = 4$$

$$Q5 = 5 - 1 = 4$$

$$Q6 = 5 - 4 = 1$$

$$Q7 = 5 - 1 = 4$$

$$Q8 = 5 - 2 = 3$$

$$Q9 = 4 - 1 = 3$$

$$Q10 = 5 - 1 = 4$$

$$\text{Total Skor R2} = 4 + 4 + 4 + 4 + 4 + 1 + 4 + 3 + 3 + 4 = 35$$

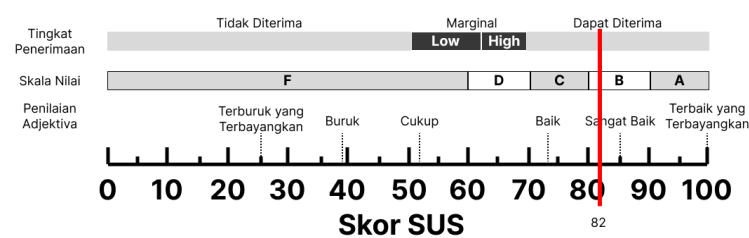
$$\text{Skor SUS} = 35 \times 2,5 = 87,5$$

Setelah diperoleh seluruh skor SUS responden, kemudian dijumlahkan untuk memperoleh skor total yang kemudian dibagi total partisipan guna mendapat Rata-Rata Skor SUS atau Hasil Akhir. Hasil Akhir perhitungan Skor SUS tampak pada Tabel 4.11 berikut.

Tabel 4. 11 Hasil Akhir Skor SUS

No.	Pertanyaan										Jumlah	Nilai
	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
R1	2	1	3	2	4	1	4	3	4	1	25	62,5
R2	4	4	4	4	4	1	4	3	3	4	35	87,5
R3	3	3	4	4	4	2	4	4	4	3	35	87,5
R4	3	3	3	4	3	1	3	3	4	3	30	75
R5	4	4	3	3	4	1	4	4	4	4	35	87,5
R6	3	4	3	4	4	4	4	4	4	4	38	95
R7	4	4	4	4	4	4	4	4	3	3	38	95
R8	4	3	4	2	4	0	3	4	4	1	29	72,5
R9	3	3	4	4	3	3	3	4	3	4	34	85
R10	3	4	3	2	4	2	4	4	3	4	33	82,5
R11	4	3	4	4	4	4	4	4	4	3	38	95
R12	3	3	3	1	3	2	4	3	4	4	30	75
R13	3	4	3	3	4	3	3	1	3	1	28	70
R14	3	3	3	2	3	2	3	3	3	2	27	67,5
R15	3	3	2	3	3	2	3	3	4	3	29	72,5
R16	4	4	3	4	4	4	3	4	3	4	37	92,5
R17	4	2	3	4	4	4	3	3	3	4	34	85
R18	4	3	3	4	3	4	4	3	4	4	36	90
R19	3	3	3	3	4	3	4	3	3	3	32	80
R20	4	3	3	3	4	1	4	3	4	4	33	82,5
Total Skor											1640	
Rata-Rata Skor (Hasil Akhir)											82	

Menurut hasil perhitungan Skor SUS pada 20 responden yang telah di rata-rata, diperoleh Skor Akhir SUS yaitu 82. Berdasarkan standar SUS, nilai melebihi 68 menunjukkan tingkat kegunaan yang memadai, sedangkan skor diatas 80 tergolong sangat baik (Haunan, 2025). Hasil tersebut menunjukkan bahwa sistem penentuan tingkat kesulitan pada *game* 'Avoid Najasa' memiliki Tingkat penggunaan yang tergolong sangat baik.



Gambar 4. 17 Grafik Hasil SUS

Pada Gambar 4.17 memperlihatkan interpretasi dari nilai SUS yang diperoleh. Pada pengujian ini, rata-rata skor SUS yaitu 82, yang tergolong kategori “Sangat Baik” serta tingkat penerimaan “Dapat Diterima”. Skor tersebut mengindikasikan bahwasanya penentuan tingkat kesulitan dalam *game* ‘Avoid Najasa’ mempunyai tingkat kegunaan yang sangat baik serta bisa diterima oleh pengguna.

4.6 Integrasi dalam Islam

Penelitian ini membawa nilai-nilai Islam dengan pengembangan tingkat kesulitan dalam *game* ‘Avoid Najasa’. Integrasi keilmuan Islam pada riset ini meliputi dua aspek, yakni hubungan manusia dengan Allah (*Mu’amalah ma’a Allah*), serta hubungan manusia dengan manusia (*Mu’amalah ma’annas*).

1. *Mu'amalah Ma'a Allah*

Mu'amalah ma'a Allah adalah hubungan manusia dengan Allah. Islam mengajarkan bahwasanya setiap ujian yang ditanggung manusia tidak melebihi batas kemampuan yang telah Allah tetapkan bagi dirinya. Dalam Al-Qur'an Surah Al-Baqarah ayat 286, Allah *Subhanahu wa ta'ala* berfirman:

...لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya...”
(Q.S Al-Baqarah: 286)

Menurut Tafsir Kementerian Agama Republik Indonesia, ayat ini mengajarkan bahwasanya setiap cobaan atau kesulitan yang diberikan kepada manusia telah diatur dengan kesanggupan tiap manusia. Tak ada ujian yang diberikan di luar batas kesanggupan manusia. Hal ini tercermin dalam sistem pemilihan tingkat kesulitan dalam *game* ‘Avoid Najasa’ yang disesuaikan berdasarkan performa pemain. Konsep ini mendorong pemain untuk terus berupaya serta berkembang, karena setiap kesulitan yang dihadapi masih berada pada batas kemampuan yang bisa mereka atasi. Dengan demikian, permainan ini tak hanya memberikan hiburan, namun juga merefleksikan nilai Islam bahwa setiap kesulitan adalah ujian yang proporsional dan dapat dilalui dengan usaha serta ketekunan.

2. *Mu'amalah Ma'a an-Nas*

Muamalah Mu'Annas merujuk pada interaksi dan hubungan manusia dengan sesamanya. Islam mendorong setiap individu untuk membantu dan memberikan manfaat kepada orang lain, terutama dalam menghadapi kesulitan. Hal ini tercermin dalam sabda Rasulullah SAW:

خَيْرُ النَّاسِ أَنْفَعُهُمْ لِلنَّاسِ

“Sebaik-baik manusia adalah yang paling bermanfaat bagi manusia.” (HR. Ahmad, ath-Thabrani, ad-Daruqutni. Hadits ini dihasankan oleh al-Albani di dalam Shahihul Jami’ no: 3289)

Hadits ini menjelaskan bahwa sebaik baiknya individu ialah yang paling bermanfaat kepada sesama. Salah satu bentuknya yaitu penelitian ini yang turut berkontribusi dalam pengembangan *game* yang berfokus pada tingkat kesulitan yang adaptif.

Dengan melakukan penelitian ini, penulis sebenarnya tengah menerapkan prinsip *muamalah mu’annas*, yakni berusaha bermanfaat dengan turut berkontribusi dalam penelitian yang dapat menjadi acuan bagi peneliti lain yang mengembangkan penelitian mengenai *game* khususnya yang berfokus pada tingkat kesulitan adaptif.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan rangkaian proses analisis, perancangan, implementasi, dan evaluasi sistem yang telah dilaksanakan, penelitian ini berhasil menghasilkan sebuah *game* berjudul ‘Avoid Najasa’ dengan *genre endless runner* menggunakan metode AHP-VIKOR yang berfokus pada penyesuaian tingkat kesulitan permainan secara adaptif. *Game* ini dirancang dengan menggabungkan unsur hiburan dan tantangan, di mana pemain harus menghindari rintangan berupa najis serta mengumpulkan item tertentu selama permainan berlangsung. Sistem dalam *game* ini tidak hanya bersifat statis, tetapi mampu menyesuaikan tingkat kesulitan menurut kemampuan pemain sehingga pengalaman bermain menjadi lebih seimbang serta tak monoton.

Untuk mengatasi permasalahan ketidaksesuaian tingkat kesulitan dalam permainan, riset ini menerapkan metode AHP (*Analytical Hierarchy Process*) dan VIKOR sebagai sistem pendukung keputusan. Metode AHP dimanfaatkan guna menghitung bobot dari delapan kriteria performa pemain, yakni jarak tempuh, durasi bertahan, persentase koin yang diperoleh, jumlah terkena najis, jumlah air yang dikumpulkan, persentase koin yang terlewat, jumlah najis yang berhasil dilewati, serta total air yang terlewat. Selanjutnya, metode VIKOR digunakan dalam pemilihan alternatif tingkat kesulitan yang paling cocok dengan kondisi pemain berdasarkan nilai kompromi (Q) yang dihasilkan.

Hasil implementasi menunjukkan bahwasanya sistem mampu menyesuaikan tingkat kesulitan permainan dengan dinamis sesuai dengan kemampuan pemain, sehingga permainan tidak terasa terlalu mudah maupun terlalu sulit. Metode ini memberi pengalaman bermain yang lebih adaptif dan personal, serta mampu meningkatkan keseimbangan antara tantangan dan kemampuan pemain. Selain itu, hasil pengujian memakai metode *System Usability Scale* (SUS) terhadap 20 responden menghasilkan skor rata-rata bernilai 82, yang tergolong pada kategori sangat baik. Hal ini mengindikasikan bahwasanya *game* yang dikembangkan bisa diterima dengan baik oleh pengguna, baik dari segi kemudahan penggunaan maupun kenyamanan dalam bermain.

Dengan demikian, dapat disimpulkan bahwa penerapan metode AHP-VIKOR dalam penentuan tingkat kesulitan pada *game* Avoid Najasa berhasil mencapai tujuan penelitian, yaitu menciptakan sistem permainan yang adaptif, seimbang, dan mampu meningkatkan pengalaman pengguna.

5.2 Saran

Dari riset yang telah dijalankan, ada saran-saran yang bisa diberikan bagi pengembangan berikutnya.

1. Penambahan variasi *gameplay* seperti mekanisme level yang lebih kompleks, penambahan fitur *power-up* (misalnya peningkat kecepatan, pelindung, atau magnet koin), serta variasi rintangan yang lebih beragam untuk meningkatkan daya tarik dan pengalaman bermain.

2. Pengembangan platform *game* ke perangkat *mobile* seperti Android atau iOS agar dapat menjangkau lebih banyak pengguna serta meningkatkan fleksibilitas dalam bermain.
3. Penambahan jumlah responden pada tahap pengujian, serta variasi karakteristik pemain (misalnya usia atau pengalaman bermain *game*), sehingga hasil evaluasi sistem menjadi lebih representatif.
4. Pengembangan kriteria performa pemain yang lebih kompleks atau dinamis, misalnya dengan mempertimbangkan pola bermain pemain secara berkelanjutan, sehingga sistem adaptif dapat bekerja dengan lebih akurat.
5. Penggunaan metode pengambilan keputusan lain sebagai pembanding, seperti TOPSIS, MOORA, atau metode berbasis kecerdasan buatan (*machine learning*), untuk mengevaluasi efektivitas metode AHP-VIKOR dalam konteks penentuan tingkat kesulitan *game*.
6. Peningkatan aspek visual, animasi, dan antarmuka pengguna agar *game* menjadi lebih menarik secara estetika tanpa mengurangi fokus utama pada sistem adaptif.

DAFTAR PUSTAKA

- Ardianto, F. (2024, October 1). *Pemilihan tingkat kesulitan pada game Life's Troubles menggunakan metode COPRAS (Complex Proportional Assessment) berdasarkan kemampuan pemain*. Etheses of Maulana Malik Ibrahim State Islamic University. <http://etheses.uin-malang.ac.id/69844/>
- Arfiningtyas, U., Ambarwati, E., & Saktini, F. (2017). PENGARUH BERMAIN VIDEO GAME TIPE ENDLESS RUNNING TERHADAP WAKTU REAKSI. *Jurnal Kedokteran Diponegoro (Diponegoro Medical Journal)*, 6(2), 1320–1330. <https://doi.org/10.14710/dmj.v6i2.18645>
- Armonitha, S., & Rahmansyah, N. (2024). Implementation of the VIKOR method and analytical hierarchy process (AHP) in prioritizing the recipients of uninhabitable house assistance funds (RTLH)". *Indonesian Journal of Computer Science*, 13(1). <https://doi.org/10.33022/ijcs.v13i1.3665>
- 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>
- Brooke, J. (1996). SUS: A “quick and dirty” usability scale. In *Usability Evaluation In Industry* (pp. 207–212). CRC Press. <https://doi.org/10.1201/9781498710411-35>
- Csikszentmihalyi, M. (2009). *Flow: the psychology of optimal experience*. Harper Collins.
- Fahrullazi, M., Riwinoto, R., & Hutapea, S. E. (2019). Implementasi Penerapan MDA pada Game Endless Runner 2D CAVE RACER Berbasis Android. *JOURNAL OF APPLIED MULTIMEDIA AND NETWORKING*, 3(2), 19–29. <https://doi.org/10.30871/jamn.v3i2.1636>
- Fencott, C., Clay, J., Lockyer, M., & Massey, P. (2012). *Game invaders: the theory and understanding of computer games*. John Wiley & Sons.
- Forman, E., & Peniwati, K. (1998). Aggregating individual judgments and priorities with the analytic hierarchy process. *European Journal of Operational Research*, 108(1), 165–169. [https://doi.org/10.1016/s0377-2217\(97\)00244-0](https://doi.org/10.1016/s0377-2217(97)00244-0)

- Hartanti, A. R. (2025, June 16). *Penentuan skenario pada game 'zakat rush' dengan metode vikor berdasarkan player performance*. Etheses of Maulana Malik Ibrahim State Islamic University. <http://etheses.uin-malang.ac.id/78387/>
- Haunan, F. A. (2025, December 23). *Lingkungan game adaptif berdasarkan suhu menggunakan TOPSIS dengan optimisasi bobot berbasis Particle-Swarm Optimization*. Etheses of Maulana Malik Ibrahim State Islamic University. <http://etheses.uin-malang.ac.id/82968/>
- Hidayat, T. N. (2024, June 27). *Anjuran pemilihan sepatu untuk karakter yang fleksibel pada game Endless Runner Lari Nusantara menggunakan metode VIKOR berfundamen gapaian pemain*. Etheses of Maulana Malik Ibrahim State Islamic University. <http://etheses.uin-malang.ac.id/65118/>
- Hwang, C.-L., & Yoon, K. (1981). *Multiple attribute decision making*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-48318-9>
- Ince, M. (2021). BiLSTM and dynamic fuzzy AHP-GA method for procedural game level generation. *Neural Computing and Applications*, 33(15), 9761–9773. <https://doi.org/10.1007/s00521-021-06180-7>
- Juul, J. (2011). *Half-real: Video games between real rules and fictional worlds*. MIT Press.
- Khuluq, A. A., Nasrulloh, N., & Farabi, M. R. F. A. (2024). Keseimbangan antara ujian dan kemampuan manusia: telaah tafsir Surat Al-Baqarah ayat 286 dalam Tafsir Al-Mishbah. *Journal of International Multidisciplinary Research*, 2(11), 16–25. <https://doi.org/10.62504/jimr954>
- Karavidas, L., Apostolidis, H., & Tsiatsos, T. (2022). Usability Evaluation of an Adaptive Serious Game Prototype Based on Affective Feedback. *Information*, 13(9), 425. <https://doi.org/10.3390/info13090425>
- Keeney, R. L., & Raiffa, H. (1993). *Decisions with multiple objectives: Preferences and value trade-offs*. Cambridge University Press.
- Kostkova, K., Ben Itzhak, N., Stijnen, L., Ortibus, E., & Jansen, B. (2023). Dynamic difficulty adjustment in serious games for cerebral visual impairment. *2023 IEEE 11th International Conference on Serious Games and Applications for Health (SeGAH)*, 1–8. <https://doi.org/10.1109/segah57547.2023.10253769>

- 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>
- Mardani, A., Jusoh, A., MD Nor, K., Khalifah, Z., Zakwan, N., & Valipour, A. (2015). Multiple criteria decision-making techniques and their applications – a review of the literature from 2000 to 2014. *Economic Research-Ekonomska Istraživanja*, 28(1), 516–571. <https://doi.org/10.1080/1331677x.2015.1075139>
- Mishra, H., Kumar, R., Mishra, K. K., Jaiswal, A., Kumar, R. R., & Muduli, D. (2025). An AI-VR integrated car racing simulation with procedural content generation for immersive and adaptive gameplay. *2025 International Conference on Innovations in Intelligent Systems: Advancements in Computing, Communication, and Cybersecurity (ISAC3)*, 1–6. <https://doi.org/10.1109/isac364032.2025.11156634>
- Muslimah.or.id. (2014, September 8). *Sebaik-baik Manusia adalah yang Bermanfaat*. Muslimah.or.Id. <https://muslimah.or.id/6435-pribadi-yang-bermanfaat.html>
- Nugroho, F., Basid, P. M. N. S. A., Bahtiar, F. S., & Buditjahjanto, I. G. P. A. (2023). Dynamic Difficulty Adjustment of Serious-Game Based on Synthetic Fog using Activity Theory Model. *International Journal of Advanced Computer Science and Applications*, 14(6). <https://doi.org/10.14569/ijacsa.2023.0140660>
- Nugroho, F. (2025, July 31). *Adaptive maze-based Islamic educational games using MOORA method*. Repository of Maulana Malik Ibrahim State Islamic University of Malang. <https://repository.uin-malang.ac.id/24223/>
- Omari, K., Chajri, Y., & Oukhatar, A. (2025). SERIOUS GAME EVALUATION SYSTEM BASED ON FUZZY AHP. *International Journal of Computer Applications*, 186(75), 32–42. <https://doi.org/10.5120/ijca2025924641>
- Opricovic, S., & Tzeng, G.-H. (2004). Compromise solution by MCDM methods: A comparative analysis of VIKOR and TOPSIS. *European Journal of Operational Research*, 156(2), 445–455. [https://doi.org/10.1016/s0377-2217\(03\)00020-1](https://doi.org/10.1016/s0377-2217(03)00020-1)

- Prakasa, A. B. (2024, March 21). *Saran virtual untuk pemilihan karakter terbaik pada game awas tersesat menggunakan metode oreste berbasis keterampilan pemain*. Etheses of Maulana Malik Ibrahim State Islamic University. <http://etheses.uin-malang.ac.id/65124/>
- Saaty, T. L. (2008). Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, 1(1), 83. <https://doi.org/10.1504/ijssci.2008.017590>
- Schell, J. (2008). *The Art of Game Design: A book of lenses*. CRC Press.
- Siamakmanesh, N., Ganji, A., Abdoos, M., & Vahidi-Asl, M. (2025). Creating a Foundation for Dynamic Difficulty Adjustment within PCG of games using Imitation Learning. *Journal of Innovations in Computer Science and Engineering (JICSE)*, 2(Special Issues 2), 49–55. <https://doi.org/10.48308/jicse.2025.239782.1071>
- Sukma, A. P., Yusuf, R., & Dai, R. H. (2023). Analisis pengukuran usability sistem informasi manajemen baznas (simba) menggunakan metode system usability scale (sus). *Diffusion: Journal of Systems and Information Technology*, 3(2), 224–231. <https://doi.org/10.37031/diffusion.v3i2.21342>
- Tekinbas, K. S., & Zimmerman, E. (2003). *Rules of play: game design fundamentals*. MIT Press.
- Triantaphyllou, E. (2000). *Multi-criteria decision making methods: A comparative study*. Springer US. <https://doi.org/10.1007/978-1-4757-3157-6>
- Ubaidillah, F. R. (2025, June 20). *Implementasi metode additive ratio assessment (ARAS) untuk pemilihan equipment dalam game edukasi lingkungan “River Run.”* Etheses of Maulana Malik Ibrahim State Islamic University. <http://etheses.uin-malang.ac.id/78493/>