JURNAL

APLIKASI FUNDRAISER DALAM MENENTUKAN RUTE TERDEKAT PENGAMBILAN DONASI DONATUR YAYASAN NURUL HAYAT CABANG KEDIRI DENGAN ALGORITMA A* BERBASIS ANDROID

OLEH :

IMRON ROSADI

NPM: 12.1.03.02.0323

Dibimbing oleh :

1. **Dr. Suryo Widodo, M.Pd**
2. **Risky Aswi Ramadhani, M.Kom**

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS NUSANTARA PGRI KEDIRI

2017

SURATPERNYATAAN
ARTIKEL SKRIPSI TAHUN2017

Yang bertandatangan dibawah ini:

NamaLengkap : IMRON ROSADI
NPM : 12.1.03.02.0323
Telepun/HP : 085735167122
AlamatSurel (Email) : imronr55@gmail.com
JudulArtikel : APLIKASI FUNDRISER DALAM MENENTUKAN RUTE TERDEKAT PENGAMBILAN DONASI DONATUR YAYASAN NURUL HAYAT CABANG KEDIRI DENGAN ALGORITMA A STAR BERBASIS ANDROID
Fakultas – Program Studi : TEKNIK – TEKNIK INFORMATIKA
NamaPerguruanTinggi : UNIVERSITAS NUSANTARA PGRI
AlamatPerguruanTinggi :Jl. K.H. Ahmad Dahlan No. 76, Mojoroto,Kota Kediri

Dengan ini menyatakan bahwa:

a. artikel yang saya tulis merupakan karya saya pribadi (bersama tim penulis**)** dan bebas plagiarisme.
b. artikel telah ditelit idan disetujui untuk diterbitkan oleh Dosen Pembimbing I dan II.

Demikian surat pernyataan ini saya buat dengan sesungguhnya. Apabila dikemudian hari ditemukan ketidaksesuaian data dengan pernyataan ini dan atau ada tuntutan dari pihak lain, saya bersedia bertanggung jawab dan diproses sesuai dengan ketentuan yang berlaku.

Mengetahui		Kediri, 30 Januari 2017
Pembimbing I Dr. Suryo Widodo, M.Pd NIDN. 0002026403	Pembimbing II Risky Aswi Ramadhani, M.Kom NIDN. 0708049001	Penulis, Imron Rosadi NPM. 12.1.03.02.0323

APLIKASI FUNDRAISER DALAM MENENTUKAN RUTE TERDEKAT PENGAMBILAN DONASI DONATUR YAYASAN NURUL HAYAT CABANG KEDIRI DENGAN ALGORITMA A* BERBASIS ANDROID

Imron Rosadi
12.1.03.02.0323
Teknik–Teknik Informatika
imronr55@gmail.com
Dr. Suryo Widodo, M.Pd, Risky Aswi Ramadhani, M.Kom
UNIVERSITAS NUSANTARA PGRI KEDIRI

ABSTRAK

Yayasan Nurul Hayat adalah lembaga yang bergerak dibidang sosial, pendidikan, dakwah dan kesehatan. Berkantor pusat di kota Surabaya dan mempunyai cabang yang menyebar di seluruh Indonesia, salah satunya di kota Kediri. Yayasan Nurul Hayat mempunyai divisi yang dinamakan divisi ZIS (Zakat Infaq Sedekah). *Fundraiser* termasuk dalam divisi ZIS, bertugas untuk mengambil donasi para donatur Yayasan Nurul Hayat setiap bulannya. Dalam setiap pengambilan donasi, *fundraiser* hanya menggunakan data manual atau kwitansi untuk melihat alamat donatur dan menentukan titik-titik wilayah donatur. Sehingga perlu dibuatkan sebuah aplikasi untuk memudahkan *fundraiser* dalam menentukan rute-rute terdekat dalam pengambilan donasi donatur. Permasalahan penelitian ini adalah bagaimana merancang aplikasi *fundriser* dalam menentukan rute terdekat pengambilan donasi donatur menggunakan algoritma *a star*. Salah satu metode yang dapat digunakan adalah dengan menggunakan algoritma *a star*. Metode ini dipilih karena algoritma *a star* meminimumkan biaya lintasan yang ditempuh. Hasil yang diperoleh dari penelitian ini adalah penentuan rute terdekat pengambilan donasi donatur Yayasan Nurul Hayat cabang Kediri oleh *fundriser* dengan berbasis *android*. Dengan bantuan aplikasi tersebut, proses penentuan rute terdekat pengambilan donasi donatur yang dilakukan *fundriser* bisa secara efektif dan efisien dalam mengoptimalkan waktu pengambilan donasi donatur.

Kata Kunci : *Fundriser*, Algoritma *A Star*, *Android*

I. LATAR BELAKANG

Yayasan Nurul Hayat adalah lembaga yang bergerak dibidang sosial, pendidikan, dakwah dan kesehatan. Berkantor pusat di kota Surabaya dan mempunyai cabang yang menyebar di seluruh Indonesia, salah satunya di kota Kediri. Yayasan Nurul Hayat mempunyai divisi yang dinamakan divisi ZIS (Zakat Infaq Sedekah).

Fundraiser termasuk dalam divisi ZIS, bertugas untuk mengambil donasi para donatur Yayasan Nurul Hayat setiap bulannya. Dalam setiap pengambilan donasi, *fundraiser* hanya menggunakan data manual atau kwitansi untuk melihat alamat donatur dan menentukan titik-titik wilayah donatur. cabang kota Kediri. Kurang

lebih ada lima ratus titik lokasi di dalam maupun di luar kota yang dikunjungi untuk mengambil donasi. Lokasi yang dikunjungi meliputi karesidenan Kediri. Data ini adalah data di bulan November 2016, setiap bulan data akan bertambah dengan adanya penambahan donatur baru.

Selama satu bulan semua titik harus terkunjungi karena mempengaruhi target perolehan. *Fundriser* harus bisa mengoptimalkan waktu yang ada, agar semua titik bisa terkunjungi. Tetapi semua ini dilakukan secara manual dalam mencari alamat-alamat donatur yang baru bergabung, dan hal ini bisa menyita waktu *fundriser* tanpa adanya bantuan sebuah sistem. Dalam hal ini *fundraiser* kesulitan dalam menentukan rute yang akan ditempuh karena mempunyai lokasi yang berbeda-beda. Sehingga perlu dibuatkan sebuah aplikasi untuk memudahkan *fundraiser* dalam menentukan rute-rute terdekat dalam pengambilan donasi donatur.

Penelitian ini menggunakan acuan jurnal dengan algoritma yang sejenis telah dilakukan oleh Ayu, dkk (2012) dalam pencarian mesin ATM terdekat di Palembang berbasis *android*. Penilitian serupa juga dilakukan oleh oleh Esther Irawat, dkk (2012) dalam Pencarian Lokasi Fasilitas Umum Terdekat Dilengkapi Dengan Rute Kendaraan Umum Lyn . Kedua jurnal ini sebagai acuan untuk membuat aplikasi dengan menggunakan algoritma *a star*. Jurnal acuan ini mempunyai perbedaan dalam penelitian penulis yaitu terletak pada bagian perancangan arsitektur sistem berjalannya aplikasi

Penggunaan algoritma *a star* dalam menentukan rute terdekat akan memberikan solusi terbaik dalam waktu yang optimal dengan meminimumkan total biaya lintasan. Karena algoritma ini merupakan perbaikan dari metode *best-first search* dengan memodifikasi fungsi heuristiknya. Dimana pencarian dilakukan dengan cara melebar ke setiap node pada level yang sama, dan nantinya menentukan rute terbaik dari titik awal sampai titik tujuan. Dan fungsi heuristiknya digunakan sebagai optimasi dalam menentukan node tujuan yang akan dipilih.

Dengan adanya aplikasi untuk menentukan rute terdekat dalam pengambilan donasi donatur ini akan memudahkan *fudraiser* dalam menentukan rute untuk mengoptimalkan waktu yang ada serta aktifitas pekerjaannya di Yayasan Nurul Hayat akan menjadi lebih optimal.

Yayasan Nurul Hayat adalah lembaga yang bergerak dibidang sosial, pendidikan, dakwah dan kesehatan. Berkantor pusat di kota Surabaya dan mempunyai cabang yang menyebar di seluruh Indonesia, salah satunya di kota Kediri. Yayasan Nurul Hayat mempunyai divisi yang dinamakan divisi ZIS (Zakat Infaq Sedekah).

Fundraiser termasuk dalam divisi ZIS, bertugas untuk mengambil donasi para donatur Yayasan Nurul Hayat setiap bulannya. Dalam setiap pengambilan donasi, *fundraiser* hanya menggunakan data manual atau kwitansi untuk melihat alamat donatur dan menentukan titik-titik wilayah donatur. cabang kota Kediri. Kurang lebih ada lima ratus titik lokasi di dalam maupun di luar kota yang dikunjungi untuk mengambil donasi. Lokasi yang dikunjungi meliputi karesidenan Kediri. Data ini adalah data di bulan November 2016, setiap bulan data akan bertambah dengan adanya penambahan donatur baru.

Selama satu bulan semua titik harus terkunjungi karena mempengaruhi target perolehan. *Fundriser* harus bisa mengoptimalkan waktu yang ada, agar semua titik bisa terkunjungi. Tetapi semua ini dilakukan secara manual dalam mencari alamat-alamat donatur yang baru bergabung, dan hal ini bisa menyita waktu *fundriser* tanpa adanya bantuan sebuah sistem. Dalam hal ini *fundraiser* kesulitan dalam menentukan rute yang akan ditempuh karena mempunyai lokasi yang berbeda-beda. Sehingga perlu dibuatkan sebuah aplikasi untuk memudahkan *fundraiser* dalam menentukan rute-rute terdekat dalam pengambilan donasi donatur.

Penelitian ini menggunakan acuan jurnal dengan algoritma yang sejenis telah dilakukan oleh Ayu, dkk (2012) dalam pencarian mesin ATM terdekat di Palembang berbasis *android*. Penilitian serupa juga dilakukan oleh oleh Esther Irawat, dkk (2012) dalam Pencarian Lokasi Fasilitas Umum Terdekat Dilengkapi Dengan Rute Kendaraan Umum Lyn . Kedua jurnal ini sebagai acuan untuk membuat aplikasi dengan menggunakan algoritma *a star*. Jurnal acuan ini mempunyai perbedaan dalam penelitian penulis yaitu terletak pada bagian perancangan arsitektur sistem berjalannya aplikasi

Penggunaan algoritma *a star* dalam menentukan rute terdekat akan memberikan solusi terbaik dalam waktu yang optimal dengan meminimumkan total biaya lintasan. Karena algoritma ini merupakan perbaikan dari metode

best-first search dengan memodifikasi fungsi heuristiknya. Dimana pencarian dilakukan dengan cara melebar ke setiap node pada level yang sama, dan nantinya menentukan rute terbaik dari titik awal sampai titik tujuan. Dan fungsi heuristiknya digunakan sebagai optimasi dalam menentukan node tujuan yang akan dipilih.

Dengan adanya aplikasi untuk menentukan rute terdekat dalam pengambilan donasi donatur ini akan memudahkan *fudraiser* dalam menentukan rute untuk mengoptimalkan waktu yang ada serta aktifitas pekerjaannya di Yayasan Nurul Hayat akan menjadi lebih optimal.

II. METODE

1. Fungsi Heuristik

Dalam metode pencarian *heuristik*, digunakan suatu metode yang digunakan untuk mengevaluasi keadaan-keadaan masalah individual dan menentukan seberapa jauh hal tersebut dapat digunakan dalam mendapatkan solusi yang diinginkan. Suatu fungsi *heuristik* dapat dikatakan baik apabila dapat memberikan biaya perkiraan yang mendekati biaya sebenarnya. Semakin mendekati biaya sebenarnya fungsi *heuristik* tersebut semakin baik.

Dalam melakukan pencarian rute terpendek, penelitian ini menggunakan *Euclidean Distance* yang diperoleh berdasarkan jarak langsung bebas hambatan seperti untuk mendapatkan nilai dari panjang garis diagonal pada segitiga. Tetapi sebelum mendapatkan hasil dari kedua titik harus dipresentasikan dalam dua koordinat dimensi (x,y), sehingga dapat dihitung dengan rumus :

$H_{a\text{-}b} = \sqrt{(x1 - x2)^2 + (y1 - y2)^2}$ ……..(2.0)

Dimana :

$H_{a\text{-}b}$: estimasi jarak a ke b

x1 : koordinat x titik b

y1 : koordinat y titik b

x2 : koordinat x titik a

y2 : koordinat x titik a

Perhitungan *euclidean distance* harus ditentukan terlebih dahulu untuk bisa menentukan biaya perkiraan yang akan dijumlahkan dengan biaya sebenarnya di dalam menentukan rute terpendek di dalam algoritma A**(Angga Ari Wijaya, 2013).*

2. **Pencarian Terbaik Pertama (*Best-First Search*)**

Best-First Search merupakan sebuah metode yang membangkitkan simpul dari simpul sebelumnya. *Best-First Search* memiliki simpul baru yang memiliki biaya terkecil diantara semua *leaf nodes* (simpul-simpul pada level terdalam) yang pernah dibangkitkan.

Ada beberapa istilah yang sering digunakan pada metode *best-first search*, yaitu :

- *Start node* adalah sebuah terminology untuk posisi awal sebuah pencarian.
- *Current node* adalah simpul yang sedang dijalankan dalam algoritma pencarian jalan terpendek.
- *Suksesor* adalah simpul-simpul yang akan diperiksa setelah *current node*
- Simpul *(node)* merupak representasi dari area pencarian.
- *Open list* adalah temapat menyimpan data simpul yang mungkin diakses dari *starting node* maupun yang sedang dijalankan.
- *Closed list* adalah tempat menyimpan data simpul yang merupakan bagian dari jalur terpendek yang telah berhasil didapatkan.
- *Goal node* yaitu simpul tujuan
- *Parent* adalah *current node* dari *suksesor*

3. ***Uniform Cost Search***

Uniform Cost Search merupakan metode yang menggunakan harga terendah yang dihitung berdasarkan harga dari *node* asal menuju ke *node* tersebut atau bisa dilambangkan *g(n)*. *Uniform Cost Search* melakukan ekspansi *node* dengan nilai yang paling kecil. Hal ini bisa dilakukan dengan membuat antrian pada suksesor yang ada berdasar kepada nilai *path*-nya.

4. ***Greedy Best First Search***

Greedy Best First Search mengambil keputusan yang dianggap terbaik hanya untuk saat itu yang diharapkan bisa memberikan solusi terbaik secara keseluruhan. Oleh sebab itu, pada setiap langkah harus diberi keputusan yang terbaik dalam menentukan pilihan.

Pada *Greedy First Search* fungsi evaluasi tidak tergantung pada *cost* sebenarnya, tetapi hanya tergantung pada fungsi

heuristik h(n) itu sendiri yang mengestimasikan arah yang benar, sehingga pencarian jalur dapat berlangusng dengan cepat. Secara sistematis fungsi evaluasi pada *Greedy Search* yaitu :

f(n) = h(n)

Dimana :

f(n) : fungsi evaluasi

h(n) : fungsi *heuristik*

5. Algoritma *A Star*

Algoritma *a star* merupakan metode *best-first search* dengan memodifikasi fungsi heuristiknya. *A star* akan meminimumkan total biaya lintasan. Pada kondisi yang tepat, *a star* akan memberikan solusi yang terbaik dalam waktu yang optimal (*Sri Kusuma Dewi, AI, 2003).*

Algoritma *a star* menggabungkan *uniform cost search* dan *greedy best first search* dengan memperhitungkan biaya dari biaya sebenarnya ditambah dengan biaya perkiraan. Dalam menentukan rute jarak terdekat dengan menggabungkan *uniform cost search*dan *greedy best first search* maka rumus yang digunakan yaitu :

f(n) = g(n) + h(n)

Dimana :

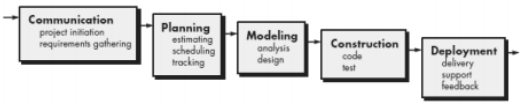
f(n) : perkiraan total *cost* terendah atau fungsi evaluasi

g(n) : *cost* yang telah didapat dari *node* awal hingga *node* yang dituju

h(n) : nilai estimasi dari suatu *node* tempat berada hingga *node goal* atau fungsi *heuristik*

III. METODE PENELITIAN

Metode penelitian yang digunakan adalah metode pengembang, yaitu metode *waterfall* menurut referensi Pressman yang artinya model klasik yang bersifat sistematis, berurutan dalam membangun *software*. Fase-fase dalam model *waterfall* menurut referensi Pressman:



Gambar 1 *Waterfall* Pressman

1. *Communication*

Langkah ini merupakan analisis terhadap kebutuhan *software*, dan tahap untuk mengadakan pengumpulan data dengan melakukan pertemuan dengan *customer*, maupun pengumpulan data-data tambahan baik yang ada di jurnal, artikel, maupun dari internet.

2. *Planning*

   Proses *planning* merupakan lanjutan dari proses *communication (analysis requirement).* Tahapan ini akan menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan *software*, termasuk rencana yang dilakukan.

3. *Modeling*

   Proses *modeling* ini akan menerjemahkan syarat kebutuhan ke sebuah perancangan *software* yang dapat diperkirakan sebulum dibuat *coding*. Proses ini berfokus pada rancangan struktur data, arsitektur *software*, representasi *interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requiment.*

4. *Construction*

   Construction merupakan proses membuat kode. *Coding* atau pengkodean merupakan penerjemah desain dalam bahasa yang dikenaali oleh komputer. *Programmer* akan menerjemahkan transaksi yang diminta oleh *user*. Tahapan inilah yang merupakan tahapan secara nyata dalam mengerjakan suatu *software*, artinya penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

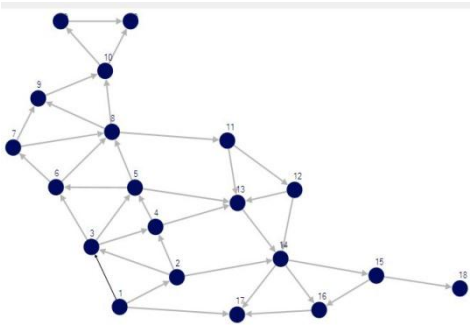
5. *Deployment*

   Tahapan ini bisa dikatakan final dalam pembuatan sebuah *software* atau sistem. Setelah melakukan analisis, desain dan pengkodean maka sistem yang sudah jadi akan digunakan oleh *user*. Kemudian *software* yang telah dibuat harus dilakukan pemeliharaan secara berkala.

IV. HASIL DAN KESIMPULAN

1. HASIL

Dengan menggunakan graf pada gambar di bawah ini maka akan di pahami alur kerja algoritma a star. Seingga perhitungan sesuai dengan kasus real yang ada



Gambar 2 *Graf* Antar *Node*

Keterangan dari graf pada Gambar 4.6 yaitu sebagai berikut :

a. *Node* dimulai dari angka satu sampai duapuluh yang setiap *node* membunyai hubungan dengan *node* yang lain
b. Untuk melihat jarak anar *node* bisa melihat pada Tabel 4.8
c. Cara membacanya yaitu harus dimulai dari *node* awal yaitu satu berhubungan dengan *node* satu, tiga dan tujuh belas. *Node* dua berhubungan dengan *node* tiga, empat dan empat belas. Begitu seterusnya hingga node yang terakir yaitu dua puluh
d. Pada jarak anar node ini berlaku bolak-balik misalnya dari *node* satu ke dua atau *node* dua ke satu mempunai nilai jarak yang sama.
e. Dalam hal ini akan di coba menentukan rute dari node 1 menuju ke node 5 dengan menggunakan data yang sudah ada

Node	Koordinat	
	Lattitude (x)	Longitude (y)
1 (titik awal)	-7.820886	112.00025
2	-7.819548	111.998883
3	-7.814089	112.005423
4	-7.810956	111.998778
5 (titik akhir)	-7.809337	112.001358

Setelah penentuan sumbu x (*lattitude*) atau disebut garis lintang dan y (*longitude*) atau disebut garis bujur, nilai heuristiknya h(n) harus ditentukan menggunakan rumus

$$H_{a\text{-}b} = \sqrt{(x - x1)^2 + (y - y2)^2}$$

Dimana :

x : koodinat *latitude* titik 1

x_1 : koodinat *latitude* titik 2

y :koodinat *longitude* titik 1

y_2: koodinat *longitude* titik 2

Berikut ini tabel perhitungannya:

Jarak	**Perhitungan**	**Hasil h(n)**
1 ke 2	$\sqrt{(-7.820886 - 7.819548)^2 + (112.00025 - 111.998883)^2}$ * 111.319	0.212935 km
1 ke 3	$\sqrt{(-7.820886 - 7.814089)^2 + (112.00025 - 112.005423)^2}$ * 111.319	0.950844 km
2 ke 3	$\sqrt{(-7.819548 - 7.814089)^2 + (111.998883 - 112.005423)^2}$ * 111.319	0.94832 km
2 ke 4	$\sqrt{(-7.819548 - 7.810956)^2 + (111.998883 - 111.998778)^2}$ * 111.319	0.956524 km

3 ke 4	$\sqrt{(-7.814089 - 7.810956)^2 + (112.005423 - 111.998778)^2} * 111.319$	0.817 81 km
3 ke 5	$(-7.814089 - 7.809337)^2 + (112.005423 - 112.001358)^2 * 111.319$	0.696 129 km
4 ke 5	$\sqrt{(-7.810956 - 7.809337)^2 + (111.998778 - 112.001358)^2} * 111.319$	0.339 068 km

Dalam perhitungan algoritma *a star* dibutuhkan jarak antar dua titik yang di simbolkan g(n), berikut tabel perhitungannya:

Jarak Antar Dua Titik	Nilai Lintasan g(n)
1 – 2	0.3 km
1 – 3	1.4 km
2 – 3	1.4 km
2 – 4	1.3 km
3 – 4	1 km
3 – 5	0.9 km
4 – 5	0.5 km.

Untuk nilai dari h(n) dan g(n) sudah dihitung, maka selanjutnya mencari nilai cost terendah yang biasa di lambangkan f(n) pada algoritma A* dengan rumus f (n) = g(n) + h (n)

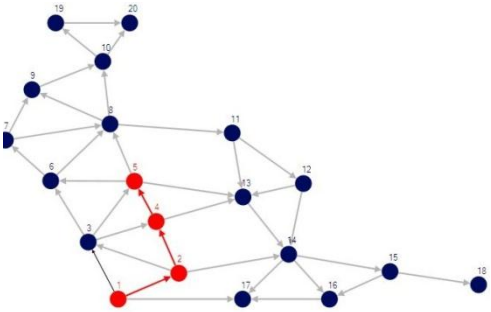
1. Node 1 dibangkitkan suksesornya yaitu 2,3 dan 17, angka 17 tidak masuk karena tidak termasuk dalam hitungan.

   F (1-2) = 0.3 + 0.212935 = 0.512935 km

   F (1-3) = 1.4 + 0.95084 = 2.35084 km
2. Dengan melihat hasilnya nilai yang lebih kecil yaitu titik 1 ke 2 sehinngga *node* 1 dibangkitkan suksesornya yaitu 3,4 dan 14. Angka 14 tidak masuk karena tidak termasuk dalam hitungan.

   F (2-3) = 1.4 + 0.94832 = 2.34832 km

   F (2-4) = 1.3 + 0.956524 = 2.256524 km
3. Dengan melihat nilai tersebut maka sudah bisa dipastikan untuk menempuh rute dari titik 1 ke 5 yaitu melalui jalur 1 – 2 – 4 – 5 dengan graf sebagai berikut :



Pada simulasi perhitungan di atas dapat diketahui dalam menentukan rute terdekat untuk pengambilan donasi donatur oleh *fundriser* bisa lebih optimal dan pada sistem ini lokasi GPS (*Global Positioning System*) harus berada pada lokasi atau kecamatan yang mau dituju, sehingga sistem bisa berjalan dengan baik.

2. KESIMPULAN

Berikut ini adalah simpulan dari penelitian antara lain:

a. Penelitian yang dilakukan menghasilkan rancangan sistem untuk aplikasi *fundriser* dalam menentukan rute terdekat pengambilan donasi donatur Yayasan Nurul Hayat cabang Kediri mengunakan algoritma *a star* berbasis android. Rancangan yang dihasilkan melewati tahap yang dimulai dari analisis sistem lama yang digunakan *fundriser* di Yasasan Nurul Hayat cabang Kediri. Rancangan sistem yang dihasilkan berupa *flowcart, use case, actiity diagram, sequence diagram*, perancangan database, dan desain *interface*. Karakteristik perancangan sesuai dengan karakteristik algoritma *a star* dan layak untuk diimplementasikan.

b. Dari rancangan yang dihasilkan, peneliti membuat program aplikasi *fundriser* dalam menentukan rute terdekat pengambilan donasi donatur Yayasan Nurul Hayat cabang Kediri menggunakan algoritma *a star* berbasis *android*. Aplikasi yang dihasilkan menggunakan perpaduan bahasa pemrograman yaitu *java* dan *PHP*. Bahasa pemrogaman *java* sebagai desain *interface* dari aplikasi sedangkan bahasa pemrograman PHP digunakan untuk mengimplementasikan perhitungan algoritma *a star*. Hasil akhir dari aplikasi ini menghasilkan penentuan rute terdekat pengambilan donasi donatur oleh fundriser Yayasan Nurul Hayat cabang Kediri yang sudah dihitung menggunakan algoritma *a star* dari aplikasi ini menampilkan *output* berupa *google maps*. Alur dari perhitungan jalannya algoritma untuk aplikasi yaitu pertama harus menentukan titik koordinat *lattitude* dan *longitude* alamat donatur yang nantinya di *input* ke dalam database yang selanjutnya akan dihitung menggunakan agloritma *a star* dan memanfaatkan teknologi *google maps* sebagai *output* dari aplikasi. Hasil perhitungan algoritma *a star* pada aplikasi sesuai dengan simulasi pada perhitungan algoritma *a star*,

sehingga aplikasi ini layak digunakan sebagai penentuan rute terdekat pengambilan donasi donatur Yayasan Nurul Hayat cabang Kediri

V. DAFTAR PUSTAKA

[1] Angga Ari Wijaya, 2013. *Integrasi Algoritma A* Untuk Pengembangan Pathfinding dan Perilaku Karakter Game*. Program Studi Teknik Informatika Universitas Jember.

[2] Anggari Ayu P, Yoga Saputra, Abdul Rahman, 2012. *Penerapan Algoritma A Star Dalam Pencarian Mesin ATM Terdekat di Palembang Berbasis Android.* Program Studi Teknik Informatika STIMK GI MDP

[3] Sri Kusuma Dewi 2003, *Artifical Intelegent*, Yogyakarta: Andi

[4] Zakaria, 2006, Teori Graf, diakses 04 November 2016 dari https://ienx.files.wordpress.com/2007/12/bab-ii-landasan-teori.pdf