

## APLIKASI PENCARIAN SHELTER TSUNAMI TERDEKAT DI KOTA PADANG MENGUNAKAN METODE BEST FIRST SEARCH

Anisya<sup>1)</sup>, Ganda Yoga Swara<sup>2)</sup>

Dosen Teknik Informatika<sup>1,2)</sup>

Fakultas Teknologi Industri Institut Teknologi Padang

Email : [anisya@itp.ac.id](mailto:anisya@itp.ac.id)<sup>1)</sup>

---

### Abstrak

Salah satu kota di Sumatera yang rawan akan bencana tsunami adalah kota Padang, dimana kota Padang ini berada pada pertemuan 2 lempeng aktif yang merupakan sumber gempa kuat yang berpotensi tsunami. Proses evakuasi yang tepat akan membantu masyarakat kota Padang dalam menghadapi bencana tersebut. Serta penyebaran penduduk yang tidak merata disetiap wilayah kota Padang juga akan berpengaruh kepada proses evakuasi masyarakat. Dalam penelitian ini membahas mengenai penentuan jalur terpendek dengan menggunakan metode heuristik best first search. Dan dalam penentuan waktu tempuh saat evakuasi menggunakan model optimasi waktu minimum. Kota Padang sendiri sebagai obyek penelitian dibagi menjadi 10 cluster/bagian, yang masing-masing cluster mempunyai beberapa *safe area* atau disebut juga dengan shelter. Hasil perhitungan numerik yang diperoleh diimplementasikan pada sebuah sistem berbasis *smartphone*. Dalam penentuan lokasi awal pengguna menggunakan 2 cara, manual dan otomatis (*geolocator*). Keluaran dari sistem ini nantinya menampilkan beberapa jalur evakuasi beserta perkiraan waktu tempuh, panjang jalur, dan kepadatan penduduk yang akan melalui jalur tersebut.

**Kata kunci** : shelter tsunami, heuristik, *smartphone*

### 1. Pendahuluan

#### 1.1 Latar Belakang Masalah

Kota Padang merupakan ibu kota Provinsi Sumatera Barat, yang juga daerah rawan bencana gempa bumi dan tsunami, yang dikarenakan posisi kota Padang ini berada pada pertemuan 2 lempeng Hindia dan Asia pada pesisir pantai barat kota Padang serta dilalui oleh sesar Semangko. Wilayah laut kota Padang merupakan wilayah patahan subduksi yang merupakan sumber gempa kuat dan sumber pembangkit tsunami. Selain merupakan daerah rawan bencana, penyebaran kepadatan populasi juga sangat memprihatinkan karena setengah dari penduduk kota Padang bertempat tinggal dekat daerah pantai dan berada pada elevasi lima meter di atas permukaan laut, sedangkan untuk daerah evakuasi itu sendiri harus melebihi sepuluh meter di atas permukaan laut.

Berdasarkan catatan gempa yang terjadi sejak tahun 2006, 2007, dan 2009, kota Padang memiliki permasalahan dalam hal evakuasi tsunami. Hal ini disebabkan beberapa faktor antara lain : infrastruktur jalan yang tegak lurus arah pantai menuju daerah tinggi sangat sedikit. Banyaknya sungai-sungai yang melewati kota Padang, jika tsunami terjadi maka daerah sekitar aliran sungai sangat rawan terhadap

genangan tsunami dalam radius 200 meter. Banyaknya jalur evakuasi yang harus melewati jembatan sedangkan secara teori jembatan tidak tahan terhadap gempa dan tsunami. Dan juga permasalahan yang dihadapi pada pengalaman gempa yang pernah terjadi adalah kemacetan pada ruas-ruas jalan di kota Padang (Fauzan, 2011).

*Heuristic* sendiri menjelaskan sebuah teknik yang mengembangkan efisiensi dalam proses pencarian, namun dengan kemungkinan mengorbankan kelengkapan (T.Sutojo, 2011). Beberapa *tools heuristic* telah berkembang dalam 2 dekade terakhir dalam hal pemecahan masalah yang sebelumnya sulit untuk dipecahkan, teknik ini menemukan solusi yang paling besar kemungkinan untuk terjadi dalam suatu penelitian di mana *heuristic* ini memiliki kemampuan untuk mengoptimalkan kelengkapan kondisi ke fungsi objek yang berbeda (Balirsingh, 2011). Algoritma yang diperoleh akan diimplementasikan pada perangkat *smartphone*, guna memperoleh informasi secara cepat dan mudah dalam pencarian lokasi *shelter tsunami* terdekat di kota Padang dengan memanfaatkan algoritma *heuristic*.

Oleh karena itu perlu dibangun suatu kecerdasan buatan yang dapat membantu warga Kota Padang dalam memperoleh informasi shelter tsunami terdekat.

Penelitian ini sebelumnya telah dilakukan oleh peneliti dalam studi Pascasarjana, dan peneliti ingin melanjutkan penelitian ini yang jadi pembedanya pada perangkat yang akan digunakan nantinya dan cara membaca lokasi pengguna berada. Dalam penelitian ini, penulis membangun aplikasi untuk pencarian *shelter* tsunami pada perangkat *smartphone* agar mudah diakses dimana saja dan kapan saja. Pencarian *shelter tsunami* yang penulis buat adalah dengan menggunakan metode *Best First Search*.

## 1.2 Perumusan Masalah

Berdasarkan latar belakang diatas maka rumusan masalah pada penelitian ini berupa bagaimana merancang sebuah aplikasi android yang dapat membantu masyarakat dalam penentuan lokasi shelter terdekat.

## 1.3 Batasan Masalah

Agar penulis tidak keluar dari permasalahan yang ada dan hasil penelitian dapat diperoleh dengan baik, sempurna dan terarah, maka penulis membatasi ruang lingkup pembahasan. Selain itu, mengingat keterbatasan waktu, biaya dan kemampuan penulis, maka penelitian mengambil batasan - batasan masalah sebagai berikut :

1. Penelitian ini hanya ditujukan untuk mengatasi masalah-masalah yang terdapat dalam lingkungan masyarakat mengenai lokasi evakuasi.
2. Dibatasi hanya untuk mengatasi kesulitan masyarakat dalam pemilihan jalur evakuasi terdekat khusus zona merah (wilayah tepi pantai).
3. Dibatasi pada data-data yang diperoleh di daerah rawan atau 2 km dari garis pantai.
4. Keragaman datanya difokuskan di kota Padang

## 1.4 Tujuan Penelitian

Melalui hasil penelitian ini diharapkan akan dihasilkan sebuah yang mampu :

1. Menghasilkan suatu sistem aplikasi yang dapat membantu masyarakat dalam mendapatkan informasi shelter tsunami terdekat di Kota Padang secara *mobile*

dengan menggunakan aplikasi kecerdasan buatan metode *heuristic*.

## 2. Publikasi Ilmiah.

## 2. Tinjauan Pustaka

### 2.1 Tinjauan Pustaka

Sebagai tinjauan pustaka peneliti menggunakan beberapa referensi dari penelitian-penelitian yang sudah ada sebelumnya. Diantaranya adalah, Shanty Dewi, 2010 dengan judul penelitian Perancangan Aplikasi Informasi Pencarian Shelter Transit Transjakarta dengan Metode *Breadth First Search*. Serta Ifatul Faizah, 2010, dengan Rancang Bangun Perangkat Lunak Penentuan Rute. Perjalanannya Wisata di Malang Menggunakan Algoritma Dijkstra.

### 2.2 Metode Heuristic

*Heuristic search* adalah suatu strategi untuk melakukan proses pencarian secara selektif dan dapat memandu proses pencarian yang memiliki kemungkinan sukses paling besar, namun dengan kemungkinan mengorbankan kelengkapan (*completeness*). Untuk menerapkan pencarian heuristik diperlukan suatu fungsi heuristik. Fungsi heuristik adalah aturan-aturan yang digunakan untuk mendapatkan solusi yang diinginkan

*Heuristic search* ini dapat mengatasi beberapa kendala yang muncul pada metode-metode sebelumnya, seperti memecahkan masalah yang memiliki kompleksitas yang tinggi.

Pencarian heuristik terdiri dari :

1. Generate and test
2. Hill Climbing
3. Best-First-Search
4. Algoritma A\*
5. Simulated Annelaing

Dalam penelitian ini, algoritma yang digunakan akan menggunakan *heuristic Best-First Search*.

### 2.2.1 Alasan Menggunakan Metode Heuristic

Beberapa Alasan yang mendasari penulis memilih cara *heuristic* dalam penelitian ini antara lain :

1. Mudah dimengerti dan mudah diimplementasikan. Membantu dalam

- melatih orang sehingga kreatif dan dapat digunakan untuk masalah yang lain.
2. Menghemat waktu perumusan.
  3. Menghemat pemrograman dan kebutuhan penyimpanan pada komputer.
  4. Menghemat waktu pemrosesan komputer yang tak perlu (kecepatan).
  5. Seringkali menghasilkan berbagai solusi yang dapat diterima.

Pendekatan *logic heuristic* melibatkan hal-hal berikut:

1. Skema klasifikasi yang mengenalkan struktur ke dalam permasalahan.
2. Analisis karakteristik dari elemen-elemen masalah.
3. Aturan-aturan untuk seleksi elemen dari setiap kategori untuk mendapatkan strategi pencarian yang efisien.
4. Aturan-aturan untuk seleksi lebih lanjut, bila diperlukan.
5. Fungsi tujuan yang digunakan untuk mengecek kelayakan solusi pada setiap tahapan seleksi atau pencarian.

### 2.3 Algoritma *Best First Search*

Algoritma *Best-First Search* adalah algoritma pencarian yang membahas grafik dengan memperluas simpul paling menjanjikan dipilih sesuai dengan aturan yang ditentukan.

Algoritma (T.Sutojo : 2011):

1. Buat sebuah stack, inialisasi, sebagai node akar (node pertama).
2. Bila node pertama  $\neq$  goal, node dihapus dan diganti dengan anak-anaknya.
3. Selanjutnya, keseluruhan node yang ada di Stack di-sort Ascendingg berdasarkan fungsi heuristic yang digunakan.
4. Bila node pertama  $\neq$  GOAL, ulangi langkah point b.
5. Bila node pertama = GOAL, cari solusi dengan cara menelusuri jalur dari GOAL ke node akar.
6. Selesai.

#### a. Fungsi Algoritma *Best-First Search*

*Best-First Search* dalam bentuk yang paling umum adalah algoritma pencarian heuristik sederhana. "*Heuristic*" di sini mengacu pada aturan umum pemecahan masalah atau seperangkat aturan yang tidak menjamin solusi terbaik atau bahkan solusi

apapun, tetapi berfungsi sebagai panduan yang berguna untuk pemecahan masalah. *Best-First Search* adalah algoritma pencarian berbasis grafik, yang berarti bahwa ruang pencarian dapat direpresentasikan sebagai serangkaian *node* yang terhubung oleh jalan.

Nama "*Best-First*" mengacu pada metode mengeksplorasi *node* dengan yang terbaik "skor" pertama. Sebuah fungsi evaluasi digunakan untuk menetapkan skor untuk setiap *node* kandidat. Algoritma memelihara dua daftar, satu berisi daftar calon belum mengeksplorasi (*OPEN*), dan satu berisi daftar *node* dikunjungi (*CLOSED*). Karena semua belum dikunjungi penerus *node* dari setiap *node* yang dikunjungi termasuk dalam daftar *OPEN*, algoritma ini tidak dibatasi untuk hanya mengeksplorasi *node* penerus dari simpul yang terakhir dikunjungi. Dengan kata lain, algoritma selalu memilih yang terbaik dari semua *node* belum dikunjungi yang telah digambarkan, bukannya dibatasi untuk hanya sebagian kecil, seperti tetangga dekat. Strategi pencarian lainnya, seperti kedalaman dan keluasan-pertama-pertama, memiliki pembatasan ini. Keuntungan dari strategi ini adalah bahwa jika algoritma mencapai *node* buntu, ia akan terus berusaha *node* lain.

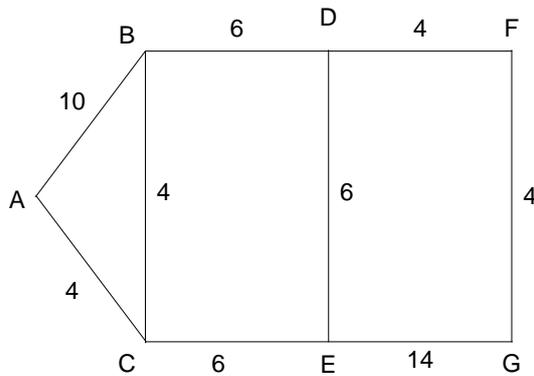
*Best-First Search* dalam bentuk yang paling dasar terdiri dari algoritma berikut: langkah pertama adalah untuk menentukan daftar *OPEN* dengan *node* tunggal, *node* awal. Langkah kedua adalah untuk memeriksa apakah atau tidak *BUKA* kosong. Jika kosong, maka algoritma mengembalikan kegagalan dan keluar. Langkah ketiga adalah untuk menghapus *node* dengan nilai terbaik, *n*, dari *OPEN* dan tempatkan di *CLOSED*. Langkah keempat "mengembang" *node* *n*, di mana ekspansi identifikasi *node* penerus *n*. Langkah kelima kemudian memeriksa setiap *node* penerus untuk melihat apakah atau tidak salah satu dari mereka adalah simpul tujuan. Jika penerus apapun adalah simpul tujuan, algoritma mengembalikan kesuksesan dan solusi, yang terdiri dari jalan ditelusuri mundur dari tujuan ke *node* awal. Jika tidak, algoritma melanjutkan ke langkah keenam. Untuk setiap simpul penggantinya, algoritma menerapkan fungsi evaluasi, *f*. Untuk itu, kemudian memeriksa untuk melihat apakah *node* telah baik *OPEN* atau *TERTUTUP*. Jika *node* belum baik, itu akan ditambahkan ke *OPEN*. Akhirnya, langkah ketujuh menetapkan struktur

perulangan dengan mengirimkan algoritma kembali ke langkah kedua. Loop ini hanya akan patah jika algoritma mengembalikan keberhasilan dalam lima langkah atau kegagalan pada langkah dua.

b. Fungsi Evaluasi

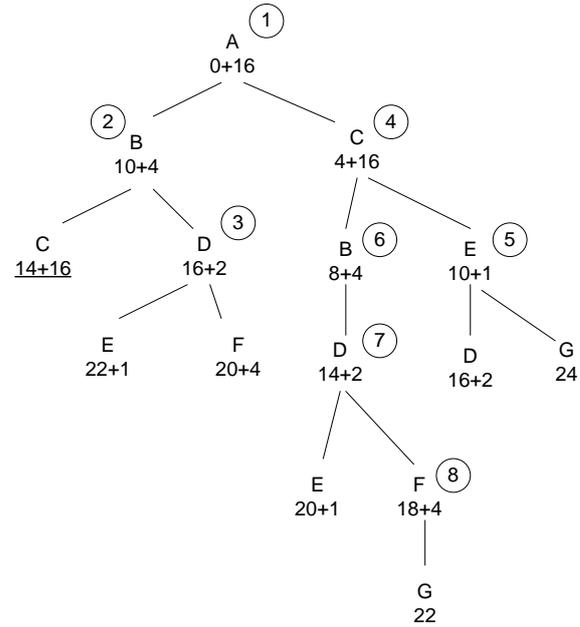
Fungsi evaluasi tertentu yang digunakan untuk menentukan nilai dari sebuah *node* tidak tepat didefinisikan dalam algoritma di atas, karena fungsi yang sebenarnya digunakan untuk penentuan *programmer*, dan dapat bervariasi tergantung pada kekhasan dari ruang pencarian. Sementara fungsi evaluasi dapat menentukan untuk sebagian besar efektivitas dan efisiensi pencarian, untuk tujuan memahami algoritma pencarian kita tidak perlu khawatir dengan kekhasan fungsi.

Sebagai contoh, dapat digunakan oleh agen musuh untuk menemukan lokasi pemain dalam dunia permainan. Dalam kasus tersebut, algoritma pencarian memperlakukan setiap ubin sebagai simpul, dengan tetangga ubin diblokir menjadi *node* penggantinya, dan simpul tujuan menjadi benteng tujuan (Koenig, 2004).



Gambar 1. Node Best-First Search

Jarak dalam	16	(Irrelevant)	From D	2
From B	4		From E	1
From C	16		From F	4



Gambar 2. Flow Best-First Search

Open	Closed
A16	
B14 C20	A16
D18	A16 B14
C20 E23 F24	A16 B14 D18
E11 B12 F24	A16 C20 D18
B12 F24 G24	A16 C20 E11
D16 F24 G24	A16 B12 C20 E11
F22 G24	A16 B12 C20 D16 E11
G22	A16 B12 C20 D16 E11F2

Fungsi *heuristic*, *h* adalah tidak monoton, namun.  $h(C) = 16$ ,  $h(B) = 4$ , dan biaya  $(C, B) = 4$ , sehingga  $h(C) - h(B) > biaya(C, B)$ . Oleh karena itu kami memiliki kejadian pertama simpul *D*, yang kemudian kembali ditempatkan oleh baik *D*. Estimasi optimis, jadi kita tahu bahwa jalan yang ditemukan adalah yang terbaik. Tetapi jika kita mengganti perkiraan jarak dari *F* ke *G* dengan 8 kita mendapatkan:

Open	Closed
A16	
B14 C20	A16
D18	A16 B14
C20 E23 F28	A16 B14 D18
E11 B12 F28	A16 C20 D18
B12 G24 F28	A16 C20 E11
D16 G24 F28	A16 B12 C20 E11
G24 F26	A16 B12 C20 D16 E11

Sekarang kita memiliki *G24* sebagai tujuan yang pas, jadi kami menghentikan pencarian. Kami telah menemukan solusi menurut untuk metode *A*, tetapi tidak yang terbaik. Perkiraan besar *F* menyembunyikan solusi terbaik.

**3. Metode Penelitian**

**3.1 Jenis Penelitian**

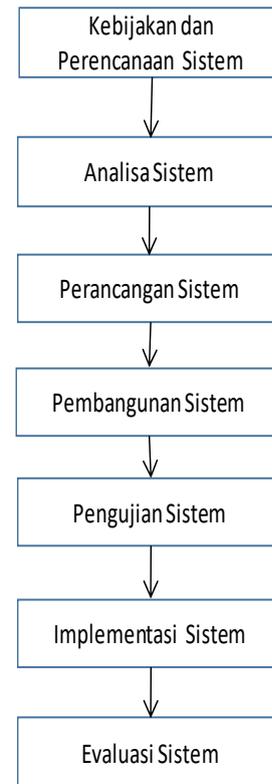
Jenis Penelitian yang penulis lakukan ini adalah penelitian tentang pembuatan sistem pencarian jalur pencarian shelter tsunami terdekat dengan menggunakan metode heuristik – best first search. Bertujuan untuk menghasilkan dan mengembangkan inovasi-inovasi yang baru dan berguna untuk membantu masyarakat jika terjadi bencana tsunami.

**3.2 Tempat Penelitian**

Penelitian ini dilakukan di kampus Institut Teknologi Padang, dengan cara menganalisa, merancang, dan membangun database.

**3.3 Kerangka Kerja**

Pada penelitian ini, penulis terlebih dahulu menjabarkan kerangka kerja, sesuai dengan Siklus Hidup Pengembangan Sistem atau *System Development Life Cycle (SDLC)*, yang merupakan langkah-langkah (pedoman) yang harus diikuti untuk mengembangkan, merancang sebuah sistem.



**Gambar 3. Kerangka Kerja Penelitian**

**3.4 Rancangan Context Diagram**

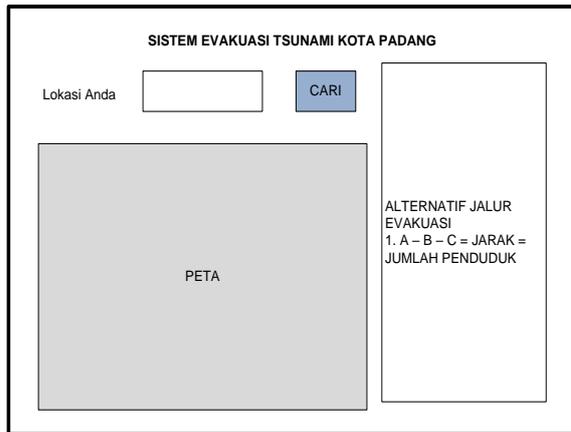


**Gambar 4. Context Diagram**

Pada gambar 4 diatas menggambarkan garis besar dari aplikasi pencarian jalur terdekat dengan melibatkan 1 entity saja, yaitu pengguna.

**3.5 Rancangan Input/Output**

Berikut rancangan dari tampilan form utama, dimana form ini nantinya akan dijalankan di browser smartphone atau langsung dari android masyarakat.



Gambar 5. Rancangan Halaman Utama

#### 4. Pembahasan dan Hasil

##### 4.1. Safe Area

Berdasarkan jumlah penduduk kota padang di high risk zone tahun 2016, kecamatan padang barat merupakan kecamatan yang penduduknya berada pada daerah tsunami paling banyak disusul oleh kecamatan padang utara.

Tabel 1. Jumlah Penduduk

No.	Kelurahan	Jumlah Penduduk (orang)	Jumlah Penduduk di Zona Tsunami	
			Orang	Persen
1	Bungus Teluk Kabung	23,858	0	0
2	Lubuk Kilangan	51,847	0	0
3	Kuranji	135,787	0	0
4	Pauh	64,864	0	0
5	Lubuk Begalung	113,217	1087	0.96
6	Padang Selatan	58,780	3868	6.58
7	Koto Tengah	174,567	18818	10.78
8	Nanggalo	59,137	16730	28.29
9	Padang Timur	78,789	30042	38.13
10	Padang Utara	70,051	61918	88.39
11	Padang Barat	45,781	45781	100

Dari hasil penelitian yang telah dilakukan oleh Siska Anggria dalam Pemodelan Optimasi Evakuasi Tsunami di Kota Padang, dimana Siska mengelompokkan kota padang dalam beberapa cluster dan setiap cluster terdapat safe area atau bisa disebut juga dengan shelter.

Tabel 2. Panjang Jalur (meter)

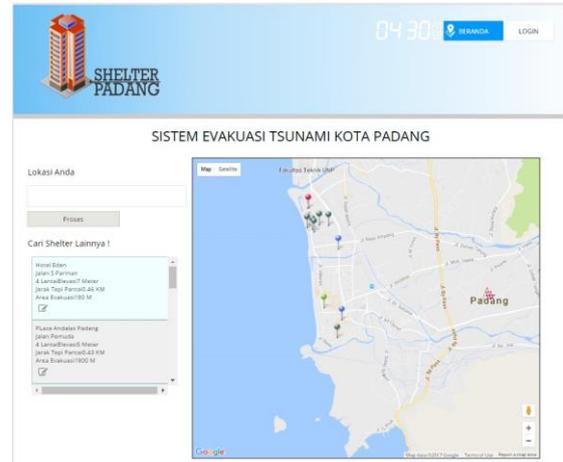
	1	2	3	4	5	6	7	8	9	10	11	12
Cluster 1	UBH 5000	DPRD 3000	Villa Hadis Permai 400	PDA 1000	ACC 400	Daihatsu 400	Yayasan Al Azhar 700					
Cluster 2	PT. Suka Fajar 400	Masjid Baitu 500	Suzuki 400	BPK 400	BPS Provinsi 600	Kantor Imigrasi 500	Kantor DPD 700					
Cluster 3	AKBP STIE 1000	SMP n 25 2500	Telkomsel 600	Stikes 500	PLN 500	SMAN 1 4000	SMPN 7 600	SMKN 5 1300	SDN 15 700			
Cluster 4	BAPPEDA 400	Masjid Raya 4000	Hotel Pangeran 1800	Ditjen Perbendah 600	Masjid Muhsinin 500	Erlangga 300	Hotel Ibis 3030	PU 1000	Univ. Tamsis 1000	Kejaksanaan Tinggi 1000	Dinas Sosial 800	Masjid Nurul 500
Cluster 5	Masjid Bustanul 500	Masjid Istiqlal 600	SDN 23-24 1500	Dinas Peternaka 600	BTN 500	Jasa Raharja 500	SMAN 2 1000	SMK Taruna 1 800	Bank Muamalat 600	Kantor Telkom 700		
Cluster 6	Polda 4000	Escape Building 4000	RS YOS Sudarso 2000	UNES 1500								
Cluster 7	ISUZU 800	Hotel Mercure 2000	Rusun 1500	Masjid Al Wustha 800	SDN 03 1500	Damar Plaza 800	Yamaha 700					
Cluster 8	Masjid Nurul Ulya 600	Bank Nagari 500	Plaza Andalas 1500	Pasar Inpres 1800	Masjid Taqwa 1000	Hotel Rocky 1500						
Cluster 9	Hotel Axana 2000	Hotel Mariani 2000	BRI 600	Hotel Pangeran 1500	SMP Maria 600	SD Agnes 600	Hotel Grand Inna 3000					
Cluster 10	Teebox 1500	BAF 600	Hotel HW 1500	Hotel Plan 1500	SMPN 4 1000	Masjid Nurul Iman 1500	Hotel Grand Zuri 2000					

Selain itu, juga menghasilkan optimasi waktu minimum untuk menuju safe area dengan formula (1).

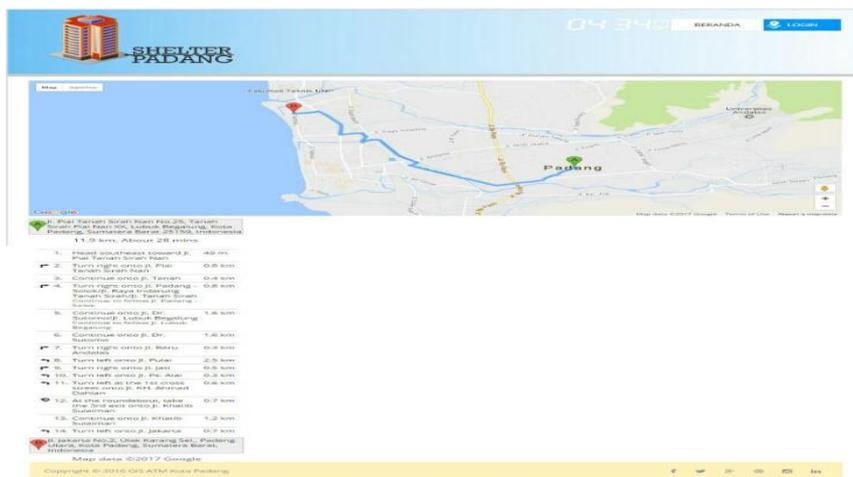
$$T_{ijk} = \frac{\left( \frac{N_{ik}}{L_{ijk}} * ijk - 1 \right) * d_{ijk} + L_{ijk}}{v_{ijk}} \dots (1)$$

#### 4.2 Halaman Utama

Halaman utama merupakan halaman awal untuk mengakses sistem. Pada sistem ini ada 2 cara dalam pmenginisialisasi lokasi awal pengguna, yakni secara otomatis maupun secara manual dengan dientrikan langsung.



Gambar 6. Tampilan Halaman Utama



Gambar 7. Tampilan Jalur Evakuasi pada Sistem.

#### 5. Kesimpulan

Pada paper ini menggunakan formulasi waktu optimum yang diperoleh berdasarkan dari kepadatan penduduk yang akan melakukan evakuasi, serta kepanjangan lajur yang dilalui serta dikombinasikan menggunakan metode heuristik best first search dalam penentuan jalur terpendek dengan tetap memperhatikan indikator lain selain panjang jalur yang ditempuh. Dan perhitungan numerik tersebut diaplikasikan ke sistem berbasis web. Dimana pengguna nanti bisa langsung mengakses sistem.

#### DAFTAR PUSTAKA

Afrianto, Irawan dan Euis Widiani Jamilah. Oktober 2012. *Penyelesaian Masalah Minimum Spanning Tree (Mst) Menggunakan Ant Colony System (ACS)*. Jurnal Ilmiah Komputer dan Informatika (KOMPUTA) Volume. I

Nomor. 2, Bulan Oktober 2012. Hal 35-40.

Alamsyah. 2010. *Pemanfaatan Metode Heuristik dalam Pencarian Minimum Spanning Tree Dengan Algoritma Semut*. MEKTEK Tahun XII NO. 3, September 2010.

Alamsyah. 2010. *Pemanfaatan Metode Heuristik pada Pencarian Jalur Terpendek dengan Algoritma Genetika*. Jurnal SMARTek, Vol. 8 No. 4. Nopember 2010. Hal 307 – 316.

Anggria ,Siska,dkk. 2016. *Pemodelan Optimasi Evakuasi Tsunami di Kota Padang*. Prosiding Seminar ACE Fakultas Teknik Sipil Universitas Andalas.

A.K.Balirsingh. 2011. *Robust FACTS Controller Design Employing Modern Heuristic Optimization Techniques*. World Academy of Science, Engineering and Technology 56 2011.

- Dekdouk, Abdelkader. 2008. *Intelligent Mobile Search Oriented to Global eCommerce*. World Academy of Science, Engineering and Technology 16 2008.
- Dewi, Shanty. 2010. *Perancangan Aplikasi Informasi Pencarian Shelter Transit Transjakarta dengan Metode Breadth First Search*. Universitas Gunadarma.
- E-Wolf Community. 2011. *Buku Wajib Programmer - Syntax*. Graha Ilmu. Yogyakarta.
- Faizah, Ifatul. 2010. *Rancang Bangun Perangkat Lunak Penentuan Rute Perjalanan Wisata di Malang Menggunakan Algoritma Dijkstra*.
- Fauzan. *Evaluasi Existing Building dan Pembuatan Peta Evakuasi Vertikal Terhadap Tsunami Di Kota Padang*. Universitas Andalas.
- Hardianto. 2013. *Implementasi Algoritma Heuristik untuk Optimasi Rute Terpendek*. Jurnal Teknologi Informasi dan Komunikasi, Volume 4 No. 2 Agustus 2013. Hal 79-88.
- Kadir, Abdul. 2008. *Tuntunan Praktis Belajar Database Menggunakan MySQL*. Andi Yogyakarta :Yogyakarta.
- Kusumadewi, Sri. 2010. *Artificial Intelligence*. Yogyakarta: Graha Ilmu.
- Na'am, Jufriadif. 2007. *Komunikasi Data dan Jaringan Komputer*. Padang
- Oktiari, Dian. 2010. *Model Geospasial Potensi Kerentanan Tsunami Kota Padang*. Jurnal Meteorologi Dan Geofisika Volume 11 Nomor 2 Tahun 2010 : 136 – 141.
- Susanta, Agus, dan Rusdi. 2006. *Model Pendekatan Heuristik pada Pemecahan Masalah dalam Pembelajaran Matematika di Sekolah Dasar*. Jurnal pendidikan dan pembelajaran Vol 4 No. 1, Universitas Lampung. Hal 13 -24.
- T. Sutojo, dkk. 2011. *Kecerdasan Buatan*. Semarang : Andi.
- Taufikurrahman , Andre Nugroho, dan Ahmad Rusdiansyah. 2014. *Perancangan Algoritma Heuristik Untuk Penyelesaian Permasalahan Swap-Body Vehicle Routing Problem*. Institut Teknologi Sepuluh Nopember. Hal 1-4.