

PERANCANGAN GAME EDUKASI SEBAGAI MEDIA PEMBELAJARAN BERBASIS *MOBILE* MENGGUNAKAN ALGORITMA *FISHER-YATES* DAN *FLOOD FILL*

Rifky Syaifulloh¹, Septi Andryana², Aris Gunaryati³

Informatika, Fakultas Teknologi Komunikasi dan Informatika, Universitas Nasional
Jalan Sawo Manila, Pasar Minggu, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta

rifkysyaifulloh03@gmail.com¹, septi.andryana@civitas.unas.ac.id²,
aris.gunaryati@civitas.unas.ac.id³

Abstract

Information technology is currently increasingly developing, especially on smartphone devices. The reason various users use smartphone devices is their practical size and of course their portability. Smartphone devices can also be a learning medium in the form of games or games, one of which is games for children's education. The problem that often occurs in children in the way of learning is that it is easy to develop boredom if the child is only encouraged by learning material without any entertainment or games. This game is designed to change the way of learning, especially in early childhood between the ages of 3 to 6 years. This mobile-based game is made using the App Inventor with an interactive and attractive display to learn to recognize numbers, fruit, letters, animals, vegetables and colors as well as quiz questions. In this educational game that was made, it was designed using the Rapid Application Development (RAD) method as a completion stage starting from the planning stage to the implementation stage of the program. This educational game also applies the Fisher-Yates algorithm to randomize questions. The results of this study show 10 quiz questions randomly from 15 existing questions, and apply the Flood Fill algorithm for coloring objects or images. With the design of games for mobile-based education, it is hoped that it can make children's brains more active and creative during their growth period and help children's learning processes in order to increase their broader knowledge.

Keyword: *App Inventor, Fisher Yates, Flood Fill, Education Game, RAD*

Abstrak

Teknologi informasi pada saat ini semakin hari semakin berkembang, khususnya pada perangkat smartphone. Alasan dari berbagai pengguna menggunakan perangkat smartphone adalah ukurannya yang praktis dan tentunya mudah dibawa. Perangkat smartphone juga bisa menjadi media pembelajaran dalam bentuk permainan atau game, salah satunya adanya game untuk edukasi anak. Permasalahan yang kerap terjadi pada anak dalam cara belajar yaitu mudah timbulnya kebosanan jika anak hanya didorong oleh materi pembelajaran saja tanpa adanya hiburan atau permainan. Dirancang game ini untuk mengubah cara belajar khususnya pada anak usia dini antara umur 3 sampai 6 tahun. Pada game berbasis mobile ini dibuat menggunakan App Inventor dengan tampilan yang interaktif dan menarik untuk belajar mengenal angka, buah, huruf, hewan, sayur dan warna serta adanya soal kuis. Pada game edukasi ini yang dibuat ini dirancang menggunakan metode Rapid Application Development (RAD) sebagai tahapan penyelesaian mulai dari tahap perencanaan sampai tahap implementasi dari program. Pada game edukasi ini juga

menerapkan algoritma Fisher-Yates untuk melakukan pengacakan soal. Pada hasil penelitian ini ditampilkan 10 soal kuis secara acak dari 15 soal yang ada, serta menerapkan algoritma Flood Fill untuk pewarnaan objek atau gambar. Dengan dirancangnya game untuk edukasi berbasis mobile ini diharapkan dapat membuat otak anak lebih aktif dan kreatif dimasa pertumbuhannya serta membantu proses belajar anak agar dapat menambah pengetahuan yang lebih luas.

Kata kunci: *App Inventor, Fisher Yates, Flood Fill, Game Edukasi, RAD*

1. PENDAHULUAN

Semakin berkembangnya zaman, teknologi komputer banyak sekali menghadapi perubahan yang dapat membuat semua manusia untuk dapat menggunakannya secara mudah dari segi apapun. Salah satu dari pemanfaatan teknologi yaitu dapat membuat sebuah aplikasi sekaligus game. Dalam game terdapat fitur yang tidak hanya berfungsi sebagai media hiburan, tetapi game sudah banyak untuk mengubah pola pikir yang dapat memperkenalkan pengetahuan agar dipahami dan diterima khususnya untuk anak yang masih dalam usia dini. Pada usia dini ini, mereka berada pada masa pertumbuhan dan perkembangan yang sangat pesat dari segi fisik ataupun mental. Masalah yang biasa terjadi pada anak dalam belajar yaitu bisa saja anak menjadi stres jika hanya didorong oleh materi pembelajaran saja tanpa adanya hiburan atau permainan. Salah satu cara untuk mendorong tingkat kreatifitas belajar pada anak yaitu dengan cara bermain sambil belajar serta menjadi cara alternatif untuk menghilangkan rasa stres pada anak. Pada game edukasi ini dapat diterapkan pada smartphone sehingga dapat mudah dibawa kemana saja dan kapan saja. Berdasarkan hasil literatur yang digunakan, untuk aplikasi game edukasi yang lain sejenis ini belum ada yang menggunakan 2 algoritma sekaligus. Jika mungkin ada tetapi kasus yang dibuat berbeda.

Game edukasi ini dibuat dengan App Inventor dimana sistem berbasis web yang dibuat dengan cara men-drag-and-drop blok kode program yang telah disediakan untuk menciptakan sebuah aplikasi game yang dapat dipakai pada perangkat smartphone. Penelitian ini bertujuan untuk membuat game bertema edukasi yang dapat digunakan sebagai media pembelajaran untuk anak usia dini dengan menggunakan App Inventor agar dapat menambah pengetahuan lebih luas dan membuat otak anak lebih aktif dan kreatif dimasa pertumbuhannya.

2. METODOLOGI PENELITIAN

Metodologi yang diterapkan untuk membuat aplikasi game edukasi ini yaitu menggunakan teknik pengumpulan data yang dibutuhkan untuk penelitian. Penelitian ini menggunakan metode RAD (*Rapid Application Development*). Tujuan dari digunakannya metode ini yaitu memberikan sebuah sistem untuk memenuhi harapan dari pengguna, tetapi sering kali pada saat melakukan pengembangan sebuah sistem tidak melibatkan pengguna sistem tersebut secara langsung, sehingga dapat menyebabkan sistem yang dibuat tidak digunakan bagi para penggunanya bahkan menolak untuk memakainya. [1]



Gambar 1. Model Metode *Rapid Application Development*

Berikut beberapa fase yang digunakan dalam metode *Rapid Application Development*, antara lain :

2.1. Perencanaan

Perancangan game edukasi sebagai media pembelajaran berbasis mobile dengan MIT App Inventor ini diharapkan dapat membantu proses pembelajaran anak dalam mengenal buah, hewan, angka, huruf, sayur, warna dan kuis serta menambahkan opsi mewarnai objek atau gambar yang diperoleh dari file manager smartphone sehingga perancangan game ini menerapkan 2 algoritma agar menciptakan keterbaruan dan tidak sama seperti literatur yang sebelumnya.

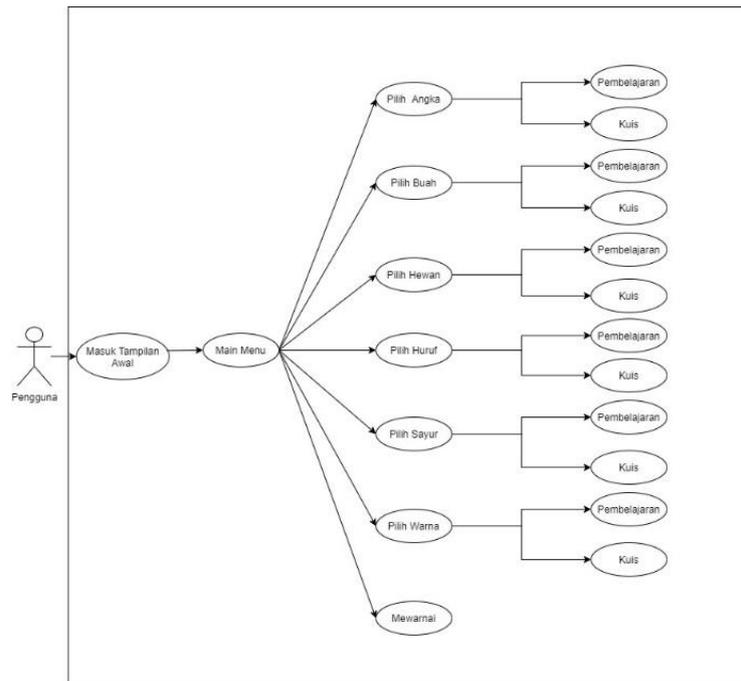
Pada fase Perencanaan terdapat sebuah analisis sistem kebutuhan, analisis ini merupakan berbagai kebutuhan dalam suatu sistem yang akan dipakai untuk menjalankan proses perancangan suatu game tersebut.

2.2. Desain Sistem

Dalam proses pembuatan sebuah game edukasi sebagai media pembelajaran berbasis mobile terdapat tahapan yang dipakai sebagai pemodelan dari aplikasi ini. Pada tahapan desain ini yaitu dengan membuat diagram *use case* dan diagram *flowchart* untuk alur sistem ataupun alur datanya.

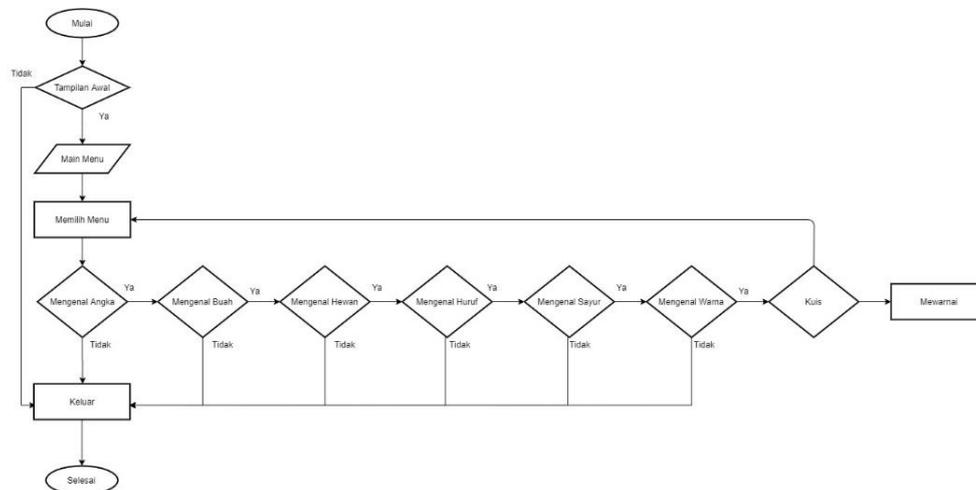
Perancangan desain sistem dilakukan dengan alasan memberikan gambaran umum sebuah sistem yang akan dirancang. Pada tahap ini peneliti mulai membuat diagram *use case* dan diagram *flowchart* untuk tampilan – tampilan visual game edukasi ini.

Pada game edukasi ini, diagram *use case* digunakan untuk menggambarkan bagaimana suatu alur proses pengguna menggunakan aplikasi tersebut sesuai dengan sistem yang telah dibuat.



Gambar 2. Perancangan *Use Case* Pengguna

Selanjutnya untuk mengetahui alur game edukasi ini yaitu menggunakan *flowchart*. *Flowchart* ini dibuat sebagai bahan acuan bagi peneliti dalam merancang sebuah desain game edukasi ini, agar proses pengerjaan dapat dengan mudah dipahami dan diikuti oleh pengguna.



Gambar 3. *Flowchart* proses Game Edukasi

2.3. Implementasi

Setelah proses pemodelan game edukasi sebagai media pembelajaran berbasis mobile telah selesai dibuat, kemudian masuk ke proses memasukkan source code dengan cara men drag & drop blok kode program yang telah disediakan oleh App Inventor berdasarkan desain yang dibuat pada tahap sebelumnya. Setelah proses dimasukkannya blok program, kemudian dilakukan tahap uji coba kepada pengguna serta menerapkan kedua algoritma yaitu *Fisher-Yates* dan *Flood Fill*.

2.3.1. Implementasi Algoritma Fisher-Yates

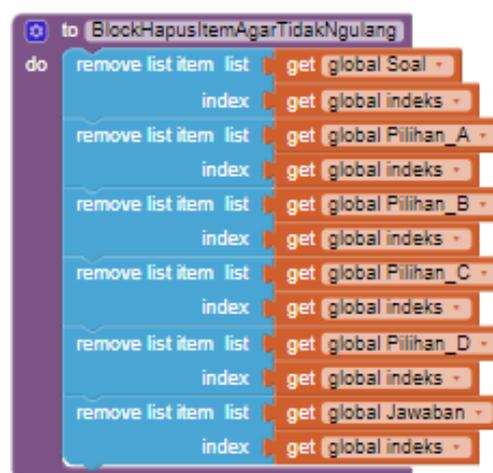
Input dari proses pengacakan dengan algoritma *Fisher-Yates* merupakan sebuah pengambilan pada data seperti pada angka ataupun pada sebuah karakter dengan sebanyak 1-N, kemudian ambil data acak pada K diantara 1-N, setelah itu coret data pada K yang belum dicoret, lalu tuliskan sebuah data ini di tempat lain, kemudian ulangi alur hingga seluruh data sudah di coret sehingga dapat menghasilkan sebuah data permutasi baru. [5]

Pada perancangan game edukasi menggunakan App Inventor ini dengan menerapkan algoritma *Fisher-Yates* terdapat blok program untuk melakukan pengacakan soal, yaitu ada prosedur dengan nama acak_soal dengan memakai fungsi kondisi *If* dan *Else*. Kemudian terdapat list dengan pilihan jika list soal habis dari global soal yang dibuat, maka akan menuju ke tampilan hasil kuis untuk menampilkan hasilnya dengan mengirimkan global niainya. Jika soal belum habis maka akan membaca global indeks secara acak dari soal 1 sampai banyaknya soal yang ada yaitu sampai soal 15 dan selanjutnya memanggil prosedur panggil_soal untuk memulai menjawab kuis secara acak.



Gambar 4. Blok Program Pengacakan Soal

Selanjutnya membuat blok program kembali agar soal yang ditampilkan secara acak tidak tampil berulang, yaitu menambahkan prosedur dengan menggunakan blok program *remove for item list* dari list global soal sampai global pilihan serta jawaban yang dipilih.



Gambar 5. Blok Program Menonaktifkan Soal Berulang

Implementasi dari algoritma *Fisher-Yates* pada game edukasi ini adalah melakukan pengacakan soal kuis yang ditampilkan sebanyak 10 dari 15 soal yang ada.

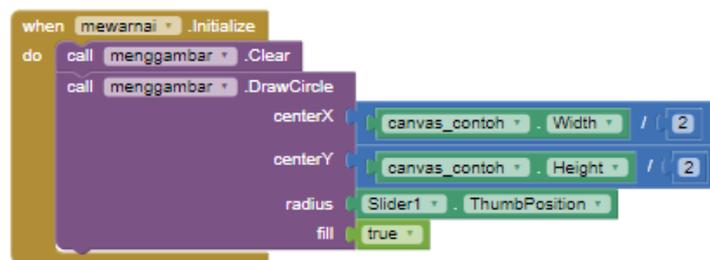
Tabel 1. Proses Perhitungan Algoritma Fisher-Yates

<i>Range</i>	<i>Roll</i>	<i>Scratch</i>	<i>Result</i>
		123456789101112131415	
1 - 15	14	1234567891011121315	14
1 - 14	4	123567891011121315	4
1 - 13	9	12356781011121315	9
1 - 12	5	1236781011121315	5
1 - 11	2	136781011121315	2
1 - 10	13	1367810111215	13
1 - 9	15	13678101112	15
1 - 8	11	136781012	11
1 - 7	8	13671012	8
1 - 6	1	3671012	1
1 - 5	6	371012	6
1 - 4	10	3712	10
1 - 3	7	312	7
1 - 2	3	12	3
Hasil Pengacakan		12 3 7 10 6 1 8 11 15 13 2 5 9 4 14	

2.3.2. Implementasi Algoritma *Flood Fill*

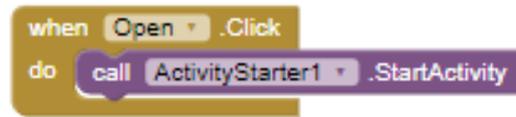
Algoritma *Flood Fill* merupakan sebuah algoritma yang berfungsi menentukan sebuah daerah yang terhubung dari berbagai urutan node. Algoritma ini memakai 3 parameter diantaranya node awal, sasaran warna dan warna pengganti. Tujuannya menemukan semua node yang terhubung dari node awal dari warna sasaran, kemudian melakukan perubahan dengan warna pengganti. [6]

Untuk penerapan algoritma *Flood Fill* pada perancangan game edukasi ini terdapat pada blok program mewarnai dengan menggunakan prosedur yang berisi kanvas yang berfungsi sebagai wadah dari image sprite yang akan dimasukkan untuk diwarnai beserta ukuran yang diinginkan. Kemudian terdapat blok Slider untuk mengubah ukuran dari pointer warna yang dipilih.



Gambar 6. Blok Program Mewarnai

Proses memasukkan gambar atau objek berformat “.png” dari perangkat smartphone yang ingin diwarnai menggunakan blok program memanggil ActivityStarter, yang berfungsi untuk menyambungkan App Inventor dengan file manager dengan menambahkan isi “android.intent.action.GET_CONTENT” pada aksi ActivityStarter tersebut.



Gambar 7. Blok Program Input Gambar

Selanjutnya membuat blok program yang berisi ImageSprite yang berfungsi agar gambar atau objek dapat berinteraksi pada sentuhan dan tarikan serta berinteraksi dengan sprite lainnya pada kanvas. Untuk ukuran imageSprite yang akan dipilih akan muncul secara otomatis



Gambar 8. Blok Program Komponen Activity Starter

Implementasi algoritma *Flood Fill* pada game edukasi ini adalah untuk mewarnai gambar / objek warna transparant apapun yang bisa diakses dari file manager smartphone pengguna.

3. HASIL DAN PEMBAHASAN

3.1. Tampilan (*Interface*)

Hasil tampilan aplikasi game termasuk Implementasi pada metode *Rapid Application Development*. Tampilan halaman hasil dari game untuk mengenali angka, buah, hewan, huruf, sayur, warna dan kuis beserta mode mewarnai berbasis mobile antara lain :

3.1.1. Tampilan Halaman Awal / Mulai

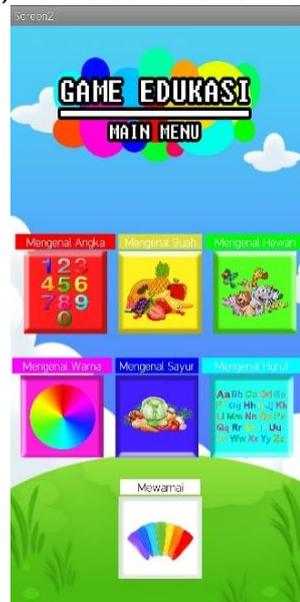
Pada halaman ini menampilkan tampilan awal / masuk ketika baru dibuka aplikasi gamenya. Terdapat pilihan mulai untuk melanjutkan ke halaman main menu dan pilihan keluar untuk keluar dari game tersebut.



Gambar 9. Tampilan Halaman Mulai

3.1.2. Tampilan Halaman Menu

Pada halaman ini menampilkan tampilan menu utama dari game tersebut. Dimana pengguna bisa memilih 7 topik pembelajaran, dari mulai belajar mengenal angka, buah, hewan, huruf, sayur, warna dan mewarnai objek gambar.



Gambar 10. Tampilan Halaman Menu

3.1.3. Tampilan Halaman Pilih Mode

Pada halaman ini menampilkan tampilan pilihan mode untuk belajar dan kuis.



Gambar 11. Tampilan Pilihan Mode

3.1.4. Tampilan Halaman Pembelajaran

Pada halaman ini menampilkan tampilan belajar. Pengguna akan diberitahu nama – nama sesuai pilihan yang dipilih pada tampilan main menu sebelumnya dan dapat mengeluarkan suara pengucapan dalam bahasa indonesia maupun bahasa inggris.



Gambar 12. Tampilan Belajar

3.1.5. Tampilan Halaman Kuis

Pada halaman ini menampilkan tampilan kuis. Pengguna akan menjawab pertanyaan yang telah disediakan secara acak dengan menerapkan algoritma Fisher-Yates.



Gambar 13. Tampilan Kuis

3.1.6. Tampilan Halaman Hasil Kuis

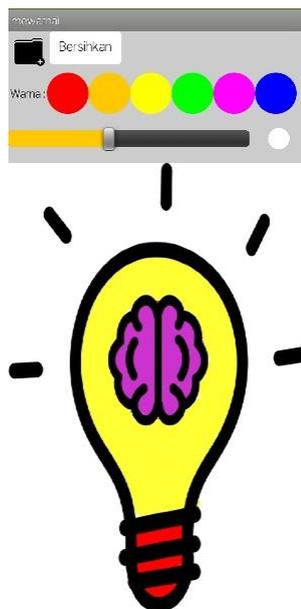
Pada halaman ini menampilkan tampilan hasil nilai dari kuis yang telah dijawab sebelumnya. Terdapat 2 pilihan yaitu untuk mengulangi kuis tersebut dan untuk menuju halaman main menu.



Gambar 14. Tampilan Hasil Kuis

3.1.7. Tampilan Halaman Mode Mewarnai

Pada halaman ini menampilkan tampilan untuk mode mewarnai. Pengguna dapat memilih objek gambar dari file manager smartphone dengan format .png dengan menerapkan algoritma *Flood Fill* untuk mewarnai objek yang di import.



Gambar 15. Tampilan Halaman Mewarnai

3.2. Pengujian

Pada uji coba berlangsung, jika aplikasi dijalankan dengan beberapa smartphone dalam waktu bersamaan tetap bisa berjalan dengan stabil dan tidak ada masalah. Penerapan algoritma *Fisher-Yates* pada game edukasi ini yaitu munculnya 10 soal acak dari 15 soal kuis yang telah disediakan pada program dan soal yang sudah muncul tersebut tidak akan keluar kembali sampai pengguna selesai menjawab kuis dan memperoleh nilainya. Dalam 1 soal yang dijawab oleh pemain bernilai 10 point/skor. Hasil uji coba dalam melakukan pengacakan soal kuis sebanyak 5 kali pada setiap menu di dapat hasil pengacakan yang berbeda beserta nilai yang didapat dan waktu pengerjaan, antara lain :

Tabel 2. Hasil Pengujian Kuis Angka

Kuis Angka			
Pemain	Soal yang keluar	Nilai Akhir	Waktu Pengerjaan
Pemain A	3, 11, 5, 12, 15, 9, 14, 1, 7, 10	80	1 Menit 2 Detik
Pemain B	15, 7, 13, 8, 12, 11, 2, 6, 9, 4	90	1 Menit 10 Detik
Pemain C	15, 9, 1, 2, 11, 6, 10, 3, 5, 8	70	57 Detik
Pemain D	11, 5, 7, 10, 12, 1, 13, 9, 4, 3	100	1 Menit
Pemain E	2, 1, 6, 10, 12, 3, 7, 11, 9, 4	80	1 Menit 5 Detik
	Rata - rata	84	1 Menit 3 Detik

Tabel 3. Hasil Pengujian Kuis Buah

Kuis Buah			
Pemain	Soal yang keluar	Nilai Akhir	Waktu Pengerjaan
Pemain A	12, 3, 8, 4, 10, 6, 13, 11, 15, 14	90	55 Detik
Pemain B	8, 15, 3, 12, 7, 10, 4, 6, 13, 11	100	1 Menit 1 Detik

Pemain C	12, 1, 5, 2, 8, 4, 13, 10, 15, 14	70	1 Menit 6 Detik
Pemain D	13, 5, 4, 8, 7, 14, 2, 3, 10, 9	80	1 Menit 12 Detik
Pemain E	8, 9, 13, 2, 15, 10, 4, 3, 6, 7	100	1 Menit 4 Detik
	Rata - rata	88	1 Menit 4 Detik

Tabel 4. Hasil Pengujian Kuis Hewan

Kuis Hewan			
Pemain	Soal yang keluar	Nilai Akhir	Waktu Pengerjaan
Pemain A	15, 3, 11, 9, 6, 4, 1, 13, 12, 2	80	1 Menit
Pemain B	5, 4, 13, 11, 1, 2, 6, 3, 12, 15	50	1 Menit 4 Detik
Pemain C	15, 4, 3, 9, 1, 11, 2, 13, 6, 10	100	1 Menit 10 Detik
Pemain D	4, 7, 9, 10, 8, 6, 3, 5, 13, 12	90	1 Menit 15 Detik
Pemain E	12, 13, 9, 6, 5, 1, 2, 3, 10, 7	100	50 Detik
	Rata - rata	84	1 Menit 4 Detik

Tabel 5. Hasil Pengujian Kuis Huruf

Kuis Huruf			
Pemain	Soal yang keluar	Nilai Akhir	Waktu Pengerjaan
Pemain A	9, 5, 15, 6, 11, 7, 14, 2, 12, 8	60	1 Menit 6 Detik
Pemain B	2, 7, 15, 4, 1, 5, 3, 14, 13, 6	60	1 Menit 13 Detik
Pemain C	13, 7, 9, 11, 6, 15, 1, 3, 2, 12	90	1 Menit 4 Detik
Pemain D	5, 11, 6, 10, 9, 1, 8, 7, 13, 12	80	1 Menit 2 Detik
Pemain E	3, 11, 2, 7, 15, 10, 8, 13, 9, 4	90	1 Menit 17 Detik
	Rata - rata	76	1 Menit 8 Detik

Tabel 6. Hasil Pengujian Kuis Sayur

Kuis Sayur			
Pemain	Soal yang keluar	Nilai Akhir	Waktu Pengerjaan
Pemain A	11, 7, 9, 2, 3, 6, 4, 10, 5, 8	100	56 Detik
Pemain B	2, 1, 6, 15, 7, 5, 4, 13, 12, 6	90	1 Menit 1 Detik
Pemain C	7, 14, 6, 10, 1, 11, 15, 5, 9, 4	70	1 Menit 3 Detik
Pemain D	10, 2, 1, 7, 15, 9, 6, 5, 11, 8	100	51 Detik
Pemain E	15, 8, 13, 12, 6, 14, 11, 3, 10, 9	90	55 Detik
	Rata - rata	90	56 Detik

Tabel 7. Hasil Pengujian Kuis Warna

Kuis Warna			
Pemain	Soal yang keluar	Nilai Akhir	Waktu Pengerjaan
Pemain A	9, 12, 6, 15, 10, 7, 1, 8, 14, 3	100	50 Detik
Pemain B	8, 3, 14, 1, 10, 7, 5, 12, 9, 13	90	52 Detik
Pemain C	2, 11, 9, 15, 13, 8, 4, 12, 7, 5	70	59 Detik
Pemain D	7, 8, 13, 1, 4, 2, 11, 15, 10, 6	100	55 Detik
Pemain E	1, 9, 4, 5, 3, 11, 6, 2, 10, 13	90	55 Detik
	Rata - rata	90	54 Detik

Pada pilihan kuis setiap menu di dalam game edukasi tersebut, dilakukan percobaan efektifitas algoritma *Fisher-Yates* dalam jumlah soal kuis yang tampil sebanyak 10 sampai 50 beserta waktu akses yang dibutuhkan.

Tabel 8. Hasil Pengujian Efektifitas Algoritma *Fisher-Yates*

No	Jumlah soal tampil	Waktu akses	Perulangan soal
1	10 Soal	706 ms	0%
2	20 Soal	1 detik 156 ms	0%
3	30 Soal	1 detik 375 ms	0%
4	40 Soal	1 detik 708 ms	0%
5	50 Soal	2 detik 021 ms	0%

4. SIMPULAN

Berdasarkan penelitian dan analisa pada game edukasi sebagai media pembelajaran berbasis mobile menggunakan App Inventor, dapat disimpulkan bahwa game ini dapat berfungsi sebagai media pembelajaran bagi anak agar berpikir lebih kreatif serta aktif pada masa pertumbuhannya. Dirancangnnya game berbasis mobile ini bisa digunakan oleh perangkat smartpone sehingga dapat dibawa kapan saja dan kemana saja. Penggunaan algoritma *Fisher-Yates* telah berhasil dan dapat digunakan untuk mengacak soal tanpa terjadinya pengulangan soal serta algoritma *Flood Fill* yang dapat mempermudah dalam proses mewarnai. Pada saat percobaan dengan 20 – 50 soal kuis mendapatkan waktu akses yang berbeda – beda serta mempunyai presentase 0% perulangan soal atau dapat dibilang tidak ada soal yang sama pada menjawab kuis. Kemudian saat percobaan mewarnai untuk mengakses gambar atau objek berformat “.png” dari file manager dan mewarnai gambar atau objek tersebut dengan warna yang telah disediakan berhasil.

DAFTAR PUSTAKA

- [1] Kristanto, Adi. 29 Oktober 2010. "**Pengertian Rapid Application Development**" <http://adikristanto.net/rapid-application-development-rad/>.

- [2] Damanik. Romanus, "**Metode Pengacakan Algoritma Fisher Yates Pada Game Edukasi Pengenalan Kosakata Bahasa Inggris**", INFORMATION SYSTEM DEVELOPMENT, VOLUME 4 NO.1, Januari 2019.
- [3] Brilliant Wijaya. Yoshua, and Utomo. Wargijono, "**RANCANG BANGUN GAME EDUKASI ANAK USIA DINI MENGGUNAKAN METODE EXTREME PROGRAMMING BERBASIS ANDROID**", JURNAL JIIFOR, Volume 1, No.1, 2017.
- [4] Andriyat Krisdiawan. Rio, and Ramdhany. Tri, "**IMPLEMENTASI ALGORITMA FISHER YATES PADA GAMES EDUKASI PENGENALAN HEWAN UNTUK ANAK SD BERBASIS MOBILE ANDROID**", JURNAL LPKIA, Vol. 11 No.2, Desember 2018.
- [5] Priantama. Rio, and Priandani. Yuda, "**IMPLEMENTASI ALGORITMA FISHER YATES UNTUK PENGACAKAN SOAL PADA APLIKASI MOBILE LEARNING KUIS FIQIH BERBASIS ANDROID**", JURNAL NUANSA INFORMATIKA, Volume 13 Nomor 2, p-ISSN : 1858-3911, e-ISSN : 2614-5405, Juli 2019.
- [6] Dini Debyanzah. Shelly, and Waworuntu. Alexander, "**Aplikasi Mewarnai Gambar Untuk Pembelajaran Pengenalan Warna Pada Anak Usia Dini Berbasis Mobile**", Jurnal Saint dan Teknologi, kalbiscentia, Volume No. 1, ISSN : 2345 - 4392, Februari 2015.
- [7] Maryono. Iyan, Budiawan Zulfika. Wildan, and Kariadinata. Rahayu, "**The implementation of fisher yates shuffle on aljabar learning media based on hybrid application**", MATEC Web of Conferences 197, 01006, 2018.
- [8] Gunawan. Wawan, and Derajad Wijaya. Herry, "**An Application of Multimedia for Basic Arabic Learning Using FisherYates Shuffle Algorithm on Android Based**", Scholars Middle East Publishers, Dubai, United Arab Emirates, ISSN 2412-9771 |ISSN 2412-897X, 2019.
- [9] Taufiq. Muhammad, Vita Amalia. Andin and Parmin, "**THE DEVELOPMENT OF SCIENCE MOBILE LEARNING WITH CONSERVATION VISION BASED ON ANDROID APP INVENTOR 2**", Unnes Science Education Journal 6 (1), p-ISSN 2252-6617, e-ISSN 2502-6232, 2017.
- [10] MUHSSIN MOHAMMAD. HAVALL, "**DEVELOPING A MOBILE LEARNING APPLICATION TO TEACH NUMBER CONCEPT FOR PRIMARY SCHOOL CHILDREN**", Computer Information Systems NICOSIA, 2017.
- [11] Swarna Latha Gade. Mary, Gajitha, and Deepthi, "**Design and Implementation of Swam Robotics using Flood Fill Algorithm**", International Journal of Robotics and Automation (IJRA), Vol. 6, No. 4, ISSN: 2089-4856, December 2017.