

Rancang Bangun Aplikasi Tes Toefl Berbasis Web di Universitas Katolik Santo Thomas Medan

¹Simon Kevin Siregar, ²Desinta Purba

¹ Universitas Katolik Santo Thomas, Jl. Setiabudi No. 479 F Tanjung Sari, Medan, Sumatera Utara,
Indonesia

E-Mail: desinta.poerba@yahoo.com¹, siregar25v@gmail.com²

Abstrak

TOEFL (*Test Of English as Foreign Language*) adalah suatu tes yang digunakan untuk mengukur kemampuan berbahasa inggris. Pada Universitas Katolik Santo Thomas Medan, tes TOEFL digunakan sebagai syarat kelulusan bagi mahasiswa yang ingin memperoleh gelar sarjana. Pada pelaksanaan tes TOEFL yang berjalan saat ini di Lembaga Bahasa Universitas Katolik Santo Thomas masih menggunakan sistem konvensional seperti eksekusi soal tes TOEFL dan memeriksa jawaban ujian tes TOEFL. Sehingga untuk mengatasi permasalahan tersebut diperlukan sebuah aplikasi tes TOEFL berbasis Web yang menerapkan Algoritma *Fisher Yates* untuk pengacakan soal dan jawaban untuk *section structure* sehingga tampilan soal dan jawaban untuk setiap peserta ujian berbeda pada section tersebut dan untuk *section listening* dan *reading* hanya dilakukan pengacakan jawaban. Algoritma *Fisher Yates* ialah sebuah algoritma untuk menghasilkan permutasi acak dari suatu himpunan terhingga. Aplikasi ini dibangun menggunakan *Laravel*, *Vue Js* dan *MySQL* serta menggunakan metode *prototype* sebagai metode pengembangan perangkat lunak. Pengujian aplikasi menggunakan *black box* untuk memastikan bahwa sistem yang dibangun dapat berjalan sesuai dengan fungsionalitasnya.

Kata Kunci: *Black box; Fisher Yates; Laravel; Prototype; TOEFL; Vue Js; Web.*

Abstract

The TOEFL (*Test of English as Foreign Language*) is a test used to measure English language skills. At Santo Thomas Catholic University, Medan, the TOEFL test is used as a graduation requirement for students who wish to obtain a bachelor's degree. The current TOEFL test implementation at the Santo Thomas Catholic University Language still uses conventional systems such as executing the TOEFL test and checking TOEFL test answers. So to overcome this problem, a Web-based TOEFL test application is needed that applies the Fisher Yates Algorithm to randomize questions and answers for the section structure so that the display of questions and answers for each test taker is different in that section and for the listening and reading sections only the answers are randomised. The Fisher Yates algorithm is an algorithm for generating random permutations from a finite set. This application was built using *Laravel*, *Vue Js* and *MySQL* and used the *prototype* method as a software development method. Application testing uses *black box* to ensure that the system built can run according to its functionality.

Keywords: *Black box; Fisher Yates; Laravel; Prototype; TOEFL; Vue Js; Web.*

PENDAHULUAN

TOEFL adalah singkatan dari (*Test of English as Foreign Language*) yang berorientasi kepada *American English*. TOEFL digunakan untuk mengukur tingkat

kemampuan bahasa inggris tiap orang. Beberapa jenis TOEFL yaitu PBT, IBT, CBT, dan ITP. Jenis tes bahasa inggris yang digunakan di dunia pendidikan yaitu TOEFL ITP (*Institutional Testing Program*). Aspek- aspek yang dinilai dalam tes TOEFL ITP yaitu *Listening comprehension*, *Structure and written expression*, dan *Reading comprehension*. Lembaga Bahasa Universitas Katolik Santo Thomas Medan merupakan salah satu lembaga tempat yang menyelenggarakan tes TOEFL.

Pelaksanaan tes TOEFL menjadi salah satu syarat wajib yang di ikuti oleh setiap mahasiswa yang akan menyelesaikan studi di Universitas Katolik Santo Thomas. Namun dalam proses eksekusi ujian tes TOEFL hingga perhitungan skor hasil nilai tes TOEFL masih di lakukan secara manual atau konvensional. Sehingga hal tersebut mungkin terjadi *human error* disaat pemeriksaan hasil ujian dan eksekusi ujian masih menggunakan *google form*. Pada sistem ujian tes TOEFL yang akan dibangun akan menerapkan pengacakan soal untuk *section structure and written expression* menggunakan metode algoritma *Fisher Yates* yaitu sebuah algoritma yang digunakan menghasilkan permutasi acak dari suatu himpunan terhingga, dengan kata lain untuk mengubah urutan masuk yang diberikan secara acak. Sehingga setiap peserta yang mengikuti ujian tes

TOEFL akan mendapatkan soal yang berbeda pada section tersebut. Dan untuk *section listening comprehension* dan *section reading comprehension* hanya menerapkan pengacakan jawaban, sehingga urutan jawaban setiap peserta berbeda. Sehingga dengan adanya aplikasi ini dapat memudahkan peserta dalam pelaksanaan ujian dari pendaftaran, eksekusi ujian dan dapat melihat sertifikat TOEFL secara langsung Ketika telah selesai menyelesaikan ujian TOEFL.

METODE PENELITIAN

Metode dalam pengembangan sistem ini menggunakan metode *Prototype*. Metode *prototype* merupakan metode yang memungkinkan pengguna atau user memiliki gambaran awal tentang perangkat lunak yang akan dikembangkan, serta pengguna dapat melakukan pengujian di awal sebelum perangkat lunak dirilis. Metode *prototype* memiliki beberapa tahapan yaitu, analisis kebutuhan, membuat dan evaluasi *prototype*, mengkodekan sistem, pengujian sistem, serta evaluasi dan penggunaan sistem. Berikut merupakan tahapan-tahapan metode *prototype* yang dilakukan oleh peneliti dalam membangun aplikasi tes TOEFL:

1. Analisis Kebutuhan, identifikasi kebutuhan dan kebutuhan sistem sehingga sistem yang akan dibangun sesuai dengan kriteria yang diharapkan.
2. Membuat dan Evaluasi *Prototype*, merupakan perancangan sistem, Dimana pada tahapan ini dilakukan analisa algoritma *Fisher Yates*, perancangan tampilan (*User Interface*) dan perancangan basis data (*Database*). Serta dievaluasi apakah rancangan *prototype* tersebut sudah sesuai kebutuhan sistem.
3. Mengkodekan Sistem, yaitu penerapan hasil perancangan sistem yang telah disepakati oleh pengembang sistem dan pengguna. Dalam mengkodekan sistem menggunakan *framework laravel* sebagai back-end dan *framework vue js sebagai front-end* dan MySQL sebagai tempat penyimpanan database.
4. Pengujian, untuk mengetahui sistem yang telah dibangun telah sesuai dengan perancangan yang dibuat, pengujian sistem dilakukan dengan menggunakan metode *black box*.
5. Evaluasi Sistem dan Penggunaan Sistem, sistem yang telah di evaluasi oleh pengguna dan berjalan sesuai harapan siap untuk di *deploy* untuk dapat digunakan secara

online.

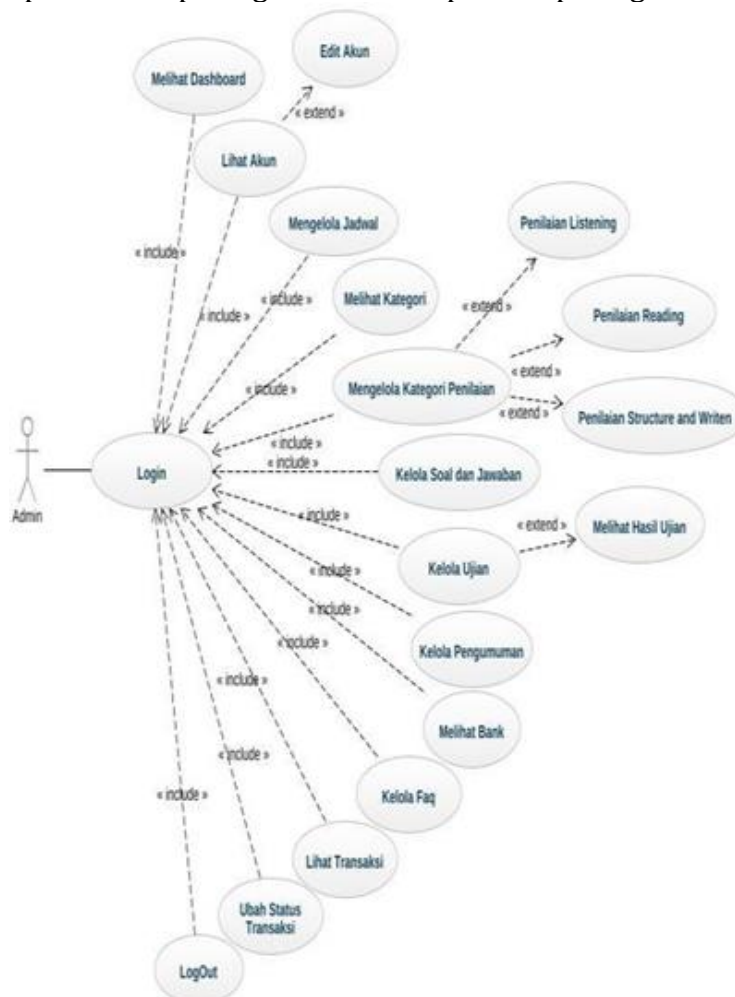
HASIL DAN PEMBAHASAN

Desain Sistem

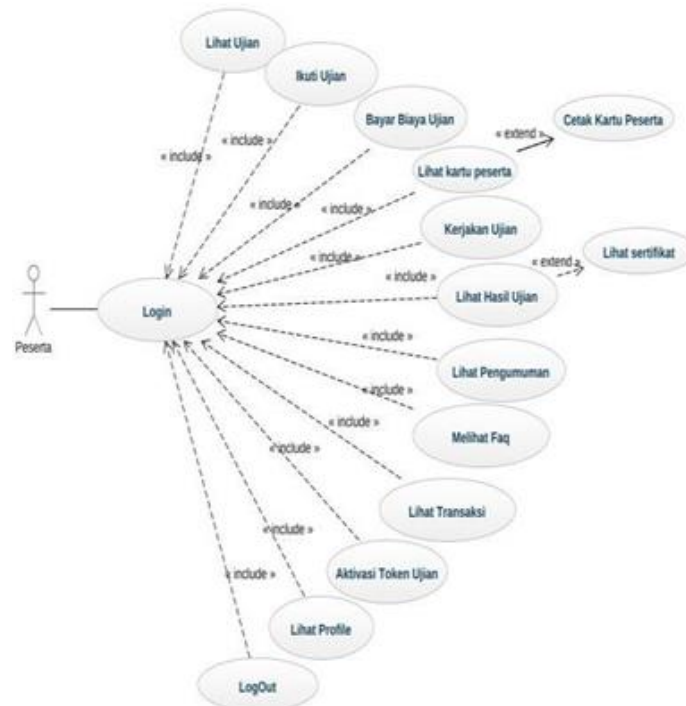
Dalam pengembangan perangkat lunak menggunakan metode *prototype*, analisa kebutuhan merupakan proses indentifikasi permasalahan kemudian memberikan solusi sesuai dengan kebutuhan sistem yang di rencanakan. Dalam penerapannya penulis menggunakan model UML (*Unified Modeling Language*), dimana tahap ini di visualisasikan dengan penggunaan *Use- case Diagram*, *Activity Diagram*, dan *Class Diagram*.

Use-case Diagram

Dalam implementasi *use-case diagram*, terdapat dua aktor yang akan dideskripsikan hal apa saja yang dilakukan aktor dalam sistem, adapun aktor yang dimaksud adalah lembaga bahasa sebagai admin dan mahasiswa sebagai peserta. Berikut *use case diagram* admin dapat di lihat pada gambar 1 dan peserta pada gambar 2.

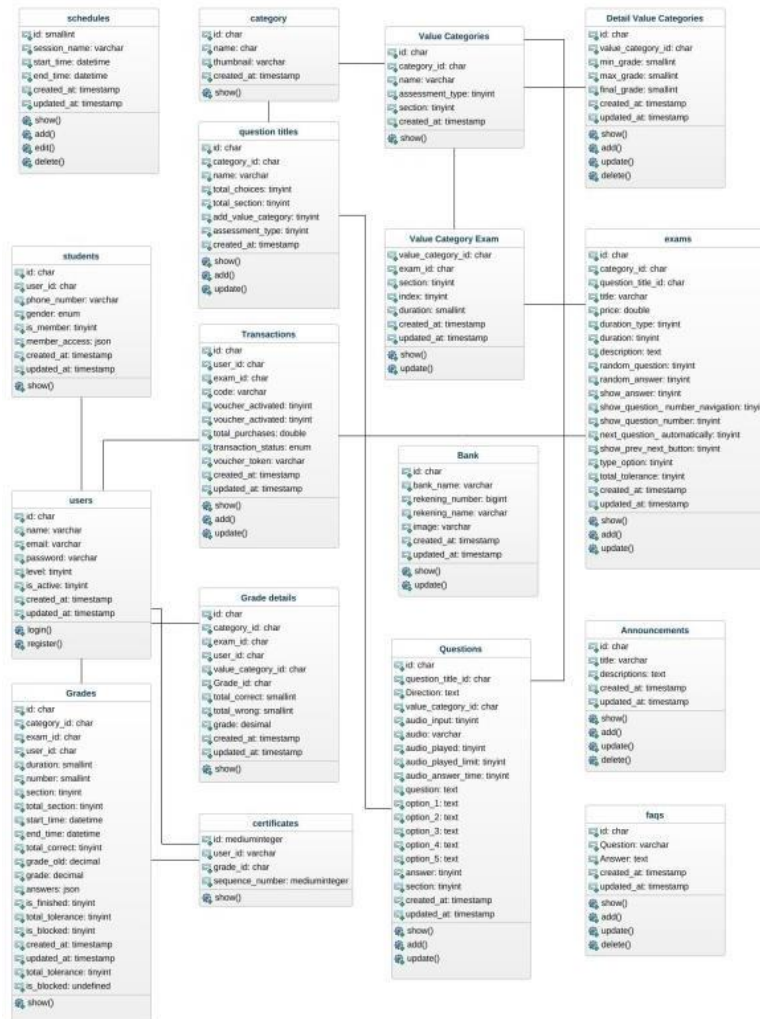


Gambar 1. Use Case Diagram Admin

[illegible]

Class Diagram

Class diagram menggambarkan struktur basis data dan interaksi atau relasi antar tabel yang ada di dalamnya. Berikut merupakan class diagram yang akan dibangun.



Gambar 4. Class Diagram

Membuat dan Evaluasi Prototype

Setelah menyelesaikan analisis sistem dan kebutuhan, telah diperoleh gambaran detail mengenai pembuatan aplikasi tes TOEFL berbasis web. Tahap berikutnya adalah perancangan sistem, yang mencakup analisis algoritma *Fisher Yates*, perancangan antar muka pengguna, dan perancangan basis data.

Analisis Algoritma *Fisher Yates*

Algoritma *Fisher Yates* diambil dari nama Ronald Fisher dan Frank Yates atau dikenal juga dengan nama Knuth Shuffle (diambil dari nama Donald Knuth), adalah sebuah algoritma yang digunakan untuk melakukan permutasi acak dari pada himpunan terhingga. Metode modern yang digunakan untuk menghasilkan permutasi acak dimulai dari angka 1 sampai N. Menurut Ekojono, dkk secara umum pengacakan Algoritma *Fisher-Yates* memiliki konsep sebagai berikut:

1. Tentukan soal dari nomor 1 sampai dengan soal nomor N.
2. Pilih sebuah soal acak K diantara 1 sampai dengan jumlah soal yang belum dicoret.
3. Dihitung dari bawah, coret soal K yang belum dicoret, dan tuliskan soal tersebut di lain tempat.

4. Ulangi langkah 2 dan langkah 3 hingga semua soal sudah tercoret.
5. Urutan soal yang dituliskan pada langkah 3 adalah permutasi acak dari soal awal.

Metode *Fisher Yates* modern melakukan pertukaran antara angka terpilih dengan angka terakhir yang belum dipilih. '*Range*' menunjukkan jumlah angka tersisa, '*roll*' adalah angka acak yang dipilih, '*scratch*' merujuk pada daftar angka tersisa, sementara '*result*' adalah hasil dari permutasi acak. Berikut adalah ilustrasi penerapan algoritma FisherYates dengan asumsi terdapat 10 (sepuluh) soal yang disediakan.

Tabel 1. Ilustrasi Algoritma *Fisher Yates*

<i>Range</i>	<i>Roll</i>	<i>Scratch</i>	<i>Result</i>
		1,2,3,4,5,6,7,8,9,10	
1 - 10	6	1,2,3,4,5,7,8,9,10	6
1 - 9	9	1,2,3,4,5,7,8,10	9,6
1 - 8	1	2,3,4,5,7,8,10	1,9,6
1 - 7	10	2,3,4,5,7,8	10,1,9,6
1 - 6	4	2,3,5,7,8	4,10,1,9,6
1 - 5	7	2,3,5,8	7,4,10,1,9,6
1 - 4	3	2,5,8	3,7,4,10,1,9,6
1 - 3	2	5,8	2,3,7,4,10,1,9,6
1 - 2	5	5	8,2,3,7,4,10,1,9,6
Hasil Pengacakan			5, 8,2,3,7,4,10,1,9,6

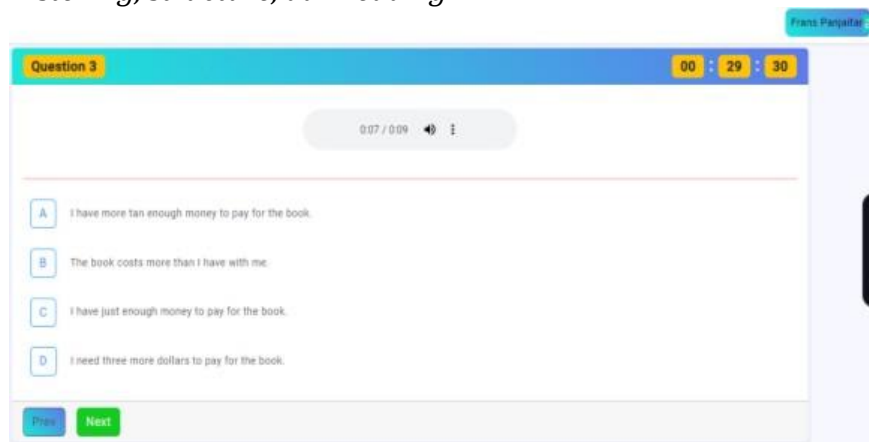
Mengkodekan Sistem

Pada Rancang Bangun Aplikasi Tes TOEFL, algoritma Fisher Yates diterapkan pada *section structure and written* untuk mengacak soal dan pilihan jawabannya. Untuk *section listening* dan *reading*, hanya pilihan jawabannya yang diacak. Berikut potongan kode algoritma Fisher Yates:

```
for($i = 1; $i <= $totalSection; $i++) {
    if($i == 2 && $exam->random_question == 1) {
        $questions = Question::where('section', $i)
->where('question_title_id',
    $exam->question_title_id)->get();
        $totalQuestions = $questions->count();
        // Algoritma Fisher-Yates
        for ($j = $totalQuestions - 1; $j > 0; $j--) {
            $randomIndex = mt_rand(0, $j);
            // Tukar posisi pertanyaan
                $temp=$questions[$j];
            $questions[$j]=$questions[$randomIndex];
            $questions[$randomIndex] = $temp;
        }
    }
};
```

Tampilan Halaman Acak Soal dan Jawaban dengan *Fisher Yates*

Pada tampilan ini, setiap peserta yang telah mendapatkan akses ujian TOEFL akan menerima soal acak untuk bagian *structure*, serta urutan pilihan jawabanya yang berbeda untuk bagian *listening*, *structure*, dan *reading*.



Gambar 5. Halaman *Listening* Acak Jawaban User 1



Gambar 6. Halaman *Structure* Acak Soal dan Jawaban User 1



Gambar 7. Halaman *Reading* Jawaban Acak User 1



Gambar 8. Halaman *Listening* Acak Jawaban User 2



Gambar 9. Halaman *Structure* Acak Soal dan Jawaban User 2



Gambar 10. Halaman *Reading* Acak User 2

Tampilan Output Sertifikat TOEFL Pada tampilan ini, peserta dapat melihat sertifikat berisi nilai yang diperoleh setelah menyelesaikan ujian, dengan rincian hasil untuk setiap bagian.



Gambar 11. Tampilan Output Sertifikat

KESIMPULAN

Berdasarkan hasil penelitian dapat disimpulkan bahwa:

1. Penelitian menunjukkan metode konvensional ujian TOEFL memiliki keterbatasan efisiensi, khususnya dalam eksekusi dan evaluasi jawaban.
2. Algoritma Fisher Yates diterapkan pada aplikasi tes TOEFL untuk mengacak soal di *section structure* serta mengacak jawaban di *section listening, structure* dan *reading*.
3. Dengan adanya aplikasi ini dapat membantu lembaga bahasa dan mahasiswa dalam proses pelaksanaan ujian tes TOEFL.

UCAPAN TERIMA KASIH

Ucapan terimakasih kepada Wali Penulis, Rektor Universitas Katolik Santo Thomas, Dekan Fakultas Ilmu Komputer, Kepala Program Studi Teknik Informatika, Lembaga Bahasa Universitas Katolik Santo Thomas dan bimbingan yang telah diberikan.

DAFTAR PUSTAKA

- [1] Ahmadi dan Hermawan. (2013). E-Business & E-Commerce. Yogyakarta: Andi.
- [2] A.S., Rosa, dan M. Shalahuddin. (2018). Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung: Informatika.
- [3] Buchari, M., Sentinowo, S., & Lantang, O. (2015). Rancang Bangun Video Animasi 3 Dimensi Untuk Mekanisme Pengujian Kendaraan. E-Journal Teknik Informatika. (diakses Januari 04, 2023).
- [4] E. Ekojono, R. Cahyaningrum, and K. S. Batubulan, "Implementasi metode Fisher-Yates Shuffle dan Fuzzy Tsukamoto pada game 2D gopoh berbasis android," J. Inform. Polinema, vol. 4, no. 3, pp. 174-180, May 2018, doi: 10.33795/jip.v4i3.203.
- [5] Ekojono, D. A. Irawati, L. Affandi, and A. N. Rahmanto, "Penerapan algoritma FisherYates pada pengacakan soal game aritmatika," in Prosiding SENTIA, Malang, Indonesia, vol. 9, 2017, Kasus: Unit Pelayanan Terpadu Bahasa Universitas Pembangunan Nasional 'Veteran' Jakarta). 28 Januari 2020.
- [7] Hanif Al Fatta. (2007). Analisis dan Perancangan Sistem Informasi. Andi Offset, Yogyakarta.
- [8] Imam Tantowi, Moh. Ali Albar, Fitri Bimantoro. "Rancang Bangun Sistem Informasi Toefl Pada Pusat Bahasa Universitas Mataram". JTIKA, Vol. 1, No. 2, September 2019.
- [9] Purwanto, Arie & Yasin, Verdi & Haroen, Rachmawaty. (2021). Perancangan aplikasi teknologi informasi helpdesk berbasis web pada instalasi rekam medik dan admisi RSCM Jakarta. JURNAL WIDYA. 2. 129-145.
- [10] Rudi Budi Agung, Rahmadi, Jefry Auliya Rahman, "Rancang Bangun Aplikasi Edukasi Toefl Berbasis Web". Jurnal Vol. 12 No. 2 (2022).
- [11] Strauss dan Forst. 2012. Modul Pembelajaran Uniform Resource Locater (URL). Jakarta. PT. Gramedia.
- [12] Sukanto, R. A., & Shalahuddin, M. (2015). Kolaborasi Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung: Informatika.
- [13] Thomas, U. K. (2022). Universitas Katolik Santo Thomas. Medan: Universitas Katolik Santo Thomas, Internet: <https://www.ust.ac.id/>, (diakses Februari 02, 2023).
- [14] Winarno, Edy dan Ali Zaki. 2014. Pemrograman Web Berbasis HTML 5, PHP, & JavaScript. Bandung: PTpp. 101-106.
- [15] Fakhri Muhtadin, Erly Krisnanik, S.Kom., MM. "Rancang Bangun Sistem Pendaftaran Dan Pretest Toefl Berbasis Web (Studi Elex Media Komputindo).
- [16] Yenny Iskandar (2018). Buku Ajar Pengantar Aplikasi Komputer. Yogyakarta: Deepublish.