



Implementasi Metode LCG Pada Aplikasi CBT Untuk Tes Potensi Akademik Berbasis Web

Rizky Rinaldi

Teknologi Informasi, Sains & Teknologi, Universitas Pembangunan Pancabudi
rizkyrinaldi055@gmail.com

Abstract

In the Academic Potential Test of new students at the Raudhatul Akmal Islamic College (STAIRA) Batang Kuis, do not yet have a Computer Based Test (CBT) or a computer-based exam for the freshman academic potential test exam. So that the exam is still carried out manually and costs a lot of money in its implementation. But it also makes the exam process less carried out because some students can cheat when filling in answers by cheating friends. Therefore, in this study aims to create a web-based Computer Based Test (CBT) application and randomize the questions in it by applying the Linear Congruent Generator (LCG) method. The result of this study is a web-based application for academic potential test exams that are expected to overcome the problems that exist in the current STAIRA Batang Kuis. The order of the questions will be automatically randomized when students conduct an academic potential test exam based on their respective student identification numbers.

Keywords: academic potential test, computer based test, linear congruent generator, website, black box testing

Abstrak

Dalam Tes Potensi Akademik (TPA) mahasiswa baru di Sekolah Tinggi Agama Islam Raudhatul Akmal (STAIRA) Batang Kuis, belum memiliki *Computer Based Test (CBT)* atau ujian berbasis komputer untuk ujian tes potensi akademik mahasiswa baru. Sehingga ujian masih dilaksanakan dengan cara manual dan menelan banyak biaya dalam pelaksanaannya. Namun itu juga menjadikan proses ujian terlaksana kurang baik karena beberapa mahasiswa dapat melakukan kecurangan ketika mengisi jawaban dengan cara mencontek teman. Oleh karena itu, dalam penelitian ini bertujuan untuk membuat sebuah aplikasi *Computer Based Test (CBT)* berbasis web dan melakukan pengacakan soal di dalamnya dengan menerapkan metode *Linear Congruent Generator (LCG)*. Hasil dari penelitian ini adalah sebuah aplikasi berbasis web untuk ujian tes potensi akademik yang diharapkan dapat mengatasi permasalahan yang ada di STAIRA Batang Kuis saat ini. Urutan soal akan otomatis teracak ketika mahasiswa melakukan ujian tes potensi akademik berdasarkan nomor induk mahasiswa masing-masing.

Kata kunci: tes potensi akademik, *computer based test*, *linear congruent generator*, *website*, *black box testing*

1. Pendahuluan

Ujian merupakan salah satu cara untuk mengevaluasi proses belajar. Dalam dunia pendidikan, ujian ditujukan untuk mengukur taraf pencapaian suatu tujuan pengajaran oleh mahasiswa sebagai peserta didik. Bila ternyata hasilnya kurang maksimal maka proses pembelajaran harus ditingkatkan dari segi kualitas maupun kuantitasnya [1].

Hampir setiap negara sedang mempertimbangkan ujian secara *online*. Penilaian berbasis komputer telah menjadi standar dibanyak negara dan menjadi semakin menarik untuk departemen pendidikan, legislatif, dan pembuat kebijakan lainnya. Diprediksi kedepannya sistem ujian *online* akan banyak digunakan dalam bidang pendidikan [2].

Di dalam dunia pendidikan tinggi, ujian konvensional kerap digunakan sebagai kegiatan untuk mengevaluasi

pengetahuan mahasiswa. Salah satu ujian yang kerap dilakukan di perguruan tinggi adalah ujian tes potensi akademik bagi mahasiswa baru. Tes potensi akademik ditujukan guna mengukur kemungkinan keberhasilan mahasiswa dalam mengikuti pembelajaran di perguruan tinggi tersebut. Ujian konvensional atau sekarang dikenal dengan istilah *Paper-Based Test (PBT)* memiliki beberapa kekurangan yaitu, rawan kebocoran pada naskah soal, kecurangan selama pelaksanaan ujian, perlu waktu lama dalam pemeriksaan lembar jawaban dan *scoring*, serta membutuhkan banyak biaya, tenaga dan waktu. Dengan demikian dapat disimpulkan bahwa ujian secara konvensional kurang efektif dan efisien [3].

Berdasarkan keterangan dari Ketua Yayasan, STAIRA Batang Kuis yang selanjutnya disebut STAI Raudhatul Akmal Batang Kuis masih harus mencetak lembar soal ujian yang membutuhkan waktu hingga satu minggu.



Lisensi

Lisensi Internasional Creative Commons Attribution-ShareAlike 4.0.

Permasalahan tidak hanya sampai disitu, berdasarkan pernyataan beliau, kerap didapati mahasiswa yang melakukan kecurangan atau mencontek saat mengerjakan ujian, dosen membutuhkan waktu yang lama dalam pengkoreksian jawaban ujian, dan penyimpanan nilai hasil ujian mahasiswa yang tidak teratur sehingga menghambat pekerjaan dosen. Dengan perkembangan teknologi informasi, membuat pihak STAI Raudhatul Akmal Batang Kuis ingin membuat ujian tes potensi akademik secara *online* untuk mahasiswa baru dalam bentuk aplikasi *Computer Based Test* (CBT).

CBT adalah tes atau penilaian yang berbasis komputer dan sebagian besar soal berbentuk pilihan ganda. Secara umum CBT dimanfaatkan untuk meningkatkan efisiensi dan efektifitas administrasi tes. CBT yang dimaksud disini adalah tes *online* dengan mode terkontrol (*Controlled Mode*). Mode ini merupakan konsep CBT dimana ujian dilakukan tanpa pengawasan, namun peserta yang dapat mengikuti tes hanya yang sudah terdaftar dan kemudian *login* [4].

Metode *Linear Congruent Generator* (LCG) merupakan jenis *Pseudorandom Number Generator* (PRNG) yang banyak dipergunakan dalam aplikasi komputer modern. LCG ditemukan oleh matematikawan asal Amerika bernama D.H Lehmer yang kemudian metode LCG tersebut banyak digunakan oleh *programmer* untuk menghasilkan bilangan *random* dalam jumlah besar dan waktu yang cepat [5].

Aplikasi berbasis *web* adalah sebuah aplikasi yang dapat diakses melalui *internet* dan pada sekarang ini ternyata lebih banyak dan lebih luas pemakaiannya. Banyak dari perusahaan-perusahaan berkembang yang menggunakan aplikasi berbasis *web* dalam merencanakan sumber daya mereka dan untuk mengelola perusahaan mereka. Sebagai contoh, aplikasi berbasis *web* dapat digunakan untuk membuat *invoice* dan memberikan cara yang mudah dalam penyimpanan data di *database* [6]. Aplikasi berbasis *web* dibangun dengan bahasa-bahasa pemrograman. Bahasa Pemrograman merupakan sarana penghubung antara yang diharapkan pembuat dengan program itu sendiri menggunakan bahasa yang dimengerti sistem [7].

PHP merupakan singkatan dari *Hypertext Preprocessor* yang, PHP sendiri dapat dikatakan sebagai bahasa pemrograman *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi *server*. Fungsi utama PHP dalam membangun *website* adalah untuk melakukan pengolahan data pada *database*. Data *website* akan dimasukkan ke *database*, diedit, dihapus, dan ditampilkan pada *website* yang diatur oleh PHP. PHP berasal dari kata *Hypertext Preprocessor*, yaitu bahasa pemrograman universal untuk penanganan pembuatan dan pengembangan sebuah situs *web* dan bisa digunakan bersamaan dengan HTML [8].

Basis data (*database*) dapat dibayangkan sebagai sebuah lemari arsip. Jika memiliki lemari arsip dan bertugas mengelolanya, maka akan melakukan hal-hal seperti memberi sampul, memberi nomor, lalu menempatkan arsip-arsip tersebut dengan urutan tertentu dalam lemari. Basis data dapat diartikan sebagai markas atau gudang, tempat bersarang atau berkumpul. Sedangkan data merupakan representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, hewan peristiwa, konsep, keadaan, dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya [9].

MySQL merupakan *database engine* atau *server database* yang mendukung bahasa *database SQL* sebagai bahasa interaktif dalam mengelola data. MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user* [10]. MySQL bersifat *open source* sehingga banyak di gunakan untuk media. Walaupun gratis, MySQL tetap berkualitas dan sudah cukup memberikan *performance* yang memadai. Penggunaan PhpMyAdmin lebih mudah digunakan karena menggunakan *interface* yang lebih mudah dipahami [11].

Sistem yang telah dibangun perlu dilakukan pengujian untuk memastikan sistem telah dapat menjalankan tugas yang telah ditetapkan. Dalam pembahasan ini pengujian sistem dilakukan dengan konsep *Black Box Testing*. *Black Box Testing* merupakan metode pengujian sebuah *software* yang diperlukan untuk menguji suatu *software* tanpa mengetahui internal kode atau program. Pengujian *Black Box* yang memiliki arti bahwa pengujian ini hanya memeriksa suatu perangkat lunak dari hasil eksekusinya, tanpa harus tahu mengetahui kode program dan hanya memperhatikan fungsionalnya saja [12].

2. Metode Penelitian

Di dalam melakukan penelitian ini terdapat beberapa langkah yang dilakukan dalam mengumpulkan data berdasarkan pada STAI Raudhatul Akmal Batang Kuis, diantaranya yaitu observasi dan wawancara. Kegiatan observasi dalam penelitian ini dilakukan dengan tinjauan langsung ke STAI Raudhatul Akmal Batang Kuis. Di kampus tersebut dilakukan analisis masalah yang dihadapi dengan cara mengamati proses ujian konvensional yang dilakukan mahasiswa dan melihat paket soal yang diberikan kepada mahasiswa yang melakukan ujian. Selain itu juga dilakukan analisis kebutuhan dari permasalahan yang ada sehingga dapat dilaksanakannya pemodelan sistem. Setelah kegiatan observasi, dilakukan wawancara kepada Ketua Yayasan untuk mengetahui apa saja permasalahan yang terjadi ketika mengadakan ujian konvensional, khususnya ujian tes potensi akademik bagi mahasiswa baru.

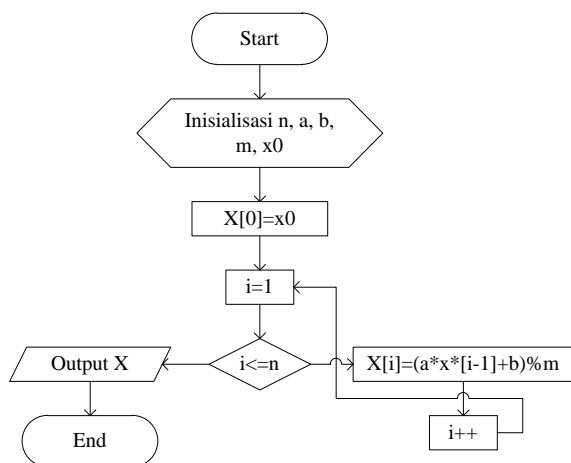
Kegiatan pengumpulan data yang dilakukan, menghasilkan beberapa data dalam bentuk dokumen dan rekaman arsip. Adapun data yang diambil berdasarkan pada STAI Raudhatul Akmal Batang Kuis adalah sampel data mahasiswa dan dokumen soal ujian tes potensi akademik, dokumen soal yang didapatkan dari STAI Raudhatul Akmal Batang Kuis berjumlah 100 butir soal yang lengkap dengan kunci jawaban.

Tabel 1. Sampel Data Mahasiswa

NIM	Nama mahasiswa	Jurusan
2019.10xx	Ade nur hajjah	PAI
2019.10xx	Adi siburian	PAI
2019.10xx	Tri utami	PAI
2019.10xx	Rifky amriza	PAI
2019.10xx	Muhammad chandra	PAI
2019.10xx	Hazzarul aswad	PAI
2019.10xx	Fauzan azli	PAI
2019.10xx	Hafiz apriadi lubis	PAI
2019.10xx	Rizky teguh	PAI
2019.10xx	Irfan irawan	PAI
2019.20xx	Wilda syaputri	PGMI
2019.20xx	Erliyani	PGMI
2019.20xx	Irma yanti	PGMI
2019.20xx	Githa syafriani lubis	PGMI
2019.20xx	Rendi lesmana	PGMI
2019.20xx	Lisa febriyanti	PGMI
2019.30xx	Fatkur rahman	PGRA
2019.30xx	Dodi irawan	PGRA
2019.30xx	Yopi astri	PGRA
2019.30xx	Emilia puspita	PGRA

Model Perancangan Sistem

Model perancangan sistem berperan penting dalam penelitian untuk menggambarkan aktivitas pengembangan dalam sistem. Algoritma sistem merupakan penjelesaian langkah-langkah penyelesaian dari suatu masalah. Berikut ini adalah *flowchart* dari metode *Linear Congruent Generator* (LCG) yaitu :



Gambar 1. Flowchart Metode LCG

3. Hasil dan Pembahasan

Berikut ini merupakan tahapan penyelesaian dengan metode *Linear Congruent Generator* (LCG) adalah sebagai berikut :

1. Menentukan konstanta a, b, m dan banyaknya pengacakan (n).
2. Menentukan umpan (X_0).
3. Mendapatkan pola pengacakan

3.1. Penentuan dan Pengujian Konstanta

Penentuan dan pengujian konstanta ini sangat penting untuk dilakukan, karena bahwa baik tidaknya bilangan acak yang dihasilkan didasarkan pada pemilihan konstanta a, b , dan m yang tepat. Seakan-akan tidak terjadi pengulangan dalam pola acaknya. Pemilihan konstanta ini juga harus memperhatikan berapa banyak butir soal yang nantinya akan ditampilkan (n) dalam aplikasi CBT STAI Raudhatul Akmal Batang Kuis. Dari hasil pengujian berhasil didapatkan konstanta yang dianggap tepat yaitu: $a = 11$, $b = 7$, $m = 100$, dan $n = 50$. Penentuan konstanta-konstanta tersebut memiliki beberapa alasan, antara lain :

1. $a = 11$, dikarenakan $a - 1$ dapat habis jika dibagi oleh semua faktor prima dari m .
2. $b = 7$, dikarenakan 7 merupakan bilangan yang relatif prima terhadap m .
3. $m = 100$, disesuaikan dengan jumlah soal yang ada di bank soal.
4. $n = 50$, disesuaikan dengan banyaknya soal yang akan ditampilkan pada aplikasi CBT.

3.2. Menentukan Umpan (X_0)

Dalam menentukan umpan (X_0) ini, akan digunakan 20 sampel data mahasiswa yang sudah didapatkan dari Sekolah Tinggi Agama Islam Raudhatul Akmal Batang Kuis. Dari data tersebut, empat angka terakhir NIM setiap mahasiswa akan digunakan sebagai umpan (X_0) untuk menghasilkan pola pengacakan yang berbeda-beda dari masing-masing umpan yang diproses. Berikut ini merupakan data dari hasil penentuan umpan (X_0) berdasarkan NIM masing-masing mahasiswa.

Tabel 2. Penentuan Umpan (X_0)

Nama Mahasiswa	NIM	Umpan (X_0)
Ade nur hajjah	2019.10xx	1079
Adi siburian	2019.10xx	1022
Tri utami	2019.10xx	1026
Rifky amriza	2019.10xx	1055
Muhammad chandra	2019.10xx	1050
Hazzarul aswad	2019.10xx	1003
Fauzan azli	2019.10xx	1033
Hafiz apriadi lubis	2019.10xx	1097
Rizky teguh	2019.10xx	1008
Irfan irawan	2019.10xx	1002
Wilda syaputri	2019.20xx	2011
Erliyani	2019.20xx	2004
Irma yanti	2019.20xx	2016
Githa syafriani lubis	2019.20xx	2020
Rendi lesmana	2019.20xx	2017
Lisa febriyanti	2019.20xx	2001
Fatkur rahman	2019.30xx	3009
Dodi irawan	2019.30xx	3088
Yopi astri	2019.30xx	3007
Emilia puspita	2019.30xx	3013

3.3. Mendapatkan Pola Pengacakan

Dalam pembahasan ini, akan digunakan umpan dan konstanta yang telah ditentukan sebelumnya. Perhitungan LCG untuk mendapatkan pola pengacakan dalam sistem jika dihitung secara manual, dapat kita lihat penyelesaiannya sebagai berikut :
 Diketahui : $a = 11, b = 7, m = 100$ dan $n = 50$.

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

Penyelesaian :

Mahasiswa I dengan umpan (X_0) 1079

- $X_1 = (11 * 1079 + 7) \bmod 100 = 76$
- $X_2 = (11 * 76 + 7) \bmod 100 = 43$
- $X_3 = (11 * 43 + 7) \bmod 100 = 80$
- $X_4 = (11 * 80 + 7) \bmod 100 = 87$
- $X_5 = (11 * 87 + 7) \bmod 100 = 64$
- $X_6 = (11 * 64 + 7) \bmod 100 = 11$
- $X_7 = (11 * 11 + 7) \bmod 100 = 28$
- $X_8 = (11 * 28 + 7) \bmod 100 = 15$
- $X_9 = (11 * 15 + 7) \bmod 100 = 72$
- $X_{10} = (11 * 72 + 7) \bmod 100 = 99$
- $X_{11} = (11 * 99 + 7) \bmod 100 = 96$
- $X_{12} = (11 * 96 + 7) \bmod 100 = 63$
- $X_{13} = (11 * 63 + 7) \bmod 100 = 0$
- $X_{14} = (11 * 0 + 7) \bmod 100 = 7$
- $X_{15} = (11 * 7 + 7) \bmod 100 = 84$
- $X_{16} = (11 * 84 + 7) \bmod 100 = 31$
- $X_{17} = (11 * 31 + 7) \bmod 100 = 48$
- $X_{18} = (11 * 48 + 7) \bmod 100 = 35$
- $X_{19} = (11 * 35 + 7) \bmod 100 = 92$
- $X_{20} = (11 * 92 + 7) \bmod 100 = 19$
- $X_{21} = (11 * 19 + 7) \bmod 100 = 16$
- $X_{22} = (11 * 16 + 7) \bmod 100 = 83$
- $X_{23} = (11 * 83 + 7) \bmod 100 = 20$
- $X_{24} = (11 * 20 + 7) \bmod 100 = 27$
- $X_{25} = (11 * 27 + 7) \bmod 100 = 4$
- $X_{26} = (11 * 4 + 7) \bmod 100 = 51$
- $X_{27} = (11 * 51 + 7) \bmod 100 = 68$
- $X_{28} = (11 * 68 + 7) \bmod 100 = 55$
- $X_{29} = (11 * 55 + 7) \bmod 100 = 12$
- $X_{30} = (11 * 12 + 7) \bmod 100 = 39$
- $X_{31} = (11 * 39 + 7) \bmod 100 = 36$
- $X_{32} = (11 * 36 + 7) \bmod 100 = 3$
- $X_{33} = (11 * 3 + 7) \bmod 100 = 40$
- $X_{34} = (11 * 40 + 7) \bmod 100 = 47$
- $X_{35} = (11 * 47 + 7) \bmod 100 = 24$
- $X_{36} = (11 * 24 + 7) \bmod 100 = 71$
- $X_{37} = (11 * 71 + 7) \bmod 100 = 88$
- $X_{38} = (11 * 88 + 7) \bmod 100 = 75$
- $X_{39} = (11 * 75 + 7) \bmod 100 = 32$
- $X_{40} = (11 * 32 + 7) \bmod 100 = 59$
- $X_{41} = (11 * 59 + 7) \bmod 100 = 56$
- $X_{42} = (11 * 56 + 7) \bmod 100 = 23$
- $X_{43} = (11 * 23 + 7) \bmod 100 = 60$
- $X_{44} = (11 * 60 + 7) \bmod 100 = 67$
- $X_{45} = (11 * 67 + 7) \bmod 100 = 44$
- $X_{46} = (11 * 44 + 7) \bmod 100 = 91$

- $X_{47} = (11 * 91 + 7) \bmod 100 = 8$
- $X_{48} = (11 * 8 + 7) \bmod 100 = 95$
- $X_{49} = (11 * 95 + 7) \bmod 100 = 52$
- $X_{50} = (11 * 52 + 7) \bmod 100 = 79$

Pola acak yang dihasilkan untuk $X_0 = 1079$ adalah 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79.

Proses perhitungan terus dilakukan sebanyak data sampel mahasiswa yang ada sehingga didapatkan hasil pola bilangan acak untuk seluruh data sampel. Berikut ini merupakan hasil selengkapnya.

Tabel 3. Pola Acak Setiap Mahasiswa

NIM	Pola Bilangan Acak Yang Dihasilkan
2019.10xx	76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79.
2019.10xx	49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22.
2019.10xx	93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26.
2019.10xx	12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55.
2019.10xx	57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50.
2019.10xx	40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3.
2019.10xx	70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33.
2019.10xx	74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97.
2019.10xx	95, 52, 79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8.
2019.1002	29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2.
2019.20xx	28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79, 76, 43, 80, 87, 64, 11.
2019.20xx	51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4.
2019.20xx	83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52.

	79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16.
2019.20xx	27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20.
2019.20xx	94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17.
2019.20xx	18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1.
2019.30xx	6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13, 50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9.
2019.30xx	75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7, 84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88.
2019.30xx	84, 31, 48, 35, 92, 19, 16, 83, 20, 27, 4, 51, 68, 55, 12, 39, 36, 3, 40, 47, 24, 71, 88, 75, 32, 59, 56, 23, 60, 67, 44, 91, 8, 95, 52, 79, 76, 43, 80, 87, 64, 11, 28, 15, 72, 99, 96, 63, 0, 7.
2019.30xx	50, 57, 34, 81, 98, 85, 42, 69, 66, 33, 70, 77, 54, 1, 18, 5, 62, 89, 86, 53, 90, 97, 74, 21, 38, 25, 82, 9, 6, 73, 10, 17, 94, 41, 58, 45, 2, 29, 26, 93, 30, 37, 14, 61, 78, 65, 22, 49, 46, 13.

Dari hasil perhitungan dengan metode LCG, dapat dilihat pada tabel diatas bahwa pola acak yang didapat dari setiap NIM mahasiswa tidak ada yang sama. Masing-masing periode pola acak yang dihasilkan, tidak didapati bilangan yang muncul lebih dari satu kali. Itu menandakan bahwa penentuan konstanta-konstanta yang dilakukan sudah tepat. Sehingga nantinya pada sistem yang dibuat urutan soal setiap mahasiswa berbeda-beda dan tidak terdapat soal yang sama muncul lebih dari satu kali pada setiap paket soal yang didapat mahasiswa baru ketika ujian tes potensi akademik menggunakan aplikasi CBT berbasis web.

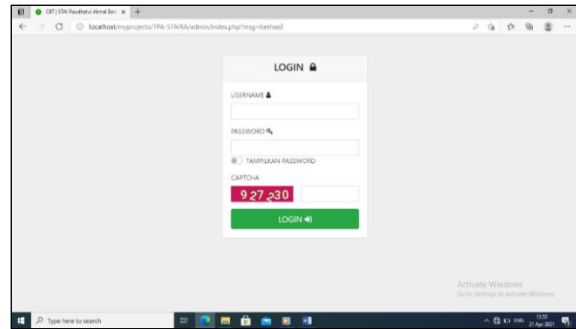
Pertanyaan ditampilkan berdasarkan hasil pengacakan dengan cara memberikan ID pertanyaan dari 0 sampai 99 (sebanyak 100 pertanyaan) sehingga pertanyaan yang ditampilkan adalah pertanyaan dengan ID yang merupakan bilangan hasil pengacakan.

3.4. Implementasi Sistem

Implementasi merupakan langkah pengoperasian sistem yang telah dibangun. Di sini akan dijelaskan bagaimana hasil tampilan sistem yang telah dibangun. Berikut ini merupakan tampilan sistem yang telah dibangun.

3.4.1 Tampilan Form Login Admin

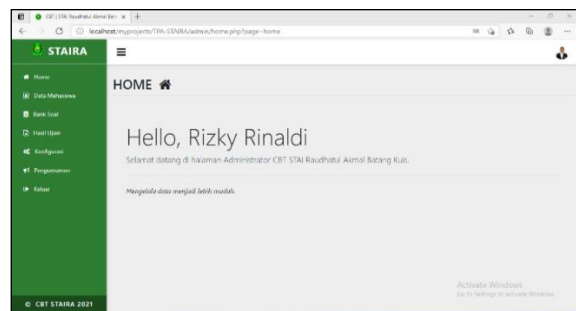
Berikut ini merupakan tampilan form login admin yang merupakan halaman dimana admin harus login melalui form ini untuk bisa masuk ke halaman menu utama.



Gambar 2. Tampilan Form Login Admin

3.4.2 Tampilan Menu Utama Admin

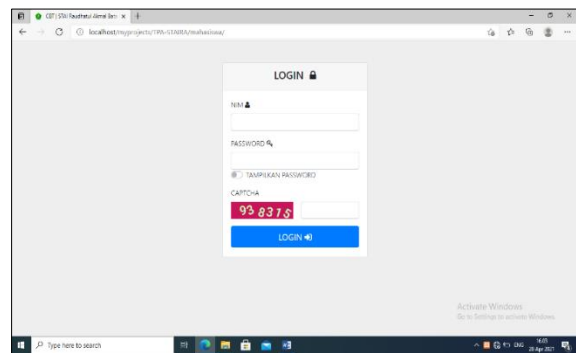
Berikut ini merupakan tampilan menu utama admin yang merupakan halaman yang akan ditampilkan apabila admin berhasil login.



Gambar 3. Tampilan Menu Utama Admin

3.4.3 Tampilan Form Login Mahasiswa

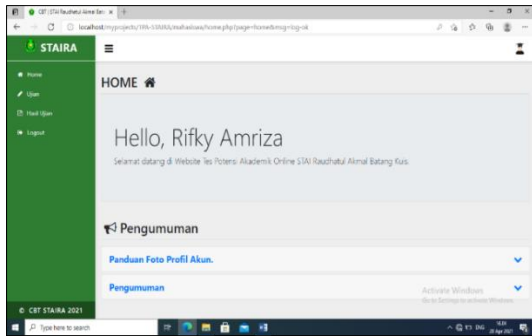
Ketika mahasiswa ingin mengakses halaman utama aplikasi, mahasiswa diminta untuk login terlebih dahulu. Untuk halaman form login mahasiswa dapat dilihat pada gambar di bawah ini.



Gambar 4. Tampilan Form Login Mahasiswa

3.4.4 Tampilan Halaman Menu Utama Mahasiswa

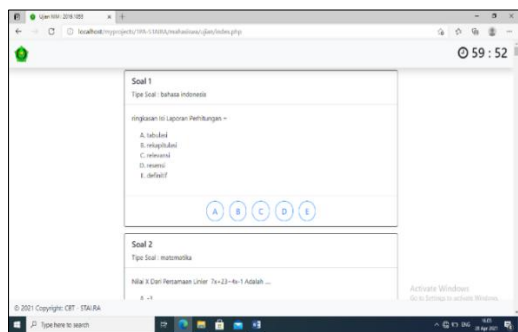
Setelah berhasil login mahasiswa dihadapkan dengan halaman menu utama. Berikut ini tampilan dari halaman menu utama mahasiswa.



Gambar 5. Tampilan Halaman Menu Utama Mahasiswa

3.4.5 Tampilan Halaman Ujian

Halaman ini digunakan mahasiswa untuk melaksanakan ujian. Tampilan halaman ujian mahasiswa dapat dilihat pada gambar berikut ini.



Gambar 6. Tampilan Form Login Mahasiswa

3.4.6 Tampilan Halaman Laporan Hasil Ujian

Hasil ujian mahasiswa dapat dicetak dalam bentuk laporan pdf atau langsung ke printer. Berikut ini tampilan halaman laporan hasil ujian.



Gambar 7. Tampilan Halaman Laporan Hasil Ujian

3.5. Pengujian Black Box

Setelah melakukan proses implementasi, proses selanjutnya adalah melakukan pengujian terhadap sistem yang telah dibangun. Pengujian ini dilakukan dengan *black box testing* untuk menemukan kemungkinan adanya kesalahan yang terjadi pada sistem CBT yang dibangun. Berikut ini adalah hasil pengujian *black box* aplikasi CBT untuk ujian tes potensi akademik mahasiswa baru di STAI Raudhatul Akmal Batang Kuis.

3.5.1 Login Admin

Berikut ini skenario pengujian dan hasil pengujian *black box* untuk fungsi login admin.

Tabel 4. Black Box Testing Login Admin

Skenario Pengujian	Uji Kasus
Admin salah memasukan <i>username</i> , <i>password</i> atau <i>captcha</i> .	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan gagal.	

3.5.2 Input Data Mahasiswa

Berikut ini skenario pengujian dan hasil pengujian *black box* untuk fungsi input data mahasiswa.

Tabel 5. Black Box Testing Input Data Mahasiswa

Skenario Pengujian	Uji Kasus
Admin memasukan NIM yang sama.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan gagal.	

3.5.3 Input Data Soal

Berikut ini skenario pengujian dan hasil pengujian *black box* untuk fungsi input data soal.

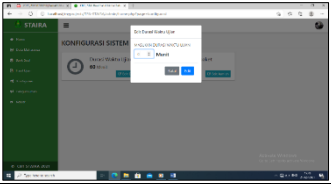
Tabel 6. Black Box Testing Input Data Soal

Skenario Pengujian	Uji Kasus
Admin mengosongkan kolom pertanyaan.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan pemberitahuan.	

3.5.4 Konfigurasi (Edit Durasi Waktu)

Berikut ini pengujian *black box* untuk fungsi *edit* durasi waktu.


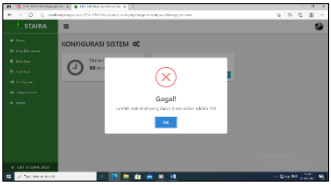
Tabel 7. Black Box Testing Edit Durasi Waktu

Skenario Pengujian	Uji Kasus
Admin mengisi <i>field</i> waktu dengan nilai 0.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan pemberitahuan.	

3.5.5 Konfigurasi (Edit Jumlah Soal Tampil)

Berikut ini pengujian *black box* untuk fungsi *edit* jumlah soal tampil.

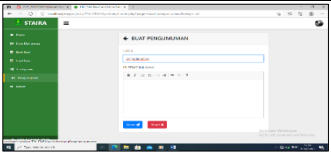
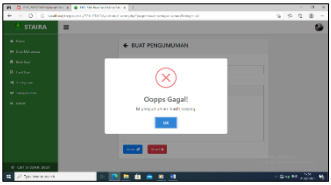
Tabel 8. Black Box Testing Edit Jumlah Soal Tampil

Skenario Pengujian	Uji Kasus
Admin memasukan nilai yang berjumlah lebih dari 100.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan pemberitahuan.	

3.5.6 Buat Pengumuman

Berikut ini pengujian *black box* untuk fungsi *buat* pengumuman.

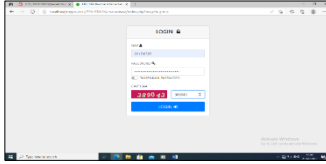
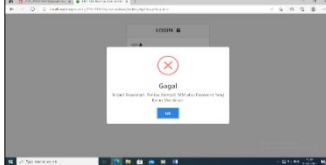
Tabel 9. Black Box Testing Buat Pengumuman

Skenario Pengujian	Uji Kasus
Admin mengosongkan isi pengumuman.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan pemberitahuan.	

3.5.7 Login Mahasiswa

Dikarenakan pada aplikasi ini memiliki dua *role user* yaitu admin dan mahasiswa maka pada table berikut ini merupakan pengujian dan hasil pengujian *black box* untuk fungsi *login* mahasiswa.

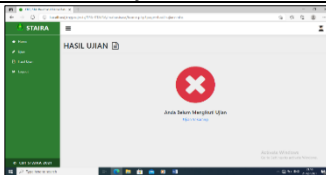

Tabel 10. Black Box Testing Login Mahasiswa

Skenario Pengujian	Uji Kasus
Mahasiswa salah memasukan nim, password atau captcha.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan gagal.	

3.5.8 Ujian

Berikut ini pengujian *black box* untuk fungsi *ujian* mahasiswa.

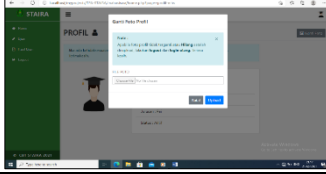
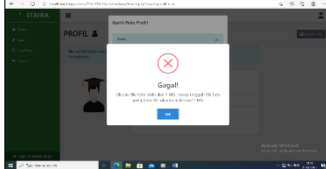
Tabel 11. Black Box Testing Ujian

Skenario Pengujian	Uji Kasus
Mahasiswa belum pernah mengikuti ujian sebelumnya.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem menampilkan halaman informasi ujian dan selanjutnya mahasiswa dapat mengikuti ujian.	

3.5.9 Ganti Foto Profil Mahasiswa

Berikut ini pengujian *black box* untuk fungsi *ganti* foto profil mahasiswa.

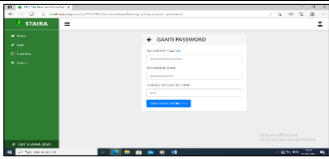
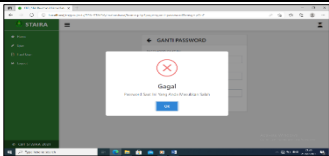
Tabel 12. Black Box Testing Ganti Foto Profil Mahasiswa

Skenario Pengujian	Uji Kasus
Mahasiswa melakukan upload foto profil dengan ukuran yang melebihi 1Mb.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan pemberitahuan.	

3.5.10 Ubah Password

Berikut ini pengujian *black box* untuk fungsi ganti ubah password.

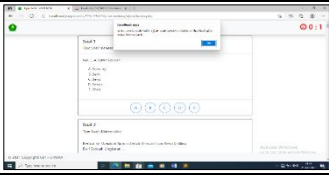
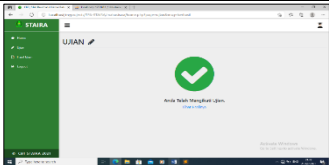
Tabel 13. *Black Box Testing* Ubah Password

Skenario Pengujian	Uji Kasus
Mahasiswa atau admin salah memasukan password lama dan password baru. Tidak sama.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem otomatis menolak dan menampilkan pesan pemberitahuan.	

3.5.11 Waktu Ujian Habis

Berikut ini pengujian dan hasil dari *black box testing* untuk fungsi waktu ujian mahasiswa jika sudah habis. Apabila waktu habis, sistem akan otomatis mengirim jawaban yang telah dipilih oleh mahasiswa. Hasil pengujian dapat dilihat pada table di bawah ini.

Tabel 14. *Black Box Testing* Waktu Ujian Habis

Skenario Pengujian	Uji Kasus
Mahasiswa melakukan ujian hingga batas waktu yang telah ditentukan.	
Hasil Yang Diharapkan	Hasil Pengujian
Sistem menampilkan pesan pemberitahuan dan secara otomatis mengirim jawaban dan memproses penilaian.	

4. Kesimpulan

Berdasarkan penelitian yang telah dilalui dalam tahap pembuatan aplikasi *Computer Based Test* (CBT) untuk ujian tes potensi akademik mahasiswa baru dengan metode *Linear Congruent Generator* (LCG) berbasis web, maka dapat diambil kesimpulan bahwa, dalam sebuah aplikasi ujian berbasis komputer, dapat diterapkan salah satu metode pembangkit bilangan acak semu yaitu *Linear Congruent Generator* (LCG). Sehingga aplikasi yang dihasilkan mampu menampilkan soal secara acak bagi tiap peserta ujian. Kemudian dalam merancang aplikasi *Computer Based Test* (CBT) mode terkontrol untuk ujian tes potensi akademik mahasiswa baru dengan metode *Linear Congruent Generator* (LCG) dilakukan tahap pertama yaitu menganalisa permasalahan dan penyelesaiannya dengan metode yang diterapkan setelah itu melakukan perancangan terhadap sistem. Penerapan sistem yang telah dirancang menggunakan *web programming* yaitu

dengan memasukkan data-data sesuai dengan yang ada, kemudian menuliskan *coding* pada *text editor* sesuai dengan konsep dan metode yang digunakan. Setelah itu dapat melakukan pengujian dengan mencoba menjalankan hasil kode program yang telah dibuat dengan menggunakan *local server* XAMPP melalui *browser* (Chrome, Mozilla Firefox, Microsoft Edge dan lain sebagainya) atau dengan menggunakan *hosting* agar dapat diakses melalui jaringan internet.

Untuk meningkatkan daya guna aplikasi ini kedepannya, diharapkan kepada peneliti-peneliti selanjutnya dapat menjadikan aplikasi ini lebih baik lagi seperti menjadikan aplikasi ini tidak hanya untuk ujian tes potensi akademik. Misalnya dapat digunakan untuk ujian semester, kuis, dan lain-lain. Menjadikan aplikasi ini mampu untuk mengembalikan ke situasi terakhir ketika peserta ujian menutup jendela *browser* serta melengkapi aplikasi ini dengan algoritma keamanan data yang baik.

Daftar Rujukan

- [1] S. N. Edusainstek *et al.*, "Aplikasi Smart Try Out System Berbasis Komputer," *Semin. Nas. Edusainstek FMIPA UNIMUS 2018*, pp. 220–225, 2018.
- [2] P. D. A. Pamungkas, "Computer Based Test (CBT) Pada Sekolah Tinggi Tarakanita Jakarta Menggunakan Metode Computerized Fixed-Form Test (CFT)," *J. Ilm. Teknol. Inf. Terap.*, vol. IV, no. 1, pp. 54–61, 2017.
- [3] M. P. Saptono and H. Widjasena, "Perancangan Dan Implementasi Aplikasi Ujian Sekolah Berbasis Komputer Atau Computer Based Test (Cbt) Di Smk Negeri 1 Kabupaten Sorong," *Electro Luceat*, vol. 5, no. 2, pp. 5–13, 2019, doi: 10.32531/jelekn.v5i2.148.
- [4] E. Setiawan, D. Antoni, and A. H. Mirza, "Analisis Penerimaan Sistem Ujian Online Berbayar Dengan Menggunakan Metode Technology Acceptance Model (Tam) Dan Webqual," *J. Bina Komput.*, vol. 1, no. 1, pp. 61–72, 2019, doi: 10.33557/binakomputer.v1i1.155.
- [5] F. Wicaksono, A. H. S. Nugraha, and R. Taufiq, "Aplikasi Tes Potensi Akademik Menggunakan Metode Anp Untuk Tes Penerimaan Mahasiswa Baru Berbasis Web," *J. Rekayasa, Teknol. dan Sains*, vol. 3, no. 1, pp. 13–20, 2019.
- [6] Rezki, A., Wijaya, I. G. P. S., & Akbar, S. (2022). *Sistem informasi pengarsipan surat berbasis*. 3(2), 166–176.
- [7] Dewi, R., & Sundari, J. (2021). *Sistem Informasi Pengelolaan Data Alumni Berbasis Web Pada SMK Bina Mandiri 2 Sukabumi Web-Based Alumni Data Management Information System for SMK Mandiri 2 Sukabumi*. 25–30.
- [8] Setiawansyah, S., Lestari Titian, D., & Megawaty Ayu, D. (2022). *Sistem Informasi Pkk Berbasis Website Menggunakan Framework Codeigniter (Studi Kasus : Kampung Purwoejo)*. 3(2), 244–253.
- [9] Yandri. (2022). *Perancangan Dan Implementasi Sistem Informasi Laporan Keuangan Apotek Jati Jaya Dengan Menggunakan Bahasa Pemrograman PHP Dan Database MySQL*. 114–135.
- [10] Sitanggang, R., Dachi, T. U., Manurung, I. H. G., Studi, P., Informasi, S., Sari, U., & Medan, M. (2022). *Rancang Bangun Sistem Penjualan Tanaman Hias Berbasis Web Menggunakan Php Dan Mysql*. 4(1), 84–90.
- [11] Kharisma, D., Saniati, S., & Neneng, N. (2022). *Aplikasi E-Commerce Untuk Pemesanan Sparepart MotOR*. 3(1), 83–89.
- [12] Desyani, T., Mulyati, S., Kurnianto, E., Afifah, N., Nur, S., & Fauziah, I. (2022). *Pengujian Black Box menggunakan teknik*
- [13] *Equivalence Partitions pada Aplikasi Sistem Pemilihan Karyawan Terbaik*. 5(2), 110–114. <https://doi.org/10.32493/jtsi.v5i2.17578>