

## Prototipe Sistem *Computer Based Test* dengan Pengacakan Soal Menggunakan Metode *Fisher-Yates Shuffle*

### *Prototype of Computer Based Test System with Question Randomization Using Fisher-Yates Shuffle Method*

ARYAJAYA ALAMSYAH<sup>1\*</sup>, IMAM MARZUKI SHOFI<sup>1</sup>, HENDRA BAYU SUSENO<sup>1</sup>

#### Abstrak

Evaluasi pendidikan di Sekolah Menengah Kejuruan Negeri 2 Depok seperti Ulangan Harian, Ulangan Tengah Semester, Ulangan Akhir Semester, dan Ujian Sekolah Berstandar Nasional belum terkomputerisasi. Evaluasi pendidikan yang belum terkomputerisasi ini mengakibatkan perlunya biaya untuk penggandaan soal dan waktu untuk proses koreksi jawaban. Oleh karena itu, solusi untuk menjawab masalah tersebut adalah pembuatan sebuah sistem terkomputerisasi untuk melakukan evaluasi pendidikan atau sering disebut sebagai *computer-based test* (CBT). Penelitian ini bertujuan untuk membuat sistem CBT di SMKN 2 Depok, sesuai dengan rencana pihak sekolah yang ingin merapkan sistem CBT pada evaluasi pendidikan. Penelitian ini menggunakan metode Fisher-Yates *shuffle* (FYS) untuk melakukan pengacakan soal. Metode pengembangan sistem yang digunakan adalah *Waterfall*. Alat yang digunakan dalam pembuatan CBT yaitu PHP, *JavaScript*, *Bootstrap* dan *MySQL*. Percobaan pada sistem CBT dilakukan dengan mengacak soal ujian yang terdiri atas 30 soal. Pengacakan soal dilakukan sebanyak 40 kali. Pengacakan soal yang dilakukan berhasil dengan baik, karena tidak terdapat urutan soal yang sama. Pengujian sistem dilakukan menggunakan metode *black box*. Hasil pengujian sistem menunjukkan bahwa fungsional sistem sudah 100% berjalan dengan baik.

Kata Kunci: *Computer Based Test*, *Fisher Yates Shuffle*, *MySQL*, *PHP*

#### Abstract

*Educational evaluations at Vocational High School 2 Depok, such as daily tests, mid-semester tests, end-semester tests, and school exams with national standards have not been computerized. This non-computerized educational evaluation resulted in costs for copying question sheets and time for checking answers. Therefore, the solution to answer this problem is the creation of a computerized system to evaluate education or often referred as computer-based test (CBT). This study aims to create a CBT system at SMKN 2 Depok, in accordance with the plans of the school who wants to implement the CBT system in educational evaluation. This study uses Fisher-Yates shuffle (FYS) method to randomize the questions. The system development method used is Waterfall. The tools used in making CBT are PHP, JavaScript, Bootstrap and MySQL. Experiments on the CBT system were carried out by randomizing the exam questions consisting of 30 questions. The randomization of the questions was carried out 40 times. The randomization of the questions was successful, because there was no sequence of questions in the same order. System testing is done using the black box method. The system test results show that the functional system is 100% running well.*

*Keywords: Computer Based Test, Fisher Yates Shuffle, MySQL, PHP.*

## PENDAHULUAN

Tolak ukur hasil pendidikan dan pembelajaran dapat diketahui dengan adanya evaluasi. Evaluasi pendidikan sering diartikan sebagai pengukuran atau penilaian hasil belajar-mengajar, padahal antara keduanya memiliki arti yang berbeda meskipun saling berhubungan. ‘Mengukur’ adalah membandingkan sesuatu dengan satu ukuran (kuantitatif), sedangkan

<sup>1</sup>Teknik Informatika, Fakultas Sains dan Teknologi, UIN Syarif Hidayatullah Jakarta, Tangerang Selatan, 15412;

\*Penulis Korespondensi: Surel: kusinalamsyah@apps.ipb.ac.id

'menilai' adalah mengambil suatu keputusan terhadap sesuatu dengan ukuran baik dan buruk (kualitatif). Adapun pengertian evaluasi pendidikan meliputi keduanya (Arikunto 2012).

Cara-cara yang digunakan untuk proses evaluasi pendidikan atau evaluasi pembelajaran terbagi menjadi enam, yaitu evaluasi dengan bentuk soal benar atau salah, pilhan ganda, pertanyaan singkat, menjodohkan, uraian dan tugas (Arikunto 2012). Pelaksanaan evaluasi pendidikan umumnya terbagi menjadi Ulangan Harian (UH), Ulangan Tengah Semester (UTS), Ulangan Akhir Semester (UAS), Ujian Sekolah Berstandar Nasional (USBN), dan Ujian Nasional (UN).

Seiring perkembangan teknologi dan informasi, pelaksanaan UN sejak tahun 2014 sudah menggunakan *computer-based test* (CBT) (Kemendikbud 2020). Berdasarkan hasil observasi yang dilakukan pada tanggal 20 Agustus 2018 pada 17 sekolah negeri (SMA dan/atau SMK) di Kota Depok, diketahui bahwa 100% sekolah sudah melaksanakan UN menggunakan CBT yang dibuat oleh Kementerian Pendidikan dan Kebudayaan (Kemendikbud). Sedangkan pelaksanaan UH, UTS, UAS, dan USBN yang menggunakan sistem CBT hanya terdapat pada dua sekolah, yaitu SMAN 1 Depok dan SMAN 6 Depok.

Evaluasi pendidikan di SMKN 2 Depok masih menggunakan cara manual. Hal ini berdasarkan hasil wawancara yang dilakukan dengan Bapak Sutarsa, M.Pd sebagai Wakil Kepala Sekolah Bidang Kurikulum. Cara manual yang dimaksud adalah guru membuat soal ujian dan menggandakan soal ujian tersebut. Setelah itu, setiap siswa diberikan soal ujian yang sama dan lembar jawaban untuk menjawab soal. Kemudian, guru memeriksa setiap lembar jawaban. Hal ini berakibat pada perlunya biaya untuk penggandaan soal dan waktu dalam mengkoreksi jawaban. Pihak sekolah belum memiliki sebuah sistem untuk menyelenggarakan evaluasi pendidikan secara terkomputerisasi. Oleh karena itu, solusi untuk masalah tersebut adalah pembuatan sebuah sistem terkomputerisasi untuk evaluasi proses pembelajaran atau yang sering disebut CBT. Hal ini sesuai dengan rencana SMKN 2 Depok untuk segera menerapkan CBT untuk evaluasi pendidikan seperti UH, UTS, UAS dan USBN, sehingga pada saat pelaksanaan UN siswa sudah terbiasa dengan sistem CBT.

CBT merupakan tes yang diselenggarakan dengan menggunakan komputer sebagai media utama. Karakteristik dari tes ini sama dengan tes konvensional, yaitu menggunakan satu perangkat tes untuk beberapa peserta dengan panjang tes yang sama (Hariyanto 2017). Pembuatan CBT dilengkapi dengan fitur pengacakan soal yang bertujuan untuk meningkatkan kejujuran, mengurangi angka mencontek, dan melatih siswa untuk fokus terhadap soal. Pengacakan soal berisi soal yang sama, namun urutan soal yang ditampilkan kepada siswa berbeda-beda satu sama lain. Penerapan CBT dengan pengacakan soal memerlukan metode untuk melakukan pengacakan soal. Metode yang dapat digunakan seperti *linear congruent generator* (LCG) dan Fisher-Yates *shuffle* (FYS).

LCG adalah pembangkitan bilangan acak bersifat rekursif linier yang dikombinasikan dengan fungsi modulus (Kakiy 2004). Kelebihan dari LCG adalah mampu menghasilkan bilangan acak tak terhingga dengan cepat, sedangkan kekurangannya adalah kurang memperhatikan hasil bilangan acak tersebut (Kakiy 2004; Gunawan dan Prabowo 2017). Hasil penelitian yang dilakukan oleh Gunawan dan Prabowo (2017) menjelaskan bahwa dari 100 soal yang telah diacak menggunakan metode LCG, masih terdapat pengulangan pada beberapa soal. Hal ini mengakibatkan soal yang sama akan muncul hingga beberapa kali.

FYS adalah sebuah metode untuk menghasilkan suatu permutasi acak dari suatu himpunan terhingga, atau dengan kata lain digunakan untuk mengacak himpunan (Saokani 2016; Kustiansyah 2017; Fiqri 2017). Kelebihan dari FYS adalah tidak menghasilkan angka yang sama dalam satu kali pengacakan karena dilakukan dengan cara permutasi. Kekurangan dari FYS adalah terjadinya kesamaan terhadap hasil pengacakan setelah melakukan beberapa kali pengacakan (Saokani 2016; Kustiansyah 2017; Fiqri 2017). Hasil penelitian yang dilakukan oleh Suprpto *et al.* (2020) menjelaskan bahwa 20 soal yang telah diacak menggunakan metode FYS tidak menghasilkan pengulangan soal.

Metode yang digunakan pada penelitian ini adalah FYS karena hasil pengacakan FYS tidak mengalami pengulangan soal. Penelitian ini bertujuan membuat prototipe sistem CBT yang dilengkapi fitur pengacakan soal. Sistem yang menerapkan CBT ini diharapkan dapat digunakan dalam proses evaluasi pendidikan seperti UH, UTS, UAS dan USBN di SMKN 2 Depok.

## METODE

### Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *Waterfall*. Model *Waterfall* sering juga disebut sebagai model sekuensial linier atau model air terjun. Model *Waterfall* menyediakan pendekatan alur hidup perangkat lunak secara sekuensial dimulai dari analisis, desain, pengkodean, dan pengujian (Pressman 2010). Metode *Waterfall* digunakan karena sistem yang dibuat masih skala kecil dan memiliki spesifikasi yang tidak berubah-ubah. Terdapat 4 tahapan dalam pengembangan sistem dengan *Waterfall*, yaitu tahap analisis, desain, pengkodean, dan pengujian.

- a. **Tahap Analisis:** Tahap analisis berisi proses pengumpulan kebutuhan yang dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak. Wawancara dilakukan untuk menganalisis sistem dalam perancangan sistem CBT yang sesuai kebutuhan. Hasil yang didapatkan dari tahap analisis adalah data siswa, data jurusan, data mata pelajaran, dan cara untuk pengacakan soal.
- b. **Tahap Desain:** Tahap desain berisi proses multi langkah yang berfokus pada empat atribut perangkat lunak yang berbeda, yaitu struktur data, arsitektur perangkat lunak, representasi tampilan antarmuka, dan detail algoritma. Tahap ini menjelaskan tentang desain perangkat lunak seperti perancangan *Data Flow Diagram* (DFD), *Entity Relationship Diagram* (ERD), pengacakan soal, dan antarmuka.
- c. **Tahap Pengkodean:** Tahap pengkodean berisi proses translasi hasil desain ke dalam kode program. Hasil dari tahap ini adalah perangkat lunak yang sesuai dengan desain yang telah dibuat sebelumnya. Penulisan kode program pada sistem CBT menggunakan bahasa pemrograman PHP dan sistem basis data MySQL.
- d. **Tahap Pengujian:** Tahap pengujian berisi pengujian yang dilakukan terhadap perangkat lunak yang difokuskan secara fungsional dan memastikan bahwa semua bagian pada sistem CBT sudah diuji. Hal ini bertujuan meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan. Pengujian sistem fungsional CBT dilakukan menggunakan metode *black-box*.

### Metode Pengacakan Soal

Jika FYS diimplementasikan dengan benar, maka hasil dari algoritma ini tidak akan berat sebelah, sehingga setiap permutasi memiliki kemungkinan yang sama (Saokani 2016; Kustiansyah 2017; Fiqri 2017). Notasi algoritma FYS dapat dilihat pada Gambar 1.

```
-- To shuffle an array a of n elements (indices 0..n-1):
for i from n-1 downto 1 do
  j ← random integer such that 0 ≤ j ≤ i
  exchange a[j] and a[i]
```

Gambar 1 Notasi algoritma FYS (Gayathri *et al.* 2016).

## HASIL DAN PEMBAHASAN

### Perancangan *Data Flow Diagram* (DFD)

DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail untuk

merepresentasikan aliran informasi atau fungsi yang lebih detail. DFD juga menyediakan mekanisme untuk pemodelan fungsional ataupun pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur, karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur (Pressman 2010).

Perancangan DFD yang dilakukan pada tahap ini terdiri dari *context diagram*, DFD level 1, dan DFD level 2. Desain *context diagram* dapat dilihat pada Gambar 2. *Context diagram* bertujuan untuk melihat gambaran umum dari sistem CBT yang dibuat. Desain DFD level 1 seperti yang terlihat pada Gambar 3 bertujuan untuk melihat rincian dari *context diagram*. Pada DFD level 1 terlihat modul-modul apa saja yang ada di sistem CBT. Desain DFD level 2 dapat dilihat pada Gambar 4, Gambar 5 dan Gambar 6. Pada DFD level 2, setiap modul-modul yang dibuat akan dirincikan menjadi sub-modul.

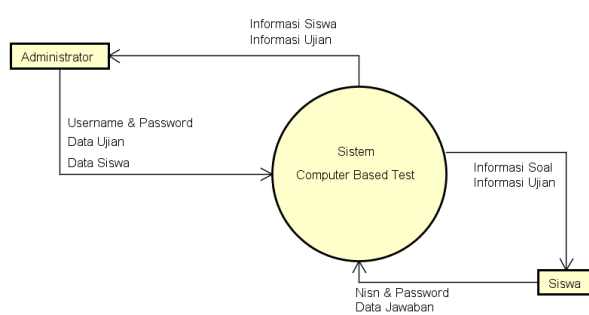
### Perancangan *Entity Relationship Diagram* (ERD)

Perancangan basis data menggunakan RDBMS MySQL berfungsi sebagai media penyimpanan data. Struktur basis data dalam bentuk ERD dapat dilihat pada Gambar 7. ERD tersebut sudah menerapkan konsep normalisasi yang bertujuan untuk menghindari duplikasi data dan mempermudah proses manipulasi data.

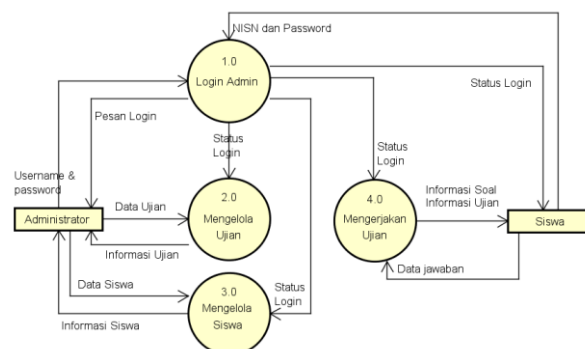
### Perancangan Sistem *Computer Best Test* (CBT)

Pengguna sistem CBT terbagi dua, yaitu administrator dan siswa. Setiap pengguna mengakses modul berbeda sesuai hak aksesnya. Administrator mengakses Modul Siswa dan Modul Ujian. Modul Siswa berisi sub-modul Siswa, Tahun Ajar, Jurusan, Jenis Kelas, dan Kelas. Modul Ujian berisi sub-modul Jenis Pelajaran, Jenis Ujian, Ujian, Soal, Jawaban, dan Peserta Ujian. Siswa mengakses modul Mengerjakan Ujian. Modul Mengerjakan Ujian berisi sub-modul Mengerjakan Ujian, Memeriksa Hasil Ujian, dan Melihat Riwayat Ujian. Sistem CBT dibuat menggunakan bahasa PHP. Tampilan antarmuka sistem CBT dibuat menggunakan JavaScript dan Bootstrap agar dapat menyesuaikan dengan layar monitor pengguna.

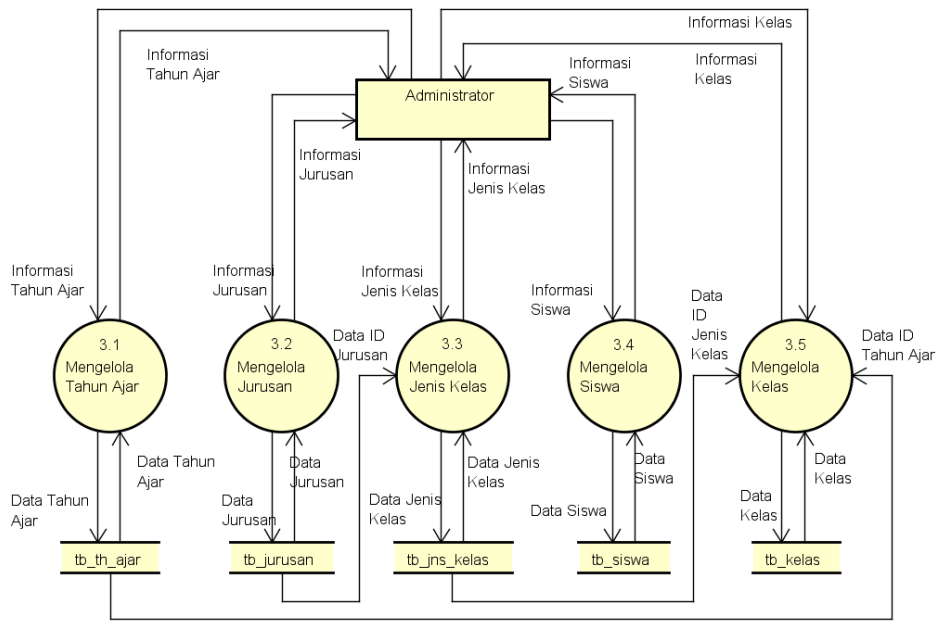
Tampilan antarmuka sub-modul Mengerjakan Ujian dapat dilihat pada Gambar 8. Antarmuka Mengerjakan Ujian sudah dilengkapi fitur untuk melihat soal yang sudah dikerjakan yang diwarnai hijau, sudah dikerjakan tetapi masih ragu-ragu yang diwarnai kuning dan belum dikerjakan yang diwarnai merah. Tampilan antarmuka sub-modul Memeriksa Hasil Ujian dapat dilihat pada Gambar 9. Antarmuka Memeriksa Hasil Ujian bertujuan agar siswa dapat mengetahui berapa banyak soal yang dijawab benar dan berapa soal yang dijawab salah. Tampilan antarmuka sub-modul Melihat Riwayat Ujian dapat dilihat pada Gambar 10. Antarmuka Melihat Riwayat Ujian berguna untuk melihat apa saja ujian yang sudah dikerjakan dan apa saja ujian yang belum dikerjakan.



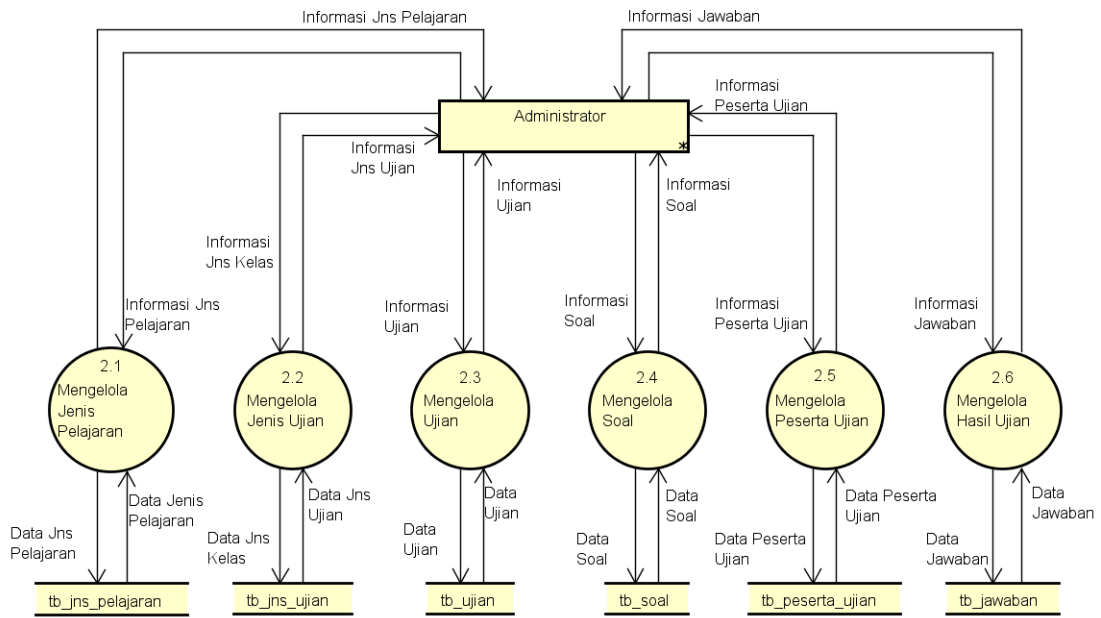
Gambar 2 *Context diagram* dari sistem CBT.



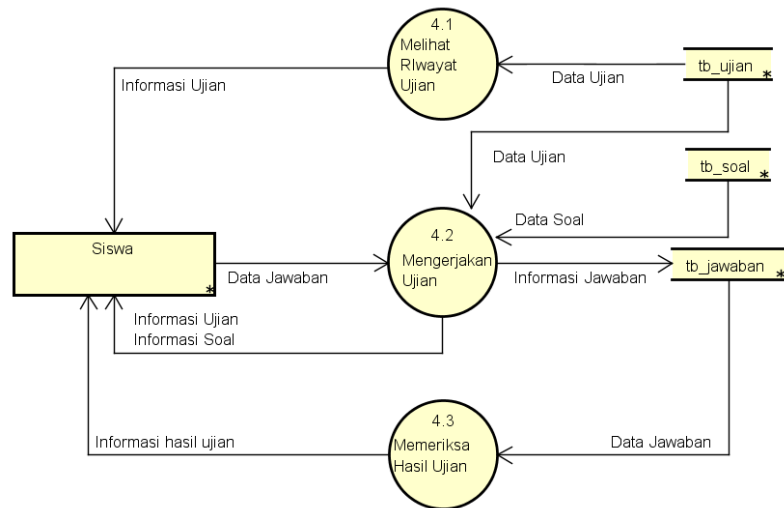
Gambar 3 *Data Flow Diagram* (DFD) level 1.



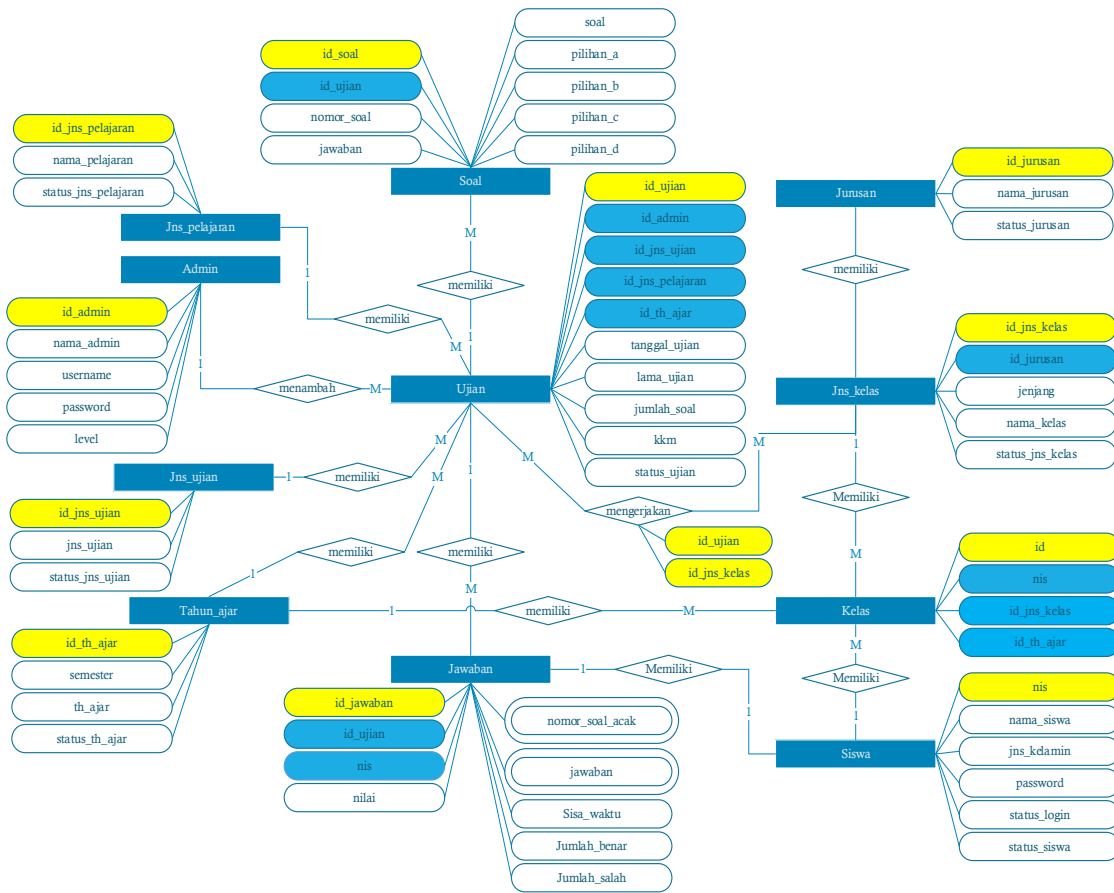
Gambar 4 DFD level 2 modul mengelola siswa.



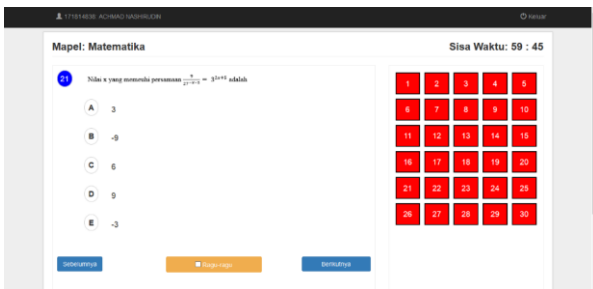
Gambar 5 DFD level 2 modul mengelola ujian.



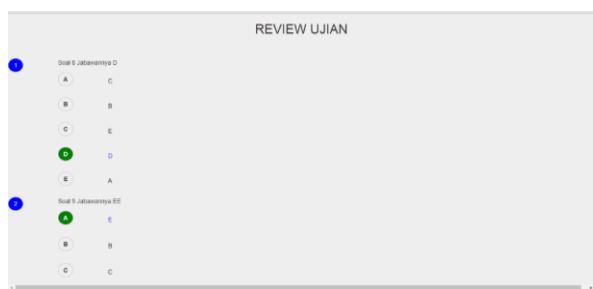
Gambar 6 DFD level 2 modul mengerjakan ujian.



Gambar 7 ERD sistem CBT.



Gambar 8 Antarmuka Mengerjakan Ujian.



Gambar 9 Antarmuka Mengerjakan Ujian.

No	Kelas	Jenis Ujian	Nama Mapel	Semester	Tahun Ajar	Tanggal	Nilai	Aksi
1	X-TAV-1	Ujian Sekolah Berstandar Nasional	Algoritma dan Pemrograman 1	Genap	2017/2018	25 Mei 2018		<a href="#">Kerjakan</a>
2	X-TAV-1	Ujian Sekolah Berstandar Nasional	Matematika	Genap	2017/2018	25 Mei 2018		<a href="#">Kerjakan</a>

Gambar 10 Antarmuka Mengerjakan Ujian.

### Hasil Metode Pengacakan Soal

Metode pengacakan soal menggunakan FYS melakukan pengacakan dengan cara permutasi terhadap soal. Tabel 1 merupakan pengujian FYS dengan contoh soal USBN mata pelajaran Matematika yang berjumlah 30 soal di tahun ajar 2016/2017 terhadap 40 siswa. Berdasarkan hasil pengujian FYS, dapat diketahui bahwa sistem CBT yang dibuat sudah dapat mengacak soal dengan baik, sehingga setiap siswa mendapat urutan soal yang berbeda-beda.

## Hasil Pengujian Sistem CBT

Pengujian sistem CBT menggunakan metode *black box*. Metode *black box* merupakan tipe pengujian yang memperlakukan sistem yang tidak diketahui kinerja internalnya. Pengujiannya dilakukan melalui pengujian fungsionalitas (Pressman 2010). Cara pengujian *blackbox* dilakukan dengan menjalankan sistem CBT. Fungsi-fungsi dari sistem CBT sudah sesuai yang diharapkan atau tidak. Hasil pengujian *blackbox* dapat dilihat pada Tabel 2. Berdasarkan hasil pengujian *black box*, diketahui bahwa sistem CBT yang dibuat sudah 100% berjalan sesuai dengan fungsionalitasnya masing-masing.

Tabel 1 Hasil pengacakan soal menggunakan Fisher-Yates *shuffle*

Kode Uji	Nomor soal	Keterangan
Uji-01	21, 30, 17, 12, 25, 8, 2, 16, 29, 18, 6, 3, 20, 4, 9, 15, 23, 5, 22, 1, 11, 13, 27, 26, 19, 10, 28, 24, 7, 14	Berhasil
Uji-02	16, 4, 25, 6, 9, 28, 23, 22, 8, 24, 30, 10, 18, 15, 21, 14, 5, 20, 27, 26, 17, 1, 19, 29, 7, 2, 11, 13, 12, 3	Berhasil
Uji-03	28, 11, 14, 2, 15, 12, 23, 29, 18, 30, 10, 26, 27, 7, 5, 13, 19, 3, 16, 25, 4, 17, 22, 21, 20, 8, 6, 1, 24, 9	Berhasil
Uji-04	11, 27, 22, 3, 23, 16, 29, 14, 13, 19, 26, 8, 17, 15, 24, 28, 21, 9, 5, 1, 2, 6, 12, 18, 20, 10, 25, 30, 7, 4	Berhasil
Uji-05	7, 25, 8, 5, 9, 16, 27, 18, 11, 22, 13, 20, 6, 23, 12, 17, 10, 3, 24, 4, 15, 14, 30, 19, 2, 28, 21, 26, 1, 29	Berhasil
Uji-06	30, 8, 18, 24, 23, 27, 1, 22, 25, 14, 7, 26, 2, 11, 12, 17, 10, 20, 3, 29, 5, 4, 16, 19, 28, 9, 13, 21, 6, 15	Berhasil
Uji-07	22, 21, 24, 30, 28, 2, 4, 13, 10, 12, 15, 19, 18, 29, 5, 17, 8, 16, 9, 23, 11, 25, 6, 27, 20, 1, 7, 3, 26, 14	Berhasil
Uji-08	20, 29, 2, 17, 26, 9, 1, 6, 7, 4, 22, 28, 25, 8, 5, 13, 15, 14, 23, 3, 18, 11, 24, 19, 27, 21, 16, 30, 10, 12	Berhasil
Uji-09	19, 3, 21, 10, 6, 1, 12, 29, 14, 24, 2, 28, 16, 8, 9, 25, 5, 20, 30, 11, 13, 15, 7, 26, 27, 17, 23, 4, 22, 18	Berhasil
Uji-10	16, 7, 29, 11, 22, 21, 3, 12, 9, 24, 8, 10, 23, 5, 26, 1, 28, 6, 14, 20, 4, 13, 2, 19, 15, 18, 27, 30, 25, 17	Berhasil
Uji-11	24, 21, 5, 8, 9, 6, 16, 1, 26, 22, 19, 27, 11, 17, 4, 30, 3, 7, 14, 12, 15, 25, 20, 23, 18, 29, 13, 28, 10, 2	Berhasil
Uji-12	14, 29, 4, 18, 2, 6, 25, 17, 27, 16, 9, 10, 5, 15, 13, 22, 26, 11, 30, 1, 7, 23, 21, 12, 24, 19, 3, 28, 20, 8	Berhasil
Uji-13	11, 16, 22, 23, 25, 27, 26, 29, 18, 14, 8, 6, 13, 9, 2, 19, 7, 15, 1, 4, 21, 5, 28, 17, 24, 10, 12, 3, 30, 20	Berhasil
Uji-14	13, 10, 24, 29, 27, 12, 25, 1, 4, 28, 21, 17, 5, 6, 19, 23, 8, 2, 15, 26, 22, 30, 16, 7, 20, 18, 3, 14, 11, 9	Berhasil
Uji-15	29, 22, 18, 28, 20, 21, 24, 2, 27, 19, 16, 26, 30, 15, 11, 9, 1, 25, 13, 17, 6, 14, 4, 5, 3, 8, 10, 7, 23, 12	Berhasil
Uji-16	4, 12, 14, 22, 24, 5, 10, 25, 28, 23, 18, 20, 6, 7, 16, 11, 8, 13, 9, 19, 26, 29, 3, 17, 21, 30, 27, 15, 2, 1	Berhasil
Uji-17	28, 19, 17, 6, 23, 13, 5, 2, 3, 26, 22, 29, 15, 16, 9, 24, 25, 11, 7, 8, 30, 21, 4, 20, 10, 12, 14, 27, 18, 1	Berhasil
Uji-18	17, 5, 8, 15, 2, 9, 20, 14, 26, 22, 29, 19, 28, 16, 24, 21, 4, 12, 1, 11, 6, 27, 10, 7, 13, 30, 18, 25, 3, 23	Berhasil
Uji-19	23, 30, 22, 16, 3, 15, 13, 26, 5, 14, 24, 1, 6, 17, 28, 21, 9, 18, 27, 25, 2, 4, 19, 12, 7, 29, 20, 11, 10, 8	Berhasil
Uji-20	8, 16, 7, 25, 3, 24, 15, 9, 5, 10, 14, 30, 18, 22, 27, 13, 11, 21, 2, 19, 17, 23, 28, 26, 4, 6, 29, 20, 1, 12	Berhasil
Uji-21	9, 8, 10, 4, 14, 7, 26, 28, 29, 24, 23, 3, 20, 12, 21, 22, 15, 11, 19, 16, 6, 2, 1, 18, 13, 17, 5, 25, 30, 27	Berhasil
Uji-22	15, 14, 18, 23, 1, 26, 7, 17, 19, 5, 28, 9, 2, 12, 22, 4, 24, 21, 11, 6, 20, 29, 10, 8, 27, 13, 30, 25, 3, 16	Berhasil
Uji-23	28, 8, 21, 7, 19, 27, 23, 12, 2, 5, 24, 22, 9, 26, 11, 16, 3, 15, 10, 6, 1, 20, 17, 30, 29, 14, 4, 13, 18, 25	Berhasil
Uji-24	28, 15, 3, 8, 30, 20, 6, 17, 1, 7, 11, 5, 21, 10, 2, 16, 29, 24, 12, 22, 19, 25, 13, 9, 27, 4, 18, 26, 23, 14	Berhasil
Uji-25	30, 12, 24, 16, 15, 2, 10, 3, 7, 1, 28, 17, 19, 20, 26, 18, 29, 4, 21, 27, 13, 9, 8, 6, 22, 11, 23, 14, 25, 5	Berhasil
Uji-26	3, 5, 29, 13, 27, 14, 28, 23, 22, 10, 2, 9, 26, 21, 7, 4, 8, 20, 16, 11, 6, 18, 17, 30, 15, 12, 24, 25, 19, 1	Berhasil
Uji-27	30, 27, 3, 10, 19, 25, 22, 8, 28, 20, 5, 17, 2, 21, 29, 24, 11, 4, 12, 13, 6, 16, 26, 23, 9, 15, 18, 7, 14, 1	Berhasil
Uji-28	9, 28, 8, 15, 6, 1, 24, 13, 22, 5, 2, 23, 14, 20, 29, 19, 3, 26, 4, 21, 12, 18, 27, 11, 30, 16, 7, 17, 10, 25	Berhasil
Uji-29	13, 7, 15, 26, 21, 6, 8, 22, 9, 5, 24, 1, 29, 18, 16, 12, 20, 3, 23, 2, 11, 28, 17, 19, 25, 14, 4, 27, 30, 10	Berhasil
Uji-30	2, 4, 5, 14, 9, 19, 13, 30, 15, 16, 17, 3, 1, 27, 23, 18, 22, 8, 24, 28, 26, 25, 12, 20, 11, 29, 21, 10, 7, 6	Berhasil
Uji-31	4, 15, 11, 28, 27, 6, 10, 16, 25, 1, 29, 20, 24, 23, 22, 17, 9, 3, 19, 5, 14, 13, 18, 7, 8, 2, 30, 21, 12, 26	Berhasil
Uji-32	25, 23, 15, 28, 24, 30, 27, 17, 11, 10, 18, 29, 13, 14, 5, 12, 22, 6, 1, 9, 21, 3, 19, 8, 2, 26, 20, 4, 7, 16	Berhasil
Uji-33	15, 12, 18, 20, 28, 17, 9, 5, 4, 8, 19, 24, 2, 1, 30, 25, 23, 29, 21, 13, 14, 16, 3, 22, 7, 27, 10, 11, 6, 26	Berhasil
Uji-34	30, 22, 18, 14, 16, 28, 7, 3, 23, 29, 21, 26, 15, 25, 24, 13, 5, 9, 4, 19, 2, 8, 11, 6, 20, 1, 10, 27, 12, 17	Berhasil
Uji-35	2, 16, 8, 27, 7, 13, 23, 26, 30, 15, 6, 19, 4, 24, 12, 17, 11, 21, 28, 20, 29, 1, 10, 18, 25, 3, 5, 14, 22, 9	Berhasil
Uji-36	26, 30, 25, 14, 21, 8, 17, 19, 23, 9, 2, 3, 6, 16, 24, 29, 7, 18, 13, 27, 28, 4, 20, 11, 1, 12, 5, 10, 22, 15	Berhasil
Uji-37	18, 24, 6, 11, 28, 2, 7, 9, 8, 15, 26, 29, 20, 22, 23, 10, 3, 27, 13, 5, 30, 12, 25, 1, 4, 14, 19, 16, 21, 17	Berhasil
Uji-38	26, 15, 18, 12, 25, 9, 30, 29, 3, 13, 5, 22, 10, 16, 11, 23, 21, 17, 4, 28, 7, 14, 8, 24, 2, 1, 6, 20, 19, 27	Berhasil
Uji-39	13, 25, 20, 9, 12, 27, 3, 16, 7, 21, 18, 17, 2, 15, 8, 4, 10, 1, 26, 6, 24, 29, 28, 22, 30, 14, 23, 19, 5, 11	Berhasil
Uji-40	19, 18, 28, 24, 20, 8, 12, 5, 21, 22, 3, 25, 15, 11, 7, 16, 6, 4, 10, 30, 2, 9, 1, 26, 27, 17, 29, 23, 13, 14	Berhasil

Tabel 2 Hasil pengujian *black box*

Sub-modul	Fungsi	Hasil yang diharapkan	Keterangan
Tahun ajar	Lihat tahun ajar	Setelah <i>login</i> , admin dapat melihat data tahun ajar yang ada dalam sistem	Berhasil
	Tambah tahun ajar	Setelah <i>login</i> , admin dapat menambah tahun ajar.	Berhasil
	Ubah tahun ajar	Setelah <i>login</i> , admin dapat merubah tahun ajar apabila terjadi kesalahan dalam melakukan penambahan data.	Berhasil
Siswa	Lihat siswa	Setelah <i>login</i> , admin dapat melihat seluruh data siswa yang ada terdaftar pada sistem.	Berhasil
	Cari siswa	Setelah <i>login</i> , admin dapat mencari data siswa yang terdaftar pada sistem. Dengan cara memasukkan nama siswa pada kolom pencarian.	Berhasil
	Tambah siswa	Setelah <i>login</i> , admin dapat menambah data siswa.	Berhasil
	Ubah siswa	Setelah <i>login</i> , admin dapat merubah data siswa apabila terjadi kesalahan data dalam proses penambahan data.	Berhasil
	<i>Reset login</i>	Setelah <i>login</i> , admin dapat me- <i>reset login</i> siswa. Apabila siswa memerlukan hal tersebut.	Berhasil
	<i>Import siswa</i>	Setelah <i>login</i> , admin dapat mengunggah file excel untuk menambah data siswa dalam jumlah banyak.	Berhasil
Jurusan	Lihat jurusan	Setelah <i>login</i> , admin dapat melihat semua jurusan.	Berhasil
	Tambah jurusan	Setelah <i>login</i> , admin dapat menambah jurusan.	Berhasil
	Ubah jurusan	Setelah <i>login</i> , admin dapat merubah jurusan apabila terjadi kesalahan dalam melakukan penambahan data	Berhasil
Jenis kelas	Lihat jenis kelas	Setelah <i>login</i> , admin dapat melihat seluruh jenis kelas yang tersedia dalam sistem.	Berhasil
	Cari jenis kelas	Setelah <i>login</i> , admin dapat mencari jenis kelas. Dengan cara menuliskan nama kelas pada kolom pencarian.	Berhasil
	Tambah jenis kelas	Setelah <i>login</i> , admin dapat menambah jenis kelas.	Berhasil
	Ubah jenis kelas	Setelah <i>login</i> , admin dapat merubah jenis kelas apabila terjadi kesalahan dalam melakukan penambahan data.	Berhasil
Kelas	Lihat kelas	Setelah <i>login</i> , admin dapat melihat seluruh data kelas yang ada pada sistem.	Berhasil
	Cari kelas	Setelah <i>login</i> , admin dapat mencari data kelas yang ada. Dengan cara masukan nama kelas pada kolom pencarian.	Berhasil
	Tambah kelas	Setelah <i>login</i> , admin dapat menambahkan data kelas.	Berhasil
	Ubah kelas	Setelah <i>login</i> , admin dapat merubah data kelas yang ada. Apabila terjadi kesalahan dalam melakukan proses penambahan data kelas.	Berhasil
	<i>Import kelas</i>	Setelah <i>login</i> , admin dapat mengunggah file excel untuk menambah data kelas.	Berhasil
Jenis pelajaran	Lihat jenis pelajaran	Setelah <i>login</i> , admin dapat melihat seluruh pelajaran yang diajarkan pada setiap jurusannya.	Berhasil
	Cari jenis pelajaran	Setelah <i>login</i> , admin dapat mencari pelajaran. Dengan cara menuliskan nama pelajaran pada kolom pencarian	Berhasil
	Tambah jenis pelajaran	Setelah <i>login</i> , admin dapat menambah jenis pelajaran yang akan diujikan.	Berhasil
	Ubah jenis pelajaran	Setelah <i>login</i> , admin dapat merubah jenis pelajaran apabila terjadi kesalahan dalam melakukan penambahan data.	Berhasil
Peserta ujian	Lihat peserta ujian	Setelah <i>login</i> , admin dapat melihat seluruh peserta ujian yang mengikuti ujian CBT.	Berhasil
	Cari peserta ujian	Setelah <i>login</i> , admin dapat mencari peserta ujian. Dengan cara memasukkan nama kelas pada kolom pencarian.	Berhasil
	Ubah peserta ujian	Setelah <i>login</i> , admin dapat merubah peserta ujian.	Berhasil
	Cetak peserta ujian	Setelah <i>login</i> , admin dapat mencetak kartu peserta ujian.	Berhasil
Jenis ujian	Lihat jenis ujian	Setelah <i>login</i> , admin dapat melihat semua jenis ujian yang ada pada sistem.	Berhasil
	Tambah jenis ujian	Setelah <i>login</i> , admin dapat menambah jenis ujian yang.	Berhasil
	Ubah jenis ujian	Setelah <i>login</i> , admin dapat merubah jenis ujian. Apabila terjadi kesalahan dalam proses penambahan jenis ujian.	Berhasil
	Tambah ujian	Setelah <i>login</i> , admin dapat menambah jadwal ujian.	Berhasil
	Ubah ujian	Setelah <i>login</i> , admin dapat merubah jadwal ujian. Apabila terjadi kesalahan dalam melakukan penambahan data.	Berhasil
Soal	Lihat soal	Setelah <i>login</i> , admin dapat melihat semua soal yang akan diujikan/dikerjakan oleh siswa.	Berhasil
	Cari soal	Setelah <i>login</i> , admin dapat mencari soal. Dengan cara masukan kata kunci pada kolom pencarian soal.	Berhasil
	Tambah soal	Setelah <i>login</i> , admin dapat menambah soal. Berdasarkan data ujian yang dipilih	Berhasil
	Ubah soal	Setelah <i>login</i> , admin dapat merubah soal. Apabila terjadi kesalahan dalam proses penambahan soal	Berhasil



Sub-modul	Fungsi	Hasil yang diharapkan	Keterangan
Mengerjakan ujian CBT	Hapus soal	Setelah <i>login</i> , admin dapat menghapus soal yang dianggap perlu untuk dihapus.	Berhasil
	Import soal	Setelah <i>login</i> , admin dapat mengunggah file excel untuk menambah soal.	Berhasil
	Lihat riwayat pengerjaan CBT	Setelah <i>login</i> , siswa melihat riwayat CBT yang pernah ia kerjakan.	Berhasil
	Mengerjakan CBT	Setelah <i>login</i> , siswa mengerjakan CBT apabila terdapat ujian yang belum siswa tersebut kerjakan.	Berhasil
	Memeriksa hasil jawaban CBT	Setelah <i>login</i> atau selesai mengerjakan CBT, siswa dapat memeriksa hasil jawaban yang pernah siswa tersebut kerjakan.	Berhasil

## SIMPULAN

Pengembangan Sistem CBT dengan pengacakan soal dibuat dengan berbasis web dan menggunakan FYS. Hasil percobaan yang dilakukan adalah dengan mengacak sebuah soal USBN mata pelajaran Matematika berjumlah 30 soal yang diacak sebanyak 40 kali dan menghasilkan urutan soal yang berbeda. Hasil pengujian sistem CBT menjelaskan sistem CBT yang dibuat sudah 100% berjalan sesuai dengan fungsionalitasnya masing-masing.

## DAFTAR PUSTAKA

- Aisyah N. 2016. Implementasi metode *Fisher Yates Shuffle* untuk pengacakan pertanyaan pada *game ali and the labirin* [skripsi]. Malang (ID): UIN Maulana Malik Ibrahim.
- Arikunto S. 2012. *Dasar-Dasar Evaluasi Proses Pendidikan Ed ke-2*. Jakarta (ID): Bumi Aksara.
- Fiqri MJ. 2017. Implementasi algoritma *Knuth Morris Pratt* dan *Fisher Yates Shuffle* pada aplikasi *speech recognition* pembelajaran shalat dan wudhu berbasis android [skripsi]. Bandung (ID): UIN Sunan Gunung Djati.
- Gayathri P, Gupta G, Santhi H, Naseera S, Siddiqui AA, Gopichand G. 2017. Music playlist manager using *fisher yates shuffling* algorithm and sorting. *World Wide Journal of Multidisciplinary Research and Development*. 3(12): 131-138.
- Gunawan, Prabowo DA. 2017. Sistem ujian *online* seleksi penerimaan mahasiswa baru dengan pengacakan soal menggunakan *linier congruent method*. *Jurnal Informatika Upgris*. 3(2): 71-79.
- Hariyanto A. 2016. *Membuat Aplikasi Computer Based Test dengan PHP MySQLi dan Bootstrap*. Yogyakarta (ID): Lokomedia.
- Kakiy TJ. 2004. *Pengantar Sistem Simulasi*. Yogyakarta (ID): Andi.
- Kustiansyah T. 2016. Implementasi algoritma *Fisher Yates Shuffle* dan *Virtual Reality* pada game pembelajaran pengenalan hewan darat untuk anak usia dini berbasis android [skripsi]. Bandung (ID): UIN Sunan Gunung Djati.
- Saokani U. 2016. Analisis perbandingan algoritma *Fisher Yates Shuffle* dan *Linier Congruent Method* untuk pengacakan soal ilmu nahwu berbasis android [skripsi]. Bandung (ID): UIN Sunan Gunung Djati.
- Suprpto DAA, Fuaziah, Hayati N. 2020. Aplikasi *smart entrance exam* berbasis android dengan algoritma *Fisher Yates Shuffle*. *Jurnal Sistem dan Teknologi Informasi*. 9(1): 25-32.
- Pressman RS. 2010. *Software Engineering: A Practitioner's Approach Ed ke-7*. Boston (US): McGraw Hill.