

Pengembangan Game Puzzle Find Grass Menggunakan Algoritma Backtracking

Yulyanto¹, Fikri Maudia Arsyad², Tito Sugiharto²

¹Fakultas Komputer, Desain Komunikasi Visual Univeritas Kuningan, Kuningan, Indonesia

²Fakultas Komputer, Teknik Informatika, Univeritas Kuningan, Kuningan, Indonesia

Email: ¹yulyanto@uniku.ac.id, ²maudiafikri@gmail.com, ³tito@uniku.ac.id

Email Penulis Korespondensi: yulyanto@uniku.ac.id

Abstrak— Game merupakan suatu sarana hiburan yang diminati dan dimainkan oleh banyak orang baik dari kalangan anak-anak, remaja maupun orang dewasa. Dengan perkembangan teknologi yang sangat pesat seperti sekarang ini, game menjadi semakin praktis. misalnya saja sekarang kita tidak perlu membeli *game console* seperti *Playstation* untuk memainkan *game*. Salah satu game yang mempunyai dampak baik bagi pengguna adalah game yang berbentuk edukasi, dari penjelasan diatas timbul ide untuk mengambil penelitian membuat game yang berjudul "Find Grass" yang berbasis android perspektif 3D dimana konsep permainan setiap level, user harus mengontrol player yaitu seekor kelinci yang mencari rumput untuk menuju ke titik tujuan akhir dengan waktu yang terbatas. Dalam pembuatan game ini membutuhkan *Algoritma Backtracking* yang dapat mencari jalur tercepat untuk sampai ke titik tujuan tertentu dan nantinya akan diterapkan ke sistem bantuan didalam game. Metode pengembangan sistem yang digunakan adalah GLDC (Game Development Life Cycle) dimana memiliki tahapan seperti initiation, pre- production, production, testing, beta, release. Untuk perancangan aplikasi menggunakan diagram UML (Unified Modelling Language). Selain itu aplikasi untuk membuat game menggunakan Unity3D. Hasil akhir dari aplikasi yang dibuat adalah game yang kompatibel pada perangkat smartphone yang bersistem operasi android minimal 5.0 (Lollipop). Game ini mempunyai beberapa menu diantaranya menu utama, menu stage yang terdiri dari 3 stage, menu panduan dan menu keluar.

Kata Kunci: *Game, Video Game, Smartphon, Algoritma Backtracking, Puzzle, Unity3D*

Abstract—Game like a tools of entertainment that is enjoyed and played by many grups ranging from children, teenagers and adults. With the rapid development of technology, games are becoming more practical. For example, nowadays we no need to buy a games console like *Playstation* to play a games. One of the games that has a good impact to users is a game in the form of education, from the explanation above, idea arises to take research to make a game entitled "FindGrass" based on 3D perspective android. That the concept of each level of the game, user must direct or control the player, namely a rabbit looking for a grass to get to the final destination poinwith limited time. In making this game requires a *Backtracking Algorithm* that can find the fastest path to get to a certain destination poin and later will be applied to the help system in the game. The system development method is GDLC (Game Development Life Cycle) which has stages such as initiation, pre-production, production, beta, release. For application design using UML (Unified Modeling Language). In addition, the application to create games uses Unity 3D. The final result of the application created is a game that is compatible on smartphone devices with an android operating system at least 5.0 (Lollipop). This game has several menus including the main menu, stage menu consisting of 3 stage guide menu and exit menu.

Keywords : *Game, Video Game, Smartphone, Backtracking Algorithm, Puzzle, Unity 3D*

1. PENDAHULUAN

Game merupakan jenis hiburan yang digemari oleh semua orang dari usia anak-anak, dewasa maupun tua. Selain digunakan untuk menghilangkan kebosanan dalam beraktivitas, sebuah permainan juga dapat berfungsi untuk melatih pola pikir seseorang untuk mencari solusi memecahkan suatu permasalahan yang ada di sebuah permainan. Dahulu permainan dimainkan secara tradisional seperti permainan kartu, catur, ular tangga, petak umpet, dan lainnya. Seiring dengan berkembangnya teknologi, permainan tersebut dikembangkan ke dalam teknologi yang lebih modern. Sekarang banyak permainan baru yang memanfaatkan teknologi modern dalam pembuatannya dan penggunaannya [1] Salah satu jenis yang dapat sering kita temui yaitu fuzzle dimana konsep permainannya adalah pada setiap level pemain harus mengarahkan player menuju titik akhir yang telah ditentukan serta harus menemukan sesuatu, memecahkan teka-teki disetiap level dan berusaha menghindari predator dengan waktu yang sudah ditentukan, solusi yang di pakai pada game ini yaitu dengan menggunakan algoritma backtracking. [2]

Algoritma Backtracking merupakan sebuah algoritma yang bisa digunakan untuk membuat aplikasi game karena cara kerjanya mencari jalan keluar yang bertujuan untuk menentukan jalur yang tepat untuk mencapai tujuan yang telah ditetapkan. Selama proses penentuan jalur tersebut, jika menemui jalan buntu maka akan dilakukan proses backtrack sampai kembali menemukan jalur yang tepat untuk mencapai tujuan. Algoritma backtracking pada persoalan ini dipandang sebagai pembentukan pohon ruang status. Akar dari pohon adalah labirin awal, dan anak-anaknya adalah labirin yang dihasilkan dari pergerakan satu langkah dari labirin semula. [3]

Penelitian terdahulu diantaranya tentang game *Treasure Of Labyrinth* Dengan menggunakan Algoritma Backtracking Berbasis android yang bertujuan mengumpulkan harta karun dan mengalahkan musuh didalam sebuah arena labyrinth serta mencari jalan keluar sebagai goal pada setiap levelnya. [4] Selanjutnya penelitian yang dilakukan oleh Muhamad

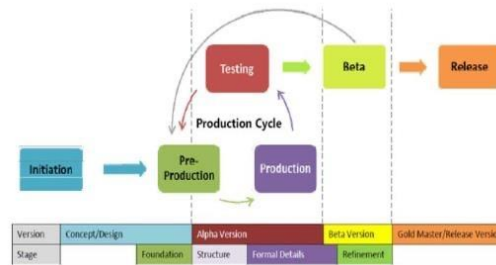
Anan dkk, yang membuat maze 3d dengan algoritma Backtracking dengan tujuan mencari jalan keluar yang bertujuan untuk menentukan jalur yang tepat untuk mencapai tujuan yang telah ditetapkan [5]. Sementara penelitian dari Baiq dkk, yang mengembangkan Game Puzzle Sebagai Edugame, aplikasi game puzzle ini berisi materi teori bilangan, aljabar, bangun datar dan bangun ruang dan dibuat dari Appsgeyser [6]

Pembuatan game dengan judul game “Find Grass” berbasis android dengan perspektif 3D menggunakan algoritma Backtracking ini memiliki konsep permainan yang berbeda dengan peneilitan terdahulu, pada game ini setiap level pemain harus mengarahkan seekor kelinci menuju titik akhir yang telah ditentukan serta harus menemukan rumput, memecahkan teka-teki disetiap levelnya dan berusaha menghindari predator dengan waktu yang terbatas agar bisa membuka level selanjutnya. Setiap kali pemain berhasil menyelesaikan maka level selanjutnya akan terbuka. Player juga akan bertemu dan berinteraksi dengan macam macam NPC (Non Player Character) berbentuk hewan.

2. METODOLOGI PENELITIAN

2.1 Metodologi Sistem Pengembangan

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah metode Game Development Life Cycle (GDLC). GDLC adalah suatu proses pengembangan sebuah game yang menerapkan pendekatan iteratif yang terdiri dari 6 fase pengembangan, dimulai dari fase ininitiation, preproduction, production, testing, beta dan realese.[7]



Gambar 1. Fase dan Proses GDLC

Algoritma Backtracking pertama kali diperkenalkan oleh D.H Lehmer pada tahun 1950. Algoritma ini cukup baik untuk digunakan dalam beberapa pemecahan masalah dan juga untuk memberikan kecerdasan buatan dalam permainan. Beberapa permainan populer seperti Sudoku, Labirin, Catur juga dapat diimplementasikan dengan menggunakan algoritma Backtracking. Algoritma Backtracking berbasis pada DFS (Depth First Search) sehingga aturan pencariannya akan mengikuti aturan pencarian DFS yaitu dengan mencari solusi dari akar ke daun (dalam pohon ruang solusi) dengan pencarian mendalam. Titik-titik yang sudah dibangkitkan (diperiksa) disebut simpul hidup (Live node). Simpul hidup yang sedang diperluas disebut simpul-E atau Expand Node [2]

2.2 Algoritma Backtracking

Langkah-langkah mencari solusi dengan algoritma backtracking adalah sebagai berikut: 1. Solusi dicari dengan membentuk jalur dari akar ke daun. Metode pencarian yang digunakan adalah pencarian dalam kedalaman (DFS). Simpul-simpul yang sudah dibangkitkan disebut simpul aktif, dan simpul aktif yang sedang diperluas disebut simpul ekspansi. Simpul diberi nomor sesuai dengan urutan pembangkitannya. 2. Jika jalur yang dibangkitkan tidak mengarah ke solusi, maka simpul ekspansi tersebut dihapus sehingga menjadi simpul mati. Simpul yang sudah mati tidak akan diperluas lagi. 3. Jika pembangkitan jalur terakhir berakhir dengan simpul mati, maka proses dilanjutkan dengan menghasilkan simpul anak lainnya. Jika tidak ada lagi simpul anak yang dihasilkan, maka pencarian solusi dilanjutkan dengan melakukan backtracking ke simpul aktif terdekat. Simpul ini kemudian menjadi simpul ekspansi yang baru. 4. Pencarian dihentikan jika telah menemukan solusi atau tidak ada lagi simpul aktif untuk dilakukan backtracking. [8]

2.3 Unity 3D

Unity 3D adalah sebuah aplikasi yang digunakan untuk mengembangkan game multiplatform yang dirancang untuk mudah digunakan. Unity 3D penuh dengan gabungan aplikasi yang profesional. editor pada Unity 3D dibuat dengan antarmuka pengguna yang sederhana. editor ini dibuat setelah ribuan jam yang telah dihabiskan untuk membuatnya menjadi nomor satu dalam urutan peringkat teratas untuk editor game. Grafik pada Unity dibuat dengan grafik tingkat tinggi untuk OpenGL dan directX. Unity mendukung semua format file, terutamanya format umum seperti semua format dari aplikasi seni. Unity kompatibel dengan versi 64-bit dan dapat beroperasi pada Mac OS x dan windows dan dapat menghasilkan permainan untuk Mac, Windows, Wii, iPhone, iPad dan Android [9].

2.4 C# (C Sharp)

Telah kita ketahui bersama bahwa Microsoft memiliki banyak sekali bahasa pemrograman seperti Visual Basic.NET, Visual Basic, Visual C++, Visual C#.NET, serta visual lain-lain yang dapat anda cari hingga kita bingung

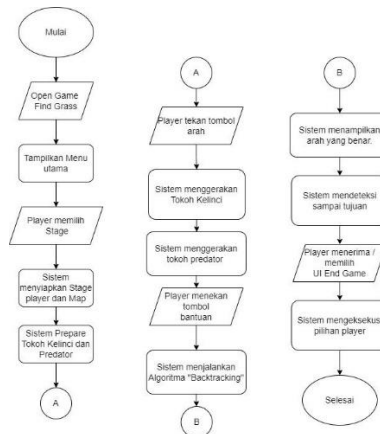
memilih mana yang terbaik. Untuk saya pribadi, mungkin akan lebih menekuni C#.NET ini ketimbang bahasa pemrograman lain seperti Visual Basic maupun C++. C# merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka .NET Framework. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain) dengan beberapa penyederhanaan.[10]

3. HASIL DAN PEMBAHASAN

3.1. Tahap PraProduksi

3.1.1 Flowchart System

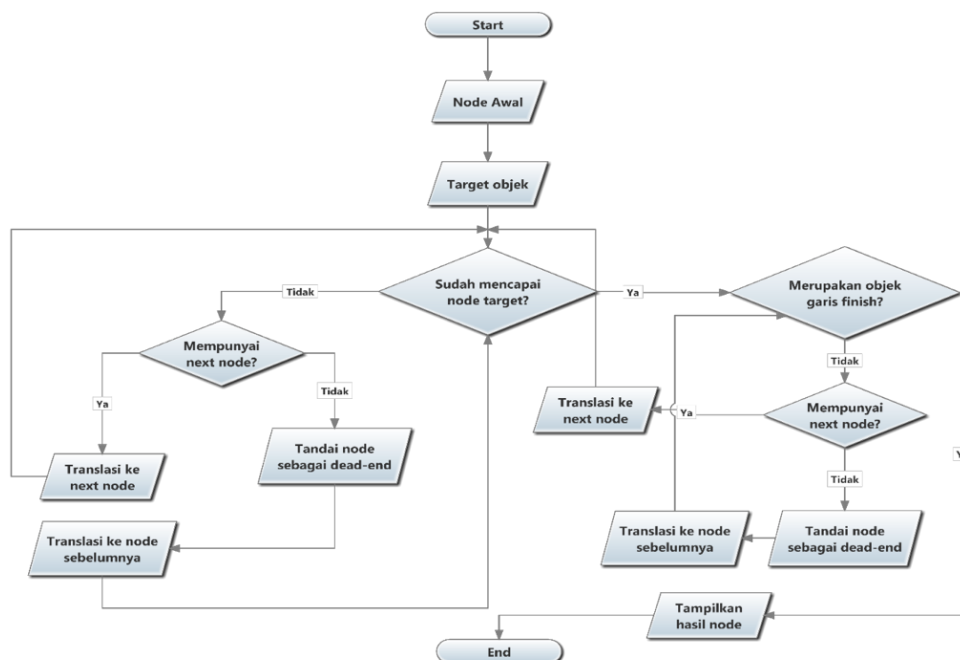
Setelah menganalisis sistem, maka didapatkan flowchart diagram sistem. Adapun alur kerja flowchart diagram sistem ditunjukkan oleh Gambar 2.



Gambar 2. Flowchart Alur Sistem

3.1.2 Flowchart algoritma Backtracking

Berikut merupakan flowchart algoritma backtracking untuk memecahkan permasalahan jalur terpendek, biasa digunakan untuk mengolah banyak jalur yang ditemukan lalu menyimpulkan mana jalur terpendek yang akan dilalui.



Gambar 3. Flowchart Algoritma Backtracking

Di mulai dari node awal kemudian Node melakukan pencarian menuju ke target objek, Kemudian apakah dalam pencarian sudah mencapai node target, jika ya maka sistem akan menampilkan node target. Jika objek yang ditemukan bukan garis finish maka proses pencarian target node masih terus berjalan, jika tidak menemukan node selanjutnya maka tandai node terakhir sebagai dead end, lalu bergeser ke node sebelumnya.

3.2. Tahap *Production* Objek Game Find Grass

1. Tokoh Player



Gambar 9. Tokoh Player/Kelinci

2. Predator



Gambar 10. Predator

3. Grass



Gambar 11. Rumput



Gambar 12. Halaman Utama

Halaman utama merupakan halaman awal yang ditampilkan pada saat pengguna membuka aplikasi game. Halaman utama menampilkan pilihan menu diantaranya menu pilih satge, menupanduan, dan menu keluar.



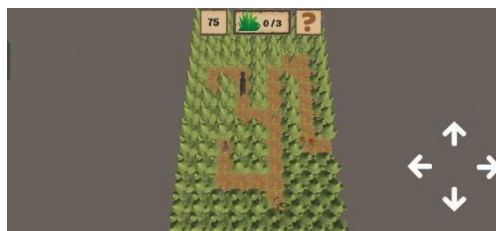
Gambar 13. Halaman Pilih Stage

Pada halaman ini terdapat 3 tombol stage dan tombol kembali yang berfungsi untuk ke menu utama.



Gambar 14. Halaman Panduan

merupakan menu untuk player bisamengetahui bagaimana cara memainkan dan fungsi menu-menu didalam game.



Gambar 15. Halaman Play Game

menampilkan tentang proses permainan game yang akan player hadapi, disini player harus mencari rumput dan menentukan jalur kelinci teraman dari predator dan terpendek untuk sampai tujuan atau finish.

4. KESIMPULAN

Dari penelitian yang telah dilakukan dapat disimpulkan bahwa algoritma *Backtracking* dapat membantu menemukan jalur terpendek yang akan dilalui oleh kelinci untuk mencari rumput dan mencapai tujuan. Perancangan Game menggunakan Unity Game Engine dengan menggunakan bahasa pemrograman C# dapat menghasilkan game edukasi dua dimensi (2D) yang menarik dan edukatif.

Pada pengujian Black box testing yang telah dibuat oleh peneliti menunjukkan penelitian ini telah sesuai dengan perancangan yang telah dikonsepsi sebelumnya yang terlihat dari hasil pengujian. Game ini layak sebagai media pembelajaran mengenal rantai makanan serta dapat meningkatkan perkembangan motorik anak dan melatih kontrol otot kecil, koordinasi mata, jari dan merangsang anak untuk lebih minat dalam belajar. Pada pengujian penerimaan menggunakan metode kuisioner yang dilakukan terhadap pengguna atau user hasil prosentase yang menyatakan setuju sebanyak 78%. Ini menunjukkan tingkat respon pengguna cukup tinggi untuk memainkan game ini.

UCAPAN TERIMAKASIH

Terima kasih disampaikan kepada pihak-pihak yang telah mendukung terlaksananya penelitian ini.

REFERENCES

- [1] S. F. Ayu and L. M. Sutardi, "RANCANG BANGUN GAME EDUKASI PUZZLE KEBUDAYAAN SULAWESI TENGGARA DENGAN ALGORITMA FISHER-YATES SHUFFLE".
- [2] B. Fahrudin, "Penerapan algoritma Backtracking pada Permainan capsabating," *JURIKOM (Jurnal Riset Komputer)*, vol. 3, no. 6, 2016.
- [3] R. A. K. Rio Andriyat Krisdiawan, H. B. Heru Budianto, T. S. Tata Sutabri, and A. K. Ade Kurniawan, "IMPLEMENTASI ALGORITMA LINEAR CONGRUENT METHOD (LCM) PADA MEDIA PEMBELAJARAN BAGIAN-BAGIAN BUNGA BERBASIS VIRTUAL REALITY (VR)," *NUANSA INFORMATIKA*, vol. 16, no. 2, pp. 94–105, 2022.
- [4] W. Diharjo, "Game Edukasi Bahasa Indonesia Menggunakan Metode Fisher Yates Shuffle Pada Genre Puzzle Game," *INTEGER: Journal of Information Technology*, vol. 5, no. 2, 2020.
- [5] A. Hoerudin, "GAME PUZZLE MAKANAN TRADISIONAL DENGAN MENGGUNAKAN ALGORITMA BLUM BLUM SHUB (BBS) SEBAGAI PENGACAKAN GAMBAR BERBASIS ANDROID," *NUANSA INFORMATIKA*, vol. 10, no. 2, 2016.
- [6] B. O. Aini, K. C. Ayu, and S. Siswati, "Pengembangan game puzzle sebagai edugame berbasis android untuk meningkatkan kemampuan berpikir matematika siswa sd," *JTAM (Jurnal Teori dan Aplikasi Matematika)*, vol. 3, no. 1, pp. 74–79, 2019.
- [7] R. A. Krisdiawan, R. Ramdoni, and A. Permana, "Rancang bangun game treasure of labyrinth dengan algoritma backtracking berbasis android," *Nuansa Informatika*, vol. 14, no. 1, pp. 46–55, 2020.
- [8] D. N. Putra, O. O. Sardjito, and C. Lawrence, "Penerapan dan Implementasi Algoritma Backtracking," *Published document accessed from*, 2005.
- [9] F. Hilviah, "Penerapan Algoritma Dynamic Programming dan Algoritma Backtracking pada Permasalahan Multiple Constraints Knapsack 0-1," 2015.
- [10] F. A. Jellyanto, I. Kuswardayan, and N. Suciati, "Rancang Bangun Pembangkit World Dinamis menggunakan Algoritma Recursive Backtracking pada Game 2D Platformer " Mine Meander", " *Jurnal Teknik ITS*, vol. 5, no. 2, pp. A591–A595, 2016.
- [11] R. D. I. Sari, "Analisis Penyelesaian Puzzle Sudoku Dengan Menerapkan Algoritma Backtracking," *Jurnal Ilmiah Teknologi dan Informasi Asia*, vol. 2, no. 2, pp. 1–18, 2008.
- [12] B. Fahrudin, "Penerapan algoritma Backtracking pada Permainan capsabating," *JURIKOM (Jurnal Riset Komputer)*, vol. 3, no. 6, 2016.
- [13] R. B. Sirait, "Perancangan Aplikasi Game Labirin Dengan Menggunakan Algoritma Backtracking," *Volume: V, Nomor*, vol. 2, 2013.
- [14] R. A. Krisdiawan, "Penerapan Model Pengembangan Gamegdlc (Game Development Life Cycle) Dalam Membangun Game Platform Berbasis Mobile," *Teknokom*, vol. 2, no. 1, pp. 31–40, 2019.
- [15] Y. Bouzid, M. A. Khenissi, F. Essalmi, and M. Jemni, "Using educational games for sign language learning-a signwriting learning game: Case study," *J Educ Techno Soc*, vol. 19, no. 1, pp. 129–141, 2016.