

RANCANG BANGUN APLIKASI GAME PUZZLE BERBASIS ANDROID MENGGUNAKAN ALGORITMA DECISION TREE

Muhaimin Hasanudin¹, Turki Salim², Arkan Nur Robby³
Dosen STMIK Raharja Tangerang Jurusan Sistem Informasi^{1,2,3}
Jl. Jendral Sudirman No 40 Modernland, Tangerang^{1,2,3}
email : muhaimin@raharja.info¹, turki@raharja.info², arkan@raharja.info³

Abstrak

Perkembangan *game* saat ini sangat pesat dan telah menjadi bagian dari gaya hidup. *Game puzzle* adalah sebuah aplikasi yang dapat digunakan untuk mengasah kemampuan berpikir dan daya ingat dimana pengguna harus mencari jalan keluar dan mengumpulkan item khusus yang telah disediakan disetiap level permainan. Pemain harus mengumpulkan semua *white orb* yang ada disetiap level agar mendapatkan kunci untuk berpindah ke level selanjutnya namun pemain akan dihadapkan oleh kendala bagaimana cara menemukan kunci dan mengambilnya. Permainan ini dirancang menggunakan algoritma decision tree dan permainan dirancang menggunakan *Game Maker Studio* yang diaplikasikan ke perangkat Android.

Kata kunci : *Puzzle game, white orb, decision tree*

Abstract

The development of games today is very rapid and has become part of the lifestyle. A puzzle game is an application that can be used to hone thinking and memory skills where users have to find a way out and collect special items that have been provided at every level of the game . The player must collect all the white orb in each level to get the key to move to the next level but the player will be confronted by the constraints on how to find the key and retrieve it . The game is designed using decision tree and game algorithms designed using Game Maker Studio that is applied to Android devices .

Key Words : Puzzle game, white orb, decision tree

1. PENDAHULUAN

1.1 Latar Belakang

Sebagian besar game puzzle memiliki pola, yaitu Teka-teki tersebut hanya memiliki 1 solusi, sehingga apabila pemain tidak dapat menyelesaikannya maka pemain tidak dapat melanjutkan permainan. Permainan yang memiliki tingkat kesulitan yang terlalu tinggi, dapat menyebabkan frustrasi yang membuat pemain tidak ingin bermain lagi, karena pemain tidak menemukan jawaban atau tidak mampu mengendalikan permainan (University of Rochester, 2014).

Android merupakan sebuah sistem operasi pada ponsel berbasis Linux yang mencakup sistem operasi dan middleware. Fasilitas opensource atau sistem operasi yang dapat dikembangkan dengan bebas bagi penggunaanya membuat banyak orang untuk mengembangkannya dengan inovasi – inovasi yang semakin berkembang terhadap sistem operasinya maupun pada pembangunan aplikasi mobile nya tersebut. Maka tak heran saat ini banyak pengembang yang membangun aplikasi mobile pada platform Android.

Elemen kunci dari sebuah permainan yaitu tujuan, peraturan, tantangan dan interaksi yang pada umumnya menstimulasi fisik dan kejiwaan para pemainnya. Beberapa jenis permainan dianggap bisa meningkatkan keterampilan sensorik dan motorik, serta memiliki fungsi sebagai

sarana latihan fisik dan mental. Game Puzzle merupakan bentuk permainan yang menantang daya kreativitas dan ingatan anak lebih mendalam dikarenakan munculnya motivasi untuk senantiasa mencoba memecahkan masalah, namun tetap menyenangkan sebab bisa di ulang-ulang. Tantangan dalam permainan ini akan selalu memberikan efek ketagihan untuk selalu mencoba, mencoba dan terus mencoba hingga berhasil. Dalam pembuatan aplikasi tersebut menggunakan Teknik Software Android Development Tools (ADT) Build v.22 includes Eclipse Platform, Java Development Kit (JDK), Android Software Development Kit (Android SDK), adobe photoshop dan *Game Maker Studio 1.2*. Hal ini menjadi latar belakang untuk melakukan penelitian yang berjudul “Rancang Bangun Aplikasi Game Puzzle menggunakan algoritma decision tree”.

1.2 Rumusan Masalah

Berdasarkan uraian dari latar belakang masalah sebelumnya, maka dapat dirumuskan beberapa permasalahan sebagai berikut:

1. Apakah klasifikasi tingkat game puzzle yang terbentuk dengan *Decision Tree* memiliki tingkat akurasi yang tinggi?
2. Bagaimana merancang bangun game puzzle berbasis Android?

1.3 Batasan Masalah

1. *Game* ini berupa Puzzle game yang dibangun dengan media Teknik Software Android Development Tools (ADT) Build v.22 includes Eclipse Platform, Java Development Kit (JDK), Android Software Development Kit (Android SDK), adobe photoshop dan *Game Maker Studio 1.2*
2. Proses perancangan dan mekanisme pembuatan sebuah *puzzle game* untuk *platform android*
3. *Game* yang dibangun diImplementasikan pada *smartphone Android*.

1.4 Manfaat Penelitian

Menjadikan permainan/*Game* untuk Meningkatkan kemampuan dalam hal konsentrasi dan mengurangi stress.

2. LANDASAN TEORI

2.1. Pengertian Rancang Bangun

Menurut Hasan dan Alvin (2010:14) Rancang bangun adalah suatu istilah umum untuk membuat atau mendesain suatu objek dari awal pembuatan sampai akhir pembuatan. Rancang bangun berawal dari kata desain yang artinya perancangan, rancang, desain, bangun. Sedangkan merancang artinya mengatur, mengerjakan atau melakukan sesuatu dan perancangan artinya proses, cara, perbuatan merancang. Dapat disimpulkan arti kata desain adalah proses, cara, perbuatan dengan mengatur segala sesuatu sebelum bertindak atau merancang

2.2 Decision Tree

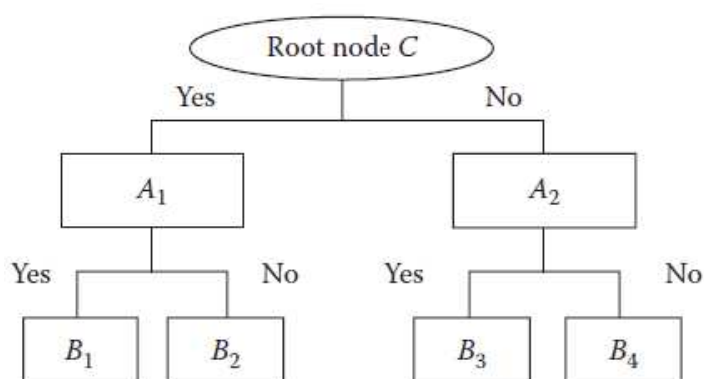
Algoritma *decision tree* didasarkan pada pendekatan *divide-and-conquer* untuk klasifikasi suatu masalah. Algoritma tersebut bekerja dari atas ke bawah, mencari pada setiap tahap atribut untuk membaginya ke dalam bagian terbaik *class* tersebut, dan memproses secara rekursif submasalah yang dihasilkan dari pembagian tersebut. Strategi ini menghasilkan sebuah *decision tree* yang dapat diubah menjadi satu set *classification rules* (Witten et all, 2011). *Decision tree* merupakan salah satu metode klasifikasi yang menggunakan representasi struktur pohon (*tree*) di mana setiap *node* merepresentasikan atribut, cabangnya merepresentasikan nilai dari atribut, dan daun merepresentasikan kelas. *Node* yang paling atas dari *decision tree* disebut sebagai *root* (Gorunescu, 2011).

Pada *decision tree* terdapat 3 jenis *node*, yaitu:

- a. *Root Node*, merupakan *node* paling atas, pada *node* ini tidak ada *input* dan bisa tidak mempunyai *output* atau mempunyai *output* lebih dari satu.

- b. *Internal Node* , merupakan *node* percabangan, pada *node* ini hanya terdapat satu *input* dan mempunyai *output* minimal dua.
- c. *Leaf node* atau *terminal node* , merupakan *node* akhir, pada *node* ini hanya terdapat satu *input* dan tidak mempunyai *output*.

Seperti ditunjukkan dalam Gambar 1, *decision tree* tergantung pada aturan *if-then*, tetapi tidak membutuhkan parameter dan metrik. Struktur sederhana dan dapat ditafsirkan memungkinkan *decision tree* untuk memecahkan masalah atribut *multi-type*. *Decision tree* juga dapat mengelola nilai-nilai yang hilang atau data *noise* (Dua & Xian, 2011). Banyak algoritma yang dapat dipakai dalam pembentukan *Decision Tree*, antara lain ID3, CART, dan C4.5 (Larose, 2005). Algoritma C4.5 dan pohon keputusan merupakan dua model yang tak terpisahkan, karena untuk membangun sebuah pohon keputusan, dibutuhkan algoritma C4.5 (Han & Kamber, 2006).



Gambar 1. Struktur *Decision Tree*

Sumber: Dua & Xian, 2011

2.3 Aplikasi *Game maker Studio*

Game Maker Studio adalah sebuah *game engine* yang bisa membantu membuat sebuah *game* komputer tanpa menggunakan codingan yang banyak dengan bahasa programan. Dengan *game maker Studio* pengguna dapat membuat *game* 2D, 3D atau *multiplayer*, dan mempunyai hak penuh terhadap *game* yang telah dibuat. *Game Maker Studio* (sering disingkat GMS) awalnya dikembangkan oleh Mark Overmars dalam bahasa pemrograman *Delphi*. Saat ini dikembangkan dan diterbitkan oleh Yoyo Games, sebuah perusahaan perangkat lunak.

Bahasa dalam *Game Maker* adalah GML (*Gamemaker Lanluange*) merupakan bahasa *scripting* utama yang ditafsirkan mirip dengan Java kompilasi *Just-In-Time* yang digunakan dalam *Game Maker*. Hal ini memungkinkan pengguna untuk lebih meningkatkan dan mengendalikan desain permainan mereka melalui pemrograman konvensional.

Dalam *Game Maker*, serangkaian tindakan *drag-and-drop* disebut *Libraries*, *libraries* ini ditampilkan sebagai tab yang berisi ikon yang disebut *Actions*. Setiap *Action* adalah *script GML* atau fungsi yang pengguna dapat digunakan dalam permainan mereka. *Game Maker* dilengkapi dengan set standar perpustakaan yang berisi tindakan yang umum digunakan oleh kebanyakan *game*, melainkan juga memungkinkan untuk membuat *libraries* menggunakan *Library Builder* disediakan secara terpisah dari *Game Maker*

2.4 *Android*

Android adalah sebuah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi. *Android* menyediakan *platform* yang terbuka bagi para pengembang untuk menciptakan aplikasi mereka. *Android* merupakan generasi baru *platform mobile*, *platform* yang memberikan pengembang untuk melakukan

pengembangan sesuai dengan yang diharapkannya. Sistem operasi yang mendasari Android dilisensikan dibawah GNU, *General Public Lisensi* Versi 2 (GPLv2), yang sering dikenal dengan istilah “*copyleft*” lisensi di mana setiap perbaikan pihak ketiga harus terus jatuh di bawah *terms*. Android didistribusikan di bawah *Lisensi Apache Software* (ASL/Apache2), yang memungkinkan untuk distribusi kedua dan seterusnya. Komersialisasi pengembang (produsen *handset* khususnya) dapat memilih untuk meningkatkan *platform* tanpa harus memberikan perbaikan mereka ke masyarakat *open source*. Sebaliknya, pengembang dapat keuntungan dari perangkat tambahan seperti perbaikan dan mendistribusikan ulang pekerjaan mereka di bawah lisensi apapun yang mereka inginkan. Pengembang aplikasi Android diperbolehkan untuk mendistribusikan aplikasi mereka di bawah skema lisensi apapun yang mereka inginkan.

2.5 Video Game

Video game adalah permainan yang menggunakan interaksi dengan antarmuka pengguna melalui gambar yang dihasilkan oleh piranti, video Game juga merupakan suatu sarana untuk menghibur diri. Bermain game sangat efektif untuk menghilangkan stres dan mengasah kemampuan otak. Game sendiri berasal dari kata bahasa inggris yang memiliki arti dasar Permainan. Permainan dalam hal ini merujuk pada pengertian kelincahan intelektual (*intellectual playability*). Game juga bisa diartikan sebagai arena keputusan dari aksi pemainnya karena ada target-target yang ingin dicapai, kelincahan intelektual pada tingkat tertentu juga merupakan ukuran sejauh mana game itu menarik untuk dimainkan secara maksimal. Pada era globalisasi saat ini, para pengelola industri game berlomba-lomba untuk menciptakan game yang lebih nyata dan menarik untuk para pemainnya. Hal inilah yang membuat perkembangan games sangat cepat. Sehingga games bukan hanya sekedar permainan untuk mengisi waktu luang atau sekedar hobi. Melainkan sebuah cara untuk meningkatkan kreatifitas dan tingkat intelektual para penggunanya bahkan sebagai sebuah lapangan pekerjaan. bermain game adalah suatu proses fine tuning (atau penyamaan frekuensi) dari logika berpikir para pemainnya dengan logika berpikir aplikasi komputer yang canggih tadi. Pada saat bersamaan, game juga secara nyata mempertajam daya analisis para penggunanya untuk mengolah informasi dan mengambil keputusan cepat.

3 HASIL DAN PEMBAHASAN

3.2 Analisis Kebutuhan Sistem

Pada pembuatan *game* ini akan menghasilkan aplikasi *game mobile* yang dapat berjalan pada *platform Android*. Sehingga perlu memahami kebutuhan dari sebuah sistem.

3.2.1 Analisis Kebutuhan Fungsional

Pada umumnya kebutuhan fungsional pengembangan *game* membahas fitur-fitur yang akan diberikan kepada pemain dan diimplementasikan ke dalam *game*. Kebutuhan fungsional membahas tindakan yang diberikan oleh sistem ketika menerima input tertentu dari pengguna. Berikut adalah analisis kebutuhan fungsional dari game tersebut.

1. *Game* ini memiliki 5 level.
2. Pada setiap level memiliki tingkat kesulitan yang berbeda.
3. *Game* ini dapat menyimpan progress permainan disetiap level. Sehingga user bisa melanjutkan permainan dengan memilih tombol *resume*.

3.2.2 Analisis Kebutuhan Non-Fungsional

1. Kebutuhan Perangkat Keras (Hardware)

Untuk kebutuhan *hardware*, penulis menggunakan satu unit PC (*Personal Computer*) dengan spesifikasi, Intel Core i5 2410M 2.3Ghz, Memori DDR3 4GB, HD 100 GB, VGA nvidia GT 540M 1GB, Monitor 15 Inch.

2. Kebutuhan Perangkat Lunak (Software)

Perangkat lunak (*software*) adalah program yang digunakan untuk menjalankan perangkat keras. Tanpa adanya perangkat lunak ini komponen perangkat keras tidak dapat berfungsi. Perangkat lunak yang digunakan dalam pembangunan game ini adalah :

1. *Operating System : Windows xp*
2. *Game Editor : Game Maker Studio 1.2*
3. *ADT Bundle windows x86*
4. *Android SDK*
5. *Java Development Kit*

3.3 Perancangan Game

3.3.1 Gambaran Umum

Prototype ini merupakan permainan yang mendeskripsikan tentang sebuah objek sederhana yang bertujuan agar objek yang kita gerakan menemukan jalan keluar serta mencari dan mengumpulkan *item* khusus yang telah disediakan disetiap level permainan. Adapun sebagian objek yang harus dihindari interaksinya secara langsung dikarenakan akan membuat player terkena *damaged* dan permainan berakhir. Berikut rincian *game* yang akan dibuat:

- a. *Game* bergenre *Puzzle*
- b. Sistem permainan *single player*
- c. *Game* terdiri lima level permainan
- d. Karakter yang ada dalam *game* ini hanya sebuah bola abu-abu

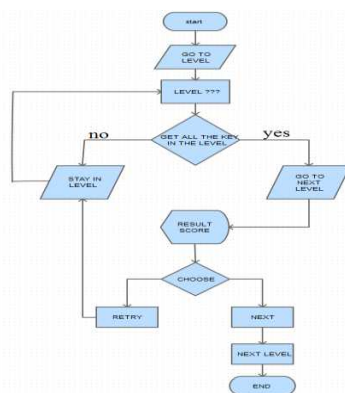
3.3.2 Aturan Permainan

Ada beberapa aturan dasar dari permainan ini diantaranya adalah:

- A. Pemain hanya akan disediakan satu aktor saja
- B. Pemain mengendalikan actor dari pandangan samping/dua dimensi
- C. Tombol panah kanan dan kiri untuk bergerak kearah kanan dan kiri, tombol panah atas untuk membuat karakter melompat dan terbang.
- D. Pemain harus mengumpulkan semua *white orb* yang ada disetiap level agar mendapatkan kunci untuk berpindah ke level selanjutnya namun pemain akan dihadapkan oleh kendala bagaimana cara menemukan kunci dan mengambilnya.

3.3.3 Rancangan Level Game

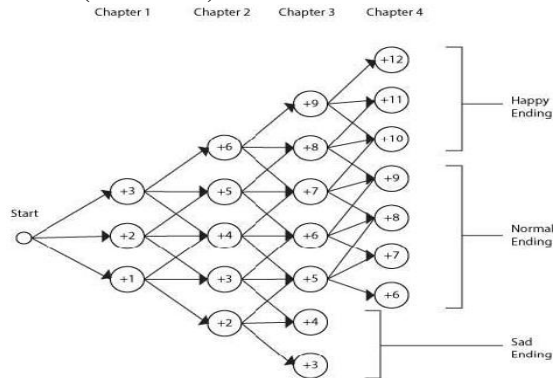
Perancangan level *game* sangat berpengaruh bagi pemain, perancangan level *game* ini berisi mengenai bermacam *segment* yang dipakai serta beberapa tingkat kesulitan masing-masing level yang akan di buat pada perancangan *game* dan juga *gameplay* yang bertahap dari mulai tingkatkan yang mudah untuk dilalui hingga tingkatan yang akan menguras kemampuan berpikir pemain nantinya.



Gambar 2. Flowchart level permainan

3.3.4 Algoritma

Berikut salah satu algoritma yang digunakan dalam permainan, bagaimana penentuan ending menggunakan decision tree (Gambar 3).



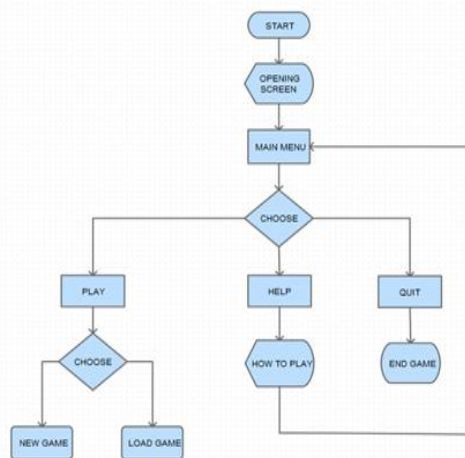
Gambar 3. Decision Tree Ending

Penentuan ending ditentukan oleh bintang yang diperoleh pemain tiap chapternya, jika pemain mendapatkan bintang emas (rank A) maka diberi nilai 3, bintang perak (rank B) diberi nilai 2, sedangkan bintang perunggu (rank C) diberi nilai 1. Jika pemain tidak memiliki nilai yang cukup, saat menyelesaikan chapter 3 pemain akan memperoleh salah 1 ending, yaitu sad ending sehingga pemain tidak dapat meneruskan permainan ke chapter selanjutnya, tapi jika pemain memiliki nilai yang cukup, maka pemain dapat melanjutkan ke chapter selanjutnya dan memperoleh salah 1 ending, yaitu normal ending atau sad ending.

Gambar 3 merupakan bentuk decision tree untuk menunjukkan ending, pada akhir chapter 3 nilai yang diperoleh di chapter sebelumnya dan chapter 3 akan dihitung, jika nilai total kurang dari sama dengan 4 maka sad ending ditampilkan. Sehingga bila kombinnasi nilai pemain dari tiap chapter ialah 3-1-2 (chapter 1 rank A, chapter 2 rank C, chapter 3 rank B) maka pada chapter 4 pemain pasti mendapatkan normal ending, namun bila pemain memiliki kombinasi 3-1-3 maka pemain harus memperoleh score sempurna untuk mendapatkan happy ending, jika tidak maka pemain mendapatkan normal ending.

3.3.5 Menu Permainan

Jika pemain menekan tombol *play* maka pemain akan dihadapkan pada dua pilihan antara memulai permainan baru atau permainan sebelumnya, jika pemain menekan tombol *help* maka pemain akan ditampilkan poin-poin yang harus dimainkan dalam *game* ini dan yang terakhir adalah tombol *exit* yang akan membawa pemain keluar dari permainan.



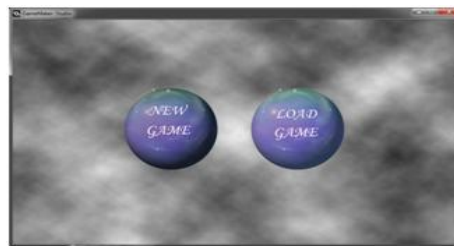
Gambar 4. flowchart menu permainan



Gambar 5. Menu Utama

3.3.6 *Play Room*

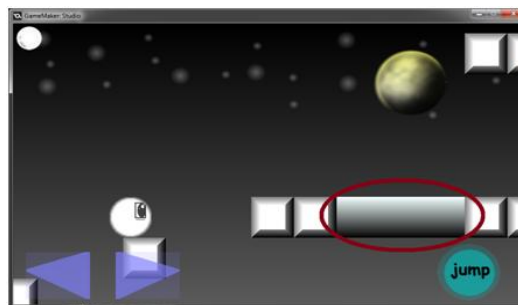
Tampilan *play room* merupakan pilihan untuk pemain ketika ingin memulai permainan dari awal atau terakhir kali bermain. Di *play room* pemain akan diberikan dua pilihan untuk memulai permainan dari awal atau terakhir kali dimainkan.



Gambar 6. *Play Room*

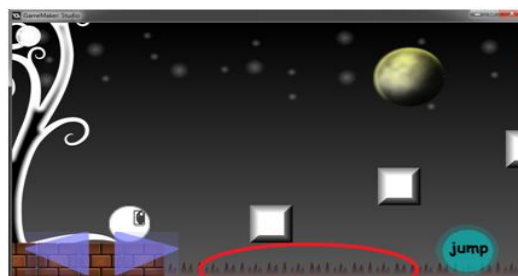
3.3.7 *level permainan*

Berikut merupakan tampilan permainan yang ada di setiap level



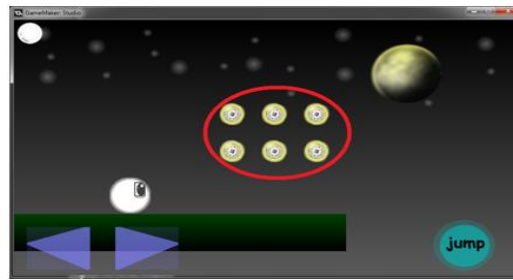
Gambar 7. Tampilan level permainan 1

Pada gambar 7. dengan objek yang dilingkari oleh warna merah, ketika player berada di area tersebut kecepatan dan tinggi lompatan player akan mengalami peningkatan



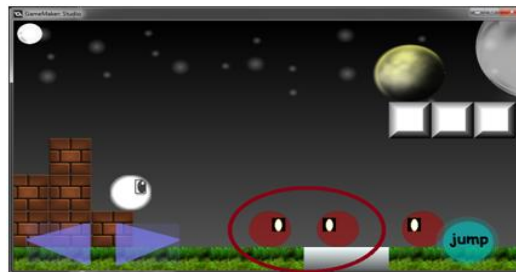
Gambar 8 Tampilan level permainan 2

Untuk gambar 8 daerah yang di lingkari oleh warna merah merupakan zona yang harus dihindari oleh pemain karena jika player terjatuh ke area tersebut, pemain akan dipaksa mengulang kembali ke awal level.



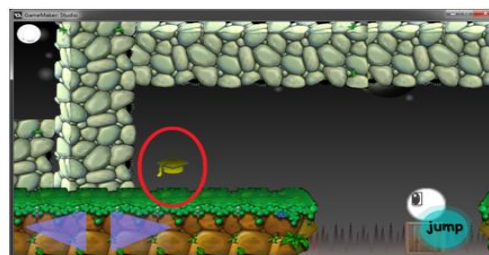
Gambar 9 Tampilan level permainan 3

Objek yang di lingkari warna merah merupakan poin-poin untuk dikumpulkan yang berada di setiap level permainan dan sebagai tolak ukur seberapa besar kemampuan pemain dapat memperolehnya.



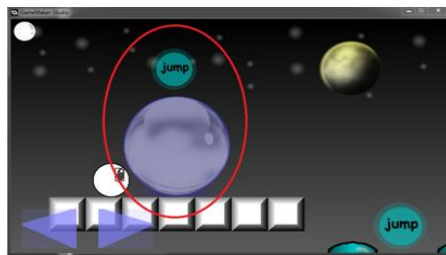
Gambar 10. Tampilan level permainan 4

untuk gambar 10. objek yang di lingkari merupakan artificial intelligence (AI) dalam game dan akan hancur apabila player menginjaknya, tetapi jangan sampai mengenai bagian sisinya karena akan membuat player kembali mengulangi level yang sama



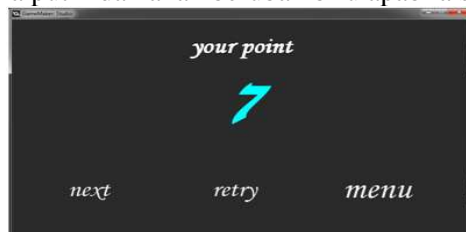
Gambar 11, Tampilan level permainan 5

Objek yang di lingkari merupakan kunci yang harus diambil yang ada di setiap level permainan, jumlahnya tidak ditentukan di setiap level tetapi jika semuanya sudah terambil maka gerbang secara otomatis akan terbuka.



Gambar 11. Tampilan level permainan 6

Pada gambar 11. untuk objek yang dilingkari adalah gerbang menuju level selanjutnya, sebelumnya berwarna putih dan akan berubah biru apabila semua kunci telah diambil.



Gambar 12. Tampilan *score* pemain

Tampilan *score* yang telah didapat pemain dari mengumpulkan objek berwarna merah seperti gambar 9 akan diperlihatkan setelah pemain memasuki gerbang seperti gambar 11

4 KESIMPULAN DAN SARAN

4.2 Kesimpulan

Berdasarkan hasil dari implementasi dan pengujian yang dilakukan peneliti, maka dapat ditarik kesimpulan sebagai berikut :

1. Model Decision Tree dapat digunakan dalam membuat klasifikasi sebagai dasar dalam menentukan tingkat kesulitan dari setiap level yang ada .
2. Game puzzle berhasil dibangun dengan lebih menarik dan interaktif dimana pemain harus mengumpulkan semua *white orb* yang ada disetiap level agar mendapatkan kunci untuk berpindah ke level selanjutnya namun pemain akan dihadapkan oleh kendala bagaimana cara menemukan kunci dan mengambilnya.

4.3 Saran

Dari kesimpulan yang dipaparkan diatas maka didapatlah suatu saran yang mudah – mudahan bermanfaat bagi pengembangan aplikasi video game selanjutnya. Berikut beberapa saran dari penulis pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak *video game* :

1. Pengembang aplikasi ini masih memiliki banyak peluang untuk membuat aplikasi yang akan di gunakan di berbagai macam platform nantinya dan dengan banyaknya teori – teori yang mendukung sehingga dapat membantu memberikan ide yang cemerlang.
2. Perkembangan sebuah video game yang semakin beragam dapat menciptakan ide – ide kreatif di masa yang akan datang, terlebih lagi dengan semakin pesatnya perkembangan industri video game untuk menjadi access point yang menjadi wadah penyaluran kreatifitas dimasa yang akan datang.

DAFTAR PUSTAKA

- [1] Adam M.Christ, *Bridging the Mobile App Gap*. Sigma Noblis Vol 11 Number 1,2011
- [2] Witten, I. H., Frank, E., Hall, M. A. 2011. *Data Mining Practical Machine Learning Tools and Techniques* (3rd ed). USA: Elsevier
- [3] Gorunescu, F. 2011. *Data Mining Concept Model and Techniques*. Berlin: Springer. ISBN 978- 3-642-19720-8
- [4] Rojers, Rick. *Learning Android Game Programming*. United States; Pearson Education, Inc. 2012
- [5] Safaat H, Nasruddin. *Andorid, Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android, Revisi Kedua*. Bandung: Informatika Bandung.2014
- [6] Siregar, Ivan Michael. *Membongkar Source Code Berbagai Aplikasi Android*. Yogyakarta : Gava Media. 2011
- [7] Dua, S. & Xian Du. 2011. *Data Mining and Machine Learning in Cybersecurity*. USA: Taylor & Francis Group. ISBN-13: 978-1- 4398-3943-0
- [8] Taru Nugrho N.W, *Semua Bisa Menjadi Programmer Android*, Andi Offset, Yogyakarta. 2012
- [9] Larose, D. T. 2005. *Discovering Knowledge in Data*. New Jersey: John Willey & Sons, Inc. ISBN 0-471-66657-2.
- [10] Han, J., & Kamber, M. 2006. *Data Mining Concept and Tehniques*. San Fransisco: Morgan Kauffman. ISBN 13: 978-1-55860-901-3
- [11] University of Rochester. (2014). Video gamers' aggression linked to frustation, not violent content. www.rochester.edu/newscenter/frustation-in-mastering-video-games-linked-to-aggression/. diakses tanggal 28 April 2015.