

Pengembangan *Game* Karambol Pada *Interactive Projected Display*

Bima Tri Atmaja¹, Eriq M. Adams Jonemaro², Issa Arwani³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹deepdrumer.19@gmail.com, ²eriq.adams@ub.ac.id, ³issa.arwani@ub.ac.id

Abstrak

Karambol adalah salah satu *game* yang menggunakan meja berbentuk persegi dengan empat lubang di setiap pojok diagonalnya, dengan alat permainannya berupa koin karambol. Permainan ini menggunakan cakram kecil seperti lempengan bola. Penggerakannya adalah dengan jari tangan yang disentilkan pada cakram sehingga mengenai cakram sasaran. Pemainnya harus memasukan koin karambol ke dalam lubang yang terdapat di pojok papan. Pemain dinyatakan menang jika memasukan semua buah koin karambol lebih cepat dari pemain lainnya. *Game* karambol ini akan dikembangkan ke dalam teknologi yang lebih modern. *Game* karambol akan ditampilkan melalui proyektor dengan menggunakan teknologi *Ubi Displays (Interactive Projected)*. *Ubi Displays* merupakan *toolkit prototype* menggunakan proyektor dan Kinect, dimana *game* dapat ditampilkan dan dimainkan pada jenis bidang permukaan padat, seperti pada permukaan tembok. Dari hasil pengujian terhadap siswa Madrasah Tsanawiyah (MTs) Wahid Hasyim 01 Dau menunjukkan bahwa *game* karambol ini dapat diterima dengan baik. Hal ini bisa dilihat bahwa terdapat lebih dari 67% siswa MTs menyatakan *game* ini menarik, karena menurut siswa MTs *game* dikemas dalam bentuk baru menggunakan *interactive projected display* yang mana semua bidang permukaan *game* menjadi *touch screen*, sehingga *game* terlihat nyata (seperti permainan aslinya).

Kata kunci: *game, interactive projected display, karambol, ubi displays*

Abstract

Carrom is a game that uses square-shaped table with four holes in each corner of the diagonal, with the tools of the game in the form of coins carrom. This game uses small discs instead of balls. The driving force is the finger shifted on the disc so hit the target. Players must put all the coins carrom into the hole located at the corner of the board. Otherwise the player wins if one of player put all the pieces carrom coins faster than other players. Carrom games will be developed into a more modern technologies. Carrom games are displayed via a projector using Ubi Displays technology (Interactive Projected). Ubi Displays a prototype toolkit using a projector and Kinect, where the game can be displayed and played on types of surfaces as on wall. From the test results of students in Islamic Junior High School Wahid Hasyim 01 Dau showed that carrom game is well received. This can be seen from more than 67% of students in Islamic Junior High School said this game interesting, because according from the students of Islamic Junior High School packed game in a new form using the interactive projected display which all surfaces into a touch screen game, so the game looks real (like the original game).

Keywords: *game, interactive projected display, carrom, ubi displays*

1. PENDAHULUAN

Permainan *carrom* atau karambol telah dimainkan kira-kira 2000 tahun yang lalu di zaman Makam Tua Abyssinian. Ada juga yang mengatakan *carrom* telah ditemukan dan diciptakan oleh Maharaja India, sementara banyak di India percaya bahwa *carrom* telah diperkenalkan oleh Inggris. Beberapa buku pada

permainan internasional termasuk Birma, Mesir, dan Ethiopia mengatakan *carrom* berasal dari India. *Game* ini menyebar luas ke berbagai negara lainnya pada tahun 1970-an. *Carrom* telah dimainkan oleh orang-orang India bersama keluarga di rumah atau di tanah lapang (Sahid, 2011: 1-2).

Turnamen *carrom* yang pertama kali di adakan di Sri Lanka pada tahun 1935, dan pada

tahun 1958 baik India dan Sri Lanka telah membentuk federasi resmi klub *carrom*, mensponsori turnamen dan pemberian hadiah (Sahid, 2011: 3).

Game karambol ini hampir mirip dengan billiard, hanya saja mejanya berbentuk persegi dengan empat lubang di setiap pojok diagonalnya, dengan alat permainannya berupa koin karambol, tidak memakai stik untuk menggerakkan koin karambol. Permainan ini menggunakan cakram kecil sebagai pengganti bola. Penggeraknya adalah jari tangan yang disentilkan pada cakram sehingga mengenai cakram sasaran. Pemainnya harus memasukan semua koin karambol ke dalam lubang yang terdapat di pojok papan. Pemain dinyatakan menang jika memasukan semua buah koin kecuali koin gacoan lebih cepat dari pemain lainnya (Win, 2011: 1-2).

Seiring dengan berkembangnya teknologi, *game* karambol ini akan dikembangkan ke dalam teknologi yang lebih modern. *Game* akan ditampilkan melalui proyektor dengan menggunakan teknologi Ubi Displays (*Interactive Projected*). Ubi Displays merupakan *toolkit prototype* menggunakan proyektor dan *Microsoft Kinect*, di mana *game* dapat ditampilkan dan dimainkan pada semua jenis bidang permukaan. Ubi Displays ini akan membuat *space* (ruangan) dari sebuah *game* menjadi tidak terbatas, orang lain dapat melihat jalannya permainannya. Selain itu, Ubi Displays juga merupakan teknologi yang bersifat NUI (*Natural User Interactive*) yang mana akan tercipta interaksi dari *game* motorik menjadi lebih natural.

Penelitian ini merujuk pada penelitian-penelitian sebelumnya di antaranya oleh Iriyanto (2007) yang meneliti tentang pengembangan aplikasi *game* karambol 3 dimensi. Iriyanto mengemas *game* karambol dengan tampilan 3 dimensi dalam media elektronik. Hardy (2013) merancang konsep Ubi Display *as toolkit for the rapid creation of interactive projected*, ia menjelaskan tentang pembuatan web interaktif dengan menggunakan Ubi Displays. Serta Alexander (2012) meneliti tentang *toolkit support for interactive projected displays*.

Berdasarkan latar belakang di atas, maka penulis menyusunnya dalam sebuah penelitian dengan judul "Pengembangan *Game* Karambol pada *Interactive Projected Display*".

2. LANDASAN KEPUSTAKAAN

A. *Game* Karambol

Game karambol adalah salah satu *game* di Indonesia yang sudah populer di masyarakat, *game* karambol ini bukanlah suatu *game* yang rumit seperti strategi atau taktik melainkan *game* ketangkasan sangat berperan selain dari kejelian menentukan arah dan sudut.

Permainan *carrom* atau karambol telah dimainkan kira-kira 2000 tahun yang lalu di zaman Makam Tua Abyssinian. *Game* ini menyebar luas ke berbagai negara lainnya pada tahun 1970-an. *Carrom* telah dimainkan oleh orang-orang India bersama keluarga di rumah atau di tanah lapang (Sahid, 2011: 1-2). *Game* karambol tidak hanya ada di Indonesia tapi di beberapa negara lain. (Iriyanto, 2007: 18).



Gambar 1. Permainan Karambol

Berikut ini peraturan yang pasti dalam bermain karambol:

- *Player* harus memilih warna koin yang ingin dimasukkan, dan seterusnya pemain harus memasukan koin sesuai dengan warna yang sama sesuai warna yang pertama kali dimasukkan sampai koin dengan warna yang sama habis.
- Maskot dapat dimasukkan setelah pemain dapat memasukan semua koin dengan warna yang sama.
- Jika pemain memasukan maskot sebelum semua koin dengan warna yang sama habis maka pemain akan kalah.
- Dalam menyentil koin gacoan tidak diperbolehkan lebih dari garis yang sudah ditentukan.

Saat menembak koin boleh memantulkan gacoan pada pembatas maupun tembak langsung (Iriyanto, 2007: 21-22)

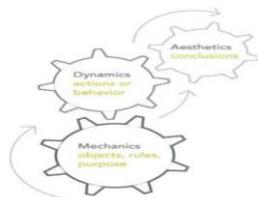
B. MDA Framework

Game dibuat oleh desainer/tim pengembang, dan digunakan oleh pemain. *Game* dibeli, digunakan, kemudian tidak digunakan seperti barang pada umumnya. Perbedaan antara *game*

dan produk hiburan lainnya adalah konsumsi yang relatif tidak bisa ditebak (Hunicke, et al., 2001).

MDA (*Mechanics, Dynamics, and Aesthetics*) adalah pendekatan secara formal untuk mengerti *game*. MDA adalah salah satu cara untuk menjembatani jarak antara desain *game* dan pengembangan, kritik *game*, dan penelitian teknis *game*. MDA menunjukkan hubungan antara *rules* (peraturan) dan pengalaman yang diperoleh oleh pemain, dan juga hubungan antara pemain dan desainer (Schreiber, 2009).

MDA Framework adalah sebuah kerangka kerja yang menjelaskan tentang mekanika (M), dinamika (D), dan estetika/keindahan (A). Framework ini digunakan untuk memahami hubungan antara mekanik yang dapat mempengaruhi dinamika, yang selanjutnya dapat mempengaruhi keindahan. MDA terdiri dari tiga bagian utama, yaitu mekanik, dinamika, dan estetika yang saling mempengaruhi satu dengan yang dapat digambarkan seperti pada Gambar 2.7.



Gambar 2 Keterkaitan MDA Framework

Analisis penerapan antarmuka alaminya pada buku Interaktif *Augmented Reality ARca* Menggunakan MDA *framework* adalah sebagai berikut:

1. *Mechanics* bisa dikatakan peraturan dalam *game*. Hal ini merupakan batasan-batasan dalam pengoperasian *game*, juga menjelaskan bagaimana sebuah *game* berakhir, dan bagaimana akhirnya ditentukan.
2. *Dynamics* menjelaskan bagaimana *game* berjalan ketika peraturan sudah ditetapkan..
3. *Aesthetics* (dalam pengertian MDA) bukan merujuk pada elemen visual dari sebuah *game*, melainkan lebih kepada pengalaman yang diperoleh pemain ketika memainkan *game* dan efek yang diperoleh pemain dari elemen *dynamics*

C. Paper Prototyping

Paper prototyping adalah membuat *prototype* secara non-digital baik itu menggunakan kertas, *cardboard*, atau apapun. Ide dasar dari *paper prototyping* adalah *game* dapat disimulasikan secara langsung dan dapat dimainkan tanpa komputer. (Schreiber, 2009). Berikut ini kelebihan *paper prototype*:

1. Murah dan sebagian besar sistem dapat dibuat *prototypenya* dengan pensil dan kertas.
2. Cepat, tidak perlu berkuat dengan program, hanya perlu menuliskan dalam secarik kertas.
3. Mudah untuk diubah.
4. Tidak merasa sayang jika tidak jadi digunakan.
5. Bisa digunakan untuk hampir semua sistem *game*.
6. Dengan membuat sesuatu yang bisa dimainkan, desainer dipaksa untuk mendesain sebuah sistem.

D. Unity

Unity adalah sebuah aplikasi *game engine* yang memungkinkan penggunaanya dapat membuat sebuah *game* dengan mudah dan cepat. Secara default, Unity telah diatur untuk pembuatan *game* ber-genre *First Person Shooter* (FPS), namun Unity juga bisa digunakan untuk membuat *game* bergenre *Role Playing Game* (RPG), dan *Real Team Strategy* (RTS). Selain itu Unity merupakan sebuah *engine multiplatform* yang memungkinkan *game* yang telah dibuat dapat digunakan untuk berbagai *platform* seperti Windows, Mac, android, IOSS, PS3 dan juga Wii (Roedavan, 2014).

E. Ubi Display

Ubi Displays adalah *toolkit prototype* untuk membuat tampilan interaktif menggunakan proyektor dan kinect. Alat ini memungkinkan untuk *drag* dan *drop* konten web interaktif dalam dunia di sekitar kita. Dapat digunakan untuk prototipe jenis baru antarmuka komputer untuk desain interior dan furniture. Anggap saja sebagai lingkungan pemrograman untuk ruang fisik.

Ubi Display menggunakan API Javascript, tampilan mampu bergerak, muncul, dan menghilang. Ini membantu mereka untuk menyatu dengan estetika fisik dari ruang atau instalasi, juga dapat memilih metode interaksi sendiri: multi-touch, sentuhan, deteksi kaki,

detektor kopi mug atau kita dapat membuat jenis sentuhan yang lain.

F. Bahasa Pemrograman C#

C# (dibaca: C sharp) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka NET Framework. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain dengan beberapa penyederhanaan.

G. Pengujian Perangkat Lunak

Pengujian perangkat lunak adalah hal yang penting dalam QA perangkat lunak dan menunjukkan ulasan dari spesifikasi, desain, dan pembuatan kode program (Pressman, 2001). Berikut ini beberapa aturan dalam melakukan pengujian.

1. Pengujian adalah sebuah proses eksekusi program dengan tujuan menemukan kesalahan
2. *Test case* yang baik adalah *test case* yang memiliki kemungkinan tinggi dalam menemukan kesalahan yang belum pernah ditemukan.
3. Pengujian yang berhasil adalah pengujian yang menemukan kesalahan yang belum pernah ditemukan.

Segala produk teknis dapat diuji dengan dua cara. Yang pertama adalah mengetahui fungsi spesifik yang didesain untuk sebuah produk. Pengujian dapat dilakukan dengan melakukan demonstrasi setiap fungsi untuk mengetahui apakah produk menjalankan fungsinya, dan pada saat yang sama juga mencari kesalahan pada setiap fungsi. Yang kedua adalah mengerti kerja internal dari produk. Pengoperasian secara internal dilakukan berdasarkan spesifikasi. Pendekatan pertama disebut *black box testing*, sedangkan pendekatan kedua disebut *white box testing*.

1. White Box Testing

White box testing adalah cara pengujian yang memberikan kesempatan bagi pengujian untuk menguji baris program secara langsung yang tidak mungkin akan dilakukan oleh pengguna tingkat akhir (Schultz, 2005). Menguji sebuah *game* hanya dengan menggunakan metode *white box testing* juga sangatlah sulit karena

hampir tidak mungkin untuk mencatat kompleksitas dari *feedback loop* pemain. Namun ada beberapa hal yang membuat *white box testing* lebih praktikal dan dibutuhkan daripada *black box testing*. Berikut ini contohnya:

- a. Pengujian dilakukan oleh pengembang untuk membuat kode baru yang diintegrasikan dengan seluruh *game*.
- b. Pengujian kode yang akan digunakan sebagai bagian dari *reusable library* yang akan digunakan untuk banyak *game* dan/atau *platform*.
- c. Pengujian kode atau fungsi yang penting dalam *game engine* atau produk untuk pengembangan.
- d. Pengujian kode dalam *game* yang mungkin akan digunakan oleh pihak ketiga.
- e. Pengujian routine level rendah yang digunakan dalam *game* untuk mendukung fungsi tertentu pada hardware terbaru seperti *graphic card* atau *audio processor*.

Dalam melakukan *white box testing*, pengujian menguji modul tertentu pada setiap alur kode ketika modul digunakan dalam berbagai macam. Masukan dalam tes ditentukan oleh jenis dan nilai dari data yang dapat dimasukkan ke dalam kode. Hasilnya ditentukan oleh nilai yang dikembalikan oleh modul, variabel global yang dipengaruhi oleh modul, dan variabel lokal yang diproses dalam modul.

2. BlackBox Testing

Hampir dalam semua proses *testing game* adalah *black box testing*, yaitu pengetesan yang dilakukan di luar aplikasi. Dalam melakukan *black box testing* tidak dibutuhkan pengetahuan atau akses ke dalam kode program. *Game tester* pada dasarnya tidak mencari kesalahan dengan membaca kode program, melainkan dengan memainkan *game* seperti yang dilakukan pemain pada umumnya. *Black box testing* adalah cara paling efektif untuk mengetes sistem yang sangat kompleks atau *video game* yang sangatlah sederhana.

3. METODOLOGI

Pada bab ini akan dijelaskan mengenai tahapan yang akan dilakukan dalam pembuatan *game* dalam skripsi yang berjudul "Pengembangan *Game* Karambol pada

Interactive Projected Display". Tahapan yang dilakukan dalam pembuatan skripsi ini meliputi studi literatur dan perancangan *game*. Terdapat pula kesimpulan dan saran yang ditujukan sebagai catatan atas aplikasi untuk digunakan sebagai acuan pengembangan aplikasi selanjutnya.

Perancangan *game* dibangun setelah melakukan studi literatur dan mengumpulkan dasar teori dari studi literatur tersebut.

Tabel 1. Konsep dan Keterangan *Game*

Elemen	Keterangan
Judul	Karambol
ESRB Rating	Everyone. Semua umur
Platform	Projected Display
Genre	Sports

Cara bermainnya yaitu setiap pemain harus mengeluarkan koin (buah karambol) terlebih dahulu dari dalam lingkaran yang berada di tengah meja karambol, setelah itu setiap *player* akan bergantian giliran untuk memasukan koin sebanyak – banyaknya kedalam lubang yang berada pada setiap sudut meja karambol, *player* yang berhasil memasukan koin akan mendapat giliran menembak sekali lagi , setiap koin memiliki nilai yang berbeda 50 untuk koin warna merah, 100 untuk koin berwarna biru, 500 untuk koin maskot, *player* yang berhasil mengumpulkan *score* terbanyak akan menjadi pemenang.

4. PERANCANGAN DAN IMPLEMENTASI

Elemen-elemen formal tersebut adalah *player*, *goal*, *resource* dan *resource management*, *game state*, *information*, *sequencing*, *player interaction*, *theme*, dan *game as system*. *Game* karambol adalah salah satu *game* yang ber-genre “*board game*” yang bisa dianggap modifikasi dari *game* biliyard.

Tabel 2.Deskripsi Game Karambol

Elemen	Keterangan
Judul <i>Game</i>	Karambol
Platform	PC / Ubi display
Target Usia	3 – ∞
Rating ESRB	E (<i>Everyone</i>)
Genre	<i>Board game</i>
Unique Selling Point (USP)	1. Sebuah desain <i>game</i> 2D yang menarik .

2. Sebagai media melatih ketangkasan.
3. Dengan bantuan ubi display, maka *game* ini dapat dimainkan pada semua padat dengan permukaan datar.

Berikut ini merupakan lingkungan perangkat keras yang digunakan untuk mengembangkan *game* karambol pada interactive projected display.

1. Prosesor : Intel(R) Core(TM) i7-6700HQ CPU @2.60-GHz (8CPU)
2. RAM : 8,00 GB
3. Graphic : Intel(R) HD Graphics 530 Card
4. Monitor : 15.6” HD LED LCD
5. Kapasitas Harddisk : 1000 GB
6. Kinect : XBOX 360
7. Projector : EPSON EB-S100

Berikut ini merupakan perangkat lunak yang digunakan untuk mengembangkan *game* karambol pada interactive projected display.

1. Sistem Operasi Windows 10 Pro 64-bit.
2. Unity 4.6.f1 versi *free*.
3. MonoDevelop
4. Visual Studio 2010.
5. Kinect for Windows SDK V1.8
6. UBI 5.4 *Trial Edition*

5. PENGUJIAN

Pengujian unit adalah salah satu bentuk pengujian *white box* dengan menggunakan teknik *basis path testing*. Pada basis path testing, pengujian dibuat berdasarkan ukuran tingkat kompleksitas dari algoritma hasil perancangan. Pengujian unit yang dilakukan untuk penulisan laporan skripsi ini hanyalah pada fungsi-fungsi paling penting dalam program.

Method
Start_____1
Choose Pemain 2,3,4_____2
Get Pemain_____3
Pemain drag Scroll_____4
Get sudut tembak gacoan__5
Pemain drag speed_____6
Get speed tembak gacoan__7
Drag gacoan _____8
Get tembakan gacoan_____9
If koin masuk ke lubang _____10
Score update_____11
Get Play again_____12
If koin tidak ada yang masuk__13
Get Ganti Pemain_____14
koin habis _____15
call winner_____16
Get choose play again or exit_____17
If play again_____18
Return
If exit _____19
Get main menu_____20
End if_____21

Gambar 3. Pengujian Unit *Gameplay*

Untuk menghitung kompleksitas siklomatik (*Cyclomatic complexity*), digunakan persamaan $V(G)=E-N+2$. Dimana E adalah edge atau jumlah garis yang ada pada flow graph, lalu N adalah jumlah node.

$$V(G)=E-N+2.....(1)$$

$$V(G)=23-21+2.....(2)$$

$$V(G)=4.....(3)$$

Jalur independen:

- 1-2-3-4-5-6-7-8-9-10-11-12-15-16-17-19-20-21
- 1-2-3-4-5-6-7-8-9-13-14-4
- 1-2-3-4-5-6-7-8-9-10-11-12-4
- 1-2-3-4-5-6-7-8-9-10-11-12-15-16-17-18-3

Pada tabel 5.1 adalah hasil pengujian yang dilakukan pada method Update. Jalur diperoleh dari perhitungan kompleksitas siklomatik dari method Update. Ada 6 kasus uji sesuai dengan jalur yang diperoleh pada perhitungan sebelumnya. Untuk lebih jelasnya silakan lihat pada tabel 3.

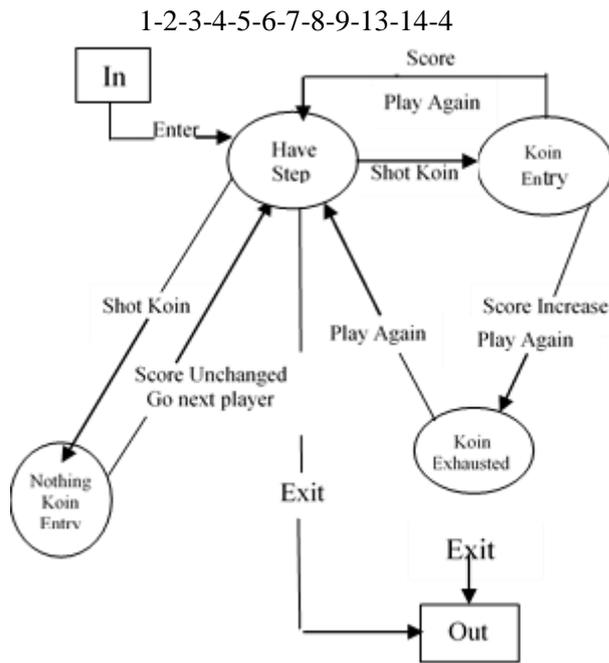
Tabel 3. Pengujian Unit Kondisi Menang dan Kalah

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Koin hijau masuk ke dalam lubang	Koin hijau yang masuk <i>destroy score</i> bertambah 100 dan pemain	Koin hijau yang masuk <i>destroy score</i> bertambah

		mendapat giliran lagi	100 dan pemain mendapat giliran lagi
2	Koin biru masuk ke dalam lubang	Koin biru yang masuk <i>destroy score</i> bertambah 200 dan pemain mendapat giliran lagi	Koin biru yang masuk <i>destroy score</i> bertambah 200 dan pemain mendapat giliran lagi
3	Koin maskotan masuk ke dalam lubang	Koin maskotanyang masuk <i>destroy score</i> bertambah 1000 dan pemain mendapat giliran lagi	Koin maskotanyang masuk <i>destroy score</i> bertambah 1000 dan pemain mendapat giliran lagi
4	Tidak ada koin yang masuk	<i>Score</i> tidak berubah dan giliran berganti ke pemain yang lain	<i>Score</i> tidak berubah dan giliran berganti ke pemain yang lain
5	Pemain tidak melakukan apapun	Langkah dan skor tidak berubah	Langkah dan skor tidak berubah
6	Koin habis	Panel win muncul dan menampilkan player yang <i>score</i> -nya tertinggi dan menampilkan <i>button play again</i> dan <i>exit</i>	Panel win muncul dan menampilkan player yang <i>score</i> -nya tertinggi dan menampilkan <i>button play again</i> dan <i>exit</i>

Pengujian black box dilakukan dengan metode *Test Flow Diagram* (TFD). TFD adalah model grafis yang merepresentasikan *behavior* dari perspektif pemain. TFD menyediakan pendekatan formal untuk mengetes sebuah desain.

Jalur yang ada dihitung dengan minimum path generation. Dengan ini pengujian dapat melakukan tes dengan jumlah yang sedikit dan lebih menyerupai. Jalur yang berhasil didapatkan berdasarkan gambar 5.7 adalah:



Gambar 4. Pengujian *Gameplay 3 Match* dengan TFD

Pada pengujian dengan TFD hanya dihasilkan satu jalur. Hasil pengujian yang diperoleh berdasarkan kasus uji dari jalur yang didapatkan bisa dilihat pada tabel 4.

Tabel 4. Pengujian pada *Gameplay*

Jalur	Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan
1	Pengujian prosedur pada <i>gameplay</i>	Menghasilkan hasil yang sesuai pada kebutuhan dan fungsionalitas tanpa ada <i>bug</i> pada setiap prosedur jika mendapat <i>item</i> pada <i>gameplay</i> collecting <i>item</i>	Menghasilkan hasil yang sesuai pada kebutuhan dan fungsionalitas tanpa ada <i>bug</i> pada setiap prosedur jika mendapat <i>item</i> pada <i>gameplay</i> collecting <i>item</i> Namun jika menggunakan <i>unity</i> terdapat <i>bug</i> pada saat melakukan <i>shot coin</i> karena keterbatasan posisi <i>camera</i>

Hasil Pengujian Kepada Pengguna (*Focus Testing*)

Pengujian kepada pengguna digunakan untuk memastikan bahwa *game* yang dibuat telah sesuai dengan tujuan saat perancangan, yaitu membuat *game* yang menyenangkan dengan media interactive display.

Tabel 5. Pengujian *Game* terhadap Beberapa Siswa SMP

Pertanyaan	Jumlah Jawaban		
	Ya	Biasa	Tidak
Apakah <i>game</i> ini menarik?	17	12	1
Apakah <i>game</i> ini menyenangkan untuk dimainkan?	17	12	1
Apakah desain <i>game</i> ini menarik?	19	7	4
Apakah permainan karambol menggunakan Display Interactive menarik?	18	7	5

Hasil Analisis *White Box Testing*

Berdasarkan hasil pengujian unit yang dilakukan, jumlah jalur pada logika setiap *method* sesuai dengan perhitungan *cyclomatic complexity*. Setiap kasus uji yang dibuat telah sesuai berdasarkan jumlah jalur yang telah diuji dan memberikan hasil sesuai dengan yang diharapkan. Dari hasil tersebut dapat ditarik kesimpulan bahwa unit modul dari program sudah memenuhi kebutuhan fungsional yang telah dirancang pada tahap perancangan.

Hasil Analisis *Black Box Testing*

Berdasarkan kesesuaian antara hasil uji terhadap implementasi dan fungsionalitas setiap prosedur yang diuji pada *game "Karambol"*, menghasilkan output yang diharapkan dalam daftar kebutuhan *game* ini. Maka dapat disimpulkan bahwa implementasi dan fungsionalitas *game* telah memenuhi kebutuhan yang telah dijabarkan dalam daftar kebutuhan.

Hasil Analisa *Focus Testing*

Berdasarkan hasil pengujian yang dilakukan terhadap siswa MTs Wahid Hasyim 01 Dau *game "Karambol"* dapat diterima dengan cukup baik namun karena setting tempat dan sensor kamera yang masih terbatas maka mengakibatkan banyak crash saat melakukan control yang bisa dilihat pada tabel 5.4. Jumlah jawaban "ya" pada semua pertanyaan melebihi 50%. Jawaban cukup lebih dari 20% untuk

semua pertanyaan. Jawaban “tidak” pada semua pertanyaan tidak melebihi 17%.

Tabel 6. Analisa Hasil Pengajian terhadap Pengguna

No	Pertanyaan	Jumlah Jawaban		
		Ya	Cukup	Tidak
1	Apakah <i>game</i> karambol ini menarik?	65%	32%	3%
2	Apakah <i>game</i> ini menyenangkan untuk dimainkan?	60%	37%	3%
3	Apakah desain <i>game</i> ini menarik?	68%	28%	4%
4	.	75%	25%	0%
Rata-rata		67%	30,5%	2,5%

Dari hasil yang didapatkan bisa ditarik kesimpulan bahwa *game* telah sesuai dan memenuhi hasil yang diharapkan. Penggunaan *interactive projected display* dapat membuat *game* menjadi lebih menarik.

6. PENUTUP

A. Kesimpulan

Berdasarkan hasil analisa, implementasi, dan pengujian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

- 1 Proses perancangan *game* karambol pada *interactive projected display* dilakukan dengan bantuan paper prototype. Dan *gameplay* yang digunakan untuk *game* karambol pada *interactive projected display* adalah memasukkan koin pada lubang papan karambol. *Gameplay* ini mengharuskan pemain memantulkan gacoan dengan koin permainan sampai koin berhasil masuk pada lubang papan karambo, sehingga didapatkan scor. Pemain yang menang adalah pemain yang berhasil memasukkan koin terbanyak dari pemain lainnya. *Game* karambol pada *interactive projected display* diimplemetasikan dengan menggunakan *game engine* Unity dengan bahasa pemrograman C# dan Java. Aset 3D dari *game* karambol pada *interactive projected display* ini dibuat dengan menggunakan perangkat lunak Blender 2.68a dan Adobe Photoshop 7.0.
- 2 Berdasarkan hasil pengujian yang telah dilakukan, didapatkan hasil bahwa

fungsionalitas *game* karambol pada *interactive projected display* telah terpenuhi.

- 3 Berdasarkan pengujian terhadap pengguna, *game* karambol pada *interactive projected display* mendapat tanggapan yang positif yang bisa dilihat dari jawaban yang diberikan setelah memainkan permainan. Sekitar lebih dari 67% pengguna menyatakan bahwa *game* karambol pada *interactive projected display* menyenangkan untuk dimainkan dan menarik

B. Saran

Saran yang bisa diberikan untuk pengembangan selanjutnya adalah:

- 1 Diperhatikan tata letak kinect atau kamera sensor, agar tidak terjadi crash.
- 2 Ada tambahan *minigames*. Jadi dalam *game* tidak hanya ada satu *gameplay*.

DAFTAR PUSTAKA

A. Clark dan A. Dunser. 2012. An Interavtive Augmented Reality Coloring Book. *Journal of IEEE Symposium on 3D User Interfaces (3DUI)* page 7 – 10.

Alexander, J. dan J. Hardy. 2012. Toolkit Support for Interactive Projected Displays. *Journal of Mobile and Ubiquitous Multimedia (MUM)*. ACM.

Hardy, J., Carl E., Jason A., dan Nigel D.. 2013. Ubi Displays: A Toolkit for the Rapid Creation of Interactive Projected Displays. *Journal of PerDis*. June 04 – 05 2013, Mountain View, CA.

Hejlsberg, A., Torgerren, M., Wiltamuth, S., & Golde, P. (2011). *The C# Pogramming Language (Covering C# 4.0)*. Addison-Wesley Professional.

Hunicke, R., LeBlanc, M., & Zubek, R. (2001). *MDA: Formal Approach to Game Design and Game Research*. San Jose.

Iriyanto, Ron. (2007). *PENGEMBANGAN APLIKASI GAME KARAMBOL 3 DIMENSI*. S1 thesis. UAJY

Kopenol, W. H. (2002). Naming Of New Elements. *Pure Application Chemistry*, 74, 787-791.

Pedersen, R. (2008). *Game Design Foundations, 2nd Edition*. Wordware Publishing, Inc.

- Pressman, R. S. (2001). *Software Engineering, A Practitioner's Approach* (5th ed.). New York: Thomas Casson.
- Roedavan, R. (2014). *UNITY Tutorial Game Engine*. Bandung: Informatika.
- Rogers, S. (2010). *LEVEL UP, The Guide to Great Video Game Design*. West Sussex: John Willey & Sons, Ltd.
- Sahid, Dayat. 2013. *Sejarah Karom Antarbangsa*. Jakarta: Scribd (<https://www.scribd.com/doc/130954974/SEJARAH-KAROM-ANTARABANGSA>), (Online) diakses 03 Oktober 2016).
- Schlutz, C. P., Bryant, R., & Langdell, T. (2005). *Game Testing All in One*. Boston: Thomson Course TEchnology.
- Schreiber, I. (2009). *Game Design Concepts*. New York: Creative Commons Attribution 3.0.
- Win, St. 2011. *Cara Bermain Karambol*. Jakarta: Scribd (<https://www.scribd.com/doc/48847412/Cara-Bermain-karambol>), (Online) diakses 03 Oktober 2011).