

# Implementasi *Finite State Machine* (FSM) Dalam Game Monopoli 3D Teknologi Informasi dan Komunikasi dengan Algoritma *Fisher-Yates Shuffle* Berbasis Android

Dimas Setio Utomo<sup>1</sup>, Dodik Arwin Dermawan<sup>2</sup>

<sup>1,2</sup>Jurusan Teknik Informatika/Teknik Informatika, Universitas Negeri Surabaya

<sup>1</sup>[dimasutomo@mhs.unesa.ac.id](mailto:dimasutomo@mhs.unesa.ac.id)

<sup>2</sup>[dodikdermawan@unesa.ac.id](mailto:dodikdermawan@unesa.ac.id)

**Abstrak**—Perkembangan game sebagai media hiburan teknologi yang populer telah memanfaatkan kecerdasan buatan *Artificial Intelligence* (AI). *Artificial Intelligence* merupakan suatu program yang membuat komputer dapat melakukan pekerjaan bertindak dan berfikir seperti manusia. Dalam game, *Artificial Intelligence* mampu membantu pemain mencapai tujuan suatu permainan. Penelitian ini memanfaatkan salah satu Algoritma *Artificial Intelligence* yaitu *Finite State Machine* (FSM) yang akan digunakan untuk menentukan keputusan komputer agar dapat berinteraksi dalam game. Penelitian ini akan mengembangkan sebuah game android monopoli 3D bertema edukasi menggunakan *Finite State Machine*. Algoritma *Finite State Machine* (FSM) yang membentuk *Non Playable Character* (NPC) tersebut dapat memberikan respon atau memiliki tingkah laku sesuai dengan keadaan yang terjadi pada pemain. Selain itu, game monopoli ini akan ditambahkan Algoritma *Fisher-Yates Shuffle* yang merupakan metode untuk melakukan pengacakan posisi atau pengacakan soal. Berdasarkan hasil pengujian didapatkan pemodelan AI *Finite State Machine* pada NPC berjalan dengan baik sesuai yang diharapkan. Sedangkan pada pengujian Algoritma *Fisher-Yates Shuffle* sebanyak 5 kali pada game dengan waktu yang berbeda mendapatkan hasil pengacakan soal yang bervariasi dan tidak berganda.

**Kata Kunci**— *Game, Finite State Machine, Fisher-Yates Shuffle*

## I. PENDAHULUAN

Pada era digital saat ini, game adalah salah satu jenis hiburan dan media teknologi yang populer di berbagai kalangan masyarakat dari anak-anak hingga dewasa. Game dianggap dapat menghilangkan kepenatan setelah beraktivitas serta untuk mengisi waktu luang. Game juga dapat berfungsi untuk mengasah pola pikir seseorang untuk memecahkan masalah dalam game. Game dapat membuat seseorang berfikir kreatif dan inovatif. Selain itu, game dapat digunakan sebagai media pembelajaran. Game dapat dikombinasikan dengan teknologi kecerdasan buatan atau *Artificial Intelligence* untuk membuat game semakin hidup dan menarik. *Artificial Intelligence* dapat digunakan ketika pemain membutuhkan bantuan untuk mencapai tujuan pada game tersebut.

Perkembangan game dari waktu ke waktu juga semakin maju. Jika pada zaman lampau game hanya dapat dimainkan secara tradisional dan dengan peralatan yang sangat sederhana, sekarang game sudah memanfaatkan perkembangan teknologi yang lebih modern dan canggih. Saat ini banyak game menggunakan inovasi saat ini dalam produksi dan penggunaannya. Pemanfaatan inovasi kecerdasan buatan atau *Artificial Intelligence* (AI) adalah salah satu faktornya.

*Artificial Intelligence* adalah program komputer yang menciptakan komputer dapat bertindak dan berpikir seperti orang. komputer bukan hanya digunakan sebagai alat hitung, tetapi juga dapat digunakan untuk melakukan segala sesuatu seperti yang dilakukan oleh manusia [1]. Kecerdasan buatan yang bisa diaplikasikan pada game yaitu *Finite State Machine* (FSM) yang merupakan pemodelan AI di game yang populer. FSM biasanya digunakan untuk membentuk NPC (*Non Playable Character*) yang cerdas, sehingga dapat merespon karakter pemain. Seolah-olah dengan adanya NPC game dianggap lebih menarik dan menantang. Pemain akan merasa tertantang untuk mengalahkan NPC saat dimainkan [2].

Seperti pada game edukasi berlatarbelakang monopoli dengan ditambahkan edukasi berupa soal yang harus dijawab untuk mendapatkan *reward*. Apabila setiap pemain menempati wilayah kesempatan/dana umum maka akan disajikan sebuah soal Teknologi Informasi dan Komunikasi (TIK). *Finite State Machine* (FSM) digunakan untuk menentukan keputusan komputer agar dapat berinteraksi dalam game, seperti mengambil keputusan untuk melakukan transaksi pembelian tanah dan membayar denda. Pada game ini juga akan digunakan algoritma *Fisher-Yates Shuffle* yang merupakan metode yang digunakan untuk input maupun pengacakan posisi. Penentuan posisi menggunakan probabilitas yang sama untuk pengacakan soal. Game ini akan menggunakan perangkat mobile atau *mobile game*. Selain lebih mudah, beberapa kalangan menilai *mobile game* dinilai lebih praktis dimainkan oleh siapapun, kapanpun, dan dimanapun [3].

Adapun beberapa penelitian yang telah dilakukan yaitu pada tahun 2020 oleh Mohamad Irfan dengan judul "*Design And Build an Early Childhood Puzzle Educational Game Using The Fisher-Yates Shuffle Algorithm as an Android-Based Scrambler For Snippets*". Algoritma *Shuffle Fisher* diaplikasikan dalam pengacakan potongan citra, sehingga game edukasi puzzle yang dibangun diharapkan mampu meningkatkan pengetahuan siswa PAUD. Setelah dilakukan uji coba *pretest* dan *post test* didapatkan peningkatan hasil sebesar 23,68% dari hasil *pretest* 64,13% dan hasil *post test* 87,81% [4].

Pada tahun 2019, penelitian diarahkan oleh Dodik dengan judul "*Simulasi Standard Operational Procedure Laboratorium Komputer Dengan Pemodelan Finite State Machine Pada Perilaku Teknisi*". Mengingat penyelidikan, tampilan *Finite State Machine* dapat digunakan untuk memodelkan simulasi *Standard Operational Procedure Laboratorium Komputer PC* sebagai permainan pura-pura

dengan profesional sebagai NPC yang memiliki 13 kemajuan dan 5 negara bagian. Dengan tujuan bahwa SOP di pusat penelitian dapat semakin mudah dipahami dan benar-benar menarik. Dalam eksplorasinya, *Standard Operational Procedure* dapat digunakan sebagai substansi permainan dengan merencanakan alur cerita permainan menggunakan *storyboard*. [5].

Selanjutnya, di tahun 2018 dilakukan penelitian oleh Rahmat dengan judul “*Rangku Alu – A Traditional East Nusa Tenggara Game in Android Platform*”. Dalam Penelitian tersebut metode pengacakan yang digunakan untuk menentukan langkah selanjutnya adalah Algoritma Fisher-Yates Shuffle, sehingga mencegah area keran muncul di tempat yang sama lebih dari sekali berturut-turut. Hasil penelitian menunjukkan bahwa daerah keran pernah muncul di tempat yang sama dua kali atau lebih. Penggunaan Algoritma randomisasi *Fisher-Yates Shuffle* mampu menentukan gerakan berikutnya sehingga permainan *Rangku alu* ini memiliki variasi ketukan dengan dua mode permainan [6].

Pada tahun 2017, juga dilakukan penelitian oleh Billy dengan judul “*Implementasi Artificial Intelligence pada Game Defender of Metal City dengan Menggunakan Finite State Machine*”. Pada penelitian ini dilakukan penerapan *Artificial Intelligence* (AI) menggunakan *Finite State Machine* yang berfungsi mengendalikan pihak komputer. Pemain seolah-olah dihadapkan dengan seorang lawan saat bermain, sehingga dapat merasakan sensasi bertanding dengan lawan. Menurut penelitian ini, penggunaan *Finite State Machine* dalam model game ini, jika dikombinasikan dengan strategi dan aturan game yang sesuai akan membentuk AI yang kuat [7].

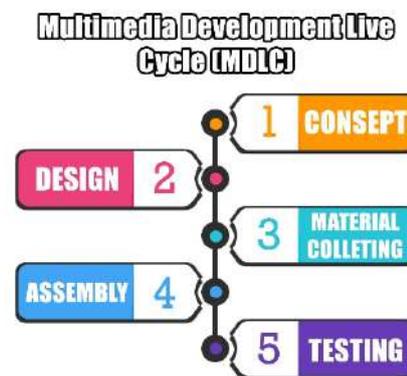
Kemudian ditahun yang sama, penelitian juga dilakukan oleh Subaeki dengan judul “*Implementasi Algoritma Fisher-Yates Shuffle pada Aplikasi Multimedia Interaktif untuk pembelajaran Tenses Bahasa Inggris*”. Penelitian ini menggunakan Algoritma *Fisher-Yates Shuffle* yang merupakan algoritma acak (*shuffling*) untuk melakukan pengacakan soal uji pemahaman. Melalui hasil dari penelitian yang dilakukan dapat diketahui bahwa Algoritma *Fisher-Yates Shuffle* bekerja sangat baik dalam mengacak (*shuffling*) data soal. Urutan soal yang dihasilkan oleh algoritma ini juga sulit ditebak, karena pada algoritmanya menggunakan *math random*. *Math random* yang akan mengubah urutan array yang dipanggil sehingga pola urutan soal sulit untuk ditebak [8].

Berdasarkan latar belakang tersebut, penelitian ini bertujuan membangun sebuah aplikasi yang berupa game *Monopoli 3D Teknologi Informasi dan Komunikasi* dengan menggunakan pemodelan *AI Finite State Machine* dan algoritma *Fisher-Yates Shuffle*. Pemodelan *AI Finite State Machine* digunakan untuk membentuk NPC dalam pengambilan mengambil keputusan pada game, sedangkan pengacakan soal akan menggunakan algoritma *Fisher-Yates Shuffle*. Pada penelitian ini, pemodelan *AI Finite State Machine* pada NPC dapat memberikan respon dan memiliki tingkah laku dengan menyesuaikan keadaan yang terjadi pada game yaitu NPC memiliki respon dalam game seperti melakukan transaksi pembelian tanah dan pembayaran denda seolah-olah NPC

tersebut berfikir supaya game lebih menarik dan menantang pemain untuk mengalahkan komputer saat dimainkan dan penerapan algoritma *Fisher-Yates Shuffle* berfungsi untuk melakukan pengacakan pada soal – soal, supaya soal dalam game menghasilkan acakan yang bervariasi dan tidak berulang-ulang. Soal pada game ditunjukkan kepada para pemain agar termotivasi bermain game sambil belajar tentang Teknologi Informasi dan Komunikasi.

## II. METODOLOGI PENELITIAN

Penelitian ini bertujuan untuk mengembangkan sebuah game monopoli edukasi yang lebih menarik dan efisien. Penambahan fitur kecerdasan buatan dengan memanfaatkan pemodelan *Finite State Machine* (FSM) yang akan digunakan dalam penelitian ini. Perancangan game edukasi monopoli TIK ini menggunakan metode pengembangan *Multimedia Development Live Cycle* (MDLC). MDLC merupakan metode pengembangan yang terdiri dari beberapa fase yaitu *concept*, *design*, *material collecting*, *assembly*, dan *testing*. Pada Gbr 1 menunjukkan alur proses dari metode pengembangan.



Gbr 1. Alur Metode Pengembangan

### A. Concept

Tahap pengembangan konsep merupakan tahap dalam merancang sebuah game edukasi Monopoli TIK. Game monopoli ini dirancang untuk anak-anak, Jadi pertanyaan yang diberikan juga berubah sesuai dengan beratnya informasi untuk anak-anak. Anak-anak dapat memilih menu yang tersedia di game. Menu utama terdiri dari menu awal untuk memulai permainan monopoli yang mengesankan, menu pengaturan untuk mengatur volume suara dari permainan, menu bahan pembelajaran dan menu keluar permainan. Dalam menu materi pembelajaran terdapat pertanyaan dan jawaban tentang soal materi TIK. Pembuatan soal yang ada di dalam *gameplay* permainan menggunakan materi yang disediakan pada menu materi pembelajaran. Menu selanjutnya *setting*, pada menu ini *player* dapat mengatur *volume* game sesuai keinginan. Selanjutnya menu keluar, digunakan untuk keluar pada game.

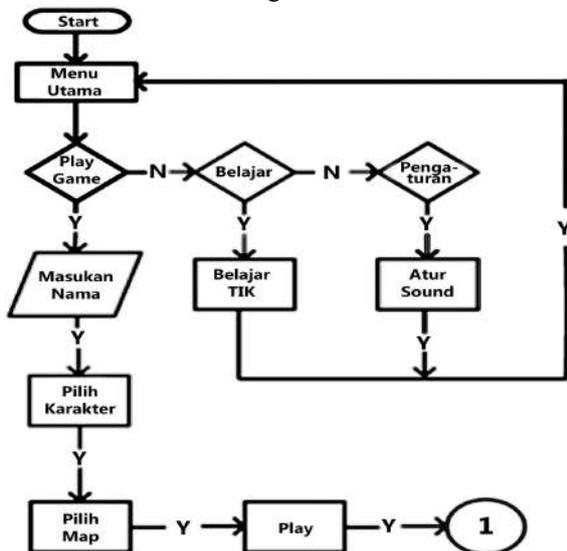
### B. Design

Pada tahap pengembangan *design* merupakan tahap merencanakan desain tampilan dan alur permainan game Monopoli TIK. Pembuatan alur game dan perancangan game menggunakan *flowchart* dan *storyboard*.

### 1. Perancangan *Gameplay* Monopoli

Pada game Monopoli edukasi terdapat 2 pemain dan maksimal 3 karakter yang dapat dipilih sesuai dengan keinginan. Pemain pertama akan memainkan game dengan kehendak sendiri sedangkan komputer akan dijalankan secara otomatis. Setiap pemain akan dinyatakan kalah jika pemain tidak dapat membayar sewa atau denda dan dinyatakan bangkrut atau kalah. *Player* dan *computer* akan bergiliran untuk memainkan game dengan menekan tombol *roll* untuk mengacak dadu.

Pemain atau *computer* yang berhenti di petak kesempatan atau dana umum nantinya akan diberikan pertanyaan edukasi mengenai TIK, jika benar maka akan diberikan hadiah berupa uang dari bank dan jika salah tidak akan mendapatkan apapun. Gbr 2 merupakan *flowchart* permainan game Monopoli yang menjelaskan proses atau alur memulai game.



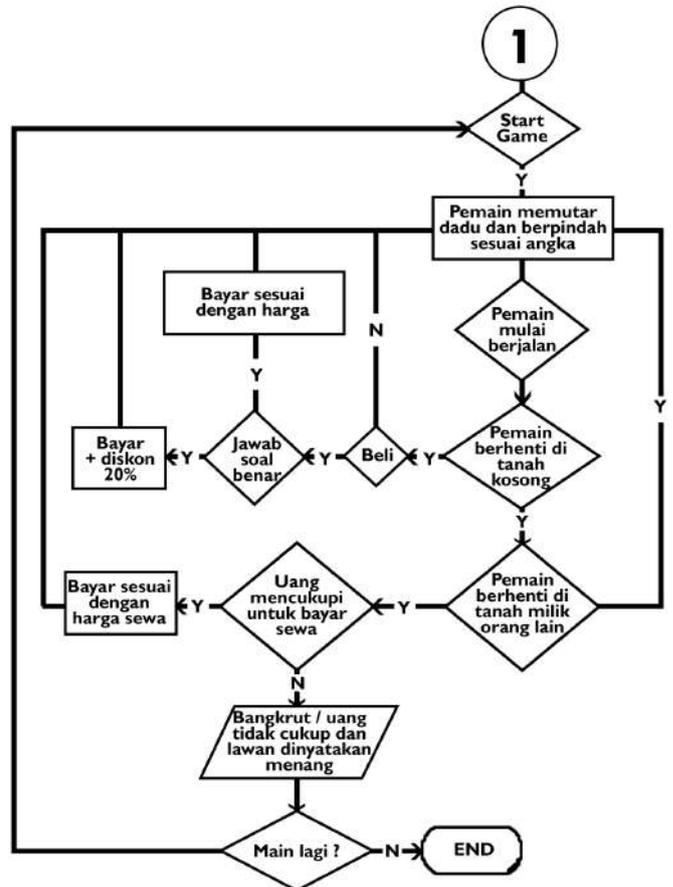
Gbr 2. Alur Memulai Game

Gbr 2 menunjukkan alur untuk memulai game. Berikut merupakan penjelasan dari alur memulai game.

- 1) Terdapat 4 menu utama yaitu *Play Game*, *Belajar*, dan *Pengaturan*.
- 2) Menu *Play Game* digunakan untuk memulai game, pertama *user* harus memasukkan nama, kemudian muncul pemilihan karakter antara lain : pesawat, robot, ikan. Setelah memilih karakter, selanjutnya pemilihan map dengan 3 pilihan meliputi jawa timur, jawa tengah, dan jawa barat. Setelah itu *user* dapat mulai memainkan game.
- 3) Menu *Belajar* berisi soal-soal dan pembahasan jawaban yang terdapat dalam game. Soal-soal dan pembahasan ini meliputi soal Teknologi Informasi dan Komunikasi (TIK).

- 4) Menu *Pengaturan* digunakan untuk mengatur besar kecilnya volume game.

Pada Gbr. 3 berikut merupakan *flowchart* permainan game Monopoli yang menjelaskan alur *gameplay* dari game.

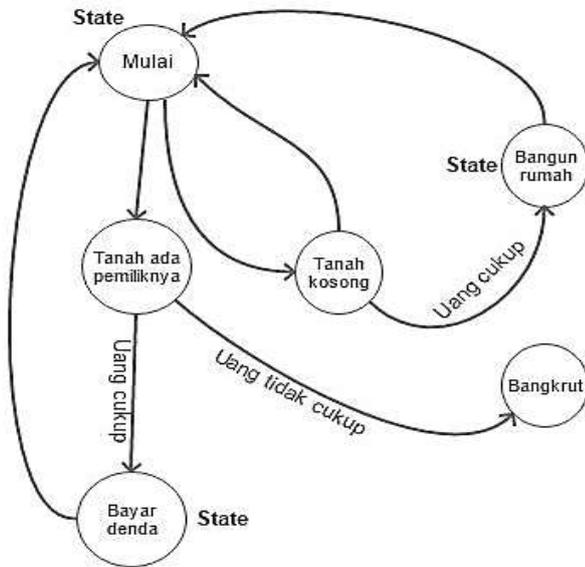


Gbr 3 Alur *Gameplay* Monopoli

Gbr 3 menunjukkan alur *Gameplay* Monopoli. Berikut merupakan penjelasan dari alur *Gameplay* Monopoli.

- 1) Dalam game monopoli ini, *user* akan memulai game dengan melawan komputer.
- 2) *User* mengawali mengacak dadu untuk menentukan langkah karakter ke petak sesuai nilai dadu, dilanjutkan giliran komputer.
- 3) Jika *user* berhenti pada petak yang belum ada pemiliknya. Maka *user* dapat membeli petak. Dalam membeli petak kosong, terdapat penawaran berupa pertanyaan TIK untuk mendapatkan potongan harga 20%. Namun jika salah menjawab pertanyaan, harus membayar dengan harga yang ditentukan.
- 4) Jika *user* berhenti pada petak milik komputer, diwajibkan membayar sewa. Uang yang dimiliki harus cukup untuk membayar harga sewa. Sedangkan bila uang tidak mencukupi, *user* dinyatakan bangkrut atau kalah dalam game.

Gbr 4 merupakan alur pemodelan AI *Finite State Machine* untuk menentukan keputusan NPC atau musuh yang berjalan dalam game.



Gbr 4. Alur FSM pada NPC atau musuh

Pada Gbr 4 terdapat beberapa *state* yang diterapkan pada NPC atau karakter musuh yaitu keliling, bayar denda dan membeli. *State* awal adalah memulai untuk berjalan "keliling" map sesuai dengan angka dadu yang bergantian dengan *player*. Jika komputer berhenti pada tanah kosong NPC akan melakukan *state* "membeli" tanah jika uang cukup dan jika uang tidak cukup akan kembali ke *state* "keliling" yang menunggu giliran *player* selesai. Sedangkan jika NPC berhenti pada tanah yang dimiliki user, maka NPC akan melakukan *state* "bayar denda" jika uang cukup, dan jika uang tidak cukup maka NPC dipaksa bangkrut.

Gbr 5 menunjukkan alur dari Algoritma *Fisher-Yates Shuffle*.



Gbr 3. Alur Algoritma Fisher-Yates

Algoritma *Fisher-Yates Shuffle* bekerja dengan menentukan angka N untuk range 1-N terlebih dahulu. Kemudian, menentukan angka acak k dari range 1-N. Angka acak k harus memenuhi syarat yaitu tidak boleh 0 dan melebihi nilai N. Selanjutnya, angka terakhir range 1-N ditukar dengan posisi k di tempat yang berbeda. Setelah itu, angka N diatur ulang,  $N=N-1$ . Apabila angka N bernilai 0, maka proses pengacakan berhenti.

Penjelasan lebih jelas mengenai Algoritma *Fisher-Yates Shuffle* dapat dilihat pada analisis algoritma ini. Analisis Algoritma *Fisher-Yates Shuffle* diharapkan dapat memberikan gambaran konsep dari algoritma acak (*shuffling*) ketika diterapkan pengacakan soal pada game. Analisis Algoritma Fisher-Yates Shuffle dijelaskan menggunakan *array* seperti pada Gbr 6 berikut.

Kondisi Array					Satu data acak yang bisa diambil	Keterangan
A	B	C	D	E	C	Semua data disebelah kanan C bergeser satu kali ke kiri
0	1	2	3	4	2	
A	B	D	E		Isikan data selain yang sudah diambil. Berikutnya, nomor indeks dapat disesuaikan lagi.	Dan begitu seterusnya hingga data tinggal satu buah
0	1	2	3			

Gbr 4. Algoritma *Fisher-Yates* menggunakan Array

Dari Gbr 6, diketahui terdapat array yang berisi 5 buah data. Dari kelima array tersebut diberi indeks atau nomor dimulai dari 0 sampai 4. Satu buah array diambil secara acak. Selanjutnya, array baru dibuat dengan menimpa array lama. Array baru yang dibuat berukuran N-1 dengan nomor indeks yang disesuaikan kembali.

### C. Material Collecting

Proses pengumpulan bahan merupakan tahap pengumpulan materi pembelajaran sesuai yang dibutuhkan. Berikut bahan-bahan yang diperlukan :

- Soal : Ebook atau buku pelajaran TIK anak SD kelas 5.
- Gambar : gambar yang digunakan ada dua jenis, untuk menjadi spesifik 2D dan 3D yang kemudian digunakan pada game sebagai simbol grafik.
- Audio : dokumen suara yang digunakan adalah jenis dan .mp3 .wav. Dokumen suara dapat diperoleh melalui situs gratis.
- Button : Berfungsi sebagai tombol petunjuk arah untuk menuju dalam dalam permainan.
- Perangkat lunak : persyaratan untuk pemrograman yang diperlukan sebagai bantuan untuk membuat game seperti Unity Game Engine, Adobe Photoshop.
- Hardware : Laptop ASUS dengan Processor Intel Core-i3 4005, RAM 4,00 GB.

### D. Assembly

Pembuatan game edukasi dimulai dengan pemodelan boardgame dan objek-objek seperti dadu, penjara dan rumah menggunakan software *Blender*. Setelah proses pemodelan selesai, maka akan dilakukan proses pembuatan game menggunakan software *Unity Game Engine*. Didalam proses pembuatan game juga dilakukan proses penulisan *source code* dan implementasi objek ke dalam game.

### E. Testing

Tahap pengujian (*testing*), tahap ini akan dilakukan setelah menyelesaikan tahap pembuatan (*assembly*). Tahap ini perlu dilakukan untuk mengetahui kinerja sebuah sistem yang telah dibuat. Serta untuk mengetahui kekurangan dari sistem yang selanjutnya diperbarui sehingga sistem akan sesuai dengan yang direncanakan.

Pengujian dilakukan dengan menjalankan aplikasi atau program, kemudian akan dianalisis penerapan algoritma dalam game yang dibangun. Tahap pengujian menu game menggunakan teknik *Black-Box testing*. Pengujian Black Box berfokus pada syarat fungsional dari perangkat lunak. Pengujian dilakukan dengan dua tahapan, yaitu Alpha dan Beta.

## III. HASIL DAN PEMBAHASAN

### A. Spesifikasi Sistem

Untuk mengembangkan game *Monopoli TIK* dengan menerapkan Algoritma *Fisher-Yates Shuffle* membutuhkan dukungan perangkat lunak berikut :

- Processor : Intel Core i3-4005 1.7 GHz.
- VGA Card : Nvidia Geforce GT 930M
- RAM 4 GB
- HDD 500 GB

Sedangkan spesifikasi minimum perangkat untuk menjalankan game *Monopoli TIK* adalah sebagai berikut :

- Sistem Operasi : Android V7.0 Nougat
- Prosesor : Octa-Core 1.5 Ghz Cortex A53
- Memori : Internal 32 GB dan RAM 3 GB
- Layar : 5.5 Inch

### B. Tampilan Antarmuka

Berikut adalah tampilan antarmuka permainan “Monopoli TIK”.



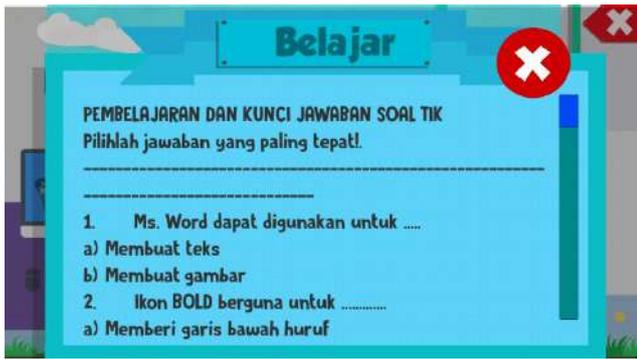
Gbr 7. Tampilan Membuka Permainan

Gbr 7 adalah tampilan ketika permainan “Monopoli TIK” dibuka pada perangkat. Permainan akan terbuka ketika *Loading* sudah mencapai 100 %.



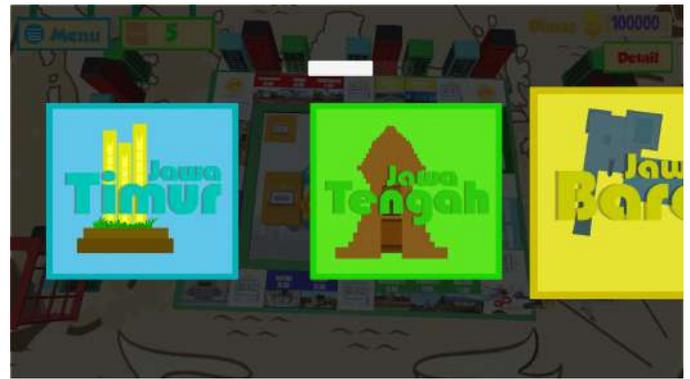
Gbr 8. Tampilan Menu Utama

Gbr 8 merupakan tampilan menu utama pada permainan *Monopoli TIK*. Pada tampilan tersebut terdapat tiga (3) pilihan menu yaitu *Main*, *Belajar* dan *Pengaturan*. Sedangkan tombol Silang (X) pada pojok kanan atas berfungsi untuk menutup aplikasi atau keluar dari permainan *Monopoli TIK*.



Gbr 9. Tampilan Menu "Belajar"

Gbr 9 merupakan menu "Belajar". Pada menu "Belajar" ini terdapat pilihan soal beserta kunci jawabannya.



Gambar 12. Tampilan pilihan Peta

Gbr 12 menampilkan pilihan peta pada permainan *Monopoli TIK*, diantaranya adalah Jawa Timur, Jawa Tengah dan Jawa Barat.



Gbr 10. Tampilan Menu "Pengaturan"

Gbr 10 merupakan tampilan untuk menu "Pengaturan". Pada tampilan ini pengguna dapat mengatur *Volume* pada permainan *Monopoli TIK*.



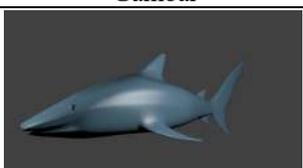
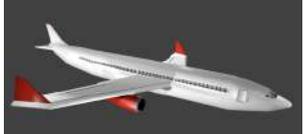
Gbr. 13 Tampilan Permainan Monopoli TIK



Gbr. 11 Tampilan Menu "Main".

Gbr 11 adalah tampilan pada menu "Main". Pada menu ini pemain harus memasukkan nama pada kolom yang telah disediakan. Lalu pada submenu *Dice* pemain dapat memilih warna dadu dan dilanjutkan dengan pemilihan karakter. Setelah selesai ketuk *Play* untuk memulai permainan *Monopoli TIK*.

TABEL I  
 NAMA KARAKTER PADA GAME

No.	Nama	Gambar
1.	Karakter Hiu	
2.	Karakter Luffy	
3.	Karakter Pesawat	

4.	Karakter Robot	
----	----------------	---

Gbr 16 merupakan tampilan pilihan untuk membeli tanah, jika menjawab soal dengan benar mendapatkan diskon 20% dari harga asli dan jika menjawab soal salah pembelian tanah tetap harga aslinya.

### C. Pengujian

Tahap pengujian selesai untuk menemukan presentasi kerangka kerja yang dibuat. Pengujian selesai dengan menjalankan aplikasi atau program, maka, pada saat itu, akan dibedah penggunaan perhitungan dalam permainan yang dibuat. Tahap pengujian menu permainan menggunakan prosedur pengujian Black-Box. Pengujian penemuan berpusat di sekitar kebutuhan produk yang berguna. Tabel II dan III menunjukkan rencana pengujian menggunakan prosedur pengujian Black-Box..

### Hasil Uji Coba Kebutuhan Fungsionalitas

TABEL III  
PENGUJIAN TAMPILAN DAN FUNGSI TOMBOL PADA GAME

No	Komponen yang diuji	Butir Uji	Validitas Y/T
1	Tampilan awal	Terdapat tampilan menu utama	Y
2	Menu Main	User dapat menekan tombol menu Main untuk memulai permainan	Y
3	Menu Belajar	Terdapat tampilan materi dan pembahasan soal	Y
4	Menu Pengaturan	User dapat menekan tombol untuk membesarkan dan mengecilkan volume	Y
5	Menu memasukkan nama user	User dapat memasukkan nama untuk memulai permainan	Y
6	Menu pilih karakter	User dapat memilih karakter yang sudah tersedia sesuai yang diinginkan	Y
7	Menu pilih map	User dapat memilih map yang sudah tersedia sesuai yang diinginkan	Y
8	Tampilan Soal	User dapat memilih jawaban pada soal yang di tampilkan	Y
9	Tampilan harga wilayah	Terdapat Tombol harga-harga rumah yang bisa di bangun	Y
10	Tampilan masuk penjara	Terdapat 2 tombol pilihan untuk keluar penjara jawab soal dan bayar denda	Y
11	Tampilan bayar denda	Terdapat tampilan denda saat user berhenti di wilayah komputer	Y



Gbr 14. Tampilan soal pada game

Gbr 14 merupakan tampilan soal pada game yang berisi dua tombol pilihan untuk menjawab pertanyaan yang benar dari soal.



Gbr 15. Tampilan bonus setelah menjawab pertanyaan dengan benar

Gbr 15 merupakan tampilan yang menampilkan bonus Rp.2000 setelah menjawab pertanyaan dengan benar.



Gbr 16. Tampilan untuk pembelian tanah

12	Tampilan uang user dan komputer	Terdapat tampilan uang yang dimiliki user dan komputer	Y
13	Tampilan Detail Sertifikat (sertifikat tanah/ wilayah yang dimiliki)	Terdapat tampilan wilayah-wilayah yang dimiliki user	Y
14	Menu Exit	Menekan tombol exit untuk keluar dari game	Y
15	Dadu	Terdapat tampilan objek dadu untuk berjalanya permainan	Y
16	Tombol memutar dadu	Terdapat tombol untuk memutar dadu	Y
17	Tombol Akhiri giliran	Terdapat tombol untuk mengakhiri giliran	Y
18	Tombol Pause dan Resume	Terdapat tombol pause untuk menjeda permainan dan tombol Resume untuk memulai permainan setelah di Pause	Y
19	Tombol Restat	Terdapat tombol Restart untuk memulai permainan dari awal	Y
20	Tampilan Menang User atau Komputer	Terdapat tampilan hasil permainan user yang menang atau komputer yang menang	Y

Tabel II menunjukkan hasil pengujian menggunakan teknik *Black-Box testing* pada tampilan dan fungsi tombol dalam game. Berdasarkan total keseluruhan 20 rancangan pengujian didapatkan nilai persentase *score* validitas di setiap komponen uji yang tinggi yaitu sebesar 100%.

TABEL IIIII  
PENGUJIAN NPC PADA GAME

No	Komponen yang diuji	Butir Uji	Validitas Y/T
1	NPC state "keliling"	NPC keliling berjalan map sesuai angka dadu	Y
2	NPC state "bayar denda"	NPC membayar denda jika berhenti pada tanah milik user/pemain	Y
3	NPC state "Membeli"	NPC memiliki uang yang cukup untuk membeli tanah yang kosong	Y

Tabel III menunjukkan hasil pengujian menggunakan teknik *Black-Box testing* pada NPC dalam game. Berdasarkan total keseluruhan 3 rancangan pengujian didapatkan nilai persentase *score* validitas di setiap komponen uji yang tinggi yaitu sebesar 100%.

TABEL IVV  
PENGUJIAN ALGORITMA FISHER-YATES SHUFFLE

Pengujian Ke-	Index Pertanyaan Yang Akan Diacak	Hasil Pengacakan Dari Algoritma Fisher-Yates Shuffle
1	0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39	30,25,8,33,3,23,36,38,21,22,2,29,19,1,16,39,17,10,31,18,0,12,24,7,4,34,15,20,35,32,28,9,14,6,37,5,26,11,13,27
2	0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39	21,26,17,16,23,38,0,6,25,13,28,27,32,10,34,12,29,39,37,14,31,2,5,19,3,1,22,24,36,20,8,35,4,30,18,33,15,7,911,
3	0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39	5,3,38,25,7,19,2,31,1,32,34,22,13,23,27,24,33,39,17,30,12,20,9,26,18,14,37,16,11,15,8,21,6,29,0,4,35,28,10,36
4	0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39	8,14,13,28,32,22,9,35,39,30,37,10,15,31,11,19,34,27,23,7,36,33,26,6,4,5,2,3,38,25,16,1,12,24,0,17,29,18,20,21
5	0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39	3,9,22,26,38,1,30,36,4,39,8,6,34,21,15,2,5,11,25,14,16,33,10,20,35,0,24,18,13,28,27,17,29,19,37,12,23,7,31,32

Dari hasil pengujian Algoritma *Fisher-Yates Shuffle*, didapatkan hasil bahwa Algoritma berjalan dengan baik sesuai dengan kegunaannya yaitu untuk melakukan *shuffle* atau pengacakan. Dalam percobaan yang dilakukan sebanyak 5 kali dengan waktu yang berbeda, didapatkan hasil pengacakan soal yang berbeda di setiap permainan. Dari hasil percobaan ini Algoritma *Fisher-Yates Shuffle* berjalan dengan baik pada setiap pengacakannya.

### Hasil Uji Coba Kuesioner

Pengujian juga dilakukan menggunakan perhitungan kuesioner. Perhitungan kuesioner yang dilakukan berdasarkan Skala *Likert*. Hasil uji coba menghasilkan sebuah keputusan yang dinilai dari pernyataan yang diajukan kepada pengguna.

Perhitungan hasil uji coba kuesioner dapat dilihat pada Tabel V berikut.

TABEL VV  
UJI COBA KUESIONER

Pertanyaan	Keterangan	Jumlah Responden
4	Sangat baik	16
	Baik	4
	Netral	0
	Kurang baik	0
	Sangat kurang sekali	0
Total		20

$$\begin{aligned}
 Y &= \text{skor tertinggi Likert} \times \text{total responden} \\
 &= 5 \times 20 \\
 &= 100
 \end{aligned}$$

$$\text{Interval} = 100/5 = 20$$

Penilaian skor berdasarkan *interval score* adalah sebagai berikut:

1. Interval 0% - 19,99% = Sangat (Tidak Setuju/buruk/ kurang sekali)
2. Interval 20% - 39,99% = Tidak Setuju / kurang baik
3. Interval 40% - 59,99% = Netral / cukup
4. Interval 60% - 79,99% = Setuju / baik / suka
5. Interval 80% - 100% = Sangat (Setuju /baik/ suka)

$$\begin{aligned}
 \text{Hasil Akhir} &= \text{Total skor}/Y \times 100 \\
 &= (80 + 16)/100 \times 100 \\
 &= 96/100 \times 100 = 96 \text{ (Sangat Baik)}
 \end{aligned}$$

Berdasarkan hasil uji coba kuesioner didapatkan prosentase pernyataan pertama sampai kedua puluh diantaranya 95%, 97%, 95%, 96%, 96%, 95%, 95%, 95%, 96%, 96%, 84%, 98%, 95%, 97%, 84%, 97%, 95%, 84%, 95%, dan 95%. Sehingga dari total keseluruhan sebanyak 20 responden berpendapat Sangat Baik dalam menggunakan aplikasi ini.

#### IV. KESIMPULAN

Berdasarkan hasil penelitian dengan menerapkan pemodelan AI *Finite State Machine* pada NPC pada game dan Algoritma *Fisher-Yates Shuffle* berfungsi untuk melakukan pengacakan soal pada game Monopoli 3D Teknologi Informasi dan Komunikasi. Algoritma *Fisher-Yates Shuffle* digunakan sebagai metode pengacakan soal pada game agar soal yang tersedia dapat keluar secara acak dan tidak berulang. Sedangkan penerapan pemodelan AI *Finite State Machine* pada NPC digunakan untuk pengambilan keputusan, seperti untuk melakukan transaksi pembelian tanah dan pembayaran denda. Hasil pengujian dari penerapan pemodelan AI *Finite State Machine* pada NPC berjalan dengan baik sesuai yang diharapkan. Sedangkan pada pengujian Algoritma *Fisher-Yates Shuffle* sebanyak 5 kali pada game dengan waktu yang berbeda mendapatkan hasil pengacakan soal yang bervariasi dan tidak berganda. Selain itu, pengujian menggunakan uji coba

kuesioner dengan total keseluruhan 20 responden menghasilkan nilai rata-rata pengguna aplikasi sebesar 80% - 100% (sangat baik). Hal ini membuktikan bahwa penerapan pemodelan AI *Finite State Machine* dan Algoritma *Fisher-Yates Shuffle* sangat cocok untuk diterapkan pada game Monopoli 3D Teknologi Informasi dan Komunikasi.

#### UCAPAN TERIMA KASIH

Ucapan Terima kasih saya kepada Allah SWT yang telah memberi saya kemudahan dan kelancaran dalam penyelesaian penelitian pengembangan game berbasis android. Saya ucapkan juga terimakasih kepada kalian yang telah memberikan dukungan dan membantu dalam penyelesaian artikel ini dengan baik.

#### REFERENSI

- [1] R. Barrera, A. S. Kyaw and T. N. Swe, *Unity 2017 Game AI Programming Third Edition*, Birmingham, B3 2PB, UK.: Packt Publishing Ltd, 2018.
- [2] F. Ganzala, *Perilaku Non Playable Character (NPC) Musuh Pada Game Sepeda Menggunakan Fuzzy State Machine (FuSM)*, Malang: Etheses UIN Maulana Malik Ibrahim Malang, 2016.
- [3] Ekojono, D. A. Irawati, d. L. Affandi and A. N. Rahmanto, "Penerapan Algoritma Fisher-Yates Pada Pengacakan Soal Game Aritmatika," *Prosiding SENTIA 2017 - Politeknik Negeri Malang*, pp. I-91 - I-100, 2017.
- [4] M. Irfan, D. R. Ramdhan, I. S. Nita, T. Priatna and A. R. Atmadja, "Design and Build an Early Childhood Puzzle Educational Game Using the Fisher-Yates Shuffle Algorithm as an Android-Based Scrambler for Snippets," *IEEE Xplore*, 2020.
- [5] D. A. Dermawan, D. F. Suyatno and M. S. Hawari, "SIMULASI STANDARD OPERATIONAL PROCEDURE LABORATORIUM KOMPUTER DENGAN PEMODELAN FINITE STATE MACHINE PADA PERILAKU TEKNISI," *SCAN Jurnal Teknologi Informasi dan Komunikasi*, pp. 7-14, 2019.
- [6] R. F. Rahmat, R. Ramadhan, D. Arisandi, M. F. Syahputra and O. Sheta, "Rangku Alu - A Traditional East Nusa Tenggara Game in Android Platform," *International Conference on Computing and Applied Informatics*, pp. 1-12, 2017.
- [7] Billy, I. Kuswardayan and W. N. Khotimah, "Implementasi Artificial Intelligence pada Game

Defender of Metal City dengan Menggunakan Finite State Machine,” *Jurnal Teknik Pomits*, pp. A640-A645, 2017.

- [8] B. Subaeki and D. Ardiansyah, “IMPLEMENTASI ALGORITMA FISHER-YATES SHUFFLE PADA APLIKASI MULTIMEDIA INTERAKTIF UNTUK PEMBELAJARAN TENSES BAHASA INGGRIS,” *Jurnal Infotronik*, pp. 67-74, 2017.