

Penerapan Algoritma A* (A Star) Pathfinding Pada *Game 3d Isometric Projection* "Survival Horror : Wabah Rabies"

Leonardo¹⁾, Ita Arfyanti²⁾, dan Muhammad Ibnu Sa'ad³⁾

^{1,2,3}Teknik Informatika, STMIK Widya Cipta Dharma
^{1,2,3}Jl. M. Yamin, Gn. Kelua, Kec. Samarinda Ulu, Kota Samarinda, Kalimantan Timur 75123
E-mail: leonardo.544099@gmail.com^{1), 2), 3)}

ABSTRAK

Rabies adalah penyakit fatal yang menyebabkan setidaknya 60.000 kematian setiap tahun. Penanganan rabies memerlukan pendekatan yang tepat, namun masih banyak kasus yang tidak ditangani secara optimal, menyebabkan tingkat kematian yang sangat tinggi. Pentingnya edukasi mengenai pencegahan dan penanganan rabies mendorong pengembangan permainan edukatif "Survival Horror Wabah Rabies". Permainan ini dirancang untuk menyampaikan informasi tentang penularan rabies, pencegahannya, serta tindakan yang harus diambil jika terkena gigitan hewan yang dicurigai terinfeksi. Dalam permainan, pemain berperan sebagai seorang polisi yang bertugas mengirimkan serum dan vaksin rabies ke rumah warga yang terinfeksi, sambil menghadapi serangan dari hewan yang terinfeksi. Algoritma A* (A Star) Pathfinding diterapkan untuk meningkatkan pengalaman bermain dan menantang dalam permainan, memungkinkan musuh untuk mengejar pemain dengan jalur tercepat. Hasil pengujian menunjukkan bahwa 60% pemain menyatakan permainan ini baik, menunjukkan bahwa *game* ini efektif sebagai media edukasi dan hiburan. Penerapan algoritma A* berhasil meningkatkan kualitas permainan, sementara elemen edukasi tetap disampaikan dengan jelas dan menarik.

Kata Kunci: permainan, A Star, Pathfinding, Unity.

Application of A (A Star) Pathfinding Algorithm in 3D Isometric Projection Game "Survival Horror : Wabah Rabies"*

ABSTRACT

Rabies is a deadly disease that causes at least 60,000 deaths each year. Treating rabies requires the right approach, but there are still many cases that are not optimally treated, resulting in a very high mortality rate. The importance of education on rabies prevention and treatment prompted the development of the educational game "Survival Horror: Wabah Rabies". The game is designed to convey information about rabies transmission, its prevention, and the actions to be taken if bitten by a suspected infected animal. In the game, the player assumes the role of a police officer tasked with delivering rabies serum and vaccine to the homes of infected citizens, while facing attacks from infected animals. The A (A Star) Pathfinding algorithm is applied to enhance the gameplay experience and challenge in the game, allowing the enemy to pursue the player with the fastest path. The test results showed that 60% of players rated the game as good, indicating that the game is effective as an educational and entertainment medium. The application of the A* algorithm successfully improves the quality of the game, while the educational elements are still clearly and interestingly conveyed.*

Keywords: Game, A star, Pathfinding, unity.

1. PENDAHULUAN

Menurut Nadeem dan Panda (2020), rabies adalah penyakit fatal yang terhitung setidaknya 60.000 kematian per tahun. Pendekatan paliatif atau agresif diperlukan untuk pasien rabies yang dicurigai atau dikonfirmasi. Namun, sebagian besar kasus tidak diobati dengan pola pikir rumah sakit bahwa penyakit ini memiliki tingkat kematian 100%. Dengan berlalunya waktu, beberapa dokter telah mencoba intervensi untuk membuat mereka bertahan hidup. Akibatnya, ada beberapa penyintas rabies yang terdokumentasi dengan baik sampai sekarang. Sebelum 2004, hanya lima kasus yang selamat yang menerima PEP (*Post-Exposure Prophylaxis*) tidak

lengkap. Idealnya, PEP suatu program vaksinasi harus dimulai segera setelah gigitan hewan segera setelah mencuci semua luka dengan sabun dan air, sehingga *viral load* dapat dikurangi di lokasi inokulasi. Penyebab kegagalan PEP yang paling umum adalah (1) kurangnya penggunaan imunoglobulin rabies, (2) tidak semua luka disuntik dengan imunoglobulin, (3) penundaan profilaksis 6 hari, (4) penjahitan luka sebelum injeksi imunoglobulin, dan (5) luka di daerah tubuh yang sangat dipersarafi seperti wajah dan tangan. Alasan kegagalan PEP adalah karena semua alasan keenam poin diatas.

Edukasi tentang bahaya virus rabies pada usia remaja perlu dilakukan oleh orang tua kepada anak remaja sejak

dini tentang hewan yang berisiko tinggi menularkan rabies adalah hewan liar atau hewan peliharaan yang tidak mendapatkan vaksin rabies, hewan yang terinfeksi bisa menularkan virus penyakit rabies melalui air liur, gigitan, atau cakaran dan jilatan pada kulit seseorang yang terluka. Namun mengedukasi anak remaja tentang virus rabies bukanlah perkara mudah, mulai dari memberi pengertian tentang apa itu virus rabies, contoh hewan yang terjangkit rabies, pencegahan virus rabies, dan tindakan pertama pada korban yang tergigit hewan terjangkit rabies seperti yang dihimbau oleh tenaga kesehatan dan Badan Kesehatan Dunia (WHO) namun tetap saja tidak memberikan informasi yang efektif kepada anak remaja.

Survival horror merupakan genre yang umum orang pilih untuk lari dari kebahagiaan duniawi. *Game* Survival horror menawarkan ketegangan tiada henti, sembari terus mendorong pemain untuk mencari jalan agar bisa bertahan hidup dan selamat hingga akhir permainan. Sesuai dengan genre sampingannya yaitu survival alias bertahan hidup dibuat dengan penambahan unsur edukasi, dengan adanya *game* edukasi yang dibuat akan membuat anak remaja lebih mudah dalam memahami sebuah pesan informasi yang diberikan dengan keseruan yang menarik.

Game yang dirancang adalah *game* yang berjudul "Survival Horror : Wabah Rabies". Untuk membuat *game* edukasi maka akan dirancang untuk memberikan pemahaman tentang cara-cara untuk menghindari penularan dan tindakan apa yang harus dilakukan jika terkena gigitan hewan yang dicurigai terjangkit rabies, algoritma A* diprogram untuk memandu musuh untuk mengejar pemain dengan mencari jalur terpendek ke lokasi pemain berada dan mengambil tindakan yang tepat untuk memburu lalu menyerang.

Hadirnya *game* ini mencakup elemen-elemen interaktif seperti informasi tambahan tentang penyakit rabies, cara-cara penularannya, dan tindakan pencegahan yang dapat diambil, anak remaja sebagai target pemain mendapat pengetahuan tentang bahaya rabies dan memberikan pemain keterampilan praktis untuk mengatasi situasi yang mungkin terjadi dalam kehidupan nyata ketika terkena gigitan hewan terinfeksi virus rabies. Solusi pencegahan virus rabies dan manfaat yang ditawarkan dari *game* edukasi yang dibuat akan menghasilkan pemikiran di masyarakat tentang bahayanya virus rabies ditambah kesadaran akan pentingnya pencegahan virus rabies dengan vaksinasi hewan peliharaan kesayangan.

Dalam *game* ini menceritakan seorang polisi yang sedang bertugas untuk mengirimkan serum dan vaksin rabies ke rumah warga yang terkena gigitan hewan rabies. Untuk membuat *game* menjadi lebih menarik dan tidak membosankan dimainkan oleh *player*, maka *player* dibuat dengan menggunakan senjata penembak suntik vaksinasi anti rabies untuk melawan virus yang

menginfeksi hewan yang nanti akan mengejar si *player* sampai kapanpun, maka *player* harus menembakkan suntik vaksinasi anti rabies tersebut ke hewan terinfeksi, *player* ditantang untuk bertahan dan menghindari dari serangan hewan terinfeksi yang ada di kota tersebut. *player* memiliki misi untuk memberi suatu *objective* yaitu serum dan vaksin rabies. selain itu, *player* juga mendapatkan senjata untuk membantu *player* bertahan hidup sampai berhasil mengantar semua *objective* untuk menyelesaikan permainan.

2. RUANG LINGKUP

2.1 Rumusan Masalah

Berdasarkan latar belakang di atas maka rumusan masalahnya adalah "Bagaimana penerapan algoritma A* (A star) Pathfinding pada *game* 3D *isometric projection* Survival Horror : wabah rabies?"

2.2 Batasan Masalah

1. Berbasis *desktop* pada *platform personal computer*.
2. *Game* dibuat untuk *Single player* dan merupakan *game offline*.
3. *Game* dijalankan pada sistem operasi Windows 10 dan di atasnya.
4. Genre Survival Horror dengan grafis 3D.
5. Dalam *game* tidak menggunakan fitur *save progress*.
6. Bahasa yang digunakan untuk menu utama menggunakan bahasa indonesia.
7. *Game* hanya terdiri dari dua level.

2.3 Tujuan Penelitian

Tujuan penelitian dalam pembuatan *game* edukasi tentang virus rabies adalah untuk menyampaikan informasi yang tepat, meningkatkan pemahaman, dan mempromosikan tindakan pencegahan yang efektif terhadap penyakit serta meningkatkan kesadaran dan pemahaman masyarakat secara umum.

2.4 Manfaat Penelitian

1. Bagi STMIK WICIDA

Sebagai penambah literatur pustaka di perpustakaan STMIK WICIDA dan dapat digunakan sebagai referensi atau kajian untuk mahasiswa lain yang ingin mengembangkan lebih lanjut terkait penerapan algoritma A* (A Star).

2. Bagi Mahasiswa

Manfaat bagi mahasiswa yaitu sebagai penambah ilmu dan menambah wawasan serta dapat dimanfaatkan sebagai pedoman dan sumber bacaan tambahan untuk penelitian lain dengan studi kasus dan metode yang berbeda.

3. Bagi Masyarakat

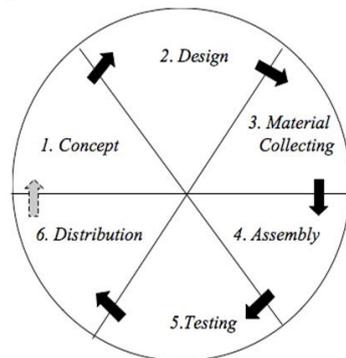
Manfaat untuk masyarakat yaitu pembuatan *game* ini untuk mengedukasi tentang bahaya virus rabies, dengan mengusung tema Survival dan Horror maka permainan akan lebih seru dan penuh tantangan.

3. BAHAN DAN METODE

3.1 *Multimedia Development Life Cycle*

Menurut Sugiarto dalam (Binanto, 2010) menjelaskan bahwa, MDLC adalah metode yang tujuannya untuk

merancang dan mengembangkan suatu aplikasi yang menggabungkan video, gambar, audio, animasi, dan lainnya. Metode MDLC memiliki 6 tahap antara lain, konsep (*concept*), perancangan (*design*), pengumpulan materi (*material collecting*), pembuatan (*assembly*), pengujian (*testing*), dan pendistribusian (*distribution*). Keenam tahapan tersebut tidak harus berurutan dalam praktiknya, setiap tahap dapat saling berganti posisi. Meskipun tahap pengonsepan memang harus menjadi hal yang paling pertama dilakukan. Adapun tahapan metode *Multimedia Development Life Cycle* (MDLC) sebagai berikut;



Gambar 1. Tahap pengembangan MDLC
Sumber : Sugiarto dalam (Binanto, 2010)

1. Konsep (*Concept*) adalah tahap untuk menentukan tujuan dan pengguna aplikasi. Selain itu juga pada tahap ini akan menentukan jenis aplikasi (presentasi, interaktif, dan lain-lain). Tujuan pengguna akhir program berpengaruh pada nuansa multimedia pencerminan dari identitas organisasi yang menginginkan informasi sampai pada pengguna akhir. Karakteristik pengguna termasuk kemampuan pengguna juga perlu di pertimbangkan karena dapat mempengaruhi pembuatan desain. Dasar aturan dalam perancangan aplikasi ditentukan pada tahap ini, misalkan target *platform*, ukuran aplikasi dan lain-lain. *Output* dari tahap ini biasanya berupa dokumen yang bersifat naratif untuk mengungkapkan tujuan proyek yang ingin di capai.
2. Desain (*Design*) adalah tahap pembuatan spesifikasi mengenai arsitektur program, gaya, tampilan, dan kebutuhan material atau bahan yang akan digunakan dalam pembuatan aplikasi. Pada tahap ini biasanya menggambarkan alur dari setiap *scene* untuk menggambarkan alur dari satu *scene* ke *scene* lain. Spesifikasi dibuat serinci mungkin sehingga pada tahap berikutnya, yaitu *material collecting* dan *assembly*, pengambilan keputusan baru tidak perlukan lagi, cukup menggunakan keputusan yang sudah ditentukan pada tahap ini. Meskipun demikian, pada prakteknya pekerjaan proyek pada tahap awal masih akan sering mengalami penambahan bahan atau pengurangan bagian aplikasi atau perubahan-perubahan lainnya.
3. Pengumpulan Materi (*Material Collecting*) adalah tahap pengumpulan bahan yang sesuai dengan kebutuhan yang digunakan. Bahan-bahan tersebut antara lain dapat

berupa gambar, foto, animasi, video, audio, serta teks baik yang sudah jadi atau pun yang masih perlu di modifikasi sesuai dengan kebutuhan yang ada. Bahan- bahan tersebut dapat di dapatkan secara gratis, membuatnya sendiri atau membelinya pada situ-situ penyedia *file-file* yang akan digunakan dalam pembuatan aplikasi *game*. Tahap ini dapat dikerjakan secara paralel dengan tahap *assembly*. Namun, pada beberapa kasus, tahap *material collecting* dan tahap *assembly* akan dikerjakan secara linier dan tidak paralel.

4. Pembuatan (*Assembly*) adalah tahap pembuatan keseluruhan aplikasi *game*. Aplikasi yang akan dibuat didasarkan pada tahap *design*. Tahap ini biasanya menggunakan perangkat lunak sesuai dengan kebutuhan.
5. Pengujian (*Testing*) Pada tahap ini bertujuan untuk memastikan bahwa aplikasi yang dikembangkan bebas dari kesalahan-kesalahan. Pada penelitian ini aplikasi akan diuji kepada pengguna dengan memberikan pertanyaan-pertanyaan mengenai *game* yang telah dibangun dan informasi yang terkait dalam aplikasi. Hasil pengujian tersebut diisi setelah pengguna memainkan *game* tersebut.
6. Distribusi (*Distribution*) Tahap ini adalah tahap terakhir dalam siklus pengembangan aplikasi multimedia. Pendistribusian dapat dilakukan setelah aplikasi dinyatakan layak pakai. Pada tahap ini juga dapat disebut tahap untuk evaluasi untuk pengembangan produk yang sudah jadi agar produk menjadi lebih baik lagi dari pada sebelumnya.

3.2 Teknik Pengumpulan Data

Untuk menjelaskan dan memudahkan dalam rangkaian penelitian menghindari terjadinya kesalahan, maka metode penelitian yang diterapkan peneliti di dasarkan pada metode-metode penelitian yang sudah umum, yaitu:

1. Studi Pustaka
Dalam melakukan penelitian menggunakan literatur, jurnal, dan skripsi yang berhubungan dengan judul, landasan teori dari sistem yang akan di kembangkan.
2. Wawancara
Wawancara merupakan salah satu teknik pengumpulan data, di mana secara langsung melakukan tanya jawab dengan narasumber yang berkaitan dengan data yang di perlukan.
3. Kuesioner
Kuesioner suatu teknik pengumpulan informasi dengan menggunakan kuesioner, analisis berupa mengukur apa yang di temukan dalam wawancara, selain itu juga menentukan berapa luas atau batasnya sentimen yang diekspresikan dalam suatu wawancara.

3.3 Algoritma A* (A star)

Menurut Wahyuni dan Mantja dalam (Suyanto, 2011), Algoritma A* didesain untuk pencarian jalur dari suatu titik ke titik lain. Menggunakan konsep *graph* dimana terdapat kumpulan *node*, yang merepresentasikan titik asal, tujuan, serta area untuk pencarian, dan *edge*, yang

yang merepresentasikan jalan penghubung antar *node*. Proses pencarian jalur dimulai dengan menerima *input* berupa *node* asal dan *node* tujuan setelah itu dilakukan pencarian rute dengan algoritma A*. *Output* yang dihasilkan ada dua kemungkinan yaitu ada rute atau tidak ada rute dari *node* asal ke *node* tujuan.

Algoritma A* (A Star) merupakan algoritma Best First Search yang menggabungkan Uniform Cost Search dan Greedy Best-First Search. *Cost* yang diperhitungkan didapat dari *cost* sebenarnya ditambah dengan *cost* perkiraan. Dengan perhitungan *cost* seperti ini, algoritma A* (A Star) dapat optimal dan lengkap dalam menentukan jalur. Sama dengan algoritma dasar Best First Search, algoritma A*(A Star) ini juga menggunakan dua senarai: *OPEN* dan *CLOSED*. Terdapat tiga kondisi bagi setiap suksesor yang dibangkitkan, yaitu: sudah berada di *OPEN*, sudah berada di *CLOSED*, dan tidak berada di *OPEN* maupun *CLOSED*. Pada ketiga kondisi tersebut diberikan penanganan yang berbeda-beda.

3.4 Pathfinding

Menurut Barnouti, dkk (2016), Pathfinding atau pencarian jalur dalam *game* komputer telah dilakukan riset selama bertahun-tahun. Pathfinding dalam *game* komputer komersial harus diselesaikan secara *real-time*, pencarian jalur tersebut dapat dilakukan dengan algoritma.

3.5 Pengujian White Box Testing

Menurut Hasanah (2020), *White Box Testing* merupakan metode pengujian perangkat lunak di mana struktur internal diketahui untuk menguji siapa yang akan menguji perangkat lunak. Pengujian ini membutuhkan pengetahuan internal tentang kemampuan sistem dan pemrograman.

Banyaknya *path* atau jalur pengujian yang dapat diambil disebut dengan kompleksitas siklomatik yang dapat dicari dengan menggunakan rumus :

$$\begin{aligned} V(G) &= E - N + 2 \\ V(G) &= \text{Kopleksitas siklomatik} \\ E &= \text{Jumlah edge grafik alir} \\ N &= \text{Jumlah node grafik alir} \end{aligned}$$

3.6 Pengujian Beta Testing

Menurut Antonius (2017), beta testing adalah pengujian yang dilakukan pada satu atau lebih pelanggan oleh pemakai akhir perangkat lunak. Tidak seperti pengujian *alpha*, pengembang biasanya tidak ada sehingga pengujian beta merupakan sebuah aplikasi "*live*" dari perangkat lunak didalamnya suatu lingkungan yang tidak dapat dikontrol oleh pengembang. Pelanggan merekam semua masalah (real atau imajiner) yang mereka temui selama pengujian beta melaporkan kepada pengembang dalam interval yang *leguler*.

Rumus Persentase yang digunakan adalah sebagai berikut :

$$Y = P/Q * 100\%$$

Keterangan :

Y = Nilai Persentase

P = Banyaknya Jawaban

Q = Jumlah Responden

4. PEMBAHASAN

Hasil akhir yang dapat diperoleh dari penelitian ini adalah pembuatan *Game Survival Horror* : Wabah Rabies dikemas menjadi suatu aplikasi yang dapat dimainkan pada *platform* Windows 10 dan di atasnya.

Tahapan dalam pembuatan aplikasi ini dimulai dari tahap pertama (konsep) sampai dengan tahap terakhir (distribusi), dan akan dijelaskan sebagai berikut:

4.1 Konsep

Di butuhkan *concept* yaitu mengenai tahapan-tahapan permainan seperti tujuan permainan, indentifikasi permainan, spesifikasi umum permainan, ukuran permainan dan jenis permainan. Yang di mana *concept* ini akan dijadikan suatu permainan yang ber-*genre* Survival Horror game. Dalam tahap ini diambil empat hal yang akan dijadikan *concept* dalam membangun "Survival Horror : Wabah Rabies" yaitu ;

1. Tujuan permainan

Berisi penjelasan tentang tujuan dibangun "Survival Horror : Wabah Rabies" yang mendefinisikan ruang lingkup yang terkait batasan fungsi permainan.

2. Jenis permainan

Berisi penjelasan tentang jenis dari *game* Survival Horror : Wabah Rabies yang menjelaskan apa saja yang dapat dilakukan pemain pada saat bermain *game* ini.

3. Indentifikasi penggunaan permainan

Berisi penjelasan tentang indentifikasi *audiens* prabuatan *game* Survival Horror : Wabah Rabies yang menjelaskan ruang kemungkinan yang terkait pada yang akan menggunakan permainan tersebut.

4. Spesifikasi umum

Berisi penjelasan tentang spesifikasi umum ataupun kebutuhan sistem yang ada dalam pembuatan *game* Survival Horror : Wabah Rabies yang di dalamnya juga akan membahas tentang ukuran , dasar perancangan, dan target yang ingin dicapai pemain.

1. Perangkat keras (*hardware*). Perangkat keras yang digunakan pada pembuatan *game* Survival Horror : Wabah Rabies yaitu ;

1) *Laptop* Aspire 5 A515-45

2) *Processor* AMD Ryzen 3 5300U 2.6GHz

3) Memori (RAM) 16 GB DDR4

4) SSD 500 GB

5) HDD 1 TB

6) VGA AMD Radeon Graphics

2. Perangkat lunak (*Software*). Perangkat lunak yang digunakan dalam pembuatan *game* yaitu ;

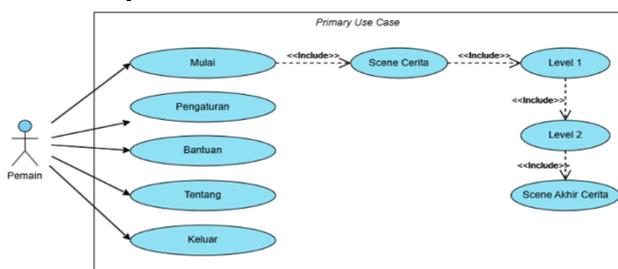
1) Sistem Operasi Windows 10 (64 Bit)

- 2) Canva
- 3) Microsoft Visual Studio Code
- 4) Unity Engine
- 5) Balsamiq Mockups
- 6) Audacity

4.2 Desain

Dalam mengembangkan game, dibutuhkan *Unified Modelling Language* (UML) untuk mengetahui alur atau jalan *game* yang akan dibuat. *Unified Modelling Language* dimulai dari aplikasi *game* dibuka sampai dengan mengakhiri *game*. Terdapat 4 diagram yang digunakan pada sistem, yakni *Flowchart*, *Activity Diagram*, *Primary Use Case*, dan *Sequence Diagram*.

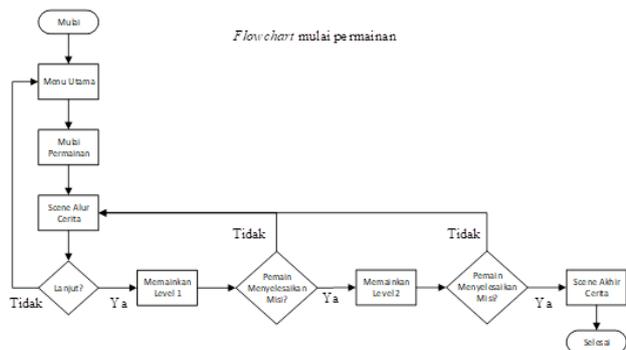
4.2.1 Primary Use Case



Gambar 2. Primary Use Case

Dari gambar 2, *Primary use case* terdapat pengaturan digunakan untuk mengatur hidup atau mati suara latar belakang serta mengatur tampilan layar penuh permainan. Bantuan menampilkan informasi petunjuk dan kontrol pada *game*. Tentang menampilkan informasi pengembang permainan. Sedangkan keluar berfungsi untuk keluar dari permainan. Mulai digunakan untuk bermain *game*. *Scene* cerita menampilkan informasi mengenai rabies, alur cerita, dan alur bermain permainan. Level 1 adalah tingkatan pertama di mana terdapat enam misi yang harus diselesaikan dengan tingkat kesulitan normal. Selanjutnya level 2 adalah tingkatan kedua memiliki sejumlah dua misi yang harus diselesaikan serta melawan musuh besar yang sulit dikalahkan. *Scene* akhir cerita menampilkan ciri-ciri rabies dan hewan penyebab rabies serta menampilkan informasi berita fiksi sumber wabah rabies sebagai penutup cerita.

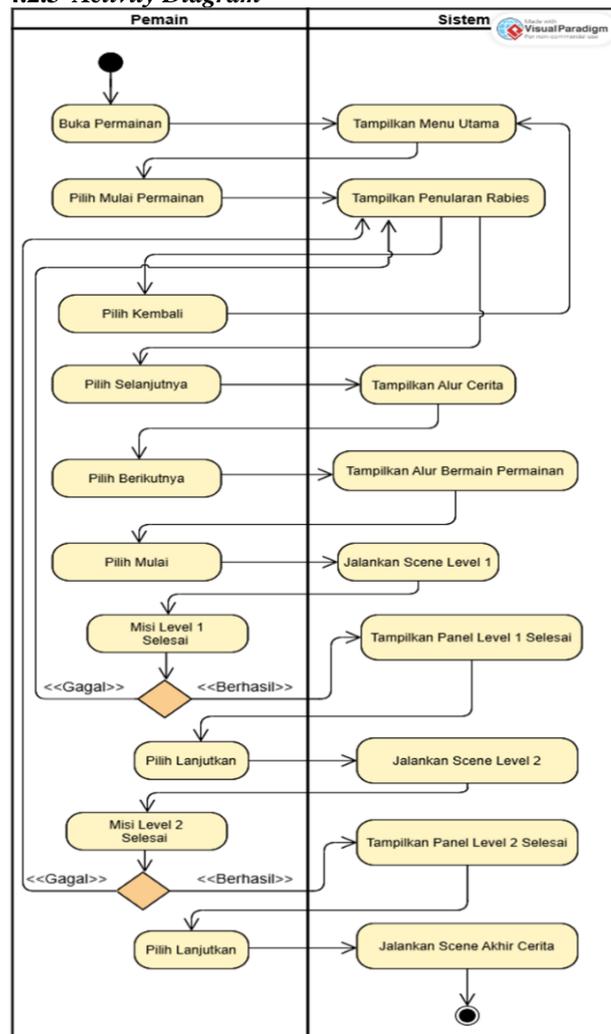
4.2.2 Flowchart



Gambar 3. Flowchart Mulai Permainan

Berdasarkan gambar 3, dari menu utama pemain menekan tombol mulai permainan akan masuk alur cerita, jika pemain menekan Tidak akan kembali menuju menu utama sedangkan menekan Ya lanjut untuk memainkan level 1, lalu pada level 1 apakah pemain menyelesaikan misi?, jika Tidak (gagal) kembali ke *Scene* alur cerita sedangkan jika Ya (sukses) lanjut memainkan level 2, lalu pada level 2 apakah pemain menyelesaikan misi?, jika Tidak (gagal) kembali ke *Scene* alur cerita sedangkan jika Ya (sukses) lanjut masuk *Scene* akhir cerita.

4.2.3 Activity Diagram



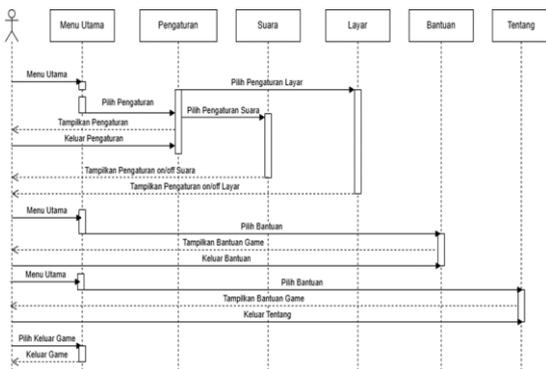
Gambar 4. Activity Diagram

Didalam permainan Survival Horror : Wabah Rabies terdapat *activity diagram* yaitu *activity diagram* dalam memulai permainan dapat diuraikan sebagai berikut :

1. Pemain atau *user* membuka permainan Survival Horror : Wabah Rabies.
2. Pada menu utama, pemain memilih mulai permainan membuka *scene* cerita.
3. Pada *scene* cerita, pemain memilih menu melanjutkan untuk berganti panel tampilan cerita.

4. Dari panel cerita, jalankan *scene* untuk memulai bermain permainan pada level satu.
5. Pada level satu, jika pemain gagal menyelesaikan misi maka menuju *scene* cerita pada panel penularan rabies, jika pemain berhasil menyelesaikan misi maka dilanjutkan bermain pada level dua.
6. Pada level dua, jika pemain gagal menyelesaikan misi maka menuju *scene* cerita pada panel penularan rabies, pemain berhasil menyelesaikan misi maka menuju *scene* akhir cerita.

4.2.4 Sequence Diagram

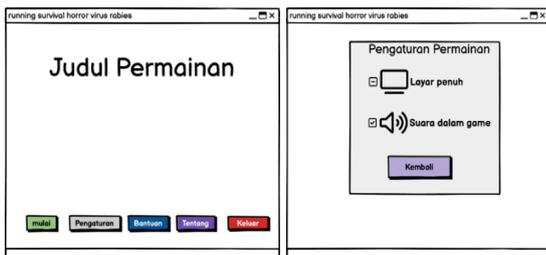


Gambar 5. Sequence Diagram Pengaturan

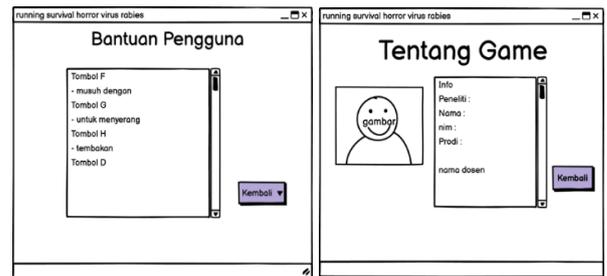
Pada gambar 5, Untuk masuk ke menu pengaturan *user* harus masuk aplikasi *game* kemudian masuk menu utama pilih menu pengaturan di dalam panel menu pengaturan ini terdapat 2 *checkbox* untuk matikan dan hidupkan suara latar belakang dan pengaturan mode layar penuh atau mode jendela. Jika pemain memilih menu tentang maka akan ditampilkan profil pengguna pada menu tentang juga menampilkan nama dosen pembimbing dan jika pemain memilih menu bantuan maka akan ditampilkan informasi petunjuk dan kontrol pada *game*. Pemain memilih keluar *game* untuk menutup permainan.

4.2.5 Desain Permainan

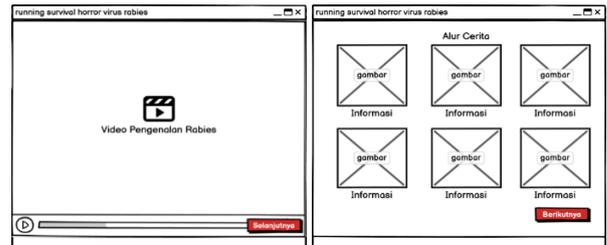
Dalam pembuatan *game* Survival Horror : Wabah Rabies ini semua desain dimulai dari perancangan antar muka. Antar muka perangkat lunak dirancang sesederhana mungkin sehingga pengguna diharapkan tidak mendapatkan kesulitan dalam menjalankan *game* ini.



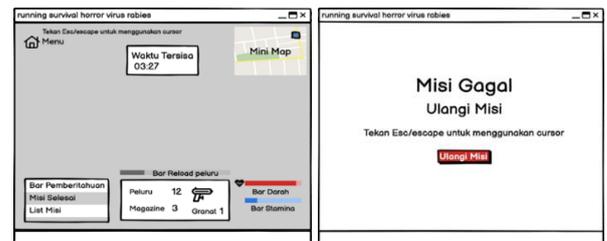
Gambar 6. Desain menu utama dan pengaturan



Gambar 7. Desain menu bantuan dan tentang



Gambar 8. Desain Panel Video dan Alur Cerita



Gambar 9. Desain Heads Up Display dan Kalah



Gambar 10. Desain Menang dan Splash screen

4.3 Pengumpulan Data

Pengumpulan data merupakan tahap ketiga adalah tahap pengumpulan bahan-bahan materi digunakan untuk mendukung dalam pembuatan *game* Survival Horror : Wabah Rabies, dimulai dari gambar, video, Audio, dan objek model 3D yang dimana berfungsi sebagai objek *asset* pembentukan dunia *game*, musik latar belakang, media alur cerita, dan musik tombol.

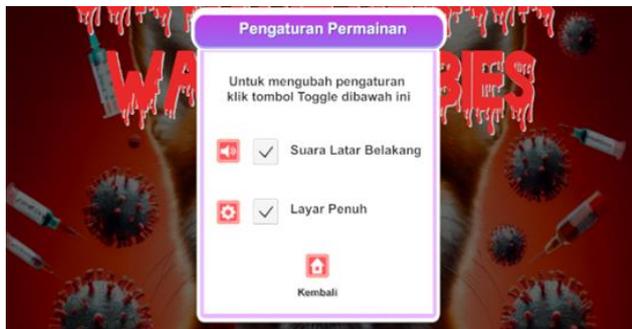
4.4 Pembuatan

Pada tahap *assembly* (pembuatan), seluruh objek dan bahan multimedia mulai di produksi. Produksi tersebut mengacu pada desain dan pengumpulan bahan atau model yang telah dilakukan sebelumnya. Tahap *assembly game* Survival Horror : Wabah Rabies dibuat dengan menggunakan Unity Engine.



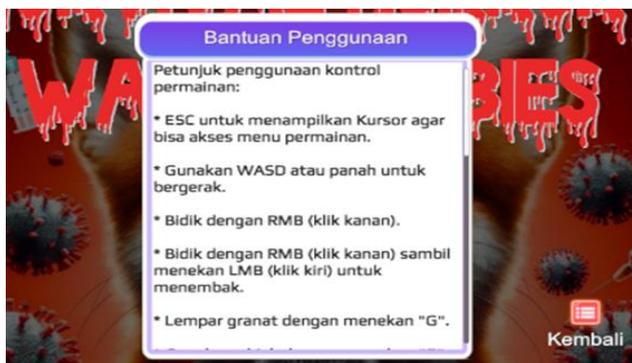
Gambar 11. Tampilan halaman menu utama

Menu utama merupakan tampilan utama dari *game* disaat permainan baru memasuki aplikasi. Pada gambar 12 menu utama terdapat judul dari *game* yaitu “Survival Horror : Wabah Rabies” dan setiap tombol menu utama memiliki fungsi masing-masing. Ketika pemain mengklik *button* mulai maka akan menuju *scene story*, saat pemain mengklik *button* pengaturan maka akan menampilkan menu pengaturan *game*, jika pemain mengklik *button* bantuan maka akan menampilkan menu bantuan, jika pemain mengklik *button* tentang maka akan menampilkan menu tentang, jika pemain mengklik *button* keluar maka pemain akan keluar dari *game*.



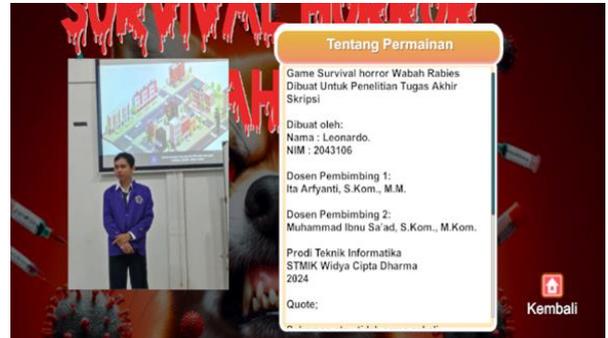
Gambar 12. Tampilan menu pengaturan

Pembuatan menu pengaturan dilakukan agar pengguna dapat mengatur pengaturan suara latar belakang, atau tampilan layar penuh pada permainan dengan klik *checkbox*, tombol kembali digunakan untuk menuju menu utama.



Gambar 13. Tampilan menu bantuan

Menu bantuan menyediakan instruksi dan petunjuk tentang cara menggunakan fitur-fitur permainan seperti kontrol mouse dan penggunaan *keyboard*, serta di dalamnya terdapat tombol kembali untuk menuju menu utama.



Gambar 14. Tampilan menu Tentang

Menu tentang memberikan informasi tentang pengembang permainan, detail nama permainan, gambar profil, dosen pembimbing, dan terdapat tombol kembali untuk menuju menu utama.



Gambar 15. Tampilan Panel Video Rabies

Panel video ini digunakan untuk menyampaikan informasi tentang apa itu rabies, bagaimana penyebarannya, gejala yang harus diwaspadai, dan pentingnya pencegahan. Ini membantu meningkatkan kesadaran akan bahaya rabies dan cara mencegahnya, dalam panel terdapat satu *checkbox* untuk hidup dan matikan suara latar belakang serta tombol lewati video untuk menuju panel alur cerita.



Gambar 16. Tampilan Panel Alur Cerita

Pada gambar 17 adalah panel cerita merupakan urutan peristiwa yang membentuk sebuah narasi dalam cerita, mulai dari seorang polisi yang ditugaskan antar vaksin ke rumah warga terinfeksi rabies, senapan anti rabies untuk melawan hewan terinfeksi, dan misi yang harus dilaksanakan. Terdapat tombol berikutnya untuk menuju panel alur permainan.



Gambar 17. Tampilan HUD Level 2

Level 1 dan level 2 pemain diperkenalkan pada lingkungan *game*, seperti mini peta, tampilan grafis, musuh, karakter utama, misi yang harus dicapai, HUD antarmuka pengguna, dan elemen-elemen visual yang akan pemain temui di level selanjutnya. level satu memberikan kesan dunia gelap malam hari dengan sumber pencahayaan yang minim serta NPC manusia berfungsi sebagai warga kota, anjing serta kucing sebagai hewan peliharaan, dan serigala sebagai musuh. Untuk level dua *Skybox* latar belakang langit *sunset cubemap* yang digunakan pada level dua memberikan kesan matahari terbenam serta terdapat dua mutasi serigala sebagai musuh besar.



Gambar 18. Tampilan gagal dan kalah

Tampilan kalah akan muncul ketika pemain kalah atau gagal menyelesaikan misi yang sedang berlangsung. Terdapat tombol mengulang untuk mengulang permainan dari awal.



Gambar 19. Tampilan menang

Tampilan menang akan muncul ketika pemain menang dan menyelesaikan misi dalam permainan. Terdapat tombol lanjutkan untuk melanjutkan permainan.



Gambar 20. Pemasangan Script algoritma A Star Pathfinding

Membuat musuh dengan menerapkan algoritma A* (A Star) dalam permainan, algoritma pencari rute terpendek di gunakan mengejar pemain dengan menambahkan *script A_Star* dan *Pathfinding* pada objek 3D musuh di mana kecepatan pergerakan A* diatur menjadi *3.5 units per second* menyesuaikan dengan animasi *Walk* (berjalan) musuh, sedangkan radius deteksi A* *300 units* artinya musuh tidak dapat mengejar pemain di luar radius deteksi.

4.5 Testing

4.5.1 Pengujian White Box

White Box testing akan menguji beberapa modul program pada *game* "Survival Horror : Wabah Rabies", pengujian meliputi jenis pengujian, jenis program atau *control* program, hasil yang diharapkan dan hasil pengujian. Modul yang diuji sebagai berikut:

1. Kode Program Algoritma Pathfinding pada musuh NPC

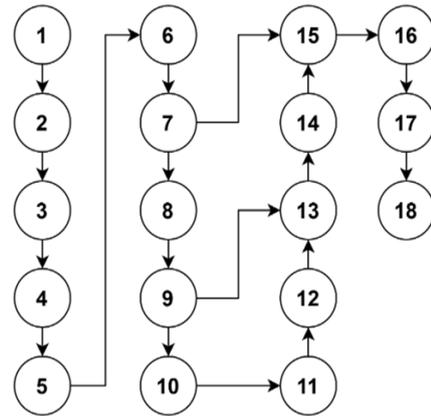
Kode Program Algoritma Pathfinding
<pre>public class Pathfinding : MonoBehaviour { public Grid grid; public List<Node> FindPath(Vector3 startPos, Vector3 targetPos) {</pre>

```

        Node startNode =
        grid.NodeFromWorldPoint(startPos);
        Node targetNode =
        grid.NodeFromWorldPoint(targetPos);
        List<Node> openSet = new List<Node>();
        HashSet<Node> closedSet = new
        HashSet<Node>();
        openSet.Add(startNode);
        while (openSet.Count > 0) {
            Node currentNode = openSet[0];
            for (int i = 1; i < openSet.Count; i++) {
                if (openSet[i].fCost < currentNode.fCost ||
                openSet[i].fCost == currentNode.fCost &&
                openSet[i].hCost < currentNode.hCost) {
                    currentNode = openSet[i];
                }
                openSet.Remove(currentNode);
                closedSet.Add(currentNode);
            }
            foreach (Node neighbour in
            grid.GetNeighbours(currentNode)) {
                if (!neighbour.walkable ||
                closedSet.Contains(neighbour)) {
                    continue;
                }
                int newMovementCostToNeighbour =
                currentNode.gCost + GetDistance(currentNode,
                neighbour);
                if (newMovementCostToNeighbour <
                neighbour.gCost || !openSet.Contains(neighbour)) {
                    neighbour.gCost =
                    newMovementCostToNeighbour;
                    neighbour.hCost =
                    GetDistance(neighbour, targetNode);
                    neighbour.parent = currentNode;
                    openSet.Add(neighbour);
                }
            }
        }
        (Node startNode, Node endNode) {
            rentNode = endNode;
            currentNode = currentNode.parent;
        }
        path.Reverse();
        grid.path = path;
        return path;
    }
    int GetDistance(Node nodeA, Node nodeB) {
        int dstX = nodeA.gridX - nodeB.gridX;
        int dstY = nodeA.gridY - nodeB.gridY;
        int distance =
        Mathf.RoundToInt(Mathf.Sqrt(dstX * dstX + dstY *
        dstY));
    }
    }
    
```

1) Flow Graph

Untuk menyederhanakan kode program dilakukan konversi ke dalam *flow graph*, dapat dilihat pada Gambar 22 berikut:



Gambar 21. Flow Graph Algoritma Pathfinding

2) Cyclomatic Complexity

Berdasarkan Gambar 22 dapat dihitung *cyclomatic complexity* sebagai berikut:

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 V(G) &= 19 - 18 + 2 \\
 V(G) &= 1 + 2 \\
 V(G) &= 3
 \end{aligned}$$

Dimana:

E= jumlah busur pada *flow graph*

N= jumlah simpul pada *flow graph*

1. Independent Path

Path 1: 1,2,3,4,5,6,7,15,16,17,18

Path 2: 1,2,3,4,5,6,7,8,9,13,14,15,16,17,18

Path 3: 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18

Berdasarkan hasil pengujian dihasilkan nilai *Cyclomatic Complexity* yang sama yaitu 3, maka dapat disimpulkan bahwa pada proses pencarian dengan algoritma Pathfinding berjalan dengan baik, karena setiap pengujian menghasilkan nilai yang sama.

2. Kode Program Algoritma A Star pada musuh NPC

```

Kode Program Algoritma Pathfinding
public class Pathfinding : MonoBehaviour {
    public Grid grid;
    public List<Node> FindPath(Vector3
    startPos, Vector3 targetPos) {
        Node startNode =
        grid.NodeFromWorldPoint(startPos);
        Node targetNode =
        grid.NodeFromWorldPoint(targetPos);
        List<Node> openSet = new
        List<Node>();
        HashSet<Node> closedSet = new
        HashSet<Node>();
        openSet.Add(startNode);
        while (openSet.Count > 0) {
            Node currentNode = openSet[0];
            for (int i = 1; i < openSet.Count; i++) {
                if (openSet[i].fCost <
                currentNode.fCost || openSet[i].fCost ==
    
```

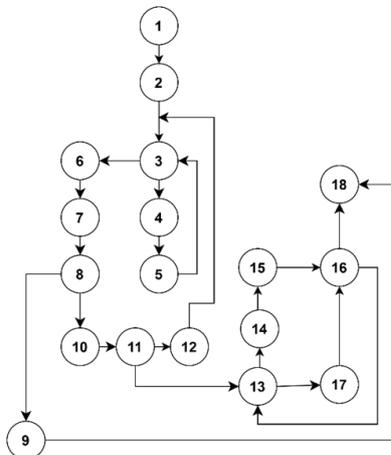
```

currentNode.fCost && openSet[i].hCost <
currentNode.hCost) {
    currentNode = openSet[i]; }
    openSet.Remove(currentNode);
    closedSet.Add(currentNode); }
    foreach (Node neighbour in
grid.GetNeighbours(currentNode)) {
        if (!neighbour.walkable ||
closedSet.Contains(neighbour)) {
            continue;
        }
        int newMovementCostToNeighbour
= currentNode.gCost +
GetDistance(currentNode, neighbour);
        if (newMovementCostToNeighbour
< neighbour.gCost ||
!openSet.Contains(neighbour)) {
            neighbour.gCost =
newMovementCostToNeighbour;
            neighbour.hCost =
GetDistance(neighbour, targetNode);
            neighbour.parent = currentNode;
            openSet.Add(neighbour);
        } }
(Node startNode, Node endNode) {
    rentNode = endNode;
    currentNode = currentNode.parent; }
    path.Reverse();
    grid.path = path;
    return path; }
int GetDistance(Node nodeA, Node nodeB)
{
    int dstX = nodeA.gridX - nodeB.gridX;
    int dstY = nodeA.gridY - nodeB.gridY;
    int distance =
Mathf.RoundToInt(Mathf.Sqrt(dstX * dstX +
dstY * dstY));}

```

1) *Flow Graph*

Untuk menyederhanakan kode program dilakukan konversi ke dalam *flow graph*, dapat dilihat pada Gambar 23 berikut:



Gambar 22. *Flow Graph* Algoritma A Star

2) *Cyclomatic Complexity*

Berdasarkan Gambar 23 dapat dihitung *cyclomatic complexity* sebagai berikut:

$$\begin{aligned}
 V(G) &= E - N + 2 \\
 V(G) &= 22 - 18 + 2 \\
 V(G) &= 4 + 2 \\
 V(G) &= 6
 \end{aligned}$$

Dimana:

E= jumlah busur pada *flow graph*

N= jumlah simpul pada *flow graph*

1. *Independent Path*

Path 1: 1,2,3,6,7,8,9,18

Path 2: 1,2,3,4,5,6,3,7,8,9,18

Path 3: 1,2,3,4,5,6,3,7,8,10,11,13,17,16,18

Path 4: 1,2,3,4,5,6,3,7,8,10,11,13,14,15,16,18

Path 5: 1,2,3,4,5,6,3,7,8,10,11,12,3,6,7,8,9,18

Path 6: 1,2,3,4,5,6,3,7,8,10,11,12,3,6,7,8,10,11,13,17,16,18

Berdasarkan hasil pengujian dihasilkan nilai *Cyclomatic Complexity* yang sama yaitu 6, maka dapat disimpulkan bahwa pada proses pencarian dengan algoritma A* (A star) berjalan dengan baik, karena setiap pengujian menghasilkan nilai yang sama.

4.5.2 *Pengujian Beta Testing*

Pengujian *beta* merupakan pengujian yang dilakukan secara objektif yang diuji secara langsung oleh pengisi kuisioner yang diisi oleh 10 responden.

Berdasarkan data hasil kuisioner tersebut, dapat dicari persentase masing-masing jawaban dengan menggunakan rumus :

$$Y = P/Q * 100\%$$

Keterangan :

Y = Nilai Persentase

P = Banyaknya Jawaban

Q = Jumlah Responden

Kesimpulan pengujian Beta Testing pada table berikut :

Tabel 4.9 Tabel pengujian beta testing

No	Pertanyaan	Penilaian				Total
		BS	B	K	KS	
1	Bagaimana tampilan desain menu utama permainan?	2	8	0	0	10
2	Bagaimana penilaian anda untuk informasi kesehatan karakter, amunisi, dan sumber daya ditampilkan dengan jelas di <i>Heads-Up Display</i> (HUD)?	4	6	0	0	10
3	Bagaimana penempatan elemen HUD seperti peta <i>mini</i> , indikator misi, dan status karakter?	3	7	0	0	10

4	Apakah <i>gameplay</i> dalam <i>game</i> ini membuat anda lebih sadar akan bahaya rabies?	2	6	0	2	10
5	Apakah anda merasa bahwa <i>game</i> ini memberikan informasi yang seimbang antara edukasi rabies dan hiburan?	4	4	1	1	10
6	Bagaimana dengan alur cerita dalam <i>game</i> ini apakah menurut anda cerita tersebut menyampaikan informasi tentang rabies dengan mudah untuk dipahami?	4	5	1	0	10
7	Penilaian anda bagian tingkat kesulitan pada level 1 dan level 2 dalam permainan apakah sesuai?	3	7	0	0	10
8	Bagaimana pendapat anda tentang cara <i>game</i> ini menggambarkan penularan rabies?	3	7	0	0	10
9	Bagaimana apakah semua fitur yang ada di dalam permainan dapat berjalan sesuai dengan fungsinya?	3	7	0	0	10
10	Penilaian anda seberapa efektif edukasi meningkatkan kesadaran tentang rabies diterapkan dalam permainan?	6	3	1	0	10
	Total	34	60	3	3	100

Dari persentase diatas dapat disimpulkan bahwa 6 dari 10 responden atau 60% menyatakan baik sekali. 3 dari 10 responden atau 30% menyatakan baik. 1 dari 10 responden atau 10% menyatakan kurang. 0 dari 10 responden atau 0% menyatakan kurang sekali.

Dari semua jawaban 10 responden terhadap 10 pertanyaan telah terkumpul 100 total pilihan dan dari ke 100 pilihan tersebut terdapat:

Baik Sekali : $34/100 * 100\% = 34\%$

Baik : $60/100 * 100\% = 60\%$

Kurang : $3/100 * 100\% = 3\%$

Kurang Sekali : $3/100 * 100\% = 3\%$

Dari hasil persentase diatas dapat disimpulkan bahwa rata-rata responden menjawab baik sekali 34%, baik 60%, kurang 3% dan 3% kurang sekali.

4.6 Distribusi

Pada tahap ini pembuatan *Game Survival Horror : Wabah Rabies* dikemas menjadi suatu aplikasi yang dapat dimainkan pada *platform* Windows 10 dan diatasnya. Distribusi aplikasi menggunakan media sosial *post* Instagram terdapat *barcode link* Google Drive dimana folder *game* dikemas menjadi satu *file* ZIP berukuran 400MB agar lebih mudah diunduh.

5. KESIMPULAN

Berdasarkan uraian dari masing-masing bab dan hasil pembahasan maka dapat disimpulkan sebagai berikut:

1. Manfaat Penerapan Algoritma A* (A Star) Pathfinding Pada *Game 3D Isometric Projection "Survival Horror : Wabah Rabies"* dapat mendeksi musuh sesuai radius deteksi dan kecepatan A* Pathfinding dalam mengejar pemain dapat diatur agar masing-masing musuh memiliki variasi kecepatan dalam mengejar pemain.
2. Manfaat Penerapan Algoritma A* (A Star) Pathfinding Pada *Game 3D Isometric Projection "Survival Horror : Wabah Rabies"* dalam permainan membantu menciptakan pengalaman yang lebih realistis, menantang, dan mendebarkan bagi pemain, dengan menerapkan Algoritma A* (A Star) Pathfinding pada musuh serigala dan musuh terakhir serigala mutasi untuk mencari jalur tercepat dan jalur terbaik menuju pemain.
3. Hasil pengujian yang telah dilakukan menurut pemain permainan dengan menggunakan metode pengujian beta adalah 60% pengguna menyatakan baik. Artinya Penerapan Algoritma A* (A Star) Pathfinding Pada *Game 3D Isometric Projection "Survival Horror : Wabah Rabies"* maka permainan ini dinilai layak untuk dimainkan dan memiliki nilai edukasi mengenai rabies yang baik dimana elemen hiburan tidak mengurangi kejelasan pesan edukatif yang ingin disampaikan.

6. SARAN

Setelah melakukan penelitian ini maka dengan ini saran-saran yang akan dikemukakan oleh peneliti adalah sebagai berikut:

1. Untuk pengembangan *Game Survival Horror : Wabah Rabies* ini, diharapkan dapat lebih dikembangkan lagi menggunakan Algoritma A* Unity Berbasis Graf Navigation Mesh.
2. Diharapkan ke depannya permainan *Survival Horror : Wabah Rabies* ini dapat dikembangkan lagi dengan menerapkan algoritma Line of Sight dan Finite State Machine (FSM).
3. Diharapkan ke depannya permainan *Survival Horror : Wabah Rabies* ini dapat dikembangkan mulai dari segi fitur menggunakan *save progress*, pengalaman bermain, grafis, maupun aspek edukatifnya, sehingga bisa memberikan pengalaman yang lebih mendalam, konten edukasi bervariasi, dan menarik bagi pemain.
4. Diharapkan untuk ke depannya permainan *Survival Horror: Wabah Rabies* akan dikembangkan untuk *smartphone* Android dan iOS.
5. Diharapkan ke depannya pemain dapat memilih variasi musuh berupa hewan sebagai NPC sesuai dengan tingkat kesulitan mulai dari tingkat mudah, normal, dan sulit.



7. REFERENSI

- Antonius, Pagintak. 2017. *“Visualisasi 3D Blender Pembelajaran Organ Dalam Tubuh Manusia Dengan Android Augmented Reality”*, Jurusan teknik Informatika, Samarinda : Sekolah Tinggi Manajemen Informatika dan Komputer Widya Cipta Dharma.
- Barnouti, N.H., Al-Dabbagh, S.S.M. dan Sahib Naser, M.A. (2016) “Pathfinding in Strategy Games and Maze Solving Using A* Search Algorithm,” *Journal of Computer and Communications*, 04(11), hal. 15–25. Tersedia pada: <http://dx.doi.org/10.4236/jcc.2016.411002>.
- Hasanah, F.N. (2020) *Buku Ajar Rekayasa Perangkat Lunak, Buku Ajar Rekayasa Perangkat Lunak*. Tersedia pada: <https://doi.org/10.21070/2020/978-623-6833-89-6>.
- Nadeem, M., & Panda, P.K. (2020). Survival in human rabies but left against medical advice and death followed – Community education is the need of the hour. *Journal of Family Medicine and Primary Care*, 9, 1736 - 1740.
- Sugiarto, H. (2018) “Penerapan Multimedia Development Life Cycle Pada Aplikasi Pengenalan Abjad Dan Angka,” *IJCIT (Indonesian Journal on Computer and Information Technology)*, Vol.3 No.1(1), hal. 26–31.
- Wahyuni, F.S. dan Mantja, S.N. (2016) “Penerapan Algoritma A * Untuk Pencarian Rute Terdekat Pada Permainan Berbasis Ubin (Tile Based Game),” hal. 168–172