



JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

Nama : Priyanka Gemet Arismoyo
NRP : 5109100178
Dosen Wali : Dwi Sunaryono, S.Kom, M.Kom

2. JUDUL TUGAS AKHIR

Penerapan Algoritma *Ant Colony* untuk *Travelling Salesman Problem* pada Perangkat Bergerak.

Application of Ant Colony Algorithm for the Traveling Salesman Problem on Mobile Devices.

3. ABSTRAKSI

Dalam menyampaikan suatu informasi kepada khalayak ramai, terdapat beberapa cara, salah satunya adalah melalui media cetak (yang bisa berupa koran, pamflet, brosur hingga reklame) dan media elektronik. Penggunaan media cetak, terutama brosur acapkali terbagi secara percuma. Hal ini, sering disebabkan oleh pembagian yang tidak merata ataupun terdapat titik pembagian yang dikunjungi lebih dari sekali. Suatu titik terkadang dilewati lebih dari sekali dikarenakan tidak ada penanda yang mengatakan bahwa daerah tersebut telah dikunjungi sebelumnya.

Solusi yang sering digunakan adalah dengan berkomunikasi menggunakan telepon genggam baik itu melalui telepon atau layanan pesan singkat. Namun dalam pelaksanaannya sering terkendala dengan situasi dari pengguna, yakni ketika

dilakukan panggilan telepon, penerima sedang tidak bisa menjawab. Sementara dengan pesan singkat, pengguna terkadang tidak mengetahui adanya pesan yang masuk ke telepon genggam. Sehingga dibutuhkan cara untuk melakukan pemetaan jalur dari masing-masing orang untuk mengunjungi titik pembagian yang memungkinkan dalam suatu daerah dan diperlukan adanya suatu penunjuk atau cara memberitahu kepada pihak lain bahwa titik tersebut sudah pernah dikunjungi andaikata ada pihak yang berada di titik yang pernah dikunjungi.

Solusi yang ditawarkan adalah dengan menggunakan algoritma *ant colony* [1] untuk pemetaan jalur pembagian dan penggunaan teknologi *location based service* sebagai pengingat [2] ketika pengguna berada di suatu lokasi pembagian. Selain itu, perangkat lunak nantinya juga menampilkan titik – titik mana saja yang sudah dikunjungi dan tentunya rute dari pengguna itu sendiri. Tujuan dari pengerjaan tugas akhir ini adalah untuk membantu penyelesaian permasalahan pembagian rute agar semua titik bisa mendapatkan kunjungan dari *salesman* tanpa perlu dilakukan kunjungan secara berulang. Pengguna dapat mengetahui lokasi mana saja yang sudah dikunjungi.

4. PENDAHULUAN

4.1. LATAR BELAKANG

Penyebaran informasi dapat dilakukan dengan berbagai cara, bisa dengan media cetak maupun media elektronik. Penyebaran melalui media cetak selain dilakukan dalam media koran, juga sering dilakukan dalam bentuk brosur. pembagian brosur yang biasa dilakukan adalah dengan mengirim beberapa tim pembagi yang sering berupa *salesmen* ke beberapa titik dalam satu daerah. Permasalahan yang sering kali timbul adalah kurangnya koordinasi yang mengakibatkan tidak semua titik pembagian terkunjungi. Bahkan ada titik yang dikunjungi lebih dari sekali.

Permasalah pengunjungan suatu tempat hanya sekali dalam suatu daerah merupakan permasalahan *Travelling Salesman Problem* (TSP). TSP sendiri bertujuan untuk menemukan rute terpendek yang mengunjungi tiap kota sekali dan kembali ke kota awal. TSP merupakan masalah klasik dalam optimasi kombinatorial dan termasuk dalam kelas *NP-complete problems*. Jenis permasalahan ini hampir tidak dapat dipecahkan jika mencari solusi yang tepat secara langsung [3]. TSP selama ini

dilakukan hanya dengan perhitungan yang statis dimana ketika sudah dilaksanakan perhitungan, maka rute tersebut akan tetap menjadi panutan. Permasalahan yang terjadi nantinya adalah ketika sudah terbagi rute, tetapi dalam pelaksanaan salah satu titik atau kota tidak dapat dikunjungi. Maka dibutuhkan sebuah penghitungan rute kembali yang harus diketahui hasilnya oleh *salesmen* yang bertugas..

4.2. RUMUSAN MASALAH

Rincian permasalahan yang dihadapi dalam pembuatan Tugas Akhir ini adalah sebagai berikut:

1. Bagaimana mengimplementasikan algoritma *ant colony* pada permasalahan TSP pada studi kasus pembagian brosur?
2. Bagaimana menerapkan pengingat tugas berdasarkan lokasi pada telpon genggam android?

4.3. BATASAN MASALAH

Permasalahan yang diajukan dalam pembuatan Tugas Akhir ini akan diberi batasan untuk membatasi lingkup pengerjaan Tugas Akhir. Batasan-batasan tersebut adalah sebagai berikut :

1. Perangkat lunak dibangun pada platform Android dengan versi minimum 2.2.
2. Menggunakan *Global Positioning System* (GPS) yang terdapat pada perangkat berbasis Android.
3. Peta yang digunakan berasal dari Google Maps.
4. Algoritma yang diterapkan *ant colony*.
5. Titik pembagian sudah ditetapkan terlebih dahulu.
6. Pengguna aplikasi statis sehingga tidak bisa menambah pengguna ketika rute sudah terbagi.
7. Pada pelaksanaan uji coba hanya menggunakan satu pengguna saja.

4.4. TUJUAN

Tujuan dari pembuatan Tugas Akhir ini adalah untuk merancang dan membangun sebuah perangkat lunak berbasis *mobile* sehingga pergerakan *salesman* dapat dimaksimalkan. Aplikasi ini nantinya diharapkan dapat membantu mengoptimalkan kegiatan dari para *salesman* agar dapat menjadi petunjuk kemana

salesman bergerak selanjutnya pada satu rute tersebut serta dapat menjadi *reminder* jika sudah berada di lokasi pembagian.

5. TINJAUAN PUSTAKA

5.1. GPS

GPS merupakan sistem navigasi satelit yang dikembangkan oleh Departemen Pertahanan Amerika Serikat (US DoD = *United States Department of Defense*). GPS memungkinkan kita mengetahui posisi geografis kita (lintang, bujur, dan ketinggian di atas permukaan laut). Jadi dimanapun kita berada di muka bumi ini, kita dapat mengetahui posisi kita dengan tepat.

Perangkat GPS menerima sinyal yang ditransmisikan oleh satelit GPS. Dalam menentukan posisi, kita membutuhkan paling sedikit 3 satelit untuk penentuan posisi 2 dimensi (lintang dan bujur) dan 4 satelit untuk penentuan posisi 3 dimensi (lintang, bujur, dan ketinggian). Semakin banyak satelit yang diperoleh maka akurasi posisi kita akan semakin tinggi. Untuk mendapatkan sinyal tersebut, perangkat GPS harus berada di ruang terbuka. Apabila perangkat GPS kita berada dalam ruangan atau kanopi yang lebat dan daerah kita dikelilingi oleh gedung tinggi maka sinyal yang diperoleh akan semakin berkurang sehingga akan sukar untuk menentukan posisi dengan tepat atau bahkan tidak dapat menentukan posisi [4].

5.2. Google Maps API

Google Maps API (*Application Programming Service*) merupakan API yang di gunakan jika ingin menambahkan fitur Google Maps kedalam perangkat lunak yang dikembangkan. Google Maps merupakan salah satu layanan gratis yang dimiliki Google untuk peta digital yang menawarkan tampilan peta dan gambar dari satelit. API yang telah disediakan oleh Google Maps ini juga memungkinkan untuk melakukan *overlay* dengan data tertentu pada peta termasuk *overlay* untuk menggambarkan jalan, posisi, dan beberapa objek lain.

5.3. *Web Service*

Web service adalah segala pelayanan yang tersedia di internet, menggunakan format data standar XML dan tidak bergantung pada sistem operasi atau bahasa pemrograman [5]. Penggunaan dari *web service* adalah sebagai penyedia layanan kepada sistem lain, sehingga bisa saling berinteraksi dengan sistem penyedia *web service*. Informasi yang tersedia memiliki format XML, sehingga data dapat diakses oleh system yang lain meski terdapat perbedaan platform, sistem operasi maupun bahasa pemrograman. *web service* berjalan di port 80 yang merupakan port standar HTTP sehingga tidak diperlukan pengaturan khusus di sisi *firewall*.

5.4. *Location Based Service*

LBS merupakan *mobile service* yang memiliki kapabilitas untuk menyediakan informasi secara *real time* berdasarkan lokasi pengguna [1]. Ada tiga tahap yang secara umum digunakan dalam LBS. Pertama, mengirim informasi lokasi atau menyimpannya dalam suatu database. Kedua, menggunakan informasi dari lokasi untuk menyimpan aksi yang akan dilakukan. Yang terakhir, atau yang ketiga perpindahan lokasi atau ketika berada di lokasi akan menjadi pemicu dari aksi yang telah disimpan.

5.5. *Ant System*

Merupakan teori yang terinspirasi dari kebiasaan nyata semut di alam. Setiap semut meninggalkan sejumlah *pheromone* di jalan yang mereka lalui dan akan memilih jalur yang memiliki paling banyak *pheromone* sisa hasil dari semut yang melewati jalur tersebut. Sehingga sejak banyaknya semut yang melewati jalur tersingkat akan mengakibatkan banyaknya semut yang akan memilih rute terpendek tersebut. Hasilnya, koloni semut akan menemukan rute terbaik untuk mencapai makanan [1].

Semut menggunakan *pheromone* sebagai alat bertukar berita [6]. Misalkan beberapa semut secara acak mencari makanan dan mereka menemukan dua rute berbeda antara sarang dan sumber makanannya. Maka semut yang menemukan makanan akan memberitahukan kepada semut lainnya untuk mengikuti jalur mereka. Pada awalnya kedua jalur akan dilewati, namun dengan semakin banyaknya feromon yang di tinggalkan di jalur yang mereka lalui, maka semakin

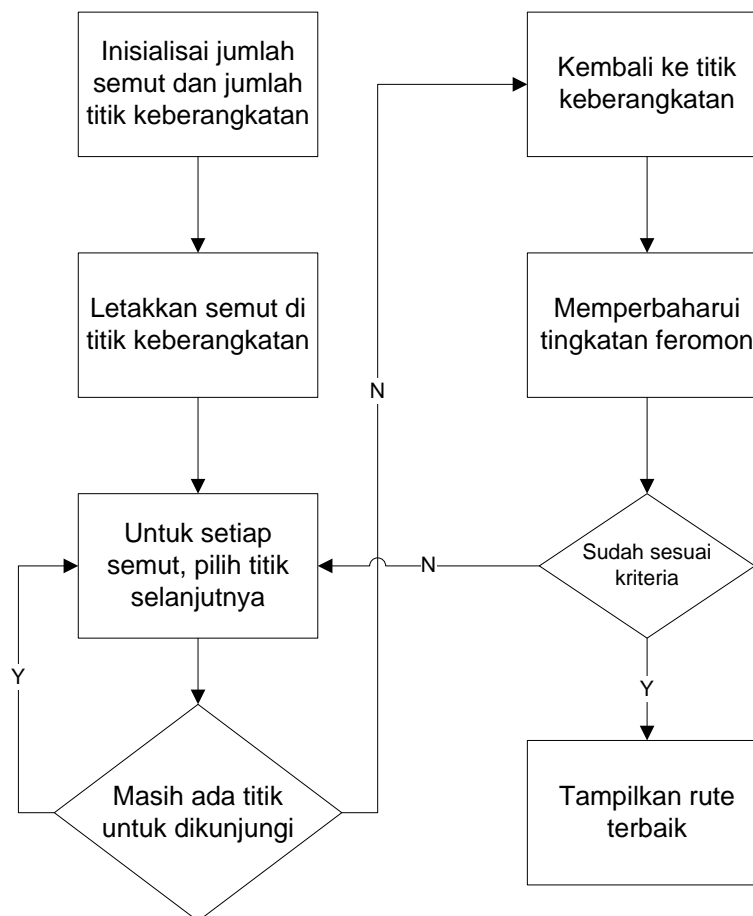
kuat feromon yang tercium, maka rute tersebut sering dilalui atau ditandai oleh feromon lebih sering. Maka secara otomatis semut lainnya akan menganggap rute tersebut adalah rute tercepat untuk mendapatkan makanan.

6. METODOLOGI

Dalam Tugas Akhir ini, penulis mengembangkan sebuah perangkat lunak pada perangkat bergerak untuk permasalahan TSP pada perangkat bergerak. Selain itu, aplikasi ini nantinya dapat menjadi penunjuk ke titik pembagian selanjutnya dan mengingatkan untuk berbagi informasi jika sudah berada di sekitar lokasi. Perangkat lunak ini khusus ditujukan pada perangkat bergerak yang mempunyai sistem operasi Android. Data mengenai rute akan disimpan di server yang nantinya akan diakses oleh perangkat *mobile* dengan teknologi *web service*. Dalam mengingatkan untuk berbagi informasi, maka aplikasi akan menggunakan konsep *Location Based Service* atau yang biasa disebut LBS dengan dibantu oleh GPS (*Global Positioning System*) [2]. Dalam membuat pengingat berbasis lokasi, langkah yang diperlukan adalah mendapatkan koordinat lokasi, kemudian menambahkan aksi apa yang akan dilakukan setelah itu semua data disimpan [7].

Cara kerja aplikasi ini, pada awalnya pengguna akan melakukan pemilihan terhadap titik keberangkatan yang tersedia. Data tersebut akan dikirim ke server dan diperhitungkan rute yang akan dilewati. Kemudian pengguna dapat mengetahui rute yang akan dilalui serta titik yang akan dikunjungi dari data yang dikirim server ke klien. Data yang dikirimkan ke klien merupakan urutan dari lokasi yang akan dikunjungi. Setelah selesai melakukan pemilihan titik keberangkatan, aplikasi akan mendapatkan hasil penghitungan rute dari server dan berjalan di *background*. Aplikasi akan kembali aktif apabila pengguna telah berada di lokasi pembagian dan mengingat untuk melaporkan kegiatan mereka. Ketika titik pembagian tersebut telah dilewati, maka server akan memberi tanda terhadap titik tersebut jika sudah pernah dikunjungi. Pengguna dapat mengurangi jumlah titik yang akan dikunjungi jika dalam pelaksanaannya, ditemukan kendala yang mengakibatkan daerah tersebut tidak bisa dikunjungi ataupun terjadi kemacetan parah yang bisa mengakibatkan terhambatnya pembagian. Pengurangan titik disini tidak berarti membuang titik tersebut selamanya, hanya dihilangkan sementara agar tidak masuk dalam proses pencarian rute. Jika

pengguna melakukan pengurangan titik, maka aplikasi akan melakukan pencarian rute baru dengan tidak mengikutsertakan rute tersebut. Namun, jika pengguna mengatakan pada rute tersebut terdapat kemacetan, maka aplikasi akan melakukan pencarian rute baru agar rute tersebut tidak dikunjungi terlebih dahulu. Terdapat dua hak akses dalam aplikasi ini, yang pertama adalah admin yang bisa menambah jumlah titik untuk dikunjungi dan pengguna biasa yang tidak bisa menambah titik kunjungan.

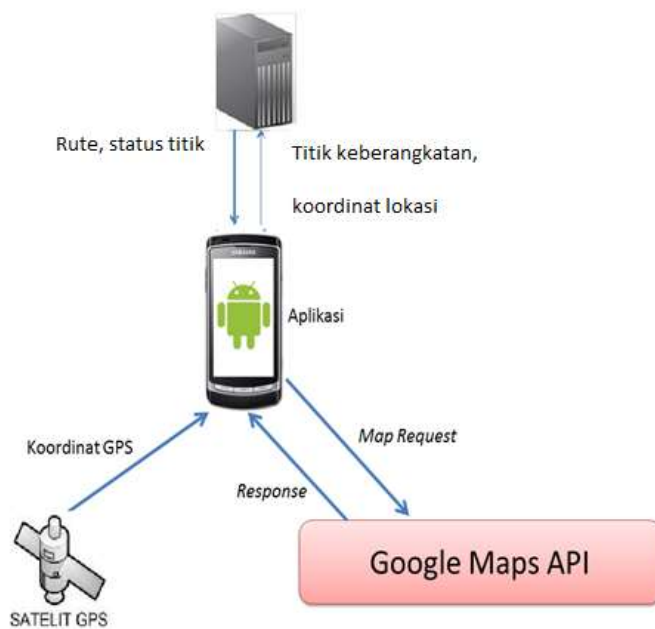


Gambar 1. Diagram alir pencarian rute

Algoritma *ant colony* digunakan untuk mencari rute dari titik keberangkatan hingga kembali ke titik awal. Cara pencarian rute menurut [8] dapat dilihat di Gambar 1, dimana pada awalnya dilakukan inisialisai terhadap banyaknya semut, banyaknya titik keberangkatan. Kemudian semut diletakkan di titik keberangkatan. Lantas dilakukan perulangan sebanyak jumlah semut, dimana untuk setiap semut pilih titik berikutnya. Cek apakah masih ada titik yang bisa dikunjungi, jika iya, kembali ke proses pemilihan titik selanjutnya, jika tidak maka kembali ke titik keberangkatan.

Kemudian memperbaharui tingkatan feromon menggunakan jarak dari masing masing semut.

Jarak yang ditempuh semut dalam keadaan rute aman atau tidak terjadi kemacetan akan berbeda jumlahnya dengan yang terjadi kemacetan. Karena rute yang mengalami kemacetan akan ditambah bobotnya sehingga akan menghasilkan rute yang berbeda. Lantas proses pengecekan rute akan berhenti jika memenuhi kriteria rute terpendek, dan akhirnya akan menampilkan rute terbaik. Untuk rute perjalanan, tim akan tetap mengikuti peta dari aplikasi kecuali jika terdapat suatu halangan di jalur tersebut, maka pengguna dapat meminta rute alternatif guna mencapai titik pembagian selanjutnya.



Gambar 2. Arsitektur aplikasi

Arsitektur aplikasi tampak pada Gambar 2, dimana pada gambar tersebut dapat diketahui bahwa pengguna harus memasukkan data inisialisasi berupa rute yang dipilih dan jumlah brosur yang dibawa. Setelah itu aplikasi secara *background* akan tetap memantau lokasi dari aplikasi apakah sudah berada di lokasi atau belum, jika sudah berada di lokasi, aplikasi akan mengaktifkan reminder untuk mengunggah data mengenai kegiatan mereka di lokasi tersebut.

Apabila pengguna meminta petunjuk rute jika melakukan perpindahan rute, maka aplikasi akan mengirim *request* dengan memanfaatkan Google Maps API untuk

mendapatkan data rute tercepat antara lokasi saat ini dengan lokasi tujuan. Yang menjadi tujuan ketika berpindah rute adalah titik kunjungan selanjutnya. Permintaan rute juga bisa dilakukan jika ada kendala antara satu titik dan titik lainnya di dalam satu rute. Data untuk lokasi yang berupa koordinat akan tersimpan didalam *database* pada server.

Untuk membantu mengingatkan pengguna sudah berada dilokasi adalah dengan mengecek apakah pengguna sudah berada di dalam radius titik pembagian. Untuk melakukan pengecekan, *device* pengguna melakukan *update* lokasi terkini setiap 2 menit yang kemudian akan di cek apakah sudah memasuki radius dari titik pembagian atau belum. Dari posisi yang di update terakhir kali, server akan menghitung jarak antara posisi dari *salesmen* yang lain dengan posisi titik pembagian yang akan dikunjungi selanjutnya. Untuk menghitung jarak, bisa digunakan *routing* Google Maps antara kedua titik tersebut.

$$d(x_r, x_s) = |x_r - x_s| = \sqrt{(x_{r1} - x_{s1})^2 + (x_{r2} - x_{s2})^2} \quad (1)$$

Dalam menghitung jarak antara dua titik koordinat, digunakan rumus *Euclidean* [9]. Rumus seperti yang ditunjukkan oleh Persamaan 1, merupakan perhitungan antara 2 koordinat. x_r dan x_s merupakan notasi dari 2 kordinat dengan x_{r1} dan x_{r2} merepresentasikan koordinat lintang dan bujur dari suatu lokasi begitu juga x_{s1} dan x_{s2} . D merupakan hasil penghitungan atau jarak antara kedua titik.

7. JADWAL Pengerjaan Tugas Akhir

Tahapan	Bulan											
	Maret				April				Mei			
Studi Kepustakaan												
Desain Sistem												
Implementasi												
Uji Coba dan Evaluasi												
Penyusunan Laporan Tugas Akhir												

8. DAFTAR PUSTAKA

- [1] Soheil Ghafurian and Nikbakhsh Javadian, "An ant colony algorithm for solving fixed destination multi-depot multiple travelling salesmen problems," *Applied Soft Computing*, vol. 11, pp. 1256-1262, Maret 2011.
- [2] Priyanka Shah, Ruta Gadgil, and Neha Tamhankar, "Location Based Reminder Using GPS For Mobile (Android)," *ARPN Journal of Science and Technology*, vol. 2, pp. 377-380, Mei 2012.
- [3] Gaifang Dong, William W. Guo, and Kevin Tickle, "Solving the traveling salesman problem using cooperative genetic ant systems," *Expert Systems with Applications*, vol. 39, pp. 5006-5011, 2012.
- [4] Anonim. (2013, Februari) GPS. [Online]. <http://www.navigasi.net/gofaq.php>
- [5] Ethan Cerami, *Web Service Essentials*.: O'Reily Media, 2002.
- [6] Harun Yahya. (2004) Menjelajah Dunia Semut. [Online]. http://id.harunyahya.com/id/books/769/menjelajah_dunia_semut/chapter/3014
- [7] Chi-Yi Lin, Ming-Tze Hung, and Wei-Hsun Huang, "A Location-based Personal Task Management Application for Indoor and Outdoor Environments," in *15th International Conference on Network-Based Information Systems*, 2012, pp. 582-587.
- [8] Ahmad Elshamli, Daniel Asmar, and Fadi Elmasri. (2003) Ant Colony Optimization.
- [9] Prof. Thomas B. Fomby. (2008, Februari) K-Nearest Neighbor Algorithm: Prediction and Classification.