

IMPLEMENTASI ALGORITMA KOLONI SEMUT DALAM PENCARIAN LOKASI TERDEKAT DI KABUPATEN SUMEDANG BERBASIS *SOCIAL SHARING WEBSITE*

¹Rikky Wisnu Nugraha, ²Nazar Haryadi

²Program Studi Teknik Informatika STMIK LPKIA

Jln. Soekarno Hatta No. 456 Bandung 40266, Telp. +62 22 75642823, Fax. +62 22 7564282

Email : r.wisnunugraha@gmail.com¹ , nazar.email@yahoo.com²

Abstrak

Kabupaten Sumedang merupakan daerah yang memiliki potensi wisata luar biasa, baik dari budaya maupun wisata alamnya, untuk meningkatkan potensi ini dibutuhkan wadah yang tepat untuk menampung dan mengelola informasinya. Pencarian lokasi tempat umum sangat diperlukan dalam hal ini, karena akan dapat membantu orang yang sedang berkunjung (wisatawan) yang tidak mengetahui jalur mana yang harus dilalui untuk menuju lokasi tempat yang ingin dituju. Dengan perancangan website ini dapat menjadi salah satu solusi untuk pengelolaan informasi tersebut. Dalam penerapannya dalam aplikasi dapat menggunakan beberapa metode algoritma pencarian rute. Salah satu metode yang terkenal baik dalam pencarian rute adalah metode koloni semut yang diambil dari perilaku koloni semut dalam pencarian jalur terpendek antara sarang dan sumber makanan. Kelebihan metode ini adalah menghasilkan rute yang baik untuk rute pendek, rute menengah, dan rute jauh serta tidak memerlukan jarak setiap tempat karena pada koloni semut jarak antar tempat dihitung setelah semut menyelesaikan perjalanannya. Dengan konsep sosial *share* setiap orang yang mempunyai informasi lokasi baru ataupun lokasi menarik di kota sumedang dapat ikut berpartisipasi dengan memasukan lokasi tersebut pada website sehingga setiap orang dapat melihat lokasi tersebut. Teknologi yang digunakan untuk membuat aplikasi ini adalah dengan menggunakan Google Maps v3 untuk modul map dan digabungkan dengan PHP sebagai bahasa pemrogramannya serta MySQL sebagai databasenya.

Kata kunci : *Lokasi, Koloni Semut, Social Share, Google Maps V3, PHP, MySQL*

1. Pendahuluan

Kabupaten Sumedang adalah salah satu kota yang sedang berkembang dan posisinya sebagai kota lalu lintas jalur selatan Jawa Barat membuat kota ini banyak dilalui oleh banyak orang. Pencarian lokasi tempat umum sangat diperlukan dalam hal ini, karena akan dapat membantu orang yang sedang berkunjung (wisatawan) yang tidak mengetahui jalur mana yang harus dilalui untuk menuju lokasi tempat yang ingin dituju. Seperti berita yang dimuat di media online pada website <http://www.sumedangonline.com/siapkah-sumedang-menjadi-tempat-kunjungan-wisata/3114/> dengan judul berita “SIAPKAH SUMEDANG MENJADI TEMPAT KUNJUNGAN WISATA” di posting tanggal Kamis, 05 April 2017 09:05:44 WIB disebutkan bahwa banyak keluhan dari tamu undangan di acara Kab. Sumedang yang kesulitan mencari penginapan, rumah makan dan kurangnya informasi tentang tempat wisata yang ada di Sumedang. Ini membuktikan kurangnya informasi yang disediakan oleh Pemerintah Kab. Sumedang dalam mengelola informasi pariwisatanya.

Dalam mengelola informasi pariwisatanya Kab. Sumedang membutuhkan wadah yang dapat menampung dan mengelola informasinya. Seperti aplikasi website pencarian rute lokasi. Dengan perancangan website ini dapat menjadi salah satu solusi untuk pengelolaan informasi tersebut. Dalam penerapannya dalam aplikasi dapat menggunakan beberapa metode algoritma pencarian rute. Salah satu metode yang terkenal baik dalam pencarian rute adalah metode koloni semut yang diambil dari perilaku koloni semut dalam pencarian jalur terpendek antara sarang dan sumber makanan. Kelebihan metode ini adalah menghasilkan rute yang baik untuk rute pendek, rute menengah, dan rute jauh serta tidak memerlukan jarak setiap tempat karena pada koloni semut jarak antar tempat dihitung setelah semut menyelesaikan perjalanannya. Dengan menggunakan website based, aplikasi ini dapat diakses dimana saja menggunakan internet sehingga bisa digunakan untuk mencari informasi ketika berada dalam keadaan darurat. Serta dengan konsep sosial *share* setiap orang yang mempunyai informasi lokasi baru ataupun lokasi menarik di kota sumedang dapat ikut berpartisipasi dengan

memasukan lokasi tersebut pada website sehingga setiap orang dapat melihat lokasi tersebut. Dengan adanya website informasi ini diharapkan dapat membantu Pemerintah Kab. Sumedang dalam mengelola informasi di daerah Sumedang dan dapat mengajak masyarakat Sumedang atau luar Sumedang untuk berpartisipasi dalam mengembangkan wilayah Sumedang. Serta dengan penggunaan algoritma koloni semut, website ini tidak hanya menampilkan informasi lokasi yang dituju tetapi juga disertai dengan estimasi jarak terdekat yang bisa ditempuh dari lokasi asal.

1.1 Rumusan Masalah

Berdasarkan latar belakang masalah, maka dapat dirumuskan suatu permasalahan yaitu :

1. Bagaimana membuat website informasi rute lokasi terdekat menggunakan algoritma koloni semut ?
2. Bagaimana menerapkan konsep *social sharing* pada website informasi lokasi ?

1.2 Batasan Permasalahan

Batasan permasalahan sebagai berikut:

Informasi lokasi yang ditampilkan hanya di daerah Kabupaten Sumedang dan hanya menentukan rute terdekat berdasarkan jarak.

1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dalam pengembangan sistem informasi ini antara lain :

1. Membuat website informasi lokasi yang dapat menampilkan informasi tujuan beserta rute terdekat yang dapat ditempuh.
2. Membuat website yang dapat memanfaatkan dan menampung informasi dari social publik dalam pengembangan atau penambahan informasi pada website.

2. Dasar Teori

2.1. Pengertian Koloni Semut

Algoritma Semut diadopsi dari perilaku koloni semut yang dikenal sebagai sistem semut (Dorigo, 1996). Secara alamiah koloni semut mampu menemukan rute terdekat dalam perjalanan dari sarang ke tempat-tempat sumber makanan. Koloni semut dapat menemukan rute terdekat antara sarang dan sumber makanan berdasarkan jejak kaki pada lintasan yang telah dilalui. Semakin banyak semut yang melalui suatu lintasan, maka akan semakin jelas bekas jejak kakinya. Hal ini akan menyebabkan lintasan yang dilalui semut dalam jumlah sedikit, semakin lama akan semakin berkurang kepadatan semut yang melewatinya, atau bahkan akan tidak dilewati sama sekali. Sebaliknya lintasan yang dilalui semut dalam jumlah banyak, semakin lama akan semakin bertambah kepadatan semut yang melewatinya, atau bahkan semua semut akan melalui lintasan tersebut. Algoritma yang

digunakan untuk menyelesaikan permasalahan pada penelitian ini, merupakan algoritma yang berdasarkan algoritma ant system dengan meningkatkan efisiensi pencarian rute yang dilalui. Konsep dasar dari algoritma koloni semut adalah dengan menggunakan sekumpulan semut yang bertujuan mencari lintasan terpendek secara paralel.

Dalam algoritma semut, diperlukan beberapa variabel dan langkah-langkah untuk menentukan jalur terpendek, yaitu:

Langkah 1:

a. Inisialisasi harga parameter-parameter algoritma. Parameter-parameter yang di inisialisasikan adalah:

1. Intensitas jejak semut antar kota dan perubahannya (τ_{ij})
2. Banyak kota (n) termasuk x dan y (koordinat) atau d_{ij} (jarak antar kota)
3. Tetapan siklus-semut (Q)
4. Tetapan pengendali intensitas jejak semut (α)
5. Tetapan pengendali visibilitas (β)
6. Visibilitas antar kota = $1/d_{ij}$ (η_{ij})
7. Banyak semut (m)
8. Tetapan penguapan jejak semut (ρ)
9. Jumlah siklus maksimum (NC_{max}) bersifat tetap selama algoritma dijalankan, sedangkan τ_{ij} akan selalu diperbaharui harganya pada setiap siklus algoritma mulai dari siklus pertama ($NC=1$) sampai tercapai jumlah siklus maksimum ($NC=NC_{max}$) atau sampai terjadi konvergensi.

b. Inisialisasi kota pertama setiap semut. Setelah inisialisasi τ_{ij} dilakukan, kemudian m semut ditempatkan pada kota pertama tertentu secara acak.

Langkah 2:

Pengisian kota pertama ke dalam *tabu list*. Hasil inisialisasi kota pertama setiap semut dalam langkah 1 harus diisikan sebagai elemen pertama *tabu list*. Hasil dari langkah ini adalah terisinya elemen pertama *tabu list* setiap semut dengan indeks kota tertentu, yang berarti bahwa setiap $tabu_k$ (I) bias berisi indeks kota antara 1 sampai n sebagaimana hasil inisialisasi pada langkah 1.

Langkah 3:

Penyusunan rute kunjungan setiap semut ke setiap kota. Koloni semut yang sudah terdistribusi ke sejumlah atau setiap kota, akan mulai melakukan perjalanan dari kota pertama masing-masing sebagai kota asal dan salah satu kota-kota lainnya sebagai kota tujuan. Kemudian dari kota kedua masing-masing, koloni semut melanjutkan perjalanan dengan memilih salah satu dari kota-kota yang tidak terdapat pada $tabu_k$ sebagai kota tujuan selanjutnya. Perjalanan koloni semut berlangsung terus menerus sampai semua kota satu persatu dikunjungi atau telah menempati $tabu_k$. Jika s menyatakan indeks

urutan kunjungan, kota asal dinyatakan sebagai $tabu_k(s)$ dan kota-kota lainnya dinyatakan sebagai $\{N - tabu_k\}$, maka untuk menentukan kota tujuan digunakan persamaan probabilitas kota untuk dikunjungi sebagai berikut:

$$P_{ij}^k = \frac{[\tau_{ij}]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{k' \in \{N - tabu_k\}} [\tau_{ik'}]^\alpha \cdot [\eta_{ik'}]^\beta} \text{ untuk } j \in \{N - tabu_k\}$$

dan $p_{ij}^k = 0$, untuk j lainnya

dengan i sebagai indeks kota asal dan j sebagai indeks kota tujuan.

Langkah 4:

a. Perhitungan panjang rute setiap semut. Perhitungan panjang rute tertutup (*length closedtour*) atau L_k setiap semut dilakukan setelah satu siklus diselesaikan oleh semua semut. Perhitungan dilakukan berdasarkan $tabu_k$ masing-masing dengan persamaan berikut:

$$L_k = d_{tabu_k(n), tabu_k(1)} + \sum_{s=1}^{n-1} d_{tabu_k(s), tabu_k(s+1)}$$

dengan d_{ij} adalah jarak antara kota i ke kota j yang dihitung berdasarkan persamaan:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

b. Pencarian rute terpendek. Setelah L_k setiap semut dihitung, akan didapat harga minimal panjang rute tertutup setiap siklus atau L_{minNC} dan harga minimal panjang rute tertutup secara keseluruhan adalah atau L_{min} .

c. Perhitungan perubahan harga intensitas jejak kaki semut antar kota. Koloni semut akan meninggalkan jejak-jejak kaki pada lintasan antar kota yang dilaluinya. Adanya penguapan dan perbedaan jumlah semut yang lewat, menyebabkan kemungkinan terjadinya perubahan harga intensitas jejak kaki semut antar kota. Persamaan perubahan ini adalah:

$$\Delta\tau_{ij} = \sum_{k=1}^m \Delta\tau_{ij}^k$$

dengan $\Delta\tau_{ij}^k$ adalah perubahan harga intensitas

jejak kaki semut antar kota setiap semut yang dihitung berdasarkan persamaan

$$\Delta\tau_{ij}^k = \frac{Q}{L_k}$$

untuk $(i,j) \in$ kota asal dan kota tujuan dalam $tabu_k$

$$\Delta\tau_{ij}^k = 0, \text{ untuk } (i,j) \text{ lainnya}$$

Langkah 5:

a. Perhitungan harga intensitas jejak kaki semut antar kota untuk siklus selanjutnya. Harga intensitas jejak kaki semut antar kota pada semua lintasan antar kota ada kemungkinan berubah karena adanya penguapan dan perbedaan jumlah semut yang melewati. Untuk siklus selanjutnya, semut yang akan melewati lintasan tersebut harga intensitasnya telah berubah. Harga intensitas jejak kaki semut antar kota untuk siklus selanjutnya dihitung dengan persamaan:

$$\tau_{ij} = \rho \cdot \tau_{ij} + \Delta\tau_{ij}$$

b. Atur ulang harga perubahan intensitas jejak kaki semut antar kota. Untuk siklus selanjutnya perubahan harga intensitas jejak semut antar kota perlu diatur kembali agar memiliki nilai sama dengan nol.

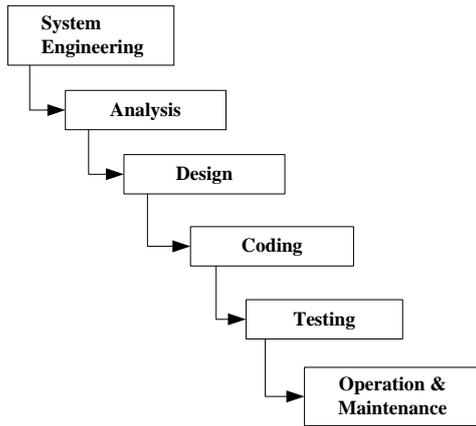
Langkah 6:

Pengosongan *tabu list*, dan ulangi langkah 2 jika diperlukan. *Tabu list* perlu dikosongkan untuk diisi lagi dengan urutan kota yang baru pada siklus selanjutnya, jika jumlah siklus maksimum belum tercapai atau belum terjadi konvergensi. Algoritma diulang lagi dari langkah 2 dengan harga parameter intensitas jejak kaki semut antar kota yang sudah diperbaharui.

(Sumber : Arna Fariza, Entin Martiana1, Fidi Wincoko Putro :*Dalam Jurnal Sistem Navigasi Perjalanan Berbasis Web Dengan Algoritma Koloni Semut (Ant Colony Algorithm)*." lecturer.eepis-its.edu, 27.08.14,22.30")

2.2 Metodologi Pengembangan Perangkat Lunak

Analisis dan pengembangan dalam membangun Sistem Informasi yang kompleks membutuhkan metoda – metoda atau paradigma pengembangan yang mampu membantu menganalisis dan mendesain secara lebih detail sehingga informasi yang dihasilkan lebih akurat. Dalam hal ini paradigma yang dipakai adalah paradigma *Waterfall Model* yang terdiri dari enam tahapan, yaitu *System Engineering, Analysis, Design, Coding, Testing, Operation and Maintenance*. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini :



Gambar 2.1 Waterfall Model

(Sumber : perangkat lunak persediaan suku cadang pada bagian gudang di bengkel duta family motor.Politeknik LPKIA Bandung.22.08.2014 22.43)

1. System Engineering

Merupakan tahap awal dalam membangun sebuah Sistem Informasi dengan mempelajari/menganalisis kebutuhan sistem, menganalisis data – data yang mengalir dan mempersiapkan perangkat penunjang (*Hardware, Software, Brainware*).

2. Analysis

Menentukan dapat di implementasikan menjadi sebuah Sistem Informasi atau tidak, menentukan prosedur – prosedur yang bekerja serta mengevaluasi situasi atau kelemahan sistem lama yang masih bersifat manual kemudian dibuat sistem yang lebih cepat, tepat dan akurat.

3. Design

Menerapkan hasil analisis ke dalam bentuk rancangan sistem. Perancangan ini meliputi

1. Rancangan Diagram UML.
2. Rancangan masukan dan keluaran

4. Coding

Menterjemahkan hasil rancangan ke dalam bentuk bahasa pemrograman.

5. Testing

Pengujian yang dilakukan baik per unit maupun secara integrasi. Adapun pengujian yang dilakukan menggunakan “ *Black-box Testing* “.Black-Box Testing terfokus pada spesifikasi fungsional dari perangkat lunak *Tester* dapat mendefinisikan kumpulan kondisi input dan melakukan pengujian pada spesifikasi fungsional program.Black Box Testing bukanlah solusi alternatif dari White-Box Testing tapilebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh White-Box Testing.

Black-Box Testing cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada

2. Kesalahan antarmuka (*interface errors*)

3. Kesalahan pada struktur data dan akses basis data

4. Kesalahan performansi (*performance errors*)

5. Kesalahan inisialisasi dan terminasi.

6. Operation dan Maintenance

Spesifikasi kebutuhan perangkat lunak dan keras yang dibutuhkan untuk menjalankan perangkat lunak serta pemeliharaan jika terjadi perubahan kondisi baik secara *Software* maupun *Hardware*.

2.3 Alat Pemodelan Perancangan Perangkat Lunak

Metode yang digunakan dalam perancangan perangkat lunak ini adalah *Object oriented* menggunakan pemodelan UML.

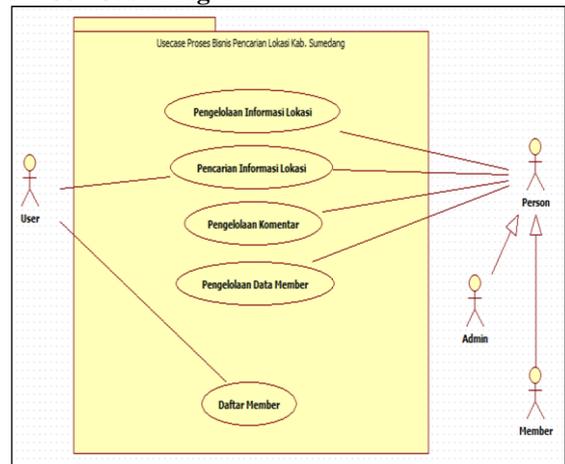
Unified Modelling Language (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak.UML menawarkan sebuah standar untuk merancang model sebuah sistem.

UML mendefinisikan diagram-diagram sebagai berikut:

1. use case diagram
2. class diagram
3. activity diagram
4. sequence diagram
5. collaboration diagram
6. component diagram
- deployment diagram

3. Gambaran Perangkat Lunak

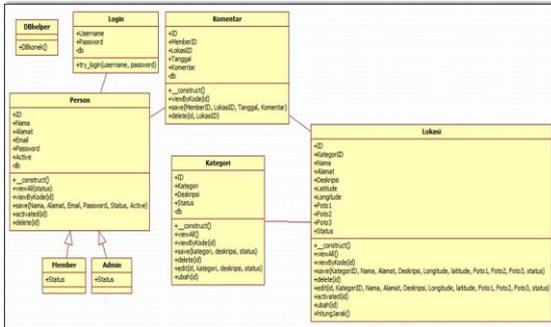
3.1 Use Case Diagram



Gambar 3.1 Use Case Diagram

3.2 Diagram Class(Class diagram)

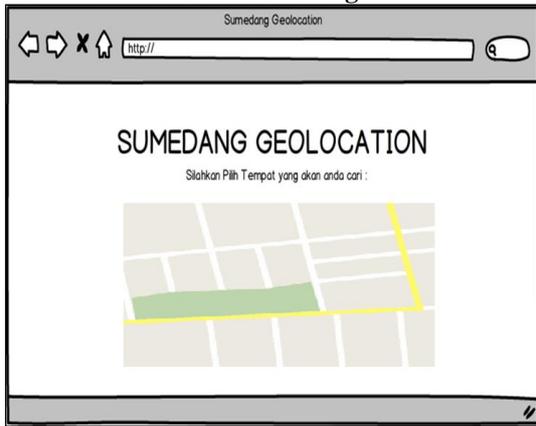
Diagram *class* dari sistem ini menunjukkan objek-objek yang terdapat dalam sistem serta keterhubungan objek-objek tersebut



Gambar 3.2 Class Diagram

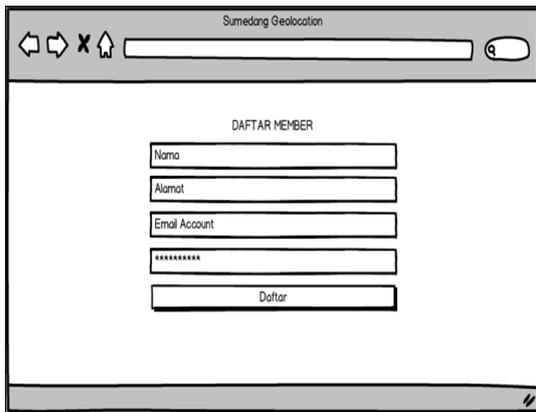
3.3 Perancangan Antarmuka

3.3.1 Antarmuka Welcome Page

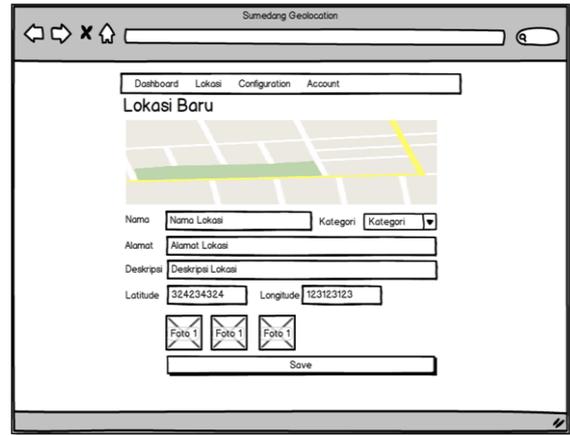


Gambar 3.3 Screen Shoot Welcome Page

3.3.2 Antarmuka Daftar Member



Gambar 3.4 Screen Shoot Daftar Member



Gambar 3.5 Peta Lokasi

4. Implementasi

Saran dari Perencanaan Layanan Sistem Informasi Studi Kasus Unit Departemen Umum di Politeknik

1. Requirement Gathering
Pada tahap ini menyiapkan seluruh kebutuhan data yang dibutuhkan oleh user dan menganalisa kebutuhan tersebut untuk digunakan membangun database dan perangkat lunak.
2. Analysis
Mempersiapkan kebutuhan perangkat keras dan perangkat lunak dalam pembangunan perangkat lunak ini. Mengumpulkan data yang mendukung terhadap perangkat lunak ini dengan memakai sumber dari *referensi buku* dan wawasan mengenai pemrograman.
3. Design
Proses pembangunan perangkat lunak yang meliputi tahapan desain database, arsitektur Perangkat Lunak, interface dan algoritma prosedural. Merujuk pada diagram UML yang digunakan.
4. Coding
Menerjemahkan desain ke dalam bahasa yang dimengerti mesin. Pada perancangan ini digunakan pemrograman PHP beserta CSS dan Javascript.
5. Testing
Tes pertama saat perangkat lunak sudah selesai dibangun yang pastinya masih terdapat kekurangan dan perlu beberapa sentuhan dan penambahan fungsionalitas dari perangkat lunak.
6. Maintenance
Proses perbaikan perangkat lunak dari hasil *testing* agar dapat digunakan kembali dengan lebih baik.

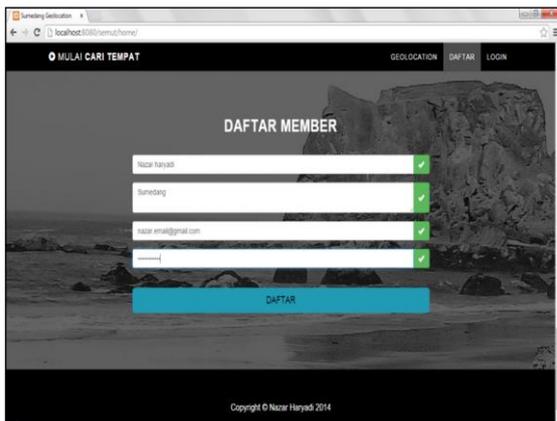
Kode Aktifitas	Aktifitas	Waktu (minggu)	Predecessor
A.	Requirement Gathering	1	-
B.	Analysis	3	A
C.	Desain	4	B
D.	Coding	3	C
E.	Testing	2	D
F.	Maintenance	2	E

Tabel 4.1 Jadwal Implementasi

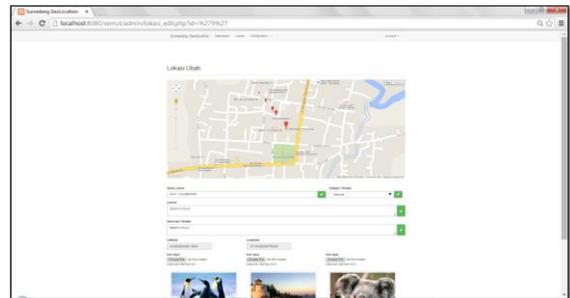
4.2 Implementasi Antarmuka Welcome Page



4.3 Implementasi Antarmuka Daftar Member



4.4 Implementasi Antarmuka Lokasi Baru



Gambar 4.3 Peta Lokasi

5. Kesimpulan

Kesimpulan dari implementasi algoritma koloni semut dalam pencarian lokasi terdekat di Kabupaten Sumedang berbasis *social sharing website* :

1. Dengan menggunakan Algoritma koloni semut dapat membantu mendapatkan perhitungan jarak optimal yang dapat ditempuh dari lokasi awal ke lokasi tujuan.
2. Konsep *social sharing* yang diterapkan pada perangkat lunak ini membantu pengguna untuk berpartisipasi mengumpulkan informasi lokasi di Kabupaten Sumedang.

DAFTAR PUSTAKA

1. Binanto, Iwan.(2005).*Konsep Bahasa Pemrograman*. Jakarta: PT Elex Media Komputindo.
2. Kadir, Abdul.(2008).Dasar Pemrograman Dinamis Menggunakan PHP. Yogyakarta: Andi.
3. Jurnal : Arna Fariza, Entin Martiana1, Fidi Wincoko Putro : *Sistem Navigasi Perjalanan Berbasis Web Dengan Algoritma Koloni Semut (Ant Colony Algorithm)*.“ lecturer.eepis-its.edu, 27.08.14,22.30”
4. Sumber : <http://dunia-teknik.com/pengertian-website.html> 17.58 29 Mei 2014
5. Sumber : *perangkat lunak persediaan suku cadang pada bagian gudang di bengkel duta family motor*.Politeknik LPKIA Bandung.22.08.2014 22.43
6. Sumber:<http://www.perkuliahan.com/apa-pengertian-studi-kepuustakaan/> 31.08.2014 12.01
7. Sumber : <http://www.anneahira.com/map-google.html> 18.15 29 Mei 2014
8. Sumber <http://id.wikipedia.org/wiki/MySQL> 31.08.2014 12.18