



PENERAPAN ALGORITMA KEURANGAN BALAM CPNS

Disusun dan Disajikan Untuk Memenuhi Persyaratan Ujian Akhir
Memperoleh Gelar Sarjana Komputer Pada Fakultas Sains dan Teknologi
Universitas Pektanginan Panca Budi

: M. ...

SKRIPSI

OLEH :

NAMA : ...
NPM : 524370809
PROGRAM STUDI : SISTEM KOMPUTER

FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEKTANGINAN PANCA BUDI
MEDAN
2019

ABSTRAK

MUHAMMAD IRFAN MAHMID

**Penerapan Algoritma Sieve Of Erathonese Dan Linear Congruat Metode
Untuk Mengatasi Kecurangan Dalam Cpns
Tahun 2019**

Pertumbuhan penduduk setiap tahunnya semakin lama semakin meningkat. Pertumbuhan yang semakin meningkat ini juga di imbangi dengan meningkatnya para pencari kerja. Salah satu pekerjaan yang paling diminati masyarakat Indonesia adalah menjadi seorang abdi negara yaitu Pegawai Negeri Sipil. Untuk itu, diperlukan sistem yang terintegrasi dalam sebuah komputer untuk memudahkan proses penyeleksian calon pegawai negeri sipil tersebut. Sistem perekrutan CPNS (Calon Pegawai Negeri Sipil) menggunakan penerapan Algoritma *Sieve of Eratosthenes dan Metode Linear Congruent* di buat dengan memanfaatkan visual basic 2008. Penerapan algoritma ini digunakan untuk membangkitkan bilangan acak, agar setiap peserta ujian mendapatkan soal yang berbeda – beda. Algoritma ini bekerja dengan mengelompokkan data yang pertama kali di dapat dengan menghapus semua bilangan pengaliannya dan secara berulang melakukan hal yang sama sampai seluruh data memiliki kelompok perkaliannya masing-masing. Sistem yang terintegrasi dengan komputer ini juga dapat memudahkan para penyelenggara ujian untuk melakukan rekap data hasil ujian setiap peserta tanpa harus memakan waktu yang lama.

Kata Kunci: CPNS, Algoritma Sieve of Eratosthenes, Metode Linear Congruent, Visual Basic 2008

KATA PENGANTAR

Puji sukur Tuhan yang Maha Esa karena dengan berkat dan kasih anugrahnya-Nya penulis masih diberikan kesehatan sehingga akhirnya penulis dapat merampungkan Skripsi ini.

Tugas akhir disusun berdasarkan hasil penelitian yang dilaksanakan pada kurang lebih 6 (enam) Bulan ini telah dapat di rampungkan dengan judul :

“Penerapan Algoritma Sieve Of Erathonese Dan Linear Congruat Metode Untuk Pengacakan Soal Cpns (Calon Pegawai Negeri Sipil).”

Dalam kesempatan ini, penulis mengucapkan terima kasih yang sebesar-besarnya kepada banyak pihak yang telah membantu dalam penyelesaian penyusunan Tugas Akhir ini. Penulis ingin mengucapkan terima kasih kepada :

1. Orang kedua orang tua yang sangat saya cintai dan saya sayangi Papa dan Mama serta kakak yang selalu memberikan semangat, motivasi, do'a yang tidak hentinya dan membantu dalam segi moril maupun materil sehingga dapat menyelesaikan penyusunan Skripsi ini.
2. Rektor Universitas Pembangunan Panca Budi, Bapak Dr. H. Muhammad Isa Indrawan, S.E, M.M
3. Rektor I, Bapak Ir. Bhakti Alamsyah, M.T, Ph.D
4. Dekan Fakultas Sains dan Teknologi, Ibu Sri Shindi Indira, ST., M.Sc
5. Ketua Program Studi Sistem Komputer, Bapak Dr. Muhammad Iqbal, S.Kom, M.Kom
6. Dosen Pembimbing I, Ibu Leni Marlina, S. Kom., M.Kom
7. Dosen Pembimbing II, Ibu Ika Devi Perwitasari, S. Kom., M.Kom

Penulis juga menyadari bahwa penyusunan Tugas Akhir ini belum sempurna baik dalam penulisan maupun isi disebabkan keterbatasan kemampuan penulis. Oleh karena itu, penulis mengharapkan kritik dan saran yang sifatnya membangun dari pembaca untuk penyempurnaan isi Tugas Akhir ini.

Medan, Agustus 2019

Penulis

Muhammad Irfan Mahmid
NPM. 1524370809

DAFTAR ISI

LEMBAR PENGESAHAN

ABSTRAK

KATA PENGANTAR.....	i
DAFTAR ISI	ii
DAFTAR GAMBAR.....	iv
DAFTAR TABEL	v

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah	1
1.2 Perumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian.....	4
1.5 Manfaat Penelitian.....	4

BAB II LANDASAN TEORI

2.1 Soal CPNS (Calon Pegawai Negeri Sipil).....	5
2.2 Algoritma	5
2.2.1 Algoritma Sieve of Eratoshenes	6
2.2.2 Visual Basic Net.....	8
2.3 <i>Unifed Modeling Language</i>	11
2.3.1 Use Case	12
2.3.2 Class Diagram.....	13
2.3.3 Activity Diagram.....	15
2.3.4 <i>Sequence Diagram</i>	16
2.4 MySQL	17
2.4.1 Sejarah <i>MySQL</i>	17
2.4.2 Keuntungan Database <i>MySQL</i>	18
2.4.3 Penerapan Bahasa <i>MySQL</i>	19

BAB III METODE PENELITIAN

3.1 Tahapan Penelitian	21
3.2 Metode Pengumpulan Data.....	22
3.3 Analisis Sistem	23
3.3.1 Pemodelan Interaksi dengan Pengguna.....	25
3.4 Pembentukan dan Implementasi Bilangan Acak	27
3.4.1 Nilai Input	28
3.4.2 Pembangkit Bilangan Acak Dengan Algoritma Sieve Of Eratoshenes.....	28
3.4.3 Pengacakan Soal-Soal Dengan Linier Congruent Method	32

3.4.4	Algoritma Pembangkit Bilangan Acak dan Pengacakan.....	35
3.5	Rancangan Database.....	37
3.6	Rancangan Output.....	39

BAB IV HASIL DAN PEMBAHASAN

4.1	Kebutuhan Spesifikasi Minimum Hardware dan Software.....	46
4.2	Tampilan Hasil Implementasi.....	47
4.3	Pengujian Sistem.....	53
4.3.1	Evaluasi Sistem.....	58

BAB V PENUTUP

5.1	Kesimpulan.....	60
5.2	Saran.....	60

DAFTAR PUSTAKA

BIOGRAFI PENULIS

SURAT PERNYATAAN

KETERANGAN PLAGIAT CHECKER DARI LPMU

SURAT BEBAS PRAKTIKUM

FORM PENGAJUAN JUDUL

FORM PERMOHONAN MEJA HIJAU

EKSISTENSI BIMBINGAN DOPING 1 DAN 2

LISTING PROGRAM

DAFTAR GAMBAR

Gambar 2.1	Tampilan Awal Visual Studio 2008	8
Gambar 2.2	Menu bar.....	9
Gambar 2.3	Menu Membuat Project Baru	9
Gambar 2.4	Menu Toolbox	10
Gambar 2.5	Server Explorer.....	11
Gambar 2.6	Diagram <i>Use Case</i> Sederhana.....	12
Gambar 3.1	<i>RAD (Rapid Application Development)</i>	21
Gambar 3.2	<i>Use Case Login</i> Pertama.....	25
Gambar 3.3	<i>Use Case</i> Pengacakan Soal	25
Gambar 3.4	<i>Use Case</i> Sistem Pengacakan.....	26
Gambar 3.5	Pembentukan dan Implementasi Pembangkit Bilangan Acak	27
Gambar 3.6	<i>Flowchart</i> Algoritma <i>Sieve Of Eratoshenes</i>	32
Gambar 3.7	Algoritma <i>Sieve Of Eratoshenes</i> dan <i>Linier Congruent Method</i>	33
Gambar 3.8	<i>Form</i> Utama	40
Gambar 3.9	<i>Form</i> Peserta	41
Gambar 3.10	<i>Form</i> Soal	42
Gambar 3.11	<i>Form</i> Nilai	44
Gambar 3.12	<i>Form</i> Evaluasi	45
Gambar 4.1	Tampilan <i>Form User</i>	47
Gambar 4.2	Tampilan <i>Form Admin</i>	48
Gambar 4.3	Tampilan <i>Form Login User</i>	48
Gambar 4.4	Tampilan Halama Depan <i>Admin</i>	49
Gambar 4.5	Tampilan Halaman Depan user.....	49
Gambar 4.6	Tampilan <i>Form</i> Soal	50
Gambar 4.7	Tampilan <i>Form</i> Jawab Soal Dengan Soal Yang Telah Di Acak.....	51
Gambar 4.8	Tampilan <i>Form</i> Menu Admin	52
Gambar 4.9	Tampilan <i>Form Score</i>	53
Gambar 4.10	Daftar Soal CPNS.....	54
Gambar 4.11	Percobaan 1 Soal 1	54
Gambar 4.12	Percobaan 1 Soal 2.....	55
Gambar 4.13	Percobaan 1 Soal 3	55
Gambar 4.14	Percobaan 2 Soal 1	56
Gambar 4.15	Percobaan 2 Soal 2.....	57
Gambar 4.16	Percobaan 2 Soal 3	57

DAFTAR TABEL

Tabel 2.1 Simbol-Simbol <i>Use Case</i>	13
Tabel 2.2 Simbol-Simbol <i>Class Diagram</i>	14
Tabel 2.3 Simbol-Simbol Diagram Activity	15
Tabel 2.4 <i>Sequence Diagram</i>	16
Tabel 2.5 Daftar Pernyataan Dasar <i>MySQL</i>	20
Tabel 3.1 Langkah Pertama	29
Tabel 3.2 Langkah Kedua	30
Tabel 3.3 Langkah Ketiga	30
Tabel 3.4 Langkah Keempat	30
Tabel 3.5 Langkah Kelima dan seterusnya	31
Tabel 3.6 Tabel Peserta	38
Tabel 3.7 Tabel Soal	38
Tabel 3.8 Tabel Nilai	40

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pembangkit bilangan acak adalah bilangan asli yang tepat hanya memiliki dua faktor, yaitu 1 dan bilangan itu sendiri. Sampai dengan abad kesembilan belas Masehi kebanyakan matematikawan menganggap 1 sebagai bilangan prima. Pada waktu itu, sebagian besar tulisan yang dihasilkan masih memasukkan 1 sebagai bilangan prima yang sah. Perubahan yang membawa 1 tidak lagi dianggap sebagai bilangan prima adalah adanya kebutuhan untuk dapat menyatakan "setiap angka dapat difaktorkan menjadi bilangan prima yang unik". (Rinaldi, M., & Sianturi, R. D., 2018)

Seperti yang kita ketahui dalam proses pembuatan soal ujian masih secara manual untuk memberikan soal-soal susah untuk dilakukan memberikan pertanyaan yang berbeda-beda. Misalkan ada 200 peserta bagaimana membuat pengacakan maka tentu dilakukan proses berulang-ulang ternyata masalah tersebut dapat diatasi dengan pengacakan didalam ilmu komputer dengan tujuan agar pengguna tidak mendapatkan pertanyaan yang sama berulang kali dibutuhkan sebuah algoritma pengacakan yaitu *Linear Congruent Method (LCM)*. *Linear Congruent Method* merupakan algoritma pengacakan yang menggunakan metode bilangan acak. Oleh karena itu, algoritma LCM memberikan bilangan acak untuk pengacakan soal.

Pengacakan soal menjadi tren dalam penyelesaian terjadinya kecurangan didalam menjawab soal-soal, seperti soal-soal Ujian Nasional, soal-soal Calon Pegawai Negeri

Sipil bahkan masuk ke Perguruan Tinggi Negeri dan lain-lain maka perlu peneliti memaparkan bagaimana mengimplementasikan tersebut dari awal yakni pembangkit bilangan acak hingga ahir pengacakan tersebut dapat diterapkan.

Algoritma *Sieve of Eratosthenes* merupakan algoritma yang digunakan untuk mengeliminasi angka yang bukan bilangan prima dari kumpulan angka yang ada. Proses ini cukup baik sebagai algoritma yang bisa diterapkan pada algoritma kriptografi yang digunakan. Banyak penelitian dilakukan untuk mendapatkan metode yang paling cepat dalam pembangkitan bilangan prima. mencapai kecepatan yang dianggap ideal pembangkitan bilangan prima tidak dapat dilakukan, sehingga banyak penelitian dilakukan untuk mendapatkan metode yang paling cepat dalam pembangkitan bilangan prima. Algoritma *sieve of eratosthenes* adalah sebuah algoritma klasik untuk menemukan seluruh bilangan prima sampai ke sebuah N yang ditentukan. (Aziz, T., 2018)

Penelitian tentang pembangkit bilangan prima dengan komputer paralel dilakukan oleh Hwang Soonwook dan Kyusik Chung dengan metode *sieve of eratosthenes* terbaik.

Oleh karena alasan itulah, penulis terdorong untuk meneliti dengan judul **“Penerapan Algoritma Sieve of Erathonese Dan Metode Linear Congruat Metode Untuk Mengatasi Kecurangan Dalam CPNS”**.

1.2 Perumusan Masalah

Adapun Perumusan masalah yang diambil didalam sebagai berikut:

- a. Bagaimana penerapan algoritma *Sieve of Eratosthenes* untuk pembangkit bilangan acak dari sekumpulan angka.
- b. Bagaimanan pembangkit bilangan acak diterapkan pada aplikasi pengacakan soal–soal dengan *Linear Congruent Method* ?
- c. Bagaimana merancang program pembangkit bilangan acak dengan penerapan algoritma *Sieve of Eratosthenes* dan diterapkan dengan aplikasi pengacakan ?

1.3 Batasan Masalah

Sesuai dengan latar belakang pemilihan judul di atas, maka yang menjadi permasalahan di dalam Skripsi ini adalah sebagai berikut:

- a. Program yang dirancang untuk menampilkan deret angka dan mengeliminasi angka yang bukan bilangan prima menggunakan algoritma *sieve of Eratosthenes*.
- b. Pembuktian dengan aplikasi pengacakan soal–soal menerapkan *Linear Congruent Method*.
- c. Bahasa pemrograman yang digunakan adalah *Visual-Basic 2008*.

1.4 Tujuan Penelitian

Tujuan yang ingin dicapai melalui penulisan skripsi ini adalah:

- a. Untuk menerapkan algoritma *Sieve of Eratosthenes* untuk proses pembangkit bilangan acak.
- b. Untuk proses pengacakan soal-soal dengan *Linear Congruent Method*.
- c. Untuk merancang program pembangkit bilangan prima dengan penerapan algoritma *sieve of eratosthenes* dan mengembangkan dengan aplikasi pengacakan dengan LCM.

1.5 Manfaat Penelitian

Dengan kemampuan aplikasi ini, maka diharapkan aplikasi ini akan sangat bermanfaat antara lain:

- a. Memahami secara lebih dalam bagaimana menggunakan bahasa pemrograman .
- b. untuk menyelesaikan proses pembangkit bilangan acak dan proses simulasi pengacakan secara aritmatika pada komputer.
- c. Menambah referensi penelitian mengenai pembangkit bilangan acak dan pengacakan.

BAB II

LANDASAN TEORI.

2.1 Soal CPNS (Calon Pegawai Negeri Sipil)

Menjawab Soal CPNS (Calon Pegawai Negeri Sipil) merupakan salah satu tahapan yang harus dilalui bagi peserta yang telah lolos seleksi administrasi penerimaan Calon Pegawai Negeri Sipil. Kemendikbud (Kementrian Pendidikan dan Kebudayaan) sebagai penyedia soal CPNS ini, telah membuat ribuan bank soal dan sistem seleksi penerimaan Calon Pegawai Negeri Sipil ini sudah diintegrasikan dengan menggunakan sebuah komputer atau yang sering dikenal dengan istilah CAT (*Computer Assist Test*) dimana setiap peserta akan menjawab soal secara langsung melalui komputer yang telah disediakan oleh panitia. Kepala Biro Humas BKN (Badan Kepegawaian Negara) Mohammad Ridwan, “Sistem CAT ini dibuat untuk menghindari kecurangan oleh setiap peserta dan menjamin keakuratan penyeleksian yang dilakukan karena setiap peserta akan menjawab soal–soal yang berbeda-beda”. (www.liputan6.com, diakses pada 4 April 2019)

2.2 Algoritma

Algoritma adalah sebuah prosedur komputasi yang mentransformasikan sejumlah input menjadi sejumlah output. Sebuah algoritma dikatakan “benar (*correct*)” jika untuk setiap inputnya menghasilkan *output* yang benar pula. Dalam hal ini algoritma dapat digunakan sebagai metode untuk mengetahui langkah-langkah

secara urut untuk mencapai tujuan. Setiap algoritma memiliki perbedaan dalam mencapai tujuan. (Jenni Veronika, 2018)

2.2.1 Algoritma Sieve of Eratoshenes

Eratosthenes (276-194 S.M) adalah petugas perpustakaan ketiga dari perpustakaan terkenal di Alexandria dan adalah seorang sarjana yang sangat hebat. Eratosthenes dikenang dengan pengukurannya terhadap keliling dari bumi, memperkirakan jarak antara bumi dengan matahari dan bulan, dan, dalam matematika, untuk penemuan dari sebuah algoritma untuk mencari bilangan-bilangan prima, yang dikenal sebagai *Algoritma Sieve Of Eratosthenes*. Banyak penelitian dilakukan untuk mendapatkan metode yang paling cepat dalam pembangkitan bilangan prima. mencapai kecepatan yang dianggap ideal pembangkitan bilangan prima tidak dapat dilakukan, sehingga Banyak penelitian dilakukan untuk mendapatkan metode yang paling cepat dalam pembangkitan bilangan prima. Penelitian tentang pembangkit bilangan prima dengan komputer paralel dilakukan oleh Hwang Soonwook dan Kyusik Chung pada tahun 2006 dengan metode *Sieve of Eratosthenes* terbaik. (D Abdillah, H Swanda, 2018)

Dalam matematika, bilangan prima adalah bilangan asli yang lebih besar dari angka 1, yang faktor pembagiannya adalah 1 dan bilangan itu sendiri. 2 dan 3 adalah bilangan prima. 4 bukan termasuk bilangan prima karena 4 dapat dibagi 2. Cara paling sederhana untuk menentukan bilangan prima yang lebih kecil dari bilangan tertentu adalah dengan menggunakan algoritma *Sieve Of Eratoshenes* .

Algoritma *Sieve Of Eratoshenes* pembangkit bilangan prima yang dapat dipilih oleh pengguna dan yang paling cepat yang dilakukan berulang rulang dibandingkan dengan metode lain sekalipun metode-metode tersebut memiliki kelebihan dan kelemahannya masing-masing. Algoritma *Sieve Of Eratosthenes* adalah sebuah algoritma klasik untuk menemukan seluruh bilangan prima sampai ke sebuah N yang ditentukan. Seperti gambar berikut :

$$\Theta(n(\log n) \log \log n) \text{ bit operasi)}$$

Yaitu mulai dengan array of integer terendah dan belum dicoret dari 2 ke N . Integer pertama yang belum dicoret yaitu 2, adalah bilangan prima pertama. Coret seluruh kelipatan dari bilangan prima ini. Ulangi pada integer selanjutnya yang belum dicoret. Sebagai contoh, berikut adalah array pada awalnya:

23456789101112131415161718192021222324252627 Karena 2 belum

dicoret, maka 2 adalah bilangan 2 pertama kita. Kita coret

seluruh kelipatan 2, yaitu 4, 6, 8, 10, 12, dan seterusnya. 2 3 4 5 6 7 8 9 10 11 12 13
14 15 16 17 18 19 20 21 22 23 24 25 26 27 .

Integer selanjutnya yang belum dicoret adalah 3, maka 3 adalah prima dan kita coret seluruh kelipatan 3, seperti 6, 9, 12, dan seterusnya. 2 3 4 5 6 7 8 9 10 11 12 13
14 15 16 17 18 19 20 21 22 23 24 25 26 27 . 5 adalah bilangan prima selanjutnya dan kita mencoret seluruh kelipatan 5. Satu-satunya bilangan yang dicoret dalam range ini adalah 25. . 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 .
Maka kita mendapatkan bilangan prima selanjutnya yaitu 7, 11, 13, 17, 19,

dan 23. Maka kita mendapatkan bilangan prima selanjutnya yaitu 7, 11, 13, 17, 19, dan 23. (R Saleh, I Imelda, 2018)

2.2.2 Visual Basic Net

Visual-Basic 2008 merupakan salah satu paket bahasa pemrograman dari Visual Studio 2008. Banyak fasilitas yang akan kita dapatkan melalui rilis Visual-Basic versi ini. Visual-Studio 2008 juga merupakan *software* membuat aplikasi *windows*, melalui *software* bisa membuat aplikasi seperti aplikasi *Database*, aplikasi *Inventory*, dan sebagainya. Kebanyakan orang lebih suka menyebut sebuah aplikasi sebagai sebuah program atau software, padahal ketiga istilah ini memiliki arti yang sama. Visual-Basic untuk *windows application* secara *default* telah terdapat sebuah *form*. *Form* tersebut bernama *Form1*. (M Monier, MM El Mahdy, 2015)



Gambar 2.1 Tampilan Awal Visual Studio 2008

Sumber: M Monier, MM El Mahdy, 2015

Adapun elemen-elemen di dalam Visual Basic.Net 2008 sebagai berikut :

1. Menu

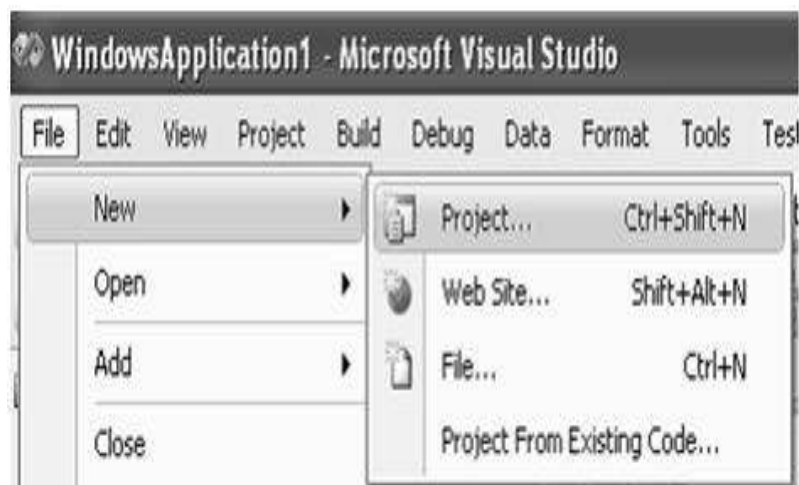
Menu adalah bagian dari IDE yang terdiri dari perintah-perintah untuk mengatur IDE, mengembangkan, memelihara dan mengeksekusi program.



Gambar 2.2 Menu Bar

Sumber : M Monier, MM El Mahdy, 2015

Menggunakan menu atau pilihan pada *menu*, Anda tinggal mengklik pada menu atau pilihan yang akan dijalankan. Sebagai contoh, untuk membuat sebuah *project* yang baru maka pilih menu *File | New | Project*.



Gambar 2.3 Menu Membuat *Project* Baru

Sumber : M Monier, MM El Mahdy, 2015

2. *Toolbar*

Toolbar fungsinya sama seperti fungsi dari menu, hanya saja pada *toolbar* pilihan-pilihan berbentuk *icon*.

3. *Toolbox*

Toolbox adalah tempat dimana kontrol-kontrol dan komponen-komponen diletakkan. Kontrol dan komponen yang terdapat pada *toolbox* dipakai dalam pembuatan program aplikasi.



Gambar 2.4 Menu *Toolbox*

Sumber : M Monier, MM El Mahdy, 2015

4. *Server Explorer*

Server Explorer adalah bagian tempat untuk mengatur hal-hal yang berhubungan dengan server dan *database* server.



Gambar 2.5 Server Explorer

Sumber : M Monier, MM El Mahdy, 2015

2.3 *Unified Modeling Language*

Unified Modeling Language (UML) adalah keluarga notasi grafis yang didukung oleh model tunggal yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO) dan diagram yang akan digunakan dalam UML adalah *use case*, *class diagram*, *activity diagram* dan *sequence diagram* (Ade Hendini, 2016)

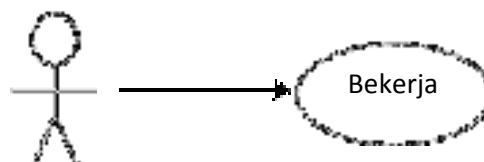
Menggunakan UML dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan class dan operation dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C. (Ade Hendini, 2016)

2.3.1 Use Case

Use case adalah teknik merekam persyaratan fungsional sebuah sistem. *Use case* mendeskripsikan intraksi tipikal antara para pengguna sistem dengan sistem itu sendiri, dengan memberi sebuah narasi tentang bagaimana sistem tersebut digunakan. Walaupun menjelaskan kegiatan, namun *use case* hanya menjelaskan apa yang dilakukan oleh *actor* dan sistem bukan bagaimana *actor* dan sistem melakukan kegiatan tersebut, (Tri Astoto Kurniawan, 2018), yaitu:

1. *Use-case Konkret* adalah *use case* yang dibuat langsung karena keperluan *actor*. *Actor* dapat melihat dan berinisiatif terhadapnya.
2. *Use-case Abstrak* adalah *use case* yang tidak pernah berdiri sendiri. *Use case* abstrak senantiasa termasuk didalam (*include*), diperluas dari (*extend*) atau memperumum (*generalize*) *use case* lainnya.

Menggambarkannya dalam *use case* model biasanya digunakan *association relationship* yang memiliki *stereotype include*, *extend* atau *generalization relationship*. Adapun contoh gambar *use case*, sebagai berikut:




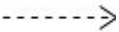


Gambar 2.6 Diagram Use Case Sederhana

Sumber : Tri Astoto Kurniawan, 2018

Simbol diatas dapat terlihat seorang *Actor* sedang berkerja. Simbol simbol *use case* dapat di lihat pada tabel berikut:

Tabel 2.1 Simbol-Simbol Use Case

No	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .



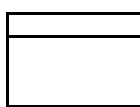




Sumber: Tri Astoto Kurniawan, 2018

2.3.2 Class Diagram

Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara sistem, *class diagram* tidak hanya digunakan secara luas tetapi juga memiliki banyak konsep pemodelan. Dimana *class*

diagram juga menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan untuk menampilkan kelas-kelas dan paket-paket di dalam system objek tersebut. (Ade Hendini, 2016)

Tabel 2.2. Simbol-Simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku.
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>)
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

(Sumber : Ade Hendini, 2016)





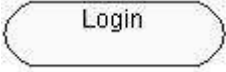
2.3.3 Activity Diagram

activity diagram (diagram aktifitas) adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. Dalam beberapa hal, diagram ini memainkan peran mirip sebuah diagram alir, tetapi perbedaan prinsip antara diagram ini dan notasi diagram alir adalah diagram ini mendukung *behavior paralel*.

Kelebihan diagram aktifitas dibandingkan dengan diagram flowchart adalah adanya dukungan konkurensi (pelaksanaan aktivitas secara bersamaan), pengiriman pesan dari *swimlane* (pelaku/penanggung jawab aktifitas). Konkurensi diawali dengan sebuah garis tebal horizontal juga diakhiri dengan garis tebal horizontal. (Ade Hendini, 2016)

Simbol-simbol *Diagram Activity* dapat di lihat pada tabel berikut:

Tabel 2.3 Simbol-Simbol *Diagram Activity*

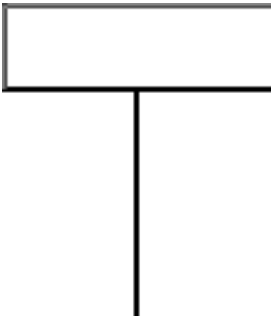
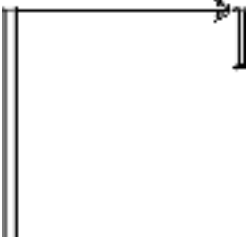
Simbol	Keterangan
	Titik Awal atau permulaan.
	Titik Akhir atau akhir dari aktivitas.
	<i>Decision</i> , atau pilihan untuk mengambil keputusan.
	Arah tanda panah alur proses.
	Aktiviti, atau aktivitas yang dilakukan oleh aktor.

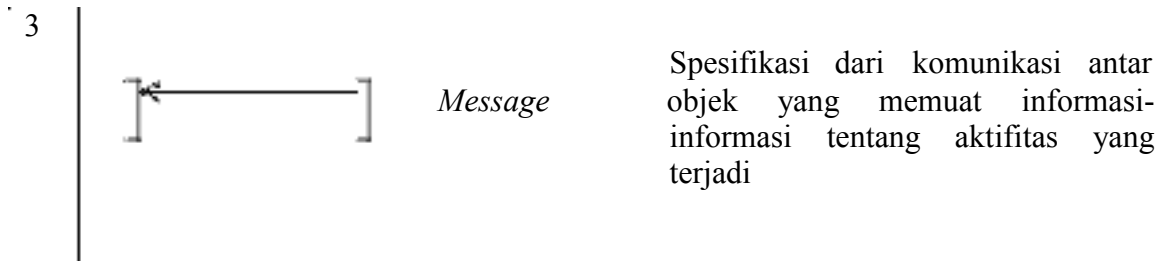
(Sumber : Ade Hendini, 2016)

2.3.4 *Sequence Diagram*

Sequence diagram adalah suatu diagram yang menggambarkan interaksi antar objek dan mengindikasikan komunikasi diantara objek-objek tersebut. Diagram ini juga menunjukkan serangkaian pesan yang dipertukarkan oleh objek-objek yang melakukan suatu tugas atau aksi tertentu. Objek-objek tersebut kemudian diurutkan dari kiri kekanan, aktor (*user*) yang menginisiasikan interaksi biasanya ditaruh dipaling kiri dari diagram. (Ade Hendini, 2016)

Tabel 2.4 *Sequence Diagram*

No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi



(Sumber : Ade Hendini, 2016)

2.4 *MySQL*

MySQL adalah database server yang sangat populer dan banyak digunakan untuk menangani data yang disajikan di halaman web. Database server adalah server yang berfungsi untuk menangani database. Database adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data. *MySQL* tergolong sebagai *database* yang relasional. Pada model ini, data dinyatakan dalam bentuk dua dimensi yang secara khusus dinamakan tabel. Tabel tersusun atas baris dan kolom. (Bunafit Nugroho, 2015)

Sehingga *MySQL* dapat diartikan sebagai sebuah tabel dengan struktur dalam bahasa pemrograman yang mudah untuk dioperasikan sehingga dapat dijadikan sebagai tempat penyimpanan yang besar sekalipun diakses oleh banyak penggunaan sekalipun untuk data yang besar dan kepentingan yang lebih luas merupakan kelebihan dari *MySQL* yang banyak digunakan oleh perkantoran dan organisasi.

2.4.1 Sejarah *MySQL*

MySQL dikembangkan sekitar tahun 1994 oleh perusahaan pengembang *software* dan konsultan *database* bernama *MySQL-AB* yang bertempat di Swedia. *MySQL* adalah satu dari sekian banyak sistem *database*, merupakan terobosan solusi

yang tepat dalam aplikasi *database*. Didukung oleh ribuan bahkan jutaan komunitas pengguna di internet yang siap membantu. Antar muka (*front end*) untuk aplikasi *database MySQL* dapat menggunakan bahasa pemrograman umum seperti bahasa pemrograman java, C/C++, *MSVisual Basic* ataupun *Borland Delphi*, hasil akhir dari model aplikasi yang dihasilkan adalah aplikasi *Client/Server* (Bunafit Nugroho, 2015)

Umumnya akses kepada *databaseMySQL* dari bahasa pemrograman jika dilingkungan *windows* menggunakan *MyODBC*, *driver* koneksi *database* dengan menggunakan standar *ODBC*. *MyODBC* adalah *driver* untuk melakukan akses *databaseMySQL* melalui standar akses *database Open Database Connectivity (ODBC)*. *ODBC* adalah antar muka pemrograman yang memungkinkan aplikasi-aplikasi untuk mengakses data dalam sistem manajemen *database* dengan menggunakan *Structured Query Language (SQL)* sebagai standar untuk mengakses data. (Bunafit Nugroho, 2015)

2.4.2 Keuntungan Database MySQL

Keuntungan yang diperoleh dengan menggunakan *database MySQL* sebagai berikut:

1. *Portability*

MySQL dapat berjalan stabil pada berbagai sistem operasi diantaranya *Windows, Linux, FreeBSD, Mac OS X Server, HP-UX* dan masih banyak lagi.

2. *Multiuser*

MySQL dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.

3. *Coloum Types*

MySQL memiliki tipe kolom yang sangat kompleks seperti *Signed/Unsigned, Integer, Float, Double, Char, Varchar, Text, Blob, Date, Time, Datetime, Timestamp* dan *Year*.

4. *Command and Functions*

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam *Query*.

5. *Security*

MySQL memiliki beberapa lapisan *sekuritas* seperti *Level Subnetmask*, nama *Host*, dan izin akses *User* dengan sistem perizinan yang mendetail dan *password* terenkripsi. (Bunafit Nugroho, 2015)

2.4.3 Penerapan Bahasa *MySQL*

Bahasa *MySQL* merupakan bahasa yang digunakan untuk mengakses basis data yang tergolong relasional dan selanjutnya disebut *Structured Query Language (SQL)*. *SQL* tidak terbatas hanya untuk mengambil data (*query*), tetapi juga dapat dipakai untuk menciptakan tabel, menghapus tabel, menambah data ke tabel, menghapus data pada tabel, mengganti data pada tabel dan operasi lainnya. (Bunafit Nugroho, 2015)

Tabel 2.5 Daftar Pernyataan Dasar *MySQL*

Pernyataan	Keterangan
<i>Select</i>	untuk mengambil data
<i>Insert</i>	untuk menambah data
<i>Update</i>	untuk mengganti data
<i>Delete</i>	untuk menghapus data
<i>Create database</i>	untuk menciptakan <i>database</i>
<i>Drop table</i>	untuk menghapus tabel
<i>Drop database</i>	untuk menghapus <i>database</i>

(Sumber : Bunafit Nugroho, 2015)

BAB III

METODE PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian ada skripsi ini digambarkan dengan siklus RAD (*Rapid Application Development*).



Gambar 3.1 RAD (*Rapid Application Development*)

(Sumber: Kendall, 2010)

1). *Requirements Planning* (Perencanaan Syarat – Syarat)

Dalam fase ini, pengguna dan penganalisis bertemu untuk mengidentifikasi tujuan – tujuan aplikasi dan sistem serta untuk mengidentifikasi syarat - syarat informasi yang ditimbulkan dari tujuan

sistem yang dibuat. Orientasi pada fase ini adalah mendapatkan informasi dalam pembuatan sistem.

2). *RAD design workshop*

Fase ini adalah fase untuk merancang dan memperbaiki yang bisa digambarkan sebagai workshop. Selama workshop design RAD, pengguna merespon *prototype* yang ada dan penganalisis memperbaiki modul – modul yang direncanakan berdasarkan respon pengguna.

3). Implementasi

Pada fase Implementasi ini, penerapan aplikasi pengacak soal ujian CPNS (Calon Pegawai Negeri Sipil) dengan menggunakan *Visual Basic*.

3.2 Metode Pengumpulan Data

Pada perancangan dan penyusunan skripsi ini, digunakan beberapa metode untuk mengumpulkan data, diantaranya sebagai berikut:

1. Literatur (Studi pustaka)

Studi pustaka dilakukan untuk mendapat referensi mengenai dasar keseluruhan sistem yang akan dibangun dan beberapa metode yang digunakan.

Studi pustaka juga digunakann untuk mendapatkan ide tentang pengembangan selanjutnya dalam upaya perbaikan terhadap sistem yang sudah berjalan.

2. Wawancara dan Observasi

Instrumen penelitian dengan teknik wawancara dan observasi, dilakukan dengan cara mengumpulkan serangkaian data terkait dengan permasalahan yang sering terjadi pada saat ujian CPNS (Calon Pegawai Negeri Sipil) berlangsung.

3.3 Analisis Sistem

Permasalahan diuraikan dalam sebuah gambaran atau *design* yang tepat mengenai cara kerja sebuah sistem tentang pembangkit bilangan acak serta diterapkan ke-pengacakan. Gambaran yang bagus mengenai sebuah sistem berarti juga menghasilkan sebuah sistem yang baik juga. Analisa sistem ini membahas mengenai bagaimana proses kerja dari sebuah sistem dari awal sampai akhir tentang pembangkit bilangan acak yaitu proses pembangkitan untuk simulasi dengan algoritma *sieve of eratosthenes*.

Analisa sebuah sistem juga sangat mempengaruhi kinerja dari sebuah sistem yang dihasilkan. Karena kinerja sebuah sistem sangat dipengaruhi oleh seberapa jauh permasalahan yang akan diselesaikan oleh sebuah sistem. Semakin luas cakupan masalah sistem, maka semakin tidak maksimal juga kinerja sebuah sistem sehingga

pada penulisan ini hanya membahas permasalahan yang telah ditetapkan dengan cara yang telah ditentukan untuk penyelesaian tujuan dapat berjalan sesuai dengan yang diharapkan dan tidak keluar dari hasil perencanaan yaitu sebelumnya penulis sudah menyampaikan di pendahuluan mulai dari rumusan masalah dan lain-lain dengan lengkap sehingga tidak keluar dari bahasan tersebut.

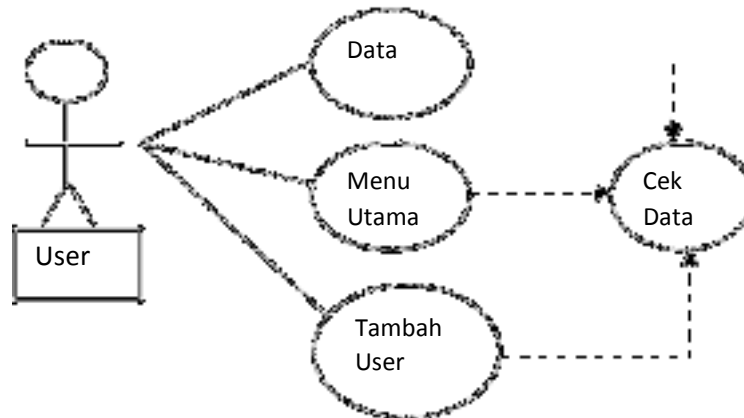
Logika dari *algoritma sieve of eratosthnes* pada pembangkit bilangan acak dengan menentukan nilai *array*[1;0] atau boleh dipahami hanya satu baris kemudian membentuk pada nilai tersebut dan di implementasi dengan metode *linear congruated*. Hal tersebut dilakukan karna penulis merasa agar *algoritma sieve of eratosthenes* benar-benar dimanfaatkan langsung sebagai pengacakan agar membuktikan algoritma tersebut benar sesuai dengan pengetahuan yang ada.

Penjabaran proses agar lebih jelas dibawah ini akan diuraikan satu persatu langkah-langkah dari awal memasukan nilai *array* untuk pembangkit bilangan acak kemudian menghasilkan sebuah proses pengacakan, dimana dalam kasus ini penulis tertarik melakukan pengacakan soal, karena pengacakan soal menjadi tren di beberapa refrensi yang penulis temui seperti yang diurasikan di landasan teori yakni pemanfaatan *linear congruent method* untuk pengacakkan soal. Bukan berarti penulis ingin memperdalam kajian *linear conrent method* namun penulis merasa agar karya ilmiah ini benar – benar terpaparkan dengan baik sampai ketahap pengacakan sekalipun algoritma yang penulis gunakan hanya untuk pembangkit bilangan acak.

3.3.1 Pemodelan Interaksi dengan Pengguna

Pemodelan interaksi dengan pengguna yaitu model diagram yang akan berjalan oleh pengguna sehingga pengguna memahami dengan mudah proses interaksi.

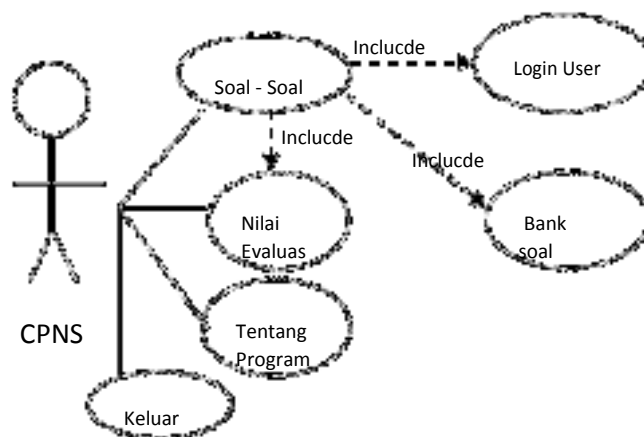
a. Use Case Login Pengguna



Gambar 3.2 Use Case Login Pengguna

Use case diatas menampilkan hubungan antara *user* ke sistem dimana pada saat login, ada beberapa data yang harus di input oleh *user* untuk diverifikasi apakah *user* tersebut telah terdaftar kedalam sistem.

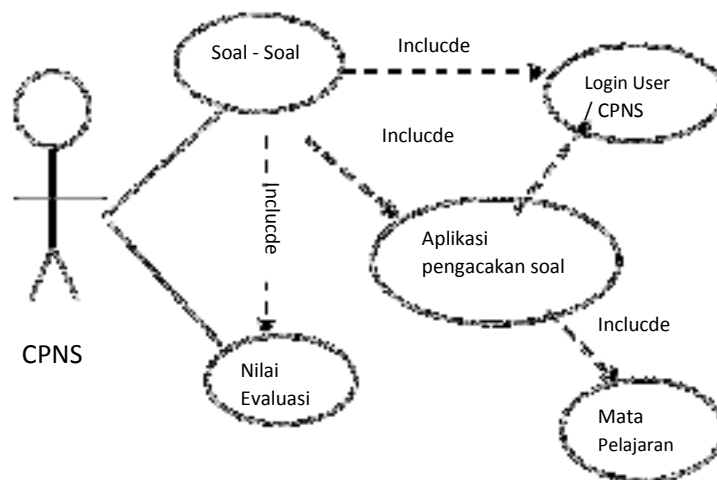
b. Use Case Pengacakan Soal



Gambar 3.3 Use Case Pengacakan Soal

Setelah *login* berhasil dengan data yang telah terverifikasi, maka sistem akan menampilkan soal – soal kepada *user*, dimana kumpulan soal tersebut telah tersimpan pada *database* bank soal. Setelah *user* berhasil menjawab keseluruhan dari soal yang telah di tampilkan sistem, maka secara otomatis tampilan aplikasi yang dibuat akan menampilkan nilai hasil dari evaluasi soal. Pada tampilan ini juga disajikan beberapa panduan dalam pengerjaan soal pada menu tentang dan menu keluar untuk meninggalkan aplikasi.

c. *Use Case* Sistem Pengacakan

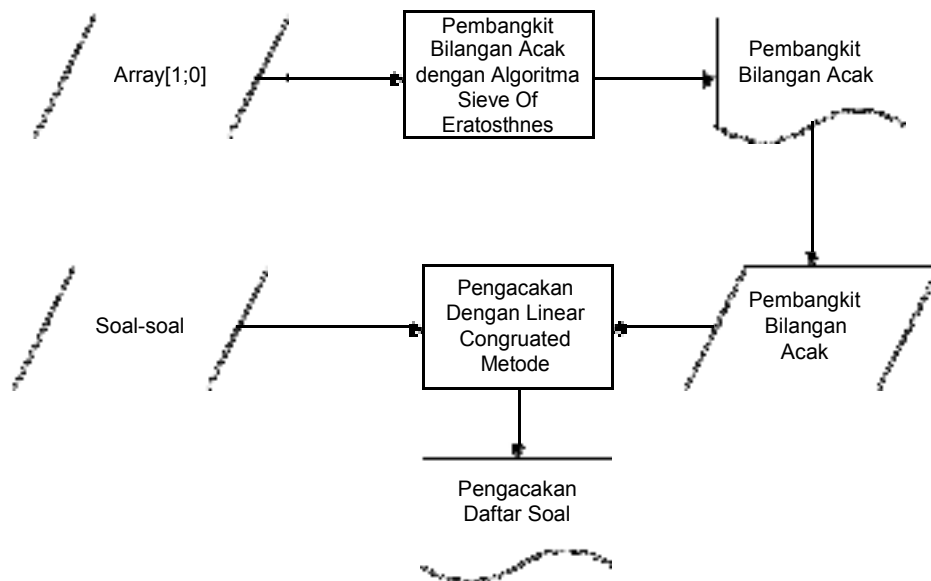


Gambar 3.4 *Use Case* Sistem Pengacakan

Keseluruhan dari sistem yang berjalan, dimana soal – soal yang disajikan berasal dari bank soal yang sebelumnya telah tersimpan dalam *database*. Aplikasi pengacakan soal secara otomatis mengambil soal – soal yang telah dibuat dan ditampilkan pada *user* dengan urutan yang berbeda untuk setiap kali seorang *user* melakukan *login* kedalam sistem.

3.4 Pembentukan dan Implementasi Bilangan Acak

Pembentukan bilangan acak merupakan program dalam *computer* untuk perhitungan aritmatika pembentukan bilangan acak terbentuk dengan algoritma *sieve of erathostenes* sehingga dapat digunakan dalam berbagai kepentingan seperti pembangkit bilangan acak untuk pengacakan soal dalam ujian, pengacakan daftar lagu, pengacakan kartu dalam permainan dan lain-lain dengan langkah lain tertentu sehingga penulis perlu mengutarakan hasil yang diberikan dalam langkah ini penulis mencari algoritma *sieve of erathostenes* kemudian manfaat hasil tersebut kedalam metode *linear congruated* kedalam pengacakan dalam kasus ini pengacakan seperti yang penulis uraikan sebelumnya. Berikut alur sistem penelitian yang digunakan.



Gambar 3.5 Pembentukan dan Implementasi Pembangkit Bilangan Acak

Sehingga struktur untuk aplikasi diatas adalah sebagai berikut:

1. *Array*[1-0] : nilai input *array* sebanyak 1 baris sesuai dengan jumlah yang diacak.
2. Pembangkit bilangan acak dengan algoritma *sieve of eratosthenes*: proses pembentukan pembangkit bilangan acak.
3. Pembangkit bilangan acak : hasil dari proses algoritma *sieve of eratoshnenes* yaitu hasil pembangkit bilangan acak.
4. Soal - soal : jumlah soal dengan proses penomoran berdasarkan banyaknya soal.
5. Pengacakan soal - soal dengan *linear congruated method* yaitu proses pengacakan berdasarkan nomor.
6. Hasil pengacakan soal untuk hasil soal yang telah diacak.

3.4.1 Nilai input

Pembentukan nilai yakni sebuah nilai *array* dengan satu bari dan seratus kolom atau dengan nilai dari 1 sampai 30. Berguna untuk input mencari bilangan pembangkit bilangan acak yang dijadikan sebagai pembangkit bilangan soal. Adapun nilai tersebut adalah [1, 2, 3, 4, . . . , 30].

3.4.2 Pembangkit Bilangan Acak Dengan Algoritma Sieve Of Eratoshenes

Pembangkit bilangan acak merupakan bilangan yang hanya akan habis dibagi oleh 1 dan bilangan pembentuk itu sendiri. Adapun mencari bilangan pembangkit bilangan acak dengan algoritma *sieve of eratosthenes* :

$$\Theta(n(\log n)) \log \log n) \text{ bit operasi) } \dots\dots\dots(1)$$

Agar lebih mudah penulis mengurutkan tahapan pencarian algoritma tersebut :

1. Tulis daftar bilangan–bilangan yang akan diuji primalitasnya, dari 2 hingga bilangan terbesar yang ditentukan. Sebut saja daftar nilai ini sebagai daftar A.
 2. Tandai angka dua dari daftar A dan pindahkan ke dalam daftar yang lain untuk menampung bilangan–bilangan pembangkit bilangan acak yang kita cari.
 3. Coret semua angka yang merupakan kelipatan dari 2 dalam daftar A.
 4. Angka yang ditemukan berikutnya dalam daftar A adalah bilangan pembangkit bilangan acak, tandai dan pindahkan ke daftar B.
 5. Coret semua angka yang merupakan kelipatan bilangan pembangkit bilangan acak dalam daftar A tadi, angka yang telah dicoret sebelumnya tidak perlu dicoret lagi.
- Ulangi proses sampai semua nilai selesai.

Berikut perhitungan yang dimaksud yakni dengan algoritma *Sieve Of Eratoshneses*:

1. Langkah pertama, tampilkan angka 1 sampai 30.

Tabel 3.1 Langkah Pertama

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30

2. Langkah kedua, menghapus angka 1 dan semua kelipatan 2.

Tabel 3.2 Langkah Kedua

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30

3. Langkah ketiga, mencari dan menghapus semua kelipatan 3.

Tabel 3.3 Langkah Ketiga

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30

4. Langkah keempat, melakukan proses seperti langkah sebelumnya, kali ini mencari dan menghapus semua kelipatan 5.

Tabel 3.4 Langkah Keempat

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30

5. Langkah kelima dengan mencari dan menghapus semua kelipatan bilangan yang belum pernah dieksekusi sebelumnya, hingga menjadi kelompok bilangan seperti yang ditampilkan pada tabel 3.6 dibawah ini.

Tabel 3.5 Langkah Kelima dan seterusnya

1	2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19	20
21	22	23	24	25	26	27	28	29	30

Hasil Akhir dari proses algoritma *sieve of Eratosthenes* adalah sebagai berikut:

Kelompok pertama: {1, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30}

Kelompok ke-2 : {3, 9, 15, 21, 27}

Kelompok ke-3 : {5, 25}

Kelompok ke-4 : {7}

Kelompok ke-5 : {11}

Kelompok ke-6 : {13}

Kelompok ke-7 : {17}

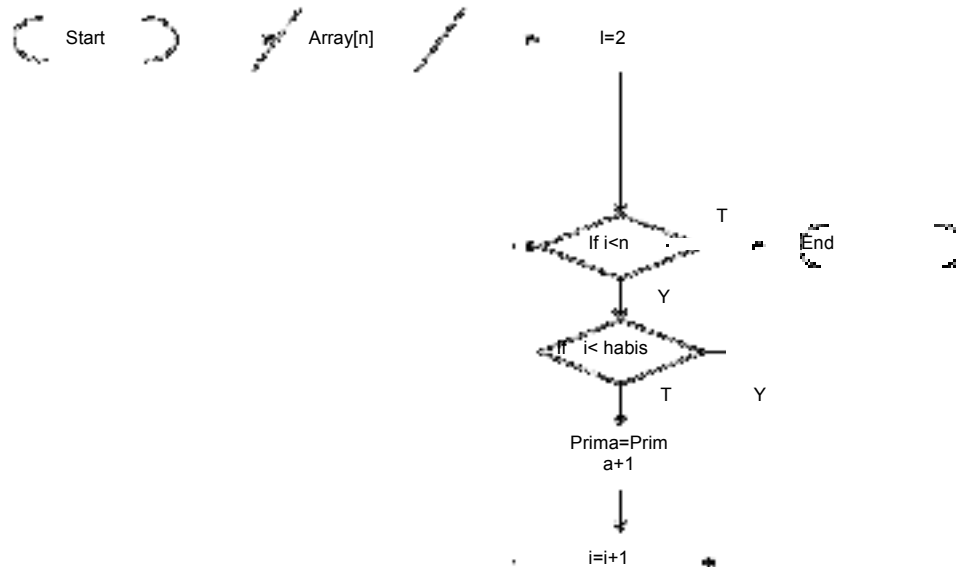
Kelompok ke-8 : {19}

Kelompok ke-9 : {23}

Kelompok ke-10 : {29}

Tabel terakhir langkah pembangkit bilangan acak dapat terlihat ada sepuluh kali seleksi atau perbandingan dengan algoritma *sieve of eratoshenes*.

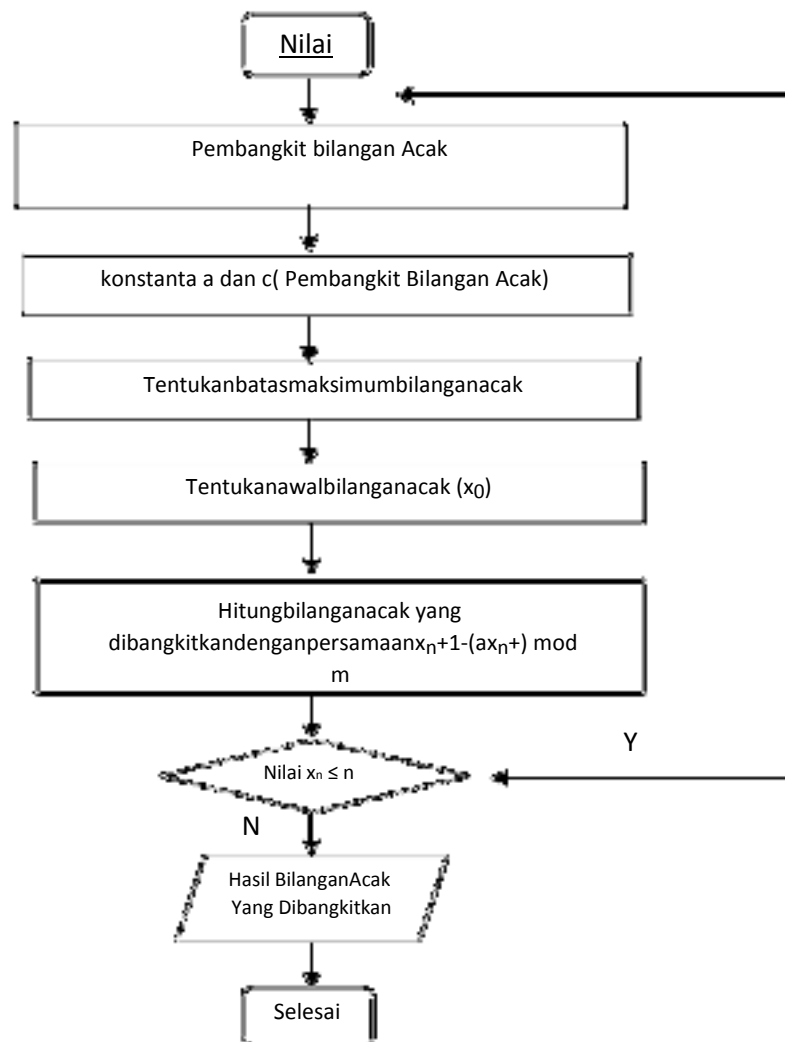
Berikut flowchart algoritma *sieve of eratoshenes*, agar lebih mudah di pahami.



Gambar 3.6 *Flowchart Agloritma Sieve Of Eratoshenes*

3.4.3 Pengacakan Soal – Soal Dengan *Linear Congruent Method*

Berdasarkan hasil analisa yang penulis lakukan terhadap metode *Linear Congruent Method* (LCM) dalam melakukan pengacakan soal – soal, Metode menampilkan soal-soal secara acak atau tidak sama dengan soal yang muncul untuk setiap peserta atau pengguna yang ikut evaluasi atau ujian, hal ini digunakan untuk meminimalisasikan kemungkinan soal yang sama untuk setiap peserta sehingga sama, misalnya ada tiap orang peserta a, b, dan c dan pertanyaan ada 10 soal.



Gambar 3.7 Algoritma *Sieve of eratosthenes* dan *Linear Congruent Method*

Hasil yang bisa di peroleh adalah sebagai berikut:

Nomor Urut	Peserta	Soal
1	A	4,1,5,2,8,6,9,3,10,7.
2	B	1,9,2,6,5,4,7,6,10,3.

3 C 6,4,8,2,5,9,1,7,3,10.

Membangkitkan bilangan acak sebanyak 10 kali dengan ketentuan $a=1$, $c=3$, $m=10$, dan $X_0 = 4$.

$$X_1 = (1(4) + 3) = 7 \text{ mod } 10 = 7.$$

$$X_2 = (1(7) + 3) = 10 \text{ mod } 10 = 0$$

$$X_3 = (1(10) + 3) = 13 \text{ mod } 10 = 3$$

$$X_4 = (1(13) + 3) = 16 \text{ mod } 10 = 6$$

$$X_5 = (1(16) + 3) = 19 \text{ mod } 10 = 9$$

$$X_6 = (1(19) + 3) = 22 \text{ mod } 10 = 2$$

$$X_7 = (1(22) + 3) = 25 \text{ mod } 10 = 5$$

$$X_8 = (1(25) + 3) = 28 \text{ mod } 10 = 8$$

$$X_9 = (1(28) + 3) = 31 \text{ mod } 10 = 1$$

$$X_{10} = (1(31) + 3) = 34 \text{ mod } 10 = 4$$

Bilangan-bilangan acak yang dibangkitkan adalah: 7, 0, 3, 6, 9, 2, 5, 8, 1 dan 4 sehingga perulangan tidak terlihat secara periodik.

3.4.4 Algoritma Pembangkit Bilangan Acak dan Pengacakan

Algoritma adalah langkah-langkah logika penyelesaian masalah yang dijawab secara sistematis dan algoritma merupakan upaya dalam mempermudah proses penulisan *coding* dengan bahasa pemrograman yang telah ditetapkan sebelumnya, maka penulis merancang algoritma aplikasi pembangkit bilangan acak dan pengacakan. Bentuk rancangan algoritma pengacakan dengan *linear congruent method* dalam menampilkan pembangkit dengan algoritma Algoritma *sieve of eratosthenes* untuk pembangkit bilangan acak, agar lebih jelas dapat diperhatikan uraian sebagai berikut :

Pada aplikasi pembangkit bilangan acak menggunakan algoritma sebagai berikut:

Mulai

Input:

a, b, c As Integer

Proses:

For a = 1 To 20

c = 0

For b = 1 To a

If a Mod b = 0 Then c = c + 1

Next

If c = 2 Then

Write(a & " ")

End If

Next

Output :

$N[\text{prima}] = \text{readln}(a \ \& \ " \ ")$

Pada aplikasi menampilkan jawab soal dengan metode *linear congruent method* diimplementasikan pada saat soal-soal yang dipilih berdasarkan level dan kategorinya dilakukan oleh pemain (*user*). Adapun algoritmanya adalah sebagai berikut :

Mulai

Input :

Level, Kategori, IdSoal, Jumlah Benar

Output :

IdSoalAcak, Score, Bonus

Proses :

{Linear Congruent Method}

$a \leftarrow 7, b \leftarrow 5, m \leftarrow 8, X_0 \leftarrow 0$

$X_n \leftarrow ((a(X_{n-1}) + b)) \bmod m$

$\text{IdSoalAcak} \leftarrow X_n$

ElseIf Level = 3 then

$X_n \leftarrow ((a(X_{n-1}) + b)) \bmod m$

$IdSoalAcak \leftarrow X_n$

$Bonus = 0$

If Jumlah Benar ≤ 8 Then

$Score \leftarrow Score + 10$

$Bonus \leftarrow Bonus + 10$

Else If Jumlah Benar ≥ 8 Then

$Score \leftarrow Score + 10$

$Bonus \leftarrow Bonus + 10$

End if

End if

Dengan algoritma diatas menjadi inti bagi penulis dalam merancang aplikasi.

Dengan algoritma tersebut juga gambaran hasil yang ditampilkan.

3.5 Rancangan Database

Kebutuhan tabel data akan simulasi sistem ini meliputi *tableuse* dan *table hasil*. Pada sebuah basis data dan seluruh *record* data, tabel-tabel ditempatkan pada sebuah basis data tersebut yang bernama “db_ivan” sehingga sebelum masuk kehamalan maka akan mengambil data dari tabel *user*.

Tabel 3.6 Tabel Peserta

Nama <i>Fields</i>	Tipe Data	Ukuran
Nisn	Int	3
<i>Username</i>	Varchar	10
<i>Password</i>	Varchar	11

Tabel *user* berguna untuk menyimpan semua data *user* yang menggunakan sistem ini agar *user* yang dapat mengakses data dari *server* serta sebagai pengenalan untuk *user name* (nama pengguna) sehingga *user* yang masuk akan dapat mengirim dan menerima pesan berdasarkan data yang tersimpan dalam *database*

Tabel 3.7 Tabel Soal

Nama <i>Fields</i>	Tipe Data	Ukuran
Id	Int	3
Pertanyaan	Varchar	10
Jawaban	Varchar	10
Kategori	Varchar	10
Jenis	Varchar	20

Tabel soal berguna untuk menyimpan semua data-data yang sedang melakukan ujian berdasarkan soal. Mulai dari data pengirim dan penerima pesan, data pesan diterima dan dikirim dan waktu pengiriman pada tabel hanya dikhususkan untuk penerapan algoritma (cara kerja sistem) yang diteliti atau dipersiapkan.

Tabel 3.8 Tabel Nilai

Nama <i>Fields</i>	Tipe Data	Ukuran
Peserta	Int	3
Nilai	Varchar	10

Tabel saring berguna untuk hasil ujian, sedangkan kata yang tidak terdapat pada *records* atau *fields* data tabel saring, maka hasil tidak dapat berjalan.

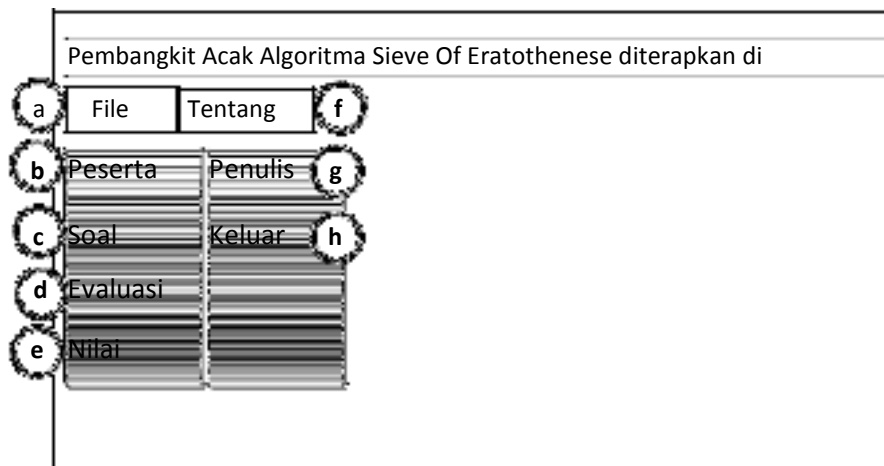
3.6 Rancangan Output

Dalam rancangan output, dibahas mengenai *form-form* yang digunakan dalam pembuatan aplikasi. Dalam rancangan output terdapat 1 *form* utama, yaitu *form* peserta dan *form* soal dan *form* nilai beberapa *form* tambahan yang tidak begitu mempengaruhi jalannya program. Adapun rancangan *form* tersebut diatas adalah :

1. *Form* Utama

Form ini adalah jendela utama dari program aplikasi. *Form* ini berisi semua menu yang tersedia, sekaligus berfungsi untuk melakukan pemindaian komputer untuk mencari gambar yang sama. Ketika program dijalankan maka *form* inilah yang akan pertama kali muncul.

Berikut ini adalah rancangan output *Form* Utama yang di *design* sesederhana mungkin dengan menggunakan *Visual Basic 2008*.



Gambar 3.8 *Form* Utama

Keterangan :

- a) Menu *File*, adalah menu utama untuk proses Penginputan peserta dan soal.
- b) Sub menu peserta untuk melihat data peserta atau melihat *form* peserta.
- c) sub menu soal untuk melihat data soal atau *form* soal.
- d) sub menu evaluasi memulai evaluasi atau membuka *form* evaluasi.
- e) sub menu nilai untuk melihat nilai atau *form* nilai masing-masing peserta.
- f) Menu *Tentang* adalah berisi infor tentang aplikasi.
- g) Sub Menu *Penulis* adalah melihat sekilas tentang penulis program.
- h) Menu *Keluar* adalah untuk keluar dari aplikasi.

2. *Form* Peserta

Form ini adalah memproses, melihat peserta dan sebagai user masuk pada program dari program aplikasi. *Form* ini berisi semua data peserta tersedia, sekaligus untuk simpant, *edit*, hapus.

Berikut ini adalah rancangan output *Form Peserta* yang di *design* sederhana mungkin dengan menggunakan *Visual Basic 2008*.

Gambar 3.9 *Form Peserta*

Keterangan:

- Text Box* Nisn sebagai pengimputan nisn pada masing-masing peserta.
- Text Box* Password berguna pengimputan password peserta menggunakan aplikasi.
- Text Box* Nama berguna pengimputan nama peserta secara lengkap
- Text Box* No.Induk berguna pengimputan No.Induk peserta.
- Combo Box* Jenis Kelamin berisikan Jenis kelamin peserta.

- f) *Text Box* Alamat untuk mengisikan alamat peserta secara lengkap
- g) *List view* sebagai melihat dan meng-edit data dengan memilih data yang ada.
- h) *Button* Baru berguna untuk menginput data baru.
- i) *Button* Simpan berguna untuk menyimpan data yang telah di input.
- j) *Button* Edit berguna untuk mengedit data.
- k) *Button* Hapus berguna untuk menghapus data.
- l) *Button* Keluar berguna untuk keluar dari aplikasi yang berjalan.

3. *Form* Soal

Form ini digunakan untuk melihat dan menyimpan serta mengedit dan hapus soal.

No	Kategori	Soal	A.	B.	C.	D.	E.	Jawaban

Gambar 3.10 *Form* Soal

Keterangan :

- a) *Combo Box* Kategori untuk memasukan kategori soal yang ingin di simpan.

- b) *Text Box* Soal untuk memasukan soal yang ingin di simpan.
- c) *Text Box* A untuk jawaban ganda A dari soal.
- d) *Text Box* B untuk jawaban ganda B dari soal.
- e) *Text Box* C untuk jawaban ganda C darisoal.
- f) *Text Box* D untuk jawaban ganda D dari soal.
- g) *Text Box* E untuk jawaban ganda E dari soal.
- h) *Combo Box* Jawaban untuk memasukan jawaban yang telah di input yang benar.
- i) *List view* sebagai melihat dan mengedit data dengan memilih data yang ada.
- j) *Button* Baru berguna untuk menginput data baru.
- k) *Button* Simpan berguna untuk menyimpan data yang telah di input.
- l) *Button* Edit berguna untuk mengedit data.

4. *Form Nilai*

Form nilai berisikan nilai peserta berdasarkan Nisn peserta jadi setiap peserta hanya memiliki satu penilain dari evaluasi, *form* ini tidak bisa di ubah dan merupakan hal yang benar yang, Nisn peserta diambil ketika saat pertama login.



Nisn	Nama	Kelamin	Benar	Salah	Nilai

Gambar 3.11 *Form Nilai*

Keterangan :

a) *List view* untuk melihat nilai peserta

5. *Form Evaluasi*

Form Evaluasi merupakan tujuan dan hasil dari pembahasan serta penelitian, untuk mengaplikasikanya peserta cukup memberi jawaban ganda pada *RadioButton* kemudian melanjutkan jawaban berikutnya. Kemudian di ahir akan di simpan nilai evaluasi sesuai dengan kode unik user. Tidak ada batas waktu untuk menjawab pertanyaan ini, mungkin cukup mencoba dan belajar sehingga peserta dapat memikirkanya sampai baik.

The image shows a screenshot of a software form titled "Form Evaluasi". The form has a light green background and a yellow footer bar. At the top, there is a label "Form Evaluasi". Below it, the text "Kategori : ----" is followed by a small circle labeled 'a'. To the left, the text "Soal No.:" is followed by a small circle labeled 'b'. A large white rectangular area, representing a text box, is bounded by a line that ends in a small circle labeled 'c' on the right. Below this text box, there are five sets of radio buttons labeled A, B, C, D, and E. Each label is followed by a dashed line and a radio button. The radio buttons are labeled with small circles: 'd' for A, 'e' for B, 'f' for C, 'g' for D, and 'h' for E. At the bottom center, there is a button labeled "Mulai" with a small circle labeled 'i' next to it.

Gambar 3.12 Form Evaluasi

Keterangan :

- a) *Label* Kategori merupakan melihat kategori soal.
- b) *Label* Nomor merupakan nomor soal yang akan di jawab.
- c) *Text Box* Soal merupakan pertanyaan dari soal di tampilkan pada *textbox*.
- d) *Radio Button* A merupakan Jawaban pilihan ganda A.
- e) *Radio Button* B merupakan jawaban pilihan ganda B.
- f) *Radio Button* C merupakan jawaban pilihan ganda C.
- g) *Radion Button* D merupakan jawaban pilihan ganda D.
- h) *Radion Button* E merupakan jawaban pilhan ganda E.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Kebutuhan Spesifikasi Minimum Hardware dan Software

Perangkat keras (*Hardware*) yang direkomendasikan untuk menjalankan aplikasi pengacak soal CPNS pada penelitian ini memiliki spesifikasi sebagai berikut:

- a. *Prosecor Intel(R) Core(TM)i3-2320M CPU@2.10GHz*
- b. *Memory (RAM) 2GB*
- c. *System type 32-bit Operating System*
- d. *VGA card 1 GB atau lebih tinggi lagi*
- e. *Monitor dengan resolusi 1366 x 768 pixel*
- f. *Keyboard dan Mouse*

Perangkat lunak (*Software*) yang digunakan dalam perancangan aplikasi ini *Microsoft Visual Basic .Net* sebagai *interface* nya dan *MySQL tool* penampung *database*. Sebelum menjalankan aplikasi pengacak soal ini, terlebih dahulu *user* menginputkan namanya melalui *form* yang telah disediakan pada tampilan awal dengan tujuan identitas inilah nantinya yang akan digunakan oleh sistem pada saat *user* menyimpan data skornya.

4.2 Tampilan Hasil Implementasi

Sesuai dengan perancangan *form* sebagai *interface* pengguna dalam mengimplementasikan aplikasi pengacak soal CPNS pada *personal computer* dengan metode *linear congruent method*, maka dihasilkan hasil implementasi sesuai dengan yang dilakukan pada tahap perancangan sebelumnya seperti pada gambar-gambar di bawah ini.

a) Tampilan *Form Main Menu*

Form ini berfungsi sebagai *interface* awal untuk menginputkan user atau admin.

Tampilan *form login* seperti berikut :



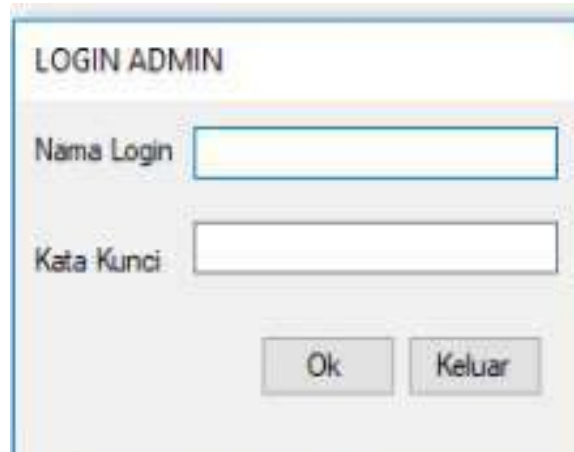
Gambar 4.1 Tampilan *Form Main Menu*

Keterangan :

Pada *form nama* ini terdapat *textbox*, yang harus di isi, seperti, nama login yang merupakan nama dari *user* yang sudah tersimpan, kata kunci yang merupakan password dari akun yang dimiliki oleh *user* dan NIK sebagai pelengkap validasi akun yang telah terdaftar.

b) Tampilan *Form Admin*

Tampilan *form admin* adalah keadaan dimana tombol admin di klik pada main menu sehingga tampilah form dibawah ini:

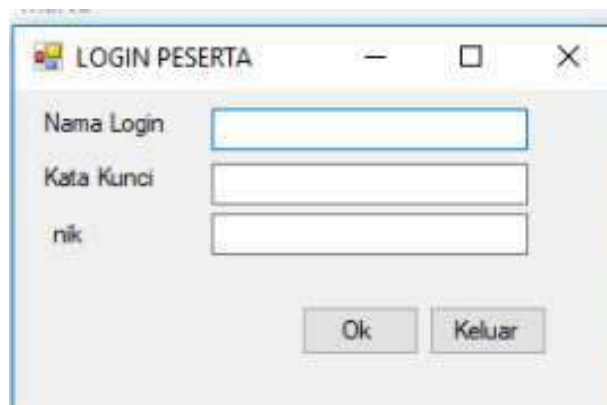


The image shows a dialog box titled "LOGIN ADMIN". It contains two text input fields. The first is labeled "Nama Login" and the second is labeled "Kata Kunci". Below these fields are two buttons: "Ok" and "Keluar".

Gambar 4.2 Tampilan *Form Login Admin*

c) Tampilan *Form Login User*

Tampilan dari form login user di tampilkan pada gambar 4.3 dibawah ini



The image shows a dialog box titled "LOGIN PESERTA". It contains three text input fields. The first is labeled "Nama Login", the second is labeled "Kata Kunci", and the third is labeled "nik". Below these fields are two buttons: "Ok" and "Keluar".

Gambar 4.3 Tampilan *Form Login User*

d) **Tampilan Halaman Depan *Admin***

Tampilan halaman depan admin di tampilkan pada gambar 4.4 dibawah ini



Gambar 4.4 Tampilan Halaman Awal *Admin*

e) **Tampilan Halaman Depan *User***

Tampilan halaman depan user di tampilkan pada gambar 4.5 dibawah ini



Gambar 4.5 Tampilan Halaman Awal *User*

f) Tampilan *Form* Input Soal - Soal

Tampilan *form* soal – soal aplikasi pengacakan jawab soal ini dapat dilihat pada gambar di bawah ini :

NO	L	Kategori	Pertanyaan	A	B	C	E
1	1	Dasar	Ranlauraport merupakan ibukota dari Labuhan Batu. Pada zaman kesultanan, apakah nama kesultanan di R	Sri Kandi	Sri Lela	Sri Batah	S
2	1	Lanjutan	Sistem perekonomian Indonesia mengikuti keseluruhan sistem. Mulai dari sistem kapitalis hingga ke si	Sistem	Indone	Sistem	It
3	2	Dasar	Sumatera Utara, selain terkenal dengan hasil pertaniannya, juga terkenal dengan hasil lautnya. Tanju	Kepah	Kerang	Komudi	U
4	3	Dasar	Medan adalah ibukota dari provinsi Sumatera Utara. Siapakah nama wakil gubernur Sumatera Utara?	Edy Ra	Djarot	Musa R	S
5	4	Lanjutan	17 Agustus 1945 merupakan hari kemerdekaan Republik Indonesia. 17 April 2019 merupakan hari ?	Komord	Kobeba	Komord	T
6	5	Lanjutan	Ibu Kota Negara Malaysia adalah Kuala Lumpur. Walau di daerah sana tidak ada lumpur, namun sebutan i	Upin da	lis dahl	Sib Nur	F
7	6	Lanjutan	Benua Australia terletak pada geografis sub tropis. Kangguru merupakan binatang khas disana. Apakah	Hewan	Bintang	Nama D	T
8	7	Lanjutan	Jika Ani merupakan adik dari Ana. Ani berusia 3x lebih muda dibandingkan dengan Ana. Ana adalah suam	A1 + AA	AA = A	AAA +	A
9	8	Lanjutan	Bali merupakan pulau dewata yang menjadi spot favorit wisatawan domestik dan internasional. Perlamua	The Ga	Scobe	The Ava	T
10	9	Lanjutan	RNI 46 ditelakan pada tahun 46 oleh kakak dari seorang calon presiden 2019. Siapakah nama kakak itu	Selican	Radon	Sesetr	S

Gambar 4.6 Tampilan *Form* Soal

Pada tampilan input soal - soal di atas terdapat *combobox* untuk memilih kategori soal, kemudian *combobox* untuk menampilkan kategori yang disediakan dalam kategori yang dipilih. Komponen selanjutnya adalah *list* untuk menampilkan jawaban soal yang benar kemudian *list* untuk jawaban soal yang telah dijawab (jawaban soal yang dijawab). Semua data harus diisi lengkap karena akan ditampilkan pada pengacakan soal – soal pada *form* proses atau proses selanjutnya. Semakin banyak soal – soal yang dilakukan semakin baik karena akan semakin banyak pengacakan layaknya sebuah kartu semakin banyak kartu yang diacak maka

semakin banyak kemungkinan dan susah untuk ditebak pada proses – proses selanjutnya.

g) Tampilan *Form* Pengacakan Soal

Tampilan *form* pengacakan soal merupakan implementasi dari algoritma *seive of erathenes* untuk pembangkit bilangan acak dan metode *linear congruent* dalam pengacakan soal dengan mengacak soal yang telah di input pada soal – soal sehingga semua soal pasti akan tampil.

The screenshot shows a software window titled "hml.kelihan". At the top right, it says "Kategori: Dasar". On the left, it says "Soal No. 7". The main area contains a text box with the question: "Seluruh tahap pertumbuhan yang dialami makhluk hidup selama hidupnya disebut . . .". Below the text box are five radio button options: A. Metamorfosis, B. Pengembangbiakan, C. Pertumbuhan, D. Daur hidup, and E. metamorfosis serang. At the bottom center, there is a yellow button labeled "Lanjut".

Gambar 4.7 Tampilan *Form* Jawab Soal Dengan Soal Yang Telah Di Acak

h) Tampilan *Form* Menu

Tampilan *form* menu admin ini hanya digunakan oleh admin yang bertugas untuk menambahkan kamus soal-soal yang diacak. Soal-soal yang telah diinputkan inilah nantinya yang akan digunakan pada setiap kategori soal - soal yang akan dilakukan pengacakan, pada dasarnya *form* menu ini merupakan administrator yang

mengembangkan bahasa pemrograman di dalam pembangunan aplikasi dalam kasus ini.



Gambar 4.8 Tampilan *Form* Menu Admin

Pada *form* ini dilengkapi dengan menu-menu pilihan yang dapat digunakan oleh admin. Menu-menu tersebut adalah menu *input* soal yang berfungsi untuk menambahkan perbendaharaan soal yang diacak setiap level, menu file (pengguna atau peserta, soal, kategori), proses pengacakan, pengguna dan bantuan cara memainkan soal dan menu untuk keluar dari aplikasi.

i) Tampilan *Form* Skor

Form *score* berfungsi untuk melihat berapa nilai yang didapatkan oleh setiap pemain pada saat memainkan jawab soal ini. Agar data permainan dapat ditampilkan pada daftar skor, maka terlebih dahulu setiap *user* harus menyimpan data permainannya melalui tombol simpan skor yang telah disediakan pada *form game* jawab soal. Informasi yang tersimpan adalah nama pemain, level yang dimainkan,

jumlah soal yang dijawab secara benar dan skor yang didapatkan oleh pemain.

Tampilannya dapat ditunjukkan pada gambar 4.6 di bawah ini :



Gambar 4.9 Tampilan *Form Score*

4.3 Pengujian Sistem

Pengujian sistem dilakukan dengan melakukan uji coba terhadap aplikasi pengacak soal CPNS yang telah dibuat. Pada penelitian ini, diberikan 10 buah sampel soal yang akan di acak sesuai dengan metode algoritma yang telah dibuat.

Untuk daftar soal yang telah dibuat secara berurutan ditampilkan pada gambar 4.6 dibawah ini.

NO	id_soal	Kategori	Pertanyaan	A	B	C^
3	10	Lanjutan	Sistem perekonomian Indonesia mengikuti keseluruhan sistem. Mulai dari sistem kapitalis hingga ke si	Sistem E...	Indonesi...	S
4	2	Dasar	Sumatera Utara ibu kota dari.	Medan	Tembung	M
5	3	Dasar	Medan adalah ibukota dari provinsi Sumatera Utara. Siapakah nama wakil gubernur Sumatera Utara?	Edy Rah...	Djarot	M
6	4	Lanjutan	17 April 2019 merupakan hari ?	Libur	Raya	N
7	5	Lanjutan	Buah Naga Berasal dari?	Induk Na...	Berastagi	K
8	6	Lanjutan	Tazmania adalah	Hewan A...	Bintang...	N
9	7	Lanjutan	Jika Ani merupakan adik dari Ana. Siapakah Anu	Tetangga	Teman	P
10	8	Lanjutan	Bali merupakan pulau dewata yang menjadi spot favorit wisatawan domestik dan internasional. Pertemuan	The Gam...	Scoobee...	T
11	9	Lanjutan	BNI 46 didirikan pada tahun 46 oleh kakek dari seorang calon presiden 2019. Siapakah nama kakek itu	Subianto	Djodjoha...	S

Gambar 4.10 Daftar Soal CPNS

Percobaan pertama:

Login dengan Akun: Selly

Soal No.1

frmLatihan

Kategori : Dasar

Soal No. 1 Medan adalah ibukota dari provinsi Sumatera Utara. Siapakah nama wakil gubernur Sumatera Utara?

A Edy Rahmayadi C M. Rajeckshah

B Djarot D Syamsul E T. Ery

Lanjut

Gambar 4.11 Percobaan 1 Soal 1

Soal No.2



frmLatihan

Kategori : **Lanjutan**

Soal No. **2** Tazmania adalah

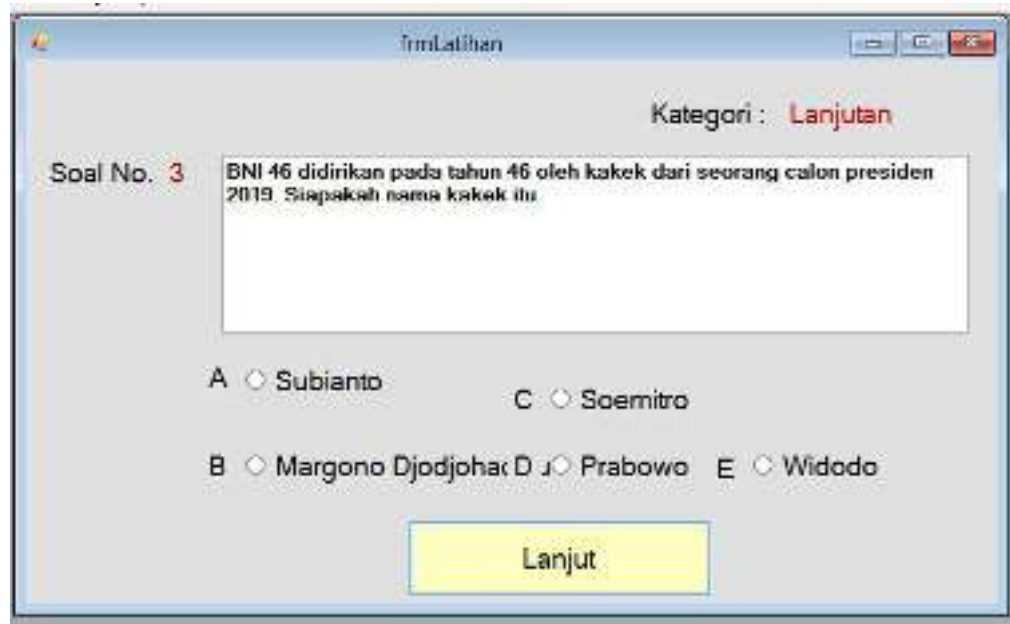
A Hewan Autralia C Nama Daerah

B Bintang Kartun D Tumbuhar E u. Suku Australia

Lanjut

Gambar 4.12 Percobaan 1 Soal 2

Soal no.3



frmLatihan

Kategori : **Lanjutan**

Soal No. **3** BNI 46 didirikan pada tahun 46 oleh kakek dari seorang calon presiden 2019. Siapakah nama kakek itu.

A Subianto C Soemitro

B Margono Djodjohar D Prabowo E Widodo

Lanjut

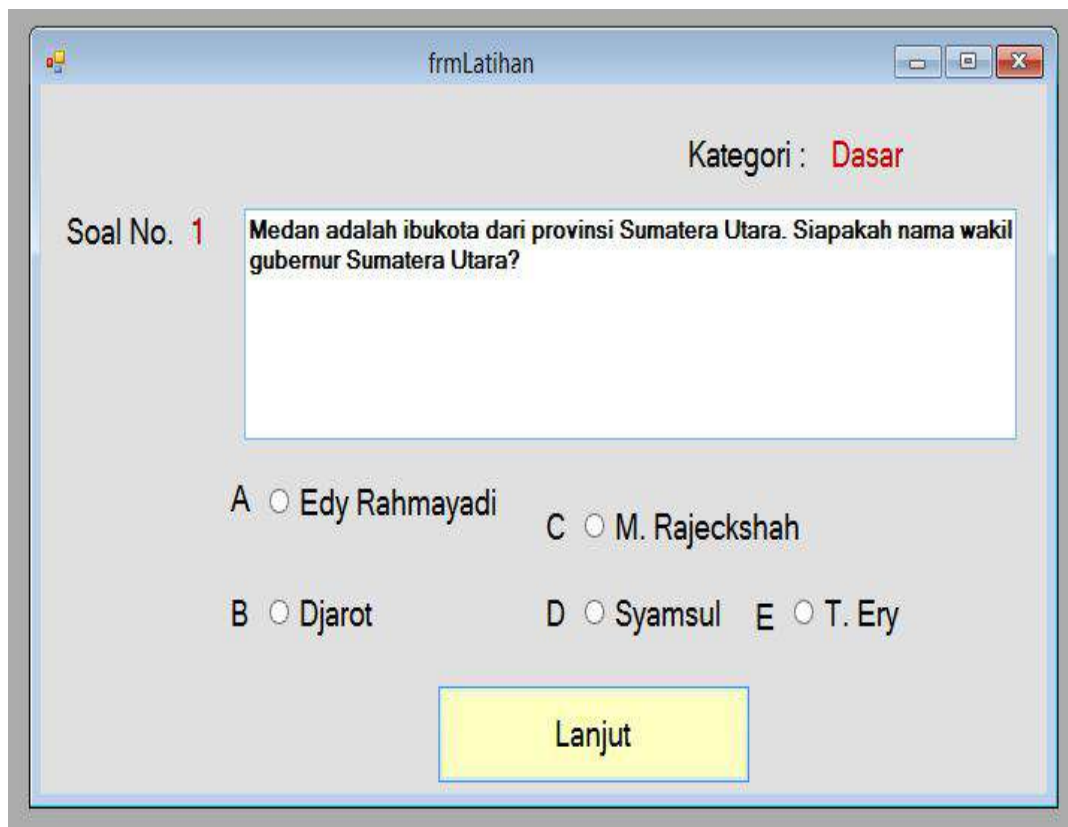
Gambar 4.13 Percobaan 1 Soal 3

Pada percobaan ini data yang di panggil adalah soal dengan nomor kode: {5, 8, 11}, dimana soal dengan kode nomor 5 merupakan pengelompokkan dari data ke-3, soal dengan kode nomor 8 merupakan pengelompokkan dari data pertama dan 11 merupakan kode soal dari pengelompokkan data ke-5.

Percobaan kedua

Login Dengan Akun: Ivan

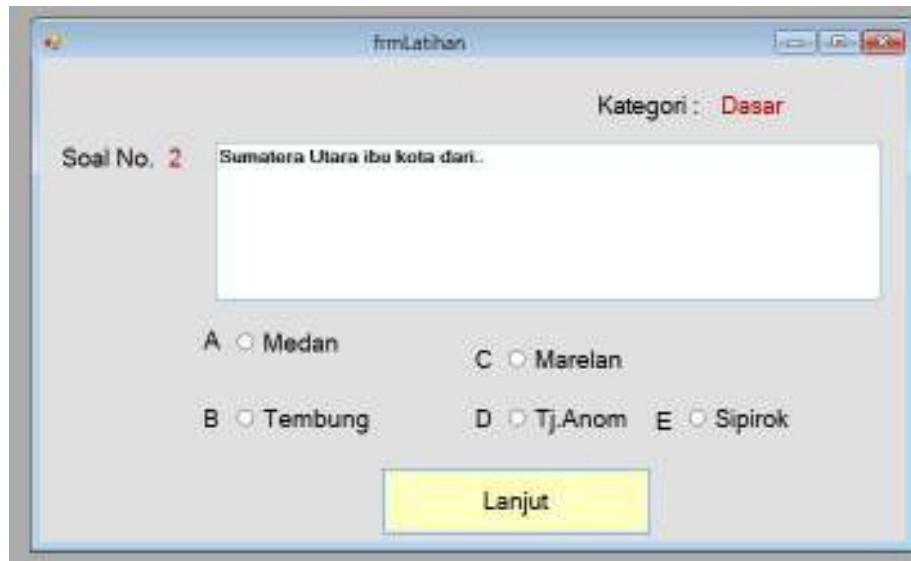
Soal 1



The screenshot shows a window titled "frmLatihan" with a standard Windows-style title bar. The main content area has a light gray background. At the top right, it says "Kategori : Dasar" in red text. On the left, "Soal No. 1" is displayed. To the right of this is a text box containing the question: "Medan adalah ibukota dari provinsi Sumatera Utara. Siapakah nama wakil gubernur Sumatera Utara?". Below the text box are five radio button options: A Edy Rahmayadi, B Djarot, C M. Rajeckshah, D Syamsul, and E T. Ery. At the bottom center, there is a yellow button labeled "Lanjut".

Gambar 4.14 Percobaan 2 Soal 1

Soal 2



fmiLatihan

Kategori : Dasar

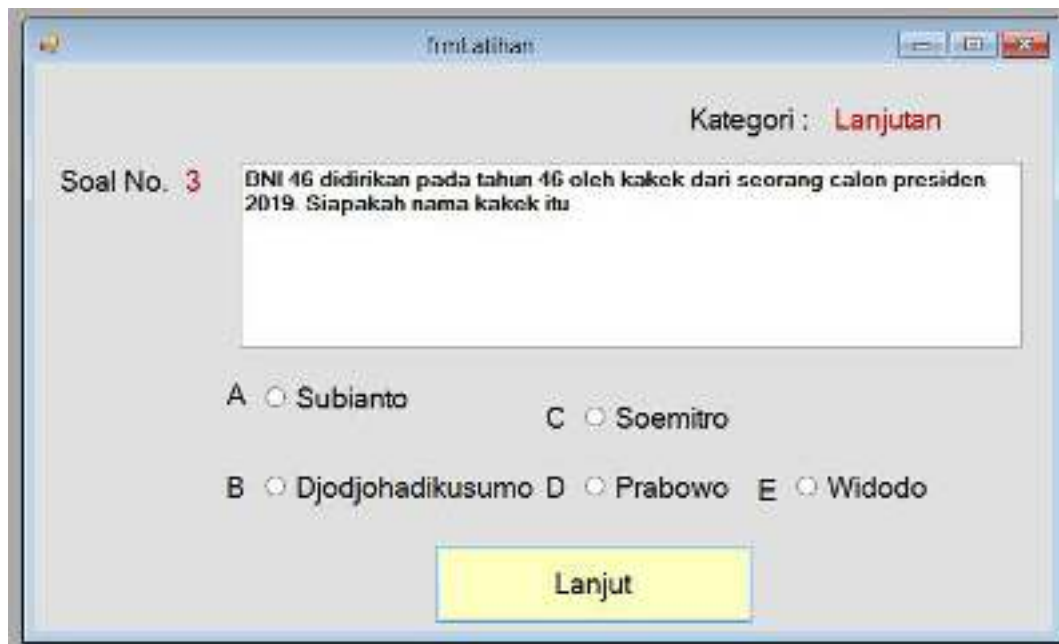
Soal No. 2 Sumatera Utara ibu kota dari..

A Medan C Marelan
B Tembung D Tj.Anom E Sipirok

Lanjut

Gambar 4.15 Percobaan 2 Soal 2

Soal 3



fmiLatihan

Kategori : Lanjutan

Soal No. 3 BNI 46 didirikan pada tahun 46 oleh kakek dari seorang calon presiden 2019. Siapakah nama kakek itu

A Subianto C Soemitro
B Djodjohadikusumo D Prabowo E Widodo

Lanjut

Gambar 4.16 Percobaan 2 Soal 3

Pada percobaan ini data yang di panggil adalah soal dengan nomor kode: {5, 4, 11}, dimana soal dengan kode nomor 5 merupakan pengelompokkan dari data ke-3, soal dengan kode nomor 4 merupakan pengelompokkan dari data pertama dan 11 merupakan kode soal dari pengelompokkan data ke-5.

4.3.1 Evaluasi Sistem

Setelah dilakukan uji coba penulis dapat mengetahui, pada saat input algoritma perhitungan metode *Linear Congruent method* (LCM) harus sesuai dengan persyaratan, jika salah maka angka akan berulang dan itu tidak sesuai dengan soal pilihan ganda yang akan di tampilkan. Hakikatnya sebuah soal tidak boleh ada yang sama dalam 1 paket soal.

Dari hasil pembahasan skripsi ini maka dapat di sajikan beberapa kelebihan dan kekurangan dari sistem yang dibuat diantaranya adalah sebagai berikut:

- a. Kelebihan:
 - 1) Metode LCM dapat digunakan untuk mempermudah melakukan pengacakan soal secara otomatis.
 - 2) *Interface* yang mudah dipahami oleh seorang *user* dalam menginput Bank soal yang akan diberikan.
 - 3) Tingkat keamanan yang tinggi karena pengacakan soal tidak dapat di definisikan secara manual.
- b. Kekurangan:

- 1) Metode LCM hanya dapat bekerja dengan baik bila jumlah minimal soal yang tersedia sebanyak 30 soal, dan maksimal sebanyak 151 soal. Jika soal yang tersedia pada Bank Soal kurang dari 30 soal atau lebih dari 151 soal, maka akan terdapat pengulangan soal pada satu akun peserta saat menjawab soal.
- 2) Pada proses perhitungan metode LCM, langkah pertama adalah dengan menghapus semua angka satu dan seluruh bilangan yang merupakan kelipatan dua. Dalam hal ini, dapat dipastikan kelompok data pada bilangan kelipatan dua merupakan kelompok soal yang memiliki jumlah data pengelompokkan yang lebih banyak diantara kelompok data pada kelipatan bilangan lain. Hal ini menyebabkan, setiap *user* akan lebih banyak menemukan soal yang terdapat pada kelompok data lain secara seragam selain dari kelompok data kelipatan bilangan dua. Sebagai contoh, pada percobaan pertama dan kedua, soal nomor 1 dan nomor 3 adalah soal yang sama dengan urutan yang sama pada *user* yang berbeda.

BAB V

PENUTUP

5.1 Kesimpulan

Adapun yang menjadi kesimpulan dalam penulisan skripsi ini adalah sebagai berikut :

- a. Algoritma *sieve of eratoshenes* untuk pembangkit bilangan acak dengan yang membantu metode *linier congruent method* diterapkan untuk membangkitkan bilangan-bilangan acak.
- b. Implementasi Algoritma *sieve of eratoshenes* untuk pembangkit bilangan acak dengan bilangan prima *linier congruent method* pada pengacakan soal memanfaatkan index dari setiap soal-soal yang akan diacak yaitu banyaknya jumlah soal, kemudian *index* tersebut diacak dengan menggunakan metode sehingga susunan soal-soal yang ditampilkan akan mengikuti susunan *index* kata hasil pengacakan.

5.2 Saran

Adapun yang menjadi saran dalam penulisan skripsi ini adalah sebagai berikut :

- a. Aplikasi ini dapat dikembangkan agar dapat dimainkan secara multi *user*, sehingga menambah ketertarikan pada aplikasi ini.
- b. Aplikasi pengacakan soal dapat dilakukan tidak hanya dengan metode lcm, melainkan dapat mengimplementasikan metode-metode lain yang relevan dengan kasus permainan yang akan dirancang namun untuk pembangkit

bilangan acak penulis menyarankan algoritma *sieve of eratothenes* merupakan algoritma yang terbaik.

- c. Bagi para pembaca boleh sekiranya memberikan masukan dan saran kepada penulis dikemudian hari sehingga penulis merasa bersukur karya ilmiah ini benar – benar dapat membangun motivasi kepada penulis berkelanjutan.

DAFTAR PUSTAKA

- Apdilah, D., & Swanda, H. (2018). *Penerapan Kriptografi RSA Dalam Mengamankan File Teks Berbasis PHP*. JURNAL TEKNOLOGI INFORMASI, 2(1), 45-52.
- Aryza, S., Irwanto, M., Lubis, Z., Siahaan, A. P. U., Rahim, R., & Furqan, M. (2018). A Novelty Design Of Minimization Of Electrical Losses In A Vector Controlled Induction Machine Drive. In IOP Conference Series: Materials Science and Engineering (Vol. 300, No. 1, p. 012067). IOP Publishing.
- Aziz, T. (2018). *Perbandingan Algoritma Fermat, Lehman, the Sieve of Eratosthenes dan the Sieve of Atkins dalam Pembangkitan Bilangan Prima pada RSA*.
- Badawi, A. (2018). Evaluasi Pengaruh Modifikasi Three Pass Protocol Terhadap Transmisi Kunci Enkripsi.
- Bahri, S. (2018). Metodologi Penelitian Bisnis Lengkap Dengan Teknik Pengolahan Data SPSS. Penerbit Andi (Anggota Ikapi). Percetakan Andi Offset. Yogyakarta.
- Fitriani, W., Rahim, R., Oktaviana, B., & Siahaan, A. P. U. (2017). Vernam Encrypted Text in End of File Hiding Steganography Technique. Int. J. Recent Trends Eng. Res, 3(7), 214-219.
- Fuad, R. N., & Winata, H. N. (2017). APLIKASI KEAMANAN FILE AUDIO WAV (WAVEFORM) DENGAN TERAPAN ALGORITMA RSA. InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan, 1(2), 113-119.
- Ginting, J. V., & Barus, E. S. (2018). *Aplikasi Penentuan Rute Rumah Sakit Terdekat Menggunakan Algoritma Dijkstra*. Jurnal Mantik Penusa, 2(2).
- Hendini, A. (2016). *Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak)*. Jurnal Khatulistiwa Informatika, 4(2).
- Khairul, K., Haryati, S., & Yusman, Y. (2018). Aplikasi Kamus Bahasa Jawa Indonesia dengan Algoritma Raita Berbasis Android. Jurnal Teknologi Informasi dan Pendidikan, 11(1), 1-6.

- Kurniawan, T. A. (2018). *Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik*. Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK), 5(1), 77-86.
- Kusrini, (2013). *Perancangan Sistem informasi dan aplikasinya*. Jakarta: Gava Media.
- Mayasari, Nova. "Comparison of Support Vector Machine and Decision Tree in Predicting On-Time Graduation (Case Study: Universitas Pembangunan Panca Budi)." *Int. J. Recent Trends Eng. Res* 2.12 (2016): 140-151.
- Monier, M., & El-mahdy, M. M. (2015). *Evaluation of automated web testing tools*. International Journal of Computer Applications Technology and Research, 4(5), 405-408.
- Nugroho, B. (2015). *Panduan Membuat Aplikasi Program Toko Berbasis Web Dengan PHP-Mysql dan Dreamweaver*. Yogyakarta: Penerbit Gava Media.
- Rahim, R. (2018, October). A Novelty Once Methode Power System Policies Based On SCS (Solar Cell System). In International Conference of ASEAN Prespective and Policy (ICAP) (Vol. 1, No. 1, pp. 195-198).
- Ramadhan, Z., Zarlis, M., Efendi, S., & Siahaan, A. P. U. (2018). Perbandingan Algoritma Prim dengan Algoritma Floyd-Warshall dalam Menentukan Rute Terpendek (Shortest Path Problem). *JURIKOM (Jurnal Riset Komputer)*, 5(2), 135-139.
- Rinaldi, M., & Sianturi, R. D. (2018). *Perancangan Aplikasi Game Edukasi Reaksi Unsur Kimia Menggunakan Metode Linear Congruent Method (LCM)*. *JURIKOM (Jurnal Riset Komputer)*, 5(3), 240-246.
- Saleh, R., & Imelda, I. (2018). *Kriptografi Email menggunakan Algoritma Rivest Code 6 (Rc6) berbasis Java Pada PT. XYZ. SENSITEK*, 1(1), 688-693.
- Suherman, S., & Khairul, K. (2018). Seleksi Pegawai Kontrak Menjadi Pegawai Tetap Dengan Metode Profile Matching. *IT Journal Research and Development*, 2(2), 68-77.
- Sumartono, I., Siahaan, A. P. U., & Mayasari, N. (2016). An overview of the RC4 algorithm. *IOSR J. Comput. Eng.* 18(6), 67-73.
- Tarigan, A. D., & Pulungan, R. (2018). Pengaruh Pemakaian Beban Tidak Seimbang Terhadap Umur Peralatan Listrik. *RELE (Rekayasa Elektrikal dan Energi): Jurnal Teknik Elektro*, 1(1), 10-15.

Tarigan, A. D. (2018, October). A Novelty Method Subjectif of Electrical Power Cable Retirement Policy. In International Conference of ASEAN Prespective and Policy (ICAP) (Vol. 1, No. 1, pp. 183-186).

Wahyuni, S., Lubis, A., Batubara, S., & Siregar, I. K. (2018, September). IMPLEMENTASI ALGORITMA CRC 32 DALAM MENGIDENTIFIKASI KEASLIAN FILE. In Seminar Nasional Royal (SENAR) (Vol. 1, No. 1, pp. 1-6).

Wibowo, P., Lubis, S. A., & Hamdani, Z. T. (2017). Smart Home Security System Design Sensor Based on Pir and Microcontroller. International Journal of Global Sustainability, 1(1), 67-73.