

Usulan PENELITIAN INTERNAL



Sistem Informasi Penjadwalan Mata Pelajaran Sekolah Menggunakan Algoritma Genetika Studi Kasus Smkn 1 Kota Jambi

Oleh:

Dibiayai oleh:

Dipa Universitas Muhammadiyah Jambi tahun anggaran 2022/2023

**UNIVERSITAS MUHAMMADIYAH JAMBI
2022**

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
DAFTAR ISI	iv
RINGKASAN	v
BAB 1 PENDAHULUAN.....	1
1.1 Belakang Belakang Penelitian.....	1
1.2 Rumusan Masalah	3
1.3 Tujuan dan Manfaat Penelitian	4
BAB 2 TINJAUAN PUSTAKA	5
2.1 Algoritma Genetika	5
2.2 Tinjauan Pustaka.....	11
BAB 3 METODE PENELITIAN	13
3.1 Tahap Penelitian	13
3.2 Sumber Informasi	14
3.3 Lokasi Penelitian.....	14
3.4 Alat dan Bahan Penelitian.....	24
JADWAL KEGIATAN PENELITIAN	25
DAFTAR PUSTAKA	26

RINGKASAN

Sistem penjadwalan mata pelajaran sekolah yang disusun secara konvensional akan dirasa kurang efektif dilakukan ketika terbentur keadaan input data banyak dan parameter yang kompleks. Sebagai contoh, pertimbangan yang dilakukan untuk menyusun jadwal perlu memperhatikan berbagai komponen yaitu, Guru, Siswa, Ruangan dan Mata Pelajaran. Dalam rangka menyusun jadwal yang baik, maka harus dilakukan korelasi antar komponen-komponen tersebut agar tidak terjadi kasus “tabrakan” jadwal. Tidak hanya tabrakan jadwal saja yang menjadi pertimbangan. Dengan banyaknya permasalahan ini, tenaga manusia yang bertugas membuat jadwal tentunya besar kemungkinan akan mendapat kesulitan. Atas dasar kesulitan inilah, peneliti mencoba melakukan pendekatan Algoritma Genetika terhadap sistem penjadwalan mata pelajaran sekolah.

Algoritma Genetika adalah teknik pencarian yang di dalam ilmu komputer untuk menemukan penyelesaian perkiraan untuk optimisasi dan masalah pencarian. Algoritma Genetika adalah kelas khusus dari algoritma evolusioner dengan menggunakan teknik yang terinspirasi oleh biologi evolusioner seperti warisan, mutasi, seleksi alam dan rekombinasi (atau crossover). Metode Heuristik salah satu dari bidang kecerdasan buatan yang digunakan untuk menyelesaikan masalah optimasi. Terdapat beberapa algoritma dari metode heuristik yang sering digunakan dalam permasalahan optimasi, diantaranya adalah algoritma genetika, algoritma pencarian tabu, jaringan saraf tiruan, dan lain-lain

SMKN 1 Kota Jambi sebagai sekolah menengah kejuruan memiliki problem penjadwalan yang klasik. Di antaranya adanya jadwal yang mengikat, jumlah guru terbatas, dan ada parameter-parameter tertentu. Proses pembuatan jadwal masih dilakukan secara manual dan belum memiliki model optimasi dengan aplikasi komputer. Salah satu permasalahan tersebut Bagaimana cara implementasi Algoritma Genetika terhadap Software Sistem Penjadwalan Mata Pelajaran Sekolah di SMKN 1 Kota Jambi dan Bagaimana kelayakan pendekatan Algoritma Genetika terhadap masalah penjadwalan mata pelajaran disekolah SMKN 1 Kota Jambi menurut kaidah correctness, functionally, usability dan portability. Dari permasalahan tersebut salah satu tujuan penelitian ini adalah Mengetahui bagaimana cara implementasi Algoritma Genetika terhadap Software Sistem Penjadwalan Mata Pelajaran Sekolah menggunakan algoritma genetika Algoritma genetika cukup baik untuk digunakan dalam penjadwalan mata pelajaran di sebuah satuan pendidikan. Algoritma genetika merupakan salah satu jalan untuk memecahkan masalah yang cukup besar dengan solusi yang cukup baik meskipun masalah tersebut membutuhkan waktu eksekusi yang lama bila dilakukan secara manual (A. Jain, dkk, 2010).

Kata kunci —Algoritma, Genetika, Mata pelajaran, Penjadwalan, Sistem Informasi

BAB I

PENDAHULUAN

1.1 Latar Belakang Penelitian

Sistem penjadwalan mata pelajaran sekolah yang disusun secara konvensional akan dirasa kurang efektif dilakukan ketika terbentur keadaan input data banyak dan parameter yang kompleks. Selain membutuhkan ketelitian yang sangat tinggi serta estimasi waktu yang relatif tidak sedikit, metode ini juga memungkinkan terjadinya kesalahan. Sebagai contoh, pertimbangan yang dilakukan untuk menyusun jadwal perlu memperhatikan berbagai komponen yaitu, Guru, Siswa, Ruang dan Mata Pelajaran.

Dalam rangka menyusun jadwal yang baik, maka harus dilakukan korelasi antar komponen-komponen tersebut agar tidak terjadi kasus “tabrakan” jadwal. Tidak hanya tabrakan jadwal saja yang menjadi pertimbangan. Namun juga beberapa parameter lain, seperti tidak boleh terjadi pengulangan jadwal yang sama dalam satu hari, jumlah jam mengajar guru yang dibatasi, jumlah jam bagi siswa yang disesuaikan dengan tingkatan kelasnya, dan beberapa pelajaran yang tidak boleh dilakukan setelah jam pagi, misalnya pelajaran praktik Olahraga. Dengan banyaknya permasalahan ini, tenaga manusia yang bertugas membuat jadwal tentunya besar kemungkinan akan mendapat kesulitan. Atas dasar kesulitan inilah, peneliti mencoba melakukan pendekatan Algoritma Genetika terhadap sistem penjadwalan mata pelajaran sekolah.

Algoritma Genetika adalah teknik pencarian yang di dalam ilmu komputer untuk menemukan penyelesaian perkiraan untuk optimisasi dan masalah pencarian. Algoritma Genetika adalah kelas khusus dari algoritma evolusioner dengan menggunakan teknik yang terinspirasi oleh biologi evolusioner seperti warisan, mutasi, seleksi alam dan rekombinasi (atau crossover). Persoalan optimasi (optimization problem) adalah persoalan yang menuntut pencarian solusi

optimum (Marwana, 2012). Persoalan optimasi dibagi menjadi dua macam, yaitu maksimasi (maximization) dan minimasi (minimization). Ada dua metode dalam

penyelesaian masalah optimasi, yaitu (1) Metode Konvensional yang diterapkan dengan menggunakan perhitungan matematika murni atau secara biasa.

Persoalan optimasi (optimization problem) adalah persoalan yang menuntut pencarian solusi optimum (Marwana, 2012). Persoalan optimasi dibagi menjadi dua macam, yaitu maksimasi (maximization) dan minimasi (minimization). Ada dua metode dalam penyelesaian masalah optimasi, yaitu (1) Metode Konvensional yang diterapkan dengan menggunakan perhitungan matematika murni atau secara biasa. Ada beberapa metode konvensional yang sering digunakan untuk menyelesaikan masalah optimasi, diantaranya: algoritma Dijkstra, algoritma Floyd- Warshall, dan algoritma Bellman-Ford. (2) Metode Heuristik salah satu dari bidang kecerdasan buatan yang digunakan untuk menyelesaikan masalah optimasi. Terdapat beberapa algoritma dari metode heuristik yang sering digunakan dalam permasalahan optimasi, diantaranya adalah algoritma genetika, algoritma pencarian tabu, jaringan saraf tiruan, dan lain- lain.

SMKN 1 Kota Jambi sebagai sekolah menengah kejuruan memiliki problem penjadwalan yang klasik. Di antaranya adanya jadwal yang mengikat, jumlah guru terbatas, dan ada parameter-parameter tertentu. Proses pembuatan jadwal masih dilakukan secara manual dan belum memiliki model optimasi dengan aplikasi komputer.

Melalui teknik Algoritma yang mulai dikembangkan sejak tahun 1970- an di New York Amerika Serikat, satu kesatuan jadwal dipahami sebagai satu individu yang memiliki sejumlah gen. Dimana gen-gen tersebut adalah perwujudan dari susunan kode guru yang diatur dengan array tertentu. Pada proses awal, akan dibangkitkan sejumlah individu dengan gen yang random. Setelah itu, akan dibobot kualitas jadwal tersebut sebagai nilai fitness. Nilai fitness adalah angka yang menyatakan kualitas individu. Semakin individu tersebut berkualitas baik, maka nilai fitness pun semakin banyak. Setelah itu, dilakukan proses seleksi menggunakan metode roulette wheel. Seleksi ini dimaksudkan untuk mencari individu dengan probabilitas tinggi, dengan fitness yang tinggi pula. Pada proses

selanjutnya, dilakukan proses crossover (kawin silang) dengan cara memotong individu dan memasangkan dengan individu lain. Tidak hanya itu, tahap selanjutnya adalah melakukan mutasi gen. Sebagai gambaran, mutasi gen dilakukan dengan mengacak nilai random untuk lokasi gen dan mengganti nilai gen di lokasi yang bersangkutan dengan nilai gen lain yang dibangkitkan secara random juga. Dengan selesainya proses-proses di atas, maka didapatkan sejumlah individu baru yang telah mengalami proses seleksi, crossover, dan mutasi. Individu-individu ini akan kembali dibobot nilai fitness dan jika belum memenuhi syarat fitness optimal, maka dilakukan kembali proses sedari awal dengan menggunakan individu-individu baru tersebut. Perulangan akan terus dilakukan hingga mendapatkan individu atau jadwal yang cocok dengan harapan.

Penelitian-penelitian terbaru menyarankan bahwa algoritma genetika merupakan metode yang layak dan efektif dalam mengatasi masalah penjadwalan (Teno dan Palgunadi, 2014). Algoritma genetika merupakan metode heuristik adaptif yang memiliki dasar pemikiran atau gagasan untuk proses seleksi alam dan genetika berdasarkan penelitian Charles Darwin. Algoritma genetika cukup baik untuk digunakan dalam penjadwalan mata pelajaran di sebuah satuan pendidikan. Algoritma genetika merupakan salah satu jalan untuk memecahkan masalah yang cukup besar dengan solusi yang cukup baik meskipun masalah tersebut membutuhkan waktu eksekusi yang lama bila dilakukan secara manual (A. Jain, dkk, 2010).

1.2 Rumusan Masalah

Berdasarkan latar belakang, maka dapat ditarik rumusan masalah sebagai berikut:

1. Bagaimana cara implementasi Algoritma Genetika terhadap Software Sistem Penjadwalan Mata Pelajaran Sekolah?
2. Bagaimana kelayakan pendekatan Algoritma Genetika terhadap masalah penjadwalan mata pelajaran sekolah menurut kaidah correctness, functionally, usability dan portability?

1.3 Tujuan dan Manfaat Penelitian

1.3.1 Tujuan Penelitian

Adapun tujuan penelitian ini adalah sebagai berikut:

1. Mengetahui bagaimana cara implementasi pembuatan Software Sistem Penjadwalan Mata Pelajaran Sekolah Menggunakan Algoritma Genetika.
2. Mengetahui bagaimana kelayakan pendekatan Algoritma Genetika terhadap masalah penjadwalan mata pelajaran sekolah dengan parameter correctness, functionally, usability dan portability.

1.3.2 Manfaat Penelitian

Penelitian tentang penggunaan Algoritma Genetika untuk Sistem Penjadwalan Mata Pelajaran Sekolah diharapkan dapat bermanfaat:

- a. Untuk pengembangan ilmu pengetahuan, terutama bagi dunia Teknik Informatika dan Ilmu Komputer.
- b. Menemukan cara pengimplementasian Algoritma Genetika terhadap Software

Sistem Penjadwalan Mata Pelajaran Sekolah.

- a. Bagi Peneliti
 - 1) Memahami proses kerja Algoritma Genetika dan bentuk implementasinya.
 - 2) Sebagai portopolio untuk peneliti yang berguna untuk masa yang akan datang.
- b. Bagi Sekolah
 - 1) Sebagai bahan referensi untuk penelitian yang akan datang.
 - 2) Sebagai bahan evaluasi bagi universitas dalam mengembangkan keilmuan, dalam hal ini yang berkaitan dengan program berbasis Algoritma Genetika.

BAB II

TINJAUAN PUSTAKA

2.1 Algoritma Genetika

Algoritma Genetika adalah algoritma pencarian yang didasarkan pada mekanisme seleksi alamiah dan genetika alamiah (Suyanto, 2005). Sesuai awal konsepnya yaitu genetika, algoritma ini juga mengambil istilah-istilah yang ada dalam ilmu genetika seperti populasi, individu, mutasi, kawin silang dan generasi. Konsep yang ada dalam kaidah genetika ini diterapkan menjadi sebuah algoritma komputasi untuk menyelesaikan masalah yang membutuhkan constraint tinggi.

Algoritma Genetika sebagai metode optimasi yang powerfull dimungkinkan telah menjadi teknik paling terkenal dalam bidang komputasi evolusioner pada saat ini (Mitsuo Gen, Runwei Cheng, 2000, h. 17).

Secara umum, sebuah Algoritma Genetika memiliki lima komponen dasar, seperti yang dilansir dari Michalewicz (1996) ditulis kembali dalam buku “Genetic Algorithm and Engineering Optimization” oleh Mitsuo Gen dan Runwei Cheng (Mitsuo Gen, Runwei Cheng, 2000, h. 17): (1) Representasi genetika untuk solusi masalah, (2) Metode menciptakan inisiasi penyelesaian dari populasi, (3) Evaluasi nilai fitness berdasarkan kemungkinan solusi, (4) Metode genetika dalam penggantian keturunan dan reproduksi, (5) Hasil akhir yang diharapkan dari pengolahan Algoritma Genetika.

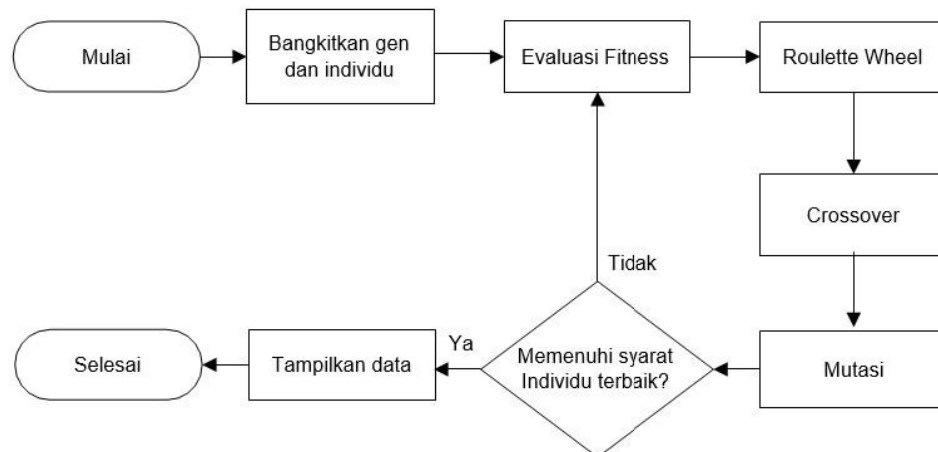
1. Istilah dalam Algoritma Genetika.

Karena mengambil dari kaidah konsep genetika biologi, beberapa istilah dalam Algoritma Genetika juga mengambil konsep yang sama seperti dilansir dari buku berjudul “Algoritma Genetika dalam Matlab” oleh Suyanto (2005), yaitu: 1) Populasi adalah sekelompok individu yang akan dicari penyelesaiannya dalam Algoritma Genetika. 2) Kromosom atau Individu adalah satu permasalahan dan atau penyelesaian yang merupakan komponen iterasi. 3) Gen adalah bagian dari kromosom yang memiliki nilai tertentu dan bertugas menyusun kromosom menjadi individu utuh. 4) Nilai fitness adalah derajat kebaikan suatu individu yang menentukan apakah individu tersebut berkualitas atau tidak. 5) Generasi adalah jumlah tingkatan peranakan sebuah kelompok populasi yang akan

dikembangbiakkan menjadi populasi yang mutan, mengalami crossover dan mutasi beberapa kali sehingga menimbulkan generasi terbaik.

2. Mekanisme Algoritma Genetika

Secara umum, Algoritma Genetika dapat dijabarkan sebagai alur dan bagan pada gambar 1 berikut:



Gambar 1. Bagan Perulangan Algoritma Genetika

Sebelum Algoritma Genetika dijalankan, maka perlu didefinisikan fungsi fitness sebagai masalah yang ingin dioptimalkan. Jika nilai fitness semakin besar, maka sistem yang dihasilkan semakin baik. Fungsi fitness ditentukan dengan metode heuristik. Algoritma Genetika sangat tepat digunakan untuk penyelesaian masalah optimasi yang kompleks dan sukar diselesaikan dengan menggunakan metode konvensional. Sebagaimana halnya proses evolusi di alam, suatu Algoritma Genetika yang sederhana umumnya terdiri dari tiga operasi yaitu: operasi reproduksi, operasi crossover (persilangan), dan operasi mutasi. Struktur umum dari suatu Algoritma Genetika dapat didefinisikan dengan langkah-langkah sebagai berikut:

- Membangkitkan populasi awal secara random.
- Membentuk generasi baru dengan menggunakan tiga operasi di atas (seleksi, crossover, mutasi) secara berulang-ulang sehingga diperoleh kromosom yang cukup untuk membentuk generasi baru sebagai representasi dari solusi baru.

- c. Evolusi solusi yang akan mengevaluasi setiap populasi dengan menghitung nilai fitness setiap kromosom hingga kriteria berhenti terpenuhi.

Bila kriteria berhenti belum terpenuhi maka akan dibentuk lagi generasi baru dengan mengulangi langkah regenerasi. Beberapa kriteria berhenti yang sering digunakan antara lain:

- a. Berhenti pada generasi tertentu.
- b. Berhenti setelah dalam beberapa generasi berturut-turut didapatkan nilai fitness tertinggi/terendah (tergantung persoalan) tidak berubah.
- c. Berhenti bila dalam n generasi berikutnya tidak diperoleh nilai fitness yang lebih tinggi/rendah.

3. Seleksi Roulette

Seleksi Roulette adalah salah satu metode seleksi individu yang tetap melibatkan keanekaragaman populasi. Sesuai dengan namanya, metode ini menirukan permainan Roulette Wheel di mana masing-masing individu menempati potongan lingkarannya pada roda Roulette secara proporsional sesuai dengan nilai fitness (Suyanto, 2005).

4. Crossover

Crossover atau kawin silang adalah prosedur memasangkan dua buah individu untuk kemudian dipisahkan masing-masing gennya dan dipasangkan dengan gen pasangannya. Sebuah individu dapat memperoleh solusi yang bagus jika dilakukan proses memindah-silangkan dua buah individu (Suyanto, 2005).

Crossover menyediakan sebuah metode yang memungkinkan terjadinya eksplorasi bagian baru dalam ranah algoritma solusi (Coley, 2000).

5. Mutasi

Dalam dunia nyata, sebuah mutasi dapat terjadi akibat suatu proses. Begitu pula yang terjadi dalam Algoritma Genetika (Coley, 2000). Secara umum, proses mutasi dilakukan dengan cara membangkitkan sebuah bilangan random yang kurang dari probabilitas mutasi (mutation rate) kemudian gen yang ada diubah

menjadi nilai kebalikannya. Semisal 0 menjadi 1, 1 menjadi 0. Prosedur ini telah disarankan oleh Suyanto dalam bukunya yang berjudul “Algoritma Genetika dalam Matlab” (2005).

B. Hypertext Preprocessor

Menurut (Das & Saikia, 2016) PHP : Hypertext Preprocessor merupakan Bahasa scripting yang sangat diketahui sering dipakai dalam pengembangan web walaupun sebenarnya tidak hanya itu kegunaannya. Berdasarkan w3techs.com PHP merupakan bahasa scripting yang sering dipakai di internet sebesar 82%.

PHP adalah singkatan dari Hypertext Preprocessor, yaitu bahasa pemrograman yang digunakan secara luas untuk penanganan, pembuatan, dan pengembangan sebuah situs web dan biasanya digunakan bersamaan dengan HTML (Oetomo dan Mahargiono, 2020:1).

C. Codeigniter

Menurut (Arrhioui et al., 2017), Codeigniter adalah kerangka kerja pengembangan aplikasi PHP berdasarkan arsitektur yang terstruktur. Codeigniter memiliki tujuan untuk memberikan alat bantu yang dibutuhkan seperti helpers and libraries untuk mengimplementasi tugas yang biasa dilakukan. Dengan demikian, pengembangan proyek menjadi lebih mudah dan cepat. Dan pengembang tidak perlu menulis lagi dari awal.

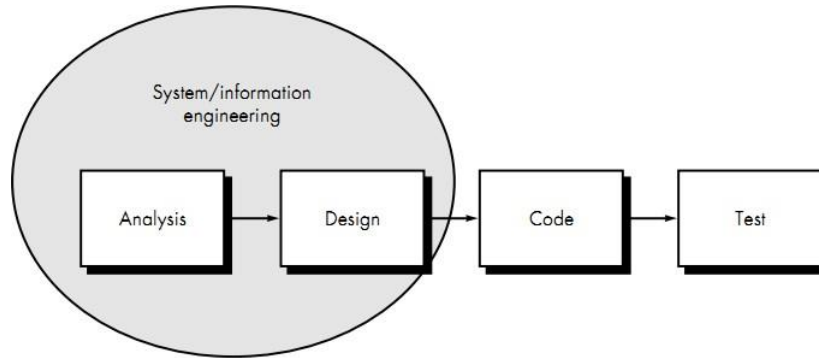
D. MVC (*Model, View, Controller*)

Menurut (Endra & Aprilita, 2018), MVC merupakan suatu metode yang digunakan dalam pengembangan system yang menggunakan prinsip memisahkan setiap kerangka kerjanya, yaitu logic, presentation, dan process atau biasa yang kita tahu model, view, controller.

E. Perancangan Perangkat Lunak

Proses perangkat lunak adalah kerangka kerja yang memuat tugas- tugas yang diperlukan untuk membangun perangkat lunak yang berkualitas baik (Pressman, 2010). Terdapat beberapa teori dalam proses perancangan perangkat lunak, antara lain adalah Waterfall / Linear Sequential Process. Proses Waterfal

merupakan metode software process yang paling klasik, dapat disebut juga dengan model sekuensial linier terdiri dari pendekatan sistematis berurutan untuk pengembangan perangkat lunak yang berisi proses analisis, desain, coding, pengujian, dan support (Pressman, 2010). Gambar 2 berikut menampilkan Linear Sequential Model:

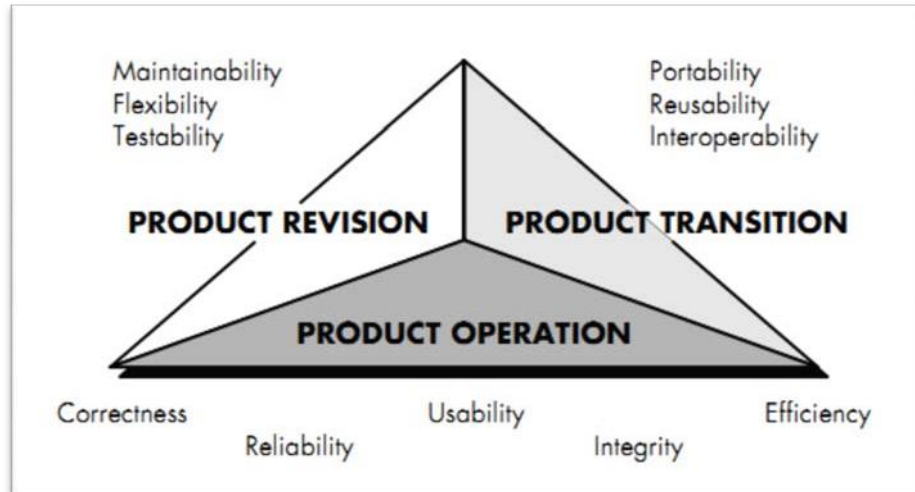


Gambar 2. Linear Sequential Software Model.

F. Kualitas Perangkat Lunak (*Software Quality*)

Kualitas Perangkat Lunak atau Software Quality adalah gabungan beberapa faktor yang mempengaruhi antara aplikasi dengan konsumen yang membutuhkan. Dalam hal ini, kualitas software diidentifikasi sebagai deskripsi dari aktifitas manusia dalam mengapresiasi sebuah software (Pressman, 2010).

Terdapat beberapa teori tentang kualitas perangkat lunak, salah satunya adalah McCall Quality Factor. McCall, Richards, dan Walter merumuskan serangkaian faktor – faktor yang menunjukkan kualitas perangkat lunak. Faktor – faktor kualitas tersebut dikategorikan menjadi tiga aspek penting dari sebuah perangkat lunak yaitu: karakteristik operasional, kemampuan untuk dalam menangani perubahan, dan kemampuan beradaptasi dengan lingkungan baru (Pressman, 2010). Kategori tersebut digambarkan dalam gambar 3 di bawah ini.



Gambar 3. Kualitas Perangkat Lunak menurut McCall

Faktor – faktor kualitas yang menunjukkan kualitas perangkat lunak tersebut antara lain:

- a. *Correctness* (ketepatan): berkaitan dengan bagaimana program mampu memenuhi spesifikasi dan tujuan yang ingin dicapai oleh pengguna.
- b. *Reliability*: berkaitan dengan bagaimana sebuah program mampu beroperasi dalam sebuah kondisi yang menuntut presisi tertentu.
- c. *Usability* (kebergunaan): berkaitan dengan usaha yang diperlukan pengguna untuk mengoperasikan, menyiapkan input, dan menginterpretasikan output dari program.
- d. *Integerity* (integritas): berkaitan dengan tingkat kontrol terhadap program oleh pengguna, baik yang mendapatkan otorisasi atau tidak.
- e. *Efficiency* (efisien): berkaitan dengan jumlah sumber daya komputer yang digunakan serta kode yang diperlukan di dalam program untuk menjalankan setiap fungsinya.
- f. *Maintainability* : berkaitan dengan usaha yang diperlukan untuk menemukan dan mengatasi kesalahan di dalam program.
- g. *Flexibility* (Fleksibel): berkaitan dengan usaha yang diperlukan untuk mengubah program yang beroperasi.

- h. *Testability* : berkaitan dengan usaha yang diperlukan untuk menguji sebuah program untuk memastikan bahwa program tersebut berfungsi sebagaimana mestinya.
- i. *Portability* : berkaitan dengan usaha yang diperlukan untuk dapat mentransfer sebuah program dari sebuah lingkungan perangkat keras atau lunak tertentu ke lingkungan yang lain.
- j. *Reusability* : berkaitan dengan bagaimana sebuah bagian program dapat digunakan kembali di dalam program lain.
- k. *Interoperability* : berkaitan dengan usaha yang diperlukan untuk menghubungkan sebuah sistem dengan sistem yang lain

2.2 Tinjauan Pustaka

Berikut ini adalah ringkasan dari beberapa penelitian sebelumnya yang berkaitan dengan sistem penjadwalan dengan algoritma genetika :

1. Gibbon Mardame Parsaroan Tamba (2004), Sistem Penjadwalan Matakuliah menggunakan Algoritma Genetika (Studi Kasus Fakultas Matematika dan IPA IPB). Dalam penelitian ini, dilakukan metode perbandingan penyusunan jadwal secara manual dengan penyusunan jadwal menggunakan Algoritma Genetika pada jadwal kuliah Fakultas MIPA IPB. Kesimpulan dari penelitian ini adalah metode Algoritma Genetika dinyatakan efektif untuk menyusun jadwal perkuliahan secara optimal.
2. Sam'ani (2012), Rancang Bangun Sistem Penjadwalan Perkuliahan dan Ujian Akhir Semester dengan Pendekatan Algoritma Genetika. Penelitian ini mengambil kasus penjadwalan perkuliahan di STMIK Palangkaraya. Dimana hasil yang didapat adalah penjadwalan dinilai menjadi lebih optimal setelah iterasi yang relatif pendek, yaitu 5-10 generasi.
3. Algoritma Genetika telah digunakan untuk menghasilkan nilai maksimum luas coverage area dan biaya minimum operasional penempatan armada kapal TNI AL di kawasan timur Indonesia. Sistem yang dibuat diuji dengan penempatan armada kapal sebanyak 27 buah untuk ditempatkan pada 28 pangkalan yang ada. Hasil yang didapatkan adalah luas coverage area

1.942.929 mil² dan biaya operasional Rp 2.853.447.000 dengan nilai fitness terbaik 6,6330. Jika dibandingkan dengan data lapangan yang ada yaitu total luas area yang harus diamankan sekitar 1.688.765 mil² dengan budget biaya Rp 5.000.000.000 maka hasil yang didapatkan lebih efektif dan efisien (Sam'ani, 2012).

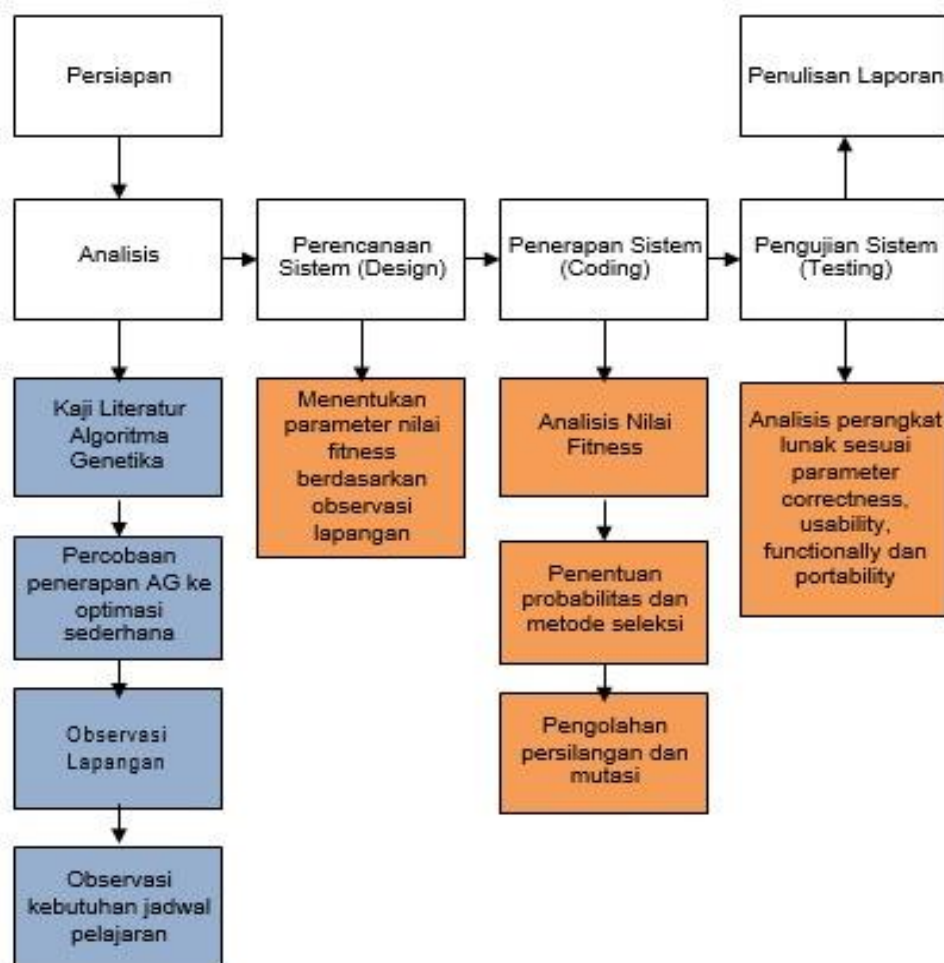
4. Algoritma Genetika juga telah digunakan untuk pengolahan pengenalan citra huruf kapital dengan metode pembangkitan nilai fitness berdasar transformasi digital seperti translasi, dilatasi dan rotasi. Dalam penelitian tersebut dihasilkan pengenalan yang optimal dengan generasi di atas 2000 (Saputro, 2003).
5. Sebagai sistem pendukung keputusan teknik perdagangan dalam masalah TSP (Traveling Salesman Product). Metode ini telah diteliti oleh Franklin Allen dan Risto Karjalainen dalam jurnal yang berjudul "Using Genetic Algorithm to Find Technical Trading" (Allen, 1999).

BAB III

METODE PENELITIAN

3.1 Tahap Penelitian

Mengembangkan dari model tahap pembuatan perangkat lunak seperti pada metode linear, prosedur dan urutan kerja dalam penelitian ini dapat dijabarkan menjadi beberapa sub prosedur pada gambar 4 sebagai berikut :



Gambar 4. Prosedur Kerja dan Urutan Kerja

Tahap pertama adalah persiapan yang meliputi perencanaan awal dalam bentuk rencana penelitian. Rencana ini berisi langkah-langkah penelitian secara

umum. Yang kedua adalah Observasi Lapangan & Kaji Literatur Algoritma Genetika (Analisis) yang dibutuhkan untuk memperoleh gambaran dari objek penelitian yaitu jadwal pelajaran. Untuk keperluan ini, observasi dilakukan di SMKN 1 Kota Jambi. Kemudian kaji literatur dilakukan dengan memperkaya bacaan tentang Algoritma Genetika, baik melalui media internet, jurnal, penelitian kontemporer, maupun buku-buku yang relevan. Kemudian tahap selanjutnya adalah Perancangan Sistem (Design), Penerapan Sistem (Coding), Pengujian Sistem (Testing), dan Penulisan Laporan.

3.2 Sumber Informasi

Sumber informan dalam uji kualitas usability adalah personal pembuat jadwal di sekolah terkait (SMKN 1 Kota Jambi) dan guru TIK yang memahami alur aplikasi. Pemilihan sumber informan ini dititikberatkan pada orang yang memang berkecimpung dalam dunia penjadwalan sekolah (personal kurikulum atau pembuat jadwal) dan guru TIK yang memiliki pemahaman terhadap aplikasi.

3.3 Lokasi Penelitian

Lokasi penelitian untuk uji kemampuan dan pendekatan Algoritma Genetika untuk penjadwalan mata pelajaran sekolah dilakukan di SMKN 1 Kota Jambi, beralamat di Jl. Jend. A. Thalib, Simpang IV Sipin, Kec. Telanaipura, Kota Jambi, Jambi 36124. Uji software jadwal sekolah untuk faktor usability, peneliti mengambil observasi di salah satu sekolah menengah atas yaitu SMKN 1 Kota Jambi.

1. Teknik Pengaplikasian Algoritma Genetika.

Secara singkat, istilah-istilah dan bagian Algoritma Genetika dalam aplikasinya sebagai perangkat lunak penjadwalan dapat dijelaskan pada tabel 4 berikut:

Tabel 1. Istilah dalam Algoritma Genetika

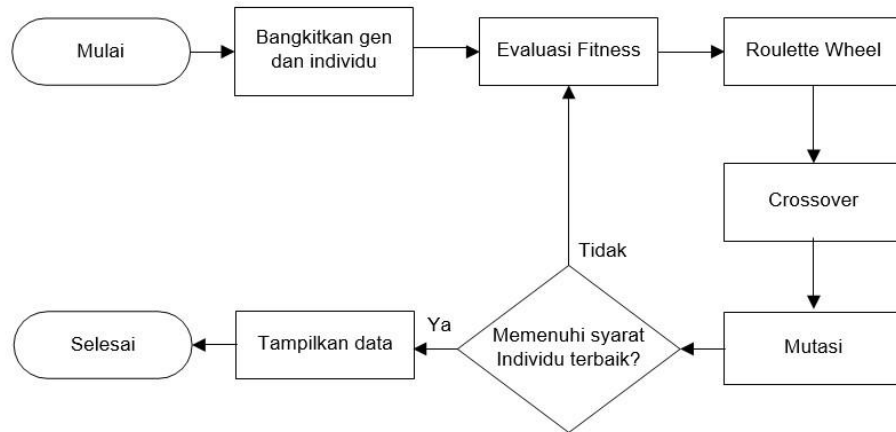
Nama Istilah	Penjelasan
--------------	------------

Populasi	Populasi adalah jumlah jadwal acak yang dibangkitkan dengan jumlah tertentu. Misal, jadwal 1 dan jadwal 2 dikatakan sebagai satu kesatuan populasi.
Kromosom = Jadwal	Satu kesatuan jadwal dinamakan individu yang memiliki identitas kromosom yang berbeda-beda. Panjang kromosom dipengaruhi oleh panjang gen, yaitu banyaknya pertemuan dalam satu jadwal.
Gen = pertemuan	Tiap pertemuan disebut gen untuk kromosom jadwal.
Iterasi	Iterasi adalah bilangan yang menunjukkan tingkat generasi dan perulangan.
<i>Crossover</i>	<i>Crossover</i> adalah proses kawin silang di mana sepasang individu dipotong melalui matriks tertentu untuk dipasangkan dengan individu pasangannya.
Mutasi	Mutasi adalah penggantian satu struktur gen di dalam individu yang dibangkitkan secara random.

a. Gambaran Umum Metode

Metode pada Algoritma Genetika memiliki beberapa tahap dan sekian perulangan. Dalam kasus ini, metode pada pembuatan jadwal kelas memiliki beberapa tahap seperti pada gambar 5 yaitu:

- 1) Pembangkitan individu.
- 2) Evaluasi fungsi objektif dan nilai fitness.
- 3) Penentuan nilai probabilitas.
- 4) Seleksi Roulette Wheel.
- 5) Crossover / Persilangan.
- 6) Mutasi
- 7) Pembangkitan individu baru.



Gambar 5. Flowchart Algoritma Genetika

b. Variabel yang Digunakan dalam Proses Algoritma Genetika

1) Jumlah Individu per Populasi

Jumlah individu per populasi ditentukan sendiri pada awal pemrograman. Semakin banyak jumlah individu per populasi, semakin besar kemungkinan individu yang unggul dan mampu berregenerasi dengan cepat dan efisien.

2) Jumlah Generasi (iterasi)

Jumlah generasi adalah jumlah maksimal iterasi / paket perulangan yang diperbolehkan. Variabel ini menentukan sampai berapa kali populasi awal akan berubah, jadi juga memiliki peran yang tak kalah penting dalam menampilkan jumlah variasi individu, yang akan berpengaruh terhadap hasil Algoritma Genetika. Dalam kasus ini dibatasi maksimal 100 generasi untuk mendukung memori yang terdapat dalam komputer yang digunakan dalam melakukan Algoritma Genetika ini. Jadi hal itu berhubungan erat dengan keterbatasan memori yang terdapat dalam sebuah komputer.

3) Fungsi Objektif dan Nilai Fitness

Fungsi objektif adalah formula yang dibuat untuk mengukur derajat kualitas individu. Formula ini yang akan dibuat untuk menghitung angka nilai fitness. Nilai fitness adalah bilangan yang menunjukkan kualitas individu. Semakin tinggi nilai fitness, semakin tinggi kualitas dan semakin tinggi pula tingkat probabilitas seleksi.

4) Tingkat Persilangan (*cross over*)

Tingkat persilangan adalah peluang untuk terjadi persilangan antara sepasang individu. Dalam kenyataannya persilangan akan selalu terjadi, hanya saja jumlah gen dan gen-gen yang disilangkan akan berbeda-beda. Oleh karena itu, biasanya TP diisi dengan nilai 1 atau 100%, yang berarti akan selalu terjadi persilangan.

5) Persentase Persilangan (*crossover rate*)

PP merupakan angka persentase yang akan mempengaruhi berapa gen yang akan disilangkan. Misalkan dalam satu individu, ada 6 gen dan $PP=0.5$. Maka jika terjadi persilangan, gen-gen yang akan disilangkan sejumlah 3 buah gen.

6) Tingkat Mutasi (*mutation rate*)

Variabel yang berupa angka persentase ini akan mempengaruhi cukup banyak terjadinya mutasi dalam suatu populasi. Variabel TM merupakan salah satu variabel yang berbentuk peluang, artinya kemungkinan terjadinya mutasi dilihat tiap individunya. Misalkan $JI=10$, angka TM diisi 0,5, bukan berarti akan ada 5 mutasi (0.5×10), tetapi peluang terjadinya mutasi untuk masing-masing individu adalah 0.5. Jadi dalam contoh diatas bisa saja terjadi 0 mutasi sampai 10 mutasi.

c. Pembangkitan Populasi Individu

Populasi merupakan kumpulan beberapa individu. Semua populasi dalam Algoritma Genetika ini berasal dari satu populasi yaitu populasi awal. Solusi atau kromosom terbaik dari populasi awal ini akan dipertahankan, dan akan mengalami proses evolusi untuk mendapatkan kemungkinan solusi yang lebih baik.

Pembuatan populasi awal ini dilakukan melalui proses pemilihan secara acak dari seluruh solusi yang ada. Pemilihan acak ini menyebabkan populasi awal dari Algoritma Genetika tidak akan sama dalam setiap kali percobaan, meskipun semua nilai variabel yang digunakan sama.

Dalam hal ini populasi adalah sebuah string yang berisi gen yang berjumlah sesuai dengan jumlah praktikum. Sebagai contoh apabila terdapat suatu gen yang panjangnya 3 karakter, maka panjang string adalah $3 \times n$, dengan n merupakan jumlah mata pelajaran yang akan di-generate.

Semua kelas yang akan disimpan pada palet-palet, terlebih dahulu akan disimpan pada sebuah array. Array tersebut berisi informasi urutan kelas tersebut pada array, dan posisinya ketika akan dimasukkan ke dalam palet. Adapun skemanya dapat dilihat pada bagan berikut ini:

Pertemuan[i] = [kodemapel[i],kodemapel[i],.....]; Pertemuan[i] =
[kodemapel[i],kodemapel[i],.....];

Dst.

Misalnya dalam satu minggu diwajibkan ada pertemuan pelajaran Matematika sebanyak 2 jam pertemuan untuk masing-masing kelas. Dan dalam satu minggu terdapat 25 pilihan waktu dari Senin hingga Sabtu.

Hal ini berlaku untuk masing-masing mata pelajaran. Dan nilai permutasi ditentukan sebanyak jumlah pertemuan yang diinginkan.

d. Evaluasi Nilai Fitness

Untuk mengetahui baik tidaknya solusi yang ada pada suatu individu, setiap individu pada populasi harus memiliki nilai pembandingnya (fitness cost). Melalui nilai pembanding inilah akan didapatkan solusi terbaik dengan cara pengurutan nilai pembanding dari individu-individu dalam populasi. Solusi terbaik ini akan dipertahankan, sementara solusi lain diubah-ubah untuk mendapatkan solusi yang lain lagi, melalui tahap cross over dan mutasi (mutation).

Sebelum melakukan penempatan jadwal kelas dilakukan dua buah pengecekan terlebih dahulu, yaitu pencarian hari dan jam yang masih kosong dan pengecekan prioritas yaitu pada hari dan jam mana yang paling tinggi prioritasnya. Proses evaluasi ini melibatkan fungsi objektif yang merupakan formula untuk menentukan jumlah nilai error dan dikalkulasikan dengan nilai fitness.

Adapun evaluasi nilai fitness dilakukan dengan parameter sebagai berikut:

1. Tidak boleh terjadi tabrakan jadwal.
2. Tidak boleh terjadi penumpukan ruangan.
3. Mata pelajaran olahraga tidak boleh di atas jam ke 2.
4. Tidak boleh terjadi penumpukan jadwal yang sama dalam satu hari.

Apabila terdapat aturan-aturan yang dilanggar maka nilai fitness cost akan dikurangi sehingga hasilnya akan menjadi lebih jelek. Contoh aturan tentang fitness cost akan dijelaskan pada tabel 5 berikut:

Tabel 2. Contoh Tabel *Fitness Cost*

Aturan	<i>Fitness cost</i>
Jadwal bertabrakan	<i>Fitness cost</i> – (jumlah mata kuliah bentrok *100)
Ruangan bertabrakan	<i>Fitness cost</i> – (jumlah responsi bentrok *100)
Mapel olahraga di atas jam ke 2	<i>Fitness cost</i> – (jumlah matakuliah semester *10)
Penumpukan jadwal yang sama dalam sehari	<i>Fitness cost</i> – (jumlah praktikum tidak sesuai *100)

e. Penentuan Nilai Probabilitas

Penentuan nilai probabilitas didasarkan pada nilai *fitness* yang ditentukan pada poin sebelumnya. Nilai *fitness* akan ditotal dari semua iterasi pembangkitan dan dikalikan angka 1000 sebagai random probabilitas. Contoh penentuan nilai probabilitas dapat dilihat pada source pada tabel 3 di bawah ini:

Tabel 3. Sintaks untuk Penentuan Probabilitas.

```
public double evaluateawal() {
    double total = 0;

    double finest = 0;

    for (int i = 0; i<SIZE; i++) {
        total += this.nilaifungsi[i];
        finest += this.fitness[i];
    }

    setTotalfitness(finest);

    System.out.println("Total nilai fungsi = "+total);
    System.out.println("Total nilai fitness = "+finest); //set
    for (int i = 0; i<SIZE; i++) {

        double probability;
        probability
```

f. Seleksi Roulette Wheel

Melanjuti tahap sebelumnya yaitu penentuan probabilitas, maka yang dilakukan adalah seleksi fitness. Dalam kasus ini yang digunakan adalah seleksi menggunakan Roulette Wheel

(Mesin Roulette) dimana setiap individu yang memiliki nilai fitness tinggi akan memiliki nilai probabilitas untuk terpilih yang tinggi. Adapun bentuk bagannya adalah sebagai berikut:

- 1) Dihitung nilai fitness masing-masing individu.
- 2) Dihitung total fitness semua individu.
- 3) Dihitung probabilitas tersebut untuk jatah 1 – 1000.
- 4) Dibangkitkan bilangan acak 1 – 1000.
- 5) Dari bilangan acak yang dihasilkan, ditentukan mana individu yang terpilih dalam proses seleksi.

Contoh *Roulette Wheel* dapat dilihat pada *source* tabel 4 di bawah ini: Tabel 4. Contoh

Sintaks untuk *Roulette Wheel*.

```
public void RouletteWheel() {
    for (int i = 0; i < GA_1Individu.SIZE ; i++) {
        int indexProb = (int) (Math.random()*1000);

        System.out.println("Hasil      index      seleksi
Roulette      "+i+"      =      "+indexProb+"      index      individu      =
"+indexuntukRolet[indexProb]);
    }
}
```

g. Crossover (Persilangan)

Setelah menjalani proses seleksi, maka individu yang terpilih akan dilakukan Crossover. Tahapan ini akan menyilangkan dua individu yang ada dalam suatu populasi, untuk mendapatkan dua individu baru. Setelah tahap, maka akan didapat populasi baru yang jumlahnya 2 kali lipat dari populasi lama. Pada kasus penyusunan jadwal pelajaran ini, yang menjadi individu adalah satu urutan penyusunan jadwal pelajaran dalam satu minggu. Adapun contohnya ada pada tabel 5.

Tabel 5. Contoh Individu Penjadwalan.

Pertemuan	Kelas							
	X1	X2	X3	X4	XI IPA 1	XI IPA 2	...	Dst
Senin1	Mat1		Fis1					
Senin2	Fis1	Mat1						

Senin3	KN1	Fis1	Mat1					
Senin4	Bio1							
Selasa1	Seni1							
Selasa2								
Selasa3								
...
Sabtu4								

Individu 1 = 001 002 003 004

Individu 2 = 005 006 007 008

Keterangan:

Mat1 = 001, Fis1 = 002, KN1 = 003, dst

Dari populasi yang ada, diambil individu sepasang demi sepasang untuk disilangkan (*cross over*). Persilangan pada kasus ini dilakukan dengan memindahkan sebagian urutan pada satu individu dan menukarkannya dengan individu yang lain. Ada 2 macam cara yaitu dengan *Two Points* dan *Uniform*. Pada *Two Points Cross Over*, dipilih secara acak 2 titik yang akan disilangkan.

Individu 1 = 001 | 002 003 | 004

Individu 2 = 005 | 006 007 | 008

Sedangkan pada *Uniform Cross Over* ada penentuan persentase gen yang akan disilangkan misalkan 50%, angka ini nantinya perlu masukkan dari *user*. Individu 1 = 001 002 | 003 004
Individu 2 = 005 006 | 007 008

Setelah disilangkan, akan dilakukan pengecekan terhadap masing- masing individu, apakah terjadi pengulangan. Kedua individu yang telah disilangkan ini diperbaiki, sehingga tidak ada pengulangan lagi. Garis besarnya adalah setiap angka yang diulang ditukar dengan pasangannya, yaitu angka yang diulang di kromosom pasangannya.

Hasil untuk *Two Points Cross Over*:

Individu 1 = 001 006 007 004

Individu 2 = 005 002 003 008

Hasil untuk *Uniform Cross Over*:

Individu 1 = 001 002 007 008

Individu 2 = 005 006 003 004

Populasi baru yang dihasilkan ini akan dibandingkan dengan populasi terbaik yang telah ada dan bila populasi terbaru lebih baik nilai *fitness cost*-nya maka populasi ini akan menggantikan populasi terbaik yang pernah ada.

h. Mutasi

Cara lain untuk mendapatkan individu yang baru yaitu dengan mutasi. Probabilitas terjadinya mutasi gen pada suatu kromosom sangatlah kecil, karena itu dalam penerapannya pada Algoritma Genetika, probabilitasnya seringkali dibuat kecil, lebih kecil dari $\frac{1}{2}$ (mutation rate). Berbeda dengan tahap cross over, dimana satu individu perlu individu yang lain, pada tahap ini tidak membutuhkan individu yang lain untuk bermutasi. Dalam kasus ini dimungkinkan terjadi 2 macam mutasi dimana probabilitas terjadinya mutasi akan ditentukan user.

Mutasi pertama yang mungkin terjadi adalah perubahan urutan kelas praktikum. Hal ini dilakukan secara acak, diambil 2 angka (nomor kelas praktikum) dari satu individu, kemudian ditukar.

Contoh:

Sebelum mutasi : 001 002 003 004

Hasil 003 002 001 004

i. Pembangkitan Populasi Baru

Satu populasi baru telah terbentuk dengan selesainya mutasi. Populasi baru tersebut akan menjadi populasi awal bagi generasi selanjutnya dan Algoritma Genetika akan mengulang tahap 2 sampai 4 secara terus menerus sampai sejumlah generasi yang telah ditentukan.

3.4 Alat dan Bahan Penelitian

Adapun perangkat yang digunakan dalam penelitian ini ditunjukkan pada Tabel 6 sebagai berikut :

Tabel 6. Alat Penelitian

Perangkat Keras, meliputi	Perangkat Lunak, meliputi
1. Laptop Dell Inspiron 13 5000 Series 2. Processor Core i7 3. Memory RAM 8 GB 4. SSD 128 GB 5. Monitor 13 inch	1. OS Windows 10 2. Visual Studio Code 3. XAMPP 4. PhpMyadmin 5. CodeIgniter (CI)

Bahan penelitian yang dibutuhkan dalam perancangan sistem ini yaitu:

1. Visi, misi, tujuan, serta surat-surat dari SMKN 1 Kota Jambi
2. Informasi jadwal pelajaran pada SMKN 1 Kota Jambi.
3. Proses bisnis dari sistem yang sudah ada.
4. Infrastruktur teknologi informasi

BAB IV

JADWAL KEGIATAN PENELITIAN

4.1 Jadwal Penelitian

Penelitian ini direncanakan akan dilaksanakan dalam kurun waktu 4 (Empat) bulan dengan rincian kegiatan sebagai berikut :

Tabel 4.1. Jadwal Kegiatan

NO	KEGIATAN	BULAN															
		Des				Jan				Feb				Mar			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Perumuskan Masalah																
2	Studi Literatur																
3	Pengumpulan Data																
4	Melakukan Analisis Sistem (data)																
5	Membuat Permodelan Sistem (Perancangan) <i>Prototype</i>																
6	Penyusunan Laporan																
7	Luaran (Publikasi)																
4	Melakukan Analisis Sistem (data)																
5	Membuat Permodelan Sistem (Perancangan) <i>Prototype</i>																
6	Penyusunan Laporan																
7	Luaran (Publikasi)																

DAFATAR PUSTAKA

- Allen, F. (1999). Using Genetic Algorithm to find Technical Trading Rules. Journal Financial Economics .
- Arikunto, P. D. (2010). Manajemen Penelitian. Jakarta: Rineka Cipta.
- Bevan, N. (1995). Measuring Usability as Quality of Use. Software Quality Journal . Coley, D. A. (2000). An Introduction to Genetic Algorithm. Singapore: World Scientific Publishing Co. Pte. Ltd.
- Fikri, R. (2005). Pemrograman Java. Yogyakarta: Penerbit ANDI.
- General Functionality and Stability Test Procedure for Certified for Microsoft Windows Logo (2005)
- Gibbon, M. P. (2004). Sistem Penjadwalan Matakuliah menggunakan Algoritma Genetika (Studi Kasus Fakultas MIPA IPB).
- Guide to Advanced Software Testing. (2008). Norwood: Artech House
- Java. (2013). Learn About Java Technology. Diakses Mei 29, 2013, from Java.com: <http://www.java.com/en/about/>
- Kusumadewi, S. &. (2005). Penyelesaian Masalah optimasi dengan Teknik-teknok Heuristik. Yogyakarta: Graha Ilmu.
- Ladjamudin, A. B. (2006). Rekayasa Perangkat Lunak. Yogyakarta: Graha Ilmu.
- McConnell, S. (2004). Code Complete. Redmond: Microsoft Press.
- Mitsuo Gen, Runwei Cheng. (2000). Genetic Algorithm and Engineering Optimization. New York: John Willey & Sons, Inc.
- Nielsen, J. (1993). Usability Engineering. New York: Morgan Kaufmann.
- Pengenalan Pemrograman Java 1. (2007). Jakarta: J.E.N.I
- Sam'ani. (2012). Rancang Bangun Sistem Penjadwalan Perkuliahan dan Ujian Akhir Semester dengan Pendekatan Algoritma Genetika.
- Saputro, N. (2003). Pengenalan Huruf dengan memakai Algoritma Genetik. Integral . Software Engineering : A Practitioner's Approach. (2010). New

York: McGraw Hill Software Engineering and Testing. (2010). Sudbury:
Jones and Bartlett Publishers