

Perancangan dan Implementasi *Finite State Automata* pada *Pusheen Cat Maze Game* dengan *Adobe Flash*

Affis Masturina Nisa*¹, Hafidz Kurniansyah ²

^{1,2}Program Studi Manajemen Informatika, Fakultas Teknik, Universitas PGRI Madiun
 e-mail: *¹nisakmsh98@gmail.com, ² hafidzbmc86@gmail.com

Abstrak— Sekarang ini banyak sekali media yang digunakan untuk pembelajaran, khususnya anak – anak, dengan menggunakan media yang menarik, seperti *game*. Selain dapat menambah minat belajar, media *game* ini cukup efektif untuk : (1) mencapai tujuan pembelajaran, (2) merangsang kreativitas, (3) melatih konsentrasi. Penelitian yang dilakukan melalui 4 tahap menurut [1] yaitu identifikasi masalah, perancangan sistem *game* dan automata, implementasi sistem, pengujian sistem dan analisis hasil pengujian. *Finite State Automata* adalah mesin abstrak berupa sistem model matematika dengan masukan dan keluaran diskrit yang dapat mengenali bahasa paling sederhana (bahasa reguler) dan dapat diimplementasikan secara nyata. Automata digunakan dalam perancangan *software* dan *hardware* dalam perancangan sebuah *game*. Penelitian yang dihasilkan berupa aplikasi *game maze* (labirin) dengan *Adobe Flash*.

Kata kunci— *Finite State Automata, Maze Game, Education Game, Adobe Flash*

I. PENDAHULUAN

Seiring dengan berkembangnya jaman, teknologi kini makin diperbaharui untuk mempermudah pekerjaan manusia, dengan teknologi yang semakin maju dan berkembang maka kehidupan masyarakat pun ikut berubah. Mulai dari kehidupan ekonomi, politik, sosial dan budaya. Dari kemajuan inilah pula muncul juga masalah “ketergantungan” masyarakat terhadap kecanggihan teknologi. Salah satu teknologi yang mungkin cukup membuat pengaruh yang kuat yaitu *game*, *game* atau permainan merupakan salah satu industri elektronik *digital* yang kini berkembang pesat tidak terkecuali di Indonesia. *Game* sendiri menjadi idaman diberbagai kalangan masyarakat yang bertujuan untuk menghilangkan rasa jenuh dari rutinitas sehari-hari.

Adanya pandangan miring masyarakat terhadap *game* tidak semuanya benar, terlebih lagi para *developer game* masa kini lebih mementingkan aspek olah otak untuk mengembangkan kinerja otak, khususnya anak-anak sehingga mereka akan lebih kreatif dan berkonsentrasi dalam menyelesaikan tantangan dalam sebuah *game* meskipun ada juga yang hanya mementingkan aspek hiburan saja.

Sebagai salah satu ilmu komputer, teori Bahasa dan Automata dijadikan dasar oleh model serta gagasan mendasar dalam membuat *game* ini. Dengan menggunakan metode ini, aplikasi “*Maze Game*” yang

bergenre *arcade* ini kita dapat memberikan kecerdasan buatan agar dapat memiliki kecerdasan yang sama dengan pemain sehingga *game* menjadi lebih menantang.

Berdasarkan latar belakang pada masalah tersebut, dilakukan penelitian dengan judul “Perancangan dan Implementasi *Finite State Automata* pada *Pusheen Cat Maze Game* dengan *Adobe Flash*”, yang lebih bersifat pada edukasi dalam permainan ini. Didapatkan tinjauan literatur sebagai berikut:

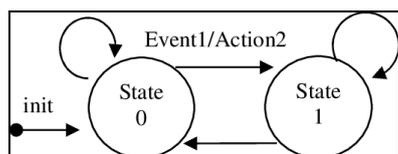
No	Judul	Terbitan, Tahun	Perbedaan Penelitian	
			Literatur	Penulis
1	Perancangan dan Implementasi <i>Finite State Automata</i> Pada <i>Game Quiz Ular</i> Berbasis <i>Android</i>	Y. Yohanes, http://repository.uksw.edu/bitstream/123456789/6366/2/T1_672007111_Full%20text.pdf , Tangga 2013	Studi Kasus yang dirancang adalah <i>Game Quiz Ular</i> Berbasis <i>Android</i>	Studi Kasus yang dirancang adalah <i>Pusheen Cat Maze Game</i> dengan <i>Adobe Flash</i>

II. LANDASAN TEORI

Permainan (*game*) adalah salah satu implementasi pada ilmu komputer. Perkembangan yang pesat ini dikarenakan mayoritas pengguna komputer yang

menghabiskan sebagian besar waktu mereka di depan komputer dalam *program* permainan. Salah satu permainan yang diminati adalah permainan Labirin. Permainan Labirin adalah permainan mencari jalan keluar yang bertujuan menentukan jalur yang tepat untuk mencapai tujuan yang telah ditetapkan. Selama proses penentuan jalur tersebut, jika menemui jalan buntu maka akan dilakukan *backtrack* sampai kembali menemukan jalur yang tepat untuk mencapai tujuan.

Finite State Automata atau biasa disebut *FSA*, merupakan sebuah model matematika dari suatu sistem yang menerima suatu *input* dan menghasilkan sebuah *output* diskrit. *FSA* memiliki *state* yang banyaknya terbatas, dan dapat dipindahkan dari satu *state* ke *state* lain[2]. *State* adalah kondisi, keadaan, atau kedudukan. Prinsip kerja *Finite State Automata* sendiri adalah dengan cara mesin pembaca perintah membaca memori masukan yang berupa *tape* yaitu 1 karakter disetiap menggunakan *head* baca yang dikendalikan oleh kontak kendali *state* berhingga dimana pada mesin tersebut terdapat sejumlah *state* berhingga. *Finite State Automata* selalu dalam kondisi yang disebut dengan *state* awal pada saat *finite automata* mulai membaca *tape*[3]. Perubahan dari *state* terjadi pada mesin saat sebuah karakter selanjutnya dibaca. Ketika sebuah *head* telah sampai pada akhir-an *tape* dan dalam kondisi *state* akhir, maka *string* yang terdapat pada *tape* akan dikatakan diterima *Finite Automata*.



Gambar 1. Diagram *State* Sederhana
(Sumber: Setiawan, 2006)

Gambar 1 memperlihatkan *FSA* dengan dua buah *state* dan dua buah *input* serta empat buah *output* yang berbeda seperti terlihat pada gambar, ketika sistem mulai dihidupkan, sistem akan bertransisi menuju *State0*, pada keadaan ini sistem akan menghasilkan *Action1* jika terjadi masukan *Event0*, sedangkan jika terjadi *Event1* maka *Action2* akan dieksekusi kemudian sistem selanjutnya bertransisi ke keadaan *State1* dan seterusnya[4]. *Game* berasal dari Bahasa Inggris yang jika diartikan menjadi permainan. Samuel Henry mengatakan bahwa *Game* merupakan bagian tak terpisahkan dari keseharian anak, sedangkan sebagian orang tua menuding *game* sebagai penyebab nilai anak

turun, anak tak mampu bersosialisasi, dan tindakan kekerasan yang dilakukan anak”[5]. Setiap *game* memiliki peraturan main yang berbeda dari beberapa *game* lain sehingga ragam *game* semakin banyak dan beragam. Fungsi dari *game* sendiri sebagai penghilang stress dan rasa jenuh akan aktifitas sehari-hari.

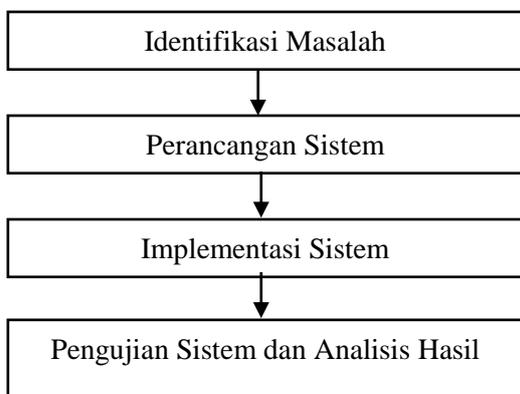
Tujuan dalam *Game* adalah sebagai mendidik, menghibur para pemain *game*. Dalam sebuah buku berjudul “*the art of computer game design*”, sebuah rancangan *game* merupakan suatu aktivitas kompleks yang dapat diturunkan ke prosedur formal. Ada beberapa langkah dalam merancang *game* yaitu: (1) Memilih tujuan dan topik, (2) Penelitian dan persiapan, (3) Tahap *design*, (4) Tahap pra-pemrograman, (5) Tahap pemrograman, (6) Tahap *playtesting*, dan (7) *Post-mortem* [6]. Permainan *Maze* (Labirin) adalah sebuah permainan mencari jalan keluar yang tujuannya untuk menentukan jalur yang tepat untuk mencapai tujuan yang telah ditetapkan. Selama proses penentuan jalur tersebut, jika menemui jalan buntu maka akan dilakukan proses *backtrack* sampai kembali menemukan jalur tepat untuk mencapai tujuan[7].

Adobe Flash atau sebelumnya *Macromedia Flash* merupakan *software* multifungsi. *Software* ini mempermudah dalam pembuatan animasi *web*, selain itu *action script* pada *Adobe Flash* juga dapat dimanfaatkan menjadi *game* yang mudah dan efektif. *Adobe Flash* menyuguhkan 5 versi *action script* yang disesuaikan kebutuhan pengguna. Versi – versi *action script* pada *Adobe Flash* yaitu (1) *Action Script* 1.0 merupakan versi sederhana dan digunakan pada kebanyakan versi *Flash Lite Player*, (2) *Action Script* 2.0 secara praktis lebih mudah dipelajari dan dipahami. Kinerjanya sedikit lebih cepat dari versi sebelumnya sehingga sesuai digunakan untuk proyek yang tidak membutuhkan komputasi yang kompleks, (3) *Action Script* 3.0 versi ini menggunakan konsep pemrograman berorientasi objek sehingga dapat mengeksekusi perintah secara cepat, (4) *Flash Lite* 1.x, 2.x, dan 3.x. Merupakan versi *Action Script* yang khusus dirancang untuk menjalankan aplikasi *Flash Lite* 1.x pada perangkat seluler [8].

III. METODE

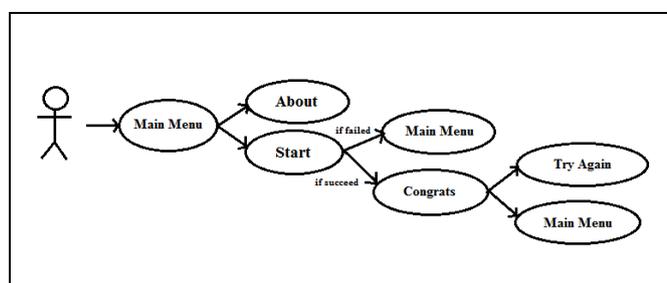
Ada 4 tahapan dalam merancang *Finite State Automata* pada sebuah *game*. Tahapannya antara lain : (1) Identifikasi Masalah, (2) Perancangan Sistem, (3)

Implementasi Sistem, dan (4) Pengujian Sistem dan Analisis Hasil Pengujian.



Gambar 2. Tahapan Penelitian

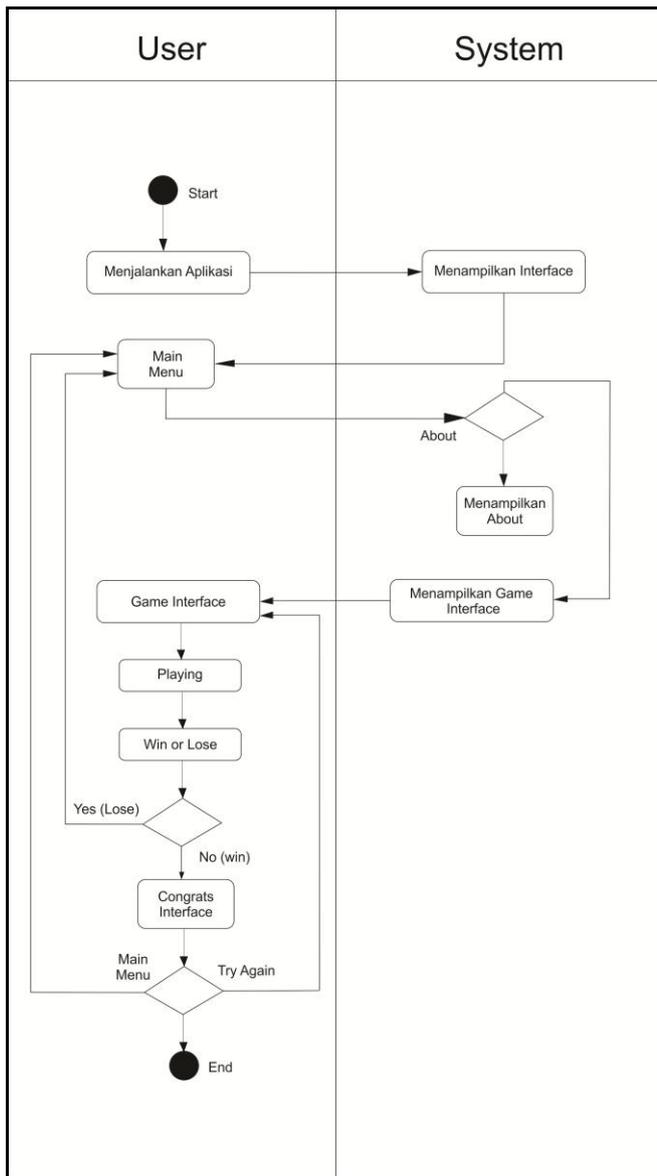
Pada gambar 2 tahapan penelitian, dapat dijelaskan sebagai berikut. Tahapan 1 : Identifikasi Masalah yaitu mengumpulkan beberapa data mengenai pembuatan game pada Adobe Flash. Tahapan 2 : Perancangan Sistem yaitu meliputi proses perancangan game dan elemen game. Tahapan 3 : Implementasi Sistem yaitu game yang sudah dirancang di uji. Tahapan 4 : Pengujian Sistem dan Analisis Hasil Pengujian yaitu pengujian ulang dan menganalisis hasil pengujian itu dihubungkan dengan sistem *Automata*[9]. *Use Case Diagram* berfungsi untuk menggambarkan kebutuhan sistem dari sudut pandang *user*, memfokuskan pada proses komputerisasi, menggambarkan hubungan antara *use case* dan *actor use case*, menggambarkan proses sistem [10].



Gambar 3. Use Case Diagram

Gambar 3 menunjukkan *Use Case Diagram*. Player berada pada *Main Menu*, dimana ada opsi *About* dan *Start*. *About* adalah *profil* dari peneliti, sedangkan *Start* untuk memulai *game*. Selanjutnya, jika gagal maka otomatis akan mengarah ke *Main Menu*, sedangkan jika

sukses, akan tampil ucapan *Congrats*. Pada *Congrats* disini, ada opsi *Try Again* dan *Main Menu*. *Try Again* adalah mengulang lagi permainan sedangkan *Main Menu* akan menuju kembali ke menu awal. *Activity Diagram* menggambarkan aliran aktivitas dalam sistem yang sedang dirancang, bagaimana masing – masing aliran dimulai, apa keputusan yang terjadi dan bagaimana aktivitas tersebut berakhir. Gambar 4 menunjukkan *activity diagram* saat aplikasi itu dijalankan. *Player* yang menjalankan aplikasi, aplikasi menampilkan *menu*, lalu *player* dapat memilih langsung *Start* untuk bermain atau melihat profil peneliti melalui *About*. Jika *game* yang dimainkan gagal, maka akan kembali ke *main menu*, sedangkan jika *game* yang dimainkan sukses maka akan berlanjut ke tampilan *Congrats* dimana di tampilan tersebut ada tombol *Try Again* itu mencoba *game* itu kembali, dan juga ada tombol *Main Menu* untuk kembali ke menu utama.



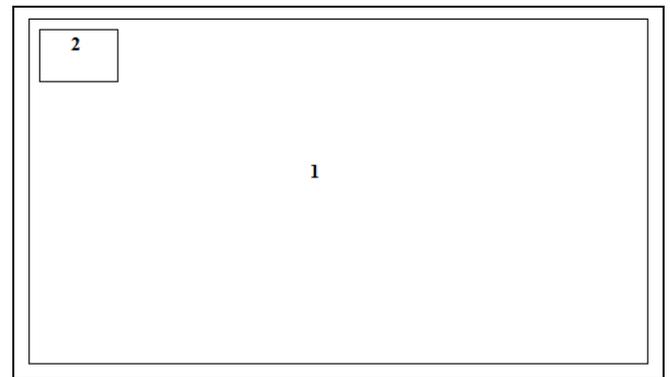
Gambar 4. Activity Diagram

Rancangan antarmuka *Game Pusheen Cat Maze* ditunjukkan pada Gambar 5.



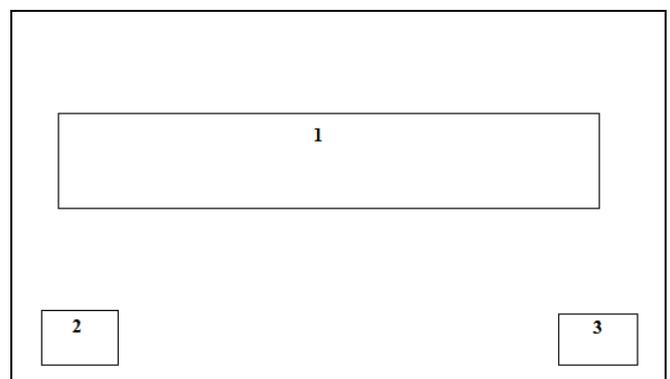
Gambar 5. Main Menu

Pada Gambar 5 menunjukkan *Main Menu* atau *Menu Awal* pada game saat dijalankan. Ada 2 tombol pada *Main Menu*, yaitu (1) *About* yang berisi identitas peneliti, (2) *Start* untuk memulai permainan. Ketika *player* memilih *Start*, *player* akan masuk ke permainan pada Gambar 6.



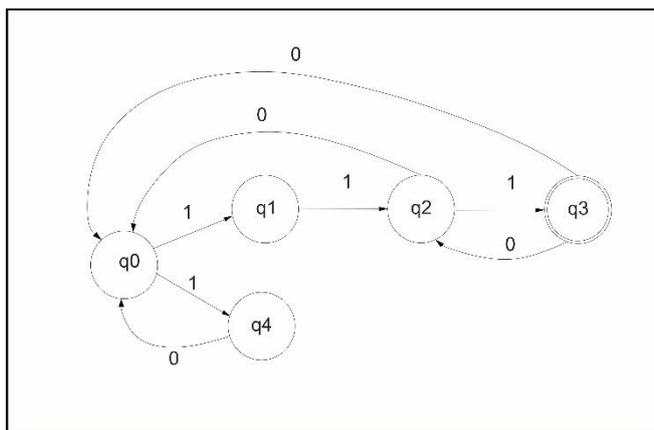
Gambar 6. Game Interface

Pada Gambar 6 menunjukkan *Game Interface*. Dijelaskan sebagai berikut: (1) *Interface* untuk Labirin, (2) Titik Hitam (*Finish*). Jika mengenai luar garis labirin, otomatis akan kembali ke *Main Menu*, sedangkan jika sukses, akan mengarah pada tampilan *Congrats* seperti pada Gambar 7.



Gambar 7. Tampilan *Congrats*

Pada Gambar 7 menunjukkan Tampilan *Congrats*. Dijelaskan sebagai berikut: (1) Tulisan *Congrats*, (2) *Try Again*, (3) *Main Menu*.



Gambar 8. Rancangan FSA Pusheen Cat Maze Game



Gambar 9. Menu Utama Game

Pada Gambar 8 menunjukkan Rancangan FSA, Konfigurasi DFA tersebut sebagai berikut:

$Q = \{Q0, Q1, Q2, Q3, Q4\}$

$\Sigma = \{1, 0\}$

$S = Q1$

$F = Q3$

δ (fungsi transisi)

Tabel 1. Tabel Konfigurasi DFA

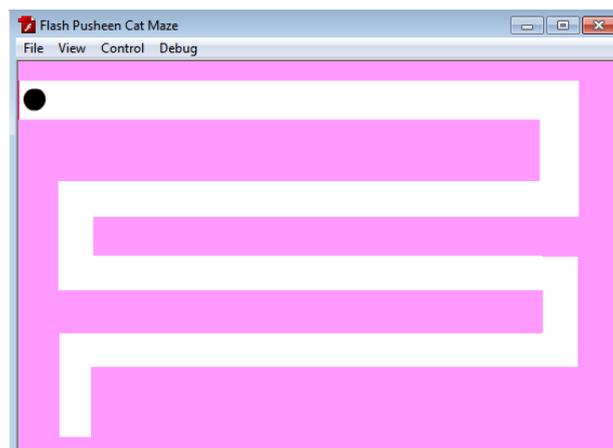
δ	0	1
Q0	-	Q1
Q1	-	Q2
Q2	Q0	Q3
Q3	Q0, Q2	-
Q4	Q0	Q0

Ada 5 state yaitu Main Menu (Q0), Start (Q1), Gameplay (Q2), Congrats (Q3), About (Q4). Sedangkan inputnya adalah 1 dan 0.

IV. HASIL

Game Pusheen Cat Maze merupakan Game berbasis Flash untuk pemain tunggal. Di dalam game ini, player bermain dengan cara mengarahkan kursor menuju lokasi akhir melalui labirin. Jika kursor terkena luar jalur labirin, otomatis gagal. Berikut adalah rancangan Game Pusheen Cat Maze.

Gambar 5 menunjukkan tampilan awal yaitu Menu Utama. Ada 2 tombol pilihan, yaitu (1) About dan (2) Start. Ketika player memilih Start maka akan langsung ke permainan pada Gambar 6.



Gambar 10. Game Interface

Gambar 6 menunjukkan game interface, dimana terdapat labirin dan titik hitam. Kursor harus menuju ke lokasi akhir (titik hitam) agar sukses. Jika gagal maka akan kembali ke Menu Utama (Main Menu). Jika sukses, akan menuju ke tampilan Congrats seperti pada Gambar 7.



Gambar 11. *Congrats Interface*

Pada Gambar 7 menunjukkan *Congrats Interface*, dimana terdapat tampilan tulisan *Congrats* karena sukses dalam permainan. Dan juga 2 tombol pilihan, (1) *Try Again* untuk mencoba permainan itu kembali, dan (2) *Main Menu* untuk menuju kembali ke Menu Utama. Hasil pengujian dari implementasi *Finite State Automata* dijelaskan sebagai berikut.

Tabel 2 Tabel Hasil Pengujian Implementasi FSA

No	Status	Hasil Pengujian
1.	<i>Main Menu</i>	Berjalan dengan baik
2.	<i>Start</i>	Berjalan dengan baik
3.	<i>Gameplay</i>	Berjalan dengan baik
4.	<i>Congrats</i>	Berjalan dengn baik

V. KESIMPULAN

Berdasarkan hasil penelitian, dan pembahasan yang dilakukan dapat ditarik kesimpulan sebagai berikut: (1) *FSA* dapat membantu pengembangan *Game* yang dijalankan, (2) *Game Pusheen Cat Maze* membantu pemain khususnya anak – anak untuk melatih konsentrasinya. Saran pengembangan yang dapat diberikan untuk penelitian lebih lanjut adalah: (1) Grafik game ditingkatkan menjadi 3D, (2) Mengembangkan sistem pada *game* seperti *reward*, *level* dsb.

DAFTAR PUSTAKA

[1] Y. Yohannes, “Perancangan dan Implementasi Finite State Automata pada Game Quiz Ular Tangga Berbasis Android,” 2013.
 [2] A. Farmanbar, S. Firouzi, S.-J. Park, K. Nakai, K. Uchamaru, dan T. Watanabe, “Multidisciplinary insight into clonal expansion of HTLV-1-infected cells in adult T-cell leukemia via modeling by

deterministic finite automata coupled with high-throughput sequencing,” 2017.
 [3] D. Nagamouttu, I. Egambaram, M. Krishnan, dan P. Narasingam, “A verification strategy for web services composition using enhanced stacked automata model,” 2015.
 [4] D. Liu, Z. Huang, Y. Zhang, X. Guo, dan S. Su, “Efficient Deterministic Finite Automata Minimization Based on Backward Depth Information,” *PLoS One*, vol. 11, no. 11, hal. e0165864, Nov 2016.
 [5] M. Mustofa, S. Sidiq, dan E. Rahmawati, “PENERAPAN FINITE STATE MACHINE UNTUK PENGENDALIAN ANIMASI PADA VIDEO GAME RPG NUSANTARA LEGACY,” *J. Sist. Komput. Musirawas*, vol. 3, no. 1, hal. 1, Jun 2018.
 [6] C. Crawford, *Chris Crawford on game design*. New Riders, 2003.
 [7] N. Purwandari, “APLIKASI PERMAINAN LABIRIN 3D MENGENAL OBJEK WISATA DI INDONESIA MENGGUNAKAN MOBILE.”
 [8] G. Agung, *Kupas Tuntas Flash CS5*. Jakarta: PT. Elex Media Komputindo, 2009.
 [9] M. Jamilah dan W. Nurmansyah, “Rancangan Dan Implementasi Aplikasi Alat Bantu Pembelajaran Visualisasi Finite State Automata (Versi Pembuktian Graph ke Tupel),” Agu 2018.
 [10] D. S. Hormansyah, A. R. T. H. Ririd, dan D. T. Pribadi, “IMPLEMENTASI FSM (FINITE STATE MACHINE) PADA GAME PERJUANGAN PANGERAN DIPONEGORO,” *J. Inform. Polinema*, vol. 4, no. 4, hal. 290, Agu 2018.