

Aplikasi Game Endless Runner 2D “Shadow Veil” Menggunakan Godot Engine Dengan Metode GDLC

Alkindy Dhiyaa Ulhaq^{1*}, Yanty Faradillah Siahaan²

^{1*,2} Program Studi Sistem Informasi, Fakultas Teknik Dan Komputer, Universitas Harapan Medan, Kota Medan, Provinsi Sumatera Utara, Indonesia.

Corresponding Email: alkindydhiyaaulhaq@gmail.com^{1*} yantyfaradillah@gmail.com²

Histori Artikel:

Dikirim 23 Agustus 2026; *Diterima dalam bentuk revisi* 06 September 2026; *Diterima* 14 September 2026; *Diterbitkan* 28 September 2026. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Indonesia Banda Aceh.

Abstrak

Teknologi seluler berkembang pesat, dan hal ini telah meningkatkan permintaan akan hiburan digital, khususnya game Android. Penelitian ini bertujuan untuk merancang dan membangun game endless runner 2D berjudul “Shadow Veil” menggunakan Godot Engine. “Shadow Veil” merupakan game endless runner dengan elemen aksi, termasuk sistem pertempuran (combat system) melawan musuh, yang dipadukan dengan tema fantasi. Dalam game ini, pemain mengendalikan karakter yang terus berlari sambil menghadapi berbagai rintangan dan musuh yang menghalangi jalan. Metode yang digunakan dalam penelitian ini adalah Game Development Life Cycle (GDLC), yang terdiri dari lima tahapan: inisiasi, pra-produksi, produksi, pengujian, dan rilis. Game ini diuji menggunakan metode Black Box Testing untuk memastikan seluruh fungsi dan fitur berfungsi dengan baik. Hasil penelitian menunjukkan bahwa game “Shadow Veil” berhasil dibangun dan berjalan dengan baik pada platform Android, dengan seluruh fitur gameplay, sistem menyerang, sistem penilaian (scoring system), serta antarmuka pengguna (user interface) yang berfungsi sebagaimana mestinya. Game ini diharapkan dapat menjadi pilihan hiburan yang menarik, sekaligus menawarkan nilai edukatif bagi pemain dengan meningkatkan refleksi, ketangkasan, dan pengambilan keputusan yang cepat saat menghadapi rintangan dan musuh.

Kata Kunci: Action Game, Black Box Testing, Endless Runner, Godot Engine, Game Development Life Cycle (GDLC).

Abstract

Mobile technology is rapidly advancing, leading to an increased demand for digital entertainment, particularly Android games. This research aims to design and build a 2D endless runner game titled “Shadow Veil” using the Godot Engine. “Shadow Veil” is an endless runner game featuring action elements, including a combat system against enemies, combined with a fantasy theme. In this game, players control a character that continuously runs while facing various obstacles and enemies that block the path. The method employed in this research is the Game Development Life Cycle (GDLC), which consists of five stages: initiation, pre-production, production, testing, and release. The game was tested using the Black Box Testing method to ensure that all functions and features operate correctly. The results indicate that the game “Shadow Veil” has been successfully developed and runs well on the Android platform, with all gameplay features, attack systems, scoring systems, and user interfaces functioning as intended. This game is expected to be an engaging entertainment option while also providing educational value to players by enhancing reflexes, agility, and quick decision-making when facing obstacles and enemies.

Keyword: Action Game, Black Box Testing, Endless Runner, Godot Engine, Game Development Life Cycle (GDLC).

1. Pendahuluan

Perkembangan industri game secara global mengalami pertumbuhan yang sangat pesat. Menurut data Newzoo (2024), pasar game dunia mencapai nilai lebih dari 184 miliar dolar AS pada tahun 2024 dan diproyeksikan terus meningkat hingga tahun 2027. Di Indonesia sendiri, industri game juga menunjukkan pertumbuhan yang signifikan seiring dengan meningkatnya smartphone dan akses internet. Hal ini mendorong munculnya banyak pengembang indie yang memanfaatkan game sebagai media hiburan sekaligus sarana pengembangan kreativitas dan keterampilan teknologi informasi. Salah satu genre yang tetap populer hingga saat ini adalah endless runner 2D. Genre ini menawarkan gameplay yang sederhana namun adiktif, sehingga cocok untuk dimainkan dalam waktu singkat di berbagai platform. Game endless runner 2D memiliki keunggulan performa yang ringan, sehingga dapat berjalan lancar pada perangkat dengan spesifikasi menengah ke bawah yang banyak digunakan masyarakat Indonesia. Namun, sebagian besar pengembangan game di Indonesia masih mengandalkan mesin berbayar seperti Unity, yang pada beberapa periode pernah memberlakukan biaya runtime dan biaya lisensi yang memberatkan pengembang pemula.

Godot Engine muncul sebagai alternatif yang sangat menarik karena bersifat open-source, gratis sepenuhnya, dan memiliki performa ringan khusus untuk pengembangan game 2D. Beberapa penelitian terbaru menunjukkan keunggulan Godot dalam pengembangan game 2D. Dalam penelitian tentang “Pengembangan Game Platformer 2D dengan Godot Engine,” dinyatakan bahwa Godot mampu memberikan performa stabil 60 FPS pada perangkat kelas menengah dengan waktu prototyping yang lebih cepat dibandingkan Unity (Bharambe, 2025). Sementara itu, penelitian lain mengimplementasikan Mesin Status Terhingga pada game platformer 2D menggunakan Godot 4 dan membuktikan fleksibilitas engine ini untuk menghasilkan mekanik yang kompleks tanpa mengorbankan performa (Samel *et al.*, 2026). Meskipun demikian, belum banyak penelitian yang menggabungkan genre endless runner dengan elemen aksi/pertempuran menggunakan Godot Engine. Game ini akan mengusung tema fantasi gelap di mana pemain berperan sebagai ksatria untuk melewati rintangan. Keaslian penelitian ini terletak pada kombinasi antara mekanik endless runner klasik dengan aksi yang unik, serta penerapan Godot Engine dan metodologi Siklus Hidup Pengembangan Game (GDLC) untuk menghasilkan game yang ringan, menarik, dan memiliki nilai replayability tinggi.

Penelitian yang mengembangkan game 2D RPG berbasis Android menggunakan Godot Engine dengan metode Extreme Programming, di mana pemrograman menggunakan Godot Script mengikuti rancangan diagram kelas dalam menggerakkan setiap karakter pada aset game yang dikembangkan. Penelitian ini relevan karena sama-sama menggunakan Godot Engine untuk pengembangan game 2D berbasis Android, namun berbeda dalam hal genre (RPG bukan endless runner), dan tidak adanya elemen pertempuran endless runner (Asgaryansyah & Wiriasto, 2023). Penelitian yang mengimplementasikan GDScript dalam pengembangan game interaktif berbasis multimedia pada Godot Engine, di mana GDScript digunakan untuk mengendalikan pergerakan karakter, sistem animasi, serta interaksi antara pemain dan objek dalam permainan. Penelitian ini relevan karena sama-sama menggunakan GDScript pada Godot Engine untuk mengembangkan sistem karakter dan pertempuran, namun berbeda dalam hal genre game yang dikembangkan (RPG bukan endless runner) dan metode pengembangan yang digunakan (Sudiro *et al.*, 2026). Penelitian yang mengimplementasikan GDScript dalam pengembangan game interaktif berbasis multimedia pada Godot Engine, di mana GDScript digunakan untuk mengendalikan pergerakan karakter, sistem animasi, serta interaksi antara pemain dan objek dalam permainan, dengan penerapan Finite State Machine untuk mengatur perilaku musuh secara dinamis (Kartadinata & Akbar, 2025).

Aplikasi game endless runner 2D “Shadow Veil” menggunakan Godot Engine untuk platform Android dengan metode Game Development Life Cycle (GDLC) mengacu pada beberapa teori utama yang menjadi dasar dalam proses perancangan sistem. Game merupakan bentuk hiburan interaktif yang melibatkan pemain dalam suatu sistem aturan untuk mencapai tujuan tertentu. Game digital didefinisikan sebagai sistem interaksi grafis yang memungkinkan pemain terlibat dalam lingkungan virtual, mengambil keputusan, dan menerima umpan balik langsung. Game digital tidak

hanya berfungsi sebagai media hiburan, tetapi juga dapat dimanfaatkan untuk pendidikan, simulasi, dan pengembangan keterampilan (Starosta *et al.*, 2024). Game digital adalah aktivitas interaktif yang difasilitasi oleh perangkat elektronik, dirancang untuk memberikan pengalaman yang menyenangkan sekaligus mendukung proses pembelajaran atau hiburan. Dalam konteks penelitian ini, game dipahami sebagai aplikasi perangkat lunak interaktif yang dikembangkan menggunakan engine game untuk menghasilkan pengalaman bermain yang menantang dan menyenangkan pada perangkat Android (Li *et al.*, 2024).

Game 2D (Two-Dimensional) adalah jenis game yang menampilkan objek dan lingkungan dalam bidang dua dimensi, yaitu panjang dan lebar, tanpa kedalaman perspektif tiga dimensi. Game 2D merupakan representasi visual yang menggunakan koordinat x dan y untuk menggambarkan posisi objek, sehingga lebih sederhana dalam hal komputasi grafis dibandingkan game 3D (Pradeep Bharambe *et al.*, 2025). Game 2D memiliki keunggulan utama berupa ukuran file yang lebih kecil, konsumsi daya komputasi yang rendah, serta performa yang lebih stabil pada perangkat dengan spesifikasi menengah ke bawah. Hal ini menjadikan game 2D sangat populer untuk pengembangan game mobile. Game 2D menggunakan teknik animasi berbasis sprite dan lingkungan berbasis tile, sehingga memungkinkan pengembang untuk lebih fokus pada gameplay dan mekanik daripada kompleksitas rendering grafis (Samel *et al.*, 2026). Penelitian ini, game 2D dipilih karena sesuai dengan tujuan membuat game "Shadow Veil" yang ringan, gameplay yang menarik, serta dapat berjalan optimal pada platform Android. Penggunaan game 2D juga mempermudah implementasi mekanik endless runner tanpa membebani performa perangkat.

Game 3D (Three-Dimensional) adalah jenis game yang menampilkan objek dan lingkungan dalam ruang tiga dimensi, yaitu memiliki panjang, lebar, serta kedalaman (depth). Game 3D menggunakan koordinat x, y, dan z untuk merepresentasikan posisi objek, sehingga mampu menghasilkan tampilan yang lebih realistis dan imersif dibandingkan game 2D (Lan, 2023). Game 3D biasanya memanfaatkan teknik rendering 3D, pencahayaan dinamis, fisika realistis, dan model karakter yang kompleks. Hal ini membuat game 3D memiliki pengalaman bermain yang lebih mendekati dunia nyata. Namun, game 3D membutuhkan sumber daya komputasi yang jauh lebih besar dibandingkan game 2D, baik dari sisi pemrosesan grafis, memori, maupun waktu pembuatan. Oleh karena itu, pemilihan antara pembuatan game 2D atau game 3D harus disesuaikan dengan tujuan, target pengguna, serta kemampuan teknis tim pengembang yang terlibat. Dalam konteks penelitian ini, pembuatan game "Shadow Veil" menggunakan model 2D, yang dinilai lebih efisien dan sesuai (Alybaev, 2025).

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat mobile layar sentuh seperti smartphone dan tablet. Awalnya dikembangkan untuk perangkat kamera digital, Android berfungsi sebagai jembatan antara perangkat dan penggunanya untuk memanfaatkan berbagai aplikasi (Rahma & Habib, 2021). Karena sifatnya yang open-source, Android menjadi platform populer untuk pengembangan aplikasi, termasuk game, dan menjadi target utama untuk game "Shadow Veil" dalam penelitian ini. Sejarah Android dimulai ketika Android, Inc. didirikan pada tahun 2003 dan diakuisisi oleh Google pada tahun 2005. Setelah akuisisi, Google mengembangkan Android sebagai platform mobile terbuka dan membentuk Open Handset Alliance (OHA) pada tahun 2007 untuk mengembangkan standar terbuka bagi perangkat seluler. Perangkat komersial pertama yang menggunakan Android adalah HTC Dream, yang diluncurkan pada tahun 2008. Sejak itu, Android terus diperbarui dengan penambahan fitur baru (Murni & Husin, 2024). Keterbukaan Android memungkinkan pengembang untuk mengakses dan memodifikasi kode sumber, mendorong pertumbuhan ekosistem developer, termasuk dalam pengembangan game. Sebagai platform pengembangan game, Android memiliki kelebihan dan kekurangan. Kelebihannya meliputi sifat open source yang memungkinkan pengembangan tanpa biaya lisensi besar, basis pengguna yang luas, dan dukungan untuk berbagai engine game gratis seperti Godot Engine. Namun, ada juga kekurangan, seperti fragmentasi perangkat yang memerlukan optimasi untuk berbagai spesifikasi hardware dan variasi versi sistem operasi yang dapat mempengaruhi kompatibilitas game.

Metode Game Development Life Cycle (GDLC) dirancang khusus untuk pengembangan game, terdiri dari enam tahapan: inisiasi, pra-produksi, produksi, pengujian, beta, dan rilis (Wahyu, 2022). Setiap tahapan memiliki fokus yang berbeda, mulai dari konsep hingga pengujian oleh pengguna eksternal. Godot Engine, sebagai game engine open source, memungkinkan pengembangan berbagai jenis game, baik 2D maupun 3D. Dikenal dengan antarmuka ramah pengguna, Godot mendukung pemrograman visual dan memiliki komunitas pengembang aktif. Versi terbaru, Godot Engine 4, dipilih untuk pengembangan game "Shadow Veil" karena kesesuaiannya untuk game 2D endless runner dengan elemen aksi. Penelitian sebelumnya menunjukkan fleksibilitas Godot dalam mengimplementasikan mekanik game yang kompleks (Sudiro *et al.*, 2026). Dalam pengembangan game, diagram UML seperti Use Case Diagram, Activity Diagram, Sequence Diagram, dan Class Diagram sering digunakan untuk menggambarkan fungsionalitas dan interaksi sistem (Wayahdi & Ruziq, 2023).

Penelitian ini merujuk pada beberapa studi sebelumnya yang membahas pengembangan game berbasis Android, khususnya dalam genre endless runner dan penggunaan game engine untuk pengembangan game 2D. Tabel 2.2 menyajikan referensi penelitian yang digunakan sebagai dasar. Contohnya, penelitian oleh Fahrullazi & Riwinoto (2019) menerapkan metode MDA pada game endless runner "Cave Racer" di Android, tetapi tidak mencakup elemen aksi/combat. Tobi *et al.* (2019) mengembangkan game side scrolling "Flying Enggang" menggunakan Godot Engine, namun juga tidak termasuk genre endless runner. Nendya & Redono (2022) menciptakan "Pocong Rush," sebuah game endless runner yang menggunakan metode Finite State Machine, tetapi tidak ada elemen aksi dan menggunakan engine Construct HTML5. Asgaryansyah & Wiriasto (2023) merancang game RPG "Save Our Sea" dengan metode Extreme Programming menggunakan Godot, sementara Samel *et al.* (2026) mengembangkan game platformer "Lost Dune" dengan arsitektur Finite State Machine menggunakan Godot 4. Penelitian-penelitian ini menunjukkan variasi dalam genre, metode pengembangan, dan elemen yang diterapkan dalam game, yang membedakannya dari penelitian ini.

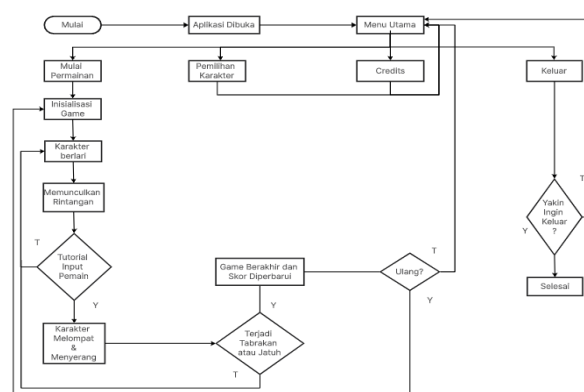
2. Metode Penelitian

Penelitian ini melibatkan beberapa tahap penting dalam pengembangan game "Shadow Veil," dimulai dengan analisis kebutuhan yang terbagi menjadi dua kategori: perangkat keras (hardware) dan perangkat lunak (software). Kebutuhan perangkat keras untuk pengembangan mencakup laptop dengan spesifikasi Intel Core i5, RAM 16 GB, dan SSD 1 TB, sedangkan pengujian dilakukan pada beberapa perangkat Android seperti Tecno Pova Neo 2 dan Realme C67. Untuk perangkat lunak, Godot Engine 4 digunakan sebagai game engine utama, bersama dengan Android SDK, Windows 11, dan Boardmix untuk perancangan antarmuka. Prosedur penelitian dilakukan secara sistematis melalui pengumpulan data, analisis kebutuhan, perancangan, implementasi, dan pengujian. Metode pengumpulan data mencakup observasi terhadap game endless runner yang ada dan wawancara dengan pemain untuk memahami preferensi mereka. Metode pengembangan yang digunakan adalah Game Development Life Cycle (GDLC), yang terdiri dari enam tahapan: Initiation, Pre-Production, Production, Testing, Beta, dan Release (Alam & Mastan, 2024). Pada tahap Initiation, konsep dasar game ditentukan, diikuti dengan perancangan gameplay dan antarmuka selama Pre-Production. Tahap Production meliputi implementasi mekanik dan integrasi aset visual serta audio. Pengujian dilakukan di tahap Testing dan Beta untuk memastikan semua fitur berfungsi dengan baik, sebelum akhirnya game dirilis dalam format APK untuk Android. Analisis kebutuhan mencakup kebutuhan fungsional, seperti sistem skor dan mekanisme deteksi tabrakan, serta kebutuhan non-fungsional yang mencakup kinerja dan keandalan game. Perancangan sistem berfokus pada pembuatan arsitektur game berdasarkan analisis kebutuhan dan spesifikasi yang telah ditetapkan.

Prosedur penelitian dilakukan secara sistematis melalui tahapan pengumpulan data, analisis kebutuhan, perancangan, implementasi, dan pengujian sistem. Seluruh tahapan mengikuti metode pengembangan perangkat lunak yang digunakan sehingga sistem dapat dievaluasi dan disempurnakan secara bertahap.

Prosedur penelitian ini dilakukan secara sistematis melalui tahapan pengumpulan data, analisis kebutuhan, perancangan, implementasi, dan pengujian sistem. Metode pengumpulan data terdiri dari observasi dan wawancara. Observasi dilakukan dengan menganalisis game endless runner yang ada, khususnya game berbasis Android, untuk memahami mekanik gameplay, sistem skor, dan elemen visual. Hasil observasi ini digunakan sebagai acuan dalam merancang mekanik dan fitur game "Shadow Veil," serta untuk memahami fitur Godot Engine yang relevan dalam pengembangan game 2D. Wawancara tidak terstruktur dilakukan kepada mahasiswa dan pihak terkait untuk menggali preferensi pemain terhadap genre game, jenis gameplay yang menarik, dan elemen yang diinginkan. Hasil wawancara membantu menentukan fitur-fitur dalam game, termasuk mekanisme pertarungan dan tampilan visual bertema dark fantasy. Metode pengembangan perangkat lunak yang diterapkan adalah Game Development Life Cycle (GDLC), yang terdiri dari enam tahapan: Initiation, Pre-Production, Production, Testing, Beta, dan Release (Alam & Mastan, 2024). Pada tahap Initiation, konsep dasar game ditentukan, yaitu game endless runner 2D dengan elemen action. Tahap Pre-Production mencakup perancangan gameplay, sistem, dan antarmuka pengguna. Pada tahap Production, semua perancangan diimplementasikan, termasuk mekanik karakter dan musuh. Tahap Testing dilakukan untuk memastikan semua fitur berfungsi sesuai rencana, sedangkan tahap Beta menggunakan metode Black Box Testing untuk memverifikasi fungsi game dari sudut pandang pengguna. Akhirnya, pada tahap Release, game di-build menjadi file APK untuk platform Android, dengan distribusi terbatas untuk keperluan pengujian.

Pada tahap analisis kebutuhan, dokumentasi mendeskripsikan fungsi-fungsi yang harus dimiliki oleh game "Shadow Veil," yang dibagi menjadi kebutuhan fungsional dan non-fungsional. Kebutuhan fungsional mencakup kemampuan sistem untuk mengimplementasikan mekanisme permainan endless runner yang menarik, menyediakan karakter yang dapat melompat dan menyerang, menghasilkan rintangan acak, sistem perhitungan skor, mekanisme deteksi tabrakan, serta antarmuka pengguna yang mencakup menu utama dan tampilan skor. Kebutuhan non-fungsional meliputi kemampuan game berjalan pada perangkat Android dengan RAM minimal 4 GB dan versi Android minimal 10, kinerja yang lancar tanpa lag, antarmuka yang intuitif, keandalan yang stabil, dan portabilitas menggunakan Godot Engine versi 4. Perancangan alur sistem menggambarkan urutan aktivitas atau aliran kerja dalam game. Ini menunjukkan bagaimana pemain berinteraksi dengan sistem dan bagaimana sistem meresponnya. Perancangan alur kerja ini dapat divisualisasikan menggunakan flowchart.

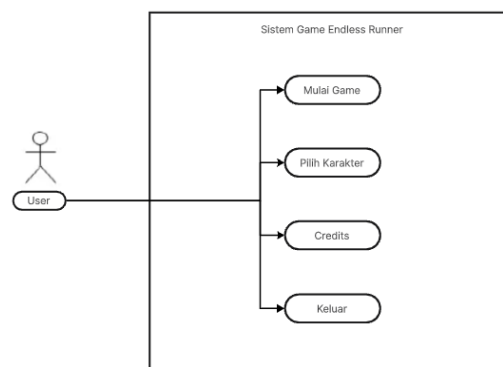


Gambar 1. Flowchart Alur Sistem

Alur sistem game "Shadow Veil" dimulai ketika pengguna menjalankan aplikasi, diikuti dengan tampilan splash screen sebelum menuju menu utama yang menawarkan pilihan untuk mulai permainan, memilih karakter, melihat credits, atau keluar. Setelah memilih untuk bermain, sistem melakukan inisialisasi permainan, memuat aset, dan mengatur posisi karakter. Karakter kemudian berlari otomatis, menghadapi rintangan dan musuh yang muncul secara acak. Pemain dapat melompat atau menyerang berdasarkan input yang terdeteksi. Jika terjadi tabrakan, permainan berakhir dan skor

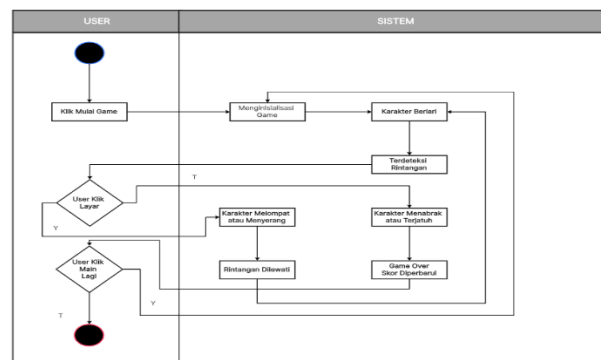
diperbarui. Pemain dapat memilih untuk mengulang permainan, memilih karakter, melihat credits, atau keluar dari aplikasi, menandai akhir proses.

Perancangan User Interface (UI) dalam game "Shadow Veil" mencakup elemen visual dan struktural yang memfasilitasi interaksi pengguna. UI dirancang agar intuitif, dengan menu utama yang menyediakan opsi untuk memulai permainan, memilih karakter, melihat credits, dan keluar. Setiap elemen, seperti tombol dan teks, disusun secara terstruktur untuk meningkatkan pengalaman bermain. Desain gameplay menampilkan latar bertema dark fantasy dengan efek parallax scrolling, karakter yang bergerak otomatis, dan rintangan yang muncul secara dinamis. Selain itu, item collectible seperti coin dan peti ditambahkan untuk meningkatkan skor. Sistem nyawa ditampilkan di sudut layar, sementara tombol pause memungkinkan pemain menghentikan permainan. Desain game over memberikan opsi untuk bermain lagi atau kembali ke menu utama. Objek game, seperti karakter dan rintangan, dirancang untuk berinteraksi secara responsif, mendukung efek visual yang dinamis dan meningkatkan gameplay.



Gambar 2. Use Case Diagram Sistem

Berdasarkan gambar 2. Use Case diagram menjelaskan interaksi pengguna dengan game. "User" merepresentasikan pemain yang berinteraksi dengan sistem. Use Case "Mulai Game" memungkinkan pengguna untuk memulai sesi permainan baru setelah aplikasi diluncurkan. "Pilih Karakter" memberikan fungsionalitas untuk memilih karakter yang akan dimainkan. Use Case "Credits" menampilkan informasi tentang pengembang dan cerita game. "Keluar" memungkinkan pengguna untuk menutup aplikasi dengan aman. Selain itu, Activity Diagram menjelaskan alur kerja gameplay, dimulai dari saat pengguna memilih menu mulai game, di mana karakter berlari, melompat, menyerang, dan menghindari tantangan hingga permainan berakhir jika pengguna menabrak atau jatuh.



Gambar 3. Activity Diagram Mulai Game

Activity Diagram Mulai Game dimulai saat pengguna menekan tombol "Mulai Game," di mana sistem menginisialisasi game dan menjalankan karakter otomatis. Sistem mendeteksi rintangan dan

memungkinkan pemain untuk melompat atau menyerang dengan klik layar. Jika karakter melewati rintangan, gameplay berlanjut; jika tidak, sesi permainan berakhir dan layar game over ditampilkan. Pemain dapat memilih "Main Lagi" untuk memulai sesi baru atau kembali ke menu utama.

Untuk menu "Pilih Karakter" memungkinkan pemain memilih karakter sebelum memulai permainan. Proses dimulai saat pemain menekan tombol "Pilih Karakter" di menu utama, dan sistem menampilkan daftar karakter beserta nama mereka. Setelah memilih, sistem secara otomatis menyimpan pilihan karakter dan mengarahkan pemain kembali ke menu utama, di mana karakter tersebut akan digunakan saat sesi permainan dimulai. Selanjutnya, "Credits" menjelaskan bagaimana informasi pengembang dan cerita game ditampilkan. Ketika pemain menekan tombol "Credits," sistem menampilkan halaman dengan animasi teks bergerak yang memuat informasi tersebut. Pemain dapat membaca informasi ini sebelum menekan tombol kembali untuk menutup halaman dan kembali ke menu utama. Terakhir, "Keluar" menjelaskan mekanisme penutupan aplikasi. Saat pemain menekan tombol "Keluar" di menu utama, sistem langsung menutup aplikasi game. Proses ini memastikan bahwa pengguna dapat dengan mudah mengakhiri sesi permainan dan keluar dari aplikasi dengan cepat. Keberadaan diagram-diagram ini membantu menggambarkan alur interaksi pemain dengan sistem secara jelas dan terstruktur.

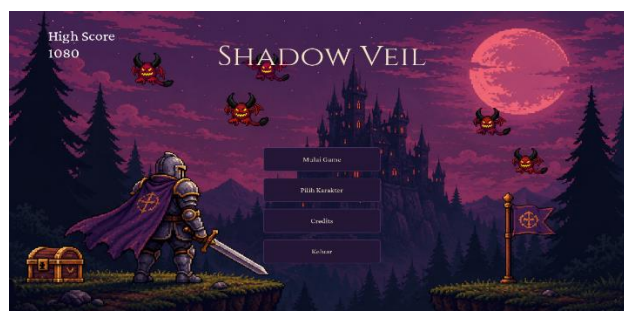
3. Hasil dan Pembahasan

3.1 Hasil

Hasil penelitian ini adalah aplikasi game endless runner 2D berjudul "Shadow Veil," yang dirancang dan diuji menggunakan Godot Engine 4 dengan metode Game Development Life Cycle (GDLC). Game ini dapat dijalankan di Android dan telah melalui pengujian fungsional dengan metode Black Box Testing di berbagai perangkat. Seluruh fitur berfungsi sesuai harapan tanpa kendala. Proses pengembangan mengikuti tahapan GDLC, mulai dari inisiasi, pra-produksi, produksi, hingga pengujian. "Shadow Veil" menggabungkan mekanik endless runner dengan elemen action melawan musuh EvilBat dalam tema dark fantasy. Karakter berlari otomatis, pemain dapat melompat atau menyerang untuk menghindari rintangan dan mengalahkan musuh. Game ini juga memiliki sistem tiga nyawa, High Score, dan item collectible untuk menambah poin.

3.1.1 Tampilan Menu Utama Game

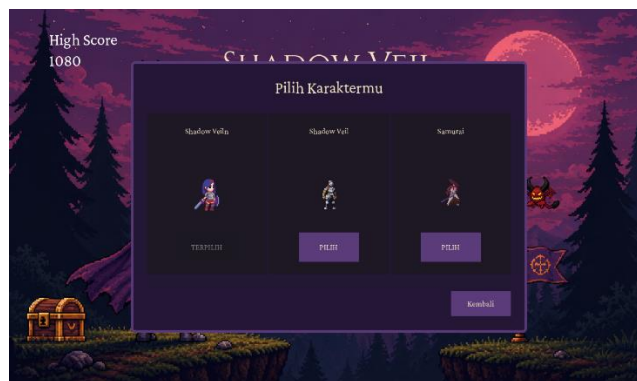
Tampilan menu utama merupakan layar pertama yang muncul ketika pemain membuka aplikasi game "Shadow Veil." Pada layar ini terdapat empat tombol navigasi utama yang tersusun secara vertikal di bagian tengah layar, yaitu tombol mulai game, pilih karakter, credits, dan keluar. Latar belakang menu utama menampilkan suasana malam bertema dark fantasy dengan dominasi warna ungu tua dan merah redup yang mencerminkan identitas visual game "Shadow Veil" secara keseluruhan. Tombol mulai game berfungsi sebagai titik masuk utama untuk memulai permainan. Tombol pilih karakter memberikan akses untuk memilih karakter yang ingin dimainkan. Tombol credits memberikan informasi tentang penulis dan cerita game, sementara tombol keluar berfungsi untuk mengakhiri dan menutup aplikasi. Tampilan ini dibuat berdasarkan perancangan pada gambar.



Gambar 4. Tampilan Menu Utama Game

3.1.2 Tampilan Menu Pilih Karakter

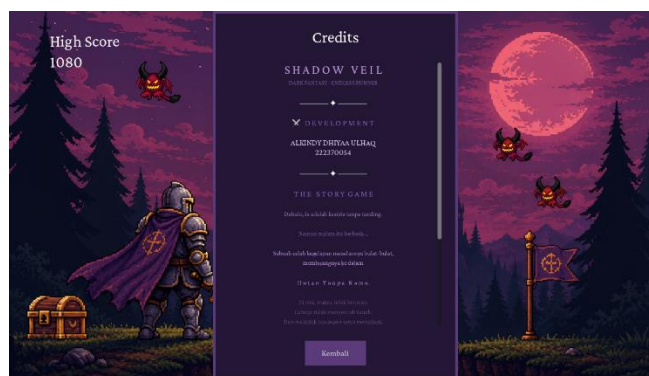
Pada menu pilih karakter, ketika pemain menekan tombol pilih karakter pada menu utama, pada layar pemain dapat melihat seluruh karakter yang tersedia dalam game Shadow Veil dan memilih karakter yang ingin digunakan sebelum memulai permainan. Setiap karakter ditampilkan dengan style yang unik dan berbeda satu sama lain, disertai dengan nama yang tertera di masing-masing karakter.



Gambar 5. Tampilan Menu Pilih Karakter

3.1.3 Tampilan Menu Credits

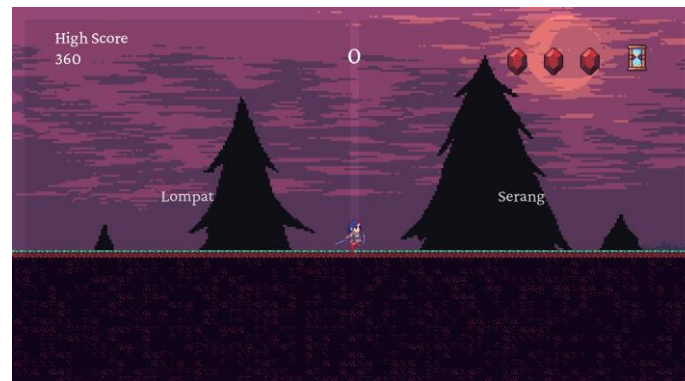
Pada menu *credits*, saat pemain menekan tombol *credits* akan menampilkan informasi mengenai pengembang game Shadow Veil serta *story game*. Layar *credits* dirancang dengan tampilan visual yang unik dengan tema *dark fantasy game* Shadow Veil, menggunakan latar belakang dan gaya teks yang selaras dengan keseluruhan identitas visual game.



Gambar 6. Tampilan Menu Credits

1.3.4 Tampilan Game Dimulai

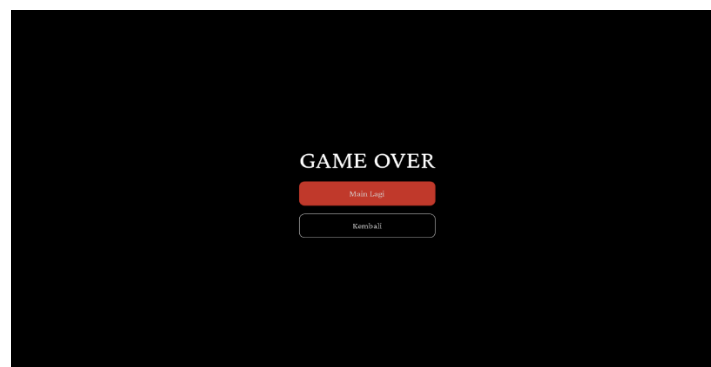
Tampilan ini merupakan layar yang muncul pada awal sesi permainan sebelum *gameplay* aktif sepenuhnya berjalan. Pada tahap ini sistem menampilkan tutorial singkat kepada pemain mengenai cara mengendalikan karakter selama permainan berlangsung. Tutorial ditampilkan dalam bentuk petunjuk visual, yang menjelaskan bahwa pemain dapat menekan sisi kiri layar untuk membuat karakter melompat guna menghindari rintangan yang ada di bawah, serta menekan sisi kanan layar untuk membuat karakter menyerang gunanya untuk mengalahkan musuh yang menghalangi perjalanan. Petunjuk tutorial ini dirancang sederhana mungkin agar dapat dipahami dengan cepat oleh pemain tanpa mengganggu pengalaman bermain, sehingga pemain dapat segera memahami kontrol dasar game dan mulai bermain tanpa memerlukan panduan tambahan.



Gambar 7. Tampilan Awal Game Dimulai

1.3.5 Tampilan Game Over

Pada saat karakter menabrak atau jatuh, game akan menampilkan *game over* dan terdapat 2 tombol yang dapat pemain pilih yaitu tombol main lagi dan kembali. Tombol main lagi dapat pemain pilih untuk mengulang sesi permainan dari awal dan skor akan di *reset*, Sedangkan tombol kembali dapat membuat pemain ke menu utama game.



Gambar 8. Tampilan Saat Game Berakhir

3.2 Pembahasan

Game "Shadow Veil" berhasil dikembangkan sebagai aplikasi game endless runner 2D berbasis Android yang mengintegrasikan mekanik lari otomatis dengan elemen aksi berupa sistem pertarungan melawan musuh dalam satu gameplay yang utuh. Proses pengembangan game ini mengikuti metodologi Game Development Life Cycle (GDLC), yang terbukti efektif dalam memandu peneliti melalui setiap tahapan secara terstruktur, mulai dari konsep awal hingga produk akhir yang siap diuji. Penerapan GDLC dalam pengembangan game ini sejalan dengan temuan Alam dan Mastan (2024), yang menunjukkan bahwa pendekatan terstruktur dalam pengembangan game dapat meningkatkan efisiensi dan kualitas produk akhir. Penggunaan Godot Engine sebagai alat pengembangan menunjukkan kemampuan untuk menghasilkan game 2D berbasis Android yang ringan dengan performa stabil. Godot Engine, yang merupakan software open-source dan gratis, memberikan kemudahan bagi pengembang untuk menciptakan game tanpa terbebani oleh biaya lisensi. Hal ini sejalan dengan analisis Alybaev (2025) yang membandingkan Godot dengan Unity, di mana Godot dinilai lebih ramah bagi pengembang indie karena biaya yang lebih rendah dan fleksibilitas yang ditawarkan. Dengan demikian, Godot Engine terbukti menjadi alternatif handal untuk pengembangan game 2D di platform Android.

Kombinasi genre endless runner dengan elemen aksi melawan musuh, seperti EvilBat, menciptakan gameplay yang lebih variatif dibandingkan endless runner konvensional yang hanya berfokus pada mekanik menghindar. Penambahan sistem tiga nyawa, item collectible berupa koin dan peti, serta mekanisme High Score memberikan motivasi tambahan bagi pemain untuk terus bermain

dan meningkatkan pencapaian mereka. Ini sejalan dengan penelitian Fahrullazi dan Riwinoto (2019) yang menunjukkan bahwa elemen gameplay yang bervariasi dapat meningkatkan keterlibatan pemain dalam game. Pengujian dan integrasi aplikasi game "Shadow Veil" dilakukan menggunakan metode black box testing untuk memastikan bahwa setiap fungsi utama beroperasi sesuai harapan, tanpa memperhatikan kode program secara internal. Fokus pengujian tertuju pada fungsionalitas yang dapat diamati oleh pengguna, termasuk seluruh menu, kontrol karakter, dan efek visual yang dinamis. Tabel pengujian mencakup berbagai skenario, langkah uji, hasil yang diharapkan, hasil aktual, dan status pengujian. Dari hasil pengujian, semua skenario yang diuji menunjukkan status "Berhasil," mulai dari menjalankan aplikasi, menampilkan menu utama, hingga fungsi tombol yang beragam seperti mulai game, pilih karakter, dan keluar. Kontrol karakter untuk melompat dan menyerang juga berfungsi dengan baik, dan rintangan muncul secara acak selama gameplay.

Deteksi tabrakan berhasil menampilkan layar game over saat karakter jatuh atau terkena serangan. Tombol "main lagi" dan "kembali" di layar game over juga berfungsi dengan baik, memungkinkan pemain untuk memulai ulang permainan atau kembali ke menu utama. Hasil pengujian ini menunjukkan bahwa seluruh fitur dalam game berjalan sesuai harapan, memastikan pengalaman bermain yang optimal bagi pengguna. Penelitian ini sejalan dengan temuan Narayanan (2021) yang menekankan pentingnya pengujian menyeluruh dalam pengembangan game untuk memastikan kualitas dan kepuasan pengguna. "Shadow Veil" tidak hanya berhasil sebagai aplikasi game yang menghibur, tetapi juga sebagai contoh yang baik dari penerapan metodologi pengembangan game yang efektif dan penggunaan alat yang tepat. Dengan mempertimbangkan umpan balik dari pengujian, pengembangan lebih lanjut dapat dilakukan untuk meningkatkan pengalaman bermain dan menambah fitur baru yang dapat menarik lebih banyak pemain.

4. Kesimpulan dan Saran

Berdasarkan hasil penelitian pengembangan aplikasi game endless runner 2D "Shadow Veil" menggunakan Godot Engine, dapat disimpulkan bahwa game ini berhasil dirancang dan dibangun dengan menerapkan metodologi Game Development Life Cycle (GDLC) dari tahap inisiasi hingga pengujian. Seluruh tahapan GDLC diimplementasikan secara sistematis, menghasilkan produk yang terstruktur dan terdokumentasi dengan baik. Game ini menggabungkan mekanik endless runner klasik dengan elemen action melawan musuh EvilBat, menciptakan gameplay yang variatif dan meningkatkan nilai replayability melalui penambahan sistem tiga nyawa, item collectible, dan mekanisme High Score. Godot Engine 4 terbukti sebagai alternatif handal untuk pengembangan game 2D berbasis Android, dan "Shadow Veil" memberikan nilai edukatif kepada pemain melalui latihan refleksi dan pengambilan keputusan.

Saran untuk pengembangan selanjutnya mencakup penambahan variasi karakter, musuh, dan rintangan yang lebih kompleks untuk meningkatkan tantangan gameplay. Fitur multiplayer atau leaderboard online juga disarankan untuk meningkatkan interaksi sosial antar pemain. Selain itu, mendistribusikan game melalui Google Play Store dapat memperluas jangkauan pengguna Android, sehingga "Shadow Veil" dapat dinikmati oleh lebih banyak pemain.

5. Daftar Pustaka

- Alam, N. C., & Mastan, I. A. (2024). Aplikasi game "Waddle" dengan menggunakan teknologi game engine Godot. *Waddle Game Application Using Godot Game Engine Technology*, 7(1), 37–48.
- Allybaev, A. (2025). Comparative analysis of Unity and Godot for 2D game development. *Preprints*, 0–3. <https://doi.org/10.20944/preprints202511.1981.v1>

- Asgaryansyah, K., & Wiriasto, G. W. (2023). Save Our Sea: Rancangan game 2D RPG berbasis Android dengan metode Extreme Programming menggunakan Godot Script. *Dielektrika – Department of Electrical Engineering University of Mataram*, 10(2), 127–138.
- Bharambe, N. P. (2025). 2D platformer game development with Godot engine. *International Journal of Advance Research*, 11(1), 363–365. <https://www.ijariit.com/manuscript/2d-platformer-game-development-with-godot-engine/>
- Fahrullazi, M., & Riwinoto, R. (2019). Implementasi penerapan MDA pada game endless runner 2D CAVE RACER berbasis Android. *Journal of Applied Multimedia and Networking*, 3(2), 19–29. <http://jurnal.polibatam.ac.id/index.php/JAMN/article/view/1636>
- Fauzi, J. R. (2020). Algoritma dan flowchart dalam menyelesaikan suatu masalah. *20330044*.
- Kartadinata, A., & Akbar, M. (2025). Implementasi sistem character player pada game RPG 2D menggunakan game engine Godot. *Jurnal Informatika Dan Teknik Elektro Terapan*, 13(3). <https://doi.org/10.23960/jitet.v13i3.7163>
- Kus Indrani Listyoningrum, D. Y. F., & Nurhasan Hamidi. (2023). Inovasi berkelanjutan dalam bisnis: Manfaatkan flowchart untuk mengoptimalkan nilai limbah perusahaan. *Jurnal Informasi Pengabdian Masyarakat*, 1(4), 100–112. <https://doi.org/10.47861/jipm-nalanda.v1i4.552>
- Lan, Y. (2023). Development and application of 3D modeling in game. *Academic Journal of Science and Technology*, 7(2), 94–97. <https://doi.org/10.54097/ajst.v7i2.11949>
- Li, Y., Chen, D., & Deng, X. (2024). The impact of digital educational games on student's motivation for learning: The mediating effect of learning engagement and the moderating effect of the digital environment. *PLoS ONE*, 19(1), 1–21. <https://doi.org/10.1371/journal.pone.0294350>
- Murni, K. C., & Husin, S. M. (2024). 337-1073-3-Pb (1). *Jurnal Spirit*, 16, 161–169.
- Narayanan, S. (2021). 3D puzzle game using Godot engine. *International Journal of Advance Research*, 9(2), 152–154.
- Nendya, M. B., & Redono, D. (2022). Pocong Rush: Endless runner game based on finite state machine. *JOINCS (Journal of Informatics, Network, and Computer Science)*, 5(1), 14–20. <https://doi.org/10.21070/joincs.v5i1.1592>
- Newzoo. (2024). *Global games market report 2024*. Newzoo.
- Pradeep Bharambe, N., Acharya, G. V., Prasad Mhaddalkar, Y., & Santosh Jadhav, V. (2025). 2D platformer game development with Godot engine. *International Journal of Advance Research*, 11(1), 2454–132. <https://www.ijariit.com>
- Rahma, A., & Habib, M. (2021). Center of Knowledge: Jurnal Pendidikan dan Pengabdian Masyarakat, Volume 1, No 1, Agustus 2021 Android dan masa depan: Analisis dampak terhadap pengguna. *1(1)*, 12–21.
- Rahman, & Alfarisi. (2019). Perancangan aplikasi permainan endless runner pada CV. KiraDev. *Jurnal IDEALIS, Universitas Budi Luhur*.

- Samel, M., Mhatre, T., Banekar, S., & Chaudhari, U. (2026). Lost Dune: A 2D platformer game using Godot 4 engine with finite state machine architecture. *International Journal of Advance Research*, 9(8), 1766–1770.
- Starosta, J., Kiszka, P., Szyszka, P. D., Starzec, S., & Strojny, P. (2024). The tangled ways to classify games: A systematic review of how games are classified in psychological research. *PLoS ONE*, 19(6). <https://doi.org/10.1371/journal.pone.0299819>
- Sudiro, S., Satria, C. D., Nugroho, A., & Nurfauzi, M. (2026). Implementasi GDScript dalam pengembangan game interaktif berbasis multimedia pada Godot engine. *International Journal of Advance Research*, 4.
- Tobi, C. A. R., Andrea, R., & Yusika, A. (2019). Membangun side scrolling game Flying Enggang berbasis Android dengan Godot engine. *Jurnal Informatika Wicida*, 8(1), 1–8. <https://doi.org/10.46984/inf-wcd.1234>
- Wahyu, S. (2022). Penerapan metode game development life cycle pada pengembangan aplikasi game pembelajaran budi pekerti. *Skanika*, 5(1), 82–91. <https://doi.org/10.36080/skanika.v5i1.2904>
- Wayahdi, M. R., & Ruziq, F. (2023). Pemodelan sistem penerimaan anggota baru dengan Unified Modeling Language (UML) (Studi Kasus: Programmer Association of Battuta). *Jurnal Minfo Polgan*, 12(1), 1514–1521. <https://doi.org/10.33395/jmp.v12i1.12870>