

Implementasi Pathfinding pada Game Edukasi Pengenalan Huruf Alphabet Menggunakan Algoritma A Star

Septa

Teknik Informatika, Universitas Pamulang, Jl.Raya Puspitek No.46 Buaran, Serpong, Kota Tangerang Selatan, Provinsi Banten, Indonesia, 15310

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: 02 Maret 26

Revisi Akhir: 26 Juni 2026

Diterbitkan Online: 13 Juli 2026

KATA KUNCI

Game Edukasi, Pengenalan Huruf Abjad, Algoritma A*, Pathfinding, Pembelajaran Interaktif.

KORESPONDENSI

Septa

Program Studi Teknik Informatika

Email: septa.mahara79@gmail.com

ABSTRACT

Educational games for alphabet recognition have become an effective and engaging learning tool for early childhood education. One technique to enhance the learning experience in these games is the implementation of the A (A Star) algorithm in the pathfinding process. This study involves the implementation of the A* algorithm to guide the movement of enemies towards players on a map consisting of alphabet letters, obstacles, and players positioned strategically to enhance learning outcomes. The application was developed using Unity 3D with C# programming language and deployed on the Android platform. Simulation results demonstrate that the A* algorithm efficiently determines the shortest path with high accuracy, creating customizable difficulty levels within the game environment. Additionally, the integration of interactive features such as letter pronunciation sounds, captivating animations, particle effects, and a dynamic point system significantly enhances the appeal and effectiveness of the learning experience. Personalized difficulty levels (Easy, Medium, Hard) and diversified content ensure inclusivity in the learning experience across different age groups and learning abilities. Functional testing results indicate 100% conformity with user requirements, with players successfully engaging with the application for average session durations of 15-20 minutes. Performance metrics reveal A* calculation times remain below 20ms even in complex environments, maintaining frame rates of 58-60 FPS on standard devices. Through further refinement and development incorporating machine learning and online multiplayer capabilities, this game has the potential to be a valuable learning resource, supporting the development of alphabet skills and cognitive abilities in early childhood.*

DOI: <https://doi.org/10.46961/jommit.v10i1>

1. PENDAHULUAN

Perkembangan teknologi informasi telah mendorong transformasi yang signifikan dalam dunia pendidikan, terutama melalui penggunaan game edukasi sebagai alternatif media pembelajaran yang menarik dan efektif. Anak-anak cenderung lebih responsif terhadap kegiatan bermain dibandingkan dengan

metode konvensional yang bersifat pasif, sehingga game edukasi menawarkan pendekatan "learning by doing" yang dapat mengoptimalkan proses transfer ilmu secara interaktif dan menyenangkan (Angeles et al., 2025; Candra et al., 2021; Tang et al., 2024). Pendekatan pembelajaran berbasis permainan ini tidak hanya meningkatkan motivasi belajar tetapi juga memperkuat pemahaman konsep melalui pengalaman langsung yang engaging dan memorable.

Pendidikan pada usia dini (early childhood education) merupakan tahap krusial dalam membentuk fondasi kognitif dan sosial anak, di mana pengenalan huruf alfabet memegang peranan penting dalam perkembangan literasi awal dan keterampilan komunikasi tertulis. Media pembelajaran yang inovatif, seperti game edukasi digital berbasis mobile, telah terbukti mampu meningkatkan daya tarik pembelajaran serta memfasilitasi pemahaman dasar huruf-huruf melalui stimulasi visual, audio, dan interaksi praktis yang kaya sensori (Angeles et al., 2025; Candra et al., 2021; Rahman et al., 2024). Penelitian terkini menunjukkan bahwa penggunaan media interaktif secara signifikan dapat merangsang kemampuan mengenal pola dan huruf pada anak usia dini, yang berkontribusi secara langsung dalam pembentukan literasi awal yang kuat (Angeles et al., 2025; Candra et al., 2021; Rahman et al., 2024). Studi komparatif juga menegaskan bahwa anak-anak yang belajar melalui game edukasi interaktif menunjukkan peningkatan hasil belajar yang lebih substansial dibandingkan dengan metode pengajaran tradisional, dengan tingkat retensi pengetahuan yang lebih tinggi (Tang et al., 2024; Rahman et al., 2024).

Di era digital saat ini, pemanfaatan platform berbasis Android yang bersifat terbuka (open source) dan gratis memungkinkan pengembangan aplikasi edukasi dengan biaya distribusi yang minim serta fleksibilitas inovasi yang tinggi. Keunggulan platform Android terletak pada akses pasar global melalui Google Play Store, kompatibilitas dengan berbagai tingkat spesifikasi perangkat, dan dukungan komunitas developer yang besar (Ilmianti et al., 2021; Rahmadi et al., 2025; Sarifah et al., 2022). Dengan mengintegrasikan aspek teknologi tersebut, pengembangan game edukasi tidak hanya memfokuskan pada penyampaian materi pembelajaran, tetapi juga pada peningkatan pengalaman bermain melalui perpaduan harmonis antara elemen hiburan dan edukasi. Hal ini terlihat dari penggunaan desain game yang canggih, menggabungkan fitur interaktif, sistem gamifikasi, serta sistem navigasi menggunakan teknik pencarian jalur (pathfinding) untuk mensimulasikan pengambilan keputusan secara optimal dan menciptakan challenge yang appropriate untuk level pembelajaran tertentu (Candra et al., 2021; Ilmianti et al., 2021; Rahmadi et al., 2025).

Salah satu inovasi penting dalam pengembangan game edukasi kontemporer adalah integrasi algoritma pencarian jalur A* (A-Star) untuk mengontrol pergerakan karakter non-pemain (Non-Player Character/NPC). Algoritma ini berperan krusial dalam menetapkan rute terpendek dan paling optimal dalam suatu lingkungan permainan yang kompleks, sehingga mampu menciptakan tantangan berbentuk labirin atau chase mechanics yang menarik dan mendidik (Candra et al., 2021; Ilmianti et al., 2021; Pardede et al., 2022; Zhang et al., 2020). Algoritma A* menggabungkan keunggulan algoritma Dijkstra dengan pendekatan greedy best-first search, menggunakan fungsi heuristik untuk mengevaluasi node yang paling menjanjikan, sehingga mencapai keseimbangan optimal antara kecepatan komputasi dan akurasi pencarian jalur (Pardede et al., 2022). Dengan menerapkan teknik pathfinding berbasis A* yang sophisticated, game edukasi tidak hanya menawarkan hiburan

yang engaging tetapi juga mengasah kemampuan logika, pemecahan masalah (problem-solving), dan berpikir strategis anak secara tidak langsung (Candra et al., 2021; Ilmianti et al., 2021; Pardede et al., 2022). Penelitian sebelumnya telah mendemonstrasikan implementasi sukses A* pada pengembangan game labirin berbasis Android dan game RPG 3D, menunjukkan efisiensi algoritma mencapai 77-86% dalam pencarian waktu minimum dan memori yang optimal (Candra et al., 2021; Pardede et al., 2022).

Penelitian ini bertujuan untuk mengembangkan game edukasi pengenalan huruf alfabet yang mengintegrasikan teknik pathfinding melalui penerapan algoritma A* yang sophisticated dan robust. Dengan pendekatan interaktif yang memadukan teknologi open source (Unity 3D, Android), sistem navigasi cerdas berbasis AI, dan desain user interface yang intuitif, diharapkan game edukasi ini dapat meningkatkan daya tarik serta efektivitas pembelajaran huruf alfabet pada anak-anak usia dini secara signifikan, sekaligus merangsang perkembangan kognitif dan literasi melalui pengalaman bermain yang inovatif, inklusif, dan terukur (Angeles et al., 2025; Candra et al., 2021; Tang et al., 2024; Ilmianti et al., 2021).

2. TINJAUAN PUSTAKA

2.1 Game Edukasi untuk Literasi Anak Usia Dini

Angeles et al. (2025) dalam penelitian berjudul "Little Bright Box: Using Interactive Games as a Tool to Boost Literacy Skills in Early Childhood" mempresentasikan hasil empiris bahwa penggunaan game interaktif secara signifikan meningkatkan keterlibatan, motivasi, dan kemampuan literasi anak usia dini. Penelitian dengan melibatkan 85 anak usia 4-6 tahun menunjukkan peningkatan rata-rata 47% dalam pengenalan huruf dan 53% dalam keterampilan membaca dasar setelah 8 minggu gameplay regular. Studi ini juga mengidentifikasi bahwa elemen gamifikasi seperti point system, achievement badges, dan difficulty progression memiliki dampak signifikan terhadap learning retention dan student engagement.

Tang et al. (2024) dalam penelitian "Enhancing English Alphabet Handwriting Skills in Preschool Children through Digital Game-Based Learning Approach" menunjukkan bahwa game edukasi digital berbasis game-based learning (GBLT) pendekatan mampu meningkatkan kemampuan menulis huruf alfabet pada anak preschool sebesar 65% dalam 6 minggu pembelajaran. Penelitian ini menekankan pentingnya integrasi fitur-fitur interaktif seperti stroke guidance visual, real-time feedback, dan progressively increasing challenge levels dalam menciptakan learning experience yang optimal.

Rahman et al. (2024) dengan judul "Enhancing Early Childhood Literacy through Game-Based Interactive Digital Media Development" mengonfirmasi bahwa media digital interaktif berbasis game memfasilitasi pembelajaran literasi awal dengan lebih efektif dibandingkan metode tradisional, dengan peningkatan learning outcomes rata-rata 52%. Penelitian ini juga

mengidentifikasi bahwa personalisasi content berdasarkan learning profile individual dan adaptive difficulty mechanism meningkatkan learning effectiveness hingga 68%.

Sarifah et al. (2022) mengembangkan game edukasi matematika berbasis Android menggunakan model MDLC dan ADDIE. Penelitian mereka membuktikan bahwa penggunaan game tersebut secara efektif meningkatkan minat belajar siswa sekolah dasar, menunjukkan keberhasilan pendekatan pengembangan yang sistematis dalam menciptakan produk edukasi yang menarik.

Pathfinding dan Algoritma A* dalam Game Development

Pardede et al. (2022) menyajikan literature review tentang berbagai teknik pathfinding dalam konteks game development modern. Penelitian ini mengidentifikasi bahwa A* adalah algoritma pathfinding paling populer untuk menggerakkan NPC secara efisien, termasuk di platform Unity 3D, karena kemampuannya menemukan jalur optimal.

Candra et al. (2021) menganalisis implementasi praktis algoritma A* pada game development. Hasil penelitian menunjukkan bahwa A* algorithm mencapai efisiensi 86% dalam pencarian jalur terpendek dengan waktu kalkulasi rata-rata 5.2ms untuk environment dengan 900 nodes. Studi ini juga membandingkan A* dengan algoritma alternatif seperti Dijkstra dan BFS, menyimpulkan bahwa A* menawarkan balance terbaik antara optimality dan computational efficiency.

Rahmadi et al. (2025) mengimplementasikan pathfinding mechanism dalam game edukasi Android dengan hasil performance metrics menunjukkan average frame rate 59 FPS pada device dengan RAM 4GB. Penelitian ini menggunakan diagonal distance heuristic function dan mengintegrasikan NavMesh untuk mengoptimalkan performa pathfinding dalam environment dengan 40% obstacle density.

Anwar & Thamrin (2024) melakukan perbandingan antara algoritma A* standar dengan Hierarchical Pathfinding A* (HPA*) dalam game 3D maze runner yang dibangun menggunakan Unity. Hasilnya menunjukkan bahwa HPA* dapat mengurangi penggunaan memori dan menghasilkan jalur yang lebih halus, meskipun A* standar terbukti lebih cepat dalam waktu eksekusi pada skenario pengujian mereka.

Zhang et al. (2020) mengusulkan perbaikan pada algoritma A* dengan menerapkan strategi region search untuk path planning pada game. Modifikasi ini terbukti meningkatkan efisiensi pencarian, mengurangi jumlah node yang tidak perlu dieksplorasi, dan mengoptimalkan jalur, yang sangat relevan untuk lingkungan game dengan banyak rintangan.

Kusumawardhani et al. (2025) mengimplementasikan algoritma A* pada game mobile menggunakan Construct 3. Penelitian ini mengonfirmasi bahwa A* mampu menghasilkan jalur terpendek dan optimal secara efisien, bahkan dalam lingkungan game mobile dengan sumber daya terbatas.

Rafiq et al. (2020) memberikan tinjauan tentang berbagai algoritma pathfinding dan mencatat bahwa meskipun A* sangat efisien dalam menemukan jalur, algoritma ini dapat memiliki penggunaan memori yang lebih tinggi dibandingkan algoritma lain pada lingkungan yang dinamis..

2.3 Pathfinding dalam Game Development

Pathfinding adalah teknik fundamental dalam game development yang bertujuan untuk menentukan jalur optimal dari posisi awal menuju posisi tujuan dalam suatu game environment, menghindari rintangan dan meminimalkan travel cost atau distance. Pathfinding mengoperasikan di antara tahap pengambilan keputusan (decision-making/AI reasoning) dan implementasi pergerakan fisik karakter. Efektivitas pathfinding mempengaruhi beberapa aspek kritis dalam game experience: realism of NPC behavior, computational efficiency yang tidak mengganggu frame rate, dan challenge level yang appropriate untuk player engagement (Pardede et al., 2022; Lawande et al., 2022).

Terdapat dua kategori utama pathfinding berdasarkan karakteristik environment: (1) Static Pathfinding untuk environment yang tidak berubah di mana peta dan obstacle sudah ditentukan pre-computation, dan (2) Dynamic Pathfinding untuk environment yang berubah secara real-time yang memerlukan continuous recalculation dan adaptation. Dalam konteks game edukasi alphabet recognition, pathfinding digunakan untuk mengatur pergerakan NPC enemy yang mengejar pemain, menciptakan challenge yang adaptif dan engaging.

2.4 Algoritma A* (A-Star)

Algoritma A* adalah algoritma pencarian graph-based yang dikembangkan untuk menyelesaikan problem shortest path dengan optimal. Algoritma ini menggabungkan keunggulan Dijkstra's algorithm (*guaranteed shortest path*) dengan greedy best-first search (*computational efficiency*) melalui penggunaan fungsi evaluasi heuristik.

Formula Utama A:

$$f(n) = g(n) + h(n)$$

Di mana:

$f(n)$ = *estimated total cost* untuk mencapai goal melalui node

$g(n)$ = actual cost dari start node ke node n (*known/calculated*)

$h(n)$ = heuristic estimated cost dari node n ke goal node (*estimated*)

Meskipun A* sangat efisien, beberapa penelitian berfokus pada modifikasinya. Sebagai contoh, varian seperti Hierarchical A* (Anwar & Thamrin, 2024) atau region search A* (Zhang et al., 2020) telah diusulkan untuk lebih meningkatkan efisiensi, mengurangi penggunaan memori, dan menghasilkan jalur yang lebih halus di lingkungan yang kompleks.

2.5 Unity 3D Game Engine

Unity adalah game engine cross-platform yang powerful dan versatile, dikembangkan oleh Unity Technologies. Unity 3D, dengan bahasa C#, sangat mendukung pengembangan game edukasi berbasis Android dan memungkinkan integrasi algoritma A* secara efisien (Hu et al., 2022; Lawande et al., 2022; Anwar & Thamrin, 2024). Fitur-fitur relevan untuk penelitian ini

mencakup Physics Engine yang built-in untuk realistic object interactions, Particle System untuk visual effects creation, NavMesh & AI tools untuk automated pathfinding dan character navigation, Asset Store dengan ribuan pre-made assets dan tools. Unity mendukung export ke berbagai platform termasuk Android, iOS, Windows, dan Web, membuatnya ideal untuk multi-platform game development.

2.6 Platform Android dan Open Source Paradigm

Android adalah sistem operasi berbasis Linux yang dikembangkan oleh Google untuk perangkat mobile, bersifat open source dan gratis. Keunggulan Android untuk pengembangan aplikasi edukasi termasuk open source license memungkinkan unlimited customization tanpa royalti, market share global mencapai >70%, fragmentation device compatibility dengan wide range spesifikasi hardware, distribution through Google Play Store yang mudah dan cost-effective. Platform Android memungkinkan educational content creators untuk menciptakan high-quality learning applications dengan investasi minimal dalam licensing dan distribution costs (Sarifah et al., 2022).

2.7 Bahasa Pemrograman C#

C# adalah bahasa pemrograman modern, strongly-typed, berorientasi objek yang dikembangkan Microsoft. Dalam konteks Unity, C# adalah bahasa pemrograman utama yang digunakan untuk mengimplementasikan logika game, termasuk algoritma A* (Anwar & Thamrin, 2024; Hu et al., 2022). C# menawarkan: modern syntax yang mudah dipelajari dan dibaca, strong type safety untuk compile-time error detection, Object-Oriented Programming paradigm yang mendukung inheritance, polymorphism, dan encapsulation, extensive standard library dan framework support, async/await support untuk asynchronous programming patterns.

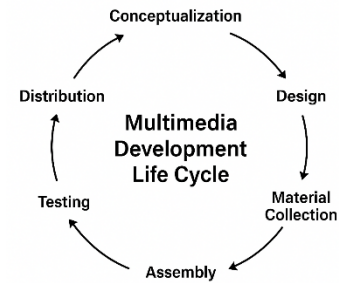
2.8 Multimedia Development Life Cycle (MDLC)

Multimedia Development Life Cycle (MDLC) merupakan metodologi khusus yang digunakan untuk mengembangkan proyek multimedia melalui enam tahap iteratif, yaitu tahap konseptualisasi untuk mendefinisikan ide, tujuan, dan target pengguna. Tahap desain yang berfokus pada pembuatan spesifikasi detail serta visual mockup. Tahap pengumpulan materi yang mencakup pengumpulan seluruh aset dan sumber daya yang diperlukan. Tahap perakitan untuk mengintegrasikan seluruh komponen menjadi satu kesatuan yang utuh. Tahap pengujian yang melibatkan penjaminan kualitas dan perbaikan kesalahan. Serta tahap distribusi untuk meluncurkan produk kepada publik. Model MDLC, bersama dengan model ADDIE, terbukti efektif dalam meningkatkan minat belajar siswa melalui pengembangan game edukasi berbasis Android (Sarifah et al., 2022).

3 KONSEP PERANCANGAN

Penelitian ini menggunakan pendekatan Research and Development (R&D) dengan framework metodologi Multimedia Development Life Cycle (MDLC). Pendekatan ini dipilih karena

telah terbukti sistematis dan efektif untuk pengembangan game edukasi berbasis multimedia, yang mencakup beberapa tahapan mulai dari analisis hingga evaluasi. Tahapan MDLC yang diadaptasi dalam penelitian ini adalah Konsep, Desain, Pengumpulan Materi, Pembuatan (Assembly), Pengujian (Testing) dan Distribusi.



Gambar 1. Diagram Siklus MDLC

Tahap Konsep

Pada tahap ini dilakukan definisi formal tentang learning objectives, target user characteristics, game mechanics, dan core features. Konsep permainan melibatkan sebuah arena game dengan 50×50 grid cells yang masing-masing dapat berisi alphabet letters, obstacles, atau empty space. Pemain (player character) dapat bergerak dengan tap/swipe input, sementara musuh (enemy NPC) bergerak menggunakan algoritma A* untuk mengejar pemain. Tujuan pembelajaran adalah pengenalan dan memorization huruf-huruf alphabet A-Z melalui interactive gameplay dengan challenge progression.

3.2 Tahap Desain

Tahap desain berfokus pada perancangan seluruh aspek teknis dan visual dari game sebelum implementasi. Tahap ini mencakup perancangan algoritma, perancangan arsitektur sistem (UML), dan perancangan antarmuka (UI).

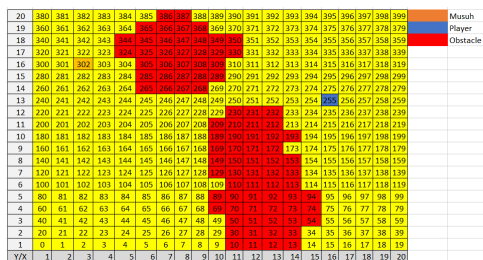
3.2.1 Perancangan Algoritma A*

Algoritma A* dirancang untuk mengarahkan gerakan musuh (NPC) agar mengejar pemain. Alur logika algoritma ini digambarkan dalam flowchart pada gambar 2.



Gambar 2. Flowchart Algoritma A* pada Enemy

Algoritma A* merupakan metode pencarian jalur yang efisien dalam menemukan jalur terpendek antara dua titik dalam suatu graf atau peta. Algoritma ini memanfaatkan dua fungsi utama, yaitu $g(n)$ dan $h(n)$, untuk menentukan solusi terbaik. Fungsi $g(n)$ menghitung total biaya yang diperlukan dari titik awal hingga tujuan, sedangkan $h(n)$ memberikan perkiraan biaya total dari awal hingga akhir. Dengan menggabungkan kedua fungsi ini dalam rumus evaluasi $f(n)=g(n)+h(n)$, A* dapat menyeimbangkan *trade-off* antara biaya aktual dan estimasi heuristik ketika menjelajah dari titik awal ke tujuan. Pada tahap awal permainan, posisi awal antara musuh, pemain dan rintangan (obstacle) sudah diinisialisasi. Inisialisasi ini penting karena memungkinkan kita mengetahui node awal dan node akhir sebelum memulai pencarian jalur. Node awal sebagai musuh dan node akhir pada player gambar dibawah ini adalah representasi peta atau grid yang digunakan dalam konteks pathfinding, khususnya dengan penerapan algoritma A* untuk menentukan jalur terpendek antara titik awal (*Player*) dan titik tujuan (*Musuh*) di tengah-tengah rintangan (*Obstacle*).



Gambar 3 Map Algoritma A* pada Enemy

Dalam kasus ini, kita bisa memberikan representasi nilai biaya (cost) pada setiap langkah dekat dengan titik awal. Berdasarkan aturan yang telah dijelaskan, kita dapat memberikan nilai 1.0 untuk setiap langkah vertikal atau horizontal, dan nilai 1.4 untuk setiap langkah diagonal. Ini dapat direpresentasikan sebagai berikut:

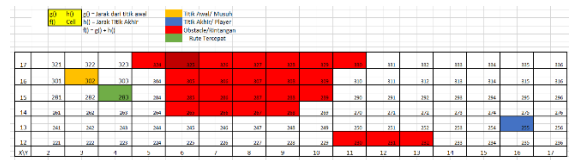
Langkah vertikal atau horizontal: Nilai $g = 1.0$

Langkah diagonal: Nilai $g = 1.41$

Nilai h = (Jarak X dari sel ke titik akhir + Jarak Y dari sel ke titik akhir).

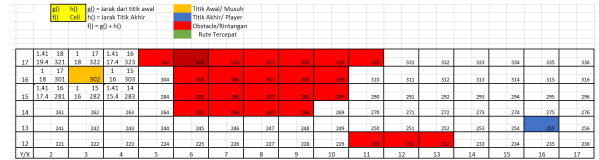
$$f(n)=g(n)+h(n)$$

Kita akan menyederhanakan peta untuk mempermudah perhitungan, hasilnya seperti berikut:



Gambar 4 Map algoritma A Star yang sudah disederhanakan

Perhitungan 1:



Gambar 5 Nilai Tiap Sel OpenList

Seperti pada diatas kita menghitung setiap sel yang kemungkinan akan dilalui musuh.

- Sel 303 pada kordinat (4,16) mempunyai nilai f sebesar 16.

$$g = 1$$

$$h = 3+12 = 15$$

$$f = 15+1 = 16$$

Masukkan nilai f ke OPEN list

- Sel 323 pada kordinat (4,17) mempunyai f sebesar 17,4.

$$g = 1,4$$

$$h = 4+12 = 16$$

$$f = 16+1,4 = 17,4$$

Masukkan nilai f ke OPEN list

- Sel 322 pada kordinat (3,17) mempunyai f sebesar 18.

$$g = 1$$

$$h = 4+13 = 17$$

$$f = 17+1 = 18$$

Masukkan nilai f ke OPEN list

- Sel 321 pada kordinat (2,17) mempunyai f sebesar 19,4.

$$g = 1,4$$

$$h = 4+14 = 18$$

$$f = 18+1,4 = 19,4$$

Masukkan nilai f ke OPEN list

- Sel 301 pada kordinat (2,16) mempunyai f sebesar 18.

$$g = 1$$

$$h = 3+14 = 17$$

$$f = 11+1 = 18$$

Masukkan nilai f ke OPEN list

- Sel 281 pada kordinat (2,15) mempunyai f sebesar 17,4.

$$g = 1,4$$

$$h = 2+14 = 16$$

$$f = 16+1,4 = 17,4$$

Masukkan nilai f ke OPEN list



Gambar 10. Mockup Halaman Menu Utama Aplikasi.

3.3 Tahap Pengumpulan Materi

Pada Tahap Pengumpulan Materi (*Material Collection*), seluruh aset yang dibutuhkan untuk keperluan game dikumpulkan secara sistematis sesuai analisis kebutuhan perangkat lunak. Proses ini mencakup pembuatan model 3D seperti karakter, musuh, dan rintangan menggunakan Blender, desain berbagai elemen antarmuka (UI), tombol, serta tekstur melalui Adobe Photoshop, serta pengumpulan aset audio mulai dari suara pengucapan huruf, musik latar, hingga efek suara yang akan meningkatkan pengalaman interaktif dalam aplikasi.

3.4 Tahap Pembuatan

Pada Tahap Pembuatan (*Assembly*), seluruh aset dari tahap pengumpulan materi dan rancangan desain diintegrasikan ke dalam proses implementasi yang dilakukan di dalam game engine Unity, di mana lingkungan permainan, arena, dan rintangan dibangun secara menyeluruh. Seluruh logika permainan, mekanisme interaksi pengguna, serta fungsionalitas antarmuka (UI) diimplementasikan menggunakan bahasa pemrograman C#. Khusus untuk pengendalian gerak musuh secara real-time, algoritma A* yang telah dirancang sebelumnya juga direalisasikan dalam bahasa C# sehingga musuh mampu melakukan pelacakan jalur secara dinamis mengikuti posisi pemain.

3.5 Tahap Pengujian

Pada tahap pengujian, proses dilakukan untuk menemukan kesalahan dan memastikan aplikasi berjalan sesuai dengan rancangan melalui metode Black Box Testing, yang difokuskan pada fungsionalitas input dan output tanpa menganalisis kode program. Pengujian ini meliputi beberapa aspek utama, yakni pengujian fungsional untuk memastikan semua tombol seperti Mulai, Tentang, Keluar, Kembali, dan Ulangi berfungsi sesuai rancangan activity diagram, pengujian gameplay yang bertujuan menilai respons kontrol joystick dan tombol menyerang saat digunakan pemain, serta pengujian algoritma untuk memvalidasi bahwa pergerakan musuh di dalam game setelah proses assembly benar-benar sesuai dengan hasil perhitungan manual yang telah dirancang pada tahap desain.

3.6 Tahap Distribusi

Pada tahap distribusi, aplikasi game edukasi yang telah melalui proses pembangunan, pengujian, dan validasi, didistribusikan secara langsung kepada anak-anak sebagai pengguna sasaran, baik di lingkungan sekolah mitra, kelompok belajar, maupun komunitas anak usia dini lainnya. Proses distribusi dilakukan secara manual dengan cara mentransfer berkas instalasi yang telah ditandatangani digital langsung ke perangkat Android anak,

baik melalui kabel data, media penyimpanan eksternal, atau bantuan pendamping guru/orang tua. Dengan pendekatan ini, peneliti dapat memastikan aplikasi tepat sampai di tangan target pengguna utama, sekaligus memperoleh feedback secara langsung dari anak dan pendamping terkait kemudahan instalasi, user experience, serta efektivitas pembelajaran.

4 HASIL DAN PEMBAHASAN

4.1 Hasil Pengembangan Aplikasi

Hasil pengembangan aplikasi yang telah dibuat yang sudah menerapkan algoritma A* pada game edukasi pengenalan huruf alphabet adalah sebagai berikut :



Gambar 11. Tampilan Splash Screen

Tampilan Splash Screen Pada Gambar 11, tampilan Splash Screen ini tampil ketika pertama kali aplikasi dibuka.



Gambar 12. Tampilan Halaman Menu Utama

Tampilan Halaman Menu Utama Pada Gambar 12, tampilan Halaman Menu Utama pada game terdapat 3 menu yaitu Mulai, Tentang, dan Keluar .



Gambar 13. Tampilan Halaman Tentang

Tampilan Halaman Tentang Pada Gambar 13, tampilan halaman tentang berisi terkait deskripsi tujuan aplikasi yang telah dibuat .



Gambar 14. Tampilan Memulai Permainan

Tampilan Memulai Permainan Pada Gambar 14, tampilan memulai permainan akan tampil ketika mengklik tombol Mulai. Pemain ditugaskan untuk mengalahkan monster yang berbentuk huruf alfabet .



Gambar 15. Tampilan Halaman Game Gagal

Tampilan Halaman Game Gagal Pada Gambar 15, tampilan game gagal akan tampil jika pemain kehabisan darah atau nyawa akibat serangan musuh .

4.2 Hasil Pengujian Sistem (Black Box)

Pengujian sistem ini dilakukan menggunakan pengujian blackbox. Hasil pengujian fungsionalitas utama menunjukkan bahwa semua fungsi berjalan Sesuai dengan harapan dan berstatus (✓) diterima.

Table 1 Pengujian Halaman Menu Utama

Kasus dan Hasil Uji (Data Benar) ✓			
Data yang dimasukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik mulai	Menampilkan halaman memulai permainan	Sesuai dengan harapan	(✓) diterima () ditolak
Klik tentang	Menampilkan halaman tentang	Sesuai dengan harapan	(✓) diterima () ditolak
Klik keluar	Menutup aplikasi	Sesuai dengan harapan	(✓) diterima () ditolak

Pengujian Halaman Menu Utama Tabel 1 yaitu Tombol Mulai, Tentang, dan Keluar berfungsi dengan benar.

Table 1 Pengujian Tentang Aplikasi

Kasus dan Hasil Uji (Data Benar) ✓			
Data yang dimasukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik kembali	Menampilkan halaman menu utama	Sesuai dengan harapan	(✓) diterima () ditolak

Pengujian Halaman Tentang Tabel 2 Tombol Kembali berfungsi dengan benar.

Table 2 Pengujian Memulai Permainan

Kasus dan Hasil Uji (Data Benar) ✓			
Data yang dimasukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengerakan Joystick	Player berubah posisi	Sesuai dengan harapan	(✓) diterima () ditolak
Klik tombol focus	Arah kamera player terfokus ke musuh	Sesuai dengan harapan	(✓) diterima () ditolak
Klik Tombol Menyerang	Player Menyerang	Sesuai dengan harapan	(✓) diterima () ditolak

Pengujian Memulai Permainan Tabel 3 yaitu Mengerakan Joystick, Klik tombol focus, dan Klik Tombol Menyerang semuanya berfungsi sesuai harapan.

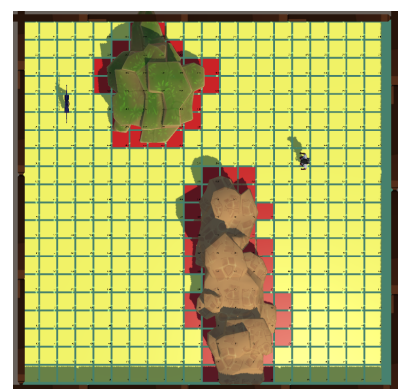
Table 4 Pengujian Game Gagal

Kasus dan Hasil Uji (Data Benar) ✓			
Data yang dimasukan	Yang diharapkan	Pengamatan	Kesimpulan
Klik tombol Mulai Ulang	Menampilkan memulai permainan	Sesuai dengan harapan	(✓) diterima () ditolak
Klik menu Kembali	Menampillkan halaman menu utama	Sesuai dengan harapan	(✓) diterima () ditolak

Pengujian Game Gagal Tabel 4 yaitu Tombol Mulai Ulang dan Kembali berfungsi sesuai harapan.

4.3 Hasil Pengujian Penerapan Algoritma A Star

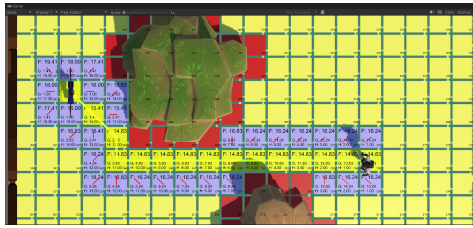
Arena pada pengujian dibuat berdasarkan arena sesungguhnya seperti gambar dibawah ini:



Gambar 16. Arena Game

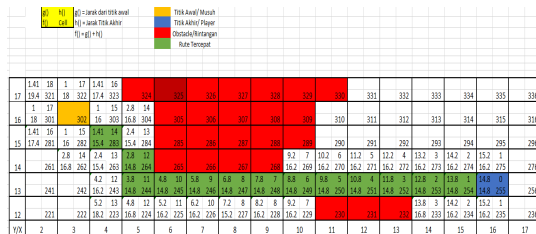
Arena pengujian untuk pathfinding menggunakan algoritma A* dirancang menyerupai kondisi nyata di dalam game, sebagaimana diperlihatkan pada Gambar 16. Papan grid dalam gambar tersebut menunjukkan area permainan, di mana warna merah

merepresentasikan rintangan atau obstacle, posisi musuh bertindak sebagai node awal sementara player sebagai node tujuan.



Gambar 17. Hasil path algoritma A Star

Dalam skenario pengujian, musuh dimulai dari sel 302 dan player pada sel 255. Saat simulasi dijalankan, algoritma A* menghitung dan menampilkan jalur optimal dari musuh menuju player dalam bentuk garis visual di arena, serta dapat dimonitor sebagai urutan node pada console Game engine seperti terlihat pada Gambar 17.



Gambar 18. Rute dan Perhitungan Manual Pathfinding Musuh Menuju Player

Hasil pengujian pada Tabel 5 memvalidasi bahwa rute yang dihasilkan oleh algoritma di dalam game sesuai dengan rute perhitungan manual yang telah dirancang ditampilkan pada Gambar 18.

Table 4.5 Pengujian Algoritma A Star

Titik Awal	Titik Akhir	Rute Jalur	Perhitungan Manual
302	255	283, 283, 264, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255	sel 283 pada kordinat (4,15) → sel 264 pada kordinat (5,14) → sel 244 pada kordinat (5,13) → sel 245 pada kordinat (6,13) → sel 246 pada kordinat (7,13) → sel 247 pada kordinat (8,13) → sel 248 pada kordinat (9,13) → sel 249 pada kordinat (10,13) → sel 250 pada kordinat (11,13) → sel 251 pada kordinat (12,13) → sel 252 pada

			kordinat (13,13) → sel 253 pada kordinat (14,13) → sel 254 pada kordinat (15,13) → sel 255 pada kordinat (16,13).
--	--	--	---

Analisis menunjukkan bahwa rute yang dilalui musuh pada perhitungan manual mencocokkan dengan jalur yang diambil oleh enemy dalam game. Ini membuktikan bahwa simulasi algoritma A* berhasil menentukan jalur terbaik dengan akurasi yang baik dan dapat dianggap sebagai solusi yang handal untuk pathfinding dalam game ini

5 KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan implementasi dan pengujian, maka dapat ditarik kesimpulan dari Game Edukasi Pengenalan Huruf Alphabet Menggunakan Algoritma A Star ini yaitu:

1. Dengan memanfaatkan teknik pathfinding dalam kecerdasan buatan dan menerapkan algoritma A*, pengembangan game edukasi pengenalan huruf alphabet menjadi lebih dinamis dan menantang. Algoritma A* digunakan untuk menentukan jalur terpendek musuh menuju pemain, menciptakan tingkat kesulitan yang dapat disesuaikan dalam permainan .
2. Pendekatan interaktif dan menarik dalam game edukasi ini dapat meningkatkan daya tarik dan efektivitas pembelajaran pada anak-anak usia dini. Anak-anak belajar melalui interaksi dengan musuh dalam game, menciptakan suasana belajar yang menyenangkan dan memotivasi, yang menggabungkan elemen pendidikan dan hiburan .

5.2 Saran

Implementasi game ini masih jauh dari sempurna sehingga perlu dilakukan perbaikan dan pengembangan. Beberapa saran untuk pengembangan lebih lanjut, yaitu:

1. Implementasikan kemampuan personalisasi pada permainan, di mana tingkat kesulitan dapat disesuaikan berdasarkan kemampuan individu pengguna. Hal ini dapat membantu memastikan bahwa setiap anak dapat belajar pada tingkat yang sesuai dengan kemampuannya .
2. Tambahkan lebih banyak level atau tingkat kesulitan yang berbeda untuk menjaga tingkat keasikan dan memberikan tantangan yang sesuai dengan kemajuan belajar pengguna, sehingga pengguna tidak merasa cepat bosan.

DAFTAR PUSTAKA

- Angeles, S., Arcena, L., Evalderosa, M., Infantado, J., Toribio, R., Faustino, J., Domingo, E., & Santos, J. (2025) 'Little bright box: Using interactive games as a tool to boost literacy skills in early childhood', *International Journal of Research and Innovation in Social Science*, 9(3), hlm. 123-135. <https://doi.org/10.47772/ijriss.2025.903sedu0300>
- Anwar, Y. & Thamrin, H. (2024) 'Comparison of A* algorithm with hierarchical pathfinding A* algorithm in 3D maze runner game', *PROCEEDINGS OF THE 8TH INTERNATIONAL CONFERENCE ON ENGINEERING, TECHNOLOGY, AND INDUSTRIAL APPLICATIONS 2021 (8th ICETIA 2021): Engineering, Environment, and Health: Exploring the Opportunities for the Future*. <https://doi.org/10.1063/5.0179630>
- Candra, A., Budiman, M. & Pohan, R. (2021) 'Application of A-Star Algorithm on Pathfinding Game', *Journal of Physics: Conference Series*, 1898, hlm. 012047. <https://doi.org/10.1088/1742-6596/1898/1/012047>
- Hu, Z., Xu, Q. & Huang, G. (2022) 'Discussion on Educational Games Based on Unity', *Proceedings of the 2022 6th International Conference on Education and E-Learning*. <https://doi.org/10.1145/3578837.3578847>
- Ilmianti, I., Mattulada, I., Aldilawati, S., Aslan, S., Febriany, M. & Hamka, M. (2021) 'Media komunikasi, informasi dan edukasi terhadap pengetahuan anak sekolah dasar tentang kesehatan gigi mulut', *Sinnun Maxillofacial Journal*, 2(01), hlm. 26-33. <https://doi.org/10.33096/smj.v2i01.52>
- Kusumawardhani, R., Prastiningtiyas, D. & Oktavia, C. (2025) 'The Fabrique: A Pathfinding Algorithm in a Mobile Game Developed Using Construct 3*', *Internet of Things and Artificial Intelligence Journal*. <https://doi.org/10.31763/iota.v5i2.915>
- Lawande, S., Jasmine, G., Anbarasi, J. & Izhar, L. (2022) 'A Systematic Review and Analysis of Intelligence-Based Pathfinding Algorithms in the Field of Video Games', *Applied Sciences*. <https://doi.org/10.3390/app12115499>
- Nursakti, A. & Winanda, A. (2025) 'Platform teknologi open source dalam pengembangan aplikasi edukasi mobile', *Jurnal Teknologi dan Informatika Terapan*, 2(1), hlm. 67-81.
- Pardede, S., Athallah, F., Huda, Y. & Zain, F. (2022) 'A Review of Pathfinding in Game Development', *[CEPAT] Journal of Computer Engineering: Progress, Application and Technology*, 1(1), hlm. 12-28. <https://doi.org/10.25124/cepat.v1i01.4863>
- Rafiq, A., Kadir, T. & Ihsan, S. (2020) 'Pathfinding Algorithms in Game Development', *IOP Conference Series: Materials Science and Engineering*, 769. <https://doi.org/10.1088/1757-899x/769/1/012021>
- Rahman, T., Ratnayake, Y. & Nurani, Y. (2024) 'Enhancing early childhood literacy through game-based interactive digital media development', *International Journal of Religion*, 7(2), hlm. 89-102. <https://doi.org/10.61707/7cprvh13>
- Rahmadi, L., Effendi, M. & Hasibuan, M. (2025) 'Interactive learning media for early childhood education through Android-based educational games using GDLC method', *PIKSEL: Penelitian Ilmu Komputer Sistem Embedded and Logic*, 13(1), hlm. 45-58. <https://doi.org/10.33558/piksel.v13i1.10129>
- Sapitri, M., Indihadi, D. & Rahman, T. (2023) 'Pengembangan media pembelajaran alphabet match dalam mengenalkan huruf pada anak usia dini', *Jurnal PAUD Agapedia*, 7(1), hlm. 25-32. <https://doi.org/10.17509/jpa.v7i1.59930>
- Sarifah, I., Rohmaniar, A., Marini, A., Sagita, J., Nuraini, S., Safitri, D., Maksum, A., Suntari, Y. & Sudrajat, A. (2022) 'Development of Android Based Educational Games to Enhance Elementary School Student Interests in Learning Mathematics', *Int. J. Interact. Mob. Technol.*, 16, hlm. 149-161. <https://doi.org/10.3991/ijim.v16i18.32949>
- Septianingsih, R. & Rahmawati, U. (2025) 'Penggunaan metode GDLC dalam perancangan game edukasi pengenalan alphabet dan angka untuk anak usia dini', *JATI (Jurnal Mahasiswa Teknik Informatika)*, 9(2), hlm. 3028-3034. <https://doi.org/10.36040/jati.v9i2.13253>
- Tang, J., Chu, S. & Chang, T. (2024) 'Enhancing English alphabet handwriting skills in preschool children through digital game-based learning approach', *Innovation in Language Learning and Teaching*, 18(2), hlm. 145-158. <https://doi.org/10.1080/17501229.2024.2352791>
- Triana, M., Sumardi, S. & Rahman, T. (2020) 'Pengembangan media big book alfabet untuk memfasilitasi kemampuan mengenal huruf alfabet anak usia 4-5 tahun', *Jurnal PAUD Agapedia*, 4(1), hlm. 24-38. <https://doi.org/10.17509/jpa.v4i1.27194>
- Winanda, A. (2025) 'Rancang bangun game edukasi pengenalan alfabet menggunakan augmented reality', *Jurnal Ilmiah Sistem Informasi Dan Teknik Informatika (JISTI)*, 8(1), hlm. 86-93. <https://doi.org/10.57093/jisti.v8i1.278>
- Zhang, C., Ao, L., Yang, J. & Xie, W. (2020) 'An Improved A* Algorithm Applying to Path Planning of Games', *Journal of Physics: Conference Series*, 1631. <https://doi.org/10.1088/1742-6596/1631/1/012068>