

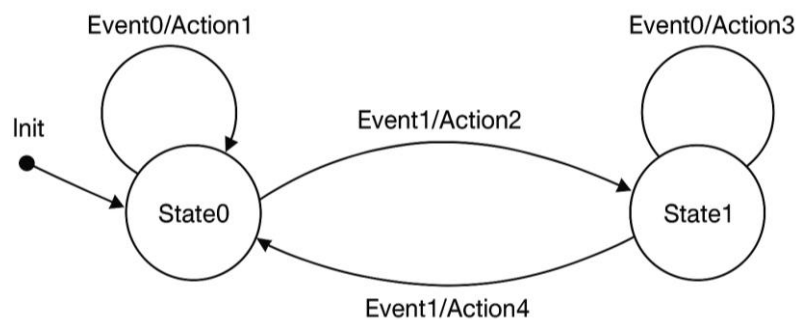
BAB II

TINJAUAN PUSTAKA

2.1. *Finite State Machine* (FSM)

Finite State Machine (FSM) adalah model yang digunakan untuk mempresentasikan perilaku suatu sistem atau objek yang kompleks dengan berbagai kondisi yang telah ditentukan (Yulsilviana & Ekawati, 2019). *Finite State Machine* yang diterapkan pada *game* RPG merupakan sistem *state per state* yang dirancang pada sistem kontrol dimana menggambarkan tingkah laku atau prinsip kerja sistem menggunakan tiga hal yaitu, *state* (keadaan), *event* (kejadian) dan *action* (aksi) (Astuti dkk., 2019).

Dalam *state machine*, sistem berada dalam satu *state* (keadaan) tertentu. Sistem akan bertransisi dari satu *state* ke *state* lain jika menerima masukan *event* (kejadian) tertentu. Sistem akan terus melakukan aksi yang sama dalam suatu keadaan hingga menerima *event* tertentu. Setiap *state* saling terhubung melalui transisi, dimana setiap transisi mengarah ke *state* lainnya. Transisi ini biasanya disertai dengan aksi yang dilakukan oleh sistem sebagai respon terhadap masukan yang diterima.



Gambar 2.1. Diagram *state* sederhana (Yulsilviana & Ekawati, 2019)

Gambar 2.1 menampilkan diagram FSM dengan dua *state*, dua *input*, dan empat aksi *output* yang berbeda. Seperti yang ditunjukkan pada Gambar 2.1, saat sistem diaktifkan, sistem akan bertransisi ke *state0*. Dalam keadaan ini, sistem akan menghasilkan *action1* jika menerima *event0*, sedangkan jika *event1* terjadi, maka *action2* akan dijalankan. Setelah itu, sistem akan bertransisi ke *state1* dan proses berlanjut sesuai dengan kondisi yang telah ditentukan.

2.2. *Multimedia Development Life Cycle (MDLC)*

Multimedia Development Life Cycle (MDLC) merupakan metode pengembangan multimedia yang dilakukan berdasarkan enam tahapan. Tahapan tersebut yaitu *concept*, *design*, *material collecting*, *assembly*, *testing*, dan *distribution*. Menurut Luther, keenam tahapan tersebut tidak harus berurutan dan dapat saling bertukar posisi dengan tahap *concept* tetap pada tahapan pertama (Mustika dkk., 2018).

2.3. *Non Playable Character (NPC)*

Non Playable Character (NPC) merupakan entitas karakter dalam *video game* yang tidak dikendalikan oleh pemain, melainkan dikendalikan oleh AI atau *script* yang diprogram oleh *game*. NPC berfungsi sebagai penggerak alur cerita dengan memberikan *quest* dan informasi, menyediakan interaksi, serta membuat dunia *game* terasa lebih hidup (Nendya dkk., 2015).

2.4. *Video Game*

Video game merupakan sebuah bentuk hiburan yang digemari oleh semua kalangan, mulai dari anak-anak hingga dewasa. Selain berfungsi sebagai hiburan, *game* juga dapat membantu melatih pola pikir seseorang dalam mencari solusi untuk menyelesaikan permasalahan yang ada pada *game*. *Game* dapat dimainkan dalam perangkat elektronik, seperti *console*, PC, hingga *smartphone* (Asri, 2022). *Video game* memiliki berbagai *genre*, seperti:

1. *Action: game* yang penuh aksi yang memberikan tantangan kepada pemain seperti respon kecepatan (contoh: Devil May Cry, Assassin's Creed)
2. *Adventure: game* yang fokus pada eksplorasi dan memecahkan *puzzle* (Contoh: Tomb Raider, Uncharted)
3. *RPG: game* dimana pemain mengendalikan atau membuat karakter dan mengikuti alur cerita (Contoh: The Elder Scroll, Final Fantasy)
4. *Simulation: game* yang meniru aktivitas manusia maupun bermain sebagai hewan yang ada di kehidupan nyata (Contoh: The Sims, Goat Simulator)
5. *Strategy: game* yang mengharuskan pemain untuk membuat rencana dan taktik (Contoh: Age of Empire, Clash of Clans)
6. *Sport: game* yang meniru permainan olahraga pada kehidupan nyata (Contoh: EAFC, Wii Sport)
7. *Racing: game* yang berfokus pada balapan (Contoh: Gran Turismo, Need For Speed)
8. *Shooter: game* yang berfokus pada tembak menembak, baik dari perspektif orang pertama (FPS) maupun ketiga (TPS) (Contoh: Battlefield, The Division)

2.5. *Role Playing Game (RPG)*

Role Playing Game (RPG) merupakan sebuah permainan dimana para pemain berperan sebagai tokoh utama atau memainkan karakter fiksi dalam *game* tersebut dan bekerja sama untuk membuat sebuah cerita (Putri dkk., 2024). *Game RPG* memiliki beberapa karakteristik tertentu, seperti pemain harus membunuh *monster* dan *boss* untuk menaikkan *level* karakter, menjalankan *main quest* dan *side quest*, menjelajah dunia *fantasy* yang luas (Saputri & Pratiwi, 2016).

RPG lebih berfokus pada karakter atau peran yang mewakili pemain sebagai tokoh utama didalam *game*. Karakter utama dalam *game* ini dapat berubah dan berkembang sesuai dengan keinginan pemain seiring berjalannya permainan. Perkembangan karakter ditentukan oleh *equipment* yang dimiliki serta kenaikan *level* yang mencakup berbagai parameter seperti kecerdasan, kecepatan, dan kekuatan (Wijaya dkk., 2023).

2.6. **RPG Maker**

RPG Maker merupakan sebuah program yang digunakan untuk membuat *game* dengan *genre* RPG. Proses pembuatan *game* dibantu oleh *tools* yang telah disediakan. RPG Maker dibuat oleh perusahaan Jepang bernama ASCII yang kemudian dilanjutkan oleh Enterbrain. Proses pembuatan *game* dengan menggunakan RPG Maker menurut (Wijaya dkk., 2023) memiliki tiga dasar tahapan yaitu:

1. Pembuatan peta

Peta merupakan lokasi dimana *game* berlatar. Pembuatan peta membutuhkan

imajinasi yang tinggi dari *developer* supaya peta yang dibuat sesuai dengan alur cerita yang dibawa dan tidak menyimpang.

2. Penggunaan database

Database memungkinkan *developer* untuk mengatur *assets* yang akan digunakan dalam game, seperti karakter, objek, suara, gambar, animasi, dan lainnya yang dibutuhkan untuk *game*.

3. Pembuatan *event*

Proses dimana *developer* memberikan *event* (peristiwa) atau aktivitas dalam *game*, seperti *event* melawan *monster* ataupun melakukan *quest* dan aktivitas jual beli dengan NPC.

2.7. Logika Interaksi

Logika interaksi pada *game* RPG merupakan perpaduan antara logika permainan dengan mekanisme interaksi yang dirancang untuk menciptakan pengalaman naratif dan pengambilan keputusan bagi pemain. Bentuk interaksi yang terjadi antara pemain dan dunia *game* akan memiliki dampak yang logis terhadap alur cerita, status karakter, dan hasil permainan (Fauzy dkk., 2023).

2.8. *Black Box Testing*

Black box testing merupakan metode pengujian berdasarkan fungsionalitas yang melibatkan pengamatan terhadap *output* berdasarkan *input* tertentu (Ardana, 2019). Pengujian fungsionalitas pada *game* membutuhkan metode yang tepat untuk memastikan kualitas dan kinerja dari *game*. Salah satu metode pengujian

yang dapat digunakan untuk menguji *game* adalah *black box testing*. Metode ini akan menguji melalui sudut pandang pengguna tanpa memerlukan akses terhadap *source code* (Novalia & Voutama, 2022). Pengujian menggunakan *black box testing* dapat membantu mengidentifikasi kesalahan yang mungkin terjadi ketika menggunakan sistem. Pengujian dengan metode ini dapat membantu *developer* dalam memastikan *user interface* (UI) dapat digunakan dengan mudah dan responsif. UI diuji dengan melakukan pengujian interaksi antar komponen untuk mengidentifikasi kesalahan yang muncul akibat interaksi tersebut.

2.9. Pengujian Skenario

Pengujian skenario merupakan metode pengujian perangkat lunak yang digunakan untuk memastikan suatu dengan menjelaskan interaksi antara aktor dan sistem dalam bentuk teks yang menceritakan kondisi awal dan kondisi akhir pada sistem. Pengujian skenario dijelaskan dalam bentuk teks dengan format penulisan yang disesuaikan dengan kebutuhan (Fitriana, 2020). Pada *game*, pengujian skenario dilakukan untuk memastikan bahwa alur *game* dan interaksi pengguna berjalan sesuai dengan fungsinya.

2.10. Pengukuran Akurasi Transisi State

Pengukuran akurasi transisi *state* pada FSM dalam *game* merupakan metrik kuantitatif yang digunakan untuk mengukur tingkat ketepatan transisi yang terjadi selama interaksi pemain dalam *game*. Akurasi transisi *state* dihitung berdasarkan perbandingan antara jumlah transisi yang benar terhadap jumlah total transisi yang

diuji (Susanto & Dermawan, 2021). Metrik ini berfungsi untuk menilai seberapa konsisten dan akurat logika FSM dalam merespons kondisi permainan, input pemain, serta aturan yang telah ditetapkan dalam desain *game*.

$$AST = \frac{\text{Jumlah Transisi State Benar}}{\text{Jumlah Total Transisi Yang Diuji}} \times 100$$

Accuracy of State Transition (AST) merupakan nilai yang didapat dari pengukuran jumlah transisi dan jumlah total transisi yang diuji.

2.11. Penelitian Terkait

Penelitian yang dilakukan (Perdana & Edi, 2024) menunjukkan hasil dari segi teknis, penerapan FSM pada pengembangan dialog antar karakter, objek, dan area sekitar map diterapkan dengan baik, memungkinkan pemain akan mendapatkan jalan cerita yang berbeda saat memilih hal yang berbeda dari sebelumnya.

Penelitian lain dengan metode yang sama dilakukan oleh (Chaniago & Yanti, 2022) mendapatkan hasil yang menunjukkan alur cerita dan misi misi dibuat yang dibuat dengan FSM diterapkan dengan baik serta NPC dapat berjalan dan berdialog sendiri pada suatu *event* dapat ditampilkan oleh sistem.

Penelitian yang dilakukan (Amri dkk., 2024) menunjukkan hasil penerapan FSM pada pengelolaan alur permainan dan logika permainan yang memungkinkan transisi yang terstruktur sehingga pengalaman bermain menjadi lebih responsif.

Penelitian yang dilakukan (Prayogo dkk., 2025) menunjukkan hasil penerapan FSM dalam mengatur transisi antar *state* pada meningkatnya interaktivitas dan kedalaman *gameplay* dimana FSM memberikan fleksibilitas dalam mengatur perilaku NPC yang memungkinkan respon realistis terhadap aksi

pemain serta menciptakan pengalaman bermain yang imersif.

Penelitian yang dilakukan (San & Handriyanti, 2022) menunjukkan hasil penggabungan metode *pathfinding* A* dengan FSM mampu membuat NPC musuh untuk bergerak dan bereaksi mengejar pemain yang berada dalam jarak jangkauan musuh. Metode *pathfinding* A* yang digabungkan dengan FSM menaikkan efisiensi sebesar 8,3% berdasarkan hasil uji coba.

Penelitian oleh (Alsveta dkk., 2024) menunjukkan bahwa penerapan FSM pada NPC lawan menghasilkan perilaku yang cerdas. Dari 9 *button* yang digunakan dalam pembuatan game dapat berjalan dengan sempurna. Penelitian ini menggunakan kuisisioner dengan 7 pertanyaan dan 10 responden dengan faktor pembagi yaitu $7 \times 10 = 70$. Diperoleh 45 individu sangat setuju, 18 setuju, dan 11 tidak setuju. Terdapat 10 peserta yang diikutsertakan, 64,28% (45 orang) menyatakan sangat setuju, 25,71% (18 orang) setuju, dan 15,71% (11 orang) tidak setuju dengan hasil tersebut.

Penelitian oleh (Karmanto dkk., 2023) menunjukkan hasil penerapan FSM pada jangkauan $<4px$ dari pemain terbukti apabila musuh dalam *state* mengejar, sedangkan $>4px$ musuh berada dalam *state* diam ditempat. Begitupun untuk musuh dalam jangkauan $3px$ dari pemain. Hal ini menunjukkan bahwa perilaku NPC berjalan dengan baik sesuai dengan perilaku pemain.

Penelitian oleh (Ramadan & Hardjianto, 2022) menunjukkan hasil penerapan FSM pada NPC musuh berjalan dengan baik, sehingga NPC musuh berperilaku sesuai dengan interaksi pemain. Penelitian ini menggunakan kuisisioner dengan hasil *usability* sebesar 61,5%.

Penelitian yang dilakukan oleh (Satrio dkk., 2023) menunjukkan hasil penerapan FSM menghasilkan perilaku musuh sesuai dengan perilaku pemain. Penerapan FSM pada jarak $<17px$ dari pemain terbukti musuh akan masuk dalam *state* mengejar, sedangkan dalam jarak $\leq 5px$ musuh masuk dalam *state* menyerang.

Penelitian oleh (Hasan Syu'aibi dkk., 2023) menghasilkan penerapan FSM pada NPC musuh, terbukti pada saat jarak musuh $>$ jangkauan *turret* dan tidak terkena *trigger turret* akan masuk *state* berlari, sedangkan jarak musuh $\leq \max$ jangkauan *turret* dan terkena *damage* maka musuh akan masuk *state* terluka, jika nyawa musuh ≤ 0 dalam *state* berlari atau terluka maka masuk *state* mati.

Tabel 2. 1 Penelitian Terkait

No	Penulis	Judul	Metode	Fokus Penelitian
1.	(Perdana & Edi, 2024)	Penerapan Metode Finite State Machine (FSM) Pada Game Role Playing Game (RPG) Ink Journeys	Finite State Machine	Menerapkan FSM pada dialog, objek, dan area sekitar map
2.	(Chaniago & Yanti, 2022)	Penerapan Metode Finite State Machine Dalam Game “Stories of Cursed Dungeon” Menggunakan RPG Maker MV	Finite State Machine	Menerapkan FSM pada alur cerita dimana pemain menyelesaikan misi dalam game, dimana misi tersebut dibuat dengan FSM supaya pemain tidak mudah bosan.
3.	(Amri dkk., 2024)	Penerapan Metode Finite State Machine Dalam Game “Keteguhan Lisa” Menggunakan RPG Maker MV dan Android Studio	Finite State Machine	Menerapkan FSM pada pengembangan alur cerita dan logika permainan memungkinkan transisi yang terstruktur sehingga pengalaman bermain menjadi responsif.

No	Penulis	Judul	Metode	Fokus Penelitian
4.	(Prayogo dkk., 2025)	Penerapan Metode Finite State Machine Pada Game Edukasi Sejarah “Legacy of Sisingamagaraja XII” Menggunakan RPG Maker MV	Finite State Machine	Menerapkan FSM pada interaktivitas dengan mengatur perilaku dinamis NPC
5.	(San & Handriyanti, 2022)	Penerapan Metode Pathfinding Pada Pengembangan Game “The Book of Aksara” Pada Perangkat Bergerak	Finite State Machine dan A star	Menggabungkan FSM dan A* sebagai area deteksi NPC dan script chase event sebagai metode pengejaran pemain
6.	(Alsveta dkk., 2024)	Rancang dan Implementasi Metode Finite State Machine (FSM) Pada Game Battle For Floryn Berbasis Android	Finite State Machine	Menerapkan FSM pada NPC lawan untuk menciptakan kecerdasan perilaku NPC
7.	(Karmanto dkk., 2023)	Implementasi Metode Finite State Machine (FSM) Pada Game “Cipher Club” 2D Berbasis Android	Finite State Machine	Menerapkan FSM pada perilaku NPC dalam mengukur jarak jangkauan yang dipengaruhi oleh perilaku pemain
8.	(Ramadan & Hardjianto, 2022)	Penerapan Finite State Machine Pada Game “Pendekar Cisdane” Berbasis Android	Finite State Machine	Penerapan FSM pada NPC musuh yang dapat berperilaku sesuai dengan perilaku pemain
9.	(Satrio dkk., 2023)	Penerapan A* Pathfinding dan FSM (Finite State Machine) Pada Game “Lost Civilization” Berbasis Android	A Star dan Finite State Machine	Menerapkan A* dan FSM dalam penentuan jarak NPC yang akan bereaksi ketika pemain berada pada jarak tertentu

No	Penulis	Judul	Metode	Fokus Penelitian
10.	(Hasan Syu'aibi dkk., 2023)	Perancangan dan Implementasi Metode FSM (Finite State Machine) Pada Game Military Defence 2D Berbasis Android	Finite State Machine	Penerapan FSM pada NPC musuh dalam <i>game</i> RTS dimana ketika musuh masuk dalam jangkauan jarak <i>turret</i> , NPC musuh akan bereaksi sesuai dengan <i>state</i> yang telah ditentukan
11.	Usulan Penelitian	Penerapan Finite State Machine Untuk Membuat Logika Interaksi Pada Game RPG Menggunakan RPG Maker Pada Platform PC	Finite State Machine	Menerapkan FSM dalam berbagai <i>state</i> interaksi dengan cara menerapkannya pada NPC, objek, serta map pada game RPG

Dari berbagai penelitian tersebut, dapat disimpulkan bahwa *Finite State Machine* berperan dalam membuat kecerdasan buatan pada *game*, terutama dalam konteks perilaku NPC, jarak jangkauan, serta alur cerita dan logika *game*. Penelitian terbaru yang dilakukan oleh (Perdana & Edi, 2024) memiliki gap dimana *game* yang dikembangkan hanya dapat dimainkan di *Android* serta penggunaan FSM pada dialog dan objek *game* terasa monoton. Penelitian ini bertujuan untuk mengisi kekosongan tersebut dengan mengkaji FSM yang menghasilkan logika interaksi yang lebih kompleks.

Perbandingan penelitian terkait dapat dilihat melalui matriks penelitian pada Tabel 2.2. Matriks penelitian ini dilakukan untuk mengelompokan metode dan objek penelitian yang telah diteliti sebelumnya.

Tabel 2. 2 Matriks Penelitian

No	Penulis	Metode			Fitur				Genre			Platform	
		FSM	A*	Lainnya	NPC	Alur Cerita	Objek	Area Map	RPG	Side Scroll	RTS	PC	Android
1.	(Perdana & Edi, 2024)	√			√		√	√	√				√
2.	(Chaniago & Yanti, 2022)	√				√			√			√	
3.	(Amri dkk., 2024)	√			√	√			√				√
4.	(Prayogo dkk., 2025)	√			√				√				√
5.	(San & Handriyantini, 2022)	√	√		√				√				√
6.	(Alsveta dkk., 2024)	√			√				√				√
7.	(Karmanto dkk., 2023)	√			√				√				√

No	Penulis	Metode			Fitur				Genre			Platform	
		FSM	A*	Lainnya	NPC	Alur Cerita	Objek	Area Map	RPG	Side Scroll	RTS	PC	Android
8.	(Ramadan & Hardjianto, 2022)	√			√					√			√
9.	(Satrio dkk., 2023)	√	√		√					√			√
10.	(Hasan Syu'aibi dkk., 2023)	√			√						√		√
11.	Usulan Penelitian	√			√		√	√	√			√	

Pada Tabel 2.2 matriks penelitian menunjukkan perbandingan fitur antara penelitian sebelumnya dan penelitian yang akan dilakukan. Penelitian sebelumnya menggunakan metode Finite State Machine dan metode gabungan antara FSM dengan A* yang berfokus pada satu atau dua fitur tertentu. Sebagai contoh, penelitian oleh (San & Handriyantini, 2022) menggunakan metode gabungan FSM dan A*, yang hanya berfokus pada fitur NPC pada game RPG Android. Penelitian lain yang dilakukan oleh (Amri dkk., 2024) berfokus pada fitur NPC dan alur cerita pada game RPG Android.

Dalam penelitian ini, FSM yang akan diterapkan akan mendukung fitur NPC, objek, dan area map pada game RPG yang akan dibuat seperti penelitian yang dilakukan oleh (Perdana & Edi, 2024) tetapi dengan membawa kebaruan untuk mengisi gap penelitian sebelumnya. Kekosongan penelitian tersebut akan diisi dengan mengembangkan *game* untuk *platform* PC dan menerapkan FSM pada NPC yang memungkinkan respon NPC sesuai terhadap aksi pemain, FSM pada Objek untuk membuat objek interaktif serta mengendalikan mekanisme *puzzle*, FSM pada Map untuk mengendalikan *event* dan mengubah kondisi map.