

PENENTU LETAK ODP DENGAN ALGORITMA K-MEDOID CLUSTERING BERBASIS WEB

Arif Nur Pratomo^{1*)}, Sukiswo², dan Sudjadi³

¹²³Departemen Teknik Elektro, Fakultas Teknik, Universitas Diponegoro, Semarang, Indonesia

^{*)}E-mail: rifpratomo@gmail.com

Abstrak

Jaringan akses Fiber To The Home (FTTH) merupakan infrastruktur yang menghubungkan pusat penyedia layanan telekomunikasi dengan rumah pengguna menggunakan serat optik. Penempatan Optical Distribution Point (ODP) dapat menentukan kualitas layanan jaringan FTTH, sehingga penting untuk menempatkan ODP pada lokasi yang tepat sesuai dengan jumlah pengguna layanan. Dalam penelitian ini, penulis mengembangkan sebuah sistem yang menggunakan algoritma K-Medoid Clustering untuk menentukan letak ODP yang dapat dimanfaatkan oleh perancang jaringan FTTH pada perumahan Punsae di Kab. Semarang. Sistem ini memanfaatkan data koordinat rumah sebagai input untuk menghitung jarak terdekat antar rumah dan mengelompokkannya ke dalam kluster-kluster ODP. Program yang dirancang mampu menghimpun data koordinat dan menghasilkan titik rekomendasi ODP dan kluster titik-titik rumah. Rancangan menghasilkan klustering dengan rata-rata error sebesar 15% dengan pengujian variasi data, dan pada pengujian pada perumahan Punsae didapatkan 16 rekomendasi varian penempatan ODP.

Kata kunci : FTTH, Optical Distribution Point, Perancangan Web, Clustering K-Medoid

Abstract

Fiber To The Home (FTTH) network is an infrastructure that connects the telecommunication service provider center with the user's home using optical fiber. The placement of Optical Distribution Point (ODP) can determine the quality of FTTH network services, so it is important to place ODP in the right location according to the number of service users. In this research, the author develops a system that uses the K-Medoid Clustering algorithm to determine the location of ODP ODP which can be useful for FTTH network designer in Punsae housing in Semarang Regency. This system utilizes house coordinate data as input to calculate the closest distance between houses and group them into ODP clusters. The designed program is able to collect coordinate data and produce ODP recommendation points and clusters of house points. The design produces *clustering* with an average error of 15% by testing data variations, and in testing the Punsae housing, 16 recommendations were obtained.

Keywords: FTTH, Optical Distribution Point, Web development, K-Medoid Clustering

1. Pendahuluan

Internet telah menjadi kebutuhan utama dalam kehidupan sehari-hari di Indonesia. Dalam berbagai aspek kehidupan, seperti bersosialisasi, berbelanja, dan bahkan hingga konsultasi kesehatan, internet sangat diperlukan. Sehingga permintaan masyarakat terhadap internet terus meningkat setiap tahunnya, yang bermuara ke kebutuhan dalam pengembangan infrastruktur telekomunikasi menjadi suatu prioritas. Salah satu solusi untuk memenuhi kebutuhan tersebut adalah melalui jaringan akses *Fiber to the Home* (FTTH).

FTTH adalah jaringan akses dengan transmisi serat optik yang menghubungkan penyedia layanan telekomunikasi dengan rumah pengguna. Ini memungkinkan penyedia layanan untuk menyediakan akses triple play. Jaringan FTTH terdiri dari perangkat *Optical Line Termination* (OLT) yang mengubah sinyal listrik menjadi sinyal optik, diteruskan melalui *Optical Distribution Network* (ODN) untuk mengarahkan sinyal optik ke *Optical Network Termination* (ONT) di rumah pelanggan, dan ONT yang mengubah kembali sinyal optik menjadi sinyal listrik. ODN terdiri dari perangkat seperti *Optical Distribution Frame*, *optical distribution cabinet* (ODC) dan *Optical Distribution Point* (ODP)[1].

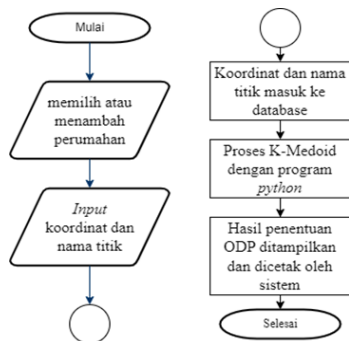
Pada makalah terdahulu telah dilakukan perancangan jaringan akses FTTH di perumahan Harmony Residence Jangli [2]. Selain itu, terdapat makalah mengenai perancangan ulang jaringan akses FTTH di perumahan Greenwood Gunungpati Semarang [3][4]. disebutkan bahwa pada perancangan jaringan akses pada makalah tersebut menggunakan *clustering* K-means sebagai dasar penentuan ODP. Pada makalah tersebut juga memiliki kebutuhan pada pelaksanaan dibutuhkan *software* pendukung agar dapat menghemat waktu dalam proses input data dan *clustering* penentuan ODP.

Dari permasalahan tersebut, makalah berfokus ini pada perancangan sistem penentu ODP berbasis web yang hasilnya dapat dimanfaatkan dalam perancangan jaringan akses FTTH di perumahan Punsae. Dalam prosesnya dilakukan pengambilan data site plan dan observasi lapangan, perancangan sistem *input* data koordinat lapangan berbasis web, dan penentuan ODP menggunakan algoritma K-medoid.

2. Metodologi Makalah

2.1. Langkah Penelitian

Perancangan sistem penentu ODP dilakukan setelah penentuan lokasi perancangan. Perancangan tersebut terdiri dari dua tahap. Tahap pertama yaitu pembuatan kerangka web untuk melakukan manajemen koordinat dari perumahan punsae dengan menggunakan bahasa pemrograman PHP dan Tahap yang kedua pembuatan program penentuan lokasi penempatan ODP menggunakan algoritma *clustering* K-Medoid menggunakan bahasa pemrograman python. Diagram alir system penentu ODP ditunjukkan pada Gambar 1



Gambar 1 diagram alir sistem penentu ODP

2.2 Analisis Kebutuhan

Analisis kebutuhan sistem ini ditunjukkan untuk menguraikan kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna dan sesuai dengan tujuan perancangan[5], yaitu merancang sistem penentu ODP berbasis web. Rancangan sistem ini

membutuhkan kebutuhan fungsional dan kebutuhan non fungsional

2.2.1 Kebutuhan fungsional

Kebutuhan fungsional adalah jenis kebutuhan yang berisi tentang proses-proses apa saja yang nantinya dilakukan oleh sistem. Kebutuhan fungsional juga berisikan tentang informasi-informasi apa saja yang harus ada dan dihasilkan oleh sistem. Kebutuhan fungsional sistem meliputi[6]:

- Tampilan utama web menampilkan daftar perumahan, akses untuk menambah perumahan dan menghapus perumahan, akses ke antarmuka peta *input* dan peta hasil *clustering*,
- Dalam halaman peta *input* menampilkan peta interaktif untuk memilih koordinat diikuti oleh *form* untuk memberi nama koordinat, yang akan ditampilkan pada peta dan tabel, pada tabel akan berisi nama titik, *latitude*, *longitude*, akses untuk menghapus koordinat, dan akses untuk meminta dilakukan program *clustering*.
- Dalam halaman peta hasil menampilkan peta hasil *clustering* dan tabel berisi anggota *cluster* dan status ODP atau bukan, dan pada halaman ini akan memberikan akses untuk melakukan *reclustering* pada peta untuk menghasilkan hasil *clustering* yang baru, akses untuk diarahkan ke peta *input* untuk mengubah titik, dan akses untuk *export* hasil ke format *file .csv*

2.2.2 Kebutuhan Non-fungsional

Analisa kebutuhan non-fungsional merupakan analisa yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi non-fungsional juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan mulai dari sistem dibangun sampai diimplementasikan.[6]. Pada analisis kebutuhan non-fungsional dijelaskan analisis kebutuhan perangkat keras, perangkat lunak diantaranya sebagai berikut:

1. Perangkat Keras (*Hardware*)

Kebutuhan perangkat keras dalam membangun sistem ini terdiri dari laptop atau PC dengan spesifikasi yang ditunjukkan pada Tabel 1

Tabel 1 Kebutuhan Perangkat Keras

| Perangkat | Deskripsi |
|------------|--|
| Processor | Intel(R) Core (TM) i5-8250U CPU @1.60GHz 1.80 GHz |
| RAM | DDR4 RAM 12GB |
| GPU | Nvidia Geforce MX150 |
| Memori | 1 Tb HDD dan 256 Gb SSD |
| Seri Model | Acer Swift 3 |

2. Perangkat Lunak (*Software*)

Adapun perangkat lunak yang digunakan dalam membangun sistem ini yang ditunjukkan pada Tabel 2

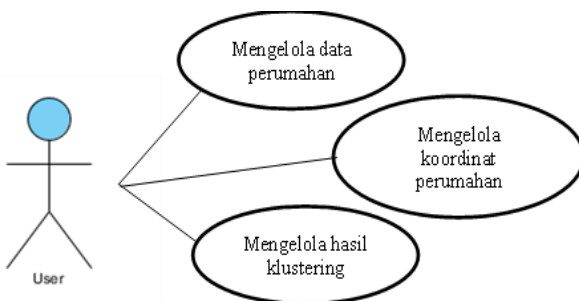
Tabel 2 Kebutuhan Perangkat Lunak

| Perangkat Lunak | Versi |
|--------------------|-----------------|
| Operating Sistem | Windows 11 Home |
| Xampp | V3.3.0 |
| PHP | V8.1.6 |
| Python | V3.10.5 |
| Visual Studio Code | V1.73 |

2.3 Desain Sistem Program

2.3.1 Use Case Diagram

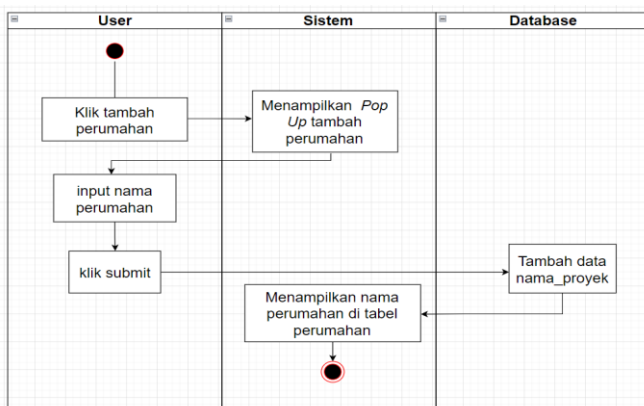
Diagram *use case* menggambarkan fungsi-fungsi yang ada pada sistem, diagram ini menunjukan fitur fitur sistem dari sudut pandang pihak luar.[6] Gambar 2 merupakan diagram *use case* yang digunakan dalam sistem penentu ODP



Gambar 2 Use Case Diagram

2.3.2 Diagram Aktivitas

Use case yang telah dipaparkan pada subbab 2.3.2 akan dijelaskan lebih lanjut menggunakan diagram aktivitas. Diagram aktivitas ini berfungsi untuk menggambarkan proses pada setiap *use case*. [6] Gambar 3 menunjukkan salah satu contoh diagram aktivitas pada perancangan Sistem Penentu ODP dalam menambah nama perumahan

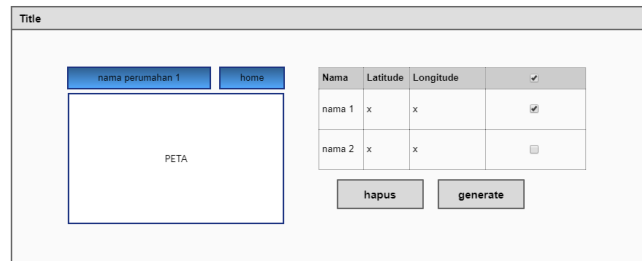


Gambar 3 Diagram aktivitas menambah nama perumahan

Diagram aktivitas menambah nama perumahan pada Gambar 3 menjelaskan proses menambah nama perumahan dimulai ketika user menekan tombol “tambah perumahan” lalu pada web akan muncul tampilan pop up berupa *form* untuk menuliskan nama perumahan dan tombol submit. Ketika ditekan “submit” maka sistem akan menambahkan data pada tabel *database* nama_proyek, yang kemudian sistem akan menampilkan nama perumahan yang telah ditambahkan pada tabel perumahan.

2.3.3 Perancangan Desain Antarmuka

Antarmuka/interface merupakan mekanisme komunikasi antara pengguna/user dengan sistem. Antarmuka/interface dapat menerima informasi dari pengguna/user dan memberikan informasi kepada pengguna/user untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan suatu solusi. Tujuan dari antar muka/interface sendiri yaitu mengkomunikasikan fitur fitur sistem yang tersedia agar user dapat mengerti dan menggunakan sistem tersebut.. Gambar 4 merupakan salah satu contoh mockup desain User Interface untuk diimplementasikan dalam Sistem Penentu ODP berbasis K-medoid:

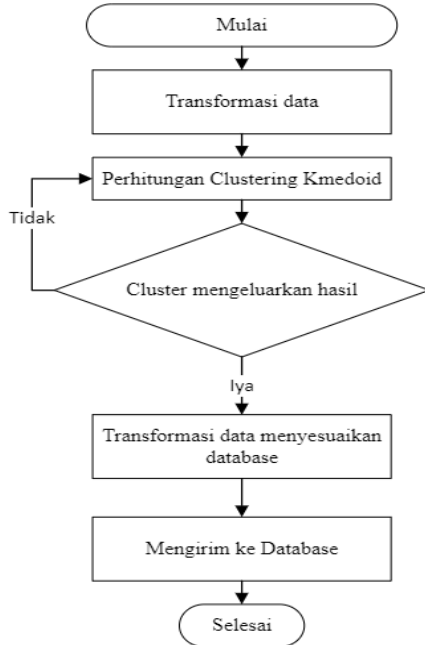


Gambar 4 Mockup antarmuka Peta Hasil

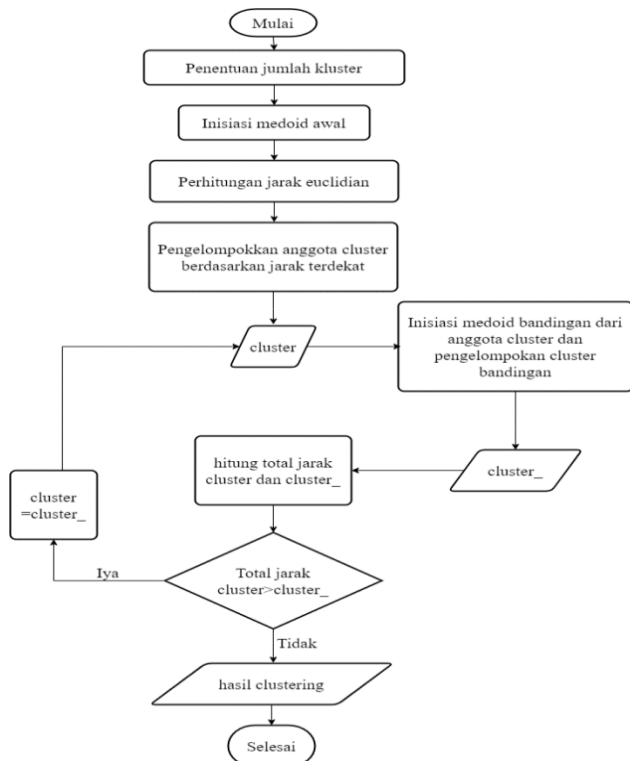
2.3.5 Perancangan Algoritma K-Medoid

Pada makalah ini, lokasi ODP ditentukan menggunakan algoritma *clustering* K-medoid. Algoritma K-Medoids membagi n objek menjadi k *cluster* dengan metode partisi. Algoritma ini memilih objek dari setiap cluster sebagai perwakilan *cluster*. Objek perwakilan disebut medoids dan berada di tengah-tengah *cluster* sehingga tahan terhadap *outlier*. *Cluster* dibentuk dengan mengukur jarak antara medoid dan objek lainnya [7]. makalah sebelumnya juga menggunakan metode *clustering*, khususnya K-means, untuk menentukan lokasi ODP. Dalam penentuan letak ODP menggunakan metode *K-means* data-data koordinat rumah pada kelompok perumahan di bagi menjadi sejumlah *cluster* dimana titik *centroid* pada *k-means* digunakan sebagai rekomendasi penentuan letak ODP. Namun, hasilnya menunjukkan bahwa titik centroid dapat terletak di tengah jalan atau di dalam rumah penduduk, sehingga perlu dilakukan penyesuaian. Oleh karena itu, dengan penggunaan *clustering* K-medoid, di mana titik medoid sudah ditentukan

mampu memberikan solusi dalam menentukan lokasi ODP. Gambar 5 dan Gambar 6 merupakan diagram alir program perhitungan *K-medoid* dan perhitungan *K-medoid*



Gambar 5 Diagram Alir Program perhitungan *K-medoid*



Gambar 6 Diagram Alir Perhitungan K-medoid

Pada perancangan kali ini ketika program K-medoid dijalankan, data yang pada *database* akan disesuaikan tipe datanya dari data teks ke data numerik sehingga dapat diproses dalam perhitungan *clustering* setelah dilakukan, hasil *clustering* akan dikelompokkan sesuai dengan *database* yang ada pada tabel 'odp', setelah disesuaikan program akan mengirimkan hasilnya ke *database*.

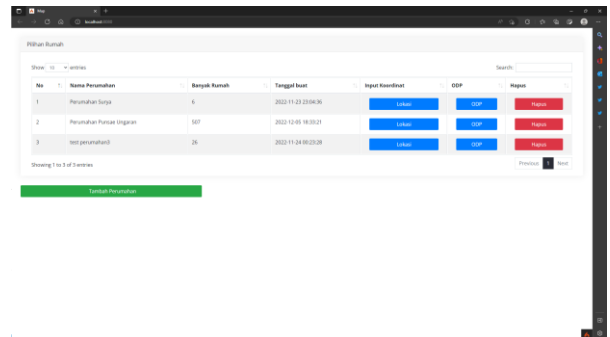
3. Analisis Hasil Perancangan

3.1 Implementasi Perancangan Web

3.1.1 Implementasi Antarmuka

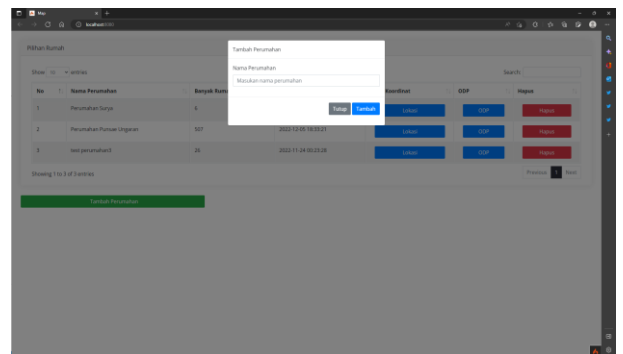
Implementasi antarmuka merupakan implementasi dari perancangan antarmuka yang telah dilakukan sebelumnya. Berikut merupakan implementasi dari perancangan sistem penentu ODP menggunakan metode K-medoid berbasis web.

1. Implementasi Halaman Daftar Perumahan
Berikut merupakan Antarmuka Halaman Daftar Perumahan yang merupakan Halaman Utama pada hasil perancangan yang ditunjukkan pada Gambar 7



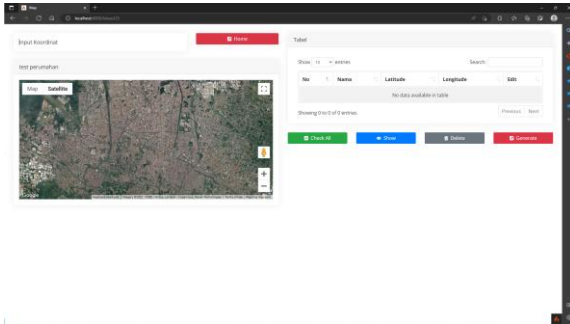
Gambar 7 Antarmuka Halaman Daftar Perumahan

2. Implementasi Halaman Tambah Perumahan
Berikut merupakan antarmuka tambah perumahan yang ada pada halaman utama aplikasi untuk memberi nama perumahan yang ditunjukkan pada Gambar 8

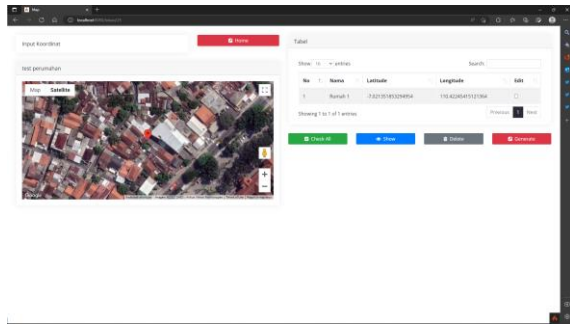


Gambar 8 Antarmuka Halaman Tambah Perumahan

- Implementasi Halaman Peta *Input* Koordinat
Berikut merupakan antarmuka Peta *Input* Koordinat sebelum ditambahkan data koordinat yang ditunjukkan pada Gambar 9 dan setelah ditambahkan data koordinat ditunjukkan pada Gambar 10

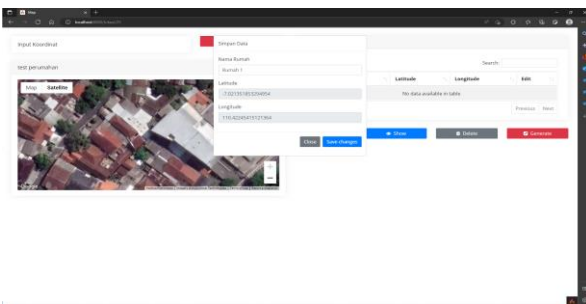


Gambar 9 Antarmuka Halaman Peta Input Koordinat



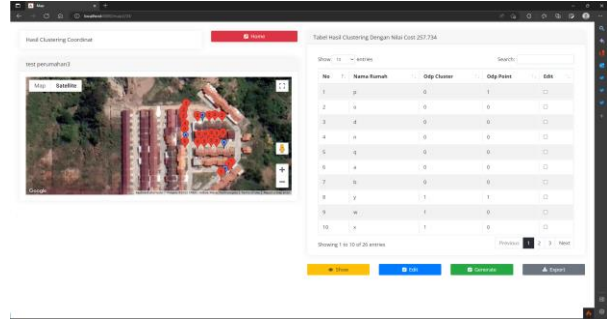
Gambar 10 Antarmuka halaman setelah ditambah koordinat

- Implementasi Halaman Tambah Koordinat
Berikut merupakan antarmuka halaman ketika menambahkan data koordinat yang ditunjukkan pada Gambar 11



Gambar 11 Antarmuka Halaman Tambah Koordinat

- Implementasi Halaman Hasil *Clustering*
Berikut merupakan antarmuka halaman hasil *clustering* yang ditunjukkan pada Gambar 12



Gambar 12 Antarmuka halaman Hasil *Clustering*

3.1.2 Pengujian Black Box

Black box testing adalah metode pengujian perangkat lunak yang mengevaluasi bagaimana aplikasi berperilaku saat menangani masukan dan menghasilkan keluaran. Tujuannya adalah untuk memastikan bahwa aplikasi berfungsi sesuai yang diharapkan dan sesuai dengan spesifikasi yang diberikan, tanpa perlu mengerti bagaimana aplikasi tersebut bekerja secara internal. Pengujian ini bertujuan untuk menguji fungsionalitas aplikasi dan mengevaluasi apakah sesuai dengan persyaratan pengguna dan pengalaman pengguna.[26] Tabel 3 merupakan hasil pengujian *black box*

Tabel 3 Pengujian *Black box*

| Aktor | Use Case | Hasil Uji |
|--------------------|-------------------------------|--|
| Perancang Jaringan | Mengelola data perumahan | a. Tambah data perumahan <i>User</i> berhasil menambah data perumahan baru |
| | | b. Hapus data perumahan <i>User</i> berhasil Menghapus data perumahan |
| Perancang Jaringan | Mengelola koordinat perumahan | a. Menambah data koordinat <i>User</i> berhasil menambah data koordinat dengan menekan suatu titik peta dan memberi nama pada koordinat |
| | | b. Menghapus data koordinat <i>User</i> berhasil Menghapus data koordinat |

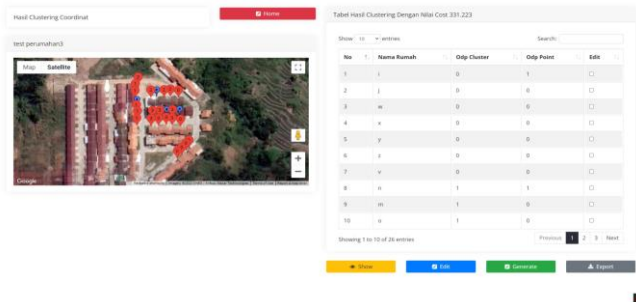
Tabel 3 Lanjutan Pengujian *Black Box*

| Aktor | Use Case | Hasil Uji |
|----------------------|--|--|
| Perancangan Jaringan | c. Melakukakan program <i>clustering</i> | Pengguna dapat menjalankan program <i>clustering</i> |
| | Mengelola data perumahan | Pengguna berhasil diarahkan ke halaman <i>input</i> mengubah data koordinat berhasil melakukan program <i>clustering</i> untuk mendapatkan hasil berbeda |
| | b. Mengulang proses <i>clustering</i> | Pengguna mendapatkan file hasil <i>clustering</i> dengan format .vsc |

3.2 Pengujian Hasil *Clustering*

3.2.1 Pengujian Nilai Cost

Pengujian nilai *cost* dilakukan dengan melakukan perbandingan nilai *cost* dari Sistem penentu ODP dibandingkan dengan perhitungan jarak manual. Pengujian ini dimaksudkan untuk membuktikan bahwa hasil yang keluar di *web* sudah sesuai dengan perhitungan. Perhitungan dilakukan dengan menggunakan pengukuran jarak Euclidian. Pengujian ini menggunakan titik uji sebanyak 26 titik. Gambar 13 merupakan hasil *clustering* uji *cost*



Gambar 13 Hasil *Clustering* Uji Cost

Tabel 4 merupakan hasil perhitungan uji *cost*

| Sumber | Nilai Cost |
|----------------------------|------------|
| Sistem Penentu ODP | 331.223 m |
| Perhitungan Kmedoid Manual | 331.222 m |

Didapatkan hasil perhitungan dengan selisih 0.001 karena hasil pembulatan oleh karena itu dapat diberi kesimpulan bahwa nilai *clustering* yang tampil di Sistem Penentu ODP sudah sama dengan hasil perhitungan

3.2.2 Pengujian *Clustering* Dengan Variasi Jumlah Data

Pengujian *clustering* akan dilakukan dengan melakukan simulasi perancangan dari variasi jumlah data untuk diketahui keandalannya. Pada pengujian ini akan dilakukan dengan melakukan *clustering* sebanyak 100, 200 dan 400 data koordinat dan masing masing memiliki dilakukan percobaan sebanyak 20 kali. Tabel 5 merupakan hasil pengujian *clustering* dengan variasi data

Tabel 5 Hasil Pengujian *Clustering* Dengan Variasi Data

| No | Variasi Jumlah data | Keandalan |
|----|---------------------|-----------|
| 1 | 100 | 90% |
| 2 | 200 | 95% |
| 3 | 400 | 80% |

3.2.3 Pengujian *Clustering* Dengan Studi Kasus Penentuan ODP Pedestal Perumahan Punsae Ungaran

Pengujian *Clustering* pada perumahan punsae bertujuan untuk “apakah sistem yang dirancang telah memberikan hasil *clustering* yang dapat dimanfaatkan dalam perancangan jaringan FTTH. Perumahan Punsae yang terdapat 507-unit rumah sehingga jumlah kluster yang sesuai untuk perancangan jaringan FTTH yang didapatkan dari membagi jumlah rumah ke kapasitas ODP sehingga didapatkan 32 sampai 64 ODP. Tabel 6 merupakan hasil pengujian penentuan ODP perumahan punsae Ungaran

Tabel 6 Hasil Pengujian Penentuan ODP Perumahan Punsae Ungaran

| Jumlah cluster | Jumlah kluster yang sesuai kapasitas |
|----------------|--------------------------------------|
| 32 -53 | 0 dari 110 percobaan |
| 54-59 | 7 dari 25 percobaan |
| 60-63 | 8 dari 25 percobaan |
| 64 | 1 dari 5 percobaan |

Tabel 6 memperlihatkan bahwa hasil perancangan mampu menghasilkan hasil penentuan ODP yang sesuai dengan

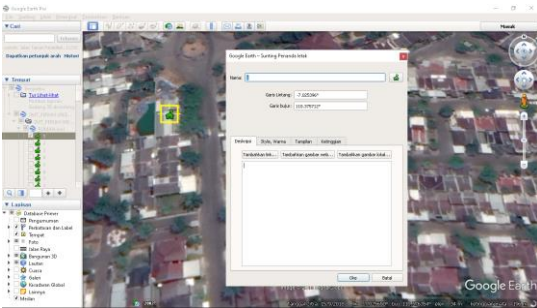
kapasitas ODP sebanyak 16 dari 165 kali percobaan dari 32 cluster ODP hingga 64 cluster ODP.

3.3 Perbedaan Implementasi Perancangan

Pada perancangan jaringan FTTH, sistem Penentu ODP berbasis web mencakup beberapa tahapan dalam proses perancangan jaringan FTTH yaitu mengupuliskan titik koordinat, dan penentuan titik ODP optimal. Terdapat perbedaan dalam proses Langkah pengerjaan pada perancangan jaringan FTTH antara menggunakan sistem penentu ODP berbasis web dengan dilakukan secara manual yaitu:

1. Mengambil data garis lintang dan garis bujur

Penggunaan sistem penentu ODP berbasis web proses pengumpulan titik dilakukan hanya dengan mengklik titik yang dituju pada peta, mengisi form pop-up untuk memberi nama titik, nilai koordinat akan muncul ditabel. Perbedaan dengan proses yang dilakukan dengan metode manual letak marker tidak langsung berada di titik yang diinginkan. Gambar 14 merupakan tampilan kotak dialog pengambilan data koordinat menggunakan aplikasi Google earth



Gambar 14 Tampilan kotak dialog pengambilan data koordinat menggunakan aplikasi Google earth

2. Proses Klustering

Dalam proses klustering dengan menggunakan sistem penentu ODP berbasis web yang telah dirancang proses klustering hanya dilakukan dengan menekan tombol *generate* yang ada di halaman input koordinat dan menjalankan program klustering python, yang dimana hasil proses klustering akan ditampilkan dihalaman ODP. Dibandingkan dengan proses klustering manual, perancang jaringan satu per satu harus melakukan hal hal berikut:

a. Menentukan jumlah cluster dan medoid awal

Dalam menentukan jumlah cluster dan medoid awal menggunakan sistem penentu ODP berbasis web yang telah dirancang aplikasi akan secara otomatis menghitung jumlah data koordinat dan dibagi 8. Dari hasil bagi tersebut aplikasi otomatis akan memilih titik data sejumlah data koordinat yang dibagi 8 secara acak. Pada proses perhitungan dengan

kmedoid secara manual, proses pembagian jumlah data koordinat dan penentuan dilakukan sendiri oleh perancang jaringan

b. Menghitung jarak masing masing antara ONT ke ODP.

Gambar 15 menunjukkan proses perhitungan jarak masing masing ONT ke masing masing medoid dengan metode perhitungan manual

| J | K | L | M | N | O | P | Q |
|-----|---------------|-------------|-------------|-------------|------------|------------|------------|
| 513 | JARAK | | | | | | |
| 514 | NOMOR RUMAH | C1 | C2 | C3 | C4 | C5 | C6 |
| 515 | Blok A-1 | 0 | 0,000429132 | 0,000235659 | 0,00055727 | 0,00078991 | 0,00097032 |
| 516 | Blok A-2 | 5,31269E-05 | 0,000481948 | 0,000234748 | 0,00058835 | 0,00080961 | 0,00098532 |
| 517 | Blok A-3 | 0,000108483 | 0,000536869 | 0,000244592 | 0,00062255 | 0,00083163 | 0,0010021 |
| 518 | Blok A-4 | 0,000160563 | 0,000588802 | 0,000266114 | 0,00065883 | 0,0008569 | 0,00102228 |
| 519 | Blok A-5 | 0,000227175 | 0,000655088 | 0,000303006 | 0,00070673 | 0,00089087 | 0,00104942 |
| 520 | Blok A-6 | 0,000281696 | 0,000709336 | 0,000339567 | 0,00074768 | 0,00092082 | 0,00107365 |
| 521 | Blok B-Office | 0,000429132 | 0 | 0,000492779 | 0,00043139 | 0,00071069 | 0,00090521 |
| 522 | Blok B-1 | 0,000186225 | 0,000281595 | 0,000220474 | 0,00038857 | 0,00064459 | 0,00083437 |
| 523 | Blok B-2 | 0,000284386 | 0,000144746 | 0,000372267 | 0,00043252 | 0,00071013 | 0,00090538 |
| 524 | Blok B-3 | 0,000363038 | 6,61048E-05 | 0,000435271 | 0,00042463 | 0,00070556 | 0,00090112 |
| 525 | Blok B1-1 | 0,000495967 | 0,000191243 | 0,000464569 | 0,00022565 | 0,00052828 | 0,00072058 |
| 526 | Blok B1-2 | 0,000431795 | 0,000202627 | 0,000389488 | 0,00022993 | 0,00051001 | 0,00070529 |
| 527 | Blok B1-3 | 0,00037809 | 0,000232134 | 0,000323553 | 0,00022555 | 0,00050382 | 0,00069936 |
| 528 | Blok B1-4 | 0,00034066 | 0,000368687 | 0,000201716 | 0,00022088 | 0,00045497 | 0,00064275 |

Gambar 15 Proses perhitungan jarak masing masing ONT ke masing masing medoid dengan metode perhitungan manual

Dalam menghitung jarak antara ont ke ODP menggunakan sistem penentu ODP berbasis web yang telah dirancang aplikasi akan secara otomatis jarak ke setiap medoid. Gambar 15 menunjukkan proses perhitungan dengan kmedoid secara manual, proses perhitungan jarak harus dilakukan satu persatu oleh perancang jaringan.

d. Menentukan pengelompokan ONT ke ODP

Gambar 16 menunjukkan proses pengelompokan ODP secara perhitungan manual

| BO | BP | BQ | BR | BS | BT | BU | BV | BW | BX | BY | BZ | CA |
|-----|----------|----------|----------|----------|----------|-----------|----------|----------|-------------|-------------|-----|-----|
| 513 | CS6 | CS7 | CS8 | CS9 | CS0 | CS1 | CS2 | CS3 | CS4 | CS5 | CS6 | CS7 |
| 514 | 0,004273 | 0,00384 | 0,0039 | 0,003737 | 0,004053 | 0,003867 | 0,00377 | 0,004002 | 0,004026 | 0 | 1 | |
| 515 | 0,004282 | 0,003867 | 0,003887 | 0,003723 | 0,004041 | 0,003935 | 0,003793 | 0,003985 | 0,004033 | 5,32399E-05 | 1 | |
| 516 | 0,004246 | 0,003823 | 0,003873 | 0,003703 | 0,004033 | 0,003943 | 0,003796 | 0,003982 | 0,004002 | 0,000104603 | 1 | |
| 517 | 0,004289 | 0,003862 | 0,003869 | 0,003819 | 0,004033 | 0,003728 | 0,00395 | 0,004062 | 0,000104603 | 1 | | |
| 518 | 0,004292 | 0,003787 | 0,003848 | 0,00367 | 0,004003 | 0,003821 | 0,003699 | 0,00393 | 0,004279 | 0,000227175 | 1 | |
| 519 | 0,004244 | 0,003775 | 0,003836 | 0,003651 | 0,003971 | 0,003811 | 0,003682 | 0,003923 | 0,004286 | 0,000104603 | 1 | |
| 520 | 0,004241 | 0,00382 | 0,003878 | 0,003844 | 0,004127 | 0,003941 | 0,003887 | 0,004117 | 0,004083 | 0 | 2 | |
| 521 | 0,004284 | 0,003775 | 0,003833 | 0,003884 | 0,003984 | 0,003929 | 0,003721 | 0,003993 | 0,004297 | 0,000104603 | 1 | |
| 522 | 0,004244 | 0,003888 | 0,003847 | 0,003803 | 0,004017 | 0,003841 | 0,003841 | 0,004074 | 0,004086 | 0,000104748 | 2 | |
| 523 | 0,004247 | 0,003904 | 0,003862 | 0,003823 | 0,004111 | 0,0039425 | 0,003883 | 0,004096 | 0,004179 | 6,83024E-05 | 2 | |
| 524 | 0,004281 | 0,003749 | 0,003802 | 0,003674 | 0,003995 | 0,003945 | 0,003737 | 0,003948 | 0,004214 | 0,000104603 | 2 | |
| 525 | 0,004248 | 0,003728 | 0,003775 | 0,003642 | 0,003974 | 0,003928 | 0,003748 | 0,003923 | 0,004297 | 0,000104603 | 2 | |

Gambar 16 Proses pengelompokan ODP secara perhitungan manual

Dalam penentuan anggota *cluster* ODP menggunakan sistem penentu ODP berbasis web yang telah dirancang aplikasi akan secara otomatis mengelompokkan ke masing masing ONT. Gambar 16 menunjukkan proses kmedoid secara manual, proses pengelompokan harus menuliskan skrip *excel* yang panjang dengan menyesuaikan jumlah cluster oleh perancang jaringan

d. Menentukan titik medoid baru

Dalam penentuan titik medoid baru menggunakan sistem penentu ODP berbasis web yang telah dirancang aplikasi akan secara otomatis mengambil acak medoid baru pada masing masing anggota kluster. Pada proses perhitungan dengan k-medoid secara manual penentuan medoid dilakukan sendiri oleh perancang jaringan.

e. Mengulang Iterasi

Pada proses iterasi perancang jaringan harus terus melakukan kembali langkah langkah sebelumnya hingga mendapatkan jarak total yang lebih besar dari kelompok kluster yang dilakukan sebelumnya. Maka diperlukan waktu yang lebih banyak dibandingkan dengan menggunakan aplikasi, dalam melakukan seluruh proses *clustering*.

4. Kesimpulan

Berdasarkan hasil perancangan dan analisis yang telah dilakukan dalam makalah ini adalah telah dilakukan. Telah dilakukan perancangan aplikasi penentu ODP berbasis web menggunakan algoritma klustering K-Medoid. dimana hasil perancangan telah diuji menggunakan pengujian *black box* yang dihasilkan bahwa hasil perancangan telah berfungsi sesuai dengan kriteria yang ditentukan dan dapat dimanfaatkan oleh perancang jaringan FTTH untuk dalam menentukan lokasi titik ODP dalam perancangan jaringan FTTH.

Pengujian hasil klustering dengan metode validasi didapatkan nilai cost klustering yang ditampilkan sesuai dengan perhitungan secara manual, pengujian klustering dengan perbandingan variasi data didapatkan nilai keaandalan 90% untuk percobaan dengan 100 data, 95 % untuk 200 data, dan 80 % untuk 400 data. Dapat disimpulkan bahwa semakin banyak jumlah data yang digunakan dalam pengujian klustering, maka kemungkinan *error* yang muncul dapat semakin meningkat.

Referensi

- [1] PT Telekomunikasi Indonesia Tbk., "Panduan Desain FTTH," Indonesia, 2012.
- [2] N. A. Adriel, "Perancangan Jaringan Akses Fiber To The Home Perumahan Harmony Residence Jangli Menggunakan Algoritma K-Means *Clustering* ," 2019.
- [3] A. A. Dewi, "Perancangan Ulang Jaringan Akses Fiber To The Home Perumahan Greenwood Gunung Pati Semarang," Departemen Teknik Elektro, Universitas Diponegoro, Semarang, 2019.
- [4] R. A. Yemima, "Perancangan Ulang Jaringan Akses Fiber To The Home Perumahan Greenwood Gunungpati Semarang pada

Instalasi Kabel Pedestal," Departemen Teknik Elektro, Universitas Diponegoro, Semarang, 2019.

- [5] Quzania, H., Christyono, Y., & Sukiswo, S. (2020). Perancangan Web Sistem Monitoring Stop Kontak Terkontrol Menggunakan Pemrograman Php Dan Mysql. *Transient: Jurnal Ilmiah Teknik Elektro*, 9(3), 412-417.
- [6] Anita (2017) Pembuatan Aplikasi Kumpulan Doa-Doa Harian Berbasis Android
- [7] Setiyawati, Astri Widiastuti (2017) Implementasi algoritma Partitioning Around Medoids (PAM) untuk pengelompokan Sekolah Menengah Atas di DIY berdasarkan nilai daya serap Ujian Nasional. Skripsi thesis, Sanata Dharma University.