

IMPLEMENTASI ALGORITMA FISHER-YATES SEBAGAI PENGACAK URUTAN SOAL

Muhammad Thoriq S.A.^{1*}, Fetty Tri Anggraeny², Ronggo Alit³

^{1,2,3} Program Studi Informatika, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur

²fettyanggraeny@upnjatim.ac.id

³rongoalit@upnjatim.ac.id

*Corresponding author email: 1634010007@student.upnjatim.ac.id

Abstrak— Kuis merupakan suatu hal yang tidak asing lagi bagi semua orang. Dimana hampir setiap orang dari usia muda hingga tua pasti pernah merasakan menjawab pertanyaan dari sebuah kuis baik itu dalam konteks pendidikan seperti pada saat melakukan ujian atau tes atau hanya sebagai hiburan atau permainan dan bahkan digunakan sebagai tema pada beberapa acara televisi. Untuk menjawab kuis pun dapat dilakukan melalui media apapun baik secara lisan maupun tulisan, dimana untuk tulisan bisa dilakukan melalui media cetak atau elektronik. Namun pada zaman modern ini yang hampir semua hal dilakukan melalui media elektronik, sehingga dalam penelitian ini, dibuatlah sebuah aplikasi kuis yang dapat diakses melalui *smartphone* berbasis Android. Dimana pada aplikasi ini akan diterapkan algoritma *Fisher-Yates Shuffle* sebagai pengacak urutan soal agar pemain tidak mudah bosan. Dalam pengembangannya, penerapan algoritma *Fisher-Yates Shuffle* pada aplikasi menunjukkan bahwa algoritma tersebut dapat mengacak urutan soal setiap memulai permainan baru dan juga hasil pengacakannya yang tidak berganda.

Kata Kunci— Permainan kuis, Android, Fisher-Yates Algorithm.

I. PENDAHULUAN

Kuis merupakan sebuah kegiatan dimana seseorang memberi pertanyaan kepada orang lain untuk menguji pengetahuan atau wawasan [1]. Dalam konteks pendidikan, kuis biasanya digunakan untuk penilaian singkat pada siswa dalam memahami materi yang baru saja diberikan oleh pengajar, dimana seringkali hal tersebut bersifat secara mendadak namun hanya memiliki jumlah soal yang lebih sedikit dengan tingkat kesulitan yang lebih rendah daripada ujian, hal itu biasa disebut sebagai “*pop quiz*” [2]. Kuis juga bisa digunakan sebagai hiburan atau permainan yang biasa disebut “tebak-tebakan”. Pada sebuah kuis, seorang penanya dapat memberikan beberapa pilihan jawaban untuk membantu dalam menjawab pertanyaan maupun dalam bentuk esai. Untuk membuat sebuah kuis terdapat beberapa macam cara, yang pertama adalah secara lisan dimana penanya langsung menanyakan suatu hal dan yang diberi pertanyaan harus langsung menjawab, atau melalui tulisan dimana seseorang menuliskan pertanyaan pada suatu media baik media cetak maupun elektronik lalu penjawab mengisi dengan jawaban yang tepat.

Untuk media elektronik, kuis dapat diterapkan pada sebuah *website* maupun aplikasi pada *smartphone*. Dalam hal ini, pengembang dapat menambahkan beberapa unsur untuk

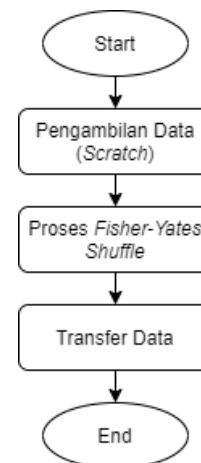
menambah tantangan dalam menjawab setiap pertanyaan salah satunya adalah adalah pengacakan pada urutan soal maupun jawaban. Terdapat berbagai macam algoritma pengacak atau *shuffling algorithm* yang dapat di-implementasikan pada sebuah aplikasi untuk mengacak urutan soal, sehingga munculnya soal yang keluar akan berbeda dan tidak terjadi duplikasi atau pengulangan [3].

Algoritma *Fisher-Yates Shuffle* adalah sebuah algoritma untuk menghasilkan suatu permutasi acak dari suatu himpunan terhingga, dimana hasil dari pengacakan algoritma ini memiliki tingkat probabilitas yang sama [4]. Algoritma *Fisher-Yates* dapat diterapkan untuk mengacak urutan soal setiap *user* memulai sesi permainan baru agar menambah tantangan dalam menjawab setiap pertanyaan yang diberikan. Algoritma *Fisher-Yates Shuffle* juga dipilih karena hasil pengacakannya yang cukup variatif dan tingkat efisiensinya dalam penggunaan memori [5].

Berdasarkan latar belakang tersebut maka penulis akan mencoba membangun aplikasi kuis yang didalamnya akan diterapkan algoritma *Fisher-Yates Shuffle* sebagai pengacak urutan soal.

II. METODOLOGI

Pada sub bab ini akan dijelaskan bagaimana penulis menerapkan algoritma *Fisher-Yates* sebagai pengacak urutan soal. Untuk langkah-langkahnya dapat dilihat pada Gambar 1:



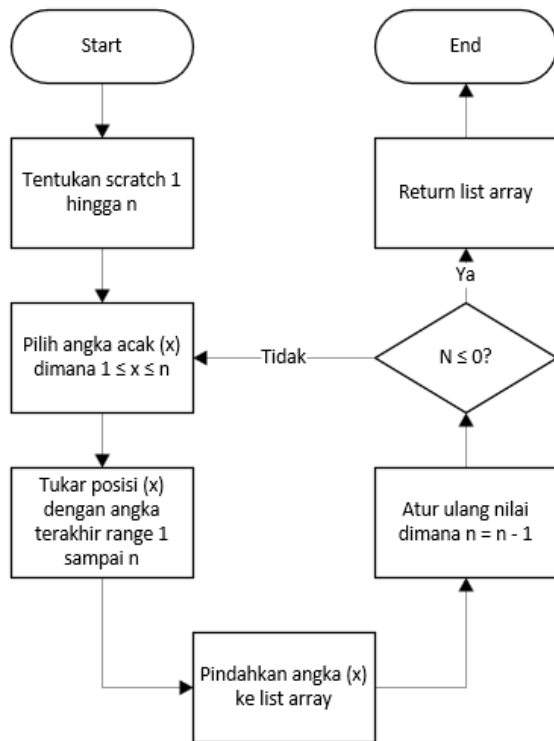
Gbr. 1 Alur Kerja Sistem

A. Pengambilan Data (Scratch)

Scratch merupakan ID pada tabel sebelum diacak. Pada tahap ini, dilakukan *query* pada *database* untuk mengambil ID pada salah satu tabel yang ingin diacak urutan soalnya. Hasil *query* berupa numerik selanjutnya dijadikan parameter untuk proses pengecakan.

B. Proses Fisher-Yates Shuffle

Tahap selanjutnya adalah menerapkan algoritma *Fisher-Yates* untuk mengacak urutan soal. Untuk alur kerja algoritma bisa dilihat pada Gambar 2.



Gbr. 2 Flowchart Algoritma *Fisher-Yates*

1. Tentukan *scratch* yang merupakan ID pada salah satu tabel soal. Asumsikan *scratch* nya adalah $ID_TABEL = [1, 2, 3, 4, 5, 6, 7, 8]$
2. Pilih angka acak (x) diantara $1 \leq x \leq n$ dimana (x) merupakan urutan soal dan (n) merupakan jumlah soal yang tersedia pada tabel.
3. Tukar posisi (x) dengan angka terakhir pada range 1 hingga n
4. Pindahkan angka (x) pada list array yang baru
5. Atur kembali nilai (n) dengan rumus $n = n - 1$
6. Jika $n \leq 0$ maka lanjut ke step berikutnya. Jika $n > 0$ maka kembali ke step 2

7. List array yang terbentuk dari step 4 merupakan array hasil acak untuk menentukan urutan soal berdasarkan ID soal

Tabel 1 dibawah ini merupakan contoh hasil perhitungan yang didapatkan dengan menggunakan algoritma *Fisher-Yates Shuffle* :

TABEL I
HASIL PERHITUNGAN ALGORITMA *FISHER-YATES*

Range ($1 \leq x \leq n$)	Roll (x)	Scratch	Result
		1 2 3 4 5 6 7 8	
1 - 8	4	1 2 3 8 5 6 7	4
1 - 7	3	1 2 7 8 5 6	3 4
1 - 6	1	6 2 7 8 5	1 3 4
1 - 5	2	6 5 7 8	2 1 3 4
1 - 4	4	6 5 7	8 2 1 3 4
1 - 3	1	7 5	6 8 2 1 3 4
1 - 2	1	5	7 6 8 2 1 3 4
			5 7 6 8 2 1 3 4

Setelah mendapatkan hasil berupa ID yang telah diacak maka selanjutnya adalah memasukkan ID tersebut ke dalam *query* yang baru untuk mendapatkan semua data dari tabel yang dipilih, dan selanjutnya adalah mengirim data tersebut ke aplikasi.

C. Transfer Data

Tahap selanjutnya adalah mengirim atau mentransfer data yang sudah teracak ke dalam aplikasi. Dalam hal ini, transfer data dilakukan dengan cara menerapkan konsep *REST web service* dimana aplikasi mengunduh data dari *web service* dalam format JSON lalu menampilkannya pada aplikasi setiap *user* memulai sesi permainan baru.

III. HASIL DAN PEMBAHASAN

Pada bagian ini akan dijelaskan hasil dari penerapan algoritma *Fisher-Yates* dalam mengacak urutan soal. Dimana pengujian dilakukan masing-masing sebanyak 3 kali pada aplikasi. Untuk pengujian hanya bisa diperlihatkan untuk 3 pertanyaan awal saja karena faktor ukuran gambar yang cukup banyak dan memakan ruang, namun sudah cukup membuktikan bahwa pertanyaan berhasil diacak oleh algoritma.

Skor	0	Skor	1	Skor	2
Nyawa	3	Nyawa	3	Nyawa	3
Apakah fungsi dari getch?		Apakah perbedaan dari endl dan ends?		Fungsi yang digunakan untuk membuat string menjadi capital adalah...	
<p>a. Menampilkan karakter tanpa menekan tombol enter</p> <p>b. Membuat program berjalan</p> <p>c. Menahan agar program dapat di compile</p> <p>d. Membaca spasi pada program</p>		<p>a. Pindah baris dan penambah karakter null</p> <p>b. Pindah spasi dan pindah baris</p> <p>c. Pindah data dan tidak pindah program</p> <p>d. Pindah karakter dan penambah spasi</p>		<p>a. Strcpy</p> <p>b. Strupr</p> <p>c. Strcmp</p> <p>d. Strcat</p>	
Pemrograman C++ - Intermediate		Pemrograman C++ - Intermediate BENAR!		Pemrograman C++ - Intermediate BENAR!	

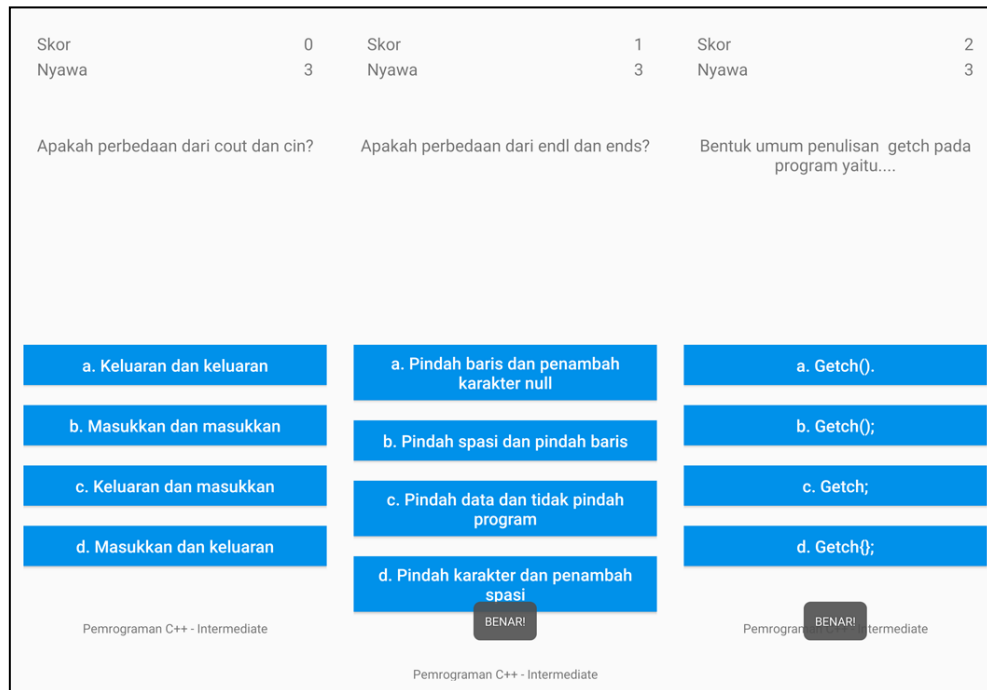
Gbr. 3 Hasil pengujian 1

Skor	0	Skor	1	Skor	2
Nyawa	3	Nyawa	3	Nyawa	3
Penulisan kode program untuk identifier yang benar adalah...		Apakah perbedaan dari cout dan cin?		Fungsi yang digunakan untuk membuat string menjadi capital adalah...	
<p>a. Long 4000;</p> <p>b. Float?2;</p> <p>c. Long break;</p> <p>d. Int a,b,c;</p>		<p>a. Keluaran dan keluaran</p> <p>b. Masukkan dan masukkan</p> <p>c. Keluaran dan masukkan</p> <p>d. Masukkan dan keluaran</p>		<p>a. Strcpy</p> <p>b. Strupr</p> <p>c. Strcmp</p> <p>d. Strcat</p>	
Pemrograman C++ - Intermediate		Pemrograman C++ - Intermediate BENAR!		Pemrograman C++ - Intermediate BENAR!	

Gbr. 4 Hasil pengujian 2

Pada Gambar 3 dan Gambar 4 bisa dibandingkan bahwa urutan 3 pertanyaan awal setiap *user* memulai permainan baru terlihat berbeda, dengan bukti bahwa kedua gambar tersebut dilakukan dalam dua sesi permainan yang berbeda terletak pada skor yang bertambah secara berurutan karena jawaban yang selalu benar.

Lalu Gambar 5 merupakan hasil pengujian aplikasi ke 3 untuk melihat apakah urutan soal masih teracak pada saat *user* memulai sesi permainan baru kembali.



Gbr. 5 Hasil pengujian 3

Hasil pengujian ke 3 menunjukkan bahwa urutan soal juga teracak kembali pada saat *user* memulai sesi permainan baru. Sehingga dengan adanya pengujian ini maka telah membuktikan bahwa algoritma *Fisher-Yates Shuffle* dapat mengacak urutan pertanyaan setiap *user* dapat mengacak urutan pertanyaan setiap *user* melakukan sesi permainan baru.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan uraian dan hasil penelitian yang telah dilakukan tentang implementasi algoritma *Fisher-Yates*, maka dapat disimpulkan mengenai beberapa hal antara lain :

- Penerapan algoritma *Fisher-Yates Shuffle* yang diimplementasikan dalam aplikasi dapat digunakan untuk mengacak urutan pertanyaan sehingga keluarannya soal tidak berulang.
- Penggunaan algoritma *Fisher-Yates Shuffle* dapat diterapkan untuk menentukan solusi pengacakan yang tidak berganda dan variatif

B. Saran

Dalam penelitian yang telah dilakukan, penulis memiliki beberapa saran untuk pengembangan aplikasi yang terkait antara lain :

- Penerapan atau implementasi algoritma *Fisher-Yates* pada aplikasi masih dilakukan melalui *web-service request* atau permintaan terhadap server setiap memulai permainan.
- Diharapkan dengan adanya penelitian ini, peneliti selanjutnya dapat mengembangkan aplikasi serupa dengan lebih optimal dan tingka efisiensi yang lebih tinggi.

REFERENSI

- [1] Collin Dictionary, (2008) Definition of Quiz on Collins Dictionary. [Online], <https://www.collinsdictionary.com/dictionary/english/quiz> , tanggal akses: 20 May 2020.
- [2] Wikipedia, (2008) Quiz on Wikipedia. [Online], <https://en.wikipedia.org/wiki/Quiz> , tanggal akses: 20 May 2020.
- [3] Fadholi, M Aprilian, "Tes Ujian Masuk Peserta Didik Baru Menggunakan Shuffling Algorithm di Madrasah Aliyah Negeri 1 Boyolali" S1. skripsi, Universitas Islam Negeri Sunan Kalijaga, Yogyakarta, Jawa Tengah, 2014.
- [4] Ahmadul, Hadi, "Pengembangan Sistem Informasi Ujian Online Berbasis Web dengan Pengacakan Soal Menggunakan Algoritma Fisher-Yates Shuffle," *Dept. Teknologi Informasi dan Pendidikan*, UNP, ISSN : 2086 - 4981, 2014.
- [5] Ekojono "Penerapan Algoritma Fisher-Yates Pada Pengacakan Soal Game Aritmatika," *SENTIA.*, hal. 95 - 100, 2017.