

Analisis Perbandingan Penggunaan Fungsi Random Mysql dan Fungsi Random *Java Class Library* pada Aplikasi CBT (*Computer Based Test*)

Achmad Fauzan¹⁾, Tito Pinandita²⁾, Harjono³⁾

¹⁾²⁾³⁾*Teknik Informatika-F.Teknik-Universitas Muhammadiyah Purwokerto
Jl. Raya Dukuhwaluh Purwokerto 53182*

¹⁾*achmadfauzanfamily@yahoo.com*

Abstrak— *Computer Based Test (CBT)* merupakan suatu aplikasi yang digunakan dalam penyelenggaraan ujian berbasis computer dengan kelebihan dapat menyajikan pertanyaan dan pilihan jawaban secara acak sehingga menghindari kemungkinan terjadinya kecurangan yang dilakukan oleh peserta ujian. Penelitian ini melakukan pengujian fungsi random terhadap aplikasi CBT berbasis Java dan database server MySQL dengan objek perbandingan yaitu fungsi random MySQL dan fungsi random Java Class Library. Variabel yang digunakan dalam penelitian yaitu waktu akses pertanyaan dan kombinasi pertanyaan dan pilihan jawaban yang sama. Pengujian dilakukan dengan cara memodelkan fasilitas yang menampilkan halaman pertanyaan dan pilihan jawaban untuk mengetahui waktu akses munculnya pertanyaan dan kombinasi yang dihasilkan. Hasil perhitungan diuji melalui pengujian kesamaan rata-rata untuk mengetahui perbedaannya. Setelah dilakukan pengujian, diketahui bahwa terdapat perbedaan pada waktu akses pertanyaan menggunakan fungsi random MySQL dengan fungsi random Java Class Library. Rata-rata waktu akses pertanyaan dengan fungsi random MySQL sebesar 41650473.6250 nanosecond adalah lebih lama dibandingkan rata-rata waktu akses pertanyaan dengan fungsi random Java Class Library sebesar 35905823.7431 nanosecond. Namun kombinasi pertanyaan dan pilihan jawaban yang sama pada pertanyaan dengan fungsi random MySQL sebesar 63.6% adalah lebih kecil dibandingkan dengan penggunaan fungsi random Java Class Library yang mencapai persentase sebesar 65.1%.

Kata-kata kunci— *Computer Based Test, Fungsi Random, MySQL, Java Class Library, Uji Statistik*

Abstract— *Computer Based Test (CBT)* is an application that is used in the operation of computer-based test with the advantages that can present a question and answer choices were randomized to avoid possible fraud

committed by the examinee. This study tested a random function of CBT applications based on Java and MySQL database server with the object of comparison is the function of a random function random MySQL and Java Class Library. Variables used in this study is the access time and a combination of questions and question the same answer choices. Testing is done by way of modeling facilities that display the page question and answer choices to determine the access time of the emergence of the question and the resulting combination. The results of the calculations was tested by examining the average similarity to know the difference. After testing, it is known that there is a difference in access time question using MySQL random function with the function of random Java Class Library. Average access time with the question of random functions for MySQL 41650473.6250 nanosecond is longer than the average access time with the question of random functions for Java Class Library 35905823.7431 nanosecond. But the combination of questions and answer choices are similar to the question by MySQL random function of 63.6% is smaller than the random use of Java Class Library function that reaches a percentage of 65.1%.

Keywords— *Computer Based Test, Random Functions, MySQL, Java Class Library, Statistics Test*

I. PENDAHULUAN

Penyelenggaraan ujian masuk perguruan tinggi sebagian besar masih dilakukan secara manual yaitu diselenggarakan dengan menggunakan media kertas (paper-based) untuk menyajikan pertanyaan maupun menuliskan jawabannya. Metode penyelenggaraan ujian secara manual membutuhkan banyak waktu dalam penyajian pertanyaan dan tingkat ketelitian yang tinggi dalam mengevaluasi hasilnya. Petugas penyelenggara ujian dituntut menghasilkan laporan yang akurat dengan batas waktu tertentu. Keadaan seperti ini memungkinkan banyak terjadinya kesalahan manusia (*human error*)

baik dalam proses persiapan penyelenggaraan ujian hingga proses evaluasi hasil ujian.

Masalah yang terjadi dalam penyelenggaraan ujian secara manual dapat diatasi dengan mengembangkan metode penyelenggaraan ujian berbasis komputer. Metode yang digunakan yaitu memanfaatkan komputer untuk menyajikan pertanyaan dan menyimpan hasil ujian sekaligus memproses hasil ujian untuk dievaluasi secara otomatis dalam suatu sistem informasi. Sistem informasi yang digunakan harus dapat menyajikan pertanyaan dan pilihan jawaban secara acak untuk menghindari kemungkinan terjadinya kecurangan yang dilakukan oleh peserta ujian. Selain itu sistem informasi yang digunakan juga dituntut dapat melakukan evaluasi terhadap hasil ujian sehingga menghasilkan laporan-laporan yang akurat. Sistem informasi ini sering disebut dengan istilah Computer Based Test (CBT).

Computer Based Test (CBT) merupakan suatu aplikasi yang digunakan dalam penyelenggaraan ujian berbasis komputer. Banyak lembaga yang telah memanfaatkan aplikasi CBT pada pelaksanaan ujian seperti pada ujian TOEFL atau ujian saringan masuk mahasiswa baru. Semakin banyaknya lembaga yang memanfaatkan aplikasi CBT dalam penyelenggaraan ujian menunjukkan bahwa aplikasi ini telah memberi banyak keuntungan dari sisi efektifitas waktu, minimalisasi kecurangan peserta, dan kemudahan dalam pengelolaan hasil ujian. Salah satu kelebihan aplikasi CBT yaitu dapat menyajikan pertanyaan dan pilihan jawaban secara acak sehingga menghindari kemungkinan terjadinya kecurangan yang dilakukan oleh peserta ujian. Hal ini dapat dilakukan karena aplikasi CBT menggunakan metode acak pada setiap pertanyaan dan pilihan jawaban yang ditampilkan [1].

Metode acak yang digunakan pada aplikasi CBT dapat dikembangkan pada antar muka aplikasi atau pada akses basis data. Pemilihan metode acak harus didasarkan pada kriteria-kriteria yang berpengaruh pada kualitas aplikasi diantaranya yaitu pertimbangan kecepatan keluaran yang ditampilkan dan besar kecilnya kemungkinan munculnya pertanyaan dan kombinasi pilihan jawaban yang sama.

II. METODE

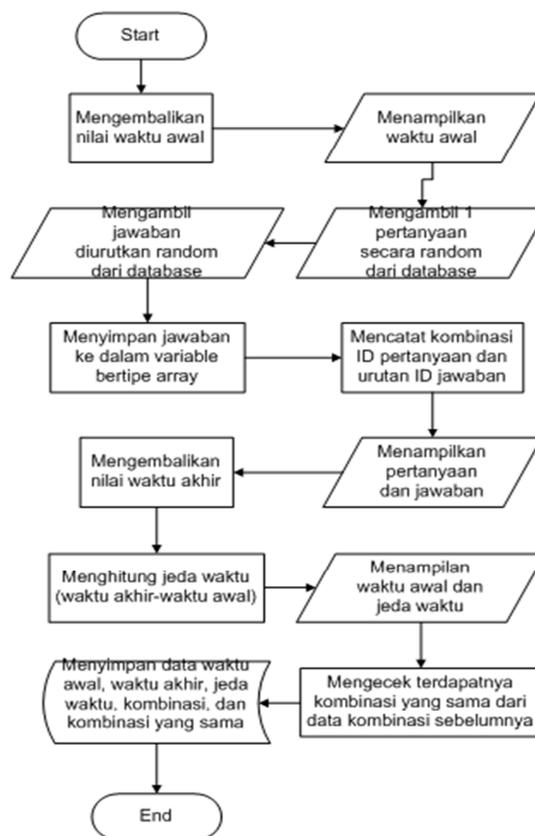
Langkah-langkah dalam melakukan penelitian ini yaitu:

1. Membuat model halaman pertanyaan yang tampil secara random bertujuan untuk

melakukan pengukuran waktu akses dan pencatatan kemunculan kombinasi pertanyaan dan pilihan jawaban yang sama.

Computer Based Testing (CBT) bekerja dengan menampilkan pertanyaan dan jawaban yang acak dengan tujuan untuk mengurangi kemungkinan terjadinya kecurangan yang dilakukan oleh peserta ujian. Fungsi inilah yang menjadi objek penelitian sehingga untuk melakukan penelitian dibuat terlebih dahulu model halaman pertanyaan aplikasi CBT yang dapat menampilkan pertanyaan dan pilihan jawaban secara random, baik menggunakan fungsi random MySQL [3] maupun fungsi random Java Class Library [2].

Diagram alir algoritma model halaman pertanyaan dengan fungsi random MySQL dapat dilihat pada Gambar 1 berikut.

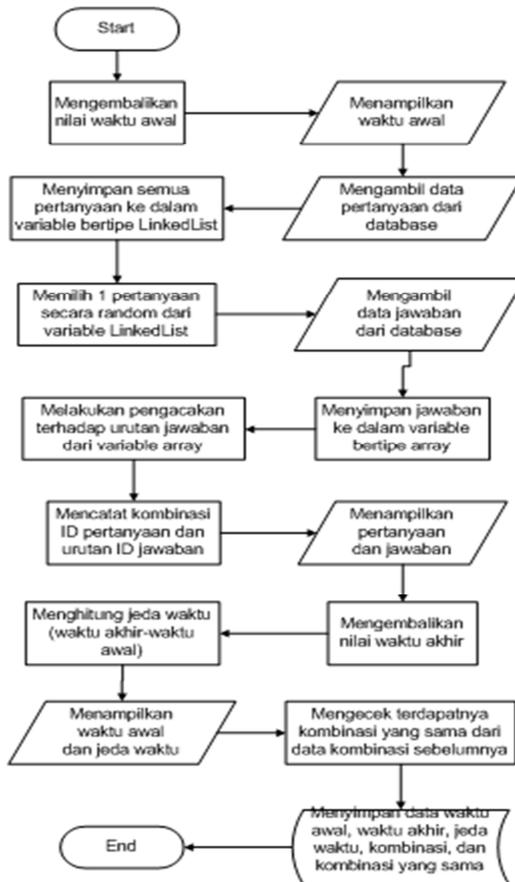


Gambar 1. Diagram Alir Algoritma Model Halaman Pertanyaan dengan Fungsi Random MySQL

Pada diagram yang ditunjukkan pada Gambar 1, ditunjukkan bahwa pertanyaan dan pilihan jawaban diambil secara acak langsung dari basis data pada proses pemanggilan pertanyaan dan

pilihan jawaban. Sehingga pertanyaan dan pilihan jawaban yang muncul merupakan hasil pengacakan dari keseluruhan pertanyaan dan pilihan jawaban yang tersedia.

Diagram alir algoritma model halaman pertanyaan dengan fungsi random Java Class Library dapat dilihat pada Gambar 2 berikut.



Gambar 2. Diagram Alir Algoritma Model Halaman Pertanyaan dengan Fungsi Random Java Class Library

Pencatatan waktu akses pertanyaan dan pilihan jawaban akan didasarkan pada lamanya sebuah pertanyaan dan pilihan jawaban yang telah diacak berhasil ditampilkan. Untuk melakukan pencatatan waktu tersebut, kode pencatat waktu disisipkan ke dalam halaman tersebut yaitu disisipkan sebelum pemanggilan pertanyaan dan setelah pilihan jawaban berhasil ditampilkan. Selisih waktu itulah merupakan waktu akses pertanyaan dan pilihan jawaban, kemudian seluruh data pencatatan waktu tersebut akan disimpan ke dalam basis data.

Pencatatan kombinasi pertanyaan dan pilihan jawaban dilakukan dengan mengambil nomor identitas unik dari pertanyaan dan pilihan jawaban yang tampil, kemudian dirangkai menjadi sebuah urutan kombinasi. Kombinasi baru dibandingkan terlebih dahulu dengan kombinasi lama yang telah tersimpan di dalam basis data. Apabila terdapat kombinasi yang sama maka nomor identitas unik akan ditandai, kemudian data kombinasi dan nomor identitas unik yang telah ditandai akan disimpan ke dalam basis data.

Berikut ini merupakan rancangan basis data yang digunakan untuk menyimpan data waktu akses dan kombinasi pertanyaan dan pilihan jawaban pada model halaman pertanyaan (Gambar 3).

| testpage_mysql | | testpage_javalib | |
|----------------|--------------------|------------------|--------------------|
| PK | <u>testpage_id</u> | PK | <u>testpage_id</u> |
| | start | | start |
| | end | | end |
| | showtime | | showtime |
| | combination | | combination |
| | comb_same_with | | comb_same_with |

Gambar 3. Rancangan Basis Data Model Halaman Pertanyaan

2. Membuat perintah berupa code untuk melakukan pengolahan data.

Pengolahan data akan dilakukan untuk mengetahui rata-rata waktu akses pertanyaan dan jawaban pada masing-masing halaman dengan fungsi random MySQL dan fungsi random Java Class Library. Selain itu juga akan dilakukan perhitungan banyaknya kombinasi pertanyaan dan pilihan jawaban yang sama.

3. Melakukan pengujian dan analisis data.

Data percobaan akan dilakukan beberapa pengujian, yaitu [5]:

- a. *Uji Normalitas data.* Pengujian ini dilakukan untuk mengetahui apakah data menyebar normal atau tidak, apabila data tidak menyebar normal maka uji-T dua sample independen tidak tepat diterapkan. Sehingga pengujian dilakukan dengan menggunakan statistik nonparametrik yaitu dengan uji Mann-Whitney sebagai alternatif uji-T dua sample independen.
- b. *Uji Mann-Whitney.* Uji Mann-Whitney disebut juga uji-U atau uji jumlah peringkat

Wilcoxon yang merupakan alternatif dari uji-T dua sample independen.

III. HASIL DAN PEMBAHASAN

Pengambilan data dilakukan pada sebuah komputer client dengan mengakses model halaman pertanyaan menggunakan aplikasi mesin peramban internet (browser) secara bergantian yaitu mengakses model halaman pertanyaan menggunakan fungsi random MySQL terlebih dahulu, dilanjutkan dengan mengakses model halaman pertanyaan menggunakan fungsi random Java Class Library sebanyak jumlah data yang diinginkan. Jumlah data yang diambil ditentukan berdasarkan banyaknya kemungkinan kombinasi antara pertanyaan dan pilihan jawaban yang tersedia. Dalam penelitian ini disediakan 30 buah pertanyaan berbeda dengan masing-masing pertanyaan disediakan 4 pilihan jawaban. Setiap pengaksesan model halaman pertanyaan, hanya akan muncul sebuah pertanyaan dan 4 pilihan jawaban yang susunan urutan akan selalu berbeda tanpa adanya pengulangan. Sehingga banyaknya data yang akan diambil merupakan hasil perkalian antara jumlah pertanyaan tersedia dengan permutasi tanpa pengulangan terhadap 4 pilihan jawabannya.

Perhitungan tersebut dapat dirumuskan seperti pada persamaan 1 berikut:

$$n = q \frac{c!}{(c-r)!} \dots\dots\dots 1)$$

- Keterangan: n: banyaknya data yang akan diambil
 q: banyaknya pertanyaan yang tersedia
 c: banyaknya pilihan jawaban yang tersedia
 r: banyaknya pilihan jawaban yang ditampilkan

Maka perhitungan banyaknya data yang akan diambil yaitu:

$$\begin{aligned} n &= q \frac{c!}{(c-r)!} \\ &= 30 \frac{4!}{(4-4)!} \\ &= 30 \frac{24}{0!} \\ &= 720 \end{aligned}$$

Setelah ditentukan banyaknya data yang akan diambil, kemudian dilakukan pengambilan data. Data hasil percobaan kemudian diolah untuk

mengetahui nilai rata-rata dan batas waktu akses yang didapatkan. Hasil pengolahan data tersebut merupakan rekapitulasi hasil pengambilan data waktu akses pertanyaan dan jawaban seperti pada Tabel I.

TABEL I
 REKAPITULASI HASIL PENGAMBILAN DATA WAKTU AKSES PERTANYAAN DAN JAWABAN

| Fungsi random | n | Rata-rata | Waktu terkecil | Waktu terbesar |
|---------------|-----|--------------|----------------|----------------|
| | | (nanosecond) | | |
| MySQL | 720 | 41650474 | 30043420 | 84701080 |
| JavaLib | 720 | 35905824 | 24353375 | 230311465 |

Berdasarkan Tabel I, diketahui bahwa model halaman pertanyaan dengan menggunakan fungsi random MySQL dapat diakses dalam waktu rata-rata 41650473.6250 nanosecond setiap pertanyaan. Dalam percobaan sebanyak 720 kali, didapatkan waktu akses pertanyaan dan jawaban terkecil yaitu 30043420 nanosecond dan waktu akses pertanyaan dan pilihan jawaban terbesar yaitu 84701080 nanosecond. Sedangkan model halaman pertanyaan dengan menggunakan fungsi random Java Class Library dapat diakses dalam waktu rata-rata 35905823.7431 nanosecond setiap pertanyaan. Dalam percobaan sebanyak 720 kali, didapatkan waktu akses pertanyaan dan jawaban terkecil yaitu 24353375 nanosecond dan waktu akses pertanyaan dan jawaban terbesar yaitu 230311465 nanosecond. Perhitungan banyaknya kombinasi pertanyaan dan pilihan jawaban yang sama pada hasil percobaan dapat ditunjukkan pada Tabel II.

TABEL II
 PERHITUNGAN BANYAKNYA KOMBINASI PERTANYAAN DAN PILIHAN JAWABAN SAMA

| Fungsi random | n | Kombinasi sama | Persentase |
|---------------|-----|----------------|------------|
| MySQL | 720 | 458 | 63.6% |
| JavaLib | 720 | 469 | 65.1% |

Berdasarkan Tabel II, diketahui bahwa dalam percobaan yang dilakukan sebanyak 720 kali pengaksesan model halaman pertanyaan dengan menggunakan fungsi random MySQL, terdapat kombinasi pertanyaan dan pilihan jawaban yang sama sebanyak 458 kali atau 63.6% dari seluruh percobaan. Sedangkan dalam percobaan yang dilakukan sebanyak 720 kali pengaksesan model halaman pertanyaan dengan menggunakan fungsi random Java Class Library, terdapat kombinasi

pertanyaan dan pilihan jawaban yang sama sebanyak 469 kali atau 65.1% dari seluruh percobaan.

Analisis hasil penelitian dilakukan untuk menguji kebenaran hipotesis yang diajukan dalam penelitian yaitu terdapat perbedaan waktu akses pertanyaan dan pilihan jawaban dan persentase kombinasi pertanyaan dan pilihan jawaban dalam penggunaan fungsi random MySQL dan fungsi random Java Class Library pada aplikasi Computer Based Test (CBT). Untuk mengetahui apakah hipotesis ini diterima atau ditolak maka akan dibandingkan hasil percobaan terhadap pengaksesan model halaman pertanyaan menggunakan fungsi random MySQL dan fungsi random Java Class Library.

A. Uji Normalitas Data

Uji normalitas data dilakukan untuk mengetahui apakah data menyebar secara normal atau tidak. Untuk melakukan uji normalitas data, digunakan uji Lilliefors (Kolmogorov-Smirnov) dan dibantu dengan menggunakan aplikasi SPSS (Statistical Product and Service Solution). Dalam melakukan uji normalitas data, digunakan asumsi tingkat kepercayaan sebesar 95% dengan hipotesis sebagai berikut.

H_0 = data berasal dari populasi yang terdistribusi normal.

H_1 = data tidak berasal dari populasi yang terdistribusi normal.

Setelah dilakukan pengujian normalitas data menggunakan aplikasi SPSS, didapatkan hasil sebagai berikut (Tabel III).

TABEL III
HASIL PENGUJIAN NORMALITAS DATA
MENGUNAKAN UJI LILLIEFORS

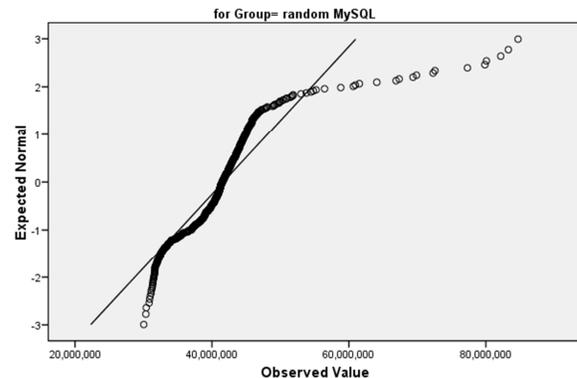
| Group | Kolmogorov-Smirnov ^a | | |
|---------------------------|---------------------------------|-----|------|
| | Statistic | df | Sig. |
| Kecepatan random MySQL | ,163 | 720 | ,000 |
| random Java Class Library | ,281 | 720 | ,000 |

a. Lilliefors Significance Correction

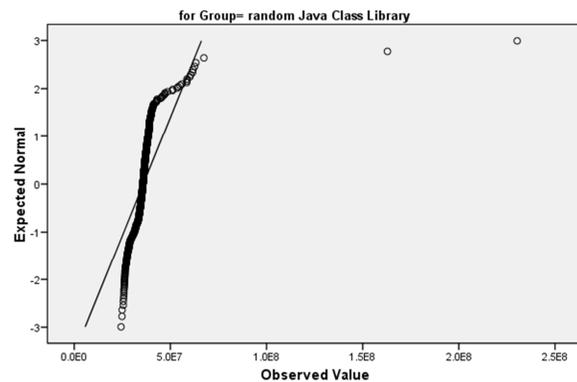
Dalam program SPSS digunakan istilah significance (yang disingkat Sig.) untuk P_{value} . Dari hasil pada Tabel 3, terlihat bahwa waktu akses pertanyaan dan pilihan jawaban untuk group random MySQL memiliki $P_{value}=0.000$ dan lebih kecil dari $\alpha=0.05$ sehingga dapat disimpulkan bahwa hipotesis yang diterima adalah H_1 yang

artinya data tidak berasal dari populasi yang terdistribusi normal.

Demikian pula untuk group random Java Class Library yang memiliki $P_{value}=0.000$ dan lebih kecil dari $\alpha=0.05$ sehingga dapat disimpulkan bahwa hipotesis yang diterima adalah H_1 yang artinya data tidak berasal dari populasi yang terdistribusi normal. Kesimpulan ini diperkuat dengan melihat hasil dari grafik normal probability plot dan detrended normal plot untuk data hasil percobaan tersebut. Grafik normal probability plot dapat ditunjukkan seperti pada Gambar 4 dan Gambar 5.

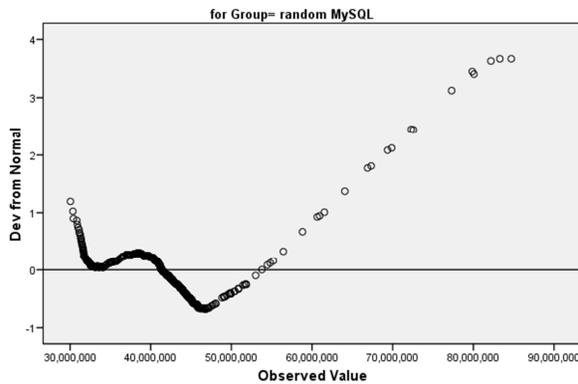


Gambar 4. Grafik Normal Probability Plot



Gambar 5. Grafik Normal Probability Plot

Pada grafik normal probability plot setiap nilai data yang diamati dipasangkan dengan nilai harapannya (*expected value*) dari distribusi normal. Jika data berasal dari populasi yang terdistribusi normal, maka titik-titik nilai data akan terletak kurang lebih dalam suatu garis lurus. Namun grafik normal probability plot pada Gambar 4 dan Gambar 5 menunjukkan bahwa titik-titik nilai data tidak terletak dalam suatu garis lurus melainkan membentuk pola yang tidak lurus. Grafik detrended normal plot dapat ditunjukkan seperti pada Gambar 6 dan Gambar 7.



Gambar 6. Grafik Detrended Normal Plot



Gambar 7. Grafik Detrended Normal Plot

Pada grafik detrended normal plot yang digambarkan adalah simpangan dari nilai data terhadap garis lurus. Jika data berasal dari populasi yang terdistribusi normal, maka titik-titik nilai data tidak akan membentuk pola tertentu dan akan terkumpul di sekitar garis mendatar yang melalui titik nol. Namun grafik detrended normal plot pada Gambar 6 dan Gambar 7 menunjukkan bahwa titik-titik nilai data tidak terkumpul di sekitar garis mendatar melainkan membentuk pola tertentu.

Konklusi dari hasil uji normalitas ini adalah data waktu akses pertanyaan dan pilihan jawaban menggunakan random MySQL maupun random Java Class Library tidak berasal dari populasi yang terdistribusi normal. Dengan demikian asumsi dari statistik parametrik tidak terpenuhi dan digunakan pengujian data secara statistik nonparametrik.

B. Uji Beda

Uji beda dilakukan untuk mengetahui apakah terdapat perbedaan pada waktu akses pertanyaan dan jawaban menggunakan fungsi random MySQL dan fungsi random Java Class Library. Berdasarkan hasil dari uji normalitas data dengan hipotesis yang diterima bahwa data tidak berasal dari populasi

terdistribusi normal, maka uji beda yang dilakukan menggunakan cara statistik nonparametrik yaitu dengan uji Mann-Whitney. Uji Mann-Whitney disebut juga uji-U atau Uji Jumlah Peringkat Wilcoxon yang merupakan alternatif dari uji-t dua sample independen.

Setelah dilakukan pengujian Mann-Whitney menggunakan aplikasi SPSS, didapatkan hasil sebagai berikut (Tabel IV).

TABEL IV
HASIL PENGUJIAN MANN-WHITNEY
MENGUNAKAN APLIKASI SPSS

| | Kecepatan |
|------------------------|-----------|
| Mann-Whitney U | 85004,000 |
| Wilcoxon W | 344564,00 |
| Z | -22,078 |
| Asymp. Sig. (2-tailed) | ,000 |

a. Grouping Variable: Group

Uji Mann-Whitney untuk hipotesis $H_0: \eta_1 \leq \eta_2$ terhadap $H_1: \eta_1 > \eta_2$ memberikan nilai $Z = -22.078$ dengan $P_{value} = 0.000$. Karena $P_{value} = 0.000$ lebih kecil dari $\alpha = 0.05$ maka $H_0: \eta_1 \leq \eta_2$ ditolak. Sehingga dapat disimpulkan bahwa median waktu akses pertanyaan dan pilihan jawaban menggunakan fungsi random MySQL lebih besar dari pada median waktu akses pertanyaan dan pilihan jawaban menggunakan fungsi random Java Class Library. Hal ini berarti bahwa pertanyaan dan pilihan jawaban akan lebih cepat diakses menggunakan fungsi random Java Class Library dari pada menggunakan fungsi random MySQL. Hal ini dibuktikan dengan perhitungan deskriptif yang menunjukkan bahwa nilai rata-rata waktu akses pertanyaan dan pilihan jawaban menggunakan fungsi random Java Class Library sebesar 35905823.7431 nanosecond adalah lebih kecil dibandingkan dengan rata-rata waktu akses pertanyaan dan pilihan jawaban menggunakan fungsi random MySQL sebesar 41650473.6250 nanosecond.

IV. PENUTUP

A. Simpulan

Berdasarkan hasil pembahasan dan uraian pada bab-bab sebelumnya, maka dapat diperoleh kesimpulan sebagai berikut.

1. Hasil akhir dari penelitian ini adalah data statistik yang mengungkapkan perbedaan penggunaan fungsi random MySQL dan fungsi

random Java Class Library pada aplikasi Computer Based Test (CBT) berupa data waktu akses pertanyaan dan pilihan jawaban dan persentase munculnya kombinasi pertanyaan dan pilihan jawaban yang sama.

2. Tingkat efektifitas penggunaan fungsi random Java Class Library lebih tinggi dibandingkan tingkat efektifitas penggunaan fungsi random MySQL berdasarkan pengukuran waktu akses pertanyaan dan pilihan jawaban. Namun berdasarkan perhitungan persentase munculnya kombinasi pertanyaan dan pilihan jawaban yang sama, tingkat efektifitas penggunaan fungsi random Java Class Library lebih rendah dibandingkan tingkat efektifitas penggunaan fungsi random MySQL.
3. Rata-rata waktu akses pertanyaan dan pilihan jawaban menggunakan fungsi random Java Class Library sebesar 35905823.7431 nanosecond adalah lebih kecil dibandingkan dengan rata-rata waktu akses pertanyaan dan pilihan jawaban menggunakan fungsi random MySQL sebesar 41650473.6250 nanosecond.
4. Persentase munculnya kombinasi pertanyaan dan pilihan jawaban yang sama pada penggunaan fungsi random MySQL sebesar 63.6% adalah lebih kecil dibandingkan dengan penggunaan fungsi random Java Class Library yang mencapai persentase sebesar 65.1%.

B. Saran

1. Perlu dilakukan penelitian lebih lanjut mengenai fungsi random MySQL dan fungsi random Java Class Library terutama menitikberatkan pada jenis algoritma random dan kode sumber fungsi random yang digunakan.
2. Penerapan metode acak pada aplikasi Computer Based Test (CBT) akan mencapai hasil maksimal atau dapat menyajikan kombinasi pertanyaan dan pilihan jawaban yang selalu berbeda dengan menambahkan parameter untuk menyaring kombinasi pertanyaan dan pilihan jawaban yang sudah pernah muncul.

DAFTAR PUSTAKA

- [1] Bartram, D., and Hambleton, R.K., 2006, Computer-based testing and the internet: issues and advances, British.
- [2] Coffey, N., How does java.util.Random work and how good is it?, http://www.javamex.com/tutorials/random_numbers/java_util_random_algorithm.shtml, diakses tanggal 3 Januari 2012.
- [3] Oracle Corporation, 2006, Mathematical Functions, <http://dev.mysql.com/doc/refman/5.0/en/mathematical-functions.html>, diakses tanggal 3 Januari 2012.
- [4] Solichin, A., 2010, MySQL 5 Dari Pemula Hingga Mahir, Jakarta.
- [5] Uyanto, S.S., 2006, Pedoman Analisis Data dengan SPSS, edisi ke 2, Graha Ilmu, Yogyakarta.