

PERANCANGAN APLIKASI SIMULASI UNBK DENGAN METODE PENGACAK LINEAR CONGRUENT METHOD BERBASIS WEB

Fiqri Azmi Fachir^{*1}, Ilyas Nuryasin², Yufis Azhar³

^{1,2,3}Universitas Muhammadiyah Malang

e-mail: fiqri.fachir12@webmail.umm.ac.id^{*1}, ilyas@umm.ac.id², yufis@umm.ac.id³

Abstrak

Keputusan Menteri pendidikan dan kebudayaan (*kemendikbud*) yang menyatakan untuk seluruh SMP, SMA, dan SMK sederajat harus menerapkan ujian nasional dan ujian sekolah berbasis komputer (UNBK). Beberapa sekolah telah bersiap untuk menghadapi UNBK dengan melatih siswa-siswi melaksanakan ulangan harian ataupun ujian *tryout* dengan menggunakan komputer sekolah [1]. Aplikasi simulasi untuk mempersiapkan UNBK banyak beredar dan banyak dipakai untuk melatih siswa-siswi untuk dalam menghadapi UNBK. Pseudo Random Number Generator (PRNG) merupakan algoritma yang menggunakan rumus matematis untuk menghasilkan deret angka secara acak. Salah satu metode yang menggunakan algoritma PRNG adalah Linear Congruent Method (LCM). Linear Congruent Method merupakan salah satu metode pembangkit bilangan acak yang memanfaatkan model linear untuk membangkitkan bilangan acak [2]. Linear Congruent Method (LCM) berhasil diterapkan kedalam aplikasi simulasi pengacak soal yang telah dibuat. Hasil dari pengujian sistem aplikasi yang telah dibuat yaitu pada pengujian algoritma pengacak soal, soal berhasil teracak dan untuk setiap siswa berhasil mendapatkan urutan soal yang berbeda-beda.

Kata kunci: UNBK, LCM, PRNG, Ujian Nasional.

Abstract

Decree of the Minister of Education and Culture (*kemendikbud*) which states that for all junior high schools, senior high schools and vocational high schools, they must apply the national examination and computer based school exams (UNBK). Some schools have prepared to face UNBK by training students to carry out daily tests or *tryout* exams using school computers [1]. Simulation applications to prepare UNBK are widely circulated and widely used to train students to deal with UNBK. Pseudo Random Number Generator (PRNG) is an algorithm that uses mathematical formulas to generate random numbers. One method that use the PRNG algorithm is Linear Congruent Method (LCM). Linear Congruent Method is a random number generator method that utilizes linear models to generate random numbers [2]. Linear Congruent Method (LCM) was successfully applied into the randomization simulation application that was created. The results of testing the application system that has been made, namely the testing of the randomization algorithm, the questions were successfully randomized and for each student managed to get a sequence of different questions.

Keywords: UNBK, LCM, PRNG, National Examination.

1. Pendahuluan

Keputusan Menteri pendidikan dan kebudayaan (*kemendikbud*) yang menyatakan untuk seluruh SMP, SMA, dan SMK sederajat harus menerapkan ujian nasional dan ujian sekolah berbasis komputer (UNBK). Beberapa sekolah telah bersiap untuk menghadapi UNBK dengan melatih siswa-siswi melaksanakan ulangan harian ataupun ujian *tryout* dengan menggunakan media komputer sekolah [1]. Aplikasi simulasi untuk mempersiapkan UNBK banyak beredar dan banyak dipakai untuk melatih siswa-siswi untuk menghadapi UNBK. Tujuannya untuk melatih siswa-siswi dalam mempersiapkan menghadapi UNBK namun aplikasi tersebut masih memiliki beberapa kekurangan seperti soal yang dihasilkan walaupun teracak namun masih banyak yang mengalami perulangan soal. Pseudo random number generator (PRNG) merupakan algoritma yang menggunakan rumus matematis untuk menghasilkan deret angka acak [2]. Salah satu metode yang menggunakan algoritma PRNG adalah Linear Congruent Method (LCM). LCM

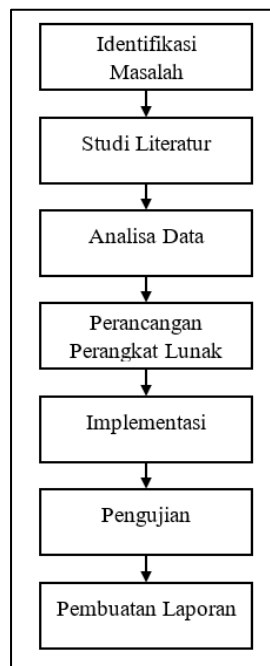
merupakan salah satu metode pembangkit bilangan acak yang memanfaatkan model linear untuk membangkitkan bilangan secara acak. Metode ini biasanya digunakan pada aplikasi yang memerlukan sumber (*seed*) yang terurut dan hasil yang teracak [2].

Salah satu penelitian terkait penggunaan metode LCM dilakukan oleh [2] dengan judul “Penerapan Metode Linear Kongruen dan Algoritma *Vigenere Chiper* pada Aplikasi Sistem Ujian Berbasis LAN” yang menjelaskan tentang penggunaan metode *linear congruent method* (LCM) sebagai pengacak soal ujian. Selain menggunakan metode pengacak soal, pada penelitian ini juga menggunakan algoritma kriptografi *Vigenere Chiper* yang bertujuan untuk mengenkripsi soal-soal yang tersimpan didatabase aplikasi. Hasil pada penelitian ini adalah aplikasi berjalan pada jaringan *Local Area Network* (LAN), soal-soal ujian menggunakan metode pengacak LCM, dan algoritma *Vigenere Chiper* sebagai kriptografi enkripsi soal untuk menghindari akses pada *record*. Selain itu ada dari [3] dengan judul “Sistem Ujian Online Seleksi Penerimaan Mahasiswa Baru dengan Pengacakan Soal menggunakan Linear Congruent Method” dimana pada penelitiannya menjelaskan pengaplikasian metode LCM pada sistem aplikasi untuk ujian seleksi mahasiswa baru. Hasil dari penelitiannya yaitu soal untuk ujian online pada seleksi mahasiswa baru dibuat acak dengan metode LCM dan soal berhasil teracak.

Pada penelitian ini menjelaskan tentang perancangan aplikasi simulasi unbk dengan metode pengacak *linear congruent method* dengan berbasis web. Penelitian ini bertujuan untuk menerapkan algoritma *pseudo random number generator* dengan metode pengacak LCM kedalam aplikasi simulasi UNBK untuk mengacak paket soal beserta jawaban untuk setiap siswa.

2. Metode Penelitian

Metode penelitian yang digunakan dalam penelitian ini yaitu sesuai dengan gambar 1 :



Gambar 1. Kerangka Metode Penelitian

2.1. Identifikasi Masalah

Melakukan pengidentifikasian permasalahan berdasarkan latar belakang. Pada sub bab ini peneliti melakukan analisa permasalahan, analisa metode, dan analisa sistem yang akan dibuat. Berdasarkan latar belakang maka dirumuskan analisa permasalahan yaitu simulasi pengacakan soal merupakan proses mengacak soal-soal dari paket soal dengan menggunakan metode pengacak.

Pengacak memiliki kata dasar “acak” yang menurut kamus besar bahasa indonesia (KBBI) artinya “tanpa pola” atau penggambaran suatu pemilihan yang tidak dibatasi atau apabila dibatasi harus diwujudkan dengan menggunakan kata peluang seperti kata nomina (kata benda) atau kata statistik [4]. Simulasi sendiri merupakan suatu teknik meniru operasi atau proses yang terjadi dalam suatu sistem [5]. Menurut KBBI, simulasi merupakan metode pelatihan yang

meragakan sesuatu dalam bentuk tiruan yang mirip dengan keadaan sesungguhnya. Selain itu bisa diartikan sebagai penggambaran suatu sistem atau proses dengan peragaan berupa model statistik atau pemeranan [6]. Metode yang digunakan adalah *linear congruent method* yang merupakan salah satu metode yang memakai algoritma pengacak deret angka *pseudo random number generator* (PRNG) [2].

Pada tahap analisa metode diketahui bahwa metode *linear congruent method* merupakan salah satu metode pembangkit bilangan acak yang memanfaatkan model linear untuk membangkitkan bilangan acak. Pada tahap analisa sistem dilakukan penganalisaan kebutuhan sistem berdasarkan latarbelakang dan analisa metode yang diketahui [5].

Pada tahap analisa kebutuhan sistem, analisa kebutuhan sistem merupakan proses meninjau sistem yang ada untuk memperjelas tujuan dari pembuatan perangkat lunak. Pada tahap ini memaparkan bagaimana karakteristik pengguna dari aplikasi yang akan dibuat [7]. Calon pengguna nantinya adalah para siswa dan guru SMP. Guru SMP bertindak sebagai admin untuk memasukkan soal dan mengelola sistem aplikasi. Sedangkan siswa bertindak sebagai pengguna aplikasi simulasi.

2.2. Studi Literatur

Melakukan studi literatur terhadap berbagai referensi dari buku, artikel, jurnal, dan skripsi serta penelitian yang berkaitan dengan pembuatan aplikasi simulasi dan metode pengacak. Berdasarkan studi literatur yang didapat, diketahui bahwa ada beberapa penelitian terkait dengan aplikasi simulasi, unbk, dan metode pengacak.

Beberapa penelitian terkait dengan penggunaan metode *Linear Congruent Method* (LCM) dan aplikasi simulasi UNBK yaitu, [7] dengan judul "Perancangan Aplikasi Game Edukasi Pembelajaran Anak Usia Dini Menggunakan *Linear Congruent Method* Berbasis Android" dimana pada penelitiannya menjelaskan tentang perancangan aplikasi untuk anak usia dini (3-8t ahun) pada *platform* android. Penelitian ini menggunakan metode belajar membaca, menulis dan berhitung dimana metode LCM digunakan untuk mengacak soal dan materi pada aplikasi game edukasi ini.

Penelitian kedua dilakukan oleh [8] dengan judul "Aplikasi Pembelajaran Mengetik Cepat Berbasis Game Edukasi dengan *Linear Congruent Method*" pada penelitiannya menjelaskan tujuan untuk mengedukasi siswa SD dengan game cerdas serta melatih siswa agar dapat meningkatkan kecepatan mengetik dengan membiasakan diri sejak dini. LCM digunakan sebagai pengacak soal yang nantinya soal tersebut harus diketik ulang oleh pengguna dengan cepat. Hasil dari penelitian ini diketahui bahwa siswa SD lebih suka dengan game edukasi berbasis "typer" yang tentunya melatih siswa dalam meningkatkan kecepatan mengetik.

Penelitian ketiga dilakukan oleh [9] dengan judul "Modifikasi Metode *Linear Congruential Generator* untuk Optimalisasi Acak" dimana metode *Linear Congruential Generator* (LCG) dan metode *Linear Congruent Method* (LCM) sama-sama merupakan metode yang menggunakan algoritma pengacak dengan linear kongruen atau biasa disebut dengan *Linear Congruent*. Pada penelitian ini membahas modifikasi dari metode yang menggunakan linear kongruen kedalam model yang lebih untuk mendapatkan hasil acak yang optimal dengan cara membuat model matematis dari rumus dasar linear kongruen. Hasil dari penelitian ini menyebutkan bahwa linear kongruen melakukan pengacakkan dengan melalui 4 tahapan yaitu, penentuan jumlah soal yang diacak ($m / \text{modulus}$), penentuan faktor pengali (a), penentuan faktor penambah (c), dan terakhir perhitungan kombinasi dari a , c , dan m . Dan untuk optimalisasi menggunakan beberapa persamaan, dimana rumus dasar linear kongruen dengan 4 tahapan tersebut, dilakukan sebanyak 2 kali dan kemudian dimasukkan ke dalam beberapa tahapan persamaan, hasil akhirnya linear kongruen yang dimodifikasi menghasilkan pola acak yang lebih rumit.

Berdasarkan studi literatur diketahui bahwa metode LCM kemungkinan terjadinya perulangan setelah pengacakkan lebih kecil dibandingkan dengan fungsi *rand()* dari php. Dan jika dibandingkan dengan metode lain seperti *monte carlo*, LCM lebih simpel dan sederhana. Apabila permasalahan bisa diselesaikan secara matematis, maka metode *monte carlo* tidak diperlukan. Selain itu dalam sisi keamanan, LCM lebih baik daripada fungsi *rand()* karena *seed* pada LCM dilakukan oleh user atau admin, bukan melalui sistem sedangkan pada fungsi *rand()* nilai *seed* ditentukan didalam sistem sehingga untuk memprediksi kemungkinan apa yang akan keluar dari hasil pengacakan akan mudah [2].

Metode LCM menggunakan perhitungan dengan menentukan nilai dari variabel a (faktor pengali), c (faktor penambah), dan m (modulus / jumlah yang akan diacak). Berikut pendefinisian metode LCM [5] :

$$X_i = (a X_n + c) \bmod m \dots \quad (2.1)$$

Dimana:

X_0 = bilangan acak nilai awal.

X_i = bilangan acak ke- i .

X_n = bilangan acak sebelumnya.

a = faktor pengali.

c = faktor penambah (increment).

m = modulus (batas maksimum bilangan acak).

Pendefinisian LCM menjelaskan bahwa X_i merupakan variabel bilangan acak ke- i , a dan c sebagai variabel konstanta LCM dan m sebagai nilai maksimum bilangan acak. Beberapa ketentuan untuk menentukan nilai dari setiap variabel sebagai berikut:

- a. Konstanta a harus lebih besar dari \sqrt{m} dan biasanya dinyatakan dengan syarat :

$$\frac{m}{100} < a < m - \sqrt{m} \quad (2.2)$$

- b. Konstanta c harus berangka ganjil apabila m bernilai pangkat dua dan tidak boleh nilai dari kelipatan m .
- c. Modulo m disarankan merupakan bilangan prima untuk mempermudah perhitungan dikomputer.
- d. Untuk X_0 harus merupakan angka integer [5].

Contoh pembangkit bilangan acak sebanyak 10 kali dengan nilai $a=21$, $c=17$, $m=20$, dan $X_0=0$ maka:

$$X_0 = (21 \cdot 0 + 17) \bmod 20 = 17$$

$$X_1 = (21 \cdot 17 + 17) \bmod 20 = 14$$

$$X_2 = (21 \cdot 14 + 17) \bmod 20 = 11$$

$$X_3 = (21 \cdot 11 + 17) \bmod 20 = 8$$

$$X_4 = (21 \cdot 8 + 17) \bmod 20 = 5$$

$$X_5 = (21 \cdot 5 + 17) \bmod 20 = 2$$

$$X_6 = (21.2 + 17) \bmod 20 = 19$$

$$X_7 = (21.19 + 17) \bmod 20 = 16$$

$$X_8 = (21.16 + 17) \bmod 20 = 13$$

$$X_9 = (21.13 + 17) \bmod 20 = 10$$

Bilangan acak yang dibangkitkan adalah :

17 14 11 8 5 2 19 16 13 10

Bilangan acak yang tidak dibangkitkan adalah :

7 4 1 18 15 12 9 6 3 0

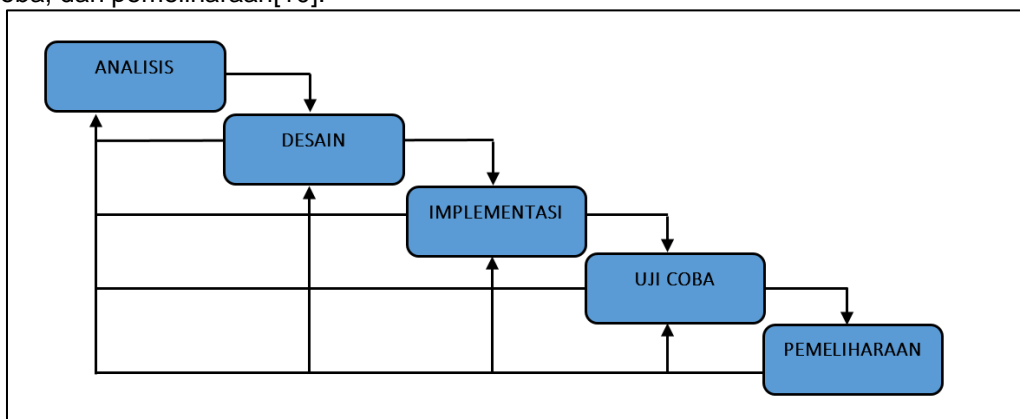
Dari contoh diatas “ m ” yang ditentukan adalah 20. “ m ” merupakan modulus atau batas jumlah bilangan yang tersedia, artinya ada 20 bilangan yang tersedia. Bilangan yang tersedia adalah 0 s/d 19. Bilangan tersebut diurutkan, kemudian diacak dan dibangkitkan sesuai dengan ketentuan. Dalam contoh diatas, ditentukan bahwa ada 10 bilangan yang dibangkitkan dan 10 sisanya tidak dibangkitkan. Dari pengacakan 20 bilangan tersebut dapat diketahui bahwa pembangkitan bilangan acak tersebut tidak mengalami perulangan.

2.3. Analisa Data

Melakukan analisa data yang dibutuhkan dalam penelitian ini. Data yang dibutuhkan untuk pengujian aplikasi simulasi adalah soal-soal *tryout*, UN tahun lalu, dan berbagai macam referensi soal. Data soal yang diambil merupakan soal-soal untuk SMP sederajat. Sesuai dengan UN tingkat SMP maka diambil 4 mata pelajaran utama yang akan digunakan sebagai data soal yaitu matematika, bahasa indonesia, bahasa inggris, dan ilmu pengetahuan alam. Jumlah soal dari masing-masing pelajaran berjumlah 100 soal dan total seluruh soal adalah 400 soal.

2.4. Perancangan Perangkat Lunak

Pada tahap ini dilakukan perancangan model untuk perangkat lunak. Model yang digunakan adalah model *waterfall*. Model waterfall memiliki tahapan-tahapan yang berurutan dan sistematis. Dalam model waterfall ada beberapa tahapan yaitu, analisis, desain, implementasi, uji coba, dan pemeliharaan[10].



Gambar 2. Model Waterfall [10].

Pada gambar 2 menjelaskan tahapan-tahapan model waterfall secara berurutan. Tahapan pertama analisis, pada tahap ini dilakukan penelitian kebutuhan sistem dan pencarian kebutuhan yang difokuskan pada pengerjaan *software*. Tahapan ini dilakukan agar *software engineer* mengerti perihal domain informasi *software*, contohnya seperti fungsi-fungsi dan fitur yang dibutuhkan serta tampilan antarmuka yang sesuai dengan permintaan pelanggan.

Tahap kedua adalah desain, yaitu dilakukan perancangan dari analisa kebutuhan kebentuk rangka kerja (arsitektur). Desain harus mengimplementasikan kebutuhan yang dibuat

pada tahapan analisis kedalam bentuk arsitektur diagram misalnya *usecase*, *activity diagram*, *flowchart*, dll.

Tahap ketiga adalah implementasi, yaitu melakukan proses penerapan dari desain kedalam bahasa pemrograman. Hasil dari tahap ini telah berbentuk sistem aplikasi *prototype* sehingga belum tentu sepenuhnya sempurna.

Tahapan keempat adalah uji coba, pada tahap ini dilakukan pengujian pada sistem yang telah berbentuk *prototype* untuk mengetahui sisi kekurangan yang perlu diperbaiki. Hal ini dilakukan untuk menghindari terjadinya *error*.

Tahapan kelima adalah pemeliharaan, yaitu melakukan perbaikan apabila pada tahap uji coba ditemukan kekurangan atau *error* pada sistem yang dibuat. Apabila tidak ditemukan *error*, tahapan ini tetap perlu dilakukan untuk pengembangan sistem agar lebih baik [10].

2.5. Implementasi

Melakukan implementasi sistem terkait *hardware* dan *software*. Pada tahap ini dilakukan coding dari desain diagram yang dibuat kedalam bentuk sistem aplikasi.

2.5.1. Implementasi Perangkat Keras (*Hardware*)

Pada implementasi perangkat keras (*Hardware*) menjelaskan spesifikasi perangkat keras yang digunakan dalam pembuatan hingga pengujian sistem. Berikut spesifikasi perangkat keras yang digunakan:

- a. Nama dan tipe perangkat keras : Laptop ASUS X450JN
- b. *Processor* : Intel(R) Core(TM) i7-4710HQ @ 2.50GHz
- c. Tipe Sistem : 64-Bit Operating System
- d. Memori : 12 GB

2.5.2. Implementasi Perangkat Lunak (*Software*)

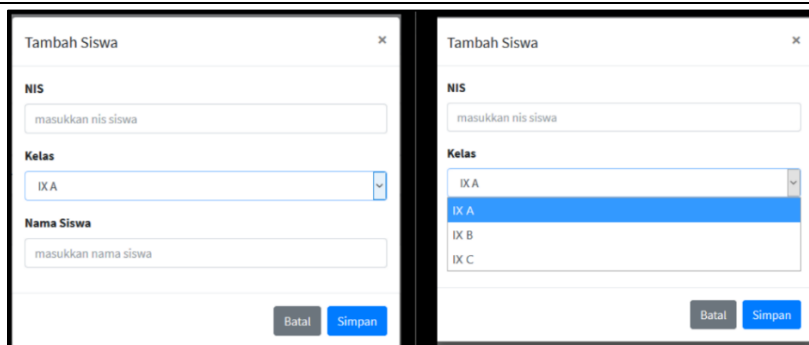
Pada implementasi perangkat lunak menjelaskan spesifikasi perangkat lunak yang digunakan dalam pembuatan hingga pengujian sistem. Berikut spesifikasi perangkat lunak yang digunakan:

- a. Sistem Operasi Windows 10 Pro 64-Bit (10.0 Build 10240)
- b. Apache Web Server v.2.4.37
- c. PHP v.7.2.12
- d. PhpMyAdmin v.4.8.3
- e. Mysql v.10.1.37
- f. HTML 5

2.6. Pengujian

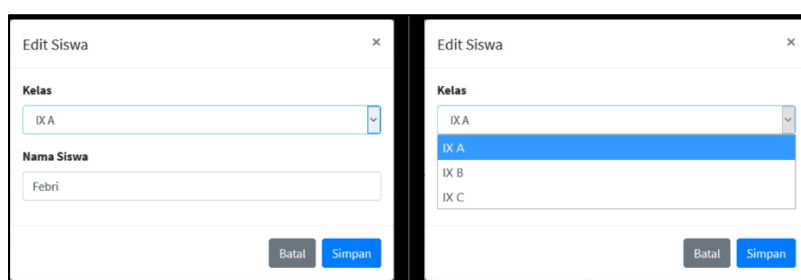
Pada bagian pengujian menjelaskan hasil pengujian sistem yang dibuat. Pengujian terbagi menjadi dua bagian, pertama yaitu pengujian sistem dan fitur. Kedua pengujian algoritma pengacak soal. Pengujian sistem dan fitur meliputi pengujian aplikasi apa telah sesuai dengan yang diharapkan, tampilan yang responsif, dan fitur yang mudah digunakan. Pada pengujian algoritma, pengujian ini dilakukan untuk mengukur keakuratan algoritma pengacak soal. Algoritma pengacak yang digunakan adalah *pseudo random number generator* dengan metode *linear congruent method*. Algoritma pada aplikasi ini digunakan untuk mengacak soal ujian, selain itu algoritma ini diatur agar setiap siswa (user) mendapatkan hasil acak yang berbeda-beda.

2.6.1. Pengujian Sistem dan Fitur



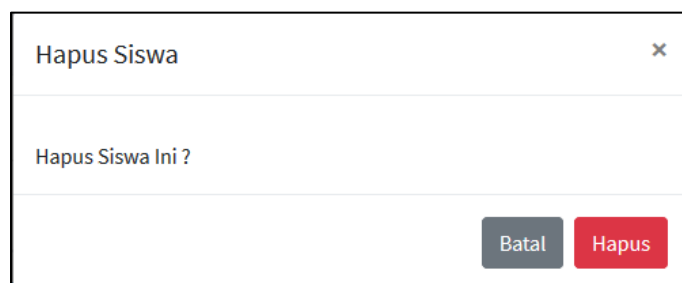
Gambar 3. Tampilan Fitur Fungsi Tambah

Pada gambar 3 menampilkan fitur fungsi tambah. Fitur ini digunakan untuk menambahkan data kedalam database aplikasi. Fitur fungsi tambah tersedia pada halaman, data siswa, bank soal, data kelas, data mata pelajaran, dan data ujian.



Gambar 4. Tampilan Fitur Fungsi Edit.

Pada gambar 4 menampilkan fitur fungsi edit. Fitur ini digunakan untuk mengubah data yang telah tersimpan di database aplikasi untuk diperbaharui. Fitur fungsi edit tersedia pada halaman, data siswa, data kelas, data mata pelajaran, data ujian dan setting algoritma.



Gambar 5. Tampilan Fitur Fungsi Hapus

Pada gambar 5 menampilkan fitur fungsi hapus. Fitur ini digunakan untuk menghapus data dari database aplikasi. Fitur fungsi hapus tersedia pada halaman, data siswa, data kelas, data mata pelajaran, data bank soal, data ujian, dan data hasil ujian.

2.6.2. Pengujian Algoritma

a. Setting algoritma

Gambar 6. Tampilan Setting algoritma

Pada gambar 6 menampilkan halaman pengaturan algoritma. Hal yang harus diperhatikan pada algoritma yang menggunakan metode *linear congruent method* adalah pengaturan nilai *value* dari “a, c, dan m”. “a” merupakan faktor pengali terhadap nilai awal [5], dalam aplikasi ini nilai awal diambil dari nis masing-masing siswa. “c” merupakan *increment* yaitu penambahan. Setelah nilai awal dikalikan dengan nilai dari *variable* “a”, maka hasil akan ditambahkan dengan nilai dari *variable* “c”. Sedangkan “m” adalah modulus, dalam aplikasi ini modulus adalah total soal.

b. Daftar siswa

| NIS | Nama Siswa | Option |
|------|------------|------------------------|
| 1028 | Febri | Lihat Hasil Pengacakan |
| 1206 | fiqri | Lihat Hasil Pengacakan |
| 2708 | syech | Lihat Hasil Pengacakan |

Gambar 7. Halaman Daftar Siswa untuk Uji Coba Algoritma.

Pada Gambar 7 menampilkan daftar nama dan nis siswa serta *buttons* untuk melihat hasil pengacakan soal untuk setiap siswa.

c. Algoritma Siswa

| No | Perhitungan | Hasil | Hasil + 1 |
|----|---|-------|-----------|
| 1 | $X(0) = (21 \cdot (28) + 17) \bmod 20 = 5$ | 5 | 6 |
| 2 | $X(1) = (21 \cdot (5) + 17) \bmod 20 = 2$ | 2 | 3 |
| 3 | $X(2) = (21 \cdot (2) + 17) \bmod 20 = 19$ | 19 | 20 |
| 4 | $X(3) = (21 \cdot (19) + 17) \bmod 20 = 16$ | 16 | 17 |
| 5 | $X(4) = (21 \cdot (16) + 17) \bmod 20 = 13$ | 13 | 14 |
| 6 | $X(5) = (21 \cdot (13) + 17) \bmod 20 = 10$ | 10 | 11 |

Gambar 8. Halaman Data Siswa Pertama dengan Algoritma Pengacak.

Pada gambar 8 menampilkan hasil algoritma pengacakan untuk siswa pertama pada daftar siswa. Siswa dengan nama Febri dan Nis 1028 akan mendapatkan hasil pengacakan soal seperti gambar 8.

| soal tampil : 4,1,18,15,12,9,6,3,20,17 soal tidak tampil : 14,11,8,5,2,19,16,13,10,7 | | | | | |
|--|---|-------|-----------|--|--|
| Search: <input type="text"/> | | | | | |
| No | Perhitungan | Hasil | Hasil + 1 | | |
| 1 | $X(0) = (21 \cdot (6) + 17) \bmod 20 = 3$ | 3 | 4 | | |
| 2 | $X(1) = (21 \cdot (3) + 17) \bmod 20 = 0$ | 0 | 1 | | |
| 3 | $X(2) = (21 \cdot (0) + 17) \bmod 20 = 17$ | 17 | 18 | | |
| 4 | $X(3) = (21 \cdot (17) + 17) \bmod 20 = 14$ | 14 | 15 | | |
| 5 | $X(4) = (21 \cdot (14) + 17) \bmod 20 = 11$ | 11 | 12 | | |
| 6 | $X(5) = (21 \cdot (11) + 17) \bmod 20 = 8$ | 8 | 9 | | |

Gambar 9. Halaman Data Siswa Kedua dengan Algoritma Pengacak.

Pada gambar 9 menampilkan hasil algoritma pengacakan untuk siswa kedua pada daftar siswa. Siswa dengan nama Fiqri dan Nis 1206 akan mendapatkan hasil pengacakan soal seperti gambar 9

| soal tampil : 6,3,20,17,14,11,8,5,2,19 soal tidak tampil : 16,13,10,7,4,1,18,15,12,9 | | | | | |
|--|---|-------|-----------|--|--|
| Search: <input type="text"/> | | | | | |
| No | Perhitungan | Hasil | Hasil + 1 | | |
| 1 | $X(0) = (21 \cdot (8) + 17) \bmod 20 = 5$ | 5 | 6 | | |
| 2 | $X(1) = (21 \cdot (5) + 17) \bmod 20 = 2$ | 2 | 3 | | |
| 3 | $X(2) = (21 \cdot (2) + 17) \bmod 20 = 19$ | 19 | 20 | | |
| 4 | $X(3) = (21 \cdot (19) + 17) \bmod 20 = 16$ | 16 | 17 | | |
| 5 | $X(4) = (21 \cdot (16) + 17) \bmod 20 = 13$ | 13 | 14 | | |
| 6 | $X(5) = (21 \cdot (13) + 17) \bmod 20 = 10$ | 10 | 11 | | |

Gambar 10. Halaman Data Siswa Ketiga dengan Algoritma Pengacak.

Pada gambar 10 menampilkan hasil algoritma pengacakan untuk siswa ketiga pada daftar siswa. Siswa dengan nama Syech dan Nis 2708 akan mendapatkan hasil pengacakan soal seperti gambar 10.

2.7. Pembuatan Laporan

Melakukan pembuatan laporan dari penelitian yang telah dikerjakan.

3. Hasil Penelitian dan Pembahasan

Hasil dari seluruh pengujian yaitu sistem aplikasi telah selesai dibuat namun masih ada beberapa kelebihan dan kekurangan. Pada pengujian algoritma pengacak soal, soal berhasil teracak dan untuk setiap siswa berhasil mendapatkan urutan soal yang berbeda-beda. Hal yang paling penting dan perlu diperhatikan yaitu penentuan nilai *variable* "a , c, dan m" karena nilai dari setiap *variable* mempengaruhi dari hasil pengacakan.

4. Kesimpulan dan Saran

Linear Congruent Method (LCM) berhasil diterapkan kedalam aplikasi simulasi pengacak soal yang telah dibuat. Hasil dari pengujian sistem aplikasi yang telah dibuat yaitu pada pengujian algoritma pengacak soal, soal berhasil teracak dan untuk setiap siswa berhasil mendapatkan urutan soal yang berbeda-beda.

Saran pengembangan dari penulis yaitu sistem dikembangkan dengan mencoba menggunakan algoritma pengacak selain dari Linear Congruent Method, seperti *Monte Carlo* atau memodifikasi algoritma LCM seperti *Couple Linear Congruent Generator* (CLCG).

Referensi

- [1] Kemendikbud, "Tentang UNBK- Kementerian Pendidikan dan Kebudayaan," *kemendikbud*, 2018. .
- [2] S. D. Nasution, "Penerapan metode linier kongruendan algoritma vigenère chipper pada aplikasi sistem ujian berbasis lan," *STMIK Budi Darma Medan*, vol. IV, no. 1, pp. 94–102, 2013.
- [3] G. Gunawan and D. A. Prabowo, "SISTEM UJIAN ONLINE SELEKSI PENERIMAAN MAHASISWA BARU DENGAN PENGACAKAN SOAL MENGGUNAKAN LINEAR CONGRUENT METHOD (Studi Kasus di Universitas Muhammadiyah Bengkulu)," *J. Inform. Upgris*, vol. 3, no. 2, pp. 143–151, 2017.
- [4] KBBI, "KBBI - Pengacak," *KBBI*, 2016. .
- [5] D. P. Hasibuan, "Perancangan Simulasi Pengacakan Soal Tryout Untuk Membentuk Paket Soal Ujian Nasional Menggunakan Linear Congruent Method (Lcm)," *Pelita Inform. Budi Darma*, vol. IV, no. Lcm, pp. 119–125, 2013.
- [6] KBBI, "KBBI - SIMULASI," *KBBI*, 2012. .
- [7] P. S. I. Dora, W. Rita, "Perancangan Aplikasi Game Edukasi Pembelajaran Anak Usia Dini Menggunakan Linear Congruent Method (Lcm) Berbasis Android," *J. Inform. Glob.*, vol. 6, no. 1, pp. 7–14, 2015.
- [8] D. Setiyadi, "Aplikasi Pembelajaran Mengetik Cepat Berbasis Game Edukasi Dengan Linear Congruent Method (LCM)," *Bina Insa. Ict J.*, vol. Vol.3, No., no. 2355–3421, p. 8, 2016.
- [9] I. M. D. Biantara, I. M. Sudana, A. F. Suni, Suryono, and A. Hangga, "Modifikasi Metode Linear Congruential Generator Untuk Optimalisasi Hasil Acak," *Semin. Nas. Inform. 2015 (semnasIF 2015)*, vol. 2015, no. November, pp. 182–186, 2015.
- [10] "Model_waterfall_Rekayasa_Perangkat_Lunak." *Academia.edu*, 2017.