

Implementasi Algoritma *Random Number Generator* (RNG) pada Aplikasi *Try Out* Berbasis Framework Laravel

M. Ridho Wahyudi Rachmadtulah¹, Agus Prihanto²

^{1,2}Jurusan Teknik Informatika, Universitas Negeri Surabaya

¹ridho.17051204064@mhs.unesa.ac.id

²agusprihanto@unesa.ac.id

Abstrak— Pada saat ini penggunaan Ujian Tulis Berbasis Komputer (UTBK) atau yang biasa dikenal juga dengan Computer Base Test (CBT) seringkali digunakan terutama untuk melakukan pelaksanaan Ujian Nasional. Sedangkan disisi lain pelaksanaan ujian tengah semester atau akhir semester sekolah masih menggunakan konsep konvensional yaitu satu soal untuk semua siswa dan dikerjakan secara manual di atas kertas. Agar siswa tidak kaget terhadap bentuk soal UTBK, maka pihak sekolah harus menyiapkan aplikasi yang cara kerjanya mirip dengan UTBK nasional untuk melatih para siswa-siswinya dalam menghadapi ujian tersebut. Pada penelitian ini memiliki beberapa tujuan 1). menerapkan algoritma *Random Number Generator* (RNG) pada soal *try out* dengan menggunakan framework laravel, 2). mengetahui tingkat dari kerandoman soal siswa dengan cara menghitung frekuensi kemunculan soal (modus) yang didapatkan oleh peserta.

Hasil dari Pengujian Validitas Aplikasi oleh Guru dan Siswa menunjukkan bahwa aplikasi telah di uji cobakan sesuai dengan harapan, sedangkan untuk pengujian keacakan soal diperoleh hasil rata-rata soal muncul yang sama adalah 15,25% dan frekuensi kemunculan soal (modus) adalah 25%..

Kata Kunci - *Tryout, CBT, RNG, Laravel, Modus, Laravel*

I. PENDAHULUAN

Teknologi informasi atau ilmu komputer selalu dilibatkan dalam institusi pendidikan. Pada era pesatnya teknologi, masih banyak yang tertinggal dari berbagai lini khususnya ilmu komputer, seperti pengimplementasian pendidikan untuk pembekalan materi pada sistem ujian yang menggunakan komputer.

Sebuah ujian atau test juga dibutuhkan pada setiap pembelajaran guna mengetahui sejauh mana tingkat siswa memahami pembelajaran yang telah diberikan [1]. Hal – hal seperti ini mendefinisikan bahwa ujian adalah sebuah uji coba yang dilakukan guna mengetahui ada atau tidaknya hasil-hasil pembelajaran yang telah dilalui oleh siswa. Hal ini membuktikan bahwa *tryout* bisa untuk mengukur sejauh mana pemahaman hasil belajar siswa.

Siswa dapat dikatakan berhasil atau menguasai materi berdasarkan nilai akhir yang telah diperoleh. Berdasarkan kemajuan jaman dan pesatnya teknologi dan informasi, *Computer Based Test* (CBT) atau yang biasa dikenal sebagai penilaian dengan menggunakan komputer mulai ditetapkan pada hasil akhir atau output belajar siswa. Penetapan hasil akhir CBT *tryout* mempunyai kelebihan

dibanding ujian menggunakan kertas atau yang biasa disebut dengan Paper Best Test (PBT) diantaranya :tingkat efisiensi dan efektifitas dalam estimasi waktu, anggaran dan energi yang digunakan[2].

Salah satu bentuk persiapan yang dilakukan oleh sekolah guna menghadapi Ujian Nasional (UN) adalah *Tryout*. Soal-soal *tryout* yang dibuat telah melalui berbagai macam pertimbangan sehingga dapat diimplementasikan saat akan menghadapi Ujian Nasional [3]. Sekolah dan siswa berharap bahwa hasil dari soal – soal *tryout* dapat menggambarkan soal – soal yang akan dihadapi saat Ujian Nasional nanti. *Tryout* juga digunakan sebagai indikator tingkat kematangan dan kesiapan siswa dalam mengikuti Ujian Nasional. *Tryout* juga sebagai metode pelatihan kepada siswa sebelum menghadapi Ujian Nasional.

Berdasarkan hasil observasi salah satu guru di SMK Negeri 1 Banyuwangi, metode ujian Paper Best Test atau yang biasa disebut PBT perlahan ingin dirubah menjadi system ujian menggunakan komputer atau CBT. Ada beberapa keuntungan yang diperoleh dari penggunaan CBT, antara lain ialah kecepatan, ketepatan serta kemudahan saat proses penilaian. Pengawas tidak lagi memeriksa secara manual pada lembar ujian siswa kemudian menghitung nilai, melainkan nilai sudah ditampilkan secara otomatis dalam aplikasi yang menggunakan basis data dan berdasarkan jawaban dari siswa, dan siswa akan mendapatkan soal yang telah teracak atau *random* sehingga siswa satu dengan siswa yang lainnya akan mendapatkan soal yang berbeda setiap nomornya. Pengurangan penggunaan kertas juga bentuk dari pengurangan kecurangan yang dilakukan siswa. Waktu dan tempat ujian dapat diatur sedemikian rupa sehingga mempersulit bagi siswa untuk melaksanakan kecurangan.

Menurut Munthe (2014), Bilangan yang tidak dapat diperkirakan hasilnya adalah suatu bilangan acak yang berasal dari sebuah metode. Operasi aritmatika yang biasa disebut dengan *pseudo-random number generator* dapat menghasilkan bilangan acak yang digunakan pada sebuah komputer. Ada berbagai macam metode yang sudah di implementasikan guna menghasilkan bilangan yang acak, diantaranya adalah : *Linear Congruent Method* Algoritma *Fisher-Yates Shuffeldan* [4].

Saat dilaksanakannya ujian perlu dilihat tindak kecurangan yang dapat di lakukan oleh siswa. Kesamaan soal pada tiap siswa menjadi salah satu faktor terjadinya tindakan kecurangan yang dilakukan oleh siswa. [5]. Menerapkan soal yang teracak adalah salah satu Tindakan untuk mencegah dilakukannya kecurangan pada saat try out.

Menghasilkan urutan t (*sequence*) adalah angka dari hasil perhitungan komputer terdistribusi sehingga angka tersebut keluar secara acak dan dipakai terus – menerus , angka angka ini merupakan suatu algoritma dari metode *Random Number Generator* (RNG) [6]. Penerapan algoritma *random number generator* sangat relevan digunakan dalam perancangan aplikasi ini dikarenakan soal tryout akan teracak secara otomatis. Algoritma *Random Number Generator* (RNG) dapat digunakan dalam sistem pengerjaan aplikasi tryout untuk mengacak nomor soal pada tryout. Algoritma RNG memiliki keunggulan yaitu pada waktu saat proses pengacakan soal karena hanya butuh sedikit operasi bit sehingga para siswa akan memperoleh soal yang berbeda satu dengan yang lain.

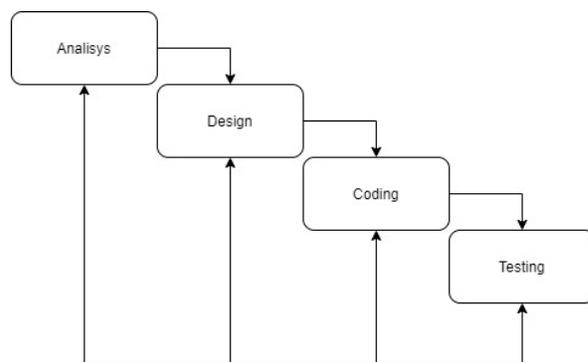
Metode yang digunakan untuk mengacak soal kali ini adalah *Random Number Generator*. Metode *Random Number Generator* menghasilkan bilangan acak melalui variabel-variabel yang dibangkitkan dengan algoritma kriptografi [7]. *Linear Congruential Generator* (LCG) adalah salah satu dari sekian banyak metode pada *Random Number Generator*. Algoritma LCG adalah algoritma yang mempunyai sifat rekursif linear yang digabungkan dengan fungsi modulus [8]. Penelitian ini diharapkan dapat mengurangi tindakan kecurangan pada *tryout* dengan menggunakan sistem pengacakan soal dengan memodifikasi metode *Linear Congruential Generator* (LCG).

Berdasarkan gagasan diatas, maka penelitian ini bertujuan 1). menerapkan algoritma Random Number Generator (RNG) pada soal try out dengan framework laravel, 2). mengetahui tingkat kerandoman soal siswa dengan menghitung frekuensi kemunculan soal (modus) yang didapatkan peserta.

II. METODE PENELITIAN

Dalam penelitian ini pengembangan sistem menggunakan model waterfall. Dalam tahapan model *waterfall* terdapat beberapa tahapan yaitu tahapan sistem analysis, design, coding, testing, dan support/maintenance .

Pemodelan pengembangan sistem dengan model tahapan *waterfall* dapat digambarkan seperti Gbr 1 dibawah ini :



Gbr. 1 model tahapan *waterfall*

A. Analysis

Pada tahap ini bertujuan untuk mengorganisasi dan menganalisa kebutuhan pengguna (sebelum diterapkan aplikasi yang baru).

Implementasi aplikasi dapat memberi kemudahan bagi pengguna aplikasi untuk mencari informasi komponen – komponen yang terdapat pada komputer. Perangkat lunak (*software*) dan perangkat keras (*hardware*) adalah perangkat pendukung proses implementasi yang memiliki spesifikasi minimum sebagai berikut .:

1. Spesifikasi Perangkat Keras (*Hardware*)

Spesifikasi minimum yang akan digunakan pada perangkat keras (*hardware*) yaitu seperti pada Tbl 1 :

TABEL I
SPESIFIKASI PERANGKAT KERAS

No	Perangkat Keras	Keterangan
1	Processor	2.10 GHz processor (recommended)
2	Memory RAM	4 GB of RAM (recommended)
3	Memory Hardisk	1 terabyte
4	Monitor	Standar
5	Keyboard	Standar
6	Mouse	Sandar

2. Spesifikasi Perangkat Lunak (*Software*)

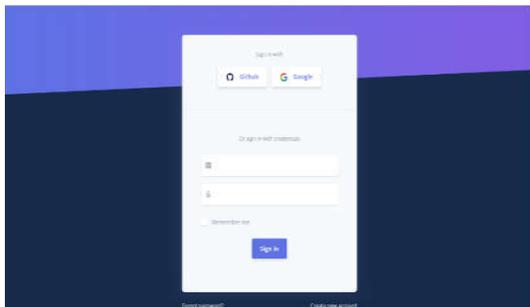
Spesifikasi yang digunakan dalam implementasi aplikasi tryout yang digunakan yaitu seperti pada tbl 2 :

TABEL III
SPESIFIKASI PERANGKAT LUNAK

No	Perangkat Lunak	Keterangan
1	Sistem Operasi	Windows 7 32/64 bit
2	Aplikasi	XAMPP, MySQL Database
3	Browser	Internet explore, Google Crhome, Mozilla Firefox

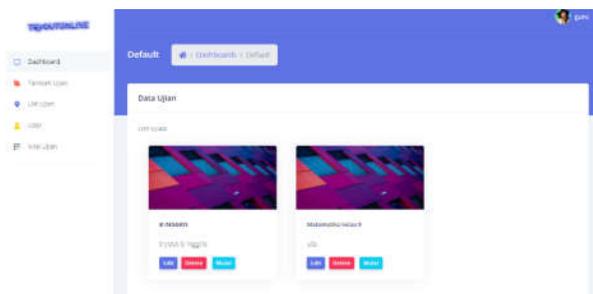
B. Design

Bertujuan untuk menentukan komponen – komponen aplikasi dan spesifikasi detail dari (data, manusia, *software*, *hardware*, *network*) dan produk-produk informasi yang sesuai dengan hasil tahapan analisis. Berikut adalah hasil interface yang dihasilkan :



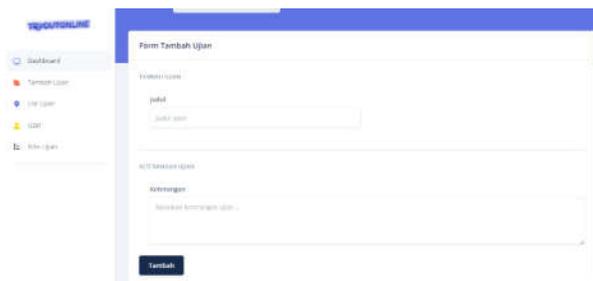
Gbr. 2 Halaman login

Pada gambar 2 halaman login user harus menginputkan email dan password yang sudah terdaftar. Apabila belum terdaftar maka harus menghubungi guru untuk mendaftarkan email dan password agar bias terdaftar ke dalam sistem



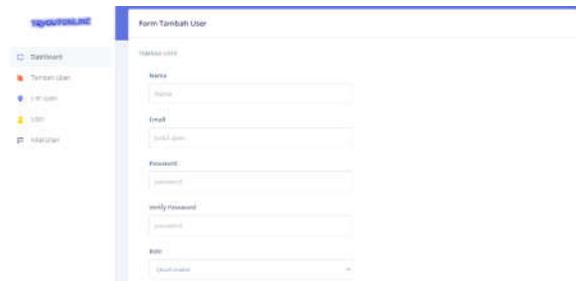
Gbr. 3 data ujian

Pada gambar 3 halaman data ujian terdapat daftar daftar mata pelajaran yang diujikan, disini guru atau pembuat ujian dapat memberikan penjelasan tentang ujian ujian yg diberikan. Ujian yang diberikan biasa berupa UTS UAS atau try out



Gbr. 4 form tambah ujian

Pada Gambar 4 form tambah ujian terdapat judul dan keterangan, form tambah ujian hanya dapat diakses oleh guru atau admin. Judul yang dimaksud disana menjelaskan tentang mata pelajaran yang akan diujikan sedangkan keterangan disana menjelaskan tentang keterangan ujian yang diberikan



Gbr. 5 halaman tambah user

Halaman ini adalah halaman yang hanya dapat diakses oleh guru atau admin. Halaman ini bersangkutan dengan halaman login, disini guru atau admin mendaftarkan email dan password siswa agar bisa tercantum ke system, jika email dan password siswa belum tercantum ke system maka siswa tersebut tidak dapat login

#	NAME	EMAIL	ROLE	ACTION
1	admin	admin@admin.com	quizcreator	Delete
2	brisia@jodie.com	brisia@jodie.com	user	Delete
3	siswa	siswa@siswa.com	user	Delete
4	fahmi	fahmi@gmail.com	user	Delete
5	kucil@gmail.com	kucil@gmail.com	user	Delete
6	guru	guru@guru.com	quizcreator	Delete
7	ridho	ridho@siswa.com	user	Delete

Gbr. 5.1 halaman tambah user

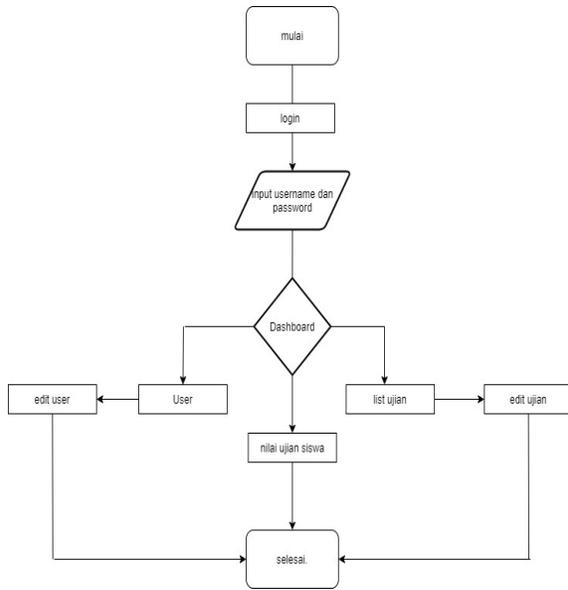
Halaman ini berisikan data data nama dan email berdasarkan status user yang telah terdaftar kedalam system, jadi guru dan admin dapat mengetahui siapa saja yang telah terdaftar dan belum mendaftarkan email dan passwordnya



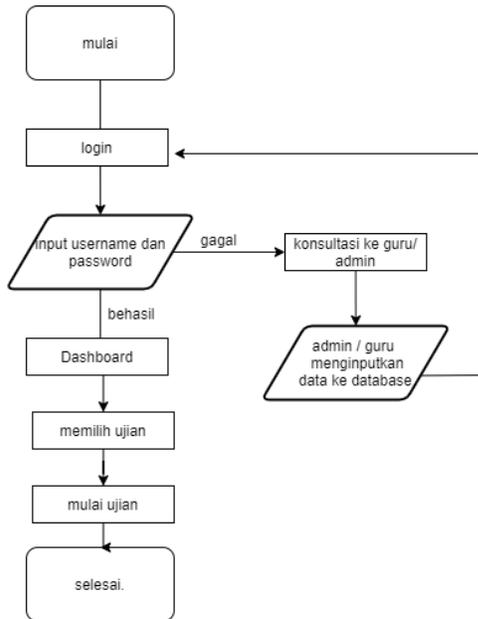
Gbr. 6 Halaman nilai

Pada halaman ini guru dan admin dapat melihat nilai siswa dan mata pelajaran yang telah dikerjakan.

Flowchart pada aplikasi kali ini dibagi menjadi menjadi dua, ada flowchart pada peserta ujian dan flowchart pada admin atau guru. Diagram alur atau flowchart pada siswa dan admin atau guru dapat dilihat pada gambar 7 dan 8.



Gbr.7 Diagram Alur pada guru dan admin



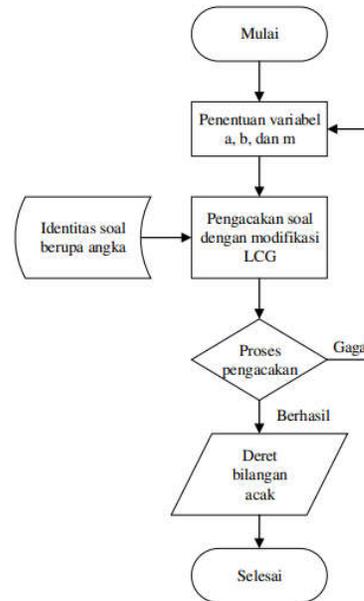
Gbr.8 Diagram Alur pada siswa

C. CODING

Desain yang telah dikerjakan diimplementasikan menggunakan Bahasa pemrograman yang sudah dipilih. Hal ini bertujuan untuk mengembangkan software atau pembentukan sistem pengacakan pada aplikasi *tryout*.

Tahap awal dalam pembuatan metode soal yang acak adalah menetapkan nilai variabel yang akan digunakan dalam metode modifikasi *Linear Congruential Generator* (LCG). Variabel a, b dan m adalah variabel yang harus ditentukan. Harus diketahui terlebih dahulu tingkat perbedaan pada pola yang akan dihasilkan dalam algoritma modifikasi LCG agar dapat diterapkan pada variabel m. Adapun 10, 20, 30, 40 adalah berbagai variasi variable m yang nilai-nilainya mewakili jumlah soal yang digunakan

pada ujian *tryout*. Bilangan prima digunakan sebagai Variabel b, fibonacci dan koprime sebagai pembatas jumlah variabel. Pola kesamaan soal dapat diperoleh dari hasil pengacakan soal menurut bilangan yang dipakai saat variabel b.. Diagram alur perancangan sistem dapat dilihat pada Gbr. 9 dibawah ini :



Gbr.9 Diagram Alur perancangan sistem

Gambar 9. dapat dilihat bahwa proses pengacakan soal dengan modifikasi LCG dilakukan setelah penentuan identitas soal yang akan digunakan dan penentuan variabel a,b dan m. Apabila terjadi kegagalan selama proses pengacakan soal maka Sistem akan meminta penentuan ulang variabel a,b dan m. Deret bilangan yang merupakan identitas masing-masing soal adalah hasil dari pengacakan soal adalah.

Algoritma modifikasi LCG yang dipakai sebagai model matematis dalam sistem pengacakan soal adalah hasil dari Penurunan algoritma LCG. Menentukan deret angka menggunakan Metode LCG sedangkan menggunakan deret yang akan dimasukkan ke dalam matrik menggunakan metode modifikasi dari LCG. Pada persamaan (1) merupakan model matematis algoritma LCG [9] :

$$x_{i+1} = (ax_i + b) \text{mod } m \quad (1)$$

dijelaskan bahwa x_{i+1} adalah bilangan yang acak pertama dari deretnya, variabel pengali adalah a, bilangan acak sebelumnya adalah x_i , variabel penambah adalah b dan variabel modulus adalah m. Persamaan (1) nantinya diturunkan menjadi matrik menggunakan orde (x, y) . Persamaan (2) dapat dipakai untuk mencari orde x_i pada baris matrik [10] :

$$x_{i+1} = (a_1x_i + b_1) \text{mod } m \quad (2)$$

dan model matematis bertujuan menentukan orde y sebagai kolom matrik yang akan memakai persamaan (3) :

$$y_{i+1} = (a_2x_i + b_2) \text{mod } m \quad (3)$$

Bilangan acak orde x ke-i dari deretnya x_{i+1} , Bilangan acak orde y ke-i dari deretnya adalah y_{i+1} ,

bilangan acak sebelumnya adalah xi, bilangan acak sebelumnya pada orde y adalah orde x dan yi .

Model matematis modifikasi LCG hasil turunan yang berasal dari model matematis LCG yang disamakan dengan matrik yang digunakan. apabila matrik M adalah matrik yang dipakai saat sistem pengacakan soal maka pembuatan matrik M bisa disimpulkan dengan memakai persamaan (4) seperti berikut [11] :

$$M_i = M[x_{i+1} \text{ mod } p][y_{i+1} \text{ mod } q] \quad (4)$$

dimana p merupakan jumlah baris matrik, q merupakan jumlah kolom matrik dan i merupakan bilangan cacah.

Dari rumus-rumus LCG diatas maka diimplementasikan pada codingan sebagai berikut :

```

1 <!--
2 namespace App\Helpers;
3 use Illuminate\Support\Facades\DB;
4 class Mf
5 {
6     protected $state = [];
7     protected $index = 025;
8     function init($seed)
9     {
10         $state = [$seed & 0xffffffff];
11
12         $int0 = $seed & 0xffff;
13         $int1 = ($seed >> 16) & 0xffff;
14
15         for ($i = 1; $i < 624; $i++) {
16             // This is a 32-bit safe version of:
17             // $state[$i] = (1812433253 * ($state[$i - 1] * ($state[$i - 1] >> 30)) + $i) & 0xffffffff;
18             $int0 ^= $int1 >> 14;
19
20             $carry = (0x9905 * $int0) + $i;
21             $int1 = ((0x0005 * $int1) + (0x0c07 * $int0) + ($carry >> 16)) & 0xffff;
22             $int0 = $carry & 0xffff;
23
24             $state[$i] = ($int1 << 16) | $int0;
25         }
26
27         $this->state = $state;
28         $this->index = $i;
29     }
30 }

```

Gbr.10 implemetasi rumus LCG

```

31 protected function twiist($u, $v, $w)
32 {
33     $y = ($u & 0x80000000) | ($v & 0x7fffffff);
34     return $w ^ (($y >> 1) & 0x7fffffff) ^ (0x9908b0df * ($v & 1));
35 }
36
37 function int32()
38 {
39     if ($this->index >= 624) {
40         if ($this->index == 625) {
41             $this->init(5489);
42         }
43
44         $state = $this->state;
45         for ($i = 0; $i < 227; $i++) {
46             $state[$i] = $this->twiist($state[$i + 397], $state[$i], $state[$i + 1]);
47         }
48         for (; $i < 623; $i++) {
49             $state[$i] = $this->twiist($state[$i - 227], $state[$i], $state[$i + 1]);
50         }
51         $state[623] = $this->twiist($state[396], $state[623], $state[0]);
52         $this->state = $state;
53         $this->index = 0;
54     }
55
56     $y = $this->state[$this->index++];
57
58     $y ^= ($y >> 11) & 0x001fffff;
59     $y ^= ($y << 7) & 0x9d2c5680;
60     $y ^= ($y << 15) & 0xefc60000;
61     $y ^= ($y >> 18) & 0x00003fff;

```

Gbr.11 implemetasi rumus LCG

D. Testing

Setelah kode dihasilkan, program pengujian dimulai. Proses pengujian berfokus dengan dapat diterapkannya algoritma RNG pada soal tryout serta metode pengacakan aplikasi tryout sehingga dapat diketahui tingkat frekuensi kerandoman soal pada tiap siswa.

III. HASIL DAN PEMBAHASAN

Sistem uji ini dilakukan dengan mencoba beberapa kemungkinan yang akan terjadi dan system uji ini menggunakan pengujian Validatas Aplikasi untuk Guru dan Siswa dan pengujian keacakan soal. Apabila terjadi

kesalahan saat pengujian, makan akan dilakukan penelusuran dan perbaikan guna memperbaiki masalah yang terjadi.

A. Pengujian Validitas Aplikasi

Tahap pengujian ini dibuat guna mengetahui apakah Penerapan Algoritma *Random Number Generator* (RNG) pada *Try Out* Berbasis Framework Laravel bisa berjalan baik tanpa adanya masalah.

1) Pengujian Validitas untuk Guru

TABEL IIIII
PENGUJIAN VALIDITAS GURU

No	Skenario Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	Memilih menu login	Menu login berhasil ditampilkan	Sesuai harapan	Valid
2	Memilih menu dashboard	Menu dashboard ditampilkan	Sesuai harapan	Valid
3	Memilih menu tambah ujian	Menu tambah ujian berhasil ditampilkan	Sesuai harapan	Valid
4	Memilih menu tambah user	Menu tambah user berhasil ditampilkan	Sesuai harapan	Valid
5	Memilih menu edit user	Menu edit user berhasil ditambahkan	Sesuai harapan	Valid
6	Memilih menu nilai ujian	Menu nilai ujian berhasil ditambahkan	Sesuai harapan	Valid
7	Memilih menu sign out	Menu sign out berhasil ditambahkan	Sesuai harapan	Valid

2) Pengujian Validitas untuk Siswa

Tabel IV
Pengujian Validitas untuk Siswa

No	Skenario Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
1	Memilih menu login	Menu login berhasil ditampilkan	Sesuai harapan	Valid
2	Memilih menu dashboard	Menu dashboard ditampilkan	Sesuai harapan	Valid
3	Memilih menu ujian	Menu ujian berhasil ditampilkan	Sesuai harapan	Valid
4	Memilih menu sign out	Menu sign out berhasil ditambahkan	Sesuai harapan	Valid

Dari 2 pengujian validitas untuk Guru dan siswa diperoleh informasi bahwa aplikasi yang telah dikembangkan Valid sesuai harapan Guru dan Siswa.

B. Hasil Pengujian Keacakan Soal

Pada pengujian ini dilakukan dengan membuat bank data soal sebanyak 20 buah yang diujikan ke 20 komputer siswa. Kemudian soal tersebut akan

dibagikan ke komputer siswa setelah mereka berhasil login ke aplikasi secara acak dengan algoritma RND (Random Number Generator). Berikut adalah rekap bank soal yang diperoleh oleh 20 siswa :

TABEL V
REKAP BANK SOAL YANG DIPEROLEH OLEH 20 SISWA

No	Nama Siswa (Y)	Urutan Soal (X)																			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Ahmad Wafa	5	14	7	1	6	13	11	20	19	15	3	4	8	9	10	2	16	17	12	18
2	Audy Yanuar	2	6	9	15	13	5	4	20	18	12	17	1	3	19	11	7	8	14	10	16
3	Annas Itaquillah	7	19	5	10	11	6	2	8	17	13	15	3	9	4	18	1	12	14	16	20
4	Isyroqi Hadyan N.	6	8	1	18	20	16	10	5	14	3	11	13	2	9	4	19	17	12	7	15
5	Rayhan Reynara	9	4	20	5	10	18	12	19	14	6	11	2	16	15	17	1	13	7	8	3
6	Dzikri Hisyam	19	7	12	14	5	16	15	17	8	18	4	6	2	10	11	9	20	1	13	3
7	Siska Oktavia	20	1	15	16	3	11	13	19	17	10	9	14	2	8	6	4	12	18	5	7
8	Bramianto Gading G.	17	12	8	2	11	5	20	18	4	14	19	10	3	13	15	9	7	16	6	1
9	ErikKanya A.	19	5	18	16	6	17	10	11	15	14	7	3	4	20	1	12	8	13	9	2
10	Atika Adawiyah	16	3	14	11	5	7	8	15	19	18	4	6	20	2	13	17	9	10	12	1
11	Elok Nur F.	2	9	14	17	3	10	4	5	12	11	13	16	6	20	15	1	7	8	19	18
12	Iza Amalina	18	9	3	2	4	8	1	15	11	17	6	10	7	12	13	16	19	14	5	20
13	Yuan Shinta	1	16	2	18	3	20	11	6	7	17	13	10	14	8	19	15	12	5	9	4
14	Mahandika Dias S.	6	20	11	15	1	10	18	3	16	4	17	7	8	13	14	9	2	12	19	5
15	Firman Maulana	12	19	16	18	10	9	17	6	15	5	3	8	4	11	20	7	13	1	2	14
16	Jalis Dwi M.	10	2	13	18	1	14	19	3	4	20	9	7	15	11	16	5	6	8	12	17
17	Bagus Arief H.	4	9	2	15	8	11	16	14	17	18	12	10	6	20	3	7	1	19	13	5
18	Arief Rahman I.S.P	11	18	20	3	9	10	6	5	16	7	13	12	15	19	1	2	14	4	8	17
19	Arsy Bilahil Tama.	3	17	14	5	12	1	8	6	18	9	19	7	4	11	16	15	2	10	13	20
20	M. Ridho Wahyudi R.	8	9	6	7	16	2	4	15	20	13	3	10	5	12	1	19	17	14	18	11

Keterangan : Cara membaca Cell (Z)= Siswa baris Y pada urutan soal X mendapatkan bank soal no Z

Dari Tabel 5 diperoleh informasi bahwa setiap mahasiswa sudah mendapatkan soal yang bersifat acak, meskipun dalam urutan soal yang sama kadang memperoleh bank soal yang sama. Berikut adalah rekap nilai kemunculan bank soal (modus) yg diperoleh 20 siswa yang diolah dari Tabel 5 di atas :

TABEL VI
REKAP FREKUENSI KEMUNCULAN (MODUS) BANK SOAL YANG DIPEROLEH OLEH 20 SISWA

No Bank Soal (Y)	Urutan Soal (X)																				Kontrol
	1	2	3	4	5	6	7	8	9	0	1	1	1	1	1	1	1	1	1	2	
1	1	1	1	1	2	1	1	0	0	0	0	1	0	0	3	3	1	2	0	2	20
2	2	1	2	2	0	1	1	0	0	0	0	1	3	1	0	2	2	0	1	1	20
3	1	1	1	1	3	0	0	2	0	1	3	2	2	0	1	0	0	0	0	2	20
4	1	1	0	0	1	0	3	0	2	1	2	1	3	1	1	1	0	1	0	1	20
5	1	1	1	2	2	2	0	3	0	1	0	0	1	0	0	1	0	1	2	2	20
6	2	1	1	0	2	1	1	3	0	1	1	2	2	0	1	0	1	0	1	0	20
7	1	1	1	1	0	1	0	0	1	1	1	3	1	0	0	3	2	1	1	1	20
8	1	1	1	0	1	1	2	1	1	0	0	1	2	2	0	0	2	2	2	0	20
9	1	4	1	0	1	1	0	0	0	1	2	0	1	2	0	3	1	0	2	0	20

No Bank Soal (Y)	Urutan Soal (X)																				Kontrol
	1	2	3	4	5	6	7	8	9	0	1	1	1	1	1	1	1	1	1	2	
10	1	0	0	1	2	3	2	0	0	1	0	5	0	1	1	0	0	2	1	0	20
11	1	0	1	1	2	2	2	1	1	1	2	0	0	3	2	0	0	0	0	1	20
12	1	1	1	0	1	0	1	0	1	1	1	1	0	2	0	1	3	2	3	0	20
13	0	0	1	0	1	1	1	0	0	2	3	1	0	2	2	0	2	1	3	0	20
14	0	1	3	1	0	1	0	1	2	2	0	1	1	0	1	0	1	4	0	1	20
15	0	0	1	3	0	0	1	3	2	1	1	0	2	1	2	2	0	0	0	1	20
16	1	1	1	2	1	2	1	0	2	0	0	1	1	0	2	1	1	1	1	1	20
17	1	1	0	1	0	1	1	1	3	2	2	0	0	0	1	1	2	1	0	2	20
18	1	1	1	4	0	1	1	1	2	3	0	0	0	0	1	0	0	1	1	2	20
19	2	2	0	0	0	0	1	2	2	0	2	0	0	2	1	2	1	1	2	0	20
20	1	1	2	0	1	1	1	2	1	1	0	0	1	3	1	0	1	0	0	3	20
Kontrol	20	20	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Keterangan : Cara membaca Cell (Z) = No bank soal Y pada urutan soal X muncul sebanyak Z

Dari Tabel 6 di atas diperoleh informasi bahwa ada beberapa siswa yang mendapatkan no bank soal yang sama meskipun nilai kemunculannya tidak terlalu sering. Berikut adalah rekap perhitungan kemunculan soal tertinggi pada setiap no bank soal.

TABEL VII
REKAP NILAI KEMUNCULAN TERTINGGI PADA SETIAP NO BANK SOAL

No Bank Soal	Max	%
1	3	15
2	3	15
3	3	15
4	3	15
5	3	15
6	3	15
7	3	15
8	2	10
9	4	20
10	5	25
11	3	15
12	3	15
13	3	15
14	4	20
15	3	15
16	2	10
17	3	15
18	4	20
19	2	10
20	3	15
Rata-rata	3,1	15,5

Dari Tabel 7 di atas diperoleh informasi bahwa nilai kemunculan tertinggi bank soal pada nomor 10, yang mana ada 5 (25%) siswa yang mendapatkan nomor yang sama dari 20 siswa yang ikut ujian. Sedangkan rata-rata kemunculan bank soal tertinggi jika dinyatakan dalam persen sebesar 15,5%. Hal ini menunjukkan bahwa Algoritma RND yang digunakan pada aplikasi Tryout memiliki rata-rata kemunculan tertinggi sebesar 15,5%.

IV. KESIMPULAN

Berdasarkan hasil pengujian dan pembahasan pada bab sebelumnya, maka dapat ditarik kesimpulan sebagai berikut :

1. Pada pengujian validitas aplikasi diperoleh hasil bahwa Aplikasi Tryout berbasis algoritma Random Number Generator (RND) telah valid sesuai harapan Guru dan Siswa.
2. Pada pengujian keacakan soal diperoleh hasil bahwa nilai kemunculan tertinggi dari 20 bank soal ada pada soal nomor 10, yang mana ada 5 (25%) siswa yang mendapatkan nomor yang sama dari 20 siswa yang ikut ujian. Sedangkan rata-rata kemunculan bank soal tertinggi jika dinyatakan dalam persen sebesar 15,5%.

V. SARAN

Apabila ada penelitian yang sejenis, peneliti disarankan menggunakan metode random lainnya lalu di uji coba mana yang lebih baik.

VI. UCAPAN TERIMA KASIH

Terima kasih sebesar-besarnya kepada tuhan Allah S.W.T, kepada keluarga dan kepada diri saya sendiri karenanya tugas yang cukup berat ini dapat saya selesaikan, dan tidak luput juga saya ucapkan terimakasih kepada bapak Agus Prihanto karna telah bersedia membimbing saya dari awal hingga akhir.

VII. REFERENSI

[1] Arikunto, & Suharsimi. (2013). *Prosedur Penelitian Suatu Pendekatan Praktik*. Jakarta: Rineka Cipta.

- [2] S. Simin dan A. Heidari, "Computer-based Assessment: Pros and Cons", *Elixir International Journal*, vol. 55, pp. 12732-12734, 2003.
- [3] Dinata, R. K., Fikry, M., & Thahir, d. H. (2018, April). Implementasi Algoritma Multiplicative Congruential Random Number Generator Pada Try-Out Smp Berbasis Client Server. *TechSI*, Vol. 10, No. 1. doi: <https://doi.org/10.29103/techsi.v10i1.608>
- [4] Munthe, D. (2014). Implementasi Linier Congruent Method (LCM) pada Aplikasi Tryout SNMPTN (Studi Kasus : Bimbingan DanPemantapan Belajar Quin Medan). *PelitaInformatika Budi Darma*, 111-115.
- [5] D.L. Mc Cabe dan W.J. Bowers, "Academic Dishonesty Among Males in College : A Thirty Year Perspective", *Journal of College Student Development*, vol. 5, pp 5-10. 1994.
- [6] Riani, L. (2010). Pembangkit Bilangan Acak. In Mata Kuliah Pemodelan & Simulasi. Bandung: Jurusan Teknik Informatika Universitas Komputer Indonesia.
- [7] K.M. Martin, "Everyday Cryptography: Fudamental Principles and Applications", Oxford University Press Inc.,New York, pp. 253-259. 2012.
- [8] D. Salomon, "Data Privacy and Security", Springer-Verlag New York Inc., New York, pp. 97-100. 2003.
- [9] D.E. Knuth, "The Art of Computer Programming 2nd Ed.", AddisonWesley Publishing Company, Canada, pp. 9-16. 1981.
- [10] R.S. Katti, dkk, "Pseudorandom Bit Generation Using Coupled Congruential Generator", *IEEE Transactions on Circuits and System II :Express Briefs*, vol. 57, pp.203-207. 2010.
- [11] I Made D. Biantara, "Implementation Coupled Linear Congruential Generator Methods for Questions of Pattern Randomization",*International Conference on Vocational Education and Electrical Engineering (ICVEE)*, pp. 247-250, 2015.