

**PENCARIAN JALUR TERPENDEK MENUJU MASJID
MENGUNAKAN *BEE COLONY OPTIMIZATION***

*Diajukan Sebagai Syarat untuk Menyelesaikan
Pendidikan Program Strata-1 Pada
Jurusan Teknik Informatika Bilingual Fakultas Ilmu Komputer*



Oleh:

Puteri Dinda Monita

NIM : 09111402035

**JURUSAN TEKNIK INFORMATIKA BILINGUAL
FAKULTAS ILMU KOMPUTER UNIVERSITAS SRIWIJAYA**

2016

LEMBAR PENGESAHAN USULAN TUGAS AKHIR

PENCARIAN JALUR TERPENDEK MENUJU MASJID DI KOTA
PALEMBANG MENGGUNAKAN BEE COLONY OPTIMIZATION

Oleh :

PUTERI DINDA MONITA

NIM : 09111402035

Palembang, Mei 2016

Pembimbing I,

Rizki Primarta, M.T

NIP 197706012009121004

Pembimbing II,

Hadipurnawan Satria, Ph.D

NIP 198004182015109101

Mengetahui,

Ketua Jurusan Teknik Informatika

Rizki Primarta, M.T

NIP 197706012009121004

PENCARIAN JALUR TERPENDEK MENUJU MASJID DI KOTA PALEMBANG MENGGUNAKAN *BEE COLONY OPTIMIZATION*

**Oleh :
PUTERI DINDA MONITA
09111402035**

ABSTRAK

Travelling Salesman Problem (TSP) adalah masalah untuk mendapatkan rute terpendek dan harus melewati semua kota dengan masing-masing kota hanya boleh dilewati satu kali sampai orang tersebut kembali ke kota asalnya. Salah satu contoh objek TSP bisa berupa masjid. Masjid telah menjadi salah satu tempat tujuan yang sering dikunjungi oleh masyarakat, tetapi tidak semua masjid dapat dijangkau dan diketahui lokasinya oleh pengunjung. Penelitian ini menggunakan *Bee Colony Optimization* (BCO) sebagai metode untuk mencari rute terpendek masjid. Faktor masukan berupa nama, alamat, *longitude* dan *latitude* masjid yang didapat menggunakan *Google Maps*. Hasil penelitian ini menunjukkan bahwa jumlah masjid serta jumlah proses mempengaruhi hasil perhitungan untuk mendapatkan *fitness* terbaik dan juga penelitian ini menunjukkan penggunaan *Bee Colony Optimization* (BCO) dapat mencari rute terpendek masjid.

Kata kunci : Rute Terbaik, *Latitude*, *Longitude*, *Bee Colony Optimization*,.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
ABSTRAK	ii
DAFTAR ISI	iii
DAFTAR GAMBAR	vii
DAFTAR TABEL	ix
DAFTAR LAMPIRAN	x

BAB I PENDAHULUAN

1.1 Latar Belakang.....	I-1
1.2 Perumusan Masalah.....	I-3
1.3 Tujuan Penelitian.....	I-3
1.4 Manfaat Penelitian.....	I-3
1.5 Batasan masalah.....	I-3
1.6 Metodologi Penelitian.....	I-4
1.7 Metode Pengembangan Perangkat Lunak.....	I-5

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya.....	II-1
2.2 Optimisasi.....	II-2
2.2.1 Penyelesaian Masalah Optimisasi.....	II-2
2.3 <i>Travelling Salesman Problem (TSP)</i>	II-3
2.3.1 Pengertian <i>Travelling Salesman Problem (TSP)</i>	II-2
2.3.2 Pencarian Jalur Terpendek.....	II-5
2.4 <i>Bee Colony Optimization</i>	II-5
2.4.1 <i>Distance</i>	II-6
2.4.2 Fase Initial.....	II-7
2.4.3 <i>Improvement Solution</i>	II-8
2.4.3.1 <i>Swap Operation</i>	II-13
2.4.3.2 <i>Swap Sequence</i>	II-14

2.4.3.3	<i>Insert Operation</i>	II-15
2.4.3.4	<i>Insert Sequence</i>	II-15
2.6	<i>Extreme Programming</i>	II-16

BAB III ANALISIS DAN PERANCANGAN

3.1	Analisis Masalah.....	III-1
3.2	Analisis <i>Bee Colony Optimization</i>	III-1
3.2.1	Jumlah Masjid.....	III-2
3.2.2	<i>Distance</i> atau jarak.....	III-2
3.3	Analisis Kebutuhan Perangkat Lunak.....	III-4
3.3.1	Deskripsi Umum.....	III-4
3.3.2	Spesifikasi Kebutuhan Perangkat Lunak.....	III-5
3.3.3	Model <i>Use Case</i>	III-6
3.3.3.1	Aktor dan tujuan.....	III-6
3.3.3.2	Diagram <i>Use Case</i>	III-7
3.3.3.3	Skenario <i>Use Case</i>	III-8
3.3.4	Kelas Analisis.....	III-10
3.3.5	Diagram Sekuensial.....	III-11
3.3.5.1	Diagram sekuensial mengubah data masjid.....	III-13
3.3.5.2	Diagram sekuensial memproses rute terbaik.....	III-14
3.3.5.2	Subsequence <i>Dance</i>	III-15
3.3.6	Kelas Diagram.....	III-16
3.4	Perancangan Perangkat Lunak.....	III-17
3.4.1	Perancangan Basis Data.....	III-17
3.4.2	Perancangan Antarmuka.....	III-17

BAB IV IMPLEMENTASI DAN PENGUJIAN

4.1	Implementasi Perangkat Lunak.....	IV-1
4.1.1	Lingkungan Implementasi Perangkat Lunak.....	IV-1
4.1.2	Implementasi Kelas.....	IV-2
4.1.3	Implementasi Antarmuka.....	IV-4

4.2	Pengujian Perangkat Lunak.....	IV-6
4.2.1	Lingkungan Pengujian.....	IV-6
4.2.2	Rencana Pengujian.....	IV-7
4.2.3	Kasus Uji.....	IV-8
4.3	Hasil Percobaan.....	IV-4
4.3.1	Analisis Hasil Percobaan.....	IV-28

BAB V KESIMPULAN DAN SARAN

5.1	Kesimpulan.....	V-1
5.2	Saran.....	V-1

DAFTAR PUSTAKA.....	VI-1
----------------------------	-------------

DAFTAR GAMBAR

	Halaman
Gambar II-1. Lintasan Travelling Salesman Problem.....	II-5
Gambar II-2. <i>Arsitektur Jaringan Hopfield</i>	II-9
Gambar II-3. Garis Antara Dua Titik Pada Permukaan Bumi.....	II-11
Gambar II-4. Struktur Proses RUP (Kruchten, 2004).....	II-18
Gambar III-1. Skema Pengambilan Titik Koordinat.....	III-2
Gambar III-2. Skema perhitungan distance atau jarak.....	III-3
Gambar III-3. Diagram <i>use case</i> perangkat lunak pencarian jalur terpendek menuju masjid menggunakan <i>Bee Colony Algorithm</i>	III-7
Gambar III-4. <i>Use case</i> Kelas Analisis Mengubah Data Masjid.....	III-11
Gambar III-5. <i>Use case</i> Kelas Analisis Memproses Rute Terbaik.....	III-11
Gambar III-6. Diagram Sekuensial Mengubah Data Masjid.....	III-13
Gambar III-7. Diagram Sekuensial Memproses rute terbaik.....	III-14
Gambar III-8. Sub Diagram <i>Dance</i>	III-15
Gambar III-9. Kelas Diagram pencarian jalur terpendek.....	III-16
Gambar III-10. Antarmuka <i>form</i> utama.....	III-28
Gambar III-11. Antarmuka <i>form</i> ubah masjid	III-29
Gambar IV-1. Antarmuka <i>Form</i> Utama (activity_main).....	IV-5
Gambar IV-2. Antarmuka <i>Form</i> ubah masjid.....	IV-6
Gambar IV-3. Gambar hasil percobaan satu kali proses menggunakan tiga masjid.....	IV-17
Gambar IV-4. Gambar hasil percobaan satu kali proses menggunakan empat masjid.....	IV-18
Gambar IV-5. Gambar hasil percobaan satu kali proses menggunakan enam masjid.....	IV-19
Gambar IV-6. Gambar hasil percobaan dua kali proses menggunakan tiga masjid.....	IV-21
Gambar IV-7. Gambar hasil percobaan dua kali proses menggunakan empat masjid.....	IV-22

Gambar IV-8. Gambar hasil percobaan dua kali proses menggunakan enam masjid.....	IV-23
Gambar IV-9. Gambar hasil percobaan tiga kali proses menggunakan tiga masjid.....	IV-25
Gambar IV-10. Gambar hasil percobaan tiga kali proses menggunakan enam masjid.....	IV-28

DAFTAR TABEL

	Halaman
Tabel I-1. Pengembangan Perangkat Lunak dengan Metode RUP.....	I-5
Tabel III-1. Kebutuhan Fungsional Perangkat Lunak.....	III-5
Tabel III-2. Kebutuhan Non Fungsional Perangkat Lunak.....	III-6
Tabel III-3 Aktor dan Deskripsi.....	III-6
Tabel III-4. Definisi <i>Use Case</i>	III-8
Tabel III-5. Skenario <i>use case</i> mengubah data masjid.....	III-9
Tabel III-6. Skenario <i>use case</i> memproses rute terbaik.....	III-10
Tabel III-7. Perancangan Basis Data Pencarian Jalur Terpendek.....	III-17
Tabel IV-1. Daftar Implentasi Kelas.....	IV-2
Tabel IV-2. Rencana Pengujian <i>Use Case</i> Mengubah data masjid.....	IV-7
Tabel IV-3. Rencana Pengujian <i>Use Case</i> Memproses rute terbaik.....	IV-8
Tabel IV-4. Kasus Uji Pada Pengujian <i>Use Case</i> Mengubah Data Masjid.....	IV-10
Tabel IV-5. Kasus Uji Pada Pengujian <i>Use Case</i> Memproses rute terbaik.....	IV-13
Tabel IV-6. Hasil Percobaan Dengan Satu Kali Proses Menggunakan tiga Masjid.....	IV-16
Tabel IV-7. Hasil Percobaan Dengan Satu Kali Proses Menggunakan empat Masjid.....	IV-18
Tabel IV-8. Hasil Percobaan Dengan Satu Kali Proses Menggunakan enam Masjid.....	IV-19
Tabel IV-9. Hasil Percobaan Dengan Dua Kali Proses Menggunakan tiga Masjid.....	IV-20
Tabel IV-10. Hasil Percobaan Dengan Dua Kali Proses Menggunakan empat Masjid.....	IV-22
Tabel IV-11. Hasil Percobaan Dengan Dua Kali Proses Menggunakan enam Masjid.....	IV-23

Tabel IV-12. Hasil Percobaan Dengan Tiga Kali Proses Menggunakan tiga Masjid.....	IV-24
Tabel IV-13. Hasil Percobaan Dengan Tiga Kali Proses Menggunakan empat Masjid.....	IV-26
Tabel IV-13. Hasil Percobaan Dengan Tiga Kali Proses Menggunakan enam Masjid.....	IV-27

DAFTAR LAMPIRAN

	Halaman
Code Program.....	L1

BAB I

PENDAHULUAN

1.1 Latar Belakang

Optimasi pada dasarnya adalah sebuah usaha yang dilakukan untuk memperoleh solusi terbaik dalam menyelesaikan sebuah masalah (Talbi, 2009). Optimasi sendiri sangat berguna hampir di segala bidang dalam melakukan usaha secara efektif dan efisien untuk mencapai target yang diinginkan. Selain itu pembuat keputusan sering dihubungkan dalam proses masalah optimasi. Salah satu masalah optimasi adalah *Travelling Salesman Problem* (TSP). Meskipun TSP adalah sebuah masalah optimasi sederhana, tetapi biasanya TSP termasuk masalah umum optimasi yang sulit untuk diselesaikan (Kumbharana dan Pandey, 2013).

TSP sendiri adalah masalah untuk mendapatkan rute terpendek oleh seseorang dan harus melewati semua kota dengan masing-masing kota hanya boleh dilewati satu kali sampai orang tersebut kembali ke kota asalnya (Amin *et al.*, 2010). Sedangkan dalam pencarian jalur terpendek, kecepatan dan keakuratan pencarian lokasi merupakan bagian yang penting. Pengguna harus dapat mencari lokasi yang ditentukan dengan cepat dan akurat.

Beberapa objek pencarian terdapat dalam TSP salah satunya adalah masjid. Masjid adalah tempat beribadahnya umat muslim. Pulau Sumatera khususnya Palembang

menurut data dari Dewan Masjid Indonesia pada tahun 2015 memiliki 425 masjid yang terdaftar. Tetapi tidak semua masjid ini dapat diketahui dan dijangkau oleh semua orang.

Hal itu menjadi alasan dibuatnya aplikasi pencarian jalur terpendek ini, Aplikasi pencarian jalur terpendek tersebut diharapkan dapat membantu pengguna untuk menemukan rute terpendek ke suatu lokasi yang telah ditentukan.

Untuk mempermudah pengguna melakukan pencarian lokasi maka disediakan fitur pencarian. Fitur pencarian dapat diimplementasi dalam bahasa pemrograman dengan menggunakan algoritma tertentu. Ada banyak metode dalam menyelesaikan TSP, seperti *Dijkstra Algorithm*, *Self Organizing Neural Network*, *Nearest Neighbor*, *Hopfield Neural Network* dan *Bee Colony Optimization*.

Dalam penelitian ini akan digunakan *Bee Colony Optimization*. Yaitu algoritma yang efisien untuk mencari jalur terpendek dengan objek masjid di kota Palembang. Meskipun tidak optimal, hasil yang didapatkan dengan menggunakan algoritma *Bee Colony Optimization* akan mendekati optimal. Algoritma *Bee Colony Optimization* diadopsi dari perilaku koloni lebah dalam proses mencari dan mengeksploitasi makanan. Dengan penyebaran informasi mengenai arah, jarak dan kualitas dari sumber makanan dilakukan dengan cara tarian (*waggle dance*) kepada koloni lebah lain yang berada di sarang (Rokhani, 2012). Karena hal inilah algoritma *Bee Colony Optimization* banyak diterapkan dalam pencarian jalur terpendek.

Perumusan Masalah

Dari latar belakang yang telah dijelaskan pada sub bab sebelumnya, permasalahan tugas akhir ini adalah bagaimana membuat perangkat lunak yang mengimplementasikan Algoritma *Bee Colony Optimization* untuk pencarian jalur terpendek dalam mencari masjid di kota Palembang.

Tujuan Penelitian

Tujuan dari penelitian ini yaitu:

1. Menggunakan *Bee Colony Optimization* sebagai metode dalam pencarian jalur terpendek.
2. Membangun aplikasi yang bisa mencari jalur terpendek menuju masjid dikota Palembang.

Manfaat Penelitian

Manfaat dari penelitian ini yaitu:

1. Mengerti jalan kerja algoritma *Bee Colony Optimization*
2. Mampu mengembangkan suatu perangkat lunak yang dapat menentukan rute terpendek menuju masjid di kota Palembang
3. Perangkat lunak dapat memberi kemudahan dalam mencari rute terpendek dari beberapa data lokasi masjid yang telah ditentukan.

Batasan Masalah

Batasan masalah pada penelitian ini yaitu:

1. Menentukan rute terbaik dari beberapa data lokasi masjid yang ada, bukan mencari masjid terdekat dari tempat kita berada.
2. Masjid yang digunakan adalah masjid yang ada dikota Palembang.
3. Rute yang dilalui adalah perbandingan rute terpendek yang ada di database.

Metodologi Penelitian

Untuk melaksanakan penelitian ini, penelitian akan melalui tahap-tahap berikut:

1. Mempelajari semua konsep algoritma *Bee Colony Optimization*.
2. Membuat proposal penelitian berdasarkan konsep-konsep yang dipelajari.
3. Mengambil data berupa nama-nama masjid yang ada di kota Palembang.
4. Mengembangkan sistem berdasarkan hasil rancangan arsitektur yang telah dibuat dengan metode XP.
5. Melakukan pengujian dan memastikan agar sistem dapat dikendalikan dari kegagalan yang akan terjadi..
6. Melakukan analisis dan pembahasan terhadap hasil pengujian perangkat lunak.
7. Menarik kesimpulan dari hasil analisis.
8. Menyelesaikan laporan penelitian

1.7 Pengembangan Perangkat Lunak dengan Metode XP

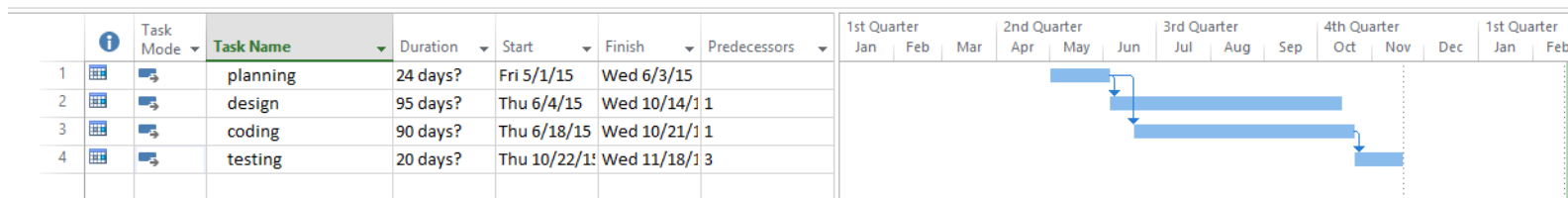
Pada sistem pencarian jalur terpendek menggunakan metode XP untuk membantu proses pengembangan system ini. Berikut ada empat fase pada XP yaitu:

Tabel I-1. Pengembangan Perangkat Lunak dengan Metode XP

Planning	<ul style="list-style-type: none"> - Mendapatkan informasi kebutuhan pengguna - Membuat daftar usecase dan deskripsi - Membuat scenario usecase yang menjadi gambaran dasar dari perangkat lunak
Design	<ul style="list-style-type: none"> - Mendesain tampilan awal prototype perangkat lunak dan penjelasan singkat cara pemaikaian nya - Membuat kelas analisis, kelas sekuensial serta kelas diagram untuk usecase pengolahan system dan pengujian - Melakukan dokumentasi dokumentasi terhadap kelas analisis, diagram sekuensial dan kelas diagram keseluruhan dalam bentuk laporan
Coding	<ul style="list-style-type: none"> - Mengidentifikasi rancangan antarmuka - Melakukan revisi prototype antarmuka menggunakan Microsoft Office Visio 2010

	<ul style="list-style-type: none"> - Melakukan implementasi coding dalam bahas PHP sesuai dengan diagram sekuensial untuk melihat hubungan antar kelas system
Test	<ul style="list-style-type: none"> - Melakukan implementasi rencana pengujian terhadap system - Pengujian versi akhir dari system mencari jalur terpendek menuju masjid dengan tiga kali proses pengujian - Dokumentasi hasil pencarian jalur terpendek menuju masjid

Berikut adalah Gantt Chart dari pengembangan system ini:



Gambar I-1. Gantt Chart dari sistem

Terlihat pada gambar I-1 bahwa planning membutuhkan waktu 24 hari. Saat selesai melakukan planning dilanjutkan dengan design dan coding. Design dan Coding dilakukan secara bersamaan, dengan masing-masing design 95 hari dan coding 90 hari. Setelah coding selesai dibuat maka dilanjutkanlah dengan Testing.

Sistematika Laporan

Sistematika penulisan laporan tugas akhir ini adalah sebagai berikut:

1. Bab I: Pendahuluan

Bab I berisi penjelasan mengenai latar belakang, perumusan masalah, tujuan dan manfaat penelitian, metodologi penelitian, metode pengembangan perangkat lunak dan sistematika penulisan laporan tugas akhir.

2. Bab II: Tinjauan Pustaka

Bab II berisi landasan teori yang akan digunakan dalam analisis, perancangan, serta implementasi tugas akhir.

3. Bab III: Analisis dan Perancangan

Bab III ini berisi analisis dan perancangan pengembangan aplikasi pencarian jalur terpendek berbasis yang menggunakan algoritma *Bee Colony Optimization*.

4. Bab IV: Implementasi dan Pengujian

Bab IV membahas lingkungan implementasi algoritma *Bee Colony Optimization* pada aplikasi pencarian jalur terpendek, implementasi program, hasil eksekusi dan hasil pengujian terhadap perangkat lunak.

5. Bab V: Kesimpulan dan Saran

Bab ini berisi kesimpulan yang didapatkan dari semua uraian dalam bab-bab sebelumnya, serta saran untuk pembangunan aplikasi pencarian jalur terpendek, optimal, dan mudah dipergunakan.

DAFTAR PUSTAKA

Ahmed, Z. H. (2010). Genetic algorithm for the traveling salesman problem using sequential constructive crossover operator. *International Journal of Biometrics & Bioinformatics (IJBB)*, 3(6), 96.

Amin, A. R., Ikhsan, M., & Wibisono, L. (2003). Traveling Salesman Problem.

Fan, D., & Shi, P. (2010, August). Improvement of Dijkstra's algorithm and its application in route planning. In *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on* (Vol. 4, pp. 1901-1904).

IEEE.

Girsang, A. S., Tsai, C. W., & Yang, C. S. (2012, September). A Fast Bee Colony Optimization for Traveling Salesman Problem. In *Innovations in Bio-Inspired Computing and Applications (IBICA), 2012 Third International Conference on* (pp. 7-12). IEEE.

Mohamed, E. T. T. A. O. U. I. L., Chakir, L. O. Q. M. A. N., Youssef, H. A. M. I., & Khalid, H. A. D. D. O. U. C. H. (2012). Task Assignment Problem Solved by Continuous Hopfield Network. *matrix*, 1(1), 1.

Puspitorini, S. (2008). Penyelesaian Masalah Travelling Salesman Problem dengan Jaringan Saraf Self Organizing. *Media Informatika*, 6(1), 39-55.

Sagayam, R., & Akilandeswari, M. K. Comparison of Ant Colony and Bee Colony Optimization for Spam Host Detection. *International Journal of Engineering Research and Developments ISSN*.

Singh, A., & Narayan, D. (2012). Augmentation of Travelling Salesman Problem using Bee Colony Optimization.

http://codingjunkie.blogspot.com/2012_10_01_archive.html

http://en.wikipedia.org/wiki/Shortest_path_problem

<http://www.geodatasource.com/developers/javascript>

<http://www.radford.edu/~nokie/classes/360/graphs-single-source-shortest-paths.html>

<http://www.dream.co.id/sim/sumatera-selatan/kota-palembang/.html/index43.html>