

BAB II

TINJAUAN PUSTAKA

2.1. Media pembelajaran

2.1.1. Pengertian media pembelajaran

Kata “Media” berasal dari bahasa latin yang merupakan bentuk jamak dari “medium”. Secara harfiah berarti perantara atau pengantar. AECT mengartikan kata media sebagai segala bentuk dan saluran yang dipergunakan untuk proses informasi. NEA mendefinisikan media sebagai segala benda yang dapat dimanipulasikan, dilihat, didengar, dibaca atau dibicarakan. Heinich mengartikan media sebagai “*the term refer to anything that carries information between a source and a receiver*” [9].

2.1.2. Fungsi dan manfaat media

Terdapat beberapa fungsi media pembelajaran adalah sebagai berikut [9]:

1. Sebagai sarana bantu untuk mewujudkan situasi pembelajaran yang lebih efektif.
2. Sebagai salah satu komponen yang saling berhubungan dengan komponen lainnya dalam rangka menciptakan situasi belajar yang diharapkan.
3. Mempercepat proses belajar.
4. Meningkatkan kualitas proses belajar-mengajar.

2.2. Unity

Unity adalah alat pengembangan *multi-platform* komprehensif untuk konten interaktif seperti membuat *game* (2D dan 3D). Membangun visualisasi, animasi 3D *real-time* sebagai fitur pada *unity*. *Unity* adalah mesin permainan profesional yang terintegrasi penuh. *Unity engine* mempunyai fitur skrip untuk membuat program permainan. Skrip khusus dapat ditulis dalam JavaScript, C #, atau Boo merupakan Bahasa yang terdapat pada *unity*. *Unity* memiliki fungsi mengedit adegan *virtual* yang kuat, fungsi *rendering*, dan fungsi manajemen logika yang kuat. Untuk pembuatan fitur tersebut terdapat pada *platform unity*. *Platform unity* memiliki fungsi implementasi fisika yang sangat kuat. Fungsi tubuh, gravitasi, dan sifat fisik lainnya

melekat pada *GameObject*. Fungsi implementasi fisika dapat digunakan untuk mensimulasikan fenomena fisik yang realistis [10].

2.3. Algoritma

2.3.1. *Pathfinding*

Pathfinders membiarkan pengguna merencanakan penyelesaian masalah terlebih dahulu. Daripada menunggu sampai saat terakhir untuk menyelesaikan sebuah masalah. Perencanaan terlebih dahulu biasanya lambat tetapi memberikan hasil yang lebih baik. Sedangkan pergerakan terlebih dahulu lebih cepat tetapi terhambat perjalanannya. Jika geografis permainan sering berubah, maka perencanaan terlebih dahulu kurang bernilai [11].

2.3.2. Algoritma A*

Algoritma A * adalah algoritma yang akan digunakan untuk mencari rute terpendek pada rute yang digunakan. Pencarian rute membutuhkan x dan y. Setiap titik tengah pada *grid* akan menjadi *node*. Untuk menentukan apakah *node* tersebut masuk ke area yang dapat dilalui atau area yang tidak bisa dilalui. Kemudian algoritma A * akan mencari rute terpendek dari *node* awal ke *node* tujuan. Pencarian rute dilakukan melalui *node* yang berada di dalam area rute yang bisa dilalui [12].

Nilai pencarian rute algoritma A* dihitung menggunakan rumus $f(n)$.

$$f(n) = g(n) + h(n) \quad (1)$$

$f(n)$ = biaya estimasi rute terpendek

$g(n)$ = biaya rute dari *node* awal ke awal ke *node* n

$h(n)$ = perkiraan biaya hambatan / nilai heuristik dari *node* awal ke *node* akhir

Nilai $f(n)$ adalah biaya estimasi rute terpendek. Sedangkan $g(n)$ menjadi biaya rute terpendek dari *node* awal ke *node* akhir, dan $h(n)$ menjadi perkiraan biaya hambatan / nilai heuristik dari *node* awal ke *node* akhir. Agar kita dapat

merekonstruksi setiap jalan. Kita perlu menandai setiap *node* dengan relatif yang memiliki nilai $f(n)$ yang optimal. Efisiensi nilai A^* sangat tergantung pada nilai heuristik $h(n)$. Tergantung pada jenis masalah, mungkin perlu menggunakan fungsi heuristik yang berbeda untuk menemukan solusi optimal [13].

2.3.4. Dijkstra

Algoritma Dijkstra adalah sebuah algoritma yang dipakai dalam memecahkan permasalahan jarak terpendek untuk sebuah graf berarah. Permasalahan rute terpendek dari sebuah titik ke akhir titik lain adalah sebuah masalah klasik optimasi. Masalah klasik optimasi ini banyak digunakan untuk menguji sebuah algoritma yang diusulkan. Permasalahan rute terpendek dianggap cukup baik untuk mewakili masalah optimisasi. Karena permasalahannya mudah dimengerti namun memiliki banyak pilihan solusi [14].

2.3.5. Perbandingan A^* dengan Dijkstra

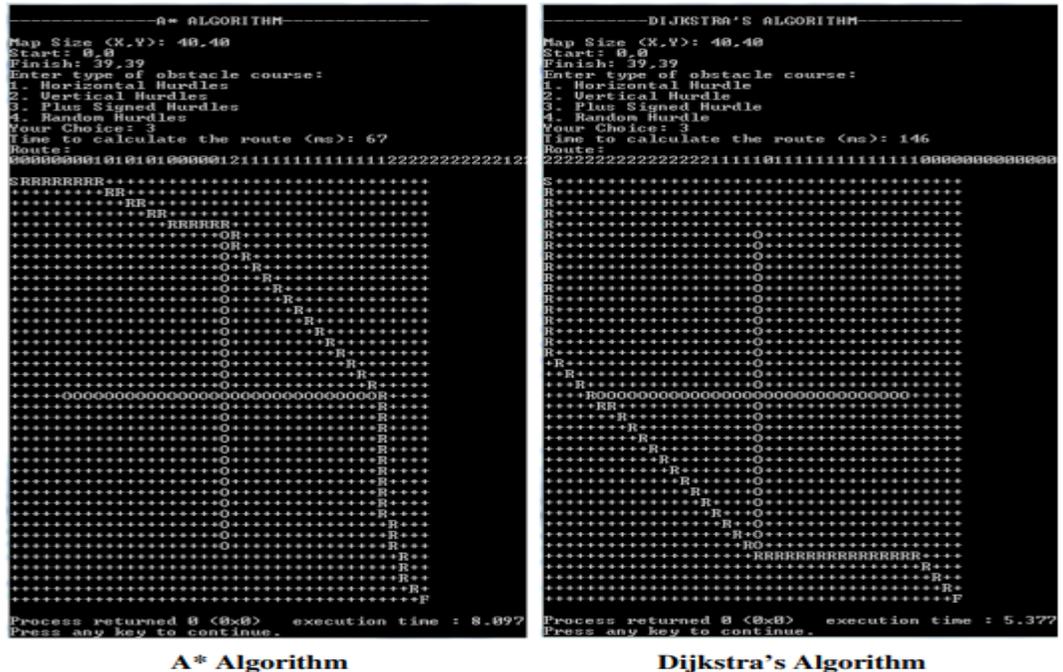
Dijkstra pada dasarnya sama dengan A^* , hanya saja tidak ada heuristik (H selalu 0). Karena tidak memiliki heuristik, ia mencari dengan memperluas secara merata ke segala arah, tetapi A^* memindai area hanya ke arah tujuan. Seperti yang Anda bayangkan, Karena Dijkstra ini biasanya berakhir menjelajahi daerah yang jauh lebih luas sebelum target ditemukan. Ini biasanya membuatnya lebih lambat dari A^* . Perbandingan A^* dengan Dijkstra's bisa dilihat pada table 2.1 [15].

Tabel 2. 1. Tabel perbandingan A* dan dijkstra

Parameters	A* Algorithm	Dijkstra's Algorithm
Search Algorithm	<i>Best First Search</i> , adalah algoritma pencarian yang menggunakan <i>node</i> paling menjanjikan dibatasi dengan aturan yang berlaku.	<i>Greedy Best First Search</i> , Metode pencarian dengan melebar secara rata ke setiap arah untuk mencari rute terpendek.
Time Complexity	Kompleksitas waktu adalah $O(n \log n)$. Kompleksitas waktu adalah jumlah waktu yang diperlukan untuk menjalankan algoritma. Dikarenakan A* memakan waktu hampir setengah waktu dijkstra pada data jurnal dan termasuk sebagai salah satu algoritma paling cepat. Oleh karena itu A* adalah $O(n \log n)$.	Kompleksitas waktu $O(n^2)$. Kompleksitas waktu adalah jumlah waktu yang diperlukan untuk menjalankan algoritma. Dikarenakan waktu Dijkstra memakan hampir 2 kali waktu A* pada data jurnal. Oleh karena itu A* adalah $O(n^2)$.
Heuristics Function	Fungsi Heuristik, $f(n) = g(n) + h(n)$, $g(n)$ mewakili biaya jalur dari titik awal ke tujuan. $h(n)$ mewakili perkiraan biaya	$f(n) = g(n)$, $g(n)$ mewakili biaya jalur dari titik awal ke tujuan. nilai $h(n) = 0$.

	hambatan dari titik awal ke tujuan.	
--	-------------------------------------	--

Perbandingan A* dengan dijkstra's bisa dilihat pada gambar 2.1 [14].



Gambar 2. 1. Gambar perbandingan A* dan dijkstra

a. $O(n \log n)$

Beberapa dari algoritma ini adalah yang paling optimal dan sering digunakan [16].

- *Merge sort*: algoritma pengurutan yang memecah kemudian menyelesaikan setiap bagian dan menggabungkannya kembali.
- *Heap sort*: metode sorting angka pada sebuah array dengan cara menyerupai *binary tree*.

b. $O(n^2)$

Algoritma ini seharusnya menjadi algoritma yang kurang efisien dibanding $O(n \log n)$. Penerapan umum $O(n^2)$ adalah *Brute Force* [16].

- *Bubble sort*: Bubble sort merupakan sebuah teknik pengurutan data dengan cara menukar dua data yang bersebelahan jika urutan dari data tersebut salah.
- *Insertion sort*: teknik pengurutan dengan cara membandingkan dan mengurutkan dua data pertama pada array.
- *Selection sort*: Selection sort merupakan sebuah teknik pengurutan dengan cara mencari nilai tertinggi / terendah di dalam array kemudian menempatkan nilai tersebut di tempat semestinya.

2.4. Virtual

Virtual berasal dari bahasa latin *virtualis* abad pertengahan. *Virtualis*, yang pada gilirannya berasal dari *virtus: strength, power*. *Virtual* tidak bertentangan dengan kenyataan, bisa menjadi bagian dari kenyataan [17]. Teknologi *virtual* selalu berkembang setiap tahun. Kemajuan dalam representasi 3D, simulasi komputer dan realitas virtual. Simulasi 3D bisa memberikan pengetahuan rinci tentang anatomi dan fungsi tubuh manusia [18].

2.5. Review penelitian terkait

2.5.1. Rancang bangun aplikasi pengenalan cerita rakyat timun mas berbasis *game* 3d

Pada penelitian permainan terdapat dua jenis karakter yaitu karakter utama dan dua karakter tambahan. Karakter utama adalah karakter yang memang ada dan menjadi elemen utama di dalam cerita. Dikarenakan karakter-karakter ini akan sangat berperan di alur jalannya cerita. Sedangkan karakter tambahan adalah karakter yang tidak ada di dalam cerita. Dimana karakter tambahan ini berperan sebagai musuh yang menghalangi pengguna dalam menyelesaikan permainan. Sehingga permainan menjadi lebih menarik dan interaktif bagi pengguna. Meskipun karakter tambahan ini disertakan di dalam permainan. tetapi tidak akan merubah cerita inti yang ada di dalam sistem ini [19].

2.5.2. Pembuatan *game action* “timun mas”

Pada penelitian permainan Timun Mas, dibuat menggunakan bahasa C# yang dapat dijalankan pada komputer atau laptop. Pembuatan permainan yang menggunakan sistem operasi *Windows* dengan komabilitas layar 1024x576 pixel. Permainan memiliki beberapa halaman. Halaman menu utama adalah halaman yang pertama kali diakses oleh pemain ketika menjalankan permainan. Pada menu utama terdapat 6 tombol pilihan *menu* yaitu mode cerita, mode survival, cara bermain, pengaturan, nilai tertinggi, dan keluar serta tombol ganti pemain [20].

2.5.3. Pembuatan *game 3d myfantasy*

Permainan 3D *My Fantasy* adalah sebuah permainan dengan tema *action adventure*. Didalam permainan terdapat 9 tahap, dimana tiap tahap berbeda lokasi [21].

1. Tahap 1 *room incubator*

Dalam tahap 1 karakter atau tokoh hero Kai memulai perjalanan untuk mengalahkan musuh utama di tahap 8 yaitu ruang bos.

2. Tahap 2 lorong

Merupakan jalan penghubung tahap 1 untuk menuju tahap selanjutnya

3. Tahap 3 gudang

Di dalam tahap 3 karakter utama hero kai akan memulai pertempuran dengan musuh/anak buah dari raja musuh.

4. Tahap 4 labirin

Didalam tahap 4 adalah sebuah lorong-lorong yang menyesatkan. Selain itu di tahap 4 karakter hero kai akan melawan anak buah raja musuh.

5. Tahap 5 *scene*

Di dalam tahap 5 karakter utama kai tidak akan melakukan pertempuran, karena di tahap 5 hanyalah jalan menuju tahap selanjutnya.

6. Tahap 6 *jump platform*

Adalah sebuah tempat dimana tidak terdapat jalan seperti tangga berundak atau yang lainnya, melainkan dengan melewati *platform* yang tersusun zig-zag keatas.

7. Tahap 7 *after jump*

Didalam tahap 7 sama seperti tahap 2 yaitu hanya sebuah jalan penghubung untuk ke tahap selanjutnya yaitu ruang bos.

8. Tahap 8 ruang bos

Didalam tahap 8 maka karakter utama akan melawan raja musuh beserta anak buahnya.

9. Tahap 9 *escape*

Tahap 9 adalah tahap terakhir dimana karakter kai telah berhasil mengalahkan semua musuh termasuk raja musuh.

2.5.4. Pembuatan aplikasi kumpulan cerita daerah di Indonesia berbasis android

Pada penelitian aplikasi cerita daerah menampilkan informasi tentang berbagai cerita rakyat yang ada di Indonesia. Aplikasi cerita terdiri dari beberapa bagian utama yang mengisi fitur – fitur aplikasi. Bagian aplikasi terdiri dari *splash screen*, menu utama, menu kumpulan cerita, menu isi cerita, menu pencarian, menu bantuan dan menu keluar. Dengan penyusunan cerita yang diurut sesuai dengan asal daerah serta adanya menu pencarian, *user* menjadi dimudahkan untuk mencari cerita yang diinginkan mengingat aplikasi ini memuat cukup banyak judul cerita. Serta desain aplikasi yang baik akan menambah ketertarikan pengguna dalam menggunakan aplikasi ini [22].