

APLIKASI GAME EDUKASI SLIDING PUZZLE MENGUNAKAN METODE BREADTH FIRST SEARCH

FURQAN YANI
MARLA SHEILAMITA SHALIN PIETER*)
MOH. R. IRJII MATDOAN*)

*)Staf Pengajar pada Program Studi Teknik Informatika-S1
Fakultas Ilmu Komputer dan Manajemen (FIKOM)
Universitas Sains dan Teknologi Jayapura (USTJ)

ABSTRAK - Kecerdasan Buatan (Artificial Intelligence - AI) dalam sebuah game adalah salah satu bidang penelitian yang sangat menarik hingga saat ini. Kecerdasan Buatan sering digunakan untuk mencari penyelesaian atau solusi, yang meliputi masalah, ruang dan keadaan. Game Sliding Puzzle adalah salah satu bentuk game yang mengasah otak untuk menyelesaikan game tersebut dan dapat memasukkan kecerdasan buatan dalam mencari solusi. Penelitian ini bertujuan untuk merancang dan membangun sebuah perangkat lunak berupa Game Edukasi Sliding Puzzle dengan menggunakan metode Breadth First Search untuk menemukan solusi dengan menggunakan bahasa pemrograman Microsoft Visual FoxPro 9. Penelitian ini menghasilkan Game Edukasi Sliding Puzzle dimana terdapat rekomendasi langkah algoritma Breadth First Search dalam memberikan solusi langkah yang optimal.

Kata Kunci : Game, Sliding Puzzle, Breadth First Search

1. PENDAHULUAN

1.1. Latar Belakang

Perkembangan dalam bidang komputer saat ini telah membuka peluang seluas-luasnya bagi para pakar dan pengambil keputusan untuk menyelesaikan semua permasalahan dengan menggunakan sistem komputer. Berbagai aplikasi komputer kemudian bermunculan, mulai dari aplikasi yang mempermudah perhitungan, yang menyediakan sarana pengelolaan data, maupun aplikasi permainan edukasi sebagai media pembelajaran. Saat ini banyak cara yang digunakan dalam proses pembelajaran, salah satunya adalah dengan cara bermain sambil belajar.

Saat ini game sangat berkembang dengan pesat, mulai dari *game portable*, *stand alone* bahkan sampai pada game berbasis jaringan (*game online*). *Game Sliding Puzzle* merupakan salah satu *game* yang dapat mengasah otak dalam berpikir. *Game Sliding Puzzle* yang dimainkan sepertinya biasa saja, tetapi dengan adanya Kecerdasan Buatan permainan menjadi beda karena terdapat materi Masalah, Ruang Keadaan dengan menerapkan beberapa algoritma. Algoritma *Breadth First Search* merupakan algoritma yang akan diterapkan pada *Game Sliding Puzzle* untuk dapat diselesaikan secara cepat dan efisien dalam mencari solusi.

Beberapa pustaka tentang topik Kecerdasan Buatan dan *Game* sudah banyak dilakukan, seperti Mariana (2013) dalam skripsi dengan judul Aplikasi Penyelesaian *Puzzle* Angka Dengan Menggunakan Metode *BI-Directional Search* (BDS) Dan Metode *BI-Directional A** (BDA*).

Menghasilkan suatu perangkat lunak yang mampu untuk menyelesaikan permasalahan dalam permainan *Puzzle* Angka dengan menggunakan algoritma pencarian BDS dan BDA*, dimana dilakukan pencarian solusi 2 arah sekaligus, yaitu pencarian maju (dari *start* ke *goal*), dan pencarian mundur (dari *goal* ke *start*). Resti Dwi Setyowati (2012) dalam skripsi berjudul Analisis Perbandingan Metode *Breadth First Search (BFS)* dan *Depth Limited Search (DLS)* Untuk Mencari Jawaban Permainan Sudoku. Menghasilkan perangkat lunak sudoku untuk mencari penyelesaian Sudoku dan membuat analisa perbandingan dari hasil pencarian dengan menggunakan metode BFS dan DLS untuk pembuktian teori. Anshar Dadiady (2011) dalam Skripsi dengan judul Analisa Perbandingan Metode *Breadth-First-Search (BFS)* dan Metode A* Dalam Penyelesaian Permainan *Puzzle* Geser. Menghasilkan suatu perangkat lunak yang mampu menyelesaikan permainan *Puzzle* Geser dengan menggunakan metode *Breadth-First-Search (BFS)* dan metode A* serta membandingkan kecepatan dan efisiensi dari metode yang digunakan.

Berdasarkan hal tersebut maka akan dibuat sebuah aplikasi Game Edukasi *Sliding Puzzle* menggunakan metode *Breadth First Search*. *Game* ini nantinya dapat digunakan sebagai media pembelajaran memperkuat kemampuan menganalisa, sehingga diharapkan nantinya *game* ini dapat lebih merangsang keinginan untuk belajar.

1.2. Rumusan masalah

Berdasarkan latar belakang di atas maka permasalahan yang didapat adalah “Bagaimana menerapkan metode *Breadth First Search* pada *Game Sliding Puzzle*?”

1.3. Tujuan

Tujuan dari penelitian ini adalah merancang bangun *Game Edukasi Sliding Puzzle* dengan menggunakan metode *Breadth First Search* sebagai media pembelajaran mata kuliah Kecerdasan Buatan pada materi Masalah, Ruang Keadaan dan Aturan.

2. TINJAUAN PUSTAKA

2.1. Kecerdasan Buatan (Artificial Intelligence/AI)

a. Defenisi

Kecerdasan Buatan berasal dari bahasa inggris “*Artificial Intelligence*” atau disingkat AI. *Intelligence* adalah kata sifat yang berarti cerdas, sedangkan *Artificial* artinya buatan. Kecerdasan Buatan yang dimaksud di sini merujuk kepada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia (*Sutojo, dkk ; 2011*).

b. Masalah, Ruang Keadaan dan Aturan

Sesuai dengan tujuannya, kecerdasan buatan sengaja diciptakan untuk memecahkan berbagai masalah yang dihadapi manusia. Secara umum definisi masalah yang paling mudah dipahami adalah “kesenjangan antara yang diharapkan dengan kenyataan yang ada”. Masalah dalam kecerdasan buatan adalah masalah-masalah yang dapat dikonversi ke dalam ruang keadaan (ada yang menyebut sebagai ruang masalah), mempunyai keadaan awal (*initial state*), dan keadaan tujuan (*goal state*), serta dapat dibuat aturan-aturan untuk mengubah suatu keadaan (*state*) ke keadaan (*state*) lainnya. Secara umum, untuk mendeskripsikan masalah dengan baik, beberapa hal harus dilakukan, seperti :

1) Mendefinisikan suatu ruang keadaan

Ruang keadaan (*state space*), yaitu suatu ruang yang berisi semua keadaan yang mungkin

2) Menetapkan satu atau lebih keadaan awal

Keadaan awal (*initial state*), adalah keadaan dimulainya sebuah pencarian

3) Menetapkan satu atau lebih tujuan (*goal*)

Keadaan akhir/tujuan (*goal*), adalah keadaan diakhirinya sebuah pencarian

4) Menetapkan kumpulan aturan

Kumpulan aturan, adalah aturan yang dapat digunakan untuk mengubah suatu keadaan (*state*) ke keadaan (*state*) lainnya.

(Sutojo,dkk ; 2011)

2.2. Game Playing

Program pertama yang dibuat AI adalah *Game Playing*. Sejarah teori *game* dimulai dari 1950, ketika computer mulai dapat diprogram. *Game* pertama yang menggunakan AI adalah catur. Inisiator teori *game* dalam AI adalah Konard Zuse (penemu pertama komputer yang dapat diprogram dan bahasa pemograman pertama), Claude Shannon (penemu teori informasi), Norbert Wiener (pencipta teori kontrol modern), dan Alan Turing. Sejak saat itu, mulai ada kemajuan dalam standar bermain *game*, sampai-sampai komputer dapat mengalahkan manusia dalam permainan catur, dan dalam banyak game kompetitif lainnya.

Berdasarkan type-nya, *game* dibagi menjadi dua, yaitu *game* dengan informasi lengkap dan *game* dengan informasi tidak lengkap.

a. Game dengan informasi lengkap (*Perfect Information Game*)

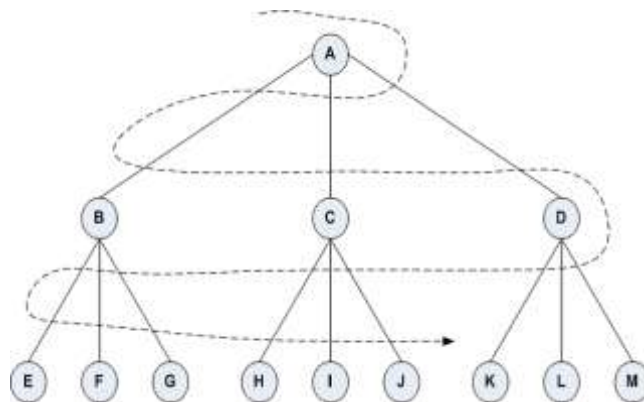
Game dengan informasi lengkap adalah suatu *game* dimana pemain mengetahui semua langkah yang mungkin terjadi dari dirinya sendiri dan dari lawan dan hasil akhir dari permainan. Contoh *game* yang termasuk dalam type ini adalah catur dan *tic tac toe*.

b. Game dengan informasi tak lengkap (*Imperfect Information Game*)

Game ini adalah *game* dimana pemain tidak tahu semua kemungkinan langkah lawan.
(Sutojo, dkk ; 2011)

2.3. Algoritma Breadth First Search (BFS)

Algoritma *Breadth First Search* (BFS) adalah pencarian yang dilakukan pada semua simpul dalam setiap level secara berurutan dari kiri ke kanan. Jika pada satu level belum ditemukan solusi, maka pencarian dilanjutkan pada level berikutnya. Demikian seterusnya sampai ditemukan solusi. Dengan cara seperti ini, BFS menjamin ditemukannya solusi, (jika solusinya memang ada) dan solusi yang ditemukan pasti yang lebih baik. Dengan kata lain, BFS adalah *complete* dan *optimal*. Tetapi, BFS harus menyimpan semua simpul yang pernah dibangkitkan. Hal ini harus dilakukan agar BFS dapat melakukan dengan penelusuran simpul-simpul sampai di level bawah. Jika *b* adalah faktor percabangan (jumlah simpul anak yang dimiliki oleh suatu simpul) dan *d* adalah kedalaman solusi, maka jumlah simpul yang harus disimpan adalah sebanyak $O(b^d)$. (Suyanto;2014)



Gambar 1. Diagram Pohon Dari BFS.

3. HASIL DAN PEMBAHASAN

3.1. Contoh Kasus Dengan Sliding Puzzle 3x3

Pada *Puzzle 3x3* diberikan kondisi awal/start state dengan nilai 3,2,4,1,5,8,6,7,0. Perpindahan/pergeseran angka 0 hanya bisa dilakukan sekali geser dan tidak dapat kembali ke posisi semula sebelum angka 0 bergeser, dikarenakan pencarian akan semakin lama dan berulang-ulang. Tujuan / Goal state yang diinginkan adalah 1,2,3,4,5,6,7,8,0. Terdapat 4 operator yang digunakan untuk ubin kosong dari satu keadaan ke keadaan yang baru, yaitu :

- a. Ubin Kosong digeser ke kiri
- b. Ubin kosong digeser ke kanan
- c. Ubin kosong digeser ke atas
- d. Ubin kosong digeser ke bawah

a. Identifikasi permasalahan dan keadaan

Dari analisa contoh kasus yang berupa angka, maka angka-angka dipetakan ke dalam ubin *Puzzle*, dimana masing-masing angka diberi nama ubin satu, ubin dua, ubin tiga dan seterusnya. Untuk angka 0 diberi nama ubin kosong

b. Keadaan Awal / Start State Dan Tujuan / Goal State

1) Keadaan awal/Start state

3,2,4,1,5,8,6,7,0

	a	b	c
1	3	2	4
2	1	5	8
3	6	7	

2) Tujuan

1,2,3,4,5,6,7,8,0

	a	b	c
1	1	2	3
2	5	5	6
3	7	8	

c. Langkah-langkah Penyelesaian

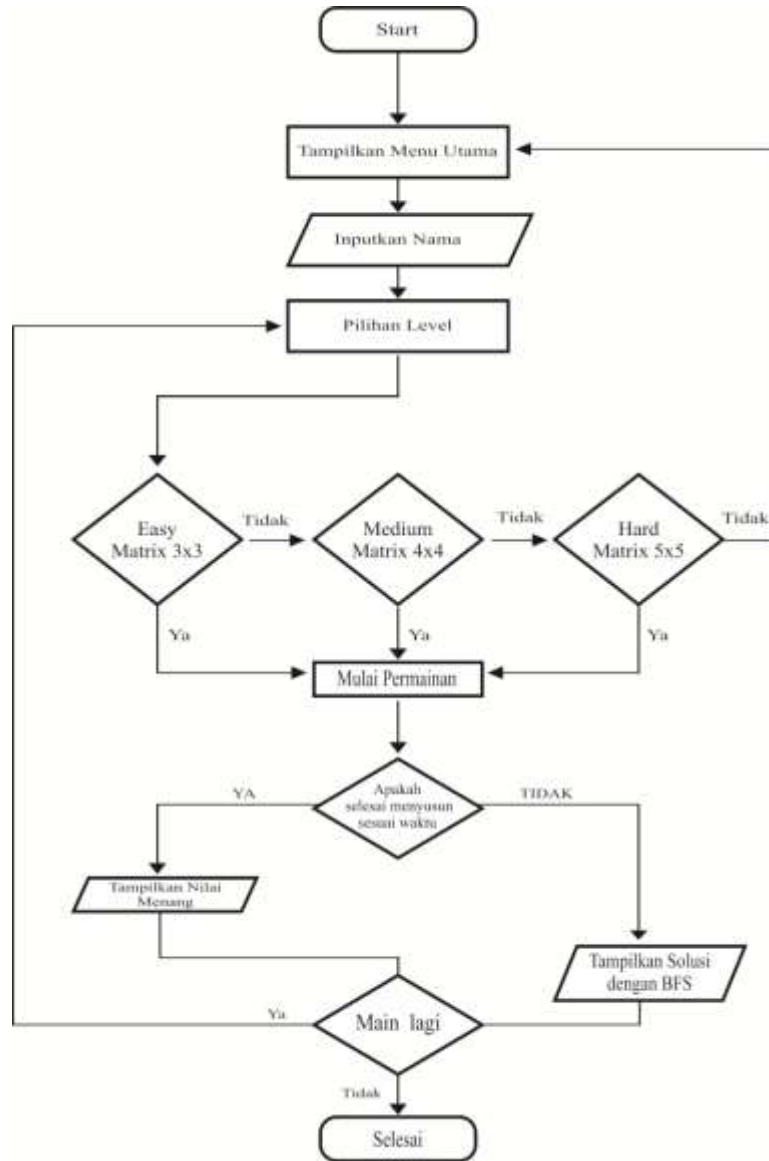
Dari analisa di atas, dapat digambarkan aturan-aturan dalam menyelesaikan permasalahan pada *Puzzle* tersebut, yaitu dengan aturan-aturan sebagai berikut:

- 1) Dicari angka 1
- 2) Setelah mendapatkan angka 1, geser ubin ke atas / ke bawah / ke kiri / ke kanan untuk meletakkan angka 1 di matriks (A,1)
- 3) Cari angka 2
- 4) Lakukan langkah ke 2 dan seterusnya untuk mendapatkan angka yang berurutan

3.2. Desain Sistem Usulan

1) Flowchart

Di bawah ini adalah alur program aplikasi Game Edukasi *Sliding Puzzle* Menggunakan Metode *Breadth First Search*.

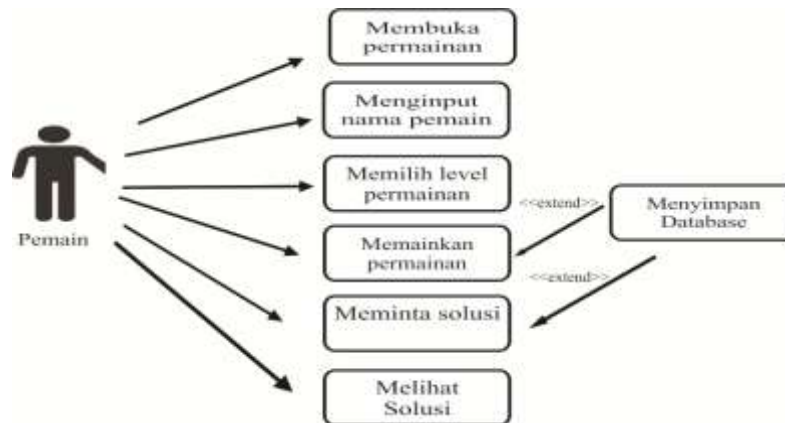


Gambar 2. Flowchart Game Edukasi *Sliding Puzzle* Menggunakan Metode *Breadth First Search*.

2) Unified Modeling Language (UML)

1) Use Case Diagram

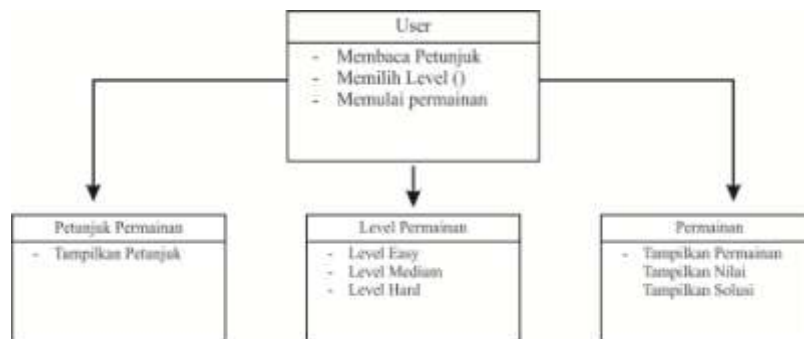
Pada *Game Sliding Puzzle* ini yang menjadi actor adalah user yang berperan sebagai pengguna dari aplikasi ini. Jika dalam permainan ini user mengalami kesulitan atau kebuntuan, maka user dapat melakukan permintaan bantuan, sehingga aplikasi akan menampilkan bantuan berupa solusi permainan.



Gambar 3. Use Case Diagram

2) Class Diagram

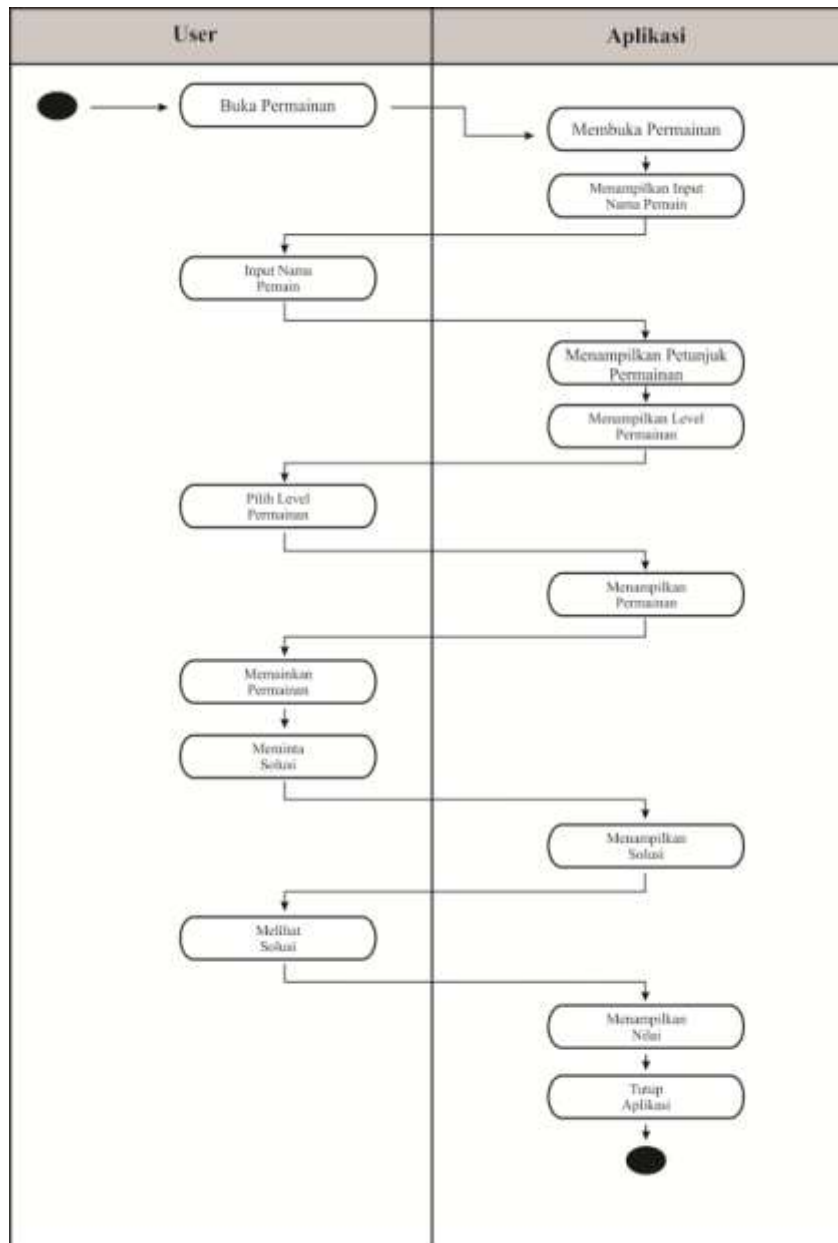
Class Diagram menggambarkan keadaan suatu sistem dari segi pendefinisian kelas – kelas yang akan dibuat. *Class* terdiri dari nama, atribut dan metoda.



Gambar 4. Class Diagram

3) Activity Diagram

Activity Diagram menggambarkan proses bisnis dan urutan aktivitas dalam sebuah proses. Memiliki struktur diagram yang mirip flowchart atau Data Flow Diagram pada perancangan terstruktur.

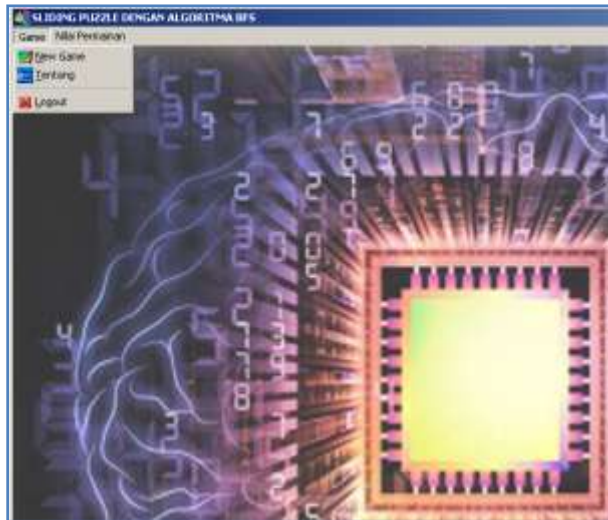


Gambar 5. Activity Diagram

3.3. Implementasi Aplikasi

a. Tampilan Form Menu Utama

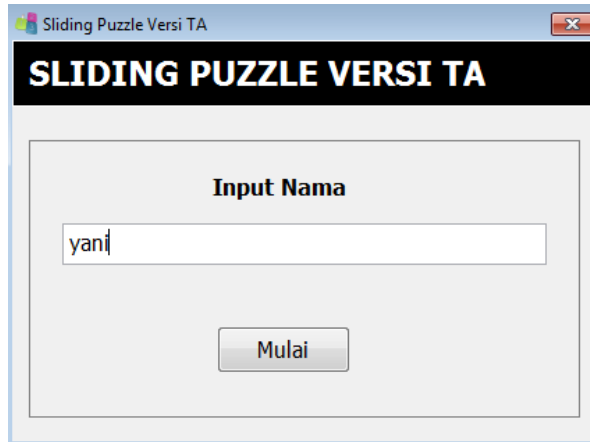
Pada Gambar 6 ditunjukkan tampilan untuk form utama. Dapat dilihat bahwa form utama memiliki dua menu utama, yaitu menu *game* dan menu nilai permainan. Menu *game* digunakan untuk memulai permainan baru dan untuk menampilkan informasi tentang aplikasi. Sedangkan menu nilai permainan digunakan untuk menampilkan nilai tertinggi yang didapatkan pemain.



Gambar 6. Tampilan Form Utama

b. Form Input Nama Pemain

Untuk memulai permainan baru, seorang pemain harus memilih menu *game – new game*. Langkah awal dari sebuah permainan adalah mengisi nama pemain. Tampilan form untuk input nama pemain dapat dilihat pada Gambar 7.

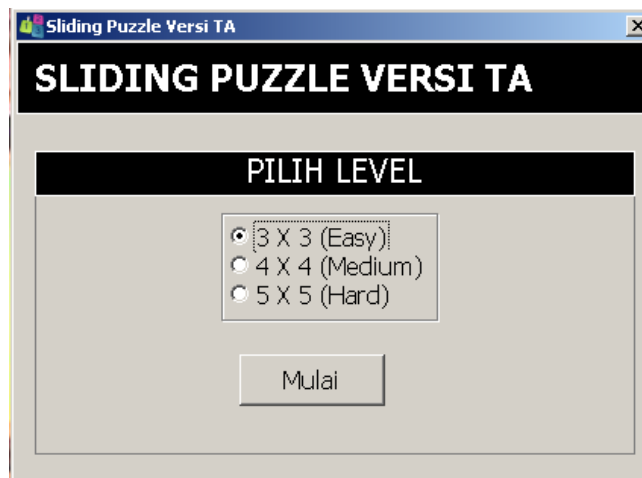


Gambar 7. Form Input Nama Pemain

Pada form input nama pemain, setelah nama pemain diisikan kemudian klik tombol mulai, maka akan masuk ke form pemilihan level.

c. Form Pemilihan Level

Setelah nama pemain diinputkan, langkah selanjutnya adalah memilih level permainan. Dalam aplikasi ini terdapat tiga level, yaitu level *easy* dengan ukuran 3x3, level *medium* dengan ukuran 4x4 dan level *hard* dengan ukuran 5x5. Tampilan form pemilihan level dapat dilihat pada Gambar 8.



Gambar 8. Form Pemilihan Level

d. Form Game

Form *Game* dijalankan setelah nama pemain diinputkan dan pemain memilih salah satu level yang disediakan. Aplikasi *game* dalam penelitian ini dibuat dinamis dan fleksibel dari sisi tampilan, mengingat terdapat beberapa level permainan. Salah satu contoh level *easy* dapat dilihat pada Gambar 9, dapat dilihat bahwa tata letak dan ukuran komponen dapat menyesuaikan dengan level yang dipilih.



Gambar 9. Tampilan Game Level Easy

Untuk memulai permainan, pemain harus mengacak puzzle dengan klik tombol *acak*. Setelah mendapatkan hasil pengacakan yang diharapkan, pemain harus klik tombol *start* untuk memulai permainan. Ketika permainan dimulai, maka akan muncul *timer* yang akan menghitung waktu yang digunakan pemain dalam menyelesaikan permainan. Pada gambar 4.13 ditunjukkan pemain dengan nama 'Yani' memulai permainan dengan level *easy* dan puzzle yang sudah diacak.



Gambar 10. Memulai Game

Sewaktu-waktu pemain dapat menghentikan permainan dengan klik tombol *stop*. Aplikasi juga dilengkapi dengan fitur *solusi*, yaitu ketika pemain tidak dapat menyelesaikan permainan

dan ingin melihat langkah untuk menyelesaikan permainan. Pada Gambar 11 ditampilkan contoh solusi untuk menyelesaikan permainan.



Gambar 11. Contoh solusi

4. PENUTUP

Kesimpulan yang dapat diambil dari Game *Sliding Puzzle* yang telah dibuat adalah :

1. Metode Breadth First Search dapat digunakan untuk mencari solusi pada Game Sliding Puzzle.
2. *Game Sliding Puzzle* ini dapat digunakan sebagai media pembelajaran yang dapat memperkuat kemampuan menganalisa pada materi Keadaan, Ruang Keadaan dan Tujuan pada Kecerdasan Buatan.
3. Metode *Breadth First Search* membutuhkan memori yang sangat besar untuk mendapatkan solusi, sehingga disarankan menggunakan komputer atau laptop dengan memori yang besar.

5. DAFTAR PUSTAKA

- Dadiady A, 2011, *Analisa Perbandingan Metode Breadth-First-Search (BFS) dan Metode A* Dalam Penyelesaian Permainan Puzzle Geser*, Program Studi Teknik Informatika-S1 FIKOM USTJ, Jayapura
- Fowler M, *UML Distilled Edisi 3*, PT ANDI, Yogyakarta.
- Kusumadewi S, 2003, *Artificial Intelligence (Teknik dan Aplikasinya)*, Graha Ilmu.
- Mariana, 2013, *Aplikasi Penyelesaian Puzzle Angka Dengan Menggunakan Metode BI-Directional Search (BDS) Dan Metode BI-Directional A* (BDA*)*, Program Studi Teknik Informatika-S1 FIKOM USTJ, Jayapura
- Setyowati R. D, 2012, *Analisis Perbandingan Metode Breadth First Search dan Depth Limited Search Untuk Mencari Jawaban Permainan Sudoku*, Program Studi Teknik Informatika-S1 FIKOM USTJ, Jayapura
- Sugiarti Y, 2013, *Analisis & Perancangan UML (Unified Modeling Language) Generated VB.6*, Graha Ilmu.
- Sutojo T, Mulyanto E, Suhartono V, 2011, *Kecerdasan Buatan, Edisi 1*, ANDI Yogyakarta.
- Suyanto, 2014, *Artificial Intelligence Searching, Reasoning, Planning dan Learning*, Informatika Bandung.