

PERBANDINGAN ALGORITMA LCG DAN SQL UNTUK PROSES PENGACAKAN SOAL UJIAN SEKOLAH (Studi kasus SMP N 1 Sedayu Bantul)

PRIYO ADI WICAKSONO

di Teknik Informatika Bisnis Dan Teknologi informasi
Universitas Teknologi Yogyakarta
Jl. Ringroad Utara Jombor Sleman Yogyakarta
E-mail: priyoadi20@gmail.com

ABSTRAK

Proses ujian di SMP Negeri 1 Sedayu masih bersifat konvensional yaitu dalam pembuatan soal ujian, pengadaan soal ujian, evaluasi ujian, serta siswa mengerjakan soal ujian dan pengisian jawaban ujian masih menggunakan kertas dan alat tulis. Hal ini menyebabkan soal ujian sama antara siswa yang satu dengan siswa yang lainnya, sehingga peluang untuk saling bertukar jawaban cukup besar. Salah satu cara untuk mengurangi keterbatasan adalah menggunakan sistem ujian online. Menerapkan metode Algoritma Linear Congruential Generator (LCG) dan SQL Random untuk membangun sistem ujian online tersebut. Berdasarkan penelitian yang dilakukan dapat disimpulkan bahwa dalam segi pengacakan soal metode LCG maupun SQL Random bisa dikatakan efektif karena tidak menghasilkan soal yang sama antara siswa satu dengan yang lainnya.

Kata Kunci : Pengacakan, Ujian online, LCG, SQL, Efektivitas

1. PENDAHULUAN

1.1 Latar Belakang

Proses ujian sekolah dapat dibedakan menjadi dua yaitu sistem ujian secara konvensional yang manual dalam sistem pengerjaannya masih menggunakan kertas dan alat tulis dan sistem ujian online yang dalam pengerjaannya melalui media elektronik berupa komputer, laptop, atau handphone yang tersambung internet. Dalam pelaksanaannya ujian online memiliki banyak kelebihan dibanding dengan ujian secara konvensional diantaranya yaitu mengurangi sumberdaya manusia, lebih efisien dalam penggunaan waktu dan biaya pelaksanaan ujian, dan pengacakan soal dalam ujian sehingga mengurangi kesempatan kepada siswa untuk saling bertukar jawaban.

Proses ujian di SMP Negeri 1 Sedayu masih bersifat konvensional yaitu dalam pembuatan soal ujian, pengadaan soal ujian, evaluasi ujian, serta siswa mengerjakan soal ujian dan pengisian jawaban ujian masih menggunakan kertas dan alat tulis. Hal ini menyebabkan soal ujian sama antara siswa yang satu dengan siswa yang lainnya sehingga peluang untuk saling bertukar jawaban cukup besar.

Dalam pembuatan aplikasi ini digunakan metode *Linear Congruential Generator (LCG)* dan *SQL*. Algoritma *Linear Congruential Generator (LCG)* dan *SQL* sama-sama digunakan untuk pengacakan soal-soal ujian dan keduanya mempunyai kelemahan dan kelebihan masing-masing. 2

Dengan adanya latar belakang tersebut maka penulis melakukan penelitian untuk membandingkan keefektifan dalam pengacakan soal-soal ujian menggunakan metode *Linear Congruential Generator (LCG)* dengan *SQL* berbasis website.

2. LANDASAN TEORI

2.1 Kajian Pustaka

(Fanani, Indriyanti, & Lazulfa, 2018) yang berjudul "Pengacakan Soal Pada Sistem *Computer Based Test (CBT)* Dengan Metode *Linear Congruential Generator (LCG)* Di Sma Negeri Jogoroto". Penerapan metode *LCG* dalam pengacakan soal ujian *CBT* berjalan dengan baik, metode *LCG* yang digunakan dapat mengacak soal ujian pada

sistem CBT. Metode *LCG* juga dapat digunakan untuk berbagai macam sistem ataupun aplikasi yang memiliki nilai acak. Pengacakan soal dengan menggunakan metode *LCG* menghasilkan nilai pengacakan yang periodik yang artinya nilai pengacakan mempunyai nilai yang sama pada periode tertentu, namun dengan memberikan variabel yang selalu berbeda – beda maka dapat mengatasi permasalahan nilai pengacakan yang periodik.

(Wahyudi & Pasaribu, 2015) yang berjudul “Perancangan Aplikasi Quiz Menggunakan Metode Pengacakan *Linear Congruential Generator (LCG)* Berbasis Android”. Aplikasi Quiz ini dirancang dengan terfokus pada penyajian soal yang berupa soal pilihan ganda serta dengan tampilan user interface yang mudah dipahami sehingga pengguna akan mudah mengerti untuk menggunakan aplikasi ini..

(Prabowo & Hangga, 2017) yang berjudul “Modifikasi *Linear Congruential Generator* Untuk Optimalisasi Penilaian Pembelajaran Computer Based Test (CBT)”. Penggunaan algoritma *MLCG* mempunyai persentase kesamaan soal berkisar 20%-35% untuk penggunaan bilangan koprima dan fibonacci, 10%-35% untuk bilangan koprima dan prima, dan 10%-35% untuk ketiga bilangan tersebut. 6 Penggunaan pembatasan bilangan pada variabel tidak memberikan perubahan yang signifikan dalam hal persentase kesamaan soal dibandingkan dengan algoritma *LCG*.

(Biantara, Sudana, Alfa Faridh Suni, & Hangga, 2015) yang berjudul “Modifikasi Metode *Linear Congruential Generator* Untuk Optimalisasi Hasil Acak”. Pada penelitian ini hasil simulasi menunjukkan metode Couple *Linear Congruential Generator* memiliki pola pengacakan yang lebih rumit dibandingkan metode *Linear Congruential Generator*. Pola pengacakan dipengaruhi oleh nilai *m* dan banyaknya pemberian kombinasi nilai pada variabel *a* dan *b*. Semakin banyak kombinasi nilai variabel dan maka semakin banyak pengacakan yang dihasilkan. Selain itu, semakin banyak nilai variabel *m* maka semakin rumit pola pengacakan yang dihasilkan. Karena dimodifikasi menggunakan matrik sehingga hanya pada kondisi tertentu dimana jumlah soal yang diacak harus sama dengan jumlah elemen yang terdapat dalam matrik. Apabila diimplementasikan dalam pengacakan soal dianjurkan menggunakan metode *Coupled Linear Congruential Generator* dikarenakan pola dan hasil pengacakan yang lebih bagus.

(Aprilia, Wibawa, & Kardian, 2017) yang berjudul “Aplikasi Ujian Online untuk SMA PKP JIS dengan Metode *Linear Congruential Generator (LCG)* Berbasis Website”. Aplikasi ujian online SMA PKP JIS adalah ap- likasi berbasis website dibuat dengan metode *Linear Congruential Generator (LCG)*, menggunakan bahasa pemrograman *PHP* dan *MySQL* untuk database. Metode *Linear Congruential Generator* digunakan untuk mengacak soal-soal ujian agar tidak terjadi kesamaan soal antara satu siswa dengan siswa lainnya.

2.2 Linear Congruential Generator

Linear Congruential Generator (LCG) merupakan salah satu pembangkit bilangan acak semu. *LCG* menggunakan metode linier dalam pembangkit bilangan acak dalam jumlah besar dan waktu yang cepat (Nelson, 2015). *Linear Congruential Generator (LCG)* merupakan pembangkit bilangan acak sederhana, mudah di mengerti teorinya, dan mudah untuk di implementasikan. *LCG* didefinisikan dalam relasi berulang berikut :

$$X_n = (aX_{n-1} + b) \bmod m$$

Keterangan:

X_n = bilangan acak ke-*n* dari deretnya

X_{n-1} = bilangan acak sebelumnya

a = faktor pengali

b = increment

m = modulus

LCG memiliki periode penuh jika :

1. *b* relatif prima terhadap *m*.
2. *a* – 1 dapat dibagi dengan setiap faktor prima dari *m*.
3. *a* – 1 adalah kelipatan 4 jika *m* kelipatan 4.

X_0 adalah kunci pembangkit atau disebut juga umpan (seed). *LCG* mempunyai periode tidak lebih besar dari *m*, dan pada kebanyakan kasus periodenya kurang dari itu.

Metode *Linear Congruential Generator (LCG)* adalah pengulangan pada periode waktu tertentu atau setelah sekian kali pembangkitan, hal tersebut adalah salah satu sifat dari metode tersebut dan Pseudorandom Generator pada umumnya. Penentuan konstanta *Linear Congruential Generator* (*a*, *b*, dan *m*) sangat menentukan baik tidaknya bilangan acak yang diperoleh, dalam arti memperoleh bilangan acak yang seakan-akan tidak terjadi pengulangan.

2.3 SQL

(Fehily, 2015) *Structured Query Language (SQL)* merupakan sekumpulan perintah khusus yang

digunakan untuk mengakses data dalam database relasional. *SQL* merupakan sebuah bahasa komputer yang mengikuti standar *ANSI (American National Standard Institute)* yang digunakan dalam manajemen database relasional. *SQL* didasarkan pada model relasional. Satu penyimpangan dari model adalah bahwa dalam *SQL*, kunci primer adalah opsional daripada wajib. Akibatnya, tabel tanpa kunci akan menerima baris duplikat, membuat beberapa data tidak dapat diakses. Kelemahan dari perbedaan ini adalah bahwa pengguna *DBMS*, dan bukan *DBMS* itu sendiri, bertanggung jawab untuk menegakkan struktur relasional. Dalam penggunaannya perintah *SQL* diategorikan menjadi tiga sub perintah, yaitu *DDL (Data Definition Language)*, *DML (Data Manipulation Language)* dan *DCL (Data Control Language)*.

Data Definition Language (DDL) merupakan sub Bahasa *SQL* yang digunakan untuk membangun kerangka database. Ada tiga perintah yang termasuk dalam *DDL* yaitu :

1. CREATE
2. ALTER
3. DROP

Data Manipulation Language (DML) merupakan sub Bahasa *SQL* yang digunakan untuk memanipulasi data dalam database yang telah terbuat. Perintah yang digunakan diantaranya :

1. INSERT
2. SELECT
3. UPDATE
4. DELETE

Data Control Language (DCL) merupakan sub Bahasa *SQL* yang digunakan untuk melakukan pengontrolan data dan server databasenya. Perintah *DCL*, di antaranya :

1. GRAND
2. REVOKE

2.4 Entity Relationship Diagram

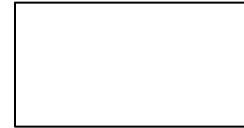
(Pal, 2015) *Entity Relationship Diagram (ERD)* merupakan gambaran sistematis yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta dari dunia nyata yang kita tinjau.

2.5 Komponen Entity Relationship Diagram

(Pal, 2015) ada dua komponen utama pembentuk model *Entity Relationship Diagram* yaitu entitas (*entity*) dan relasi. Kedua komponen ini

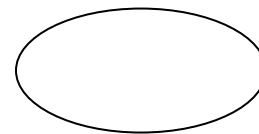
dideskripsikan lebih jauh melalui sejumlah atribut atau properti. Komponen-komponen *ERD* tersebut yaitu :

1. Entitas (*Entity*) Entitas merupakan individu yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Sekelompok entitas yang sejenis dan berada dalam lingkup yang sama membentuk sebuah himpunan entitas.



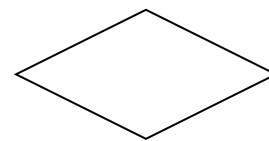
Gambar 2.1: Simbol entitas

2. Atribut (*Attributes/Properties*) Atribut atau bentuk elips adalah sesuatu yang menjelaskan apa sebenarnya yang dimaksud entitas atau relationship dan mewakili atribut dari masing-masing entitas.



Gambar 2.2 : Simbol Atribut

3. Relasi (*Relationship*) Relasi menunjukkan Adanya hubungan diantara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Kumpulan semua relasi diantara entitas-entitas yang terdapat pada himpunan entitas membentuk himpunan relasi.



Gambar 2.3: Simbol relasi

2.6 Data Flow Diagram

(Pal, 2015) *Data Flow Diagram (DFD)* merupakan alat untuk membuat diagram yang serbaguna. *Data Flow Diagram* terdiri dari notasi penyimpanan data (*data store*), proses (*process*), aliran data (*flow data*), dan sumber masukan entitas (*entity*).

2.7 Komponen Data Flow Diagram

Adapun komponen *Data Flow Diagram* menurut (Pal, 2015):

1. Kesatuan luar (eksternal entity)

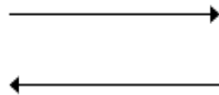
Setiap sistem mempunyai batas sistem yang memisahkan suatu sistem dengan lingkungan luarnya, sedangkan kesatuan luar merupakan kesatuan di lingkungan sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau menerima output dari sistem.



Gambar 2.4: Simbol Entitas

2. Arus data

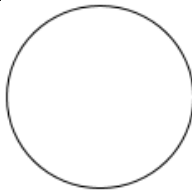
Arus data mengalir diantara proses, simpanan data dan kesatuan luar. Arus data sebaiknya diberi nama yang jelas dan mempunyai arti. Nama dari arus data dituliskan disamping garis panahnya.



Gambar 2.5: Simbol Arus Data

3. Proses

Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.



Gambar 2.6: Simbol Proses

4. Simpanan Data

Simpanan data merupakan simpanan dari data yang dapat berupa file atau database



Gambar 2.7: Simpan Data

2.8 Ujian Online

Menurut Prasetyo (2015) ujian online adalah proses pelaksanaan Ujian yang dilaksanakan secara langsung melalui komputer/laptop/tablet PC yang

terhubung dengan sambungan internet. Dengan pelaksanaan ujian online ini, maka peserta ujian diwajibkan untuk mempersiapkan diri dengan komputer/notebook/netbook/tablet masing-masing dan terhubung dengan sambungan internet pada jadwal yang telah ditentukan.

3. METODE PENELITIAN

3.1 Objek Penelitian

Penelitian tugas akhir ini dilakukan pada Sekolah Menengah Pertama Negeri 1 Sedayu yang beralamat di Jalan Agromulyo, Sedayu, Bantul, Yogyakarta 55752.

3.2 Metode Penelitian

Metode yang digunakan dalam menunjang penelitian mengenai sistem ujian online meliputi :

- Metode Observasi
- Metode Wawancara
- Metode Penelitian

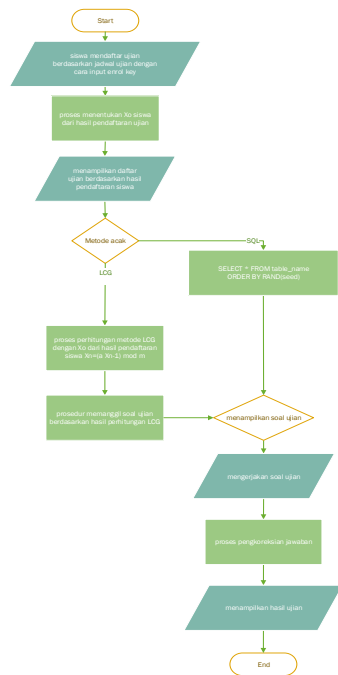
3.3 Analisis Kebutuhan Sistem

Pada tahap ini penulis melakukan analisis mengenai kebutuhan yang dibutuhkan untuk membangun perancangan dan Implementasi LCG dan SQL untuk proses pengacakan soal ujian sekolah di SMP Negeri 1 Sedayu. Sistem yang berjalan saat ini proses ujian di SMP Negeri 1 Sedayu masih bersifat konvensional yaitu dalam pembuatan soal ujian, penggandaan soal ujian, evaluasi ujian, serta siswa mengerjakan soal ujian dan pengisian jawaban ujian masih menggunakan kertas dan alat tulis.

Hal ini menyebabkan soal ujian sama antara siswa yang satu dengan siswa yang lainnya, sehingga peluang untuk saling bertukar jawaban cukup besar. Dengan adanya pembuatan aplikasi ini diharapkan dapat meminimalisir segala bentuk kecurangan siswa dan mempercepat pengolahan hasil ujian tersebut. Aplikasi ujian online ini akan dianalisis menggunakan DAD (*Diagram Arus Data*) sebagai metode alur datanya dan ERD (*Entity Relationship Diagram*) sebagai pemodelan datanya.

3.4 Flowchart

Flowchart digunakan untuk menggambarkan urutan proses dan hubungan antar proses secara mendetail dalam suatu program dan menentukan alur kerja pada sistem.



Gambar 3.1 : Flowchart

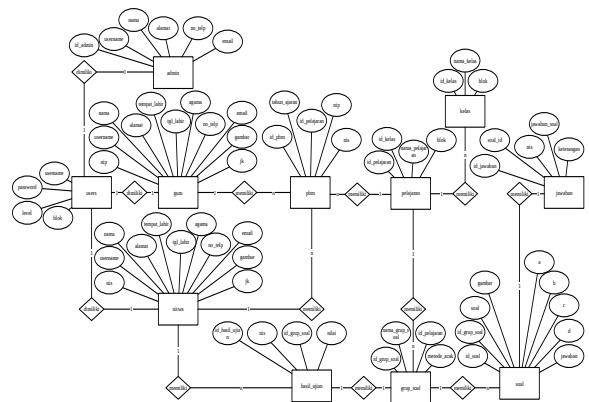
4. Hasil dan Pembahasan

Berdasarkan proses perbandingan pengacakan metode *Linear Congruential Generator (LCG)* dan *SQL Random* pada gambar 5.39 dan gambar 5.47 tersebut *Linear Congruential Generator (LCG)* hanya membutuhkan waktu 0,0021 detik sedangkan *SQL Random* membutuhkan waktu 0,3988 detik. Maka dapat disimpulkan bahwa Algoritma *Linear Congruential Generator (LCG)* lebih cepat dalam proses pengacakan dibandingkan *SQL Random*.

4.1 Perancangan Entity Relationship Diagram

Rancangan *ERD* Aplikasi Ujian Online SMP Negeri 1 Sedayu sebagai berikut:

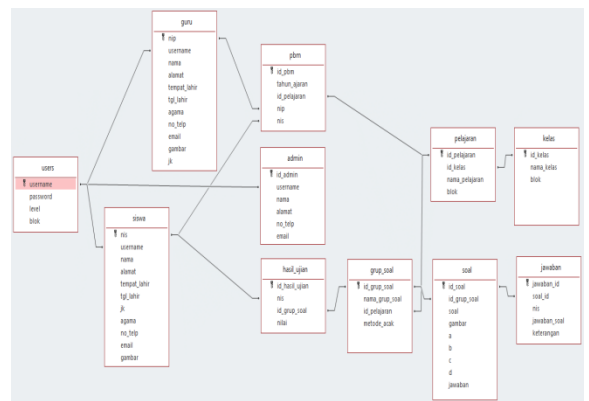
- Entitas users
- Entitas admin
- Entitas siswa
- Entitas kelas
- Entitas pelajaran
- Entitas pbm
- Entitas grup soal
- Entitas soal
- Entitas hasil ujian
- Entitas jawaban



Gambar 4.1 :Perancangan ERD

4.2 Relasi Antar Tabel

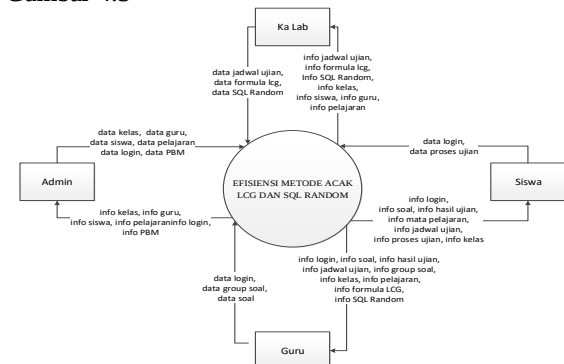
Rancangan dari relasi antar table di dalam database akan dijelaskan pada gambar berikut



Gambar 4.2 : Relasi Tabel

4.3 Diagram Konteks

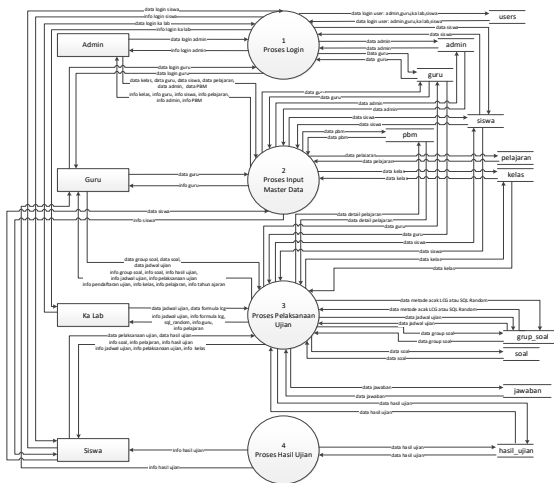
Hubungan antara Admin, Kepala Lab (Ka Lab), Guru dan Siswa terhadap sistem dijelaskan dengan diagram konteks. Untuk lebih jelas dapat dilihat pada Gambar 4.3



Gambar 4.3 :Diagram Konteks

4.4 DAD Level 1

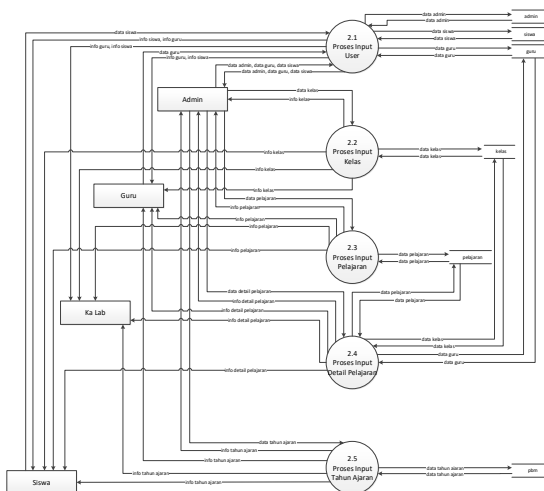
Dalam diagram arus data level 1 Perbandingan Algoritma *Linear Congruential Generator (LCG)* dan *SQL Random* Untuk Proses Pengacakan Soal Ujian Online di SMP Negeri 1 Sedayu terdapat empat proses Diagram arus data level 1 dijelaskan pada halaman selanjutnya Gambar 4.4.



Gambar 4.4 :DAD Level 1

4.5 DAD Level 2 Proses 1

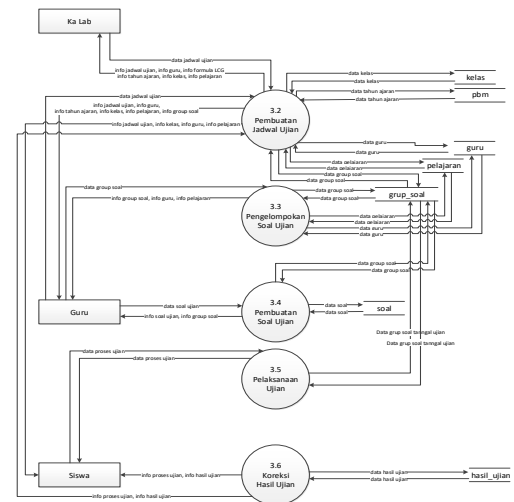
Diagram arus data level 2 proses 2 menjelaskan proses pelaksanaan ujian. Entitas kepala lab mengolah data formula *LCG* yang nantinya akan dijadikan variabel untuk pengacakan soal ujian dan mengolah data jadwal ujian Entitas. Berikut dijelaskan diagram arus data level 2 proses 2 dapat dilihat pada Gambar 4.6.



Gambar 4.5 :DAD Level 2 Proses 1

4.6 DAD Level 2 Proses 2

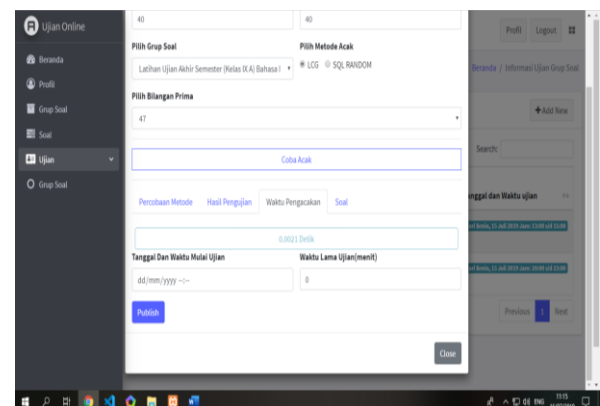
Diagram arus data level 2 proses 2 menjelaskan proses pelaksanaan ujian. Entitas kepala lab mengolah data formula *LCG* yang nantinya akan dijadikan variabel untuk pengacakan soal ujian dan mengolah data jadwal ujian diantaranya yaitu jumlah soal yang akan ditampilkan, jumlah siswa yang akan diujikan, dan pemilihan metode acak. Berikut dijelaskan diagram arus data level 2 proses 2 pada halaman selanjutnya Gambar 4.6.



Gambar 4.6 :DAD Level 2 Proses 2

4.7 Halaman Pengacakan LCG

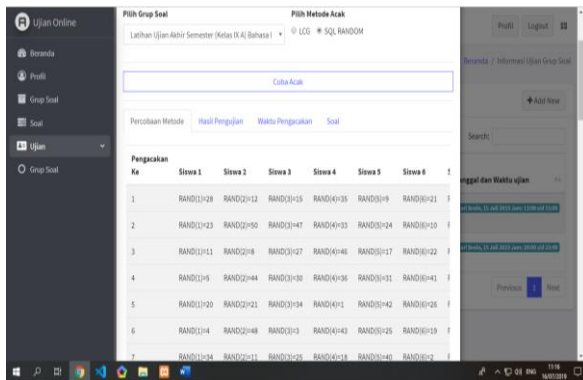
Pengacakan soal menggunakan metode *LCG* tersebut dengan pola pengacakan sesuai rumus. Setelah kepala lab menginputkan jumlah soal, jumlah siswa, pilih grup soal, dan metode acak kepala lab dapat mencoba pengujian. Proses acak *LCG* membutuhkan waktu 0,0021 detik. Hasilnya seperti gambar di bawah ini.



Gambar 4.7 : Pengacakan LCG

4.8 Halaman Pengacakan SQL

Pengacakan soal dengan metode *SQL* tersebut dengan pengacakan menggunakan *query*. Setelah kepala lab menginputkan jumlah soal, jumlah siswa, pilih grup soal, dan metode acak kepala lab dapat mencoba pengujian. Proses acak *SQL* membutuhkan waktu 0,3988 detik. Hasilnya seperti gambar di bawah ini. Hasilnya seperti gambar di bawah ini.



Gambar 4.8 : Pengacakan SQL

5. PENUTUP

5.1 Kesimpulan

Berdasarkan implementasi yang sudah dilakukan, maka dapat disimpulkan bahwa Perbandingan Algoritma *Linear Congruential Generator (LCG)* dan *SQL Random* Untuk Proses Pengacakan Soal Ujian Sekolah SMP N 1 Sedayu Yogyakarta. Hasil penelitian ini diantaranya :

1. Hasil pengacakan 50 soal untuk 40 siswa menggunakan metode *LCG* maupun *SQL Random* bisa dikatakan berhasil karena hasil acak menunjukkan soal tiap siswa satu dengan yang lainnya berbeda.
2. Adanya perbedaan waktu pengacakan yaitu 50 soal untuk 40 siswa lebih cepat menggunakan metode *Linear Congruential Generator (LCG)*. Karena metode *Linear Congruential Generator (LCG)* hanya membutuhkan waktu 0,0021 detik sedangkan *SQL Random* membutuhkan waktu 0,3988 detik.
3. Adanya perbedaan waktu pengacakan yaitu 50 soal untuk 40 siswa lebih cepat menggunakan metode *LCG* karena 1 request pada perintah *query* sudah dapat diimplementasikan sebagai pembangkit acak 50 soal untuk 40 siswa sedang *SQL Random*

membutuhkan 1 request perintah *query* untuk setiap siswanya.

5.2 Saran

Dari hasil penelitian, perancangan dan implementasi yang sudah dilakukan, maka ada beberapa saran yang dapat dikemukakan untuk membangun aplikasi yang lebih baik lagi untuk kedepannya, yaitu :

1. Menampilkan hasil perbandingan yang lebih banyak.
2. Menemukan perhitungan manual *SQL Random* tanpa menggunakan bantuan software.
3. Membandingkan pengacakan untuk jumlah yang lebih besar.

DAFTAR PUSTAKA

- [1] Aprilia, D., Wibawa, & Kardian, A. R. (2017). Aplikasi Ujian Online untuk SMA PKP JIS dengan Metode *Linear Congruential Generator (LCG)* Berbasis Website, 16(September), 145–150.
- [2] Biantara, I. M. D., Sudana, I. M., Alfa Faridh Suni, S., & Hangga, A. (2015). MODIFIKASI METODE *LINEAR CONGRUENTIAL GENERATOR* UNTUK OPTIMALISASI HASIL ACAK. SemNaSIF, 2015 (November), 182–186. <https://doi.org/10.1109/IROS.2008.4650764>
- [3] Boell, S. K., & Cecez-Kecmanovic, D. (2015). What is an information system? *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2015–March (January), 4959–4968. <https://doi.org/10.1109/HICSS.2015.587>
- [4] Fanani, A., Indriyanti, A. D., & Lazulfa, I. (2018). PENGACAKAN SOAL PADA SISTEM *COMPUTER BASED TEST (CBT)* DENGAN METODE *LINEAR CONGRUENTIAL GENERATOR (LCG)* DI SMA NEGERI JOGOROTO Achmad Fanani Abstrak, 03, 20–29.
- [5] Fehily, C. (2015). *SQL Database Programing*.
- [6] Pal, M. (2015). *Microsoft Dynamics CRM 2015 Application Design*.
- [7] Prabowo, H. E., & Hangga, A. (2017). MODIFIKASI *LINEAR CONGRUENTIAL GENERATOR* UNTUK OPTIMALISASI PENILAIAN PEMBELAJARAN *COMPUTER BASED TEST (CBT)*, (Aisi 1045), 41–46.
- [8] Wahyudi, R., & Pasaribu, H. H. S. (2015). Perancangan Aplikasi Quiz Menggunakan Metode Pengacakan *Linear Congruential Generator (LCG)* Berbasis Android. *Riau Journal of Computer Science*, 1(1), 17–26.