

IMPLEMENTASI METODE RANDOM FOREST PADA KLASIFIKASI GAYA KEPEMIMPINAN DALAM SERIOUS GAME THE ARCHITECT: REBIRTH OF SERAYA

Damar Nabil Buqori, Paulus Harsadi

Teknik Informatika, Universitas Tiga Serangkai Surakarta
Jl. K.H. Samanhudi No. 84-86 57142 Laweyan Jawa Tengah
damarnabil4@gmail.com

ABSTRAK

Permainan video bergenre simulasi tata kota memiliki potensi edukasi manajerial yang tinggi, namun seringkali memberikan umpan balik yang kaku akibat ketiadaan sistem pemodelan pemain cerdas yang mampu mengenali ideologi pengambilan keputusan penggunanya. Penelitian ini bertujuan untuk mengimplementasikan kecerdasan buatan guna mengklasifikasikan gaya kepemimpinan pemain secara waktu nyata (*real-time*) dalam *serious game* "The Architect: Rebirth of Seraya". Pengembangan sistem berpedoman pada metodologi *Game Development Life Cycle* (GDLC) menggunakan perangkat lunak Godot Engine 4.x berbasis C#. Model pemodelan dibangun menggunakan algoritma *Random Forest* yang mengekstraksi data telemetri kota secara berkala ke dalam lima pohon keputusan independen, kemudian diproses melalui mekanisme *Weighted Majority Voting* dan *Penalty Logic*. Evaluasi keandalan diukur menggunakan *Confusion Matrix* untuk validasi komputasi klasifikasi, serta *Game Experience Questionnaire* (GEQ) kepada 50 responden untuk analisis pengalaman pengguna. Hasil pengujian menunjukkan bahwa algoritma *Random Forest* berhasil mencetak tingkat akurasi komputasi sebesar 93,3%, sementara evaluasi pengalaman pengguna memperoleh nilai rata-rata 4,14 dari skala 5,00 yang tergolong "Baik". Kesimpulannya, pemrosesan algoritma *Random Forest* terbukti sangat presisi dan berhasil mentransformasi simulasi tata kelola perkotaan menjadi pengalaman edukasi interaktif yang adaptif.

Kata kunci : *Serious game, Pemodelan pemain, Random forest, Telemetri, Godot.*

1. PENDAHULUAN

Permainan video bergenre strategi dan simulasi pembangunan kota (*city building*) telah terbukti efektif sebagai *serious game* edukatif untuk memodelkan sistem pengambilan keputusan strategis dan kebijakan publik. Melalui pengalaman interaktif, pemain dapat belajar menyeimbangkan variabel tata kelola wilayah yang kompleks, seperti memprioritaskan pertumbuhan ekonomi versus menjaga kelestarian lingkungan. Namun, tantangan fundamental sekaligus permasalahan utama dalam pengembangan *serious game* saat ini adalah ketiadaan sistem yang mampu memahami karakteristik unik dan pola pengambilan keputusan pemain secara cerdas [1]. Selama ini, banyak simulasi hanya menilai performa pemain berdasarkan metrik numerik statis tanpa mendeteksi perilaku atau ideologi di balik keputusan tersebut [2]. Ketiadaan pemodelan pemain (*player modeling*) ini menyebabkan umpan balik permainan menjadi generik, kaku, dan tidak mampu membedakan apakah seorang pemain cenderung menerapkan gaya kepemimpinan "Kapitalis", "Sosial", "Humanis", atau "Otoriter".

Sebagai rencana penyelesaian atas permasalahan tersebut, penelitian ini mengimplementasikan algoritma *Random Forest* sebagai mesin analitik cerdas (*intelligence engine*) untuk melakukan klasifikasi perilaku pemain secara waktu nyata (*real-time*) di dalam *serious game* "The Architect: Rebirth of Seraya". Pendekatan *Ensemble Learning* pada *Random Forest* dipilih karena keunggulannya dalam menangani kompleksitas data simulasi yang memiliki

tingkat variabilitas tinggi dan saling tumpang tindih [4]. Algoritma ini dirancang untuk bekerja dengan mengekstraksi data telemetri pemain secara berkala yang meliputi indikator kas kota, tingkat pendidikan, indeks kesehatan, kondisi lingkungan, ketahanan pangan, angka kemiskinan, dan tingkat kepuasan warga. Data multivariabel tersebut kemudian didistribusikan untuk diproses secara asinkron melalui lima pohon keputusan (*decision trees*) independen, dan diakhiri dengan mekanisme pemungutan suara mayoritas (*majority voting*) untuk menghasilkan label gaya kepemimpinan akhir yang stabil dan presisi [6].

Berdasarkan pemaparan tersebut, penelitian ini bertujuan untuk merancang perangkat lunak simulasi tata kota yang adaptif menggunakan perangkat lunak Godot Engine 4.x dengan bahasa pemrograman C# [12], serta memvalidasi tingkat akurasi komputasi model *Random Forest* dalam mengenali profil pengguna. Melalui tujuan tersebut, integrasi klasifikasi kecerdasan buatan ini diharapkan mampu menciptakan ekosistem permainan yang lebih responsif dan personal, sehingga pesan edukasi terkait manajerial tata kota dapat tersampaikan secara tepat sasaran sesuai dengan ideologi gaya bermain masing-masing pengguna [7].

2. TINJAUAN PUSTAKA

2.1. Serious Game dan Afordansi Genre City Building

Serious game dirancang dengan tujuan utama melampaui sekadar hiburan interaktif, yakni untuk keperluan edukasi, simulasi sistem yang kompleks,

dan eksperimen kebijakan berisiko tinggi di dalam lingkungan virtual yang aman. Penerapan genre pembangunan kota (*city building*) telah divalidasi memiliki efektivitas tinggi dalam melatih kemampuan berpikir sistemis (*systems thinking*) melalui pemodelan konsekuensi spasial dan kausalitas antara keputusan manajerial dengan tingkat kesejahteraan virtual dari suatu populasi [13].

2.2. Telemetri dan Pemodelan Pemain (Player Modeling)

Pemodelan pemain merupakan aspek esensial untuk menciptakan sistem yang adaptif. Proses ini bergantung pada data metrik dinamis atau telemetri permainan (*gameplay telemetry*) yang direkam secara waktu nyata [2]. Ekstraksi kumpulan variabel metrik secara kontinu seperti fluktuasi kas, indeks polusi, dan kepuasan warga terbukti secara empiris mampu mengidentifikasi pola keputusan prediktif serta membentuk profil komputasional dari setiap pengguna secara presisi, tanpa harus mengganggu imersi bermain [1].

2.3. Algoritma Random Forest dan Majority Voting

Random Forest merupakan algoritma *Machine Learning* berjenis *ensemble learning* yang beroperasi dengan membangun sekumpulan pohon keputusan (*decision trees*) independen guna menghasilkan prediksi komputasi yang jauh lebih akurat, tangguh, dan stabil dibandingkan model klasifikasi tunggal [6]. Pendekatan ini efektif menangani kompleksitas variabel yang saling tumpang tindih dalam simulasi dengan cara mendelegasikan parameter spesifik ke pohon evaluasi yang berbeda [4]. Setelah tahap inferensi selesai, algoritma ini melakukan agregasi pemungutan suara mayoritas (*majority voting*) menggunakan pencarian nilai modus untuk memastikan klasifikasi prediksi terhindar dari anomali data dan lebih mudah diinterpretasikan [5].

2.4. Godot Engine 4.x dan Pemrograman C#

Pengembangan perangkat lunak pada penelitian ini disandarkan pada Godot Engine 4.x, sebuah kerangka kerja lintas platform dengan arsitektur *rendering Vulkan API* yang tangguh dalam memproses ribuan objek 2D isometrik secara simultan. Integrasi bahasa pemrograman C# (C-Sharp) digunakan sebagai inti dari komputasi karena terbukti memiliki keunggulan performa yang sangat signifikan dalam hal kecepatan dan stabilitas saat mengeksekusi operasi perhitungan matematis tingkat tinggi secara asinkron, dibandingkan dengan penggunaan bahasa skrip bawaan (GDScript) [12].

2.5. Penelitian Terdahulu

Sebagai landasan studi, penelitian ini mengacu pada beberapa literatur terdahulu yang telah mengeksplorasi implementasi *Machine Learning* dan *serious game*. Penelitian oleh Habibi et al. (2023)

berhasil mengimplementasikan *Random Forest* untuk memodelkan kepribadian pemain, namun pendekatan tersebut masih berfokus pada pengelompokan atribut psikologis umum dan bukan pada analisis keputusan manajerial [1]. Secara paralel, studi oleh Tan et al. (2024) memanfaatkan algoritma *Random Forest* untuk memprediksi metrik performa (*player performance*) berdasarkan data historis permainan [2]. Meskipun model prediktif yang dihasilkan memiliki akurasi yang baik, sistem tersebut hanya menilai angka statistik statis dan tidak dirancang untuk mengekstraksi ideologi gaya bermain yang melatarbelakangi performa tersebut. Dalam konteks simulasi manajerial, Ghodsvali et al. (2022) membuktikan tingginya efektivitas *serious game* sebagai medium simulasi pengambilan keputusan kebijakan publik yang kompleks [13], akan tetapi platform tersebut belum mengintegrasikan mekanisme kecerdasan buatan otonom untuk memberikan umpan balik personal kepada penggunanya.

Berdasarkan komparasi literatur terdahulu tersebut, dapat diidentifikasi adanya celah penelitian (*research gap*) terkait belum adanya sistem klasifikasi profil manajerial yang terintegrasi secara langsung di dalam simulasi tata kelola wilayah. Oleh karena itu, kebaruan (*novelty*) dari penelitian ini terletak pada transformasi penggunaan *Random Forest* yang tidak sekadar menilai performa akhir pemain, melainkan mendeteksi dan mengklasifikasikan ideologi gaya kepemimpinan ("Kapitalis", "Sosial", atau "Otoriter") secara waktu nyata (*real-time*). Selain itu, inovasi arsitektur komputasi pada penelitian ini diperkuat melalui penyuntikan mekanisme modifikasi bobot (*Penalty Logic*) dan agregasi *Weighted Majority Voting* untuk mencegah bias resolusi prediksi, sebuah arsitektur hibrida yang belum dieksplorasi pada model *player modeling* di literatur-literatur sebelumnya.

3. METODE PENELITIAN

Penelitian ini mengadopsi kerangka kerja *Game Development Life Cycle* (GDLC) sebagai landasan metodologis yang menjamin pengembangan perangkat lunak berlangsung secara terstruktur dan iteratif [14]. Tahapan pengembangan merangkum fase inisialisasi konsep, pra-produksi, produksi (implementasi Godot Engine berbasis bahasa pemrograman C#), pengujian, hingga rilis [12]. Fokus utama pada fase rekayasa arsitektur difokuskan pada perancangan modul *Leadership Manager*. Modul ini merupakan wujud integrasi kecerdasan buatan berbasis *Random Forest* yang dirancang untuk beroperasi secara asinkron di latar belakang (*background process*), sehingga pemrosesan data analitik tidak mendegradasi performa rendering grafis maupun mengganggu interaktivitas simulasi tata kota.

3.1 Tahapan Pengembangan Sistem

Pengembangan sistem dibagi menjadi lima tahapan utama:

- inisialisasi: Penentuan genre *City Building* dan target platform PC Desktop (Windows 64-bit).
- Pra-Produksi: Perancangan *Game Design Document* (GDD), pemetaan alur FSM, dan perumusan matematis algoritma EDDA.
- Produksi: Implementasi kode menggunakan bahasa C#.NET 8 pada engine Godot 4.x.
- Pengujian: Validasi *Black Box Testing* terhadap seluruh skenario aksi pemain.
- Rilis: Kompilasi akhir aplikasi *standalone*.

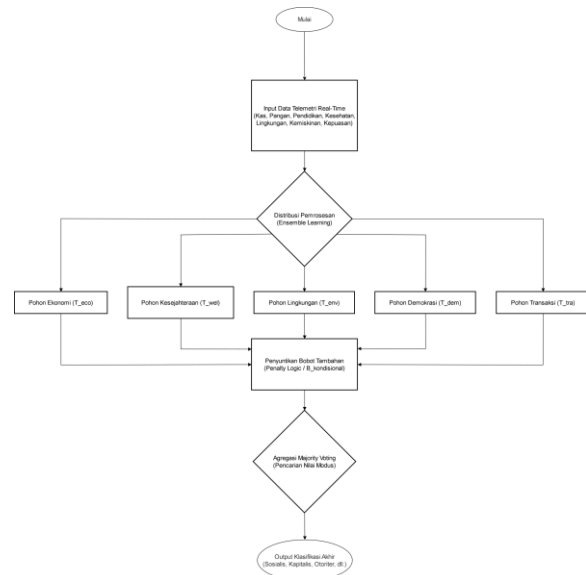
3.2 Arsitektur Sistem Klasifikasi Random Forest

Sistem klasifikasi dibangun melalui alur pemrosesan data (*data pipeline*) yang berjalan secara kontinu. Arsitektur algoritma ini diorkestrasikan agar mampu memecah kompleksitas data melalui pendekatan *Ensemble Learning*:

- Input Data Telemetri: Sistem secara otonom mengekstraksi tujuh variabel kota secara real-time, yang meliputi Kas, Pangan, Pendidikan, Kesehatan, Kualitas Lingkungan, Angka Kemiskinan, dan Tingkat Kepuasan Warga.
- Distribusi Pemrosesan (*Ensemble Learning*): Rangkaian data fitur tersebut didistribusikan secara paralel ke dalam lima Pohon Keputusan (*Decision Trees*) yang terisolasi secara logis, yaitu: Pohon Ekonomi (T_{eco}), Pohon Kesejahteraan (T_{wel}), Pohon Lingkungan (T_{env}), Pohon Demokrasi (T_{dem}), Pohon Transaksi (T_{tra}).
- Penyuntikan Bobot Tambahan (*Penalty Logic*): Modul melakukan evaluasi terhadap kondisi ekstrem untuk menyuntikkan nilai penalti atau bonus (*reward*) kondisional ($B_{kondisional}$) pada label gaya kepemimpinan tertentu.
- Agregasi *Majority Voting*: Kalkulasi komputasional untuk mencari nilai modus (kemunculan tertinggi) dari seluruh label yang dihasilkan oleh pohon keputusan.
- Output Klasifikasi Akhir: Penetapan predikat gaya kepemimpinan pemain (sebagai contoh: "Sosial", "Kapitalis", atau "Otoriter") secara definitif.

Gambar 1 merepresentasikan topologi alur data (*data pipeline*) dari modul Leadership Manager yang ditengahi oleh algoritma Random Forest. Secara struktural, arsitektur ini beroperasi melalui mekanisme bottom-up yang diawali dari lapisan ekstraksi hingga mencapai lapisan inferensi akhir.

Pada tahap awal, sistem melakukan pengikatan (*binding*) terhadap tujuh variabel metrik kota utama. Vektor fitur mentah ini kemudian dipecah dan didistribusikan secara paralel menggunakan pendekatan *Ensemble Learning* menuju lima lapisan *Decision Trees* ($T_{eco}, T_{wel}, T_{env}, T_{dem}, T_{tra}$). Keberadaan kelima pohon yang terisolasi secara logis ini bertujuan untuk mengevaluasi dimensi keputusan manajerial pemain dari sudut pandang yang berbeda, sehingga mencegah terjadinya bias analisis pada satu variabel tunggal.



Gambar 1. Arsitektur pemrosesan modul random forest

Setelah setiap pohon memproduksi tebakan dasar (label sementara), arus data bergerak menuju lapisan *Penalty Logic*. Pada titik persimpangan ini, mesin algoritma mengevaluasi ketiadaan atau keberadaan metrik ekstrem. Apabila terdeteksi anomali (misalnya krisis multidimensi yang disengaja oleh pemain), sistem akan menyuntikkan bobot penalti atau bonus ($B_{kondisional}$) guna memodifikasi bobot murni dari setiap label.

Pada tahap final, akumulasi nilai dari seluruh cabang dieksekusi oleh mesin agregator *Majority Voting*. Proses komputasi ini mencari nilai modus atau probabilitas suara tertinggi, yang kemudian ditetapkan sebagai deklarasi definitif (*Output Klasifikasi Akhir*) untuk memetakan gaya kepemimpinan ideologis pengguna selama sesi simulasi berlangsung. Pendekatan arsitektur terdesentralisasi ini menjamin operasi algoritmik tetap presisi tanpa membebani memori utama (RAM) secara berlebihan.

3.3 Ekstraksi Vektor Fitur dan Penilaian Pohon Keputusan

Di dalam operasionalnya, modul Leadership Manager bertindak sebagai penyedia data waktu-nyata (*real-time data provider*) yang menyintesis tujuh variabel makroekonomi kompleks menjadi sebuah vektor fitur (*input features*). Vektor numerik tunggal ini kemudian diumpungkan secara simultan ke dalam lima *Decision Trees*

independent ($T_{eco}, T_{wel}, T_{env}, T_{dem}, T_{tra}$). Setiap pohon dilengkapi dengan aturan ambang batas logika (*rule-based threshold*) spesifik yang dikalibrasi untuk membedakan dimensi tata kelola tertentu. Apabila sebuah pohon keputusan berhasil mengklasifikasikan perilaku pemain ke dalam sebuah label kepemimpinan, pohon tersebut akan mendonasikan nilai dasar (T_i) sebesar 1 poin terhadap label yang bersangkutan.

3.4 Formulasi Matematis *Weighted Majority Voting* dan *Penalty Logic*

Guna memitigasi potensi bias klasifikasi dan menghindari kegagalan resolusi saat terjadi anomali suara imbang (tie-breaker), sistem tidak murni bergantung pada pemungutan suara biasa. Penelitian ini mengimplementasikan modifikasi logis berupa penyuntikan bobot kondisional ($B_{kondisional}$). Sebagai ilustrasi teknis, apabila pemain membiarkan metrik kepuasan warga hancur hingga menyentuh margin ekstrem di bawah 15%, sistem secara otomatis akan menjatuhkan poin penalti pada kalkulasi akhir untuk label yang bernuansa pro-rakyat. Akumulasi total nilai voting (V). uuntuk suatu label spesifik dirumuskan melalui Persamaan 1:

$$V_{label} = \sum_{i=1}^n T_i + B_{kondisional} \quad (1)$$

Setelah total bobot dari seluruh probabilitas label terkumpul, algoritma *Random Forest* mengeksekusi tahapan agregasi akhir menggunakan *Weighted Majority Voting*. Algoritma ini menjalankan fungsi pencarian nilai modus untuk mengidentifikasi dan menetapkan label gaya kepemimpinan dengan akumulasi bobot matematis tertinggi sebagai prediksi final (*final prediction*), sebagaimana direpresentasikan pada Persamaan 2:

$$Label_{final} = \text{mode}\{V_{label1}, V_{label2}, \dots, V_{labelN}\} \quad (2)$$

3.5 Instrumen dan Prosedur Pengujian Sistem

Evaluasi komprehensif terhadap keandalan sistem dilaksanakan melalui dua pendekatan instrumen yang mengukur objektivitas teknis dan subjektivitas pengguna. Pertama, pengujian akurasi algoritmik dari model *Random Forest* divalidasi menggunakan metode *Confusion Matrix* (Matriks Kebingungan) [8]. Pengujian ini krusial untuk mengukur tingkat presisi komputasi klasifikasi kecerdasan buatan ketika dikomparasikan dengan kebenaran aktual (*ground truth*). Kedua, evaluasi empiris terkait interaksi pengguna dilakukan menggunakan instrumen *Game Experience Questionnaire* (GEQ) [15]. Kuesioner didistribusikan kepada 50 responden (batas usia minimum 16 tahun) yang telah memenuhi kriteria penyelesaian minimum satu sesi permainan. Responden memberikan penilaian terhadap lima dimensi psikometrik Kompetensi, Imersi, *Flow*, Ketegangan, dan Tantangan menggunakan Skala Likert 1 hingga 5. Pendekatan ini secara komprehensif membuktikan sejauh mana efektivitas intervensi sistem kecerdasan buatan mampu mempertahankan keseimbangan pengalaman bermain yang adaptif [7].

4. HASIL DAN PEMBAHASAN

4.1 Hasil Implementasi Sistem Klasifikasi

Modul kecerdasan buatan berbasis algoritma *Random Forest* diimplementasikan ke dalam Godot Engine 4.x menggunakan bahasa pemrograman C#. Arsitektur komputasi ini dirancang untuk beroperasi secara asinkron di latar belakang (*background process*), sehingga pemrosesan analitik tidak

membebani siklus perenderan grafis (*frame rate*) utama dari permainan. Ekstraksi data telemetri dieksekusi secara periodik menggunakan mekanisme pewaktuan internal dengan interval 30 detik untuk mengevaluasi fluktuasi tujuh variabel makroekonomi secara real-time. Pada tahap konklusi sesi simulasi, sistem mengeksekusi fungsi agregasi untuk mengalkulasi nilai modus melalui mekanisme *Majority Voting* dari kelima cabang *Decision Tree* (Ekonomi, Kesejahteraan, Lingkungan, Demokrasi, dan Transaksi). Agregasi analitik ini berfungsi untuk menetapkan dan mendeklarasikan profil gaya kepemimpinan pengguna secara komprehensif.

Sebagai bukti visual dari implementasi arsitektur komputasi tersebut, antarmuka utama permainan beserta panel pemantauan (*monitoring panel*) algoritma *Random Forest* dapat dilihat pada Gambar 2.



Gambar 2. Antarmuka permainan dan panel

pemantau algoritma *random forest* Berdasarkan Gambar 2, modul *Leadership Manager* menyediakan panel pemantau internal yang memungkinkan observasi terhadap pemrosesan asinkron kecerdasan buatan secara langsung. Panel tersebut menampilkan status ekstraksi tujuh variabel telemetri secara waktu nyata, seperti indikator Kas (Uang), Tingkat Kepuasan (Puas), Pendidikan (Edu), Kesehatan (Sehat), Pangan, Lingkungan (Lingk), dan Kemiskinan. Lebih lanjut, pada bagian "Voting 5 Pohon", sistem secara transparan menampilkan hasil tebakan logis dari masing-masing pohon keputusan (sebagai contoh pada tangkapan layar: cabang ekonomi dan demokrasi menghasilkan label "Otoriter", sedangkan cabang kesejahteraan menghasilkan "Populis"). Prediksi dari kelima pohon independen tersebut kemudian diakumulasikan ke dalam "Papan Skor" melalui mekanisme *Majority Voting* untuk mendeklarasikan gaya kepemimpinan dominan pengguna pada sesi tersebut.

4.2 Hasil Pengujian Fungsionalitas (Black Box Testing)

Pengujian fungsionalitas dilakukan menggunakan metode *Black Box Testing* yang berfokus pada pengamatan fungsionalitas *Input-Output* dari mekanika interaksi dasar permainan tanpa melihat struktur kode di baliknya. Pengujian ini bertujuan untuk memastikan sistem manajemen

sumber daya dan antarmuka pemain berjalan sesuai dengan batasan aturan (*ruleset*) yang ditetapkan. Hasil seluruh skenario pengujian tercantum pada Tabel 1.

Tabel 1. Hasil pengujian fungsionalitas (black box testing)

ID	Skenario pengujian	Hasil Aktual	Hasil yang Diharapkan	Status
BB-01	Pemain meletakkan bangunan saat saldo Kas Kota mencukupi.	Saldo berkurang, bangunan terinstansiasi dengan presisi Y-Sorting, dan SFX konstruksi berbunyi.	Sistem memotong saldo pemain, memunculkan sprite bangunan di atas TileMap, dan memainkan SFX konstruksi.	Valid
BB-02	Pemain mencoba meletakkan bangunan saat dana kurang dari harga bangunan.	Bangunan batal dibuat, muncul teks peringatan UI warna merah, SFX Error berbunyi.	Sistem menolak aksi konstruksi dan memunculkan notifikasi “Dana Tidak Cukup”.	Valid
BB-03	Pemain membangun Klinik, dana cukup, namun level Edukasi kota di bawah syarat (<70%).	Konstruksi dibatalkan, panel UI memberikan umpan balik alasan kegagalan spesifik.	Sistem memblokir peletakan bangunan dan menampilkan notifikasi spesifik “Syarat Edukasi Tidak Terpenuhi”.	Valid
BB-04	Pemain menghapus pabrik yang sedang aktif mempekerjakan warga.	Bangunan terhapus, saldo bertambah setengah harga, angka kemiskinan instan melonjak.	Bangunan hilang (QueueFree), pemain mendapat dana refund 50%, pekerja dikembalikan ke status pengangguran.	Valid

4.3 Hasil Pengujian Presisi Algoritma (Confusion Matrix)

Validasi keandalan komputasi algoritma diukur menggunakan parameter Confusion Matrix guna memverifikasi tingkat presisi kecerdasan buatan terhadap kebenaran aktual (ground truth). Pengujian dilakukan melalui simulasi 30 skenario tata kelola

yang direayasa ke dalam tiga polaritas gaya bermain ekstrem: Sosialis (berorientasi pada kesejahteraan warga), Kapitalis (berorientasi pada akumulasi finansial), dan Otoriter (pengabaian terhadap kepuasan publik). Hasil komputasi prediksi sistem dalam mengklasifikasikan ketiga skenario tersebut direpresentasikan pada Tabel 2.

Tabel 2. Hasil pengujian confusion matrix Random Forest

Kelas Aktual (Ground Truth)	Prediksi Sosialis	Prediksi Kapitalis	Prediksi Otoriter	Total Skenario
Skenario Sosialis	9 (TP)	1 (FN)	0 (FN)	10
Skenario Kapitalis	0 (FP)	10 (TP)	0 (FN)	10
Skenario Otoriter	1 (FP)	0 (FP)	9 (TP)	10

Berdasarkan matriks pengujian pada Tabel 2, algoritma Random Forest yang diperkuat dengan mekanisme Weighted Majority Voting berhasil memprediksi 28 dari 30 skenario dengan tepat, menghasilkan tingkat akurasi komputasi sebesar 93,3%. Rendahnya rasio False Positive (FP) dan False Negative (FN) menjadi indikator empiris bahwa logika penalti kondisional yang disuntikkan ke dalam inti sistem sangat efektif dalam mengeliminasi anomali pelabelan, khususnya saat algoritma menghadapi masalah suara imbang (tie-breaker) antar-pohon keputusan.

4.4 Hasil Pengujian Pengalaman Pengguna (GEQ)

Pengukuran efektivitas intervensi adaptif dan penerimaan pengguna terhadap sistem dievaluasi secara kuantitatif menggunakan instrumen Game Experience Questionnaire (GEQ). Kuesioner didistribusikan kepada 50 responden dengan demografi usia dominan pada rentang 19–22 tahun (46%) dan di atas 23 tahun (50%). Seluruh instrumen dinyatakan memenuhi standar validitas dan reliabilitas dengan perolehan nilai Cronbach's Alpha > 0,70. Data

ordinal berskala Likert 1–5 diekstraksi ke dalam lima dimensi evaluasi psikometrik, sebagaimana ditunjukkan pada Tabel 3.

Tabel 3. Rekapitulasi skor rata-rata dimensi GEQ

Dimensi	Mean	Cronbach's Alpha	Interpretasi
Immersion	4,120	0,8580	Baik
Presence	4,147	0,7248	Baik
Flow	4,044	0,8799	Baik
Absorption	3,813	0,8908	Baik
Kausalitas	4,156	0,9287	Baik
Mekanik	4,180	0,8539	Baik
Evaluasi	4,120	0,9023	Baik
Rata-Rata Keseluruhan	4,087	> 0,70	Baik

Hasil analisis statistik deskriptif pada Tabel 3 memaparkan bahwa perangkat lunak simulasi tata kota ini memperoleh skor komposit rata-rata sebesar 4,08 (Kategori "Baik"). Dominasi perolehan nilai pada dimensi Tantangan (4,18) dan Imersi (4,12) memberikan afirmasi bahwa arsitektur gim mampu mempertahankan tingkat kesulitan yang dinamis dan adaptif. Secara lebih spesifik, penjabaran performa

modul kecerdasan buatan menunjukkan bahwa algoritma Random Forest secara independen memperoleh nilai evaluasi sebesar 4,14 (sementara modul EDDA dan FSM secara berurutan mencetak 4,07 dan 4,18). Tingginya nilai validasi pada sub-modul klasifikasi ini menegaskan bahwa profil kepemimpinan akhir yang ditetapkan oleh mesin dirasakan sangat akurat, natural, dan relevan oleh pengguna. Konklusinya, tujuan instruksional terkait kausalitas kebijakan publik berhasil ditransmisikan secara optimal tanpa mendegradasi tingkat retensi keterlibatan pemain (player engagement).

5. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian, implementasi algoritma *Random Forest* sebagai sistem pemodelan pemain di dalam *serious game* "The Architect: Rebirth of Seraya" telah berhasil mengklasifikasikan gaya kepemimpinan secara presisi dan otonom. Pendekatan *Ensemble Learning* yang mendistribusikan ekstraksi data telemetri ke dalam lima pohon keputusan independen, dikombinasikan dengan mekanisme *Weighted Majority Voting* dan *Penalty Logic*, terbukti sangat tangguh dalam mengeliminasi anomali klasifikasi serta menyelesaikan permasalahan suara imbang (*tie-breaker*). Keberhasilan arsitektur ini tervalidasi melalui pengujian akurasi komputasi sebesar 93,3% serta hasil pengujian pengalaman pengguna (GEQ) yang mencetak nilai rata-rata 4,08 (Kategori "Baik"), di mana umpan balik profil kepemimpinan dinilai sangat relevan dan natural oleh responden dengan skor spesifik 4,14. Untuk penyempurnaan sistem di masa mendatang, disarankan agar sistem klasifikasi berbasis aturan (*rule-based threshold*) ini ditransformasikan menjadi model *Machine Learning* yang dapat dilatih secara dinamis (*dynamic training*) menggunakan *dataset* riwayat pemain aktual berskala masif. Selain itu, pengembangan selanjutnya diharapkan dapat memperluas ekstraksi fitur hingga ke metrik mikro, seperti waktu respons (*response time*) pengambilan keputusan pemain, serta mengintegrasikan dampak klasifikasi *Random Forest* secara langsung ke dalam transformasi visual lingkungan kota (misalnya: memunculkan efek kabut asap pekat bagi pemain bergaya "Industrialis") guna mengoptimalkan penyampaian nilai edukasi kausalitas dan imersi simulasi.

DAFTAR PUSTAKA

- [1] R. Habibi, J. Pfau, and M. S. El-Nasr, "Modeling Player Personality Factors from In-Game Behavior and Affective Expression," in Proc. 2023 11th Int. Conf. Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW), 2023, doi: 10.48550/arXiv.2308.14224.
- [2] Y. Tan et al., "Visualization of players' game performance based on random forest and particle swarm models," *Highlights in Science, Engineering and Technology*, vol. 115, pp. 122–131, 2024, doi: 10.54097/kb9ecd48.
- [3] D. David and A. Zahra, "Player Churn Prediction in Free to Play Game Using Ensemble Learning," *Action Research Literate*, vol. 8, no. 4, pp. 543–549, Apr. 2024, doi: 10.46799/ar.v8i4.280.
- [4] Y. Li and Y. Mu, "Research and performance analysis of random forest-based feature selection algorithm in sports effectiveness evaluation," *Scientific Reports*, vol. 14, no. 26275, 2024, doi: 10.1038/s41598-024-76706-1.
- [5] M. Haddouchi and A. R. Berrado, "A Survey on Methods and Techniques for Interpreting Random Forest Models," *arXiv preprint arXiv:2407.12759*, 2024, doi: 10.48550/arXiv.2407.12759.
- [6] X. Lv et al., "Momentum prediction models of tennis match based on CatBoost regression and random forest algorithms," *Scientific Reports*, vol. 14, no. 18834, 2024, doi: 10.1038/s41598-024-69876-5.
- [7] J. Li and X. Chen, "Modeling student satisfaction in online learning using random forest," *Scientific Reports*, vol. 15, no. 23254, 2025, doi: 10.1038/s41598-025-06686-3.
- [8] B. Mutale et al., "A performance evaluation of random forest, artificial neural network, and support vector machine learning algorithms to predict spatio-temporal land use-land cover dynamics," *Frontiers in Environmental Science*, vol. 12, no. 1431645, Sep. 2024, doi: 10.3389/fenvs.2024.1431645.
- [9] A. Ahmed et al., "An optimized ensemble model with advanced feature selection for network intrusion detection," *PeerJ Computer Science*, vol. 10, no. e2472, Nov. 2024, doi: 10.7717/peerj-cs.2472.
- [10] T. Turino, R. E. Saputro, and G. Karyono, "Penerapan Model Ensemble Learning dengan Random Forest dan Multi-Layer Perceptron untuk Prediksi Gempa," *Jurnal Pendidikan dan Teknologi Indonesia*, vol. 5, no. 2, pp. 493–504, Feb. 2025, doi: 10.52436/1.jpti.667.
- [11] P. A. E. Lim and C. H. Park, "A collaborative ensemble construction method for federated random forest," *Expert Systems with Applications*, 2024, doi: 10.48550/arXiv.2407.19193.
- [12] S. Alchimowicz and M. Plechawska-Wójcik, "Comparative analysis of the implementation performance using selected scripting languages in the Godot game engine," *Journal of Computer Sciences Institute*, vol. 31, pp. 68–72, Jun. 2024, doi: 10.35784/jcsi.5428.
- [13] M. Ghodsvai, G. Z. Dane, and B. de Vries, "An online serious game for decision-making on food-water-energy nexus policy," *Sustainable Cities and Society*, vol. 87, no. 104220, Dec. 2022, doi: 10.1016/j.scs.2022.104220.
- [14] E. A. Winardy et al., "Game Development Life Cycle: A Framework for Interactive Multimedia Production," *Jurnal Teknik Informatika*, vol. 15, no. 2, pp. 1–10, Apr. 2023.
- [15] J. H. Brockmyer et al., "The development of the Game Engagement Questionnaire," *Journal of Experimental Social Psychology*, vol. 45, no. 4, pp. 624–634, 2009.