

IMPLEMENTASI ALGORITMA FISHER-YATES SHUFFLE DALAM PENGACAKAN SOAL PADA APLIKASI KUIS BERBASIS ANDROID

Muhammad Naufal Firdaus*, Adhi Rizal², E. Haodudin Nurkifli³

^{1,2,3,4,5}Universitas Singaperbangsa Karawang; Jl. H.S. Ronggowaluyo, Puseurjaya, Telukjambe Timur, Karawang, Jawa Barat 41361; Telp: (0267) 641177; Fax: (0267) 641367

Keywords:

Fisher-Yates Shuffle,
Aplikasi kuis, Android,
Flutter, Laravel.

Correspondent Email:

naufalfirdaus978@gmail.com



JITET is licensed under
a Creative Commons
Attribution-NonCommercial
4.0 International License

Abstrak. Pengacakan soal dalam aplikasi kuis memainkan peran penting dalam memastikan keadilan dan ketepatan penilaian, khususnya pada pelaksanaan ujian digital. Dalam penelitian ini, algoritma Fisher-Yates Shuffle diadopsi untuk melakukan pengacakan urutan soal secara efisien, sehingga setiap peserta ujian menerima soal dalam susunan yang berbeda. Tujuan dari penelitian ini adalah untuk menerapkan dan mengevaluasi efektivitas algoritma tersebut dalam konteks pengacakan soal pada aplikasi kuis berbasis *mobile*. Proses pengembangan aplikasi mengikuti metode *Mobile Application Development Life Cycle* (MADLC), dengan pemrosesan data soal yang bersumber dari backend berbasis Laravel. Hasil pengujian memperlihatkan bahwa algoritma Fisher-Yates Shuffle mampu menghasilkan pengacakan yang optimal tanpa mengakibatkan soal yang terduplikasi atau terlewat. Implementasi ini mendukung terciptanya pengalaman ujian yang lebih adil dan bervariasi bagi para peserta.

Abstract. Question randomization in quiz applications plays an important role in ensuring the fairness and accuracy of assessments, especially in digital exams. In this study, the Fisher-Yates Shuffle algorithm is adopted to efficiently randomize the question order, so that each examinee receives questions in a different order. The purpose of this study is to implement and evaluate the effectiveness of the algorithm in the context of question randomization in a mobile-based quiz application. The application development process follows the Mobile Application Development Life Cycle (MADLC) method, with question data processing sourced from a Laravel-based backend. The test results show that the Fisher-Yates Shuffle algorithm is able to produce optimal randomization without resulting in duplicated or missed questions. This implementation supports the creation of a fairer and more varied exam experience for participants..

1. PENDAHULUAN

Perkembangan teknologi mobile dalam beberapa tahun terakhir telah memengaruhi berbagai aspek kehidupan, termasuk dalam dunia pendidikan. Aplikasi mobile kini menjadi sarana yang efektif untuk menyampaikan

materi pembelajaran, mengelola penilaian, serta memberikan pengalaman interaktif yang menarik bagi pengguna. Penggunaan metode Mobile Application Development Life Cycle (MADLC) dalam pengembangan aplikasi mobile mampu menghasilkan aplikasi yang

terstruktur dan efisien dalam memenuhi kebutuhan pengguna, khususnya di bidang edukasi.[1]

Salah satu fitur yang sering dimanfaatkan dalam aplikasi edukasi adalah kuis atau tes, yang bertujuan untuk mengukur pemahaman siswa terhadap materi yang diajarkan. Untuk menciptakan pengalaman pengguna yang menarik dan mencegah kebosanan, pengacakan soal menjadi elemen penting. Penerapan algoritma Fisher-Yates Shuffle dalam aplikasi kuis budaya Betawi dapat menyajikan urutan soal yang berbeda-beda di setiap percobaan, memberikan keunikan dalam pengalaman belajar pengguna. [2]

Pengacakan soal memiliki peran strategis dalam sistem kuis untuk memastikan bahwa soal yang disajikan kepada peserta tidak monoton atau mudah ditebak. Penggunaan soal kuis dalam media pembelajaran dapat menumbuhkan minat belajar siswa apabila soal-soalnya tersaji secara bervariasi [3]. Algoritma Fisher-Yates Shuffle dapat membantu menghadirkan variasi soal yang acak secara menyeluruh, tanpa mengulang soal yang sama dalam satu sesi ujian [4].

Selain aspek teknis pengacakan, kemudahan dalam mengembangkan aplikasi kuis juga menjadi perhatian. Platform Kodular untuk merancang aplikasi kuis pemrograman berbasis Android yang mampu menghadirkan kuis interaktif tanpa harus menulis kode secara kompleks [5]. Integrasi aplikasi manajemen kuis dalam sistem pembelajaran berbasis web memungkinkan guru memberikan latihan soal yang bervariasi dan mudah diakses [6].

Dengan latar belakang tersebut, penelitian ini bertujuan untuk mengimplementasikan algoritma Fisher-Yates Shuffle dalam pengacakan soal pada aplikasi kuis berbasis mobile. Tujuannya adalah untuk meningkatkan variasi soal yang diberikan kepada peserta secara efisien dan merata, serta menghindari pola soal yang dapat diprediksi. Penggunaan algoritma ini juga diharapkan dapat menjaga performa aplikasi tetap responsif meskipun harus mengacak sejumlah besar soal dalam waktu yang singkat.

2. TINJAUAN PUSTAKA

2.1. Algoritma Fisher Yates Shuffle

Algoritma Fisher-Yates Shuffle adalah metode pengacakan elemen dalam sebuah array

yang bekerja dengan menukar setiap elemen dengan elemen lain secara acak, mulai dari indeks terakhir hingga indeks pertama. Proses ini menghasilkan susunan elemen yang benar-benar acak, di mana setiap kemungkinan urutan memiliki peluang yang sama untuk muncul. Dalam aplikasi kuis online, algoritma ini digunakan untuk mengacak urutan soal agar setiap peserta mendapatkan soal dalam susunan yang berbeda, sehingga dapat meningkatkan keadilan dan mengurangi kemungkinan kecurangan [7].

2.2. Aplikasi

Aplikasi merupakan perangkat lunak yang dirancang untuk membantu pengguna dalam menyelesaikan tugas tertentu secara efisien. Dalam konteks pengembangan sistem informasi, aplikasi berperan sebagai sarana interaktif antara pengguna dan sistem yang telah dirancang agar proses layanan menjadi lebih mudah, cepat, dan terstruktur. pengembangan aplikasi berbasis Android memungkinkan pengguna untuk mengakses layanan secara fleksibel dan real-time [8].

2.3. Android

Android merupakan sistem operasi mobile berbasis Linux yang dikembangkan oleh Google, dirancang khusus untuk perangkat seperti smartphone dan tablet. Sebagai platform open-source, Android memberikan keleluasaan bagi para pengembang untuk membuat serta memodifikasi aplikasi menggunakan bahasa pemrograman seperti Java dan Kotlin. Sistem ini mendukung berbagai fitur canggih, termasuk kemampuan multitasking, notifikasi yang interaktif, serta kompatibilitas yang luas dengan beragam perangkat keras, Android menjadi salah satu sistem operasi mobile terpopuler dan terus berkembang dalam industri teknologi [9].

2.4. Flutter

Flutter merupakan framework open-source besutan Google yang digunakan untuk mengembangkan aplikasi lintas platform hanya dengan satu basis kode, yang dapat dijalankan di berbagai platform seperti Android, iOS, web, dan desktop. Dengan menggunakan bahasa pemrograman Dart, Flutter menawarkan keunggulan dalam hal efisiensi pengembangan, kinerja yang optimal, serta kemudahan dalam merancang antarmuka pengguna yang responsif dan menarik. Arsitekturnya terdiri dari beberapa komponen utama, seperti Flutter Engine untuk rendering UI, sistem widget yang

fleksibel, serta fitur Hot Reload yang memungkinkan pengembang melihat perubahan kode secara langsung tanpa recompile penuh [10].

2.5. Laravel

Laravel merupakan framework PHP yang mengadopsi arsitektur Model-View-Controller (MVC), dirancang untuk menyederhanakan proses pembuatan aplikasi web melalui sintaks yang jelas dan elegan. Framework ini mendukung pengembangan aplikasi yang rapi dan efisien, serta menyediakan berbagai fitur seperti sistem routing, migrasi basis data, dan Object-Relational Mapping (ORM) melalui Eloquent yang memudahkan pengelolaan database. Laravel dapat dipilih karena kemampuannya dalam mempercepat proses pengembangan sistem informasi, meningkatkan keamanan aplikasi, serta mempermudah pemeliharaan kode berkat arsitektur modular yang dimilikinya [11].

2.6. Database

Database merupakan kumpulan data yang terorganisir dan saling berelasi, yang dikelola oleh sistem manajemen basis data (DBMS) untuk mempermudah penyimpanan, pengolahan, dan pengambilan data secara efisien [12].

3. METODE PENELITIAN

Perancangan aplikasi ini menggunakan metode Mobile Application Development Life Cycle (MADLC), yang mencakup lima tahap utama, yaitu: identifikasi (*identification*), perancangan (*design*), pengembangan (*development*), pembuatan prototipe (*prototyping*), dan pengujian (*testing*).

3.1. Identification

Tahap awal pengembangan aplikasi adalah *identification*, yang bertujuan mengidentifikasi kebutuhan pengguna dan merumuskan tujuan pengembangan secara jelas. Diskusi dilakukan dengan siswa dan guru untuk memahami kendala dalam pelaksanaan kuis digital, seperti kurangnya variasi soal dan ketidakteraturan dalam penyajian soal kepada siswa. Studi literatur dan benchmarking terhadap aplikasi kuis digital juga dilakukan untuk menentukan fitur yang relevan dan menciptakan keunikan aplikasi. Hasil dari tahap ini berupa daftar kebutuhan fungsional, seperti pengacakan soal menggunakan algoritma Fisher-Yates.

3.2. Design

Tahap selanjutnya adalah *design*, yang bertujuan merancang antarmuka aplikasi. Pada tahap ini, dibuat mockup interaktif untuk memvisualisasikan tata letak, navigasi, dan alur kerja aplikasi. Desain mencakup elemen seperti halaman login, dashboard, tampilan kuis, dan tampilan nilai. Proses perancangan dilakukan secara iteratif berdasarkan masukan dari pengembang dan calon pengguna, yaitu guru dan siswa. Hasil akhir tahap ini berupa prototipe antarmuka yang menjadi acuan dalam tahap pengembangan.

3.3. Development

Tahap *development* mencakup pengkodean aplikasi berdasarkan prototipe yang telah dirancang. Flutter digunakan untuk membangun antarmuka pada perangkat Android, sementara *back-end* dikembangkan dengan Laravel dan database MySQL. Fitur utama yang diimplementasikan meliputi autentikasi pengguna, pengelolaan kuis dengan pengacakan soal menggunakan algoritma Fisher-Yates Shuffle serta pengelolaan nilai.

3.4. Prototyping

Setelah pengembangan, aplikasi diuji pada tahap *prototyping* oleh guru dan siswa. Umpan balik dari pengguna awal digunakan untuk menyempurnakan antarmuka dan fungsi aplikasi. Jika belum sesuai kebutuhan, aplikasi dikembalikan ke tahap pengembangan untuk diperbaiki. Proses ini dilakukan secara iteratif hingga aplikasi siap diuji lebih luas.

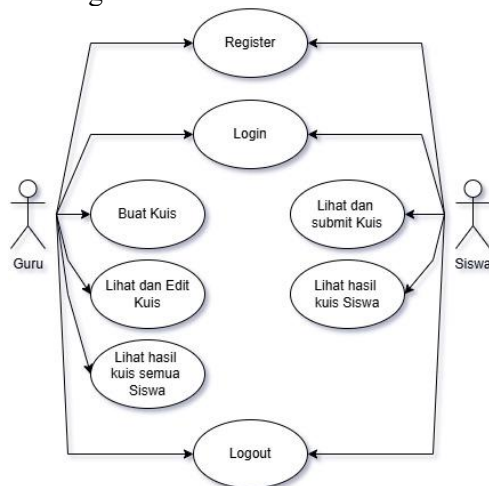
3.5. Testing

Tahap terakhir adalah *testing*, yang dilakukan dengan melibatkan sejumlah siswa untuk langsung menguji aplikasi kuis yang telah dikembangkan. Dalam pengujian ini, setiap siswa menggunakan aplikasi secara langsung untuk mengerjakan kuis dengan soal-soal yang telah diacak menggunakan algoritma Fisher-Yates Shuffle. Hasil pengacakan soal dari setiap siswa kemudian direkapitulasi ke dalam bentuk tabel untuk dianalisis. Pendekatan ini memungkinkan pengembang mengevaluasi efektivitas fitur pengacakan soal secara nyata serta menilai apakah distribusi soal telah berjalan dengan baik dan sesuai dengan prinsip keacakan yang diharapkan.

4. HASIL DAN PEMBAHASAN

4.1. Hasil Identifikasi

Hasil diskusi dengan beberapa guru dan siswa menghasilkan dua kebutuhan utama dalam pengembangan aplikasi kuis. Pertama, diperlukan sistem pengacakan soal yang memastikan setiap siswa menerima urutan soal yang berbeda tanpa adanya duplikasi nomor soal dalam satu sesi. Kedua, dibutuhkan sistem bank soal yang dirancang untuk meminimalisir kemungkinan soal yang sama muncul pada setiap siswa, sehingga mendorong keadilan dan keunikan dalam proses evaluasi. Berdasarkan hasil tersebut, kemudian dibuat *use case diagram* untuk menggambarkan interaksi antara pengguna dan sistem dalam aplikasi yang akan dikembangkan.



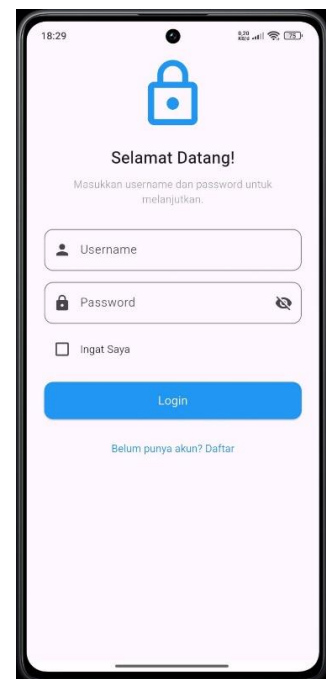
Gambar 1 Use Case Aplikasi

Use case diagram pada gambar 1, menggambarkan interaksi antara dua pengguna utama dalam aplikasi kuis, yaitu guru dan siswa. Baik guru maupun siswa memiliki akses untuk melakukan *register* dan *login* ke dalam sistem. Setelah *login*, guru dapat membuat kuis, melihat dan mengedit kuis yang telah dibuat, serta melihat hasil kuis dari semua siswa. Sementara itu, siswa dapat melihat dan mengerjakan kuis yang tersedia, serta melihat hasil kuis yang telah mereka kerjakan. Seluruh pengguna juga memiliki opsi untuk *logout* setelah selesai menggunakan aplikasi.

4.2. Desain Aplikasi

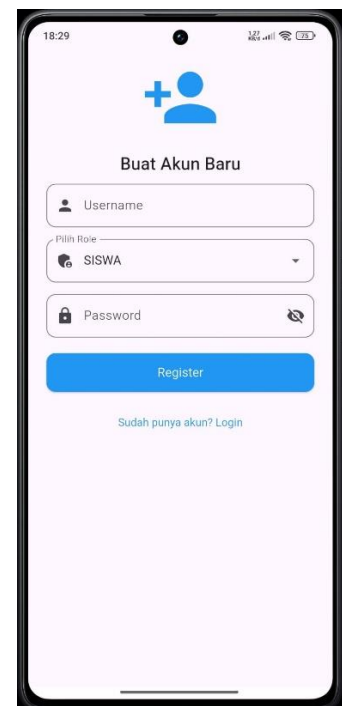
Berikut adalah tampilan *user interface* dari aplikasi kuis terbagi menjadi 3 halaman sebagai berikut;

4.2.1. Halaman Awal

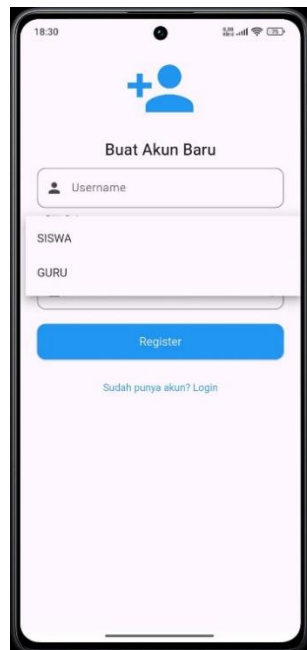


Gambar 2 Login Page

Pada gambar 2, pengguna baik siswa maupun guru dapat melakukan *login* dengan mencantumkan *username* dan juga *password* agar dapat masuk ke halaman utama.



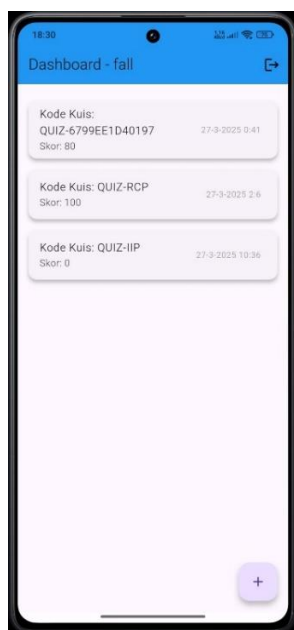
Gambar 3 Register Page



Gambar 4 Role Register Page

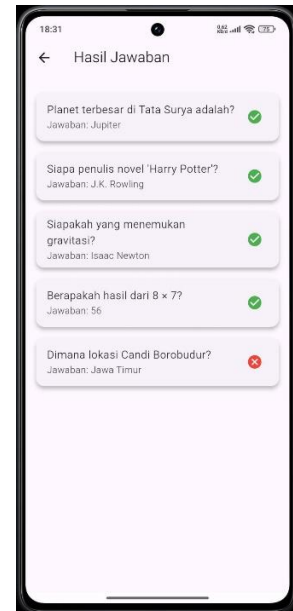
Jika pengguna belum memiliki akun, maka dapat masuk kehalaman *Register* yang tampilannya dapat dilihat pada gambar 3, dihalaman ini pengguna dapat menginputkan *username* dan juga *password* serta *role* (posisi) nya baik itu sebagai siswa atau sebagai guru. Untuk tampilannya dapat dilihat pada gambar 4.

4.2.2. Halaman Utama (Siswa)



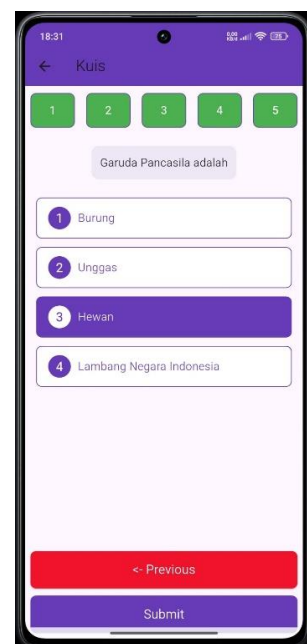
Gambar 5 Dashboard Siswa

Pada gambar 5, adalah tampilan paling awal setelah melakukan *login*. Halaman ini menampilkan Riwayat kuis yang telah dikerjakan oleh siswa.



Gambar 6 Answer Page

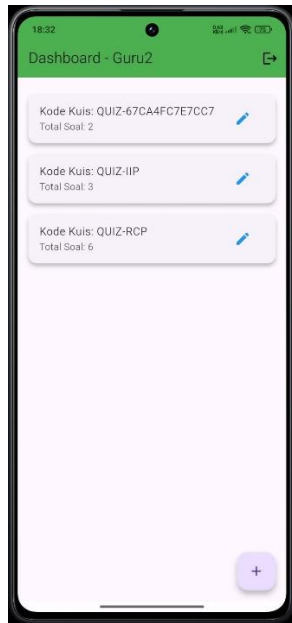
Pada gambar 6 menampilkan halaman hasil jawaban yang menampilkan nilai serta jawaban yang telah dipilih oleh setiap siswa dan juga terdapat keterangan benar atau salahnya.



Gambar 7 Quiz Page

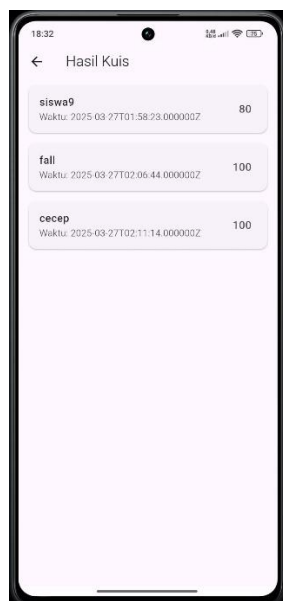
Pada gambar 7 menampilkan halaman kuis. Pada halaman ini siswa dapat mengerjakan dan mengumpulkan kuis yang telah dibuat oleh guru.

4.2.3. Halaman Utama(Guru)



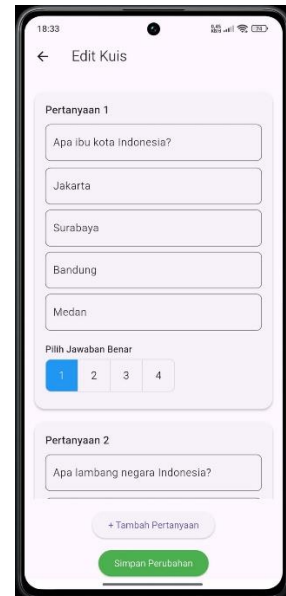
Gambar 8 Dashboard Guru

Gambar 8 adalah tampilan paling awal setelah guru melakukan *login*. Halaman ini menampilkan daftar kuis yang telah dibuat serta button yang berfungsi untuk membuat dan mengedit kuis.



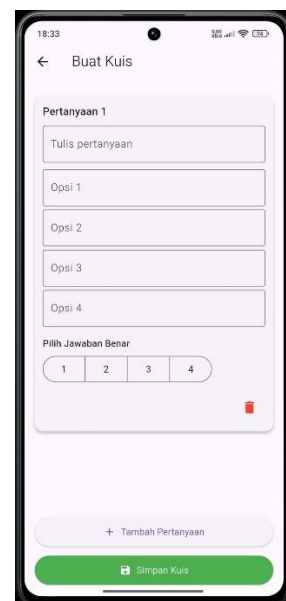
Gambar 9 Quiz Result Page

Pada gambar 9 menampilkan halaman Hasil kuis, dihalaman ini guru dapat melihat nilai serta jawaban dari setiap siswa.



Gambar 10 Edit Quiz Page

Pada gambar 10, menampilkan halaman Edit kuis, dihalaman ini guru dapat mengedit soal dari kuis yang telah dibuat, seperti mengubah soal, mengubah pilihan jawaban serta mengubah jawaban yang benar.



Gambar 11 Create Quiz Page

Pada gambar 11, menampilkan halaman Buat kuis. Pada halaman ini guru dapat menuliskan soal, pilihan jawaban serta jawaban yang benar.

4.3. Hasil Pengujian Aplikasi

Pengujian algoritma Fisher-Yates Shuffle pada aplikasi kuis dilakukan dengan cara mengacak soal yang terdapat dalam bank soal sebanyak 20 butir, untuk didistribusikan kepada 15 siswa. Setiap siswa mendapatkan lima soal yang berbeda secara acak. Tabel 1 berikut ini menampilkan hasil pengacakan yang diperoleh oleh masing-masing siswa berdasarkan ID mereka.

Tabel 1 Hasil Pengacakan Algoritma Fisher Yates Shuffle

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

Tabel 1 di atas menunjukkan hasil implementasi dari algoritma Fisher-Yates Shuffle dalam mengacak soal pada aplikasi kuis. Setiap baris menunjukkan kombinasi soal unik yang diberikan kepada masing-masing siswa. Dari hasil tersebut, terlihat bahwa tidak ada dua siswa yang memperoleh urutan soal yang identik. Hal ini menegaskan bahwa proses pengacakan berjalan secara efektif dan sesuai dengan tujuan algoritma, yaitu menghasilkan permutasi yang acak dan merata.

Distribusi soal dalam tabel menunjukkan bahwa dari total 20 soal yang tersedia dalam bank soal, masing-masing soal muncul dengan frekuensi dan urutan yang bervariasi pada setiap

siswa. Beberapa soal, seperti nomor 4, 13, dan 15, tampak muncul lebih sering dibanding soal lainnya. Misalnya, soal nomor 15 muncul pada siswa dengan ID 2, 3, dan 10, sedangkan soal nomor 13 terlihat pada siswa dengan ID 1, 4, dan 15. Di sisi lain, ada pula soal-soal yang hanya muncul sekali atau dua kali, seperti soal nomor 6 dan 20, yang menunjukkan penyebaran acak tanpa pola yang berulang. Meskipun tidak terjadi distribusi yang sepenuhnya merata, hal ini justru mencerminkan karakteristik algoritma Fisher-Yates Shuffle yang mengedepankan keacakan alami. Pengacakan semacam ini sangat penting dalam konteks kuis berbasis digital, karena mampu mengurangi potensi kecurangan dengan menyajikan variasi soal yang unik bagi tiap siswa tanpa mengandalkan pola tetap.

Penting juga dicatat bahwa dalam konteks aplikasi kuis berbasis mobile, keunikan kombinasi soal pada tiap siswa menjadi salah satu cara untuk meningkatkan keamanan dan integritas pelaksanaan kuis. Dengan tidak adanya kemiripan kombinasi antar siswa, potensi kecurangan melalui penyalinan jawaban menjadi sangat minim. Selain itu, pengacakan yang adil juga mendorong siswa untuk lebih fokus pada pemahaman materi secara menyeluruh, bukan pada hafalan soal.

Secara keseluruhan, penerapan algoritma Fisher-Yates Shuffle terbukti efektif dalam menciptakan pengalaman kuis yang lebih adil, acak, dan efisien. Sistem pengacakan soal ini bukan hanya mampu menghasilkan kombinasi soal yang unik bagi setiap siswa, tetapi juga meningkatkan keandalan dan keamanan aplikasi kuis yang dibangun.

5. KESIMPULAN

Berdasarkan hasil implementasi dan analisis terhadap pengacakan soal menggunakan algoritma Fisher-Yates Shuffle dalam aplikasi kuis, dapat disimpulkan bahwa algoritma ini mampu menghasilkan distribusi soal yang acak dan unik untuk setiap siswa. Dari data yang diperoleh, tidak ditemukan kombinasi soal yang sama antar siswa, yang menunjukkan keberhasilan algoritma dalam mencegah pola dan duplikasi soal. Penggunaan algoritma ini juga efektif dalam mengurangi potensi kecurangan selama pelaksanaan kuis, karena setiap siswa menerima urutan soal yang berbeda. Selain itu, performa pengacakan

berjalan dengan cepat dan efisien dalam aplikasi mobile, menjadikan Fisher-Yates Shuffle sebagai solusi yang tepat untuk mendukung sistem evaluasi berbasis digital secara adil, acak, dan aman.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak terkait yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] A. G. S. Muhamad Alda, Elvan Dito Siregar, Arjuna Sitepu, "Perancangan Aplikasi Quiz Pemrograman berbasis Android Menggunakan Web Kodular," vol. 9, no. 1, pp. 276–283, 2025.
- [2] D. Prastyo and H. Zakaria, "Penerapan Algoritma Fisher Yates Shuffle Pada Aplikasi Quiz Game Pengenalan Budaya Betawi Berbasis Android (Studi Kasus : Tempat Kursus Rumah Aljabar)," *Biner J. Ilmu Komput. ,Teknik dan Multimed.*, vol. 1, no. 3, pp. 711–724, 2023, [Online]. Available: <https://journal.mediapublikasi.id/index.php/Biner>
- [3] Y. Prayoga, R. A. Sukmawati, N. Alkaf, B. Saputra, and H. S. Purba, "Rancang Bangun Aplikasi Manajemen Embedding Quiz untuk Suplemen Media Pembelajaran Berbasis Web," vol. 4, pp. 42–51, 2024.
- [4] Irfansyah, Rizki Muliono, and Susilawati, "Rancang Bangun Aplikasi E-Learning Dengan Implementasi Algoritma Fisher Yates Shuffle Dalam Pengacakan Soal Ujian," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 302–307, 2023, doi: 10.36040/jati.v7i1.6296.
- [5] M. Alfian *et al.*, "RANCANG BANGUN E-MARKETPLACE UMKM PASTRY (BAKEHOUSE) BERBASIS MOBILE," vol. 13, no. 1, 2025.
- [6] I. K. Rachmawan and E. I. Sela, "Perancangan Aplikasi Quiz Sebagai Media Pembelajaran Sejarah," *J. Indones. Manaj. Inform. dan Komun.*, vol. 5, no. 1, pp. 1056–1063, 2024, doi: 10.35870/jimik.v5i1.575.
- [7] B. Pramono, "Implementasi Algoritma Fisher-Yates Shuffle Pada Aplikasi Quiz Online Materi Pemrograman Dasar," *AnoaTIK J. Teknol. Inf. dan Komput.*, vol. 2, no. 1, pp. 22–29, 2024, doi: 10.33772/anoatik.v2i1.31.
- [8] T. S. Jaya, P. A. Pratomo, and F. K. Ikhsan, "Pengembangan Aplikasi Mobile Pendeteksi Penyakit Daun Tanaman Jagung dengan Metode Mobile Application Development Life Cycle (MADLC)," vol. 16, no. 1, pp. 252–262, 2024.
- [9] A. Dillah, G. F. Nama, D. Budiyanto, and M. A. Muhammad, "Rancang Bangun Aplikasi Monitoring Operasi P2Tl Pengukuran Tidak Langsung 2 Fasa Di Pt. Pln (Persero) Unit Pelaksana Pelayanan Pelanggan (Up3) Metro," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, 2024, doi: 10.23960/jitet.v12i3.4458.
- [10] A. S. Putri, A. Eviyanti, and H. Hindarto, "Rancang Bangun Aplikasi E-Commerce Berbasis Android Pada Toko Suryamart Menggunakan Framework Flutter," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 3, pp. 257–265, 2023, doi: 10.47233/jteksis.v5i3.851.
- [11] A. Murod, R. Hadiwiyanti, and D. S. Y. Kartika, "Rancang Bangun Sistem Informasi Manajemen Persediaan Barang Menggunakan Framework Laravel (Studi Kasus : Pt. Jazeera Inti Sukses)," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, pp. 2210–2219, 2024, doi: 10.23960/jitet.v12i3.4706.
- [12] S. Assani, R. Hurriyah, M. Machmud, T. Rahman, A. R. Al Haidar, and A. F. Mahera, "Sistem Informasi Dan Penerimaan Siswa Baru Berbasis Web Dengan Menggunakan Framework Laravel," *J. Inform. dan Tek. Elektro Terap.*, vol. 6, no. 2, pp. 145–152, 2024, doi: 10.51401/jinteks.v6i2.4004.