

BAB IV

Hasil dan Pembahasan

Penelitian ini dimulai dengan pembangunan sistem terlebih dahulu untuk membantu proses penelitian. Sistem yang dibangun digunakan untuk membantu penulis dalam melakukan pengumpulan data (terutama data spasial), penyeragaman data, dan melakukan proses *clustering* serta menyajikan hasil *clustering* kedalam peta spasial. Hasil dan pembahasan dalam penelitian ini dapat dijelaskan sebagai berikut.

4.1 Perancangan Sistem

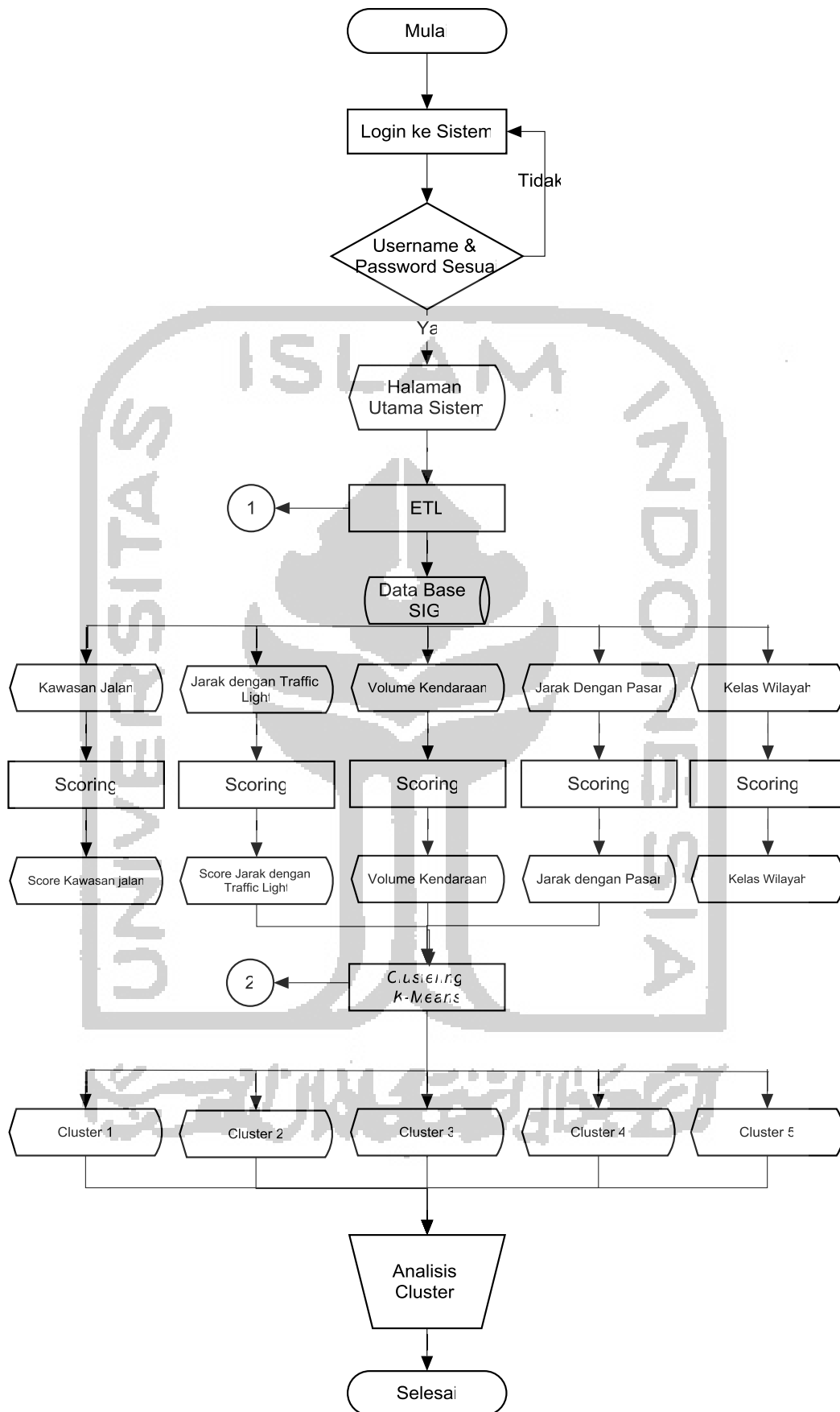
4.1.1 Diagram Alir Sistem

Pada tahap perancangan sistem, dapat digambarkan perancangan mengenai sistem yang dibangun. Sistem dibangun menggunakan bahasa pemrograman PHP 5.2 dan basisdata mysql yang dihubungkan dengan Google Maps. Gambar 4.1 menunjukkan diagram alir sistem.

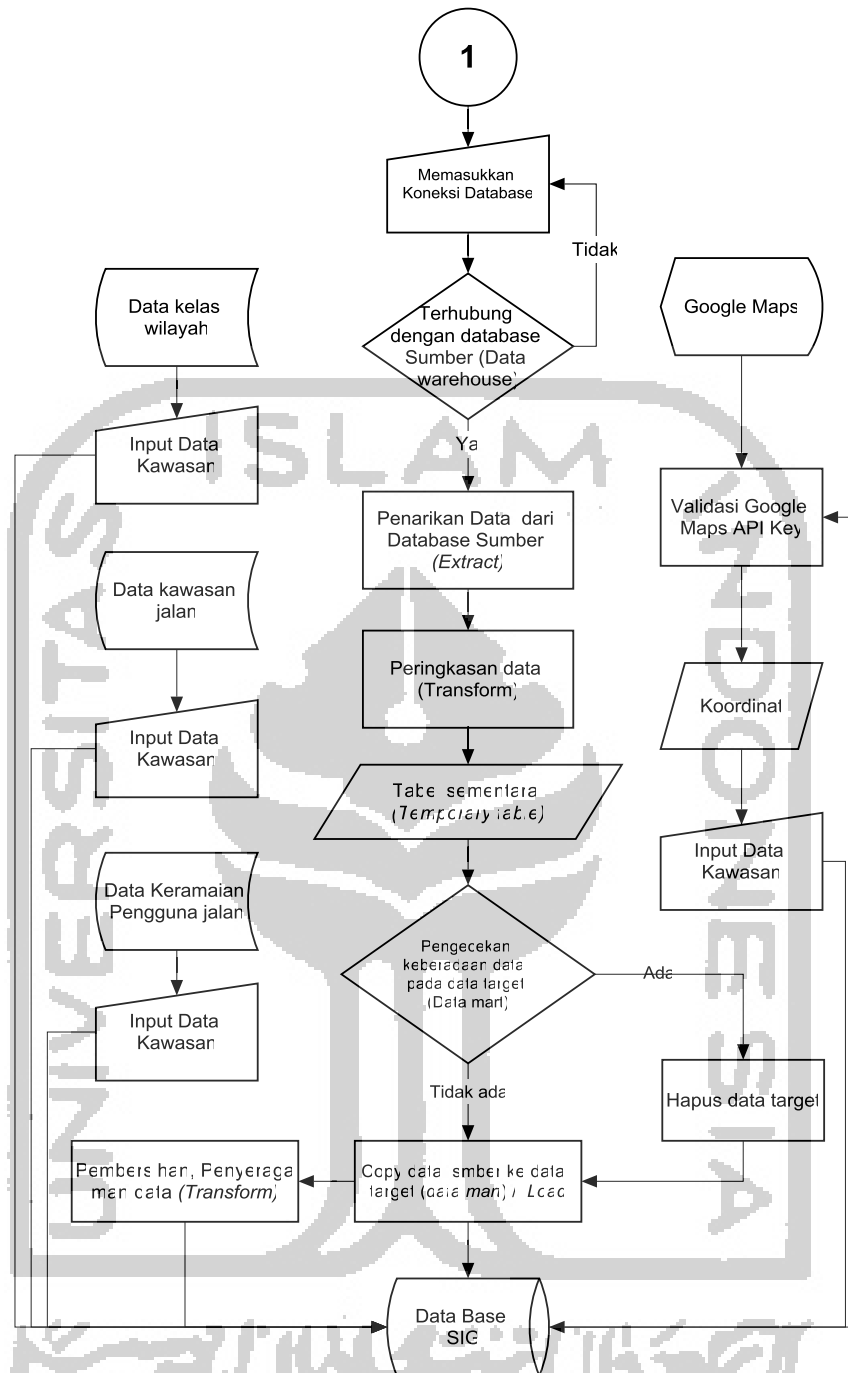
Berdasarkan Gambar 4.1, alur sistem dimulai dengan melakukan login ke dalam sistem terlebih dahulu. Apabila *username* dan *password* sesuai, maka user bisa memasuki halaman utama sistem. Pada halaman ini user dapat melakukan proses ETL yang dijelaskan pada bab selanjutnya. Setelah proses ETL selesai, maka terbentuklah basisdata SIG. Dari basisdata tersebut dilakukan proses *scoring* terhadap variabel-variabel yang telah dikumpulkan untuk proses analisis. Hasil *scoring* kemudian dilakukan *clustering* menggunakan algoritma *k-means*. Hasil *clustering* selanjutnya dianalisis untuk mendapatkan pemahaman ataupun pengetahuan baru tentang persebaran reklame.

4.1.2 Diagram Alir Pembentukan Basisdata

Pembentukan basisdata dilakukan dengan menggunakan sistem yang dibangun. Gambar 4.2 menunjukkan pembentukan basisdata SIG.



Gambar 4.1 Diagram Alir Sistem



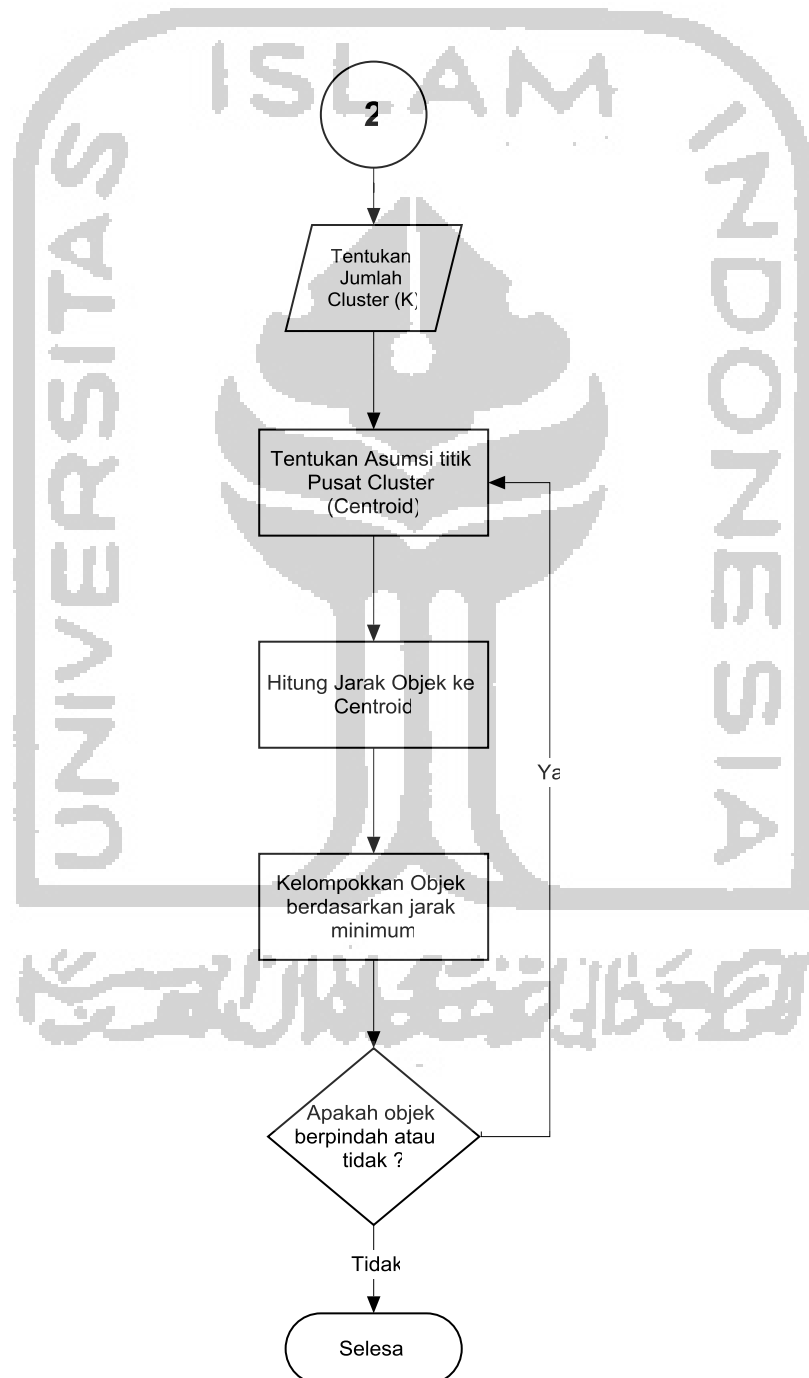
Gambar 4.2 Pembentukan Basisdata SIG

Berdasarkan gambar 4.2, pembentukan basisdata SIG dimulai dengan memasukkan koneksi basisdata SIAPADA melalui form yang ada pada aplikasi SIG. Setelah terhubung dengan basisdata SIAPADA maka dilakukan penarikan data. Data-data pada basisdata SIAPADA tidak ditarik semuanya, melainkan diringkas sesuai dengan kebutuhan penelitian. Data-data tersebut kemudian ditampung di dalam tabel sementara (*temporary table*). Sebelum data dimasukkan ke dalam basisdata SIG, terlebih dahulu dilakukan pengecekan keberadaan data pada basisdata SIG. Apabila ada data maka dilakukan

penghapusan data terlebih dahulu, namun kalau tidak ada data maka dilakukan proses *copy* data ke dalam basisdata SIG.

4.1.3 Diagram Alir *Clustering K-means*

Proses *Clustering* dilakukan menggunakan sistem yang dibangun. Gambar 4.3 menunjukkan diagram alir *clustering k-means*.

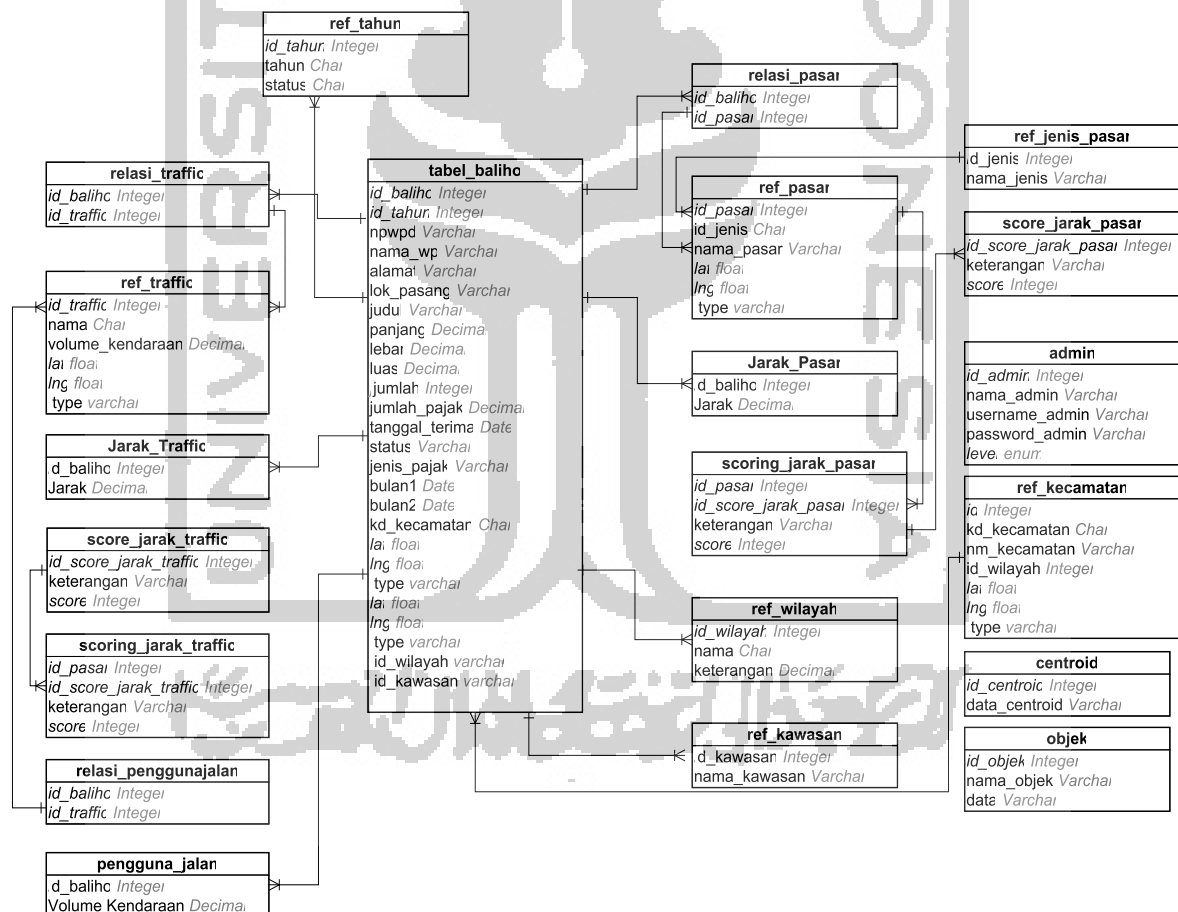


Gambar 4.3 Diagram Alir *Clustering K-means*

Berdasarkan Gambar 4.3, *Clustering K-means* dimulai dengan menentukan jumlah *cluster* (K) terlebih dahulu, kemudian menentukan asumsi titik pusat *cluster* (*centroid*). Setelah *centroid* terbentuk, selanjutnya adalah menghitung jarak objek ke *centroid*. Setelah jarak objek diketahui, objek-objek tersebut dikelompokkan berdasarkan jarak minimum. Apabila masih ada objek yang berpindah *cluster*, maka dilakukan pembaharuan *centroid* kembali, namun bila objek sudah tidak berpindah, maka proses dihentikan.

4.1.4 ERD (Entity Relationship Diagram)

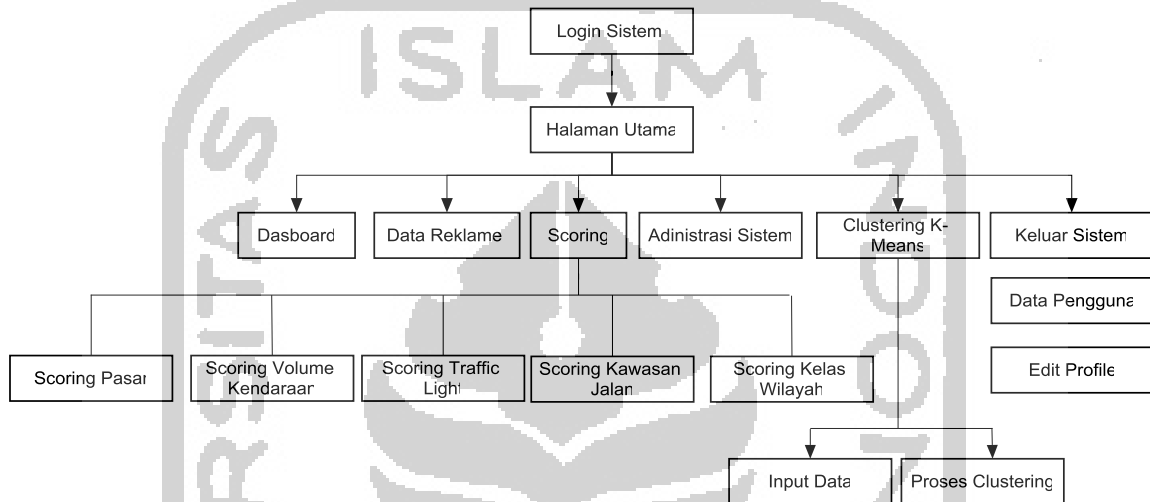
ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basisdata berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD dari sistem yang dibangun sebagaimana terlihat pada Gambar 4.4.



Gambar 4.4 ERD (Entity Relationship Diagram)

4.1.5 Perancangan Antarmuka (*User Interface*)

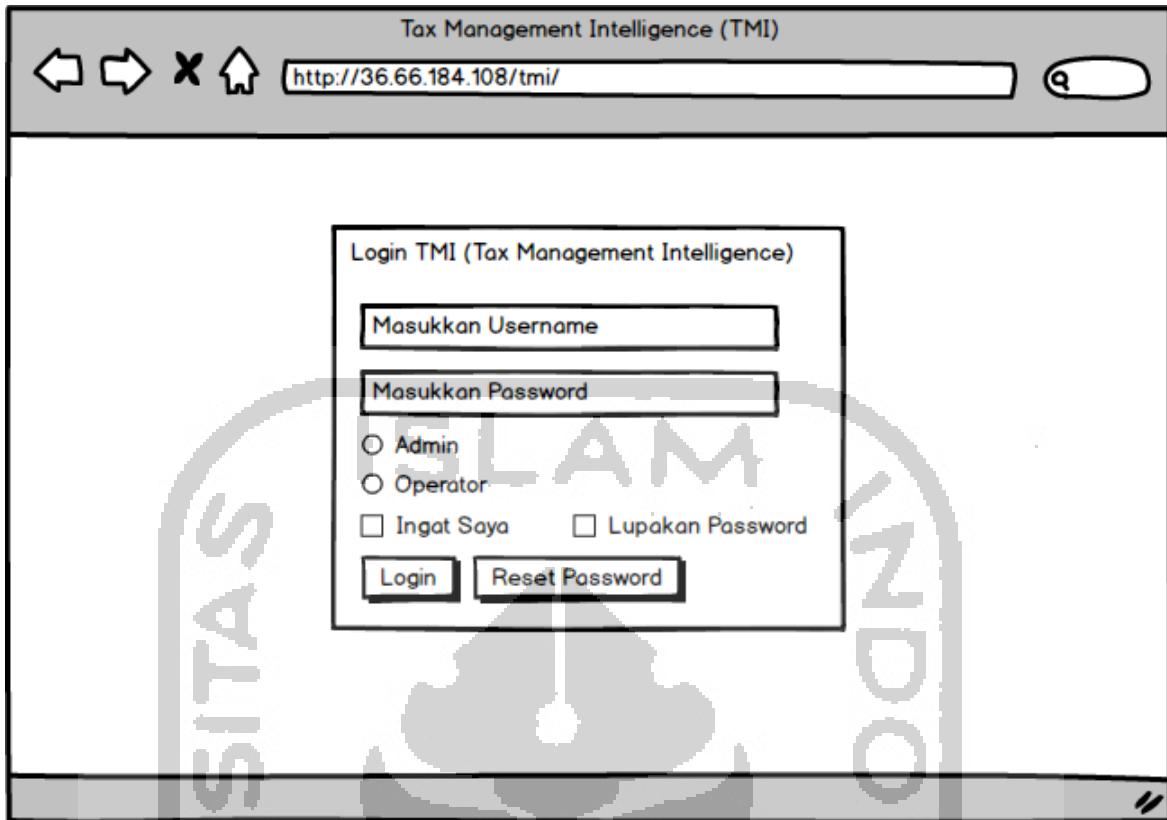
Perancangan antarmuka yang akan dilakukan terdiri dari perancangan struktur menu, desain dan layout. Perancangan ini dibuat menu yang dapat mengintegrasikan seluruh data suatu sistem. Selain itu juga untuk mempermudah pengguna sistem dalam menjalankan aplikasi yang dibangun, sehingga pada saat menjalankan sistem tidak mengalami kesulitan dalam memilih menu-menu yang diinginkan, adapun struktur menu pada aplikasi yang dibangun sebagaimana terlihat pada Gambar 4.5.



Gambar 4.5 Struktur Menu Aplikasi

a. Rancangan halaman Login sebelum pengguna mengakses sistem

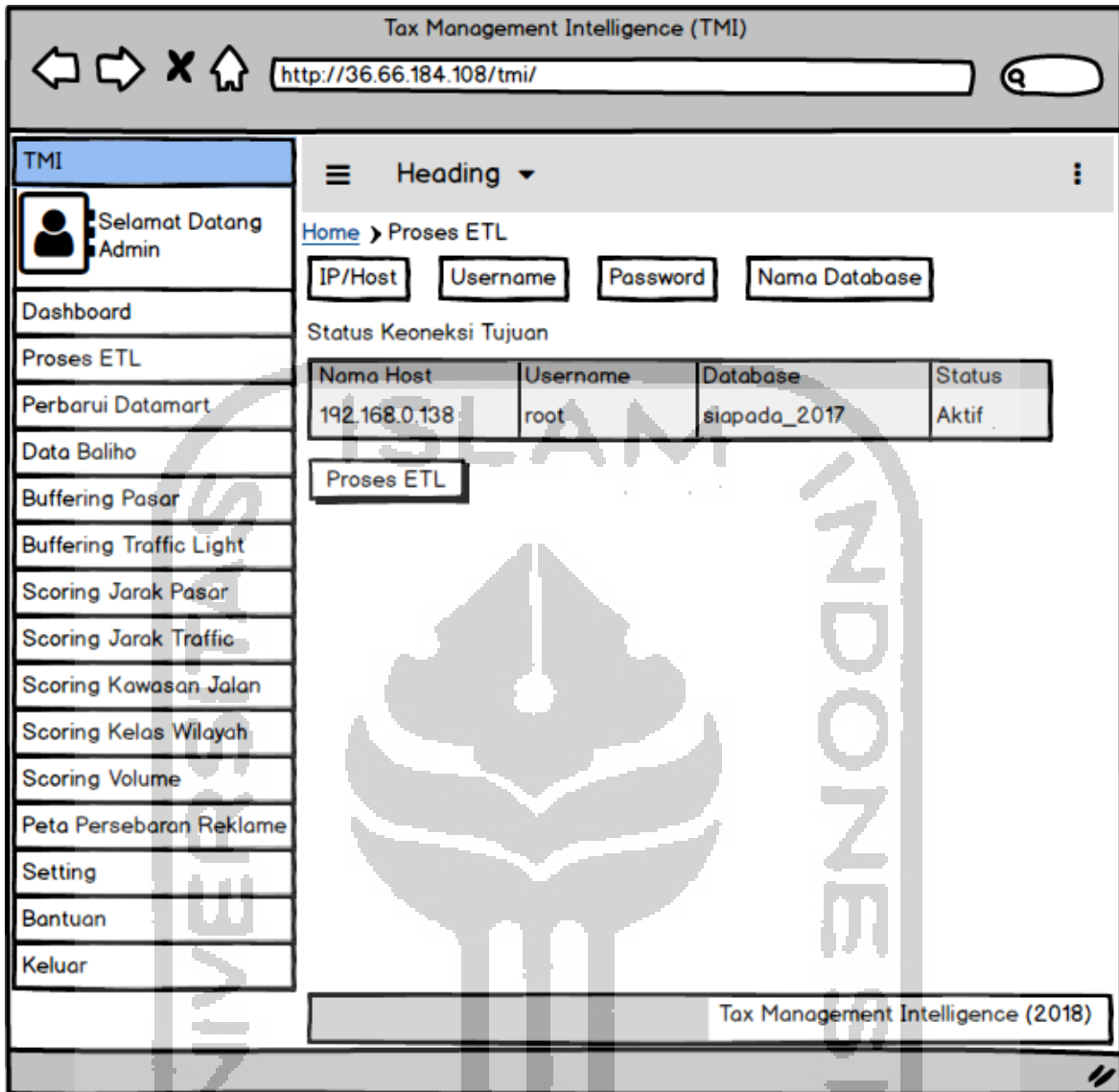
Rancangan Halaman login pada aplikasi terdiri atas kontrol *textbox* “*Username*” dan *textbox* “*Password*”, dan *button* “*Login*” untuk melakukan proses validasi pengguna sistem sebelum masuk ke halaman utama aplikasi. Rancangan halaman login sebagaimana terlihat pada Gambar 4.6 .



Gambar 4.6 Rancangan Halaman Login

b. Rancangan halaman *Extract* Basisdata

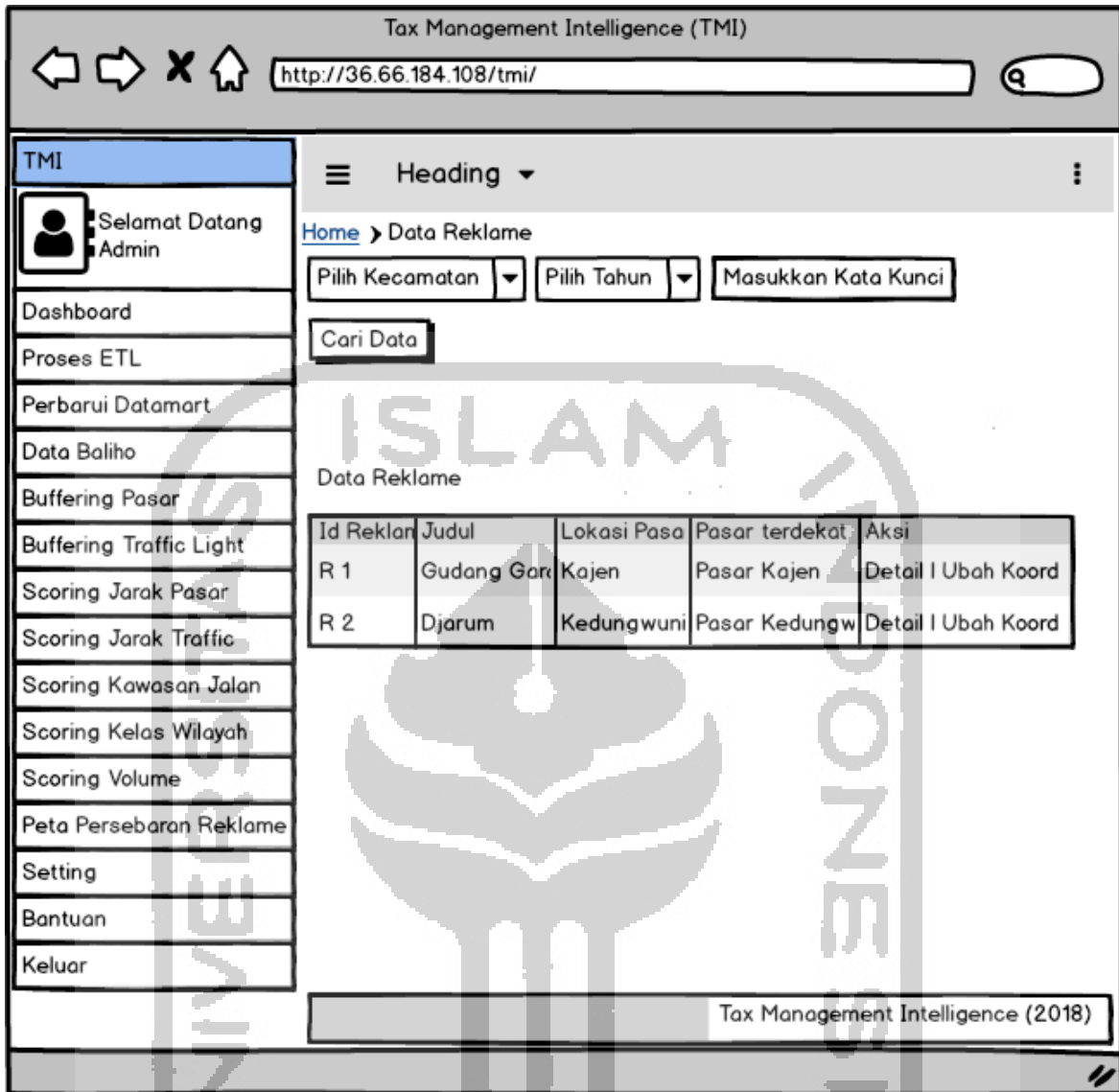
Rancangan halaman peta persebaran reklame pada aplikasi, terdiri atas *textbox* "IP/Host, Username, Password, Nama Basisdata" dan *button* "Proses ETL" untuk melakukan proses. Rancangan halaman ekstraksi basisdata sebagaimana terlihat pada Gambar 4.7.



Gambar 4.7 Rancangan Halaman *Extract* Basisdata

c. Rancangan Halaman Data Reklame

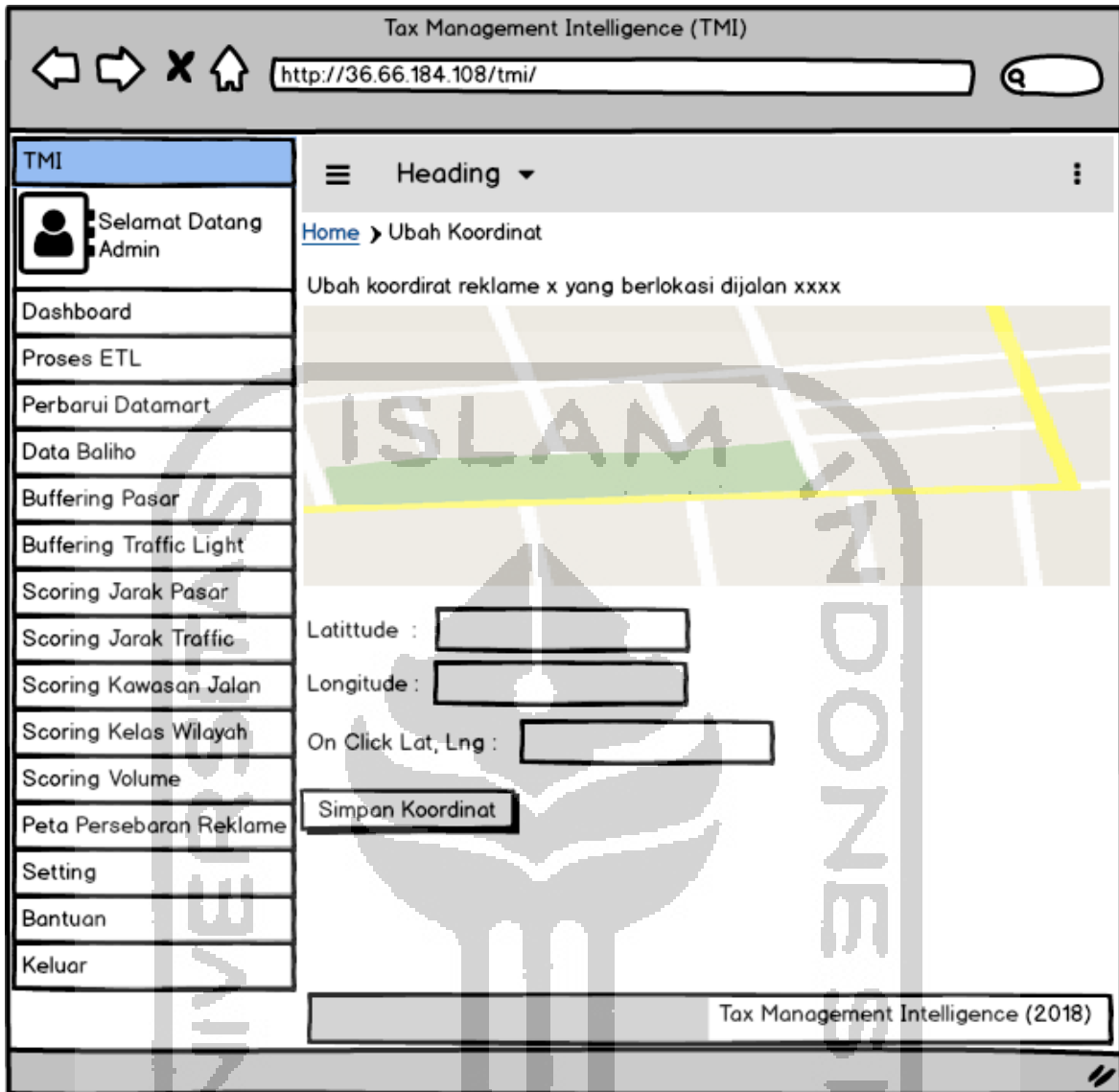
Rancangan halaman peta persebaran reklame pada aplikasi, terdiri atas *combobox* “pilih kecamatan”, *combobox* “pilih tahun”, *textbox* “kata kunci” dan *button* “cari data” untuk melakukan proses pencarian data. Rancangan halaman data reklame sebagaimana terlihat pada Gambar 4.8



Gambar 4.8 Rancangan Halaman Data Reklame

d. Rancangan Halaman Ubah Koordinat

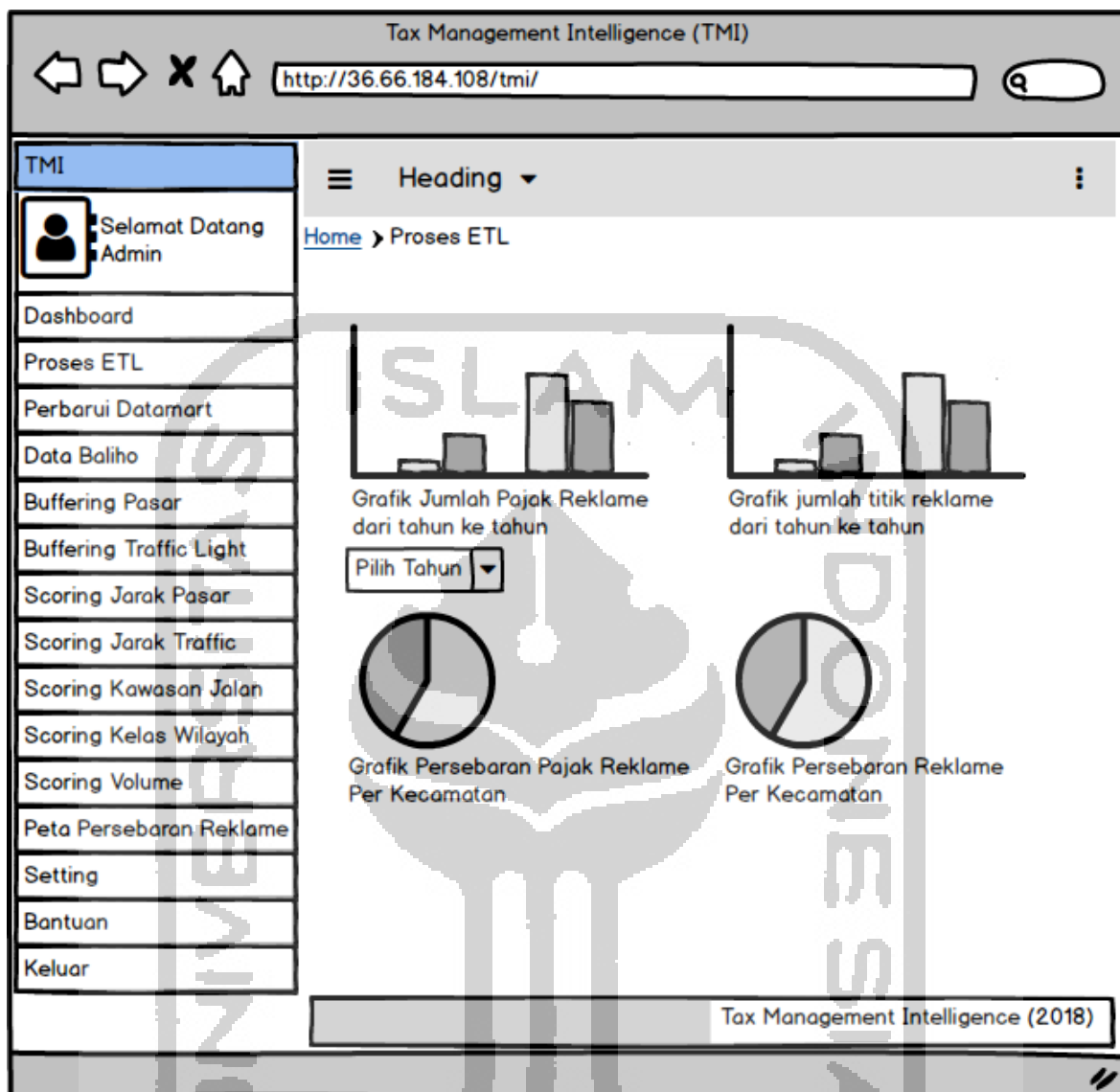
Rancangan halaman peta persebaran reklame pada aplikasi, terdiri atas *textbox* “*latittude*, *longitude*” dan *button* “ubah koordinat”. Rancangan halaman ubah koordinat sebagaimana terlihat pada Gambar 4.9.



Gambar 4.9 Rancangan Halaman Ubah Koordinat

e. Rancangan Halaman Dasbor

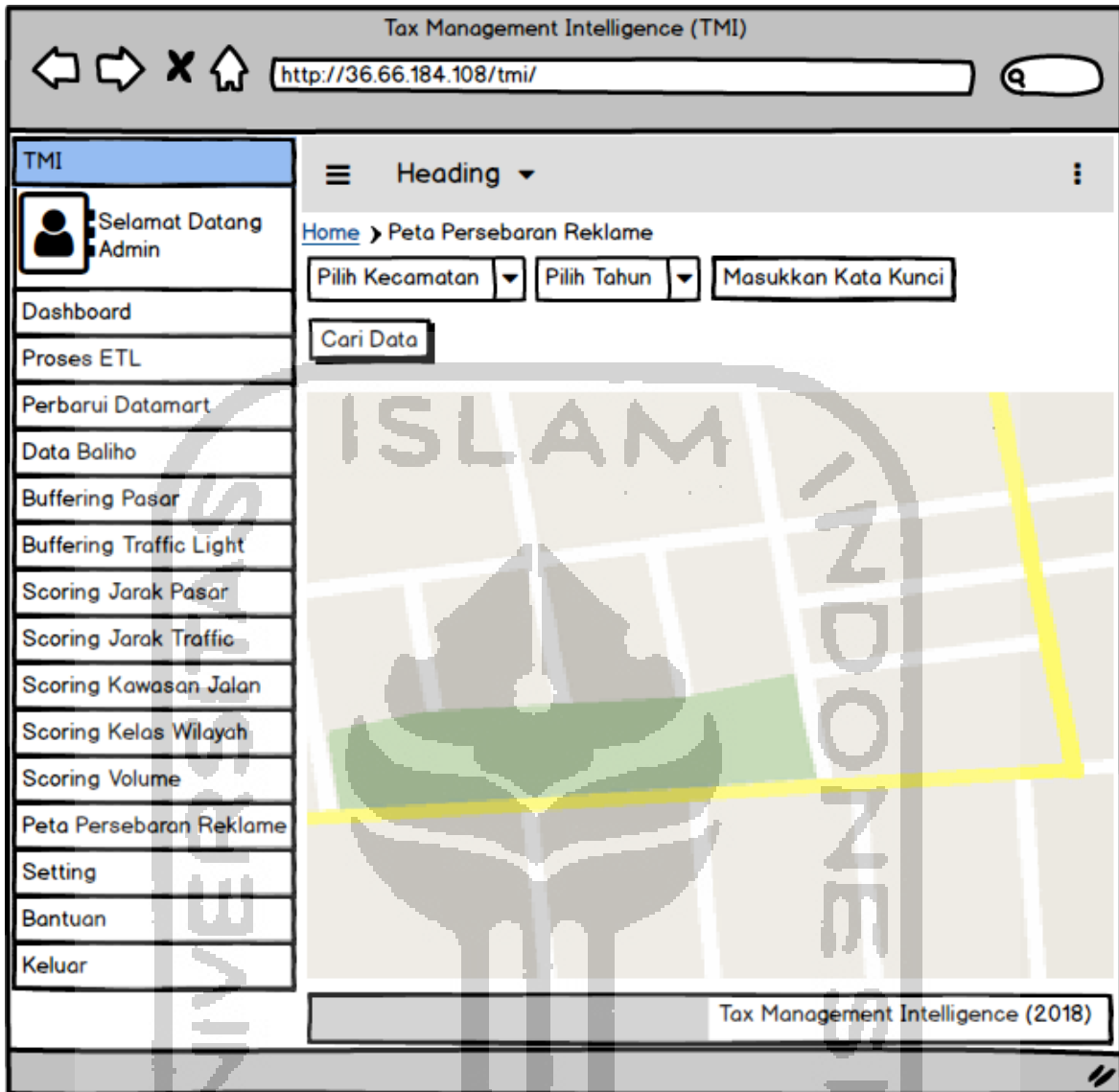
Halaman dasbor digunakan untuk menyajikan diagram ataupun grafik persebaran reklame baliho. Rancangan dasbor yang akan disajikan pada sistem ini adalah diagram batang pajak reklame dari tahun ke tahun berdasarkan jumlah penerimaan pajaknya, kemudian diagram batang pajak reklame dari tahun ke tahun berdasarkan titik reklame, serta diagram lingkaran persebaran reklame per-kecamatan. Rancangan halaman dasbor sebagaimana terlihat pada Gambar 4.10.



Gambar 4.10 Rancangan Halaman Dasbor

f. Rancangan halaman peta persebaran reklame

Rancangan halaman peta persebaran reklame pada aplikasi, terdiri atas *combobox* “pilih kecamatan”, *combobox* “pilih tahun”, *textbox* “kata kunci” dan *button* “cari data” untuk melakukan proses pencarian data, serta peta untuk menampilkan hasil dari proses pencarian data. Rancangan halaman peta persebaran reklame sebagaimana terlihat pada Gambar 4.11.

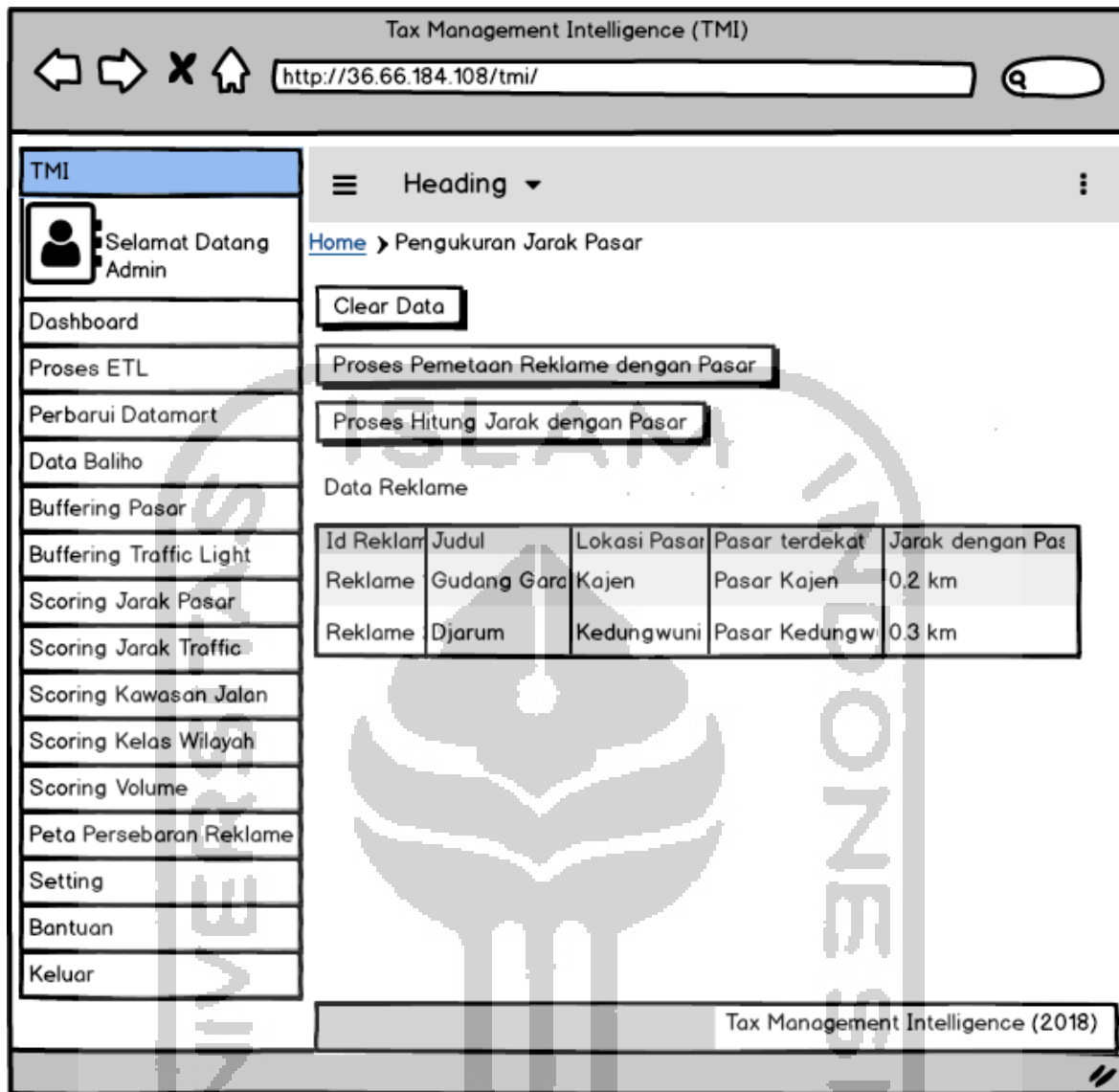


Gambar 4.11 Rancangan Halaman Peta Persebaran Reklame

g. Rancangan Halaman Pengukuran Jarak Pasar

Rancangan halaman pengukuran jarak pasar pada aplikasi, terdiri atas dan *button* “clear data”, “Proses Pemetaan Reklame dengan Pasar”, dan “Proses Hitung Jarak dengan Pasar”.

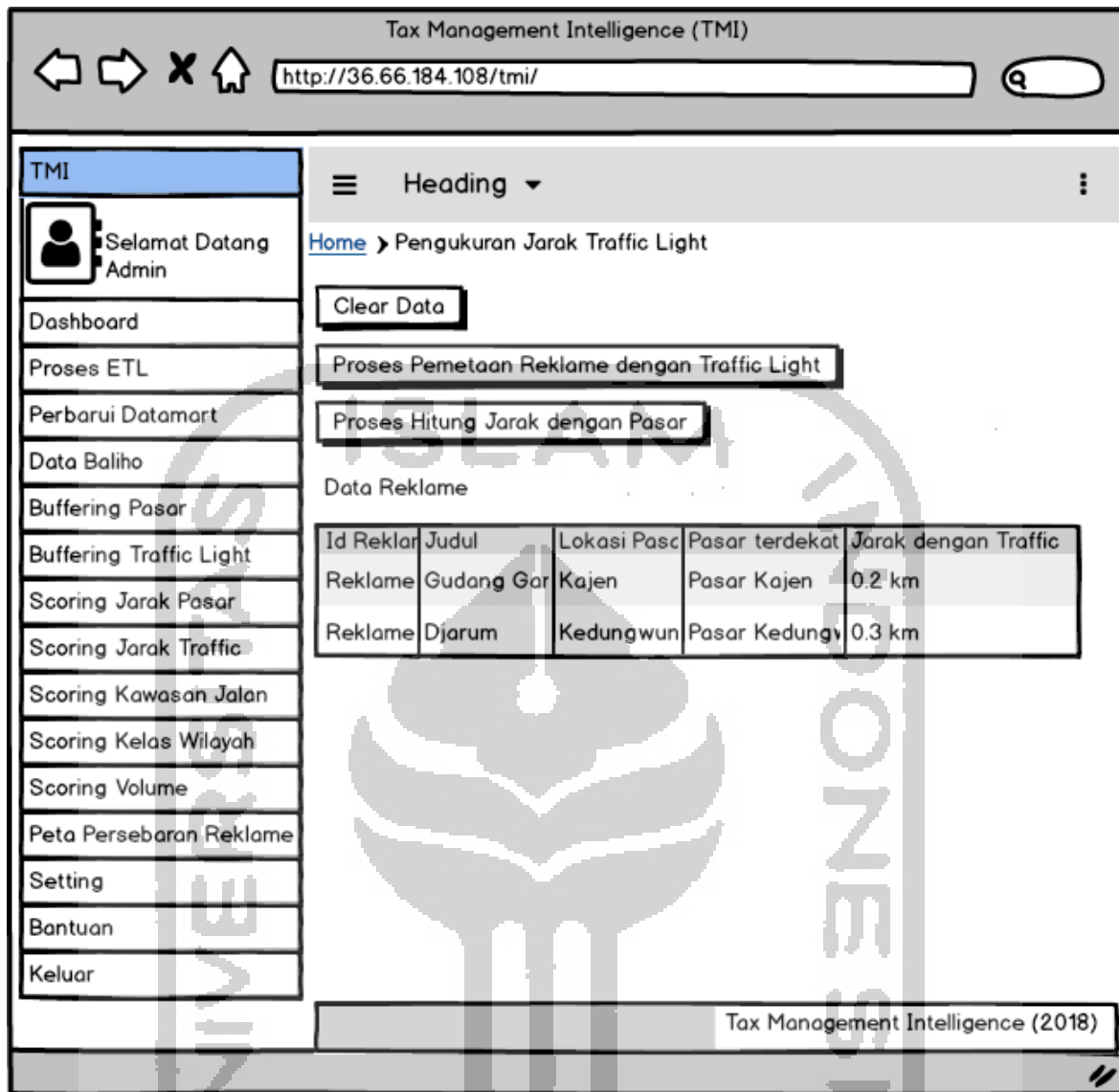
Rancangan halaman pengukuran jarak pasar sebagaimana terlihat pada Gambar 4.12.



Gambar 4.12 Rancangan Halaman Pengukuran Jarak Pasar

h. Rancangan Halaman Pengukuran Jarak *Traffic light*

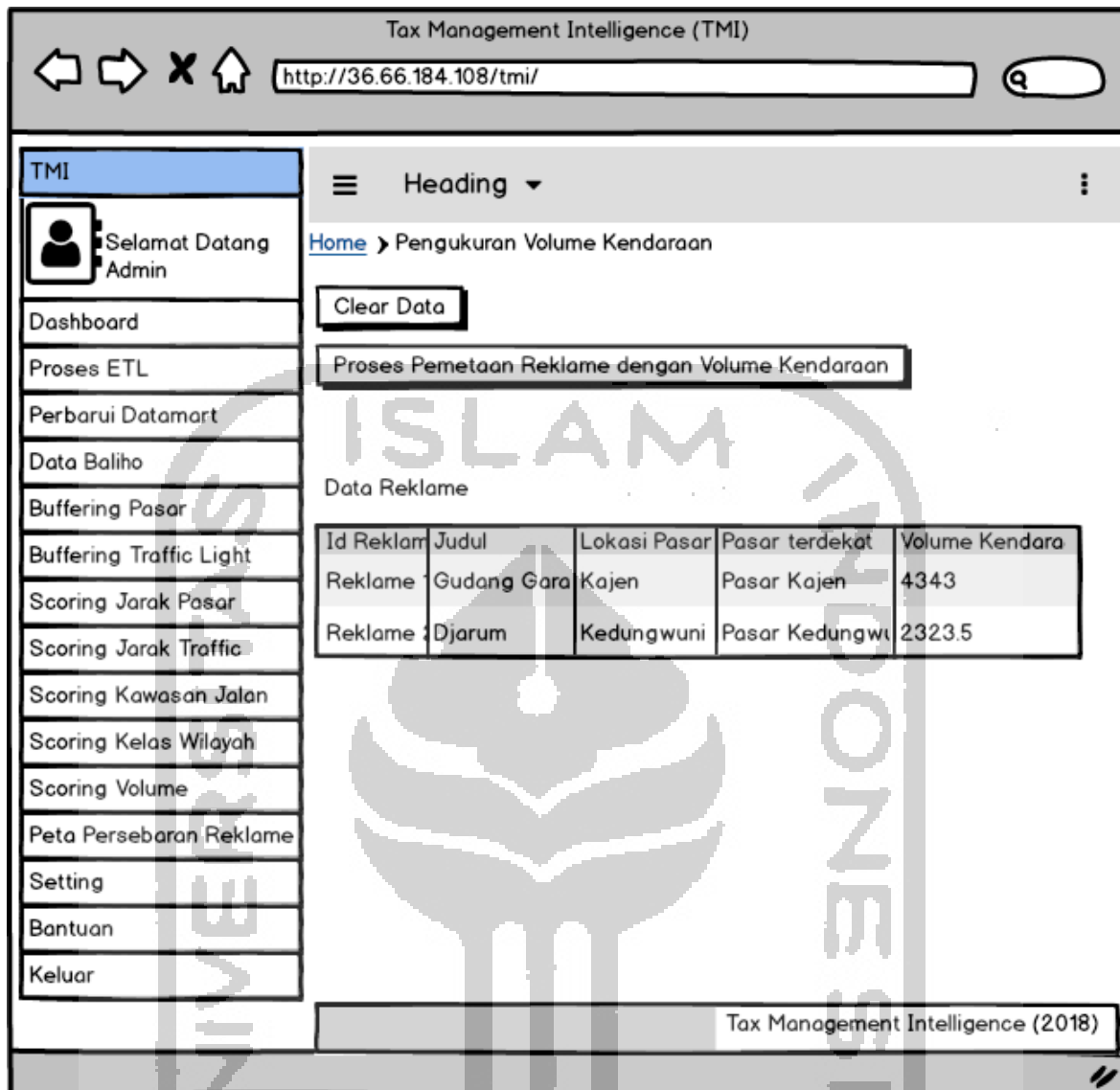
Rancangan halaman pengukuran jarak pasar pada aplikasi, terdiri atas dan *button* “clear data”, “Proses Pemetaan Reklame dengan *Traffic light*”, dan “Proses Hitung Jarak dengan *Traffic light*”. Rancangan halaman pengukuran jarak *traffic light* sebagaimana terlihat pada Gambar 4.13.



Gambar 4.13 Rancangan Halaman Pengukuran Jarak *Traffic light*

i. Rancangan Halaman Pengukuran Volume Kendaraan

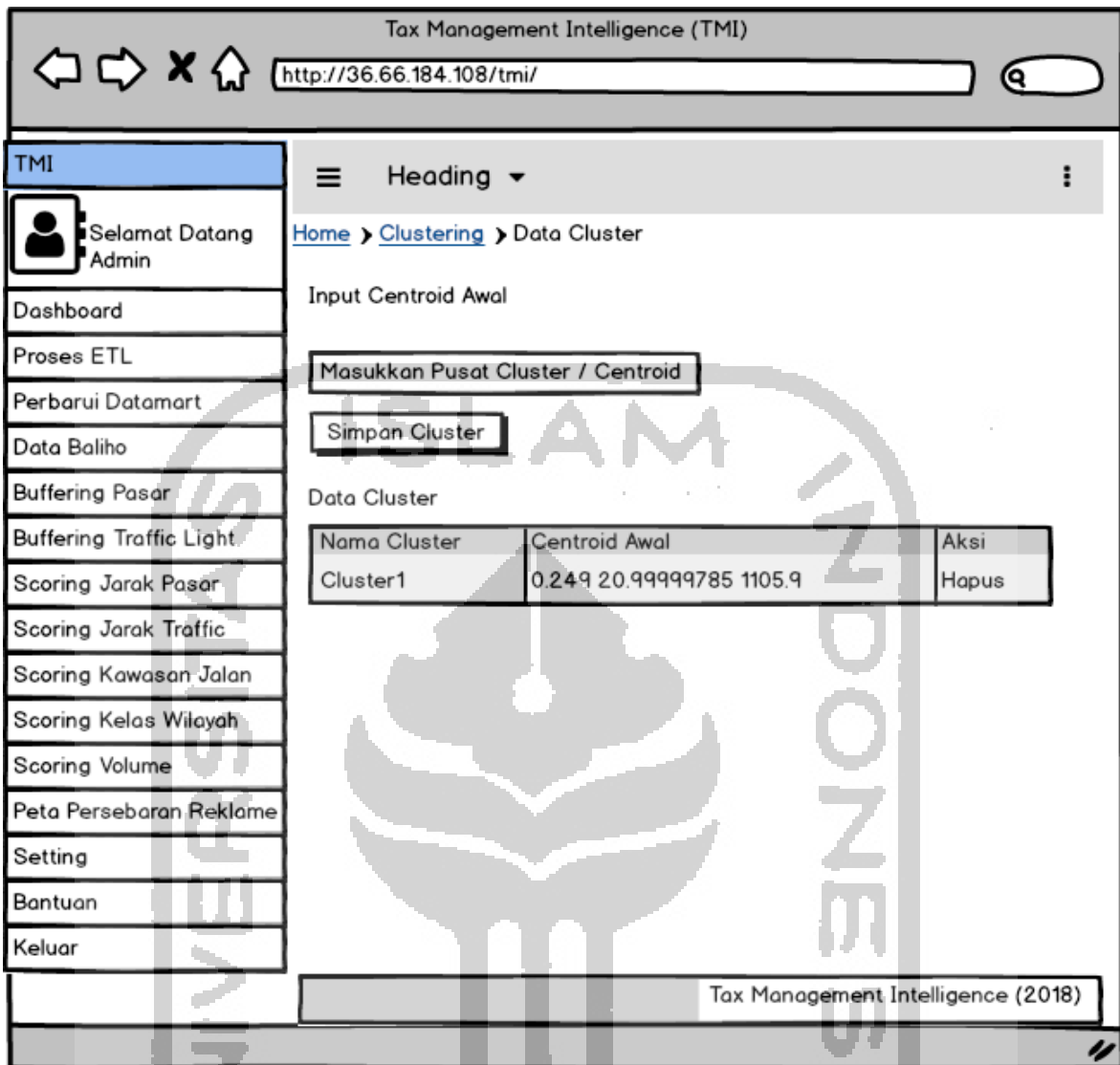
Rancangan halaman pengukuran jarak pasar pada aplikasi, terdiri atas dan *button* “clear data”, dan *button* “Proses Pemetaan Reklame dengan Volume Kendaraan. Rancangan halaman pengukuran volume kendaraan sebagaimana terlihat pada Gambar 4.14.



Gambar 4.14 Rancangan Halaman Pengukuran volume kendaraan

j. Rancangan halaman Data Centroid (Pusat Cluster)

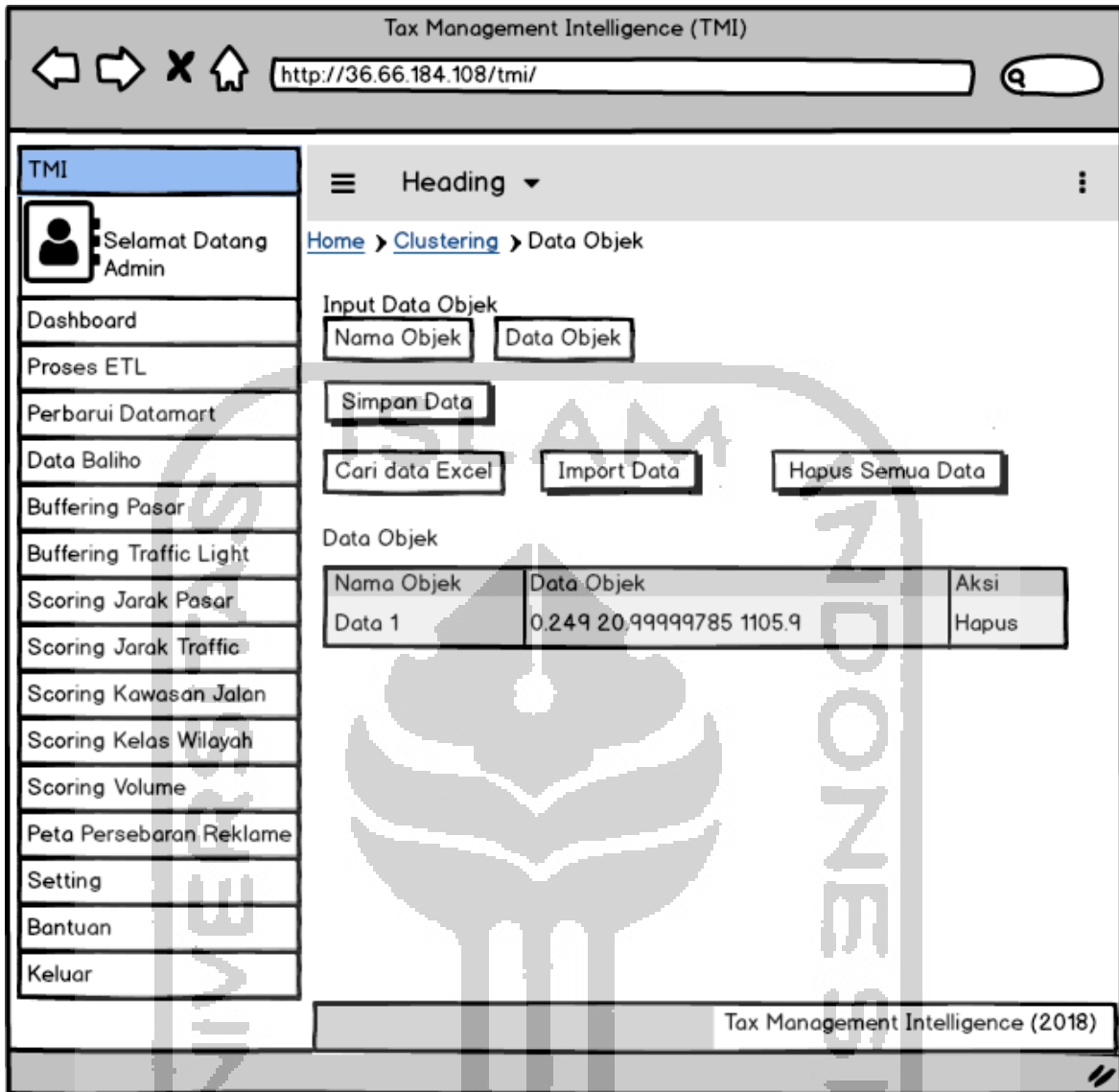
Rancangan halaman peta persebaran reklame pada aplikasi, terdiri atas *textbox* “Masukkan Centroid/Pusat Cluster Awal” dan *button* “Simpan Cluster” untuk melakukan proses. Rancangan halaman data *centroid* sebagaimana terlihat pada Gambar 4.15.



Gambar 4.15 Rancangan Data *Centroid*

k. Rancangan halaman Data Objek

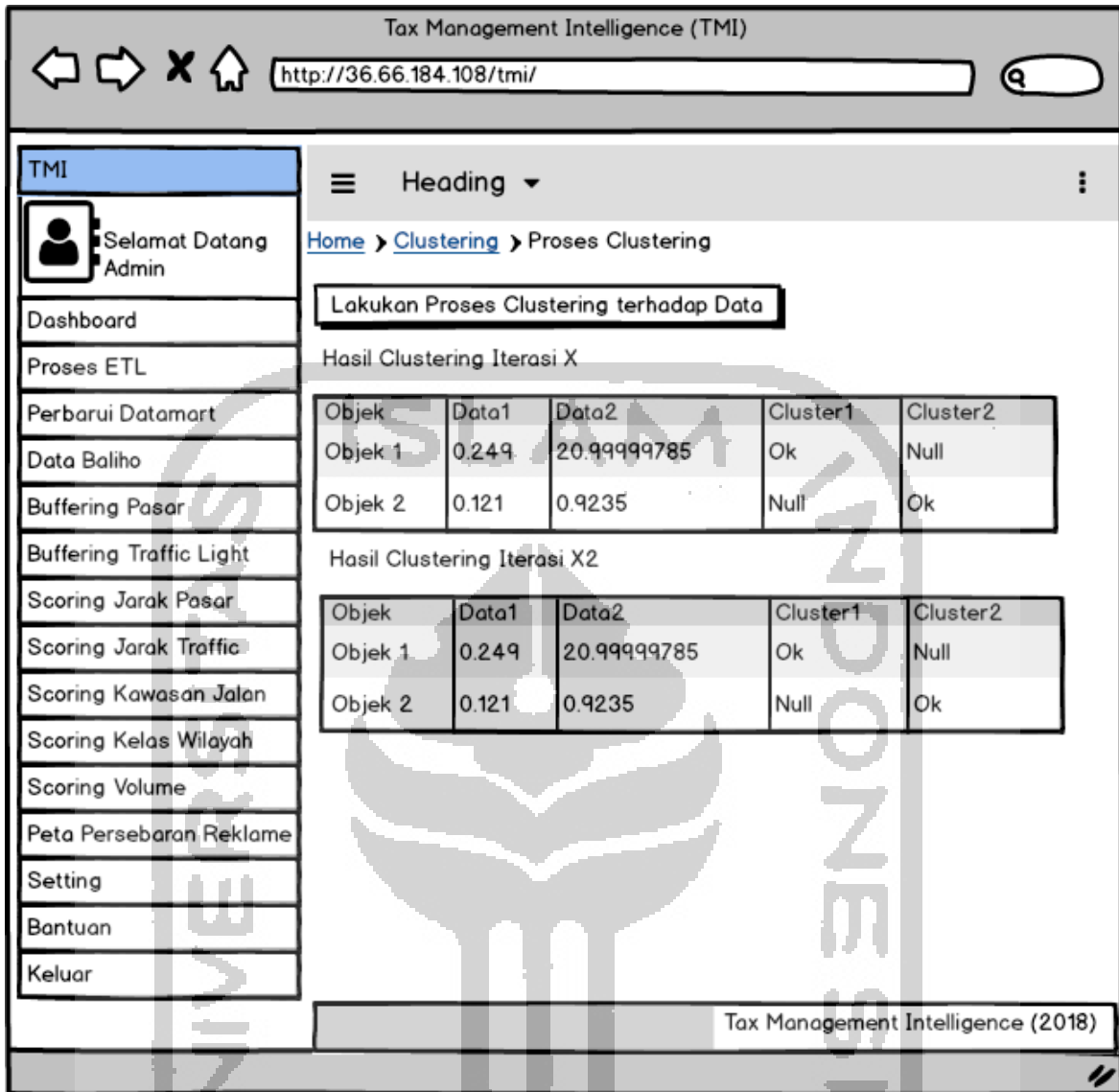
Rancangan halaman peta persebaran reklame pada aplikasi, terdiri atas *textbox* “Nama Objek dan Data Objek”, input file “Cari data Excel”, *button* “Simpan Data”, Button “Import Data Excel”, dan button “Hapus Semua Data”. Rancangan data objek sebagaimana terlihat pada Gambar 4.16.



Gambar 4.16 Rancangan Data Objek

1. Rancangan halaman Hasil *Clustering*

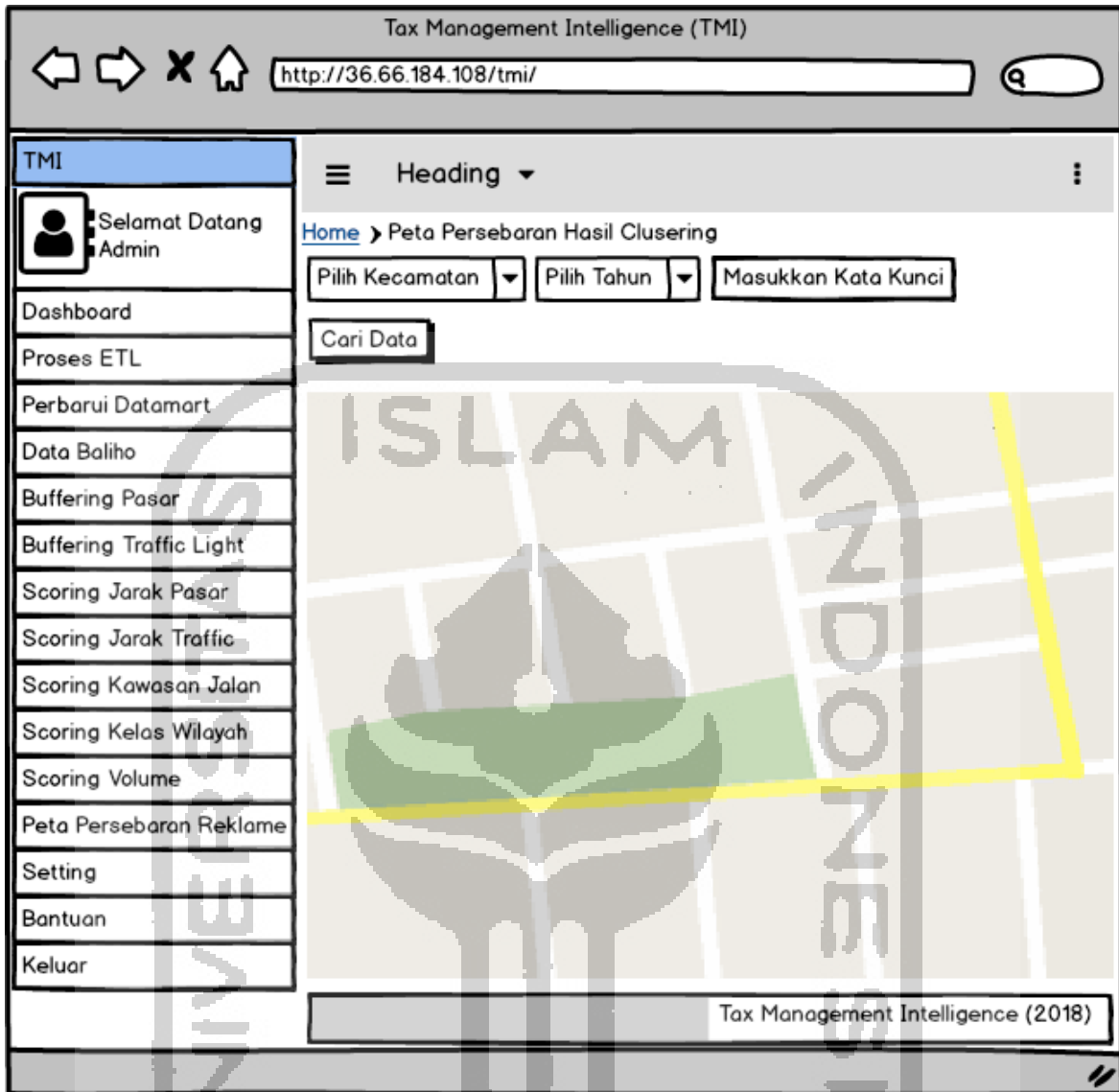
Rancangan halaman hasil *clustering* pada aplikasi, terdiri dari Button “Lakukan Proses *Clustering*” dan tabel hasil proses iterasi. Rancangan halaman Hasil *Clustering* sebagaimana terlihat pada Gambar 4.17.



Gambar 4.17 Rancangan Hasil *Clustering*

m. Rancangan halaman Peta Persebaran Hasil *Clustering*

Rancangan halaman peta persebaran hasil *clustering* pada aplikasi, terdiri atas combobox “pilih kecamatan”, combobox “pilih tahun”, textbox “kata kunci” dan button “cari data” untuk melakukan proses pencarian data, serta peta untuk menampilkan hasil dari proses pencarian data. Rancangan halaman peta persebaran hasil *clustering* sebagaimana terlihat pada Gambar 4.18.



Gambar 4.18 Rancangan Peta Persebaran Hasil *Clustering*

4.2 Pengembangan Sistem

Pengembangan sistem meliputi aktivitas pembuatan tampilan antarmuka sistem untuk dapat memvisualisasikan data dan informasi kepada pengguna, serta aktivitas pemrograman. Bahasa pemrograman yang dipakai dalam pengembangan sistem ini adalah bahasa pemrograman PHP, sedangkan metode yang digunakan adalah *waterfall method*. Alat yang digunakan dalam pengembangan sistem adalah sebagai berikut.

1. Perangkat komputer (PC) dengan spesifikasi: Processor Intel (R) Core (TM) i3 2365M CPU @1.40GHz (4 CPUs), Memory 2 GB, dan Monitor Diagonal screen size 14 inch wide.

2. Perangkat lunak MySQL Server sebagai sistem basisdata dan tempat menyimpan data SIG. Pemilihan MySQL dikarenakan kebutuhan kapasitas basisdata yang tidak terlalu besar.
3. Bahasa pemrograman PHP Bahasa Pemrograman yang digunakan dalam pengembangan sistem adalah bahasa pemrograman PHP dengan framework bootstrap untuk mengatur desain tampilan dari aplikasi.
4. Web Server Apache untuk menjalankan kode sumber PHP yang telah dibuat.
5. Web Browser untuk melihat hasil running kode sumber PHP.

4.3 Pengujian Fungsional Sistem

Aplikasi yang dikembangkan harus dipastikan bebas dari kesalahan karenanya perlu dilakukan proses pengujian fungsional terhadap sistem yang telah dikembangkan. Pengujian sistem dilakukan dengan metode *black box* dengan melibatkan petugas pengelola pajak reklame BPKD Kabupaten Pekalongan. Tabel 4.1 menunjukkan skenario pengujian dengan *Black Box Testing*.

Tabel 4.1. Skenario pengujian dengan *Black Box Testing*

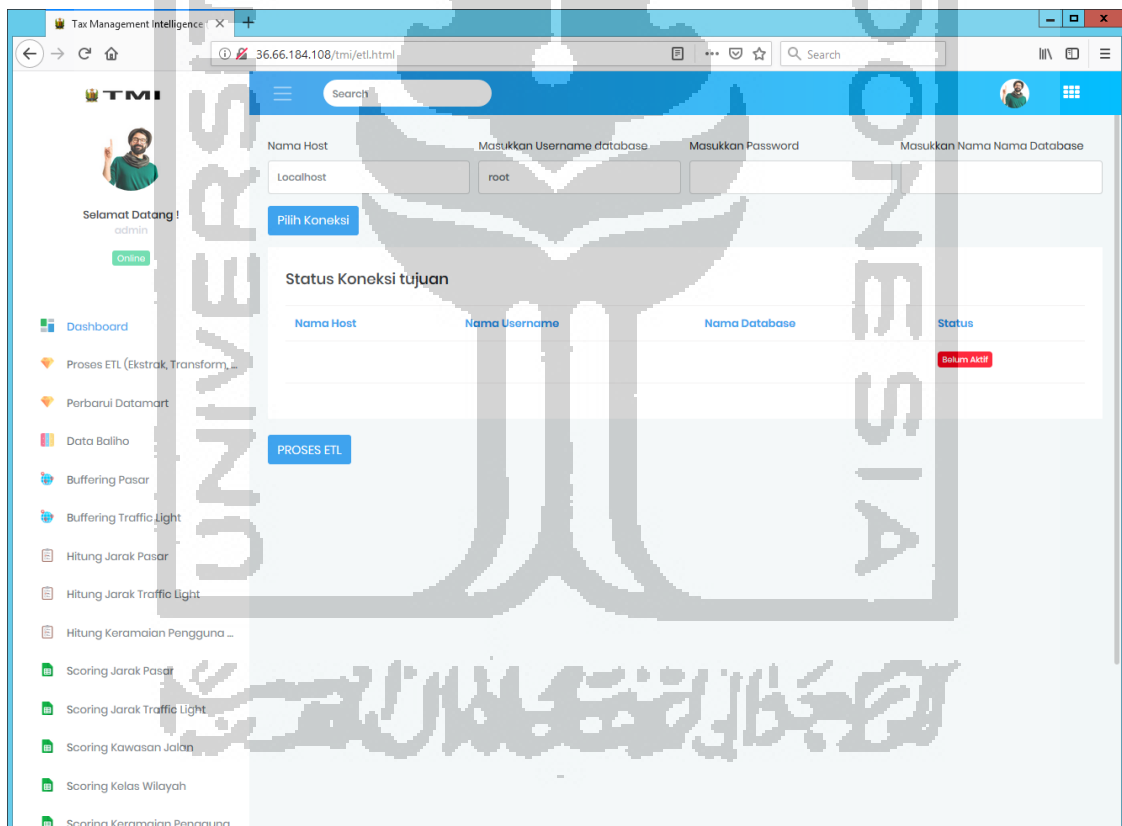
No	Skenario pengujian	Hasil yang diharapkan	Hasil pengujian
1	Menjalankan proses Ekstraksi Basisdata dengan menghubungkannya dengan basisdata sumber (basisdata SIAPADA) kemudian memasukkan informasi koneksi basisdata kemudian menekan <i>button</i> pilih koneksi	Sistem dihubungkan dengan basisdata SIAPADA menampilkan status aktif dan berubah menjadi warna hijau pada status koneksi. Ketika koneksi tidak berhasil, menampilkan <i>message box</i> "Koneksi anda salah"	Sesuai
2	Pengguna melakukan <i>login</i> pada sistem informasi sesuai dengan <i>username</i> yang sudah didaftarkan administrator	Sistem melakukan validasi terhadap proses login, untuk masuk pada halaman utama aplikasi.	Sesuai

No	Skenario pengujian	Hasil yang diharapkan	Hasil pengujian
3	Pengguna masuk ke dalam aplikasi, memasuki semua halaman, dan melihat beberapa <i>chart</i> di dalamnya.	Sistem menampilkan informasi sesuai dengan yang dipilih pengguna dalam bentuk visual (<i>chart</i>).	Sesuai
4	Pengguna memasuki menu-menu scoring yaitu dengan memilih menu scoring sesuai variabel yang dikehendaki	Sistem menampilkan hasil scoring terhadap variabel-variabel yang terpilih	Sesuai
5	Pengguna memasuki halaman peta persebaran reklame dengan memilih menu peta persebaran reklame	Sistem menampilkan titik-titik reklame yang berada pada wilayah Kabupaten Pekalongan	Sesuai
6	Pengguna memasuki halaman <i>Cluster K-means</i> dengan memasukkan jumlah <i>cluster</i>	Sistem melakukan proses penyimpanan data <i>cluster</i> kemudian menampilkan kembali hasil inputan	Sesuai
7	Pengguna memasuki halaman <i>Cluser K-means</i> dengan memasukkan data <i>cluster</i> dalam format excel	Sistem melakukan import data excel ke dalam suster dan menampilkan data hasil import	Sesuai
8	Pengguna memasuki halaman <i>Cluster K-means</i> dengan melakukan proses <i>Clustering</i>	Sistem melakukan proses <i>clustering</i> dan menampilkan hasil <i>clustering</i>	Sesuai

Skenario pengujian dengan *black box testing* digunakan untuk mengamati kesesuaian dari skenario pengujian dengan hasil yang diharapkan. Terlihat pada tabel 4.1, bahwa dari 8 (delapan) pengujian menunjukkan hasil yang sesuai.

4.4 Pengumpulan Data

Dalam penelitian ini penulis menggunakan data pembayaran pajak reklame tahun 2018. Data pembayaran pajak reklame tersebut diambil dari basisdata pada Aplikasi SIAPADA (Sistem Informasi Administrasi Pajak Daerah) yang berada pada BPKD Kabupaten Pekalongan. Aplikasi SIAPADA tidak hanya digunakan untuk administrasi pajak reklame saja melainkan digunakan untuk administrasi pajak lain seperti pajak air tanah, pajak sarang burung walet, pajak mineral bukan logam, pajak hiburan, pajak restoran, pajak hotel, pajak penerangan jalan, dan pajak parkir. Agar tidak terjadi *chaos* dalam pengolahan data, maka data-data tersebut perlu untuk difilter ataupun distandarisasi sehingga diperoleh data-data yang valid. Proses pengumpulan data dimulai dengan mengambil data yang berada pada basisdata SIAPADA, untuk proses pengambilan data dilakukan melalui sistem yang dibangun. Proses pengambilan data sebagaimana terlihat pada Gambar 4.19.



Gambar 4.19 Proses Ekstract Basisdata

Berdasarkan Gambar 4.19 dapat bahwa proses pengambilan data dilakukan dengan memasukkan nama host/ip, kemudian *username*, *password*, dan nama basisdata. Selanjutnya adalah melakukan proses dengan menekan tombol pilih koneksi. Apabila koneksi benar, maka status koneksi pada tabel status koneksi tujuan akan terisi dan *field*

status akan berubah warna menjadi hijau yang menunjukkan bahwa koneksi basisdata telah berhasil dilakukan, sebagaimana terlihat pada Gambar 4.20.

Nama Host	Nama Username	Nama Database	Status
localhost	root	siapada_2017	AKTIF

Gambar 4.20 Status Koneksi Tujuan

Setelah berhasil melakukan koneksi langkah yang selanjutnya adalah mengekstraksi data-data dari basisdata SIAPADA kedalam basisdata SIG. Data-data yang dimasukkan ke dalam basisdata SIG perlu difilter terlebih dahulu agar tidak tercampur dengan data-data yang lain. Gambar 4.21 menunjukkan potongan *script* untuk proses pengambilan data.

```

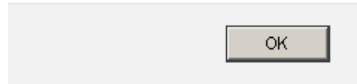
1 <?php
2 if (isset($_POST['simpan'])) {
3     session_start();
4     $db_tujuan = $_SESSION[database2] ;
5
6     $minta = mysql_query("select tahun from $db_tujuan.konfigurasi");
7     $data=mysql_fetch_array($minta);
8     $tahun=trim($data[tahun]);
9
10    mysql_query("delete from tabel_reklame where trim(tahun)='$tahun'");
11
12    mysql_query("insert into tabel_reklame (tahun,nomor_skp,npwpd>Nama Wp,alamat,lok_pasang,
13    judul,panjang,lebar,luas,jumlah,jumlah1,tgl_terima,`status`,jenis_pajak,bulan1,bulan2)
14
15    select (select tahun from $db_tujuan.konfigurasi) as tahun,a.no_skp as nomor_skp,a.no_wpr as npwpd,
16    b.nama_wpr as Nama WP,b.alamat, c.lok_pasang,c.judul,c.panjang,c.lebar,c.luas,c.jumlah,
17    a.jml_tetap as jumlah1,d.tgl_terima,if(ISNULL(d.tgl_terima),'BLM LUNAS','LUNAS') as STATUS,
18    jenis_pajak, a.bulan1, a.bulan2 from $db_tujuan.notad a
19    LEFT JOIN $db_tujuan.wpr b on a.no_wpr = b.no_wpr
20    LEFT JOIN $db_tujuan.sptd_reklame c on a.no_spt = c.no_spt
21    LEFT JOIN $db_tujuan.setorand d on a.no_skp = d.no_spt
22    where a.jenis_spt = 'Reklame'");
23
24    echo"<Script>alert('Database berhasil di Ekstrak ke database $db_tujuan');
25    window.history.go(-1)</script>";
26 }
27 ?>
28

```

Gambar 4.21 Potongan *Script* Pengambilan Data

Berdasarkan potongan *script* yang terlihat pada Gambar 4.21, baris ke-2 merupakan kondisi untuk mengecek variabel *simpan* dengan metode *post*. Baris ke-3 dan ke-4 merupakan perintah untuk menyimpan *session* basisdata. Baris ke-6 sampai dengan ke-10 merupakan perintah untuk menghapus data target. Pengambilan data ditunjukkan pada baris ke-12 sampai dengan baris ke-22. Setelah berhasil melakukan proses penyimpanan data maka akan muncul *alert/notifikasi* sebagaimana terlihat pada Gambar 4.22.

Database berhasil di Ekstrak ke database



Gambar 4.22 Alert Proses ETL berhasil dieksekusi

Gambar 4.22 merupakan notifikasi ketika proses ekstraksi basisdata dari basisdata SIAPADA ke basisdata SIG. Gambar 4.23 menunjukkan potongan data yang berhasil di *extract*.

tahun	nomor_sko	lok_pasang	judul	panjang	lebar	luas	jumlah	jumlah1	tgl_terima	status	jenis_pajak	bulan1	bulan2
2018	00001	PEREMPATAN SIBEDUG - KAJEN	AROMA	4	6	24	1	4276800.00	2018-01-10	LUNAS	Balho	2018-01-01	2018-12-31
2018	02784	JL. RAYA PANTURA - WIRADESA (DEKAT HONDA SURYA) PI	ROKOK GUIDANG GARAM	4	6	24	1	4276800.00	2018-11-09	LUNAS	Balho	2018-11-10	2019-11-09
2018	01885	JL. RAYA PERTIGAAN PODO - KEDUNGWUNI	PT. PISMA GAJAH PUTRA	4	6	24	1	3888000.00	2018-07-16	LUNAS	Balho	2018-07-01	2019-06-30
2018	01292	JL. A. YANI PEKALONGAN - WIRADESA	PRODUK DJARUM	10	5	50	1	8910000.00	2018-05-02	LUNAS	Balho	2018-05-15	2019-05-14
2018	01293	JL. RAYA PACAR - TIRTO	AROMA	4	6	24	1	4276800.00	2018-05-02	LUNAS	Balho	2018-05-15	2019-05-14
2018	02309	PERTIGAAN PODO - KEDUNGWUNI	PROMOSI AYAM GEMES	3	5	14	1	2700000.00	2018-09-03	LUNAS	Balho	2018-09-01	2019-10-31
2018	00398	JL. P. DIPONEGORO SIBEDUG - KAJEN	ESK	4	6	24	1	3888000.00	2018-02-07	LUNAS	Balho	2018-02-14	2019-02-13
2018	02450	JL. RAYA ROWOKEMBU - WONOPRINGGO	SAPPHIRE RESIDENCE	4	6	24	2	7776000.00	2018-11-07	LUNAS	Balho	2018-09-01	2019-08-31
2018	00932	JL. DIPONEGORO SEBELAH ALFAMART KAJEN - PEKALONGAN	INDOSAT	4	6	24	1	3888000.00	2018-03-16	LUNAS	Balho	2018-01-01	2018-12-31
2018	00933	JL. RAYA ROWOKEMBU - WONOPRINGGO	INDOSAT	4	6	24	1	3888000.00	2018-03-16	LUNAS	Balho	2018-01-01	2018-12-31
2018	01978	JL. RAYA PEREMPATAN SIBEDUG - KAJEN	PRODUK PT. DJARUM	4	6	24	1	4276800.00	2018-07-20	LUNAS	Balho	2018-08-01	2019-07-31
2018	01979	JL. RAYA PERTIGAAN PASAR BOJONG	PRODUK PT. DJARUM	4	6	24	1	4276800.00	2018-07-20	LUNAS	Balho	2018-08-01	2019-07-31
2018	01981	JL. DIPONEGORO (PASAR KAJEN)	PRODUK GUIDANG GARAM	4	6	24	2	8553600.00	2018-07-20	LUNAS	Balho	2018-08-01	2019-07-31
2018	01982	JL. RAYA SRAGI	PRODUK GUIDANG GARAM	4	6	24	2	8553600.00	2018-07-20	LUNAS	Balho	2018-08-01	2019-07-31
2018	00959	JL. RAYA BEBEKAN SEBELAH JEMBATAN - KEDUNGWUNI	PRODUK SAMPOERNA	4	6	24	1	4276800.00	2018-04-03	LUNAS	Balho	2018-04-01	2019-03-31
2018	00960	JL. RAYA ROWOKEMBU DEPAN PASAR WONOPRINGGO	YAMAHA AGUNG MOTOR	4	6	24	1	3888000.00	2018-04-16	LUNAS	Balho	2018-04-01	2019-03-31
2018	00961	JL. RAYA BOJONGMINGGIR DEPAN POLSEK BOJONG	YAMAHA AGUNG MOTOR	4	6	24	1	3888000.00	2018-04-16	LUNAS	Balho	2018-04-01	2019-03-31
2018	00962	JL. RAYA PEREMPATAN SIBEDUG KAJEN	YAMAHA AGUNG MOTOR	4	6	19	1	3118500.00	2018-04-11	LUNAS	Balho	2018-04-01	2019-03-31
2018	00984	JL. RAYA SEBELAH PASAR BOJONG - PEKALONGAN	PRODUK SAMPOERNA	4	6	24	1	4276800.00	2018-04-09	LUNAS	Balho	2018-04-01	2019-03-31
2018	02514	JL. RAYA DEPAN PASAR WIRADESA - PEKALONGAN	HONDA ASTRA	8	4	32	1	5184000.00	2018-09-21	LUNAS	Balho	2018-08-01	2019-07-31
2018	01512	JL. RAYA KERTIJAYAN - BUARAN	ROKOK AROMA	4	6	24	1	4276800.00	2018-05-17	LUNAS	Balho	2018-05-01	2019-04-30
2018	02516	JL. RAYA PERTIGAAN PACAR - TIRTO PEKALONGAN	XL AXIATA	4	6	24	1	3888000.00	2018-09-21	LUNAS	Balho	2018-10-23	2019-10-22
2018	02517	JL. RAYA BEBEKAN - KEDUNGWUNI	XL AXIATA	4	6	24	1	3888000.00	2018-09-21	LUNAS	Balho	2018-10-23	2019-10-22
2018	01521	JL. RAYA AMBOKEBANG - KEDUNGWUNI	PRODUK GUIDANG GARAM	4	6	24	1	4276800.00	2018-05-21	LUNAS	Balho	2018-05-20	2019-05-19
2018	01532	JL. RAYA PERTIGAAN PODO - KEDUNGWUNI	ROKOK AROMA	4	6	24	1	4276800.00	2018-05-24	LUNAS	Balho	2018-06-01	2019-05-31
2018	01013	JL. RAYA KETTITANGDUL - BOJONG	SAPPHIRE RESIDENCE	4	6	24	2	5760000.00	2018-04-10	LUNAS	Balho	2018-01-01	2018-12-31
2018	00624	DEPAN PASAR WIRADESA	PRODUK LION WINGS	6	3	18	1	2916000.00	2018-02-27	LUNAS	Balho	2018-02-01	2019-01-31
2018	01558	JL. RAYA PERTIGAAN PODO - KEDUNGWUNI	TELKOMSEL	5	10	50	1	8100000.00	2018-06-07	LUNAS	Balho	2018-04-01	2019-03-31
2018	01562	JL. A. YANI WIRADESA	AVITEX	10	5	50	2	16200000.00	2018-06-07	LUNAS	Balho	2018-06-01	2019-05-31

Gambar 4.23 Potongan data yang berhasil di *extract* ke dalam basisdata SIG

Terlihat pada gambar 4.23, data-data yang berhasil di *extract* adalah data reklame. Setelah proses pengambilan selesai yang selanjutnya adalah melakukan filter terhadap data-data yang akan dijadikan sebagai bahan analisis. Data yang digunakan pada penelitian ini adalah data reklame dengan jenis baliho yang mempunyai masa pajak lebih dari satu tahun. Pajak reklame yang masa pajaknya kurang dari satu tahun bisa jadi merupakan reklame *insidentil*, sehingga tidak sesuai untuk dimasukkan dalam penelitian. Gambar 4.24 menunjukkan potongan *script* dari proses pengambilan data baliho yang masa pajaknya lebih dari satu tahun.


```

1 <?php
2 if (isset($_POST['simpan'])) {
3     session_start();
4     $db_tujuan = $_SESSION[database2] ;
5
6     mysql_query("insert into tabel_baliho (tahun,
7 nomor_skp,npwpd>Nama_Wp,alamat,lok_pasang,judul,panjang,lebar,luas,jumlah,jumlah1,
8 tgl_terima,`status`,`jenis_pajak,bulan1,bulan2)
9
10 SELECT tahun,nomor_skp,npwpd>Nama_Wp,alamat,lok_pasang,judul,panjang,lebar,luas,
11 jumlah,jumlah1,tgl_terima,status,`jenis_pajak,bulan1,bulan2 FROM tabel_reklame
12 WHERE jenis_pajak LIKE '%baliho%'
13 and DATEDIFF(bulan2,bulan1) > 360 and `status`='lunas'");
14
15 echo"<Script>alert('Data Mart Baliho Berhasil diperbarui');window.history.go(-1)</script>";
16 }
17 ?>

```

Gambar 4.24 Potongan *Script* dari Proses Pengambilan Data Baliho yang Masa Pajaknya Lebih dari Satu Tahun

Berdasarkan potongan *script* yang terlihat pada Gambar 4.24, baris ke-2 merupakan kondisi untuk mengecek variabel *simpan* dengan metode *post*. Baris ke-3 dan ke-4 merupakan perintah untuk menyimpan *session* basisdata. Baris ke-6 sampai dengan ke-13 merupakan perintah untuk pengambilan data baliho. *Filtering* data yang masa pajaknya lebih dari satu tahun ditunjukkan pada baris ke-13.

4.5 Penyeragaman Data

Penyeragaman data dilakukan dengan memberikan kode untuk proses analisis. Kode disini adalah kode-kode yang tidak ada dalam data yang telah diambil dari basisdata pada aplikasi SIAPADA. Kode yang tersebut adalah kode wilayah, kode kawasan dan kode kecamatan. Pemberian kode dilakukan dengan menggunakan *query* yang ditanamkan pada sistem. Proses pemberian kode dimulai dengan mempelajari cirikhas dari inputan pada aplikasi SIAPADA, sebagai contoh : pada aplikasi SIAPADA ada *field* yang namanya lokasi, namun pada *field* tersebut masih belum standar dalam menyebutkan nama kecamatan. Penyebutan nama kecamatan perlu dikonversi menjadi kode agar standar. Disini peneliti mengamati beberapa penyebutan nama kecamatan dalam *field* lokasi yang tidak standar seperti : Wiradesa dengan wides, Karanganyar dengan kringanyar, kranyar dan lain-lain. Dari hasil pengamatan tersebut kemudian dipetakan dan dituangkan dalam bahasa pemrograman. Gambar 4.25 menunjukkan potongan *script* untuk penyeragaman *field* lokasi kecamatan.

```

1 <?php
2 if (isset($_POST['simpan'])) {
3     mysql_query("update tabel_baliho set KD_KECAMATAN='110' WHERE lok_pasang LIKE '%BOJONG%'
4     OR lok_pasang LIKE '%WANGANDOWO%'");
5     mysql_query("update tabel_baliho set KD_KECAMATAN='140' WHERE lok_pasang LIKE '%BUARAN%'
6     OR lok_pasang LIKE '%PEKAJANGAN%'");
7     mysql_query("update tabel_baliho set KD_KECAMATAN='060' WHERE lok_pasang LIKE '%DORO%'");
8     mysql_query("update tabel_baliho set KD_KECAMATAN='080' WHERE lok_pasang LIKE '%KAJEN%'");
9     mysql_query("update tabel_baliho set KD_KECAMATAN='010' WHERE lok_pasang LIKE '%KANDANGSERANG%'");
10    mysql_query("update tabel_baliho set KD_KECAMATAN='070' WHERE lok_pasang LIKE '%KARANG ANYAR%' or
11    lok_pasang LIKE '%KARANGANYAR%' or lok_pasang LIKE '%KR. ANYAR%' or lok_pasang LIKE '%KR.ANYAR%'
12    or lok_pasang LIKE '%KARANGANYA%' ");
13    mysql_query("update tabel_baliho set KD_KECAMATAN='131' WHERE lok_pasang LIKE '%KARANGDADAP%'");
14    mysql_query("update tabel_baliho set KD_KECAMATAN='130' WHERE lok_pasang LIKE '%KEDUNGWUNI%'
15    OR lok_pasang LIKE '%KDW%' OR lok_pasang LIKE '%KD.WUNI%' OR lok_pasang LIKE '%PODO%' OR
16    lok_pasang LIKE '%SUROBAYAN%'");
17    mysql_query("update tabel_baliho set KD_KECAMATAN='090' WHERE lok_pasang LIKE '%KESESI%'");
18    mysql_query("update tabel_baliho set KD_KECAMATAN='030' WHERE lok_pasang LIKE '%LEBAKBARANG%'");
19    mysql_query("update tabel_baliho set KD_KECAMATAN='020' WHERE lok_pasang LIKE '%PANINGGARAN%'");
20    mysql_query("update tabel_baliho set KD_KECAMATAN='040' WHERE lok_pasang LIKE '%PETUNG KRIYONO%'
21    or lok_pasang LIKE '%PETUNGKRIYONO%'");
22    mysql_query("update tabel_baliho set KD_KECAMATAN='101' WHERE lok_pasang LIKE '%SIWALAN%'");
23    mysql_query("update tabel_baliho set KD_KECAMATAN='100' WHERE lok_pasang LIKE '%SRAGI%'");
24    mysql_query("update tabel_baliho set KD_KECAMATAN='050' WHERE lok_pasang LIKE '%TALUN%'");
25    mysql_query("update tabel_baliho set KD_KECAMATAN='150' WHERE lok_pasang LIKE '%TIRTO%'");
26    mysql_query("update tabel_baliho set KD_KECAMATAN='160' WHERE lok_pasang LIKE '%WIRADESA%'");
27    mysql_query("update tabel_baliho set KD_KECAMATAN='161' WHERE lok_pasang LIKE '%WONOKERTO%'");
28    mysql_query("update tabel_baliho set KD_KECAMATAN='120' WHERE lok_pasang LIKE '%WONOPRINGGO%'
29    or lok_pasang LIKE '%WN.PRINGGO%'");
30    echo "<Script>alert('Lokasi Kecamatan Baliho Berhasil diperbarui');window.history.go(-1)</script>";
31 }
32 ?>

```

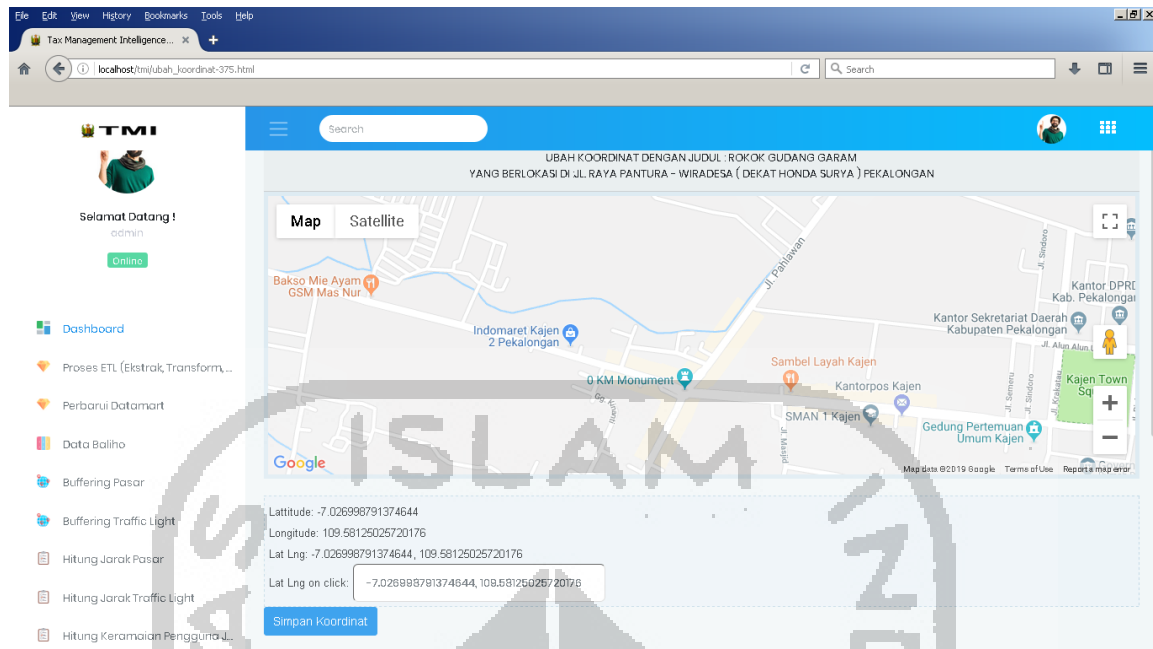
Gambar 4.25 Potongan Script Penyeragaman (Pengkodean) Kecamatan

Berdasarkan potongan script yang terlihat pada Gambar 4.25, baris ke-2 merupakan kondisi untuk mengecek variabel *simpan* dengan metode *post*. Baris ke-3 sampai dengan baris ke-29 merupakan perintah untuk pengkodean data kecamatan. Baris ke-30 merupakan perintah untuk menampilkan notifikasi/*alert*. Setelah data berhasil *diextract*, *difilter*, dan diseragamkan, maka data reklame dapat ditampilkan ke dalam sistem. Gambar 4.26 menunjukkan tampilan halaman baliho.

id	Nama WP	Alamat	Lokasi Pasang	Judul	Jumlah Pajak
375	PT. MANTRA GUNA POLAMANDIRI	PRM BD JL. ANGGREK 1 NO 335 BANYUMANIK SEMARANG	JL RAYA PANTURA - WIRADESA (DEKAT HONDA SURYA) PEKALONGAN	ROKOK GUDANG GARAM	4.276.800,00
370	SAPPHIRE RESIDENCE	JL. P. MANDUROREJO - KAJEN	JL RAYA ROWOKEMBU - WONOPRINGGO	SAPPHIRE RESIDENCE	7.776.000,00
374	MOCH MACHRUS (AL FAIRUS)	SAMPANGAN Gg.6 NO.168 KAUMAN PEKALONGAN TIMUR	JL RAYA BEBEKAN - KEDUNGWUNI	PROMOSI UMROH	3.888.000,00
371	CV. LESTARI INDAH ABADI / ASTRA	JL. CILIWUNG 9 NO. 22 SEMARANG	JL RAYA DEPAN PASAR WIRADESA - PEKALONGAN	HONDA ASTRA	5.184.000,00
372	CV. WICAKSANA ADVERTISING	JL SOEKARNO HATTA NO.19 - 20 SEMARANG	JL RAYA PERTIGAAN PACAR - TIRTO PEKALONGAN	XL AXIATA	3.888.000,00
373	CV. WICAKSANA ADVERTISING	JL SOEKARNO HATTA NO.19 - 20 SEMARANG	JL RAYA BEBEKAN - KEDUNGWUNI	XL AXIATA	3.888.000,00
367	PR. SUKUN KUDUS	JL. RAYA BESITO GEBOG KUDUS - JAWA TENGAH	JL RAYA PACAR - TIRTO PEKALONGAN	PRODUK ROKOK SUKUN	17.820.000,00

Gambar 4.26 Tampilan Halaman Baliho

Penelitian ini membutuhkan data spasial untuk proses analisis, sehingga diperlukan data koordinat untuk melakukan perhitungan jarak antar variabel-variabel yang diperlukan untuk keperluan analisis. Pengambilan koordinat dilakukan dengan menggunakan layanan *google maps*. *Google maps* merupakan aplikasi yang dibuat oleh *Google Inc.* untuk memberikan kemudahan dalam membangun aplikasi yang menggunakan peta digital berbasis web sesuai yang diinginkan oleh user. Aplikasi dari Google ini sering disebut dengan istilah API (*Application Programming Interface*). Teknologi API bertujuan menyisipkan segala macam fasilitas dan teknologi yang ada dalam *Google Maps* ke dalam dokumen web (HTML). Penggunaan API pada dokumen web harus memperhatikan *API key*. *API key* merupakan penanda yang wajib diikutsertakan apabila teknologi ini digabungkan dengan sistem web. *API key* yang digunakan untuk web yang berbeda akan menghasilkan nilai yang berbeda juga. Berikut tampilan dari proses pengambilan koordinat.



Gambar 4.27 Proses Pengambilan Koordinat

Berdasarkan Gambar 4.27 merupakan proses pengambilan koordinat untuk reklame dengan judul Rokok Gudang Garam. Proses pengambilan koordinat dilakukan melalui pencarian titik koordinat berdasarkan keterangan lokasi yang berada pada basisdata SIG, serta sesuai dengan keterangan dari petugas pemungut pajak yang menangani reklame di wilayah tersebut. Potongan *script* dari proses pengambilan koordinat sebagaimana terlihat pada Gambar 4.28.

```

10 <div class="main">
11 <div id="mapa"></div>
12 <form method="post" action="module/mod_baliho/aksi.php">
13 <input type="hidden" name="id" value="<?php echo $id;?>">
14 <div class="eventtext">
15 <div>Latitude: <span id="latspan"></span></div>
16 <div>Longitude: <span id="lngspan"></span></div>
17 <div>Lat Lng: <span id="latlong"></span></div>
18 <div>Lat Lng on click:
19 <input type="text" name="koordinat" id="latlongclicked" style="width:300px; border:1px inset gray;">
20 </div>
21 </div>
22 <div align="left"><button type="submit" name="urut" class="btn btn-primary">Simpan Koordinat</button></div>
23 </form>
24 </div>
25 <script type="text/javascript">
26 if (@BrowserIsCompatible())
27 {
28 map = new GMap2(document.getElementById("mapa"));
29 map.addControl(new GLargeMapControl());
30 map.addControl(new GMapTypeControl(3));
31 map.setCenter(new GLatLng(-7.029593, 109.614967), 11,0);
32
33 GEvent.addListener(map,"mousemove",function(point)
34 {
35 document.getElementById('latspan').innerHTML = point.lat()
36 document.getElementById('lngspan').innerHTML = point.lng()
37 document.getElementById('latlong').innerHTML = point.lat() + ', ' + point.lng()
38 });
39 GEvent.addListener(map,"click",function(overlay,point)
40 {
41 document.getElementById('latlongclicked').value = point.lat() + ', ' + point.lng()
42 });
43 }
44 </script>

```

Gambar 4.28 Potongan Script Proses Pengambilan Koordinat

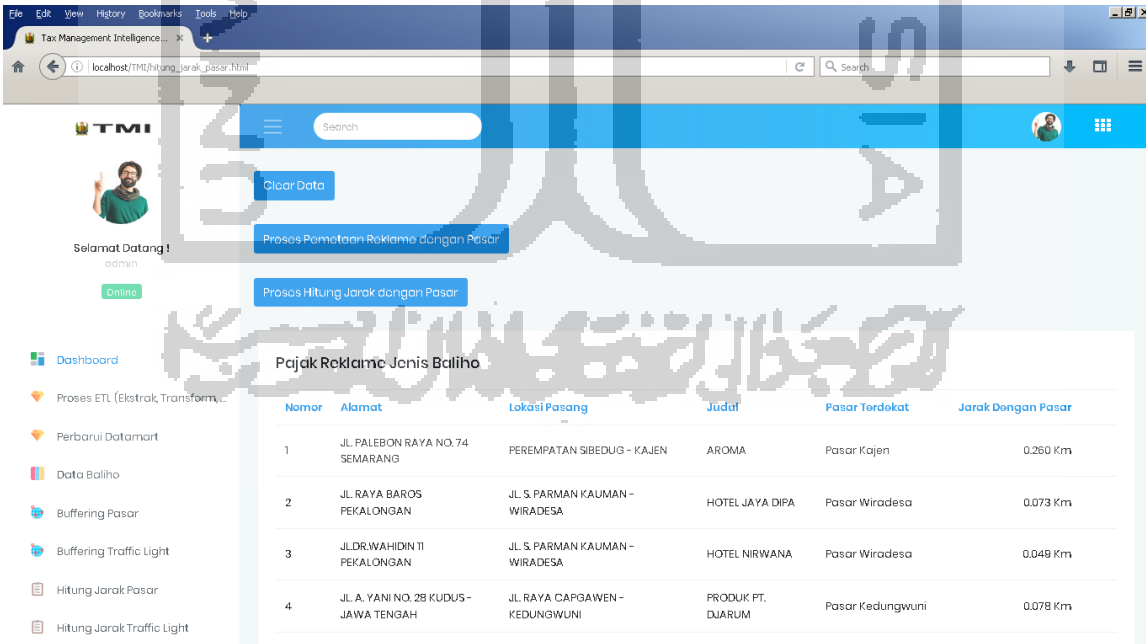
Berdasarkan potongan script yang terlihat pada Gambar 4.28, baris ke-10 sampai dengan baris ke-25 merupakan form untuk eksekusi penyimpanan koordinat. Baris ke-26 sampai dengan baris ke-44 merupakan javascript untuk menampilkan peta wilayah Kabupaten Pekalongan untuk memilih lokasi reklame.

4.6 Pengukuran Variabel

Pengukuran variabel adalah proses pemberian nilai atau atribut pada suatu objek. Variabel yang digunakan pada penelitian ini adalah variabel jarak dengan pasar, jarak dengan *traffic light*, dan volume kendaraan. Berikut penjelasan dari pengukuran untuk masing-masing variabel :

4.6.1 Pengukuran variabel jarak dengan pasar

Pengukuran jarak dengan pasar dimulai dengan memetakan pasar yang terdekat dari reklame yang akan diukur. Proses pemetaan dilakukan menggunakan sistem yang telah dibangun, kemudian mengukur jarak pasar terdekat dengan reklame dengan mengukur jarak antar titik koordinat yang telah diinput kedalam sistem. Proses pengukuran variabel jarak dengan pasar sebagaimana terlihat pada Gambar 4.29.



Nomor	Alamat	Lokasi Pasang	Judul	Pasar Terdekat	Jarak Dengan Pasar
1	JL. PALEBON RAYA NO. 74 SEMARANG	PEREMPATAN SIBEDUG - KAJEN	AROMA	Pasar Kajen	0.260 Km
2	JL. RAYA BAROS PEKALONGAN	JL. S. PARMAN KAUMAN - WIRADESA	HOTEL JAYA DIPA	Pasar Wiradesa	0.073 Km
3	JL. DR. WAHIDINI PEKALONGAN	JL. S. PARMAN KAUMAN - WIRADESA	HOTEL NIRWANA	Pasar Wiradesa	0.048 Km
4	JL. A. YANI NO. 28 KUDUS - JAWA TENGAH	JL. RAYA CAPGAWEN - KEDUNGWUNI	PRODUK PT. DJARUM	Pasar Kedungwuni	0.078 Km

Gambar 4.29 Tampilan pengukuran variabel jarak dengan pasar

Berdasarkan Gambar 4.29 terlihat ada 3 *button*, yaitu *button* “Clear Data”, *button* “Proses Pemetaan Reklame dengan Pasar”, dan *button* “Proses hitung Jarak dengan Pasar”. *Button* “Clear Data” digunakan untuk mengosongkan data Jarak Dengan Pasar. *Button* “Proses Pemetaan Reklame dengan Pasar” digunakan untuk memetakan reklame dengan pasar terdekat. Pada proses pemetaan ini ada proses manualnya karena ada beberapa kecamatan yang tidak terdapat pasarnya. Reklame yang berada pada kecamatan yang tidak ada pasarnya, dipetakan secara manual dengan pasar terdekat melalui *update field* *id_pasar* pada tabel *relasi_pasar*. Potongan *script* pemetaan reklame terhadap pasar sebagaimana terlihat pada Gambar 4.30.

```

1 <?php
2 if(isset($_POST['urut'])){
3     mysql_query("delete from relasi_pasar");
4     mysql_query("insert into relasi_pasar
5     select
6     id as id_baliho,
7     case KD_KECAMATAN
8     when 080 then 3 #1Kajen dengan pasar umum kajen
9     when 090 then 6 #2kesesi dengan pasar umum kesesi
10    when 110 then 7 #3Bojong dengan pasar umum bojong
11    when 120 then 8 #4wonopringgo dengan pasar umum wonopringgo
12    when 160 then 9 #5Wiradesa dengan pasar umum wiradesa
13    when 070 then 11 #6karanganyar dengan pasar umum karanganyar
14    when 060 then 12 #7doro dengan pasar umum doro
15    when 130 then 13 #8kedungwuni dengan pasar umum kedungwuni
16    when 140 then 15 #9Buaran dengan pasar umum buaran
17    when 150 then 16 #10Tirto dengan pasar umum tirto
18    when 100 then 20 #11Sragi dengan pasar umum sragi
19    when 020 then 2 #12Panninggaran dengan pasar desa Panninggaran
20    when 030 then 11 #13Lebakbarang dengan pasar Umum Karanganyar
21    when 010 then 1 #14Kandangserang dengan pasar Desa Kandangserang
22    when 131 then 13 #15Karangdadap dengan pasar Umum Kedungwuni
23    when 101 then 20 #16Siwalan dengan pasar umum sragi
24    when 040 then 12 #17Petungkriyono dengan pasar umum doro
25    when 050 then 12 #18Talun dengan pasar umum doro
26    when 161 then 22 #19 Wonokerto dengan pasar desa Api - Api Wonokerto
27    ELSE 0 END AS id_pasar
28    from tabel_baliho where tahun=2018");
29    echo"<script>alert('Pemetaan Lokasi Reklame dengan Pasar Berhasil diperbarui');window.history.go(-1)</script>";
30 }
31
32 ?>

```

Gambar 4.30 Potongan *Script* Pemetaan Reklame dengan Pasar

Berdasarkan potongan *script* yang terlihat pada Gambar 4.30, baris ke-2 merupakan kondisi untuk mengecek variabel *urut* dengan metode *post*. Baris ke-3 merupakan perintah untuk menghapus data tabel *relasi_pasar*. Baris ke-4 sampai dengan baris ke-28 adalah perintah untuk memetakan *id* reklame dengan *id_pasar* berdasarkan kecamatan dan baris ke-29 merupakan perintah untuk menampilkan pesan, jika perintah berhasil dilakukan. Setelah semua reklame berhasil dipetakan dengan pasar terdekat, maka langkah selanjutnya adalah menghitung jarak reklame dengan pasar terdekat. Disini penulis telah membuat sistem yang mampu menghitung jarak pasar dengan reklame berdasarkan titik koordinat pada masing- masing variabel. Potongan *script* dari pengukuran jarak reklame dengan pasar sebagaimana terlihat pada Gambar 4.31.

```

1 <?php
2 if(isset($_POST['urut'])){
3 function rad($x){ return $x * M_PI / 180; }
4 function distHaversine($coord_a, $coord_b){
5     # jarak kilometer dimensi (mean radius) bumi
6     $R = 6371;
7     $coord_a = explode(" ", $coord_a);
8     $coord_b = explode(" ", $coord_b);
9     $dLat = rad(($coord_b[0]) - ($coord_a[0]));
10    $dLong = rad(($coord_b[1]) - ($coord_a[1]));
11    $a = sin($dLat/2) * sin($dLat/2) + cos(rad($coord_a[0])) * cos(rad($coord_b[0])) * sin($dLong/2) * sin($dLong/2);
12    $c = 2 * atan2(sqrt($a), sqrt(1-$a));
13    $d = $R * $c;
14    # hasil akhir dalam satuan kilometer
15    return number_format($d, 3, '.', '');
16 }
17
18 mysql_query("delete from jarak_pasar");
19
20 $minta= mysql_query("select a.id as kode, concat(a.lat,a.lng) as koordinat_reklame, concat(c.lat,c.lng) as koordinat_pasar
21 from tabel_baliho a join relasi_pasar b on a.id=b.id_baliho join ref_pasar c on b.id_pasar=c.id ");
22
23 while ($data=mysql_fetch_array($minta)){
24     $id=$data[kode];
25     $a=$data[koordinat_reklame];
26     $b=$data[koordinat_pasar];
27     $hasil = distHaversine($a, $b);
28     mysql_query("insert into jarak_pasar(id_baliho, jarak) values ($id,$hasil)");
29 }
30 echo"<script>alert('Penetaan Lokasi Reklame dengan Pasar Berhasil diperbarui');window.history.go(-1)</script>";
31 }
32 ?>

```

Gambar 4.31 Potongan Script Pengukuran Jarak Reklame dengan Pasar

Berdasarkan Gambar 4.31, pengukuran jarak reklame dengan pasar digunakan formula *haversine*. Baris ke-3 sampai dengan 16 menunjukkan fungsi dari formula *haversine*. Formula *haversine* ini akan menghasilkan jarak terpendek antara dua titik yang diambil dari garis bujur (*longitude*) dan garis lintang (*latitude*). Formula ini pertama kali ditemukan oleh Jamez Andrew di tahun 1805, dan digunakan pertama kali oleh Josef de Mendoza y Ríos di tahun 1801. *Haversine* merupakan formula yang populer digunakan untuk menemukan jarak antara 2 titik. Tetapi belum ada penelitian yang membuktikan dari sekian banyak formula untuk mencari jarak mana yang “terbaik”. Berikut penjelasan dari formula *haversine*.

$$\Delta lat = lat2 - lat1$$

$$\Delta long = long2 - long1$$

$$a = \sin^2(\Delta lat/2) + \cos(lat1) * \cos(lat2) * \sin^2(\Delta long/2) \dots\dots\dots (4.1)$$

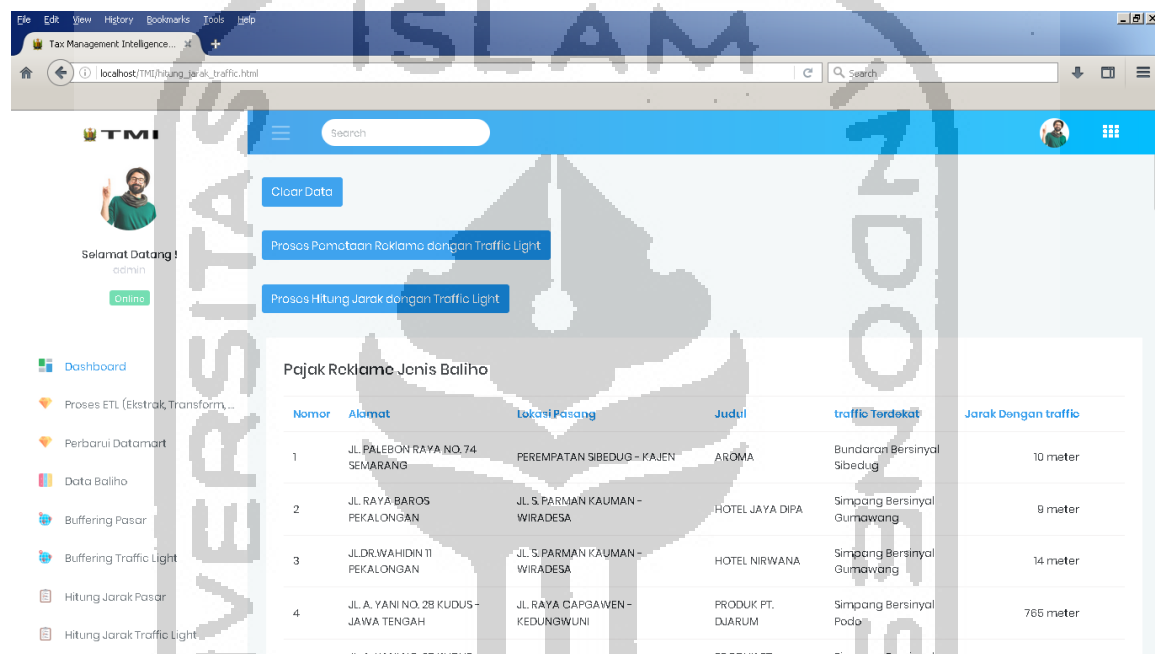
$$c = 2 \operatorname{atan2}(\sqrt{a}, \sqrt{1-a})$$

$$d = R * c$$

Dimana, Δlat adalah besaran perubahan *latitude*, $\Delta long$ adalah besaran perubahan *longitude*, c adalah kalkulasi perpotongan sumbu, R adalah radius bumi (radius = 6371 km). dan d adalah jarak.

4.6.2 Pengukuran variabel jarak dengan *traffic light*

Seperti halnya pengukuran variabel jarak dengan pasar, pengukuran variabel jarak dengan *traffic light* dimulai dengan memetakan *traffic light* yang terdekat dari reklame yang akan diukur, kemudian mengukur jarak pasar terdekat dengan reklame tersebut. Proses pengukuran dilakukan melalui sistem yang telah dibangun, yaitu dengan mengukur jarak antar titik koordinat yang telah diinput kedalam sistem. Proses pengukuran variabel jarak dengan *traffic light* sebagaimana terlihat pada Gambar 4.32.



Nomor	Alamat	Lokasi Pasang	Judul	traffic Terdekat	Jarak Dengan traffic
1	JL PALEBON RAYA NO.74 SEMARANG	PEREMPATAN SIBEDUG-KAJEN	AROMA	Bundaran Bersinyal Sibedug	10 meter
2	JL RAYA BAROS PEKALONGAN	JL. S. PARMAN KAUMAN - WIRADESA	HOTEL JAYA DIP	Simpang Bersinyal Gumawang	9 meter
3	JLDR.WAHIDIN 11 PEKALONGAN	JL. S. PARMAN KAUMAN - WIRADESA	HOTEL NIRWANA	Simpang Bersinyal Gumawang	14 meter
4	JL. A. YANI NO. 28 KUDUS-JAWA TENGAH	JL RAYA GAPGAWEN-KEDUNGWUNI	PRODUK PT. DJARUM	Simpang Bersinyal Podan	765 meter

Gambar 4.32 Tampilan pengukuran variabel jarak dengan *traffic light*

Berdasarkan Gambar 4.32 terlihat ada 3 *button*, yaitu *button* “Clear Data”, *button* “proses pemetaan reklame dengan *traffic light*”, dan *button* “proses hitung jarak dengan *traffic light*”. *Button* “Clear Data” digunakan untuk mengosongkan data jarak dengan *traffic light*. *Button* “proses pemetaan reklame dengan *traffic light*” digunakan untuk memetakan reklame dengan *traffic light* terdekat. Potongan *script* pemetaan reklame terhadap pasar sebagaimana terlihat pada Gambar 4.33.


```

1 <?php
2 if(isset($_POST['urut'])){
3
4     mysql_query("delete from relasi_traffic");
5     mysql_query("insert into relasi_traffic
6     select
7     id as id_baliho,
8     if((KD_KECAMATAN = 160 or KD_KECAMATAN = 101 or KD_KECAMATAN=161),1,
9     if((KD_KECAMATAN = 150),3,
10    if((KD_KECAMATAN = 110 or KD_KECAMATAN = 100),6,
11    if((KD_KECAMATAN = 130 or KD_KECAMATAN = 131 or KD_KECAMATAN=140 or KD_KECAMATAN=060 or KD_KECAMATAN=050),4,
12    if((KD_KECAMATAN = 080 or KD_KECAMATAN = 090 or KD_KECAMATAN=010 or KD_KECAMATAN=030 or KD_KECAMATAN=020),7,
13    if((KD_KECAMATAN = 070 or KD_KECAMATAN = 120),5,0)))) AS id_traffic
14    from tabel_baliho where tahun=2018");
15     echo"<Script>alert('Pemetaan Lokasi Reklame dengan traffic Berhasil diperbarui');window.history.go(-1)</script>";
16 }
17
18 ?>

```

Gambar 4.33 Potongan Script Pemetaan Reklame dengan *Traffic light*

Berdasarkan potongan script yang terlihat pada Gambar 4.33, baris ke-2 merupakan kondisi untuk mengecek variabel *urut* dengan metode *post*. Baris ke-4 merupakan perintah untuk menghapus data tabel *relasi_traffic*. Baris ke-5 sampai dengan baris ke-16 adalah perintah untuk memetakan *id_baliho* dengan *id_traffic* berdasarkan kecamatan dan baris ke-15 merupakan perintah untuk menampilkan pesan, jika perintah berhasil dilakukan. Setelah semua reklame berhasil dipetakan dengan *traffic light* terdekat, selanjutnya adalah menghitung jarak reklame dengan *traffic light* terdekat. Disini penulis telah membuat sistem yang mampu menghitung jarak reklame dengan *traffic light* berdasarkan titik koordinat pada masing- masing variabel. Seperti halnya pengukuran jarak reklame dengan pasar, pada pengukuran jarak reklame dengan *traffic light*, penulis juga menggunakan fungsi formula *haversine*. Potongan *script* dari pengukuran jarak reklame dengan *traffic light* sebagaimana terlihat pada Gambar 4.34.

```

1 <?php
2 if(isset($_POST['urut'])){
3     function rad($x){ return $x * M_PI / 180; }
4     function distHaversine($coord_a, $coord_b){
5         # jarak kilometer dimensi (mean radius) bumi
6         $R = 6371;
7         $coord_a = explode(" ", $coord_a);
8         $coord_b = explode(" ", $coord_b);
9         $dLat = rad(($coord_b[0]) - ($coord_a[0]));
10        $dLong = rad(($coord_b[1]) - ($coord_a[1]));
11        $a = sin($dLat/2) * sin($dLat/2) + cos(rad($coord_a[0])) * cos(rad($coord_b[0])) * sin($dLong/2) * sin($dLong/2);
12        $c = 2 * atan2(sqrt($a), sqrt(1-$a));
13        $d = $R * $c;
14        # hasil akhir dalam satuan kilometer
15        return number_format($d, 3, '.', '');
16    }
17    mysql_query("delete from jarak_pasar");
18
19    $minta= mysql_query("select a.id as kode, concat(a.lat,a.lng) as koordinat_reklame, concat(c.lat,c.lng) as koordinat_pasar
20    from tabel_baliho a join relasi_pasar b on a.id=b.id_baliho join ref_pasar c on b.id_pasar=c.id ");
21    while ($data=mysql_fetch_array($minta)){
22        $id=$data[kode];
23        $a=$data[koordinat_reklame];
24        $b=$data[koordinat_pasar];
25        $hasil = distHaversine($a, $b);
26        mysql_query("insert into jarak_pasar(id_baliho,jarak) values ($id,$hasil)");
27    }
28    echo"<Script>alert('Pemetaan Lokasi Reklame dengan Pasar Berhasil diperbarui');window.history.go(-1)</script>";
29 }
30 ?>

```

Gambar 4.34 Potongan Script Pengukuran Jarak Reklame dengan *Traffic light*

Berdasarkan Gambar 4.34 bahwa pada pengukuran jarak reklame dengan *traffic light*, digunakan fungsi formula *haversine*. , baris ke-2 merupakan kondisi untuk mengecek variabel urut dengan metode *post*. Baris ke-3 sampai dengan 16 menunjukkan fungsi dari formua *haversine*. Baris ke-17 merupakan perintah untuk menghapus data tabel jarak_pasar. Baris ke-19 sampai dengan ke-27 adalah perintah untuk menyimpan id dan perhitungan jarak reklame dengan *traffic light* dan baris ke-28 merupakan perintah untuk menampilkan pesan, jika perintah berhasil dilakukan.

4.6.3 Penghitungan volume kendaraan

Penghitungan volume kendaraan dilakukan berdasarkan volume kendaraan dimana reklame itu dipasang. Untuk volume kendaran diambil dari data survei CTMC (*Classified Turning Movement Counting*) yang dilakukan (Nugraha, 2016) pada Buku Kinerja Keselamatan Transportasi Jalan di Kabupaten Pekalongan. Tabel 4.2 menunjukkan hasil Survey Gerakan Membelok CTMC (*Classified Turning Movement Counting*) pada Simpangan Bersinyal (APILL/ Alat Pengatur Isyarat Lalu lintas).

Tabel 4.2 Hasil Survey Gerakan Membelok CTMC (*Classified Turning Movement Counting*) pada Simpangan Bersinyal (APILL/Alat Pengatur Isyarat Lalu lintas)

Jumlah Kendaraan yang melewati simpang 3 Podo

Jenis Kendaraan	Kaki Simpang Utara			Kaki Simpang Barat			Kaki Simpang Selatan			Jumlah	Nilai SMP smp/jam	Volume
	Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)					
	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan			
LV		25	171	179		78	76	59		588	1	588
HC		1	20	7		10	8	1		47	1,3	61,1
MC		367	727	1389		512	557	705		4257	0,4	1702,8
Total	0	393	918	1575	0	600	641	765	0	4892	Jumlah	2351,9

Jumlah Kendaraan yang melewati simpang 3 Sedayu

Jenis Kendaraan	Kaki Simpang Utara			Kaki Simpang Barat			Kaki Simpang Selatan			Jumlah	Nilai SMP smp/jam	Volume
	Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)					
	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan			
LV		216	19	12		56	7	4		314	1	314
HC		14	6	12		14	7	4		57	1,3	74,1
MC		804	191	402		419	203	1497		3516	0,4	1406,4
Total		1034	216	426		489	217	1505		3887	Jumlah	1794,5

Jumlah Kendaraan yang melewati simpang 3 Tirto

Jenis Kendaraan	Kaki Simpang Barat			Kaki Simpang Timur			Kaki Simpang Selatan			Jumlah	Nilai SMP smp/jam	Volume
	Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)					
	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan			
LV		426	13	10	340		18		13	820	1	820
HC		304	3	0	305		0		0	612	1,3	795,6
MC		3138	530	199	1674		191		208	5940	0,4	2376
Total		3868	546	209	2319		209		221	7372	Jumlah	3991,6

Jumlah Kendaraan yang melewati simpang 3 Mayangan

Jenis Kendaraan	Kaki Simpang Barat			Kaki Simpang Timur			Kaki Simpang Utara			Jumlah	Nilai SMP smp/jam	Volume
	Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)					
	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan			
LV	2	407			392	18	21		14	854	1	854
HC	0	297			280	1	0		3	581	1,3	755,3
MC	176	2616			1565	149	208		223	4937	0,4	1974,8
Total	178	3320			2237	168	229		240	6372	Jumlah	3584,1

Jumlah Kendaraan yang melewati simpang 4 Gumawang

Jenis Kendaraan	Kaki Simpang Barat			Kaki Simpang Timur			Kaki Simpang Utara			Kaki Simpang Selatan			Jumlah	Nilai SMP smp/jam	Volume
	Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)					
	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan			
LV	2	315	22	69	230	5	7	18	5	2	2	84	761	1	761
HC	0	306	0	11	230	0	0	0	0	14	1	6	568	1,3	738,4
MC	27	1608	103	349	1048	41	97	127	46	107	188	664	4405	0,4	1762
Total	29	2229	125	429	1508	46	104	145	51	123	191	754	5734	Jumlah	3261,4

Jumlah Kendaraan yang melewati simpang 3 Bojong

Jenis Kendaraan	Kaki Simpang Utara			Kaki Simpang Selatan			Kaki Simpang Timur			Jumlah	Nilai SMP smp/jam	Volume
	Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)					
	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan			
LV	22	127			93	20	19		28	309	1	309
HC	5	21			26	1	0		17	70	1,3	91
MC	432	849			733	513	282		310	3119	0,4	1247,6
Total	459	997	0	0	852	534	301	0	355	3498	Jumlah	1647,6

Jumlah Kendaraan yang melewati simpang 4 Sibebug

Jenis Kendaraan	Kaki Simpang Barat			Kaki Simpang Timur			Kaki Simpang Utara			Kaki Simpang Selatan			Jumlah	Nilai SMP smp/jam	Volume
	Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)			Gerakan kendaraan (Kendaraan/jam)					
	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan	Kiri	Lurus	Kanan			
LV	2	235	24	34	112	4	11	21	5	5	67	84	604	1	604
HC	0	0	0	0	1	0	0	0	0	2	1	3	7	1,3	9,1
MC	27	345	176	232	543	41	97	127	345	112	232	664	1232	0,4	492,8
Total	29	580	200	266	656	45	108	148	350	119	300	751	3552	Jumlah	1105,9

Keterangan

- LV : *Light Vehicle* / Kendaraan Ringan -> Kendaraan bermotor roda empat dengan dua gandar berjarak 2.0-3.0 m (termasuk kendaraan penumpang) seperti oplet, mikro bis, pick up dan truk kecil , sesuai klasifikasi bina marga)
- HV : *Heavy Vehicle* / Kendaraan Berat - > yaitu kendaraan bermotor dengan dua gandar, dengan jarak 3.5 - 5.00 (termasuk bis kecil, truk dua as dengan enam rodan sesuai klasifikasi bina marga), Truk Besar, Bis Besar
- MC : *Motorcycle* - > yaitu kendaraan bermotor beroda dua atau tiga (termasuk sepeda motor dan kendaraan beroda 3 sesuai klasifikasi bina marga)
- SMP : Satuan Mobil Penumpang

Berdasarkan tabel 4.2 dapat diketahui jumlah volume kendaraan pada simpang bersinyal / APILL yang berada di Kabupaten Pekalongan. Rumus (4.1) merupakan rumus dalam mencari volume kendaraan.

$$Q = P * Qv \dots\dots\dots (4.1)$$

Dimana Q = Volume kendaraan bermotor (smp/jam) , P = Faktor satuan mobil penumpang, Qv = Volume kendaraan bermotor (kendaraan/jam).

Berdasarkan rumus (4.1) didapatkan volume kendaraan pada simpang 3 Podo adalah 2351,9, simpang 3 Sedayu sebesar 1794,5, simpang 3 Tirto sebesar 3991,6, simpang 3 Mayangan sebesar 3584,1, simpang 4 Gumawang sebesar 3261,4 simpang 3 Bojong sebesar 1647,6 dan simpang 4 sipedug sebesar 1105,9. Setelah didapatkan volume kendaraan yang selanjutnya adalah memasukkan data-data volume kendaraan ke dalam basisdata SIG, untuk kemudian dilakukan proses perhitungan volume kendaraan. Proses Penghitungan variabel volume kendaraan sebagaimana terlihat pada Gambar 4.35.

The screenshot shows a web application interface for 'Tax Management Intelligence' (TMI). The main content area displays a table titled 'Pajak Reklame Jenis Baliho'. The table has six columns: Nomor, Alamat, Lokasi Pasang, Judul, Simpang Terdekat, and Volume Kendaraan. There are five rows of data visible. The interface also includes a search bar, a 'Clear Data' button, and a 'Proses Pemetaan Reklame dengan Volume Kendaraan' button. A sidebar on the left contains a navigation menu with items like Dashboard, Proses ETL, Perbarui Datamart, Data Baliho, Buffering Pasar, Buffering Traffic Light, Hitung Jarak Pasar, and Hitung Jarak Traffic Light.

Nomor	Alamat	Lokasi Pasang	Judul	Simpang Terdekat	Volume Kendaraan
1	JL. PALEBON RAYA NO.74 SEMARANG	PEREMPATAN SIBEDUG - KAJEN	AROMA	Bundaran Bersinyal Sibedug	1105.900
2	JL. RAYA BAROS FEKALONGAN	JL. S. FARMAN KAUMAN - WIRADESA	HOTEL JAYA DIPA	Simpang Bersinyal Gumawang	3261.400
3	JLDR.WAHIDIN TI FEKALONGAN	JL. S. FARMAN KAUMAN - WIRADESA	HOTEL NIRWANA	Simpang Bersinyal Gumawang	3261.400
4	JL. A. YANI NO.28 KUDUS - JAWA TENGAH	JL RAYA CAPGAWEN - KEDUNGWUNI	PRODUK PT. DJARUM	Simpang Bersinyal Podo	2351.900
5	JL. A. YANI NO.28 KUDUS - JAWA TENGAH	JL RAYA PODO - KEDUNGWUNI	PRODUK PT. DJARUM	Simpang Bersinyal Podo	2351.900

Gambar 4.35 Penghitungan variabel volume kendaraan

Berdasarkan Gambar 4.35 terlihat ada 3 (tiga) *button*, yaitu *button* “Clear Data” dan *button* “Proses Pemetaan Reklame dengan Volume Kendaraan. *Button* “Clear Data” digunakan untuk mengosongkan data Jarak Dengan *Traffic light*. *Button* “Proses Pemetaan Reklame dengan Volume Kendaraan” digunakan untuk memetakan reklame dengan volume kendaraan. Potongan *script* pemetaan reklame terhadap volume kendaraan sebagaimana terlihat pada Gambar 4.36.

```

1 <?php
2 if(isset($_POST['ubnt'])){
3
4 mysql_query("delete from relasi_penggunajalan");
5 mysql_query("delete from penggunajalan");
6 mysql_query("insert into relasi_penggunajalan
7 select
8 id as id_baliho,
9 if((KD_KECAMATAN = 160 or KD_KECAMATAN = 101 or KD_KECAMATAN=161),1,
10 if((KD_KECAMATAN = 150),3,
11 if((KD_KECAMATAN = 110 or KD_KECAMATAN = 100),6,
12 if((KD_KECAMATAN = 130 or KD_KECAMATAN = 131 or KD_KECAMATAN=140 or KD_KECAMATAN=060 or KD_KECAMATAN=050),4,
13 if((KD_KECAMATAN = 090 or KD_KECAMATAN = 090 or KD_KECAMATAN=010 or KD_KECAMATAN=030 or KD_KECAMATAN=020),7,
14 if((KD_KECAMATAN = 070 or KD_KECAMATAN = 120),5,0)))) AS id_traffic
15 from tabel_baliho where tahun=2018");
16
17 mysql_query("
18 INSERT into penggunajalan
19 select a.id,c.volume_kendaraan from tabel_baliho a
20 join relasi_penggunajalan b on a.id=b.id_baliho
21 join ref_traffic c on b.id_traffic = c.id
22 ");
23
24 echo"<Script>alert('Pemetaan Lokasi Reklame dengan kepadatan pengguna jalan Berhasil diperbarui');window.history.go(-1)</script>";
25
26 }
27
28 ?>

```

Gambar 4.36 Potongan *Script* Pemetaan Reklame dengan Volume Kendaraan

Berdasarkan potongan script yang terlihat pada Gambar 4.36, baris ke-2 merupakan kondisi untuk mengecek variabel urut dengan metode *post*. Baris ke-4 dan ke-5 merupakan perintah untuk menghapus data tabel relasi_penggunajalan dan penggunajalan. Baris ke-6 sampai dengan baris ke-15 adalah perintah untuk memetakan id_baliho dengan id_traffic berdasarkan kecamatan. Baris k-17 sampai dengan baris ke-22 merupakan perintah untuk menyimpan data hasil pemetaan ke dalam tabel penggunajalan dan baris ke-24 merupakan perintah untuk menampilkan pesan, jika perintah berhasil dilakukan.

4.7 Normalisasi Data

Data-data yang sudah dikumpulkan tidak bisa langsung dilakukan pemrosesan dikarenakan terdapat jarak atau besaran angka yang cukup jauh. Perbedaan besaran jarak atau angka tersebut terjadi dikarenakan ada beberapa kecamatan yang tidak ada *traffic light*nya dan tidak ada pasar umumnya. Seperti pada variabel *traffic light*, pengukuran reklame jenis baliho dilakukan dengan memetakan reklame pada *traffic light* pada kecamatan dimana baliho dipasang, karena pada kecamatan tersebut tidak ada *traffic light*nya maka pemetaan untuk pengukuran jarak dengan *traffic light* dipetakan dengan *traffic light* terdekat. Perbedaan besaran angka mengakibatkan distribusi data menjadi tidak normal. Untuk mengetahui normalitas data, perlu dilakukan pengujian normalitas. Pada penelitian ini pengujian normalitas dengan menggunakan metode *shapiro wilk*. Shapiro dan Wilk dalam Razali dan Wah (2011) menyampaikan bahwa uji *shapiro-wilk* merupakan uji yang paling baik untuk semua jenis distribusi dan ukuran sampel. Gambar 4.37 menunjukkan pengujian normalitas menggunakan metode *Shapiro wilk* di RStudio.



```
18:1 (Top Level) R Script
Console Terminal
~#
> shapiro.test(baliho$jarakpasar)
      shapiro-wilk normality test


data:  baliho$jarakpasar
W = 0.82204, p-value = 9.14e-06
>
```

Gambar 4.37 Pengujian normalitas menggunakan metode *Shapiro wilk* di RStudio

Berdasarkan Gambar 4.37, pengujian normalitas menggunakan metode *shapiro wilk* dihasilkan *p-value* $< \alpha$ ($9.14e-06 < 0,05$), sehingga kesimpulannya adalah data tidak berdistribusi normal. Langkah selanjutnya yaitu dengan melakukan *cleaning data*. Proses

cleaning dilakukan dengan membuang data-data dengan nilai yang ekstrim atau yang disebut dengan data *outlier*. Data pencilan atau data *outlier* itu muncul karena ada reklame yang muncul pada kecamatan yang tidak ada *traffic light*nya sehingga pengukuran jarak reklame dipetakan dengan *traffic light* yang terdekat dari wilayah kecamatan dimana reklame berada. Setelah dilakukan *cleaning*, data diuji kembali dengan pengujian normalitas menggunakan metode *Shapiro will* seperti diawal.

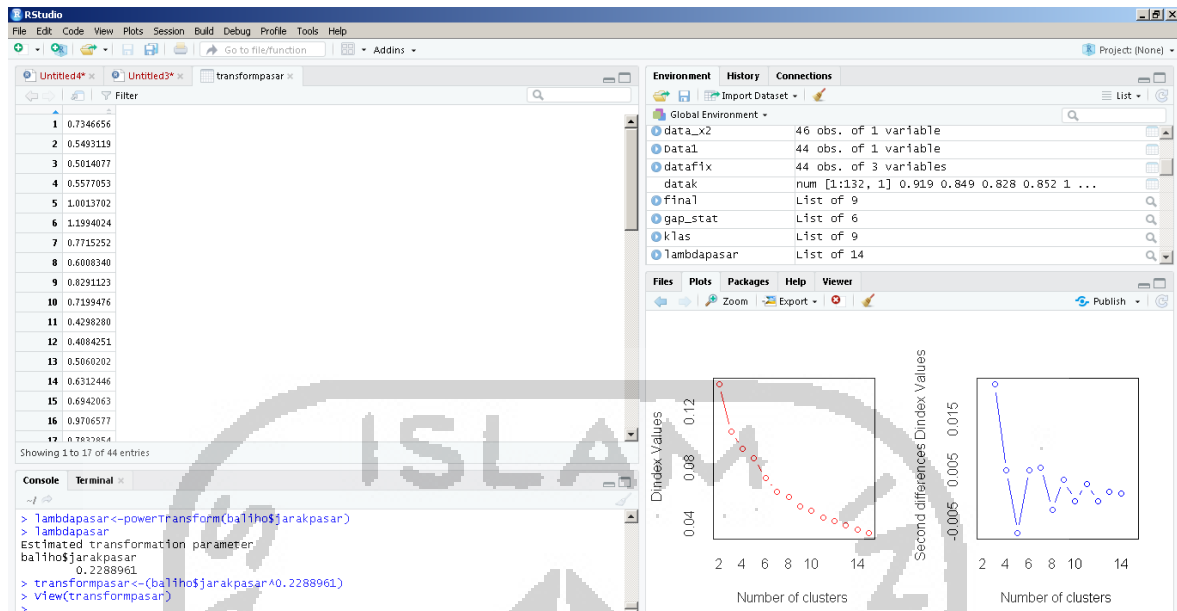
Setelah dilakukan pengujian, data tersebut masih belum terdistribusi secara normal. Maka langkah yang selanjutnya adalah dengan melakukan transformasi data. Transformasi data adalah upaya yang dilakukan dengan tujuan utama untuk mengubah skala pengukuran data asli menjadi bentuk lain sehingga data dapat memenuhi asumsi-asumsi yang mendasari analisis ragam. Untuk mengatasi ketidaknormalan data, penelitian ini menggunakan Tranformasi *Box Cox*.(Hadi, 2007). Transformasi *Box Cox* adalah transformasi pangkat berparameter tunggal λ (lambda). Nilai λ biasanya antara -5 sampai 5. Pendugaan parameter dapat dicari dengan menggunakan Metode Kemungkinan Maksimum (*Maximum Likelihood Metode*). Untuk melakukan tranformasi *box cox*, dapat dilakukan dengan menggunakan RStudio yaitu dengan memanfaatkan *library car*. Tranfromasi dengan *box cox* ini dimulai dengan mencari nilai λ dengan *Maximum Likelihood Metode*. Gambar 4.38 menunjukkan proses pencarian nilai λ dengan menggunakan *function powerTransform* pada *library car* di RStudio.



```
Console Terminal x
~1 →
w = 0.82204, p-value = 9.14e-06
> lambdapasar<-powerTransform(balihof$jarakpasar)
> lambdapasar
Estimated transformation parameter
balihof$jarakpasar
0.2288961
>
```

Gambar 4.38 proses pencarian nilai λ (lambda) di RStudio

Berdasarkan Gambar 4.38, dapat diketahui bahwa nilai λ yang dihasilkan berdasarkan *Maximum Likelihood Metode* adalah 0.2288961. Setelah ditemukan nilai λ , maka yang selanjutnya adalah melakukan transformasi data dengan rumus $Y=X^\lambda$. Gambar 4.37 menunjukkan proses dan hasil tranformasi data di RStudio.



Gambar 4.39 Proses dan hasil transformasi data di RStudio

Setelah data berhasil ditransformasi, langkah selanjutnya adalah uji normalitas kembali dari data yang telah ditransformasi. Untuk uji normalitas pada penelitian ini menggunakan beberapa metode, untuk memastikan data yang ditransformasi telah normal. Gambar 4.40 menunjukkan uji normalitas dengan *shapiro wil* di RStudio.



Gambar 4.40 Uji normalitas data dengan *shapiro will* di RStudio

Berdasarkan Gambar 4.38, dapat diketahui bahwa $p\text{-value} = 0.08834$ maka keputusan $p\text{-value} > \alpha$ atau $0.08834 > 0,05$, sehingga kesimpulannya adalah data telah berdistribusi normal. Untuk memastikan data telah terdistribusi normal maka dilakukan uji normalitas menggunakan metode lain. Metode yang digunakan adalah *Jarque-Bera test for normality* dan *Shapiro-Francia*, dengan menggunakan *libray normtest* dan *nortest* pada RStudio. Gambar 4.41 menunjukkan uji normalitas dengan metode lain di RStudio.


```
Console Terminal x
w = 0.95551, p-value = 0.08834

> library(normtest)
> library(nortest)
> jb.norm.test(transformpasar)

      Jarque-Bera test for normality

data: transformpasar
JB = 1.5127, p-value = 0.314

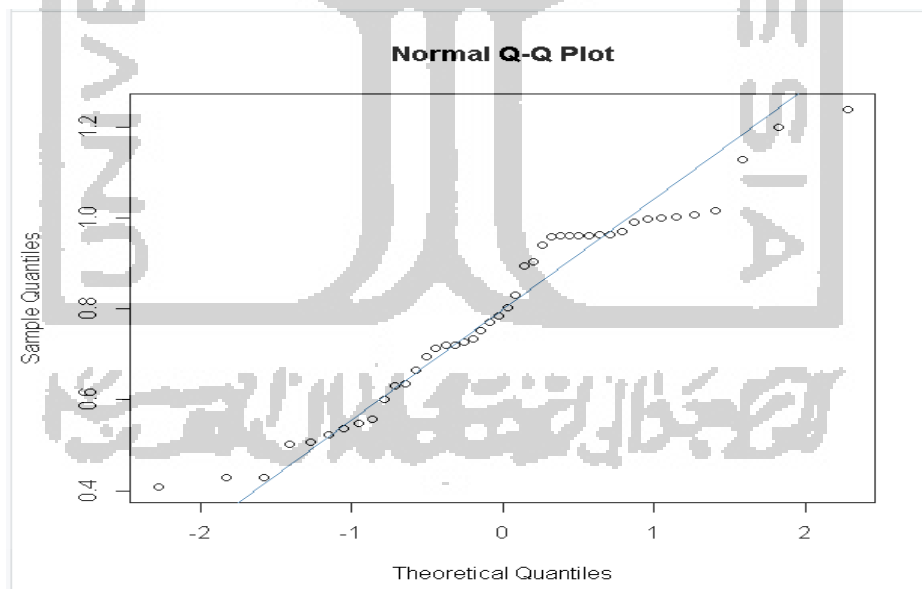
> sf.test(transformpasar)

      Shapiro-Francia normality test

data: transformpasar
w = 0.96247, p-value = 0.1421
```

Gambar 4.41 Uji normalitas data dengan *Jarque-Bera* dan *Shapiro-Francia* di RStudio

Berdasarkan Gambar 4.41, dapat diketahui bahwa *p-value* pada *Jarque-Bera* dan *Shapiro-Francia* adalah 0.1421 dan 0.03744, maka keputusan *p-value* pada *Jarque-Bera* dan *Shapiro-Francia* $> \alpha$ atau $0.1421 < 0,05$ dan $0.03744 < 0,05$, sehingga dapat disimpulkan bahwa data telah berdistribusi normal. Proses selanjutnya adalah melakukan *plotting* data dari data yang telah ditransformasi dan juga telah diuji normalitas. Dengan menggunakan perintah *qqnorm* pada RStudio. Gambar 4.42 menunjukkan *plotting* data di RStudio.



Gambar 4.42 Hasil *plotting* data di RStudio

Berdasarkan Gambar 4.42, dapat dikatakan data berdistribusi normal, karena plot cenderung membentuk garis lurus jika dihubungkan dengan garis.

4.8 Clustering K-means

Cluster Analisis Reklame pada Kabupaten Pekalongan Tahun 2018 menggunakan Algoritma *K-means* dimulai dengan menentukan jumlah *cluster* untuk pengelompokan data. Yang menjadi kelemahan dalam algoritma *K-means* adalah menganalisa dan menentukan jumlah k terbaik dalam mengcluster data pada suatu dataset. Identifikasi jumlah *cluster* k merupakan cara yang paling penting dan utama pada proses *clustering* dengan menggunakan algoritma KMeans, dimana hasil *cluster* akan bergantung pada jumlah *cluster* awal. Sehingga jika jumlah *cluster* yang ditentukan tidak baik, maka hasil *cluster* juga tidak akan sesuai dengan yang diharapkan yaitu tidak akan menghasilkan informasi yang dibutuhkan oleh pengguna. Untuk menentukan jumlah *cluster* yang optimal, penelitian ini menggunakan library NBClust pada RStudio dengan menggunakan semua indeks yang ada.(Charrad, 2014). Tabel 4.3 menunjukkan daftar indeks yang berada pada librari NBClust.

Tabel 4.3 Indeks yang diterapkan dalam paket *NbClust*

No.	Nama index di NbClust	Metode penentuan jumlah cluster yang optimal
1.	<i>ch</i> (Calinski dan Harabasz, 1974)	Nilai maksimum pada indeks
2.	<i>duda</i> (Duda dan Hart, 1973)	Jumlah <i>cluster</i> terkecil, dimana nilai indeks > <i>critical value</i>
3.	<i>pseudot2</i> (Duda dan Hart, 1973)	Jumlah <i>cluster</i> terkecil, dimana indeks < <i>critical value</i>
4.	<i>cindex</i> (Hubert dan Levin, 1976)	Nilai minimum pada indeks
5.	<i>gamma</i> (Baker dan Hubert, 1975)	Nilai maksimum pada indeks , dimana nilai kritis > = alpha
6.	<i>beale</i> (Beale, 1969)	Jumlah <i>cluster</i> dimana nilai kritis > = alpha
7.	<i>ccc</i> (Sarle, 1983)	Nilai maksimum pada indeks
8.	<i>ptbiserial</i> (Milligan 1980 , 1981)	Nilai maksimum pada indeks
9.	<i>gplus</i> (Rohlf, 1974 ; Milligan, 1981)	Nilai minimum pada indeks
10.	<i>db</i> (Davies dan Bouldin, 1979)	Nilai minimum pada indeks
11.	<i>frey</i> (Frey dan Van Groenewoud, 1972)	Level <i>cluster</i> sebelumnya nilai indeks < 1,00
12.	<i>hartigan</i> (Hartigan, 1975)	Perbedaan maksimum antara tingkat hierarki indeks
13.	<i>tau</i> (Rohlf, 1974 ; Milligan, 1981)	Nilai maksimum pada indeks
14.	<i>ratkowsky</i> (Ratkowsky dan Lance, 1978)	Nilai maksimum pada indeks
15.	<i>scott</i> (Scott dan Symons, 1971)	Perbedaan maksimum antara tingkat hierarki indeks
16.	<i>marriot</i> (Marriot, 1971)	Nilai perbedaan yang maksimum antar kedua tingkat indeks

No.	Nama <i>index</i> di NbClust	Metode penentuan jumlah <i>cluster</i> yang optimal
17.	<i>ball</i> (Ball and Hall, 1965)	Perbedaan maksimum antar tingkat hierarki indeks
18.	<i>trcovw</i> (Milligan dan Cooper, 1985)	Perbedaan maksimum antar tingkat hierarki indeks
19.	<i>tracew</i> (Milligan dan Cooper, 1985)	Nilai maksimal pada perbedaan kedua antar level
20.	<i>friedman</i> (Friedman dan Rubin, 1967)	Perbedaan maksimum antara tingkat hierarki indeks
21.	<i>mcclain</i> (McClain dan Rao, 1975)	Nilai minimum pada indeks
22.	<i>rubin</i> (Friedman dan Rubin, 1967)	Nilai minimum kedua perbedaan antar level
23.	<i>kl</i> (Krzanowski dan Lai, 1988)	Nilai maksimum pada indeks
24.	<i>silhouette</i> (Rousseeuw, 1987)	Nilai maksimum pada indeks
25.	<i>gap</i> (Tibshirani, dkk. 2001)	Jumlah <i>cluster</i> terkecil seperti itu bahwa $criticalValue > = 0$
26.	<i>dindex</i> (Lebart, dkk. 2000)	Metode grafis
27.	<i>dunn</i> (Dunn 1974)	Nilai maksimum pada indeks
28.	<i>hubert</i> (Hubert dan Arabie, 1985)	Metode grafis
29.	<i>sdindex</i> (Halkidi, dkk, 2000)	Nilai maksimum pada indeks
30.	<i>sdbw</i> (Halkidi dan Vazirgiannis, 2001)	Nilai maksimum pada indeks

Penentuan jumlah *cluster* pada penelitian ini menggunakan Aplikasi RStudio. Data yang telah dinormalisasi kemudian di proses di RStudio untuk menentukan jumlah *cluster* yang optimal. Gambar 4.43 menunjukkan proses penentuan jumlah *cluster* menggunakan library *NBClust* pada RStudio.

```

> View(hasiltransform)
> NbClust(hasiltransform, method="complete", index="all")
*** : The Hubert index is a graphical method of determining the number of clusters.
In the plot of Hubert index, we seek a significant knee that correspond
s to a significant increase of the value of the measure i.e the significant peak in Hubert index second differences plot.

*** : The D index is a graphical method of determining the number of clusters.
In the plot of D index, we seek a significant knee (the significant peak in D index second differences plot) that corresponds to a significant increase of the value of the measure.

***** conclusion *****

* Among all indices:
* 4 proposed 2 as the best number of clusters
* 5 proposed 3 as the best number of clusters
* 3 proposed 4 as the best number of clusters
* 1 proposed 5 as the best number of clusters
* 2 proposed 7 as the best number of clusters
* 4 proposed 10 as the best number of clusters
* 1 proposed 11 as the best number of clusters
* 3 proposed 15 as the best number of clusters

* According to the majority rule, the best number of clusters is 3

*****
$All.index
      KL      CH Hartigan      CCC      Scott Marriot TrCovW TraceW Friedman
2  0.8656 42.1513 36.9480 26.6346 353.8296 2.0850 4.4803 5.4106 1042.262
3  2.7195 56.7071 18.1123 20.5119 406.8339 1.4065 1.1350 2.8784 2233.749
4  0.7906 59.0663 21.1159 17.9746 440.5684 1.1615 0.9636 1.9965 2461.034

```

Gambar 4.43 Proses Penentuan Jumlah *Cluster* pada RStudio

Berdasarkan 4.43, proses penentuan jumlah *cluster* menggunakan aplikasi RStudio. Penentuan jumlah *cluster* dilakukan dengan memanfaatkan *library NBClust* dengan semua index yang ada. Gambar 4.44 menunjukkan rincian nilai index dari hasil penentuan jumlah *cluster* di RStudio.

```

$All.index
  KL      CH Hartigan      CCC      Scott Marriot TrCovw Tracew Friedman
2  0.8656 42.1513 36.9480 26.6346 353.8296 2.0850 4.4803 5.4106 1042.262
3  2.7195 56.7071 18.1123 20.5119 406.8339 1.4065 1.1350 2.8784 2233.749
4  0.7906 59.0663 21.1159 17.9746 440.5684 1.1615 0.6636 1.9965 2461.034
5  0.9173 71.1402 28.2622 17.1480 492.6570 0.5555 0.3596 1.3067 2674.865
6  1.7407 101.1456 22.4128 17.5531 538.6909 0.2810 0.0783 0.7576 5775.102
7  4.5086 134.1127 7.7587 17.9436 606.0663 0.0827 0.0303 0.4766 18224.604
8  0.3866 136.3791 17.7755 17.2940 624.5404 0.0710 0.0171 0.3939 18512.402
9  2.4724 175.4620 9.3861 17.8385 655.6305 0.0443 0.0080 0.2637 19249.035
10 3.3927 193.1542 4.2860 17.7534 715.9505 0.0139 0.0054 0.2080 70436.567
11 0.5014 190.4114 6.4330 17.2238 724.5213 0.0138 0.0041 0.1847 70491.201
12 1.2260 201.1444 5.6750 17.0682 745.1038 0.0103 0.0034 0.1546 70924.681
13 1.1522 210.7558 5.2221 16.9190 761.0886 0.0084 0.0025 0.1313 74485.395
14 0.8674 220.3714 6.0754 22.9244 777.9011 0.0067 0.0020 0.1123 75228.876
15 0.8780 238.2878 7.4360 23.2414 799.8456 0.0046 0.0015 0.0934 76494.572

  Rabin Cindex      DB silhouette      Duda Pseudot2      Beale Ratkowsky Ball
2  251.5326 0.4017 0.8798 0.4682 0.3896 45.4393 2.5786 0.2992 2.7053
3  472.8098 0.3616 0.7634 0.4954 0.3013 25.5050 3.6183 0.3047 0.9595
4  681.6798 0.4541 0.6179 0.5446 0.5386 18.8454 1.3949 0.3168 0.4991
5  1041.5367 0.4199 0.6533 0.5400 0.2724 37.3898 4.2435 0.3556 0.2613
6  1796.3086 0.3009 0.6341 0.6308 0.1339 45.2628 9.6320 0.3362 0.1263
7  2855.7904 0.2744 0.5426 0.6761 85.7408 -4.9417 -1.4021 0.3261 0.0681
8  3454.6365 0.1073 0.4985 0.6764 0.1029 26.1560 11.1321 0.3078 0.0492
9  5160.4080 0.0966 0.3955 0.7020 277.0357 -1.9928 -1.1308 0.3009 0.0293
10 6544.2940 0.0924 0.3474 0.7247 497.7867 -4.9900 -1.4158 0.2867 0.0208
11 7369.2664 0.0887 0.3313 0.7142 0.5014 8.9504 1.5237 0.2749 0.0168
12 8805.8172 0.0809 0.4215 0.6250 249.2578 -5.9759 -1.4534 0.2724 0.0129
13 10367.4777 0.0734 0.4050 0.6165 328.2616 -3.9878 -1.3578 0.2637 0.0101
14 12113.9174 0.0679 0.4117 0.6066 441.1456 -2.9932 -1.2739 0.2558 0.0080
15 14567.1274 0.0630 0.3829 0.6321 0.0224 130.9182 55.7192 0.2480 0.0062

  Ptbiserial Frey McClain Dunn Hubert SDindex Dindex SDbw
2  0.5865 0.3473 0.3848 0.2918 0.1244 15.0837 0.3133 0.5205
3  0.6911 0.4592 0.6041 0.3610 0.1624 9.5464 0.2315 0.2052
4  0.6989 0.9223 0.6581 0.4973 0.1778 7.7083 0.1933 0.1233
5  0.6135 0.2673 1.0513 0.2905 0.1808 9.5980 0.1463 0.0964
6  0.6069 0.1681 1.0274 0.3187 0.1857 10.0893 0.1079 0.0739
7  0.6087 0.3369 0.9269 0.3626 0.1900 9.3557 0.0860 0.0487

```

Gambar 4.44 Rincian nilai index pada *library NBClust*

Critical Values digunakan untuk menentukan jumlah *cluster* yang terbaik pada metode *Duda*, *PseudoT2*, dan *Beale*. Gambar 4.45 menunjukkan rincian nilai *critical values* dari hasil penentuan jumlah *cluster* di RStudio menggunakan *library NBClust*.

```

$All.criticalvalues
  Critvalue_Duda Critvalue_PseudoT2 Fvalue_Beale
2  0.3869 45.9547 0.0588
3  0.1687 54.1943 0.0231
4  0.3322 44.2307 0.2521
5  0.2298 46.9294 0.0105
6  0.0438 152.9360 0.0003
7  -0.0559 -94.5213 1.0000
8  -0.2104 -17.2571 0.0022
9  -0.3282 -8.0930 1.0000
10 -0.0559 -94.5213 1.0000
11  0.1148 69.3995 0.2309
12 -0.0014 -4400.8778 1.0000
13 -0.1234 -36.4022 1.0000
14 -0.2104 -17.2571 1.0000
15 -0.2104 -17.2571 0.0000

```

Gambar 4.45 Rincian nilai *critical values*

Untuk menentukan jumlah *cluster* yang optimal ada metode yang memerlukan *critical values* untuk penentuan jumlah *clusternya*. Sebagai contoh pada *metode Duda*, bahwa untuk penentuan jumlah *cluster* yang terbaik adalah mencari jumlah *cluster* terkecil, dimana nilai indeks harus lebih besar dari *critical value*. Berdasarkan Gambar 4.44 dan Gambar 4.45, bahwa Jumlah *cluster* yang terkecil yang nilai indeksnya lebih besar dari *critical value* adalah 2 dengan nilai index 0.3896 dan *critical value* 0.3869. Gambar 4.46 menunjukkan jumlah *cluster* terbaik yang diusulkan dari Aplikasi RStudio dengan memanfaatkan *library NBClust*.

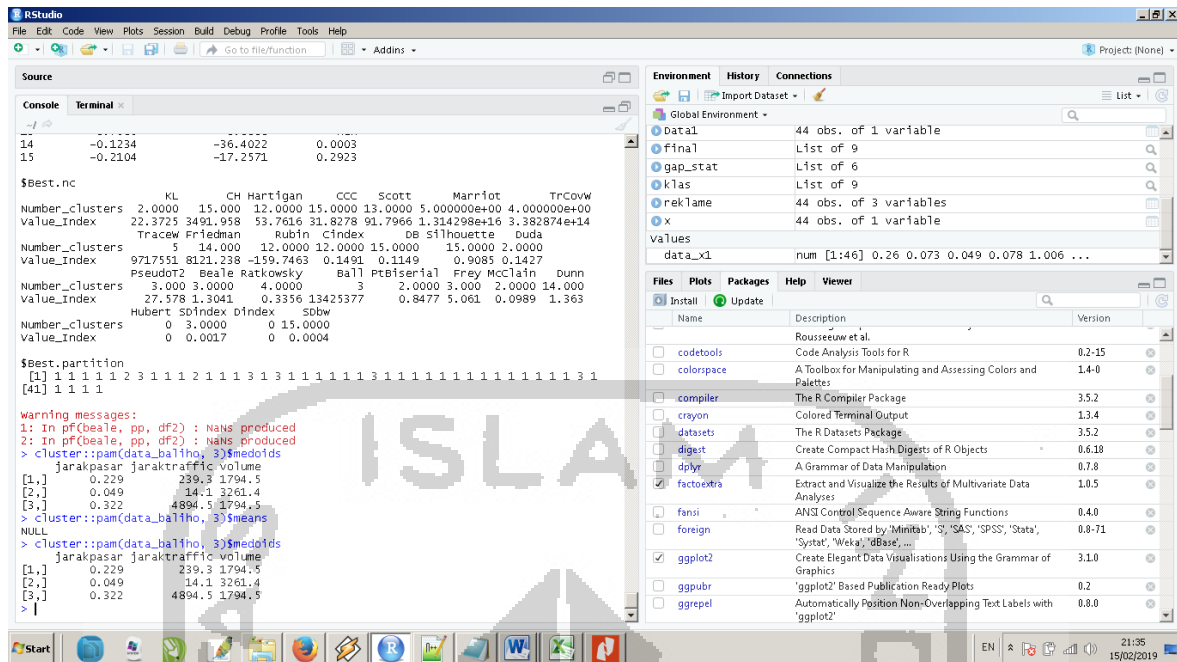
```

$Best.nc
      KL      CH Hartigan      CCC      Scott Marriot TrCovW Tracew
Number_clusters 7.0000 15.0000  3.0000  2.0000  7.0000  3.0000  3.0000  3.0000
Value_Index     4.5086 238.2878 18.8357 26.6346 67.3754  0.4337  3.3453  1.6502
      Friedman      Rubin Cindex      DB Silhouette      Duda PseudoT2      Beale
Number_clusters 10.00  10.0000 15.000 11.0000  10.0000  2.0000  2.0000  4.0000
Value_Index     51187.53 -558.9136  0.063  0.3313  0.7247  0.3896  45.4393  1.3949
      Ratkowsky      Ball PtBiserial Frey McClain      Dunn Hubert SDindex
Number_clusters  5.0000  3.0000  4.0000  1  2.0000 10.0000  0  4.0000
Value_Index     0.3556  1.7458  0.6989  NA  0.3848  0.6493  0  7.7083
      Dindex      SDbw
Number_clusters  0 15.0000
Value_Index     0 0.0061

```

Gambar 4.46 Jumlah *cluster* Terbaik yang Diusulkan dari Aplikasi RStudio dengan Memanfaatkan *Library NBClust*

Gambar 4.46 menunjukkan bahwa ada 4 (empat) *metode* (*CCC*, *Duda*, *PseudoT2* dan *McClain*) yang mengusulkan jumlah *cluster* terbaik adalah 2 (dua) *cluster*, 5 (lima) *metode* (*Hartigan*, *Marriot*, *TrcovW*, *Tracew* dan *Ball*) yang mengusulkan 3 (tiga) *cluster*, 3 (tiga) *metode* (*Beale*, *PtBiserial* dan *SDindex*) mengusulkan 4 (empat) *cluster*, 1 (satu) *metode* (*Ratkowsky*) mengusulkan 5 (lima) *cluster*, 2 (dua) *metode* (*Scot* dan *KL*) mengusulkan 7 (tujuh) *cluster*, 4 (empat) *metode* (*Friedman*, *Rubin*, *Shilouette* dan *Dunn*) mengusulkan 10 (sepuluh) *cluster*, 1 (satu) *metode* (*DB*) mengusulkan 11 (sebelas) *cluster* dan 3 (tiga) *metode* (*CH*, *Cinde* dan *SDbw*) mengusulkan 15 (lima belas) *Cluster*. Berdasarkan aturan mayoritas, maka jumlah *cluster* yang terbaik adalah 3 *cluster*. Setelah mendapatkan jumlah *cluster* yang terbaik, selanjutnya adalah pusat *cluster* awal (*centroid*). Penentuan *centroid* pada RStudio sebagaimana terlihat pada gambar 4.47



Gambar 4.47 Penentuan *centroid* (pusat *cluster*) awal pada RStudio

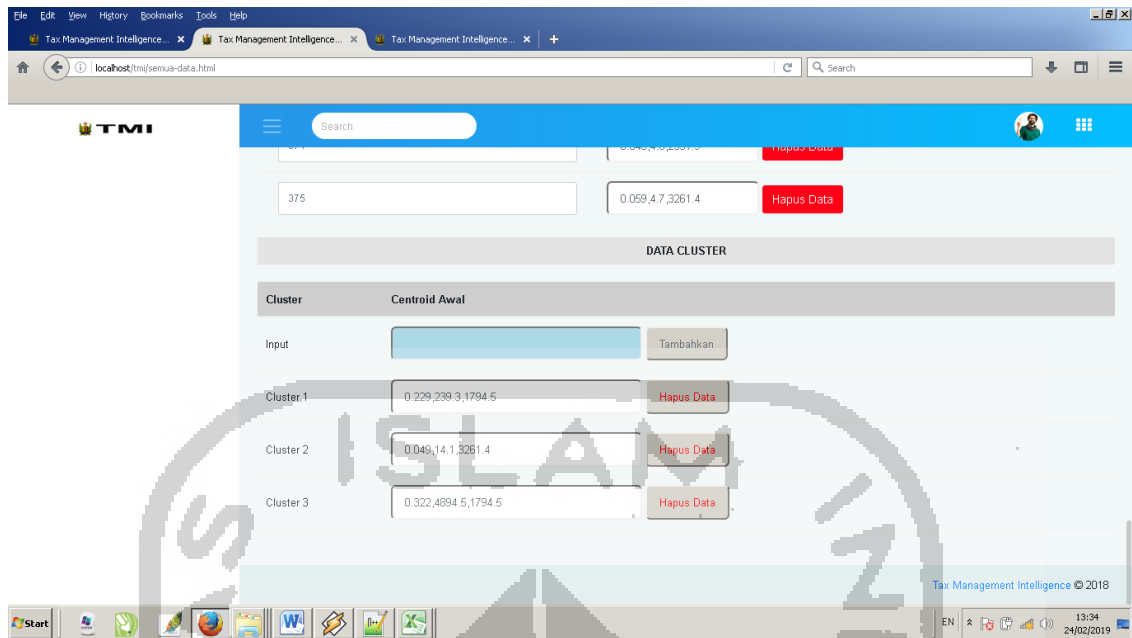
Berdasarkan Gambar 4.47, jumlah *cluster* yang dihasilkan dari RStudio adalah 3 (tiga) *Cluster*. Tabel 4.4 menunjukkan pembentukan *cluster*.

Tabel 4.4 Pembentukan *Cluster*

<i>Cluster</i>	Jarak dengan <i>Traffic light</i>	Jarak dengan Pasar	Tingkat Keramaian
1	0,229	239,30	1794,50
2	0,049	14,10	3261,40
3	0,322	4894,50	1794,50

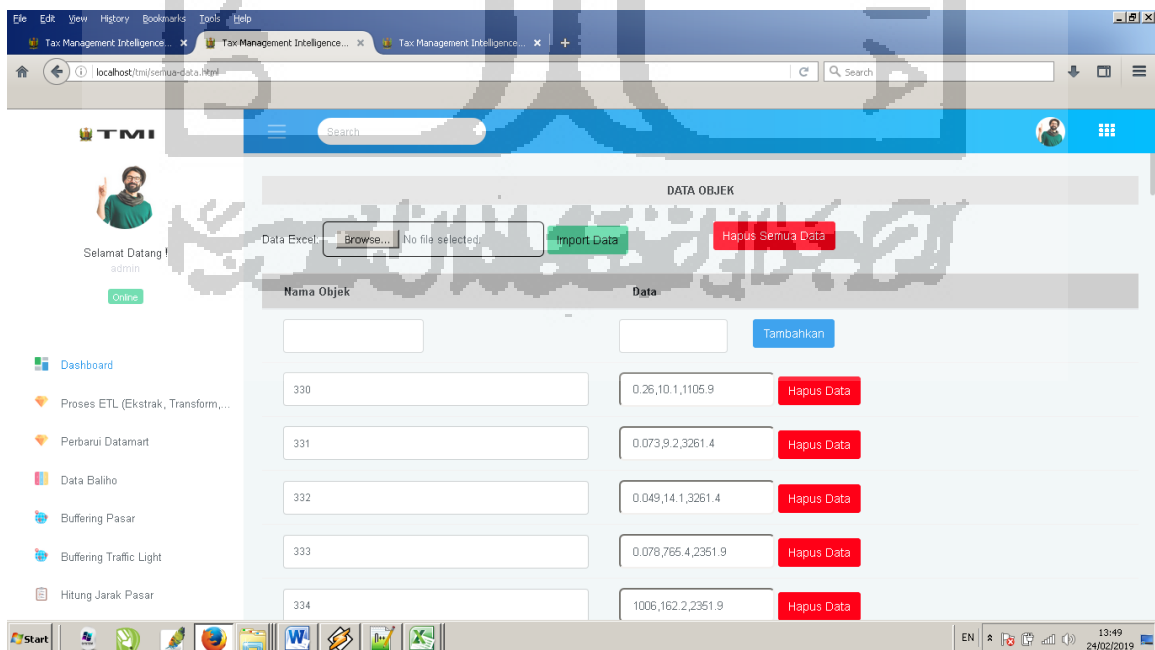
Setelah menentukan jumlah *cluster*, selanjutnya adalah mengalokasikan masing-masing data ke *centroid*/rata-rata terdekat. Untuk mengukur jarak data ke pusat kelompok (*centroid*), menggunakan *Metode Euclidean*. Pengukuran jarak pada ruang jarak (*distance space*) *Euclidean* dapat dicari menggunakan rumus (3.1).

Penelitian ini dibangun sebuah sistem yang mampu melakukan *clustering* dengan menggunakan Algoritma *K-means*. Proses *clustering* dilakukan dengan memasukkan pusat *cluster* (*centroid*) awal kedalam sistem terlebih dahulu. Tampilan proses input *centroid* (pusat *cluster*) awal sebagaimana terlihat pada Gambar 4.48



Gambar 4.48 Proses input *centroid* awal

Setelah berhasil memasukkan *centroid* awal yang selanjutnya adalah memasukkan data yang akan dilakukan *clustering*. Data yang telah diolah melalui sistem kemudian diextract kedalam format excel, data yang diextract kemudian diolah kembali agar dapat dilakukan proses *clustering*, kemudian dimasukkan kedalam tabel objek. Proses input data kedalam sistem melalui aplikasi import data yang telah dibangun. Proses import data sebagaimana terlihat pada Gambar 4.49



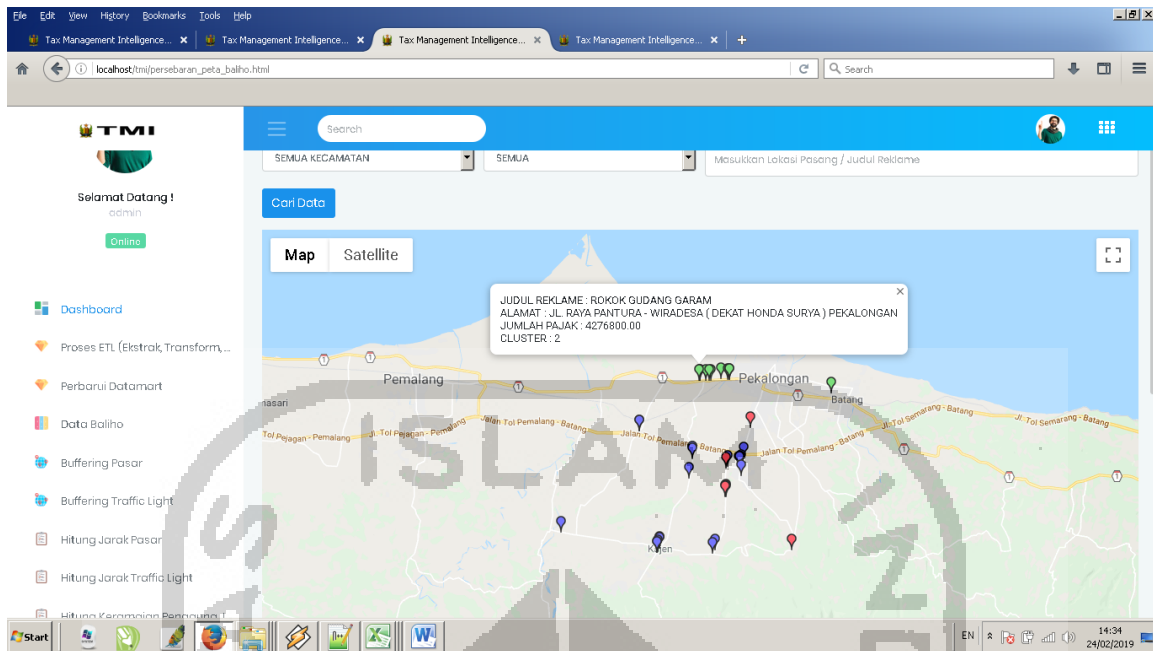
Gambar 4.49 Proses import data

Berdasarkan Gambar 4.49, proses input data ke sistem dapat dilakukan melalui *import* data dengan format .xls. Button “Import Data” digunakan untuk melakukan proses *import* data dari excel ke sistem dengan memilih file yang berextensi .xls terlebih dahulu. Button “Hapus Semua Data” digunakan untuk menghapus semua data objek yang berada pada sistem. Setelah data berhasil diimport yang selanjutnya adalah melakukan proses *clustering* dengan menekan button “Lakukan Proses *Clustering*”. Hasil *clustering* sebagaimana terlihat pada Gambar 4.50.

Objek	Data 1	Data 2	Data 3	Cluster 1	Cluster 2	Cluster 3
Objek 1	0.26	10.1	1105.9	OK	Null	Null
Objek 2	0.073	9.2	3261.4	Null	OK	Null
Objek 3	0.049	14.1	3261.4	Null	OK	Null
Objek 4	0.078	765.4	2351.9	OK	Null	Null
Objek 5	1006	162.2	2351.9	OK	Null	Null
Objek 6	2213	7429.8	1794.5	Null	Null	OK
Objek 7	0.322	4894.5	1794.5	Null	Null	OK
Objek 8	0.108	98.2	1794.5	OK	Null	Null
Objek 9	0.441	1435.4	1105.9	OK	Null	Null
Objek 10	0.238	2489.2	1647.6	OK	Null	Null
Objek 11	0.025	7435	2351.9	Null	Null	OK
Objek 12	0.02	250.3	1105.9	OK	Null	Null
Objek 13	0.051	12.2	3261.4	Null	OK	Null
Objek 14	0.134	404.2	1105.9	OK	Null	Null
Objek 15	0.203	5013.8	1794.5	Null	Null	OK
Objek 16	0.878	34.2	2351.9	OK	Null	Null
Objek 17	0.344	4872	1794.5	Null	Null	OK
Objek 18	0.644	117.2	1647.6	OK	Null	Null
Objek 19	0.287	17.1	1105.9	OK	Null	Null
Objek 20	0.612	149.7	1647.6	OK	Null	Null
Objek 21	2547	1785.6	1647.6	OK	Null	Null
Objek 22	0.168	104.5	3261.4	Null	OK	Null
Objek 23	0.99	123	3991.6	Null	OK	Null
Objek 24	0.959	3558.6	2351.9	Null	Null	OK
Objek 25	1695	851.6	2351.9	OK	Null	Null

Gambar 4.50 Hasil *Clustering*

Berdasarkan Gambar 4.50 proses *clustering* telah berhasil dilakukan. Untuk menguji kebenaran hasilnya, penulis melakukan perbandingan dengan melakukan *clustering* menggunakan data yang sama di RStudio. Setelah dilakukan perbandingan dan hasilnya sama. Setelah proses *clustering* berhasil dilakukan, hasil dari proses *clustering* tersebut ditampilkan dalam peta spasial. Gambar 4.51 menunjukkan tampilan dari hasil *clustering* dalam peta spasial.



Gambar 4.51 Hasil *Clustering* dalam Peta Spasial

Berdasarkan Gambar 4.51 dapat diketahui bahwa, hasil dari *clustering* dengan menggunakan algoritma *k-means* didapatkan 3 (tiga) *cluster*. Dapat terlihat disitu bahwa *cluster* 1 ditunjukkan dengan warna biru, kemudian *cluster* 2 (tiga) ditunjukkan dengan warna hijau dan *cluster* 3 (tiga) ditunjukkan dengan warna biru. Untuk hasil rincian proses *clustering* menggunakan algoritma *k-means* dapat dijelaskan secara rinci sebagai berikut ;

Hasil penghitungan jarak ke *centroid* pada iterasi ke-I menggunakan rumus (3.1) sebagaimana terlihat pada Tabel 4.5

Tabel 4.5 Hasil Penghitungan Jarak ke *Centroid* pada Iterasi ke-I

Id Reklame	C1	C2	C3	Jarak terdekat
330	725,7427926	2155,503722	4932,700409	C1
331	1484,837245	4,900058775	5100,779525	C2
332	1484,085807	0	5096,086719	C2
333	766,4698251	1179,678745	4166,552727	C1
334	562,7075384	921,4796665	4765,01401	C1
335	7190,500274	7559,391959	2535,300705	C3
336	4655,200001	5096,086719	0	C3
337	141,1000519	1469,308825	4796,300005	C1
338	1380,154055	2581,912875	3526,973601	C1
339	2254,690582	2954,736957	2409,781672	C1
340	7217,256632	7476,426089	2600,92966	C3

Id Reklame	C1	C2	C3	Jarak terdekat
343	688,6878855	2168,402797	4694,972172	C1
344	1484,375307	1,900001053	5097,906332	C2
345	708,0691908	2190,515525	4542,792543	C1
346	4774,5	5210,450626	119,3000594	C3
347	593,9370263	909,7224562	4892,158129	C1
348	4632,700001	5074,543143	22,50001076	C3
349	191,0188269	1617,090104	4779,558034	C1
350	723,5625774	2155,502101	4925,769049	C1
351	172,0695112	1619,486992	4747,073492	C1
352	1553,263877	2396,365776	3112,369478	C1
353	1473,080668	90,40007832	5009,580385	C2
354	2200,176056	738,2764628	5253,043033	C2
355	3365,775956	3659,326622	1447,523394	C3
356	828,0146129	1236,365322	4081,144087	C1
357	605,8105527	909,5808314	4924,149803	C1
358	605,5366996	909,5717879	4923,454303	C1
359	1485,810663	11,10001302	5106,717921	C2
360	0	1484,085807	4655,200001	C1
361	601,1739848	909,5003345	4912,127765	C1
362	605,2635321	909,5632829	4922,758804	C1
363	722,374453	2155,511053	4921,907376	C1
364	280,1960993	1613,855789	4896,004314	C1
365	708,8687172	2191,123572	4539,431859	C1
366	2254,391219	2954,48566	2410,081114	C1
367	2200,54598	737,3041087	5259,220451	C2
368	1717,960988	1119,400466	4036,944038	C2
369	606,4390721	909,6035246	4925,739522	C1
370	4592,300003	5035,881449	62,90003155	C3
371	1480,566156	24,50001176	5072,62856	C2
372	2202,276062	733,9710353	5284,491093	C2
373	605,0299359	909,5564225	4922,162663	C1
374	604,7192681	909,5478978	4921,367811	C1
375	1485,541248	9,400005319	5105,08959	C2

Dari tabel 4.5 Hasil penghitungan jarak ke *centroid* pada Iterasi ke-I, didapatkan keanggotaan sebagaimana terlihat pada Tabel 4.6

Tabel 4.6 Keanggotaan pada Iterasi ke-I

<i>Cluster</i>	ID Reklame	Jumlah
<i>Cluster 1</i>	330, 333, 334, 337, 338, 339, 343, 345, 347, 349, 350, 351, 352, 356, 357, 358, 360, 361, 362, 363, 364, 365, 366, 369, 373, 374	26
<i>Cluster 2</i>	331, 332, 344, 353, 354, 359, 367, 368, 371, 372, 375	11
<i>Cluster 3</i>	335, 336, 340, 346, 348, 355, 370	7

Langkah selanjutnya adalah menghitung rasio antara besaran BCV (*Between Cluster Variation*) dengan WCV (*Within Cluster Variation*). Untuk menentukan besaran BVC menggunakan rumus (3.2) dan WCV menggunakan rumus (3.3). Tabel 4.7 menunjukkan penghitungan BVC pada iterasi ke-I.

Tabel 4.7 Penghitungan BVC pada Iterasi ke-I

<i>Cluster</i>		Jarak Antar Cluster
C1	C2	2152020,81
C1	C3	4655,200001
C2	C3	2156676,01
TOTAL BCV		4313352,02

Berdasarkan tabel 4.7, Total BVC pada iterasi ke-I adalah 4313352,02. Setelah menentukan BVC, yang selanjutnya adalah menghitung WCV. Tabel 4.8 menunjukkan penghitungan WCV pada iterasi ke-I.

Tabel 4.8 Tabel Penghitungan WCV pada Iterasi ke-I

ID Reklame	Min Jarak	Kuadrat Min Jarak
725,7427926	526702,601	725,7427926
4,900058775	24,010576	4,900058775
0	0	0
766,4698251	587475,9928	766,4698251
562,7075384	316639,7737	562,7075384
2535,300705	6427749,666	2535,300705
0	0	0
141,1000519	19909,22464	141,1000519
1380,154055	1904825,215	1380,154055
2254,690582	5083629,62	2254,690582
2600,92966	6764835,098	2600,92966
688,6878855	474291,0037	688,6878855

ID Reklame	Min Jarak	Kuadrat Min Jarak
1,900001053	3,610004	1,900001053
708,0691908	501361,979	708,0691908
119,3000594	14232,50416	119,3000594
593,9370263	352761,1912	593,9370263
22,50001076	506,250484	22,50001076
191,0188269	36488,19223	191,0188269
723,5625774	523542,8034	723,5625774
172,0695112	29607,91669	172,0695112
1553,263877	2412628,673	1553,263877
90,40007832	8172,174161	90,40007832
738,2764628	545052,1355	738,2764628
1447,523394	2095323,976	1447,523394
828,0146129	685608,1992	828,0146129
605,8105527	367006,4258	605,8105527
605,5366996	366674,6945	605,5366996
11,10001302	123,210289	11,10001302
0	0	0
601,1739848	361410,16	601,1739848
605,2635321	366343,9433	605,2635321
722,374453	521824,8504	722,374453
280,1960993	78509,85409	280,1960993
708,8687172	502494,8583	708,8687172
2254,391219	5082279,77	2254,391219
737,3041087	543617,3487	737,3041087
1119,400466	1253057,402	1119,400466
606,4390721	367768,3482	606,4390721
62,90003155	3956,413969	62,90003155
24,50001176	600,250576	24,50001176
733,9710353	538713,4806	733,9710353
605,0299359	366061,2233	605,0299359
604,7192681	365685,3932	604,7192681
9,400005319	88,3601	9,400005319
	WCV	40397587,8

Berdasarkan Tabel 4.7 dan Tabel 4.8, dapat diketahui bahwa Total BCV adalah 4313352,02 dan total WCV adalah 40397587,8. Dari data tersebut dapat diketahui besar rasio menggunakan rumus (3.4), sehingga didapatkan rasio sebesar 0,106772514. Karena ini merupakan Iterasi pertama maka dilakukan Iterasi selanjutnya, yaitu Iterasi ke-II. Sebelum melakukan Iterasi kedua dilakukan Pembaharuan *Centroid* dengan rumus (3.5). Hasil pembaharuan *Centroid* sebagaimana terlihat pada Tabel 4.9.

Tabel 4.9 Pembentukan *Cluster* Baru pada Iterasi ke-II

<i>Cluster</i>	Jarak dengan <i>Traffic light</i>	Jarak dengan Pasar	Tingkat Keramaian
1	0,632615385	452,2807692	1811,030769
2	0,415636364	149,7636364	3460,545455
3	0,635857143	5433,614286	1953,757143

Selanjutnya adalah mengalokasikan masing-masing data ke *centroid*/rata-rata terdekat. Hasil penghitungan jarak ke *centroid* pada Iterasi ke-II menggunakan rumus (3.1) sebagaimana terlihat pada Tabel 4.10.

Tabel 4.10 Hasil Penghitungan Jarak ke *Centroid* pada Iterasi ke-II

ID Reklame	C1	C2	C3
330	832,3060574	2358,783833	5489,38695
331	1516,539412	243,7563647	5579,802909
332	1515,115056	240,9640403	5575,039487
333	624,9667872	1268,110007	4685,161968
334	613,7479262	1108,715363	5286,428515
335	6977,538992	7468,242105	2002,529084
336	4442,249999	5028,740461	562,1451317
337	354,4668288	1666,843232	5337,790621
338	1209,84828	2682,762769	4087,123591
339	2043,465127	2959,684739	2960,288472
340	7003,635324	7369,10875	2040,603559
343	733,4889284	2356,790812	5252,200405
344	1515,665638	242,0388379	5576,886501
345	706,7682864	2368,352414	5100,379402
346	4561,549204	5141,454779	449,0067476
347	683,6162042	1114,652373	5414,073552
348	4419,750155	5007,516723	583,758108
349	372,8119343	1813,237895	5325,222348
350	828,6083653	2358,379714	5482,471055
351	343,8964069	1812,945466	5292,776434
352	1343,299465	2441,87149	3660,839259
353	1491,483351	204,224803	5487,202305
354	2205,290938	531,7288335	5688,183166
355	3153,05547	3584,586502	1916,819347
356	672,3068507	1312,124484	4599,279701
357	703,7700901	1118,449381	5446,186879
358	703,322427	1118,357116	5445,488753

ID Reklame	C1	C2	C3
359	1518,362419	247,3835065	5585,830438
360	213,6217154	1668,449655	5196,755132
361	696,0904754	1116,915188	5434,119327
362	702,8751758	1118,265281	5444,790626
363	826,5667854	2358,163548	5478,618009
364	480,2445463	1819,063305	5441,533777
365	706,5451277	2368,720094	5097,026739
366	2043,166088	2959,447614	2960,586864
367	2206,31652	532,1144481	5694,532319
368	1602,382831	1003,691473	4494,542569
369	704,7948631	1118,66189	5447,7826
370	4379,350437	4969,436507	622,7231332
371	1508,211873	228,0710826	5551,225933
372	2210,721949	534,5884363	5720,499928
373	702,4921465	1118,186908	5444,192233
374	701,9819157	1118,082907	5443,394376
375	1517,860264	246,3787691	5584,177692

Dari tabel 4.10 Hasil Penghitungan Jarak ke *Centroid* pada Iterasi ke-II didapatkan keanggotaan sebagaimana terlihat pada Tabel 4.11.

Tabel 4.11 Tabel Keanggotaan pada Iterasi ke-II

Cluster	ID Reklame	Jumlah
Cluster 1	330, 333, 334, 337, 338, 339, 343, 345, 347, 349, 350, 351, 352, 356, 357, 358, 360, 361, 362, 363, 364, 365, 366, 369, 373, 374	26
Cluster 2	331, 332, 344, 353, 354, 359, 367, 368, 371, 372, 375	11
Cluster 3	335, 336, 340, 346, 348, 355, 370	7

Langkah selanjutnya adalah menghitung rasio antara besaran BCV dengan WCV pada Iterasi ke-II. Untuk menentukan besaran BVC menggunakan rumus (3.2) dan WCV menggunakan rumus (3.3). Tabel 4.12 menunjukkan penghitungan BVC pada Iterasi ke-II.

Tabel 4.12 Tabel Penghitungan BVC pada Iterasi ke-II

Cluster		Jarak Antar Cluster
C1	C2	2721201,214
C1	C3	25352,15125
C2	C3	2275694,867
TOTAL BCV		5022248,232

Berdasarkan tabel 4.12, Total BVC pada iterasi ke-I adalah 5022248,232. Setelah menentukan BVC, yang selanjutnya adalah menghitung WCV. Tabel 4.13 menunjukkan penghitungan WCV pada iterasi ke-II.

Tabel 4.13 Tabel Penghitungan WCV pada Iterasi ke-II

ID Reklame	Min Jarak	Kuadrat Min Jarak
330	832,3060574	692733,3732
331	243,7563647	59417,16533
332	240,9640403	58063,66872
333	624,9667872	390583,4851
334	613,7479262	376686,5169
335	2002,529084	4010122,731
336	562,1451317	316007,1491
337	354,4668288	125646,7327
338	1209,84828	1463732,86
339	2043,465127	4175749,725
340	2040,603559	4164062,885
343	733,4889284	538006,0082
344	242,0388379	58582,79908
345	706,7682864	499521,4107
346	449,0067476	201607,0594
347	683,6162042	467331,1146
348	583,758108	340773,5286
349	372,8119343	138988,7384
350	828,6083653	686591,8231
351	343,8964069	118264,7387
352	1343,299465	1804453,452
353	204,224803	41707,77017
354	531,7288335	282735,5524
355	1916,819347	3674196,411
356	672,3068507	451996,5015
357	703,7700901	495292,3398
358	703,322427	494662,4363
359	247,3835065	61198,59927
360	213,6217154	45634,2373
361	696,0904754	484541,9499
362	702,8751758	494033,5128
363	826,5667854	683212,6508
364	480,2445463	230634,8242
365	706,5451277	499206,0175
366	2043,166088	4174527,663
367	532,1144481	283145,7859
368	1003,691473	1007396,573

ID Reklame	Min Jarak	Kuadrat Min Jarak
369	704,7948631	496735,7991
370	622,7231332	387784,1007
371	228,0710826	52016,41871
372	534,5884363	285784,7962
373	702,4921465	493495,2159
374	701,9819157	492778,61
375	246,3787691	60702,49785
TOTAL WCV		36360347,23

Berdasarkan Tabel 4.12 dan Tabel 4.13 dapat diketahui bahwa Total BCV adalah 5022248,232 dan total WCV adalah 36360347,23. Dari data tersebut dapat diketahui besar rasio menggunakan rumus (3.4), sehingga didapatkan rasio pada Iterasi ke-II sebesar 0,138124320. Dari perbandingan antara rasio pada iterasi I dengan rasio pada Iterasi ke-II data diketahui bahwa Rasio II > Rasio I ($0,106772514 > 0,138124320$). Karena Rasio Sekarang lebih besar dari rasio sebelumnya, maka proses iterasi dilakukan kembali.

Selanjutnya adalah melakukan Iterasi ketiga. Sebelum melakukan Iterasi ke-III dilakukan Pembaharuan *Centroid* menggunakan rumus (3.5). Pembaharuan *Centroid* sebagaimana terlihat pada Tabel 4.14.

Tabel 4.14 Pembentukan *Cluster* Baru pada Iterasi ke-III

<i>Cluster</i>	Jarak dengan <i>Traffic light</i>	Jarak dengan Pasar	Tingkat Keramaian
1	0,632615385	452,2807692	1811,030769
2	0,415636364	149,7636364	3460,545455
3	0,635857143	5433,614286	1953,757143

Selanjutnya adalah mengalokasikan masing-masing data ke *centroid*/rata-rata terdekat. Hasil penghitungan jarak ke *centroid* pada Iterasi ke-III menggunakan rumus (3.1) sebagaimana terlihat pada Tabel 4.15.

Tabel 4.15 Hasil Penghitungan Jarak ke *Centroid* pada Iterasi ke-III

ID Reklame	C1	C2	C3
330	832,3060574	2358,783833	5489,38695
331	1516,539412	243,7563647	5579,802909
332	1515,115056	240,9640403	5575,039487
333	624,9667872	1268,110007	4685,161968
334	613,7479262	1108,715363	5286,428515
335	6977,538992	7468,242105	2002,529084
336	4442,249999	5028,740461	562,1451317

ID Reklame	C1	C2	C3
337	354,4668288	1666,843232	5337,790621
338	1209,84828	2682,762769	4087,123591
339	2043,465127	2959,684739	2960,288472
340	7003,635324	7369,10875	2040,603559
343	733,4889284	2356,790812	5252,200405
344	1515,665638	242,0388379	5576,886501
345	706,7682864	2368,352414	5100,379402
346	4561,549204	5141,454779	449,0067476
347	683,6162042	1114,652373	5414,073552
348	4419,750155	5007,516723	583,758108
349	372,8119343	1813,237895	5325,222348
350	828,6083653	2358,379714	5482,471055
351	343,8964069	1812,945466	5292,776434
352	1343,299465	2441,87149	3660,839259
353	1491,483351	204,224803	5487,202305
354	2205,290938	531,7288335	5688,183166
355	3153,05547	3584,586502	1916,819347
356	672,3068507	1312,124484	4599,279701
357	703,7700901	1118,449381	5446,186879
358	703,322427	1118,357116	5445,488753
359	1518,362419	247,3835065	5585,830438
360	213,6217154	1668,449655	5196,755132
361	696,0904754	1116,915188	5434,119327
362	702,8751758	1118,265281	5444,790626
363	826,5667854	2358,163548	5478,618009
364	480,2445463	1819,063305	5441,533777
365	706,5451277	2368,720094	5097,026739
366	2043,166088	2959,447614	2960,586864
367	2206,31652	532,1144481	5694,532319
368	1602,382831	1003,691473	4494,542569
369	704,7948631	1118,66189	5447,7826
370	4379,350437	4969,436507	622,7231332
371	1508,211873	228,0710826	5551,225933
372	2210,721949	534,5884363	5720,499928
373	702,4921465	1118,186908	5444,192233
374	701,9819157	1118,082907	5443,394376
375	1517,860264	246,3787691	5584,177692

Dari tabel 4.15 Hasil Penghitungan Jarak ke *Centroid* pada Iterasi ke-III didapatkan keanggotaan sebagaimana terlihat pada Tabel 4.16.

Tabel 4.16 Keanggotaan pada Iterasi ke-III

<i>Cluster</i>	ID Reklame	Jumlah
<i>Cluster 1</i>	330, 333, 334, 337, 338, 339, 343, 345, 347, 349, 350, 351, 352, 356, 357, 358, 360, 361, 362, 363, 364, 365, 366, 369, 373, 374	26
<i>Cluster 2</i>	331, 332, 344, 353, 354, 359, 367, 368, 371, 372, 375	11
<i>Cluster 3</i>	335, 336, 340, 346, 348, 355, 370	7

Langkah selanjutnya adalah menghitung rasio antara besaran BVC dengan WCV Pada Iterasi ke-III. Untuk menentukan besaran BVC menggunakan rumus (3.2) dan WCV menggunakan rumus (3.3). Tabel 4.17 menunjukkan penghitungan BVC pada iterasi ke-III.

Tabel 4.17 Penghitungan BVC pada Iterasi ke-III

<i>Cluster</i>		Jarak Antar Cluster
C1	C2	2721201,214
C1	C3	25352,15125
C2	C3	2275694,867
TOTAL BVC		5022248,232

Berdasarkan tabel 4.17, Total BVC pada iterasi ke-III adalah 5022248,232. Setelah menentukan BVC, yang selanjutnya adalah menghitung WCV. Tabel 4.13 menunjukkan penghitungan WCV pada iterasi ke-III. Tabel 4.18 menunjukkan penghitungan WCV pada iterasi ke-III.

Tabel 4.18 Penghitungan WCV pada Iterasi ke-III

ID Reklame	Min Jarak	Kuadrat Min Jarak
330	832,3060574	692733,3732
331	243,7563647	59417,16533
332	240,9640403	58063,66872
333	624,9667872	390583,4851
334	613,7479262	376686,5169
335	2002,529084	4010122,731
336	562,1451317	316007,1491
337	354,4668288	125646,7327
338	1209,84828	1463732,86
339	2043,465127	4175749,725
340	2040,603559	4164062,885
343	733,4889284	538006,0082

ID Reklame	Min Jarak	Kuadrat Min Jarak
344	242,0388379	58582,79908
345	706,7682864	499521,4107
346	449,0067476	201607,0594
347	683,6162042	467331,1146
348	583,758108	340773,5286
349	372,8119343	138988,7384
350	828,6083653	686591,8231
351	343,8964069	118264,7387
352	1343,299465	1804453,452
353	204,224803	41707,77017
354	531,7288335	282735,5524
355	1916,819347	3674196,411
356	672,3068507	451996,5015
357	703,7700901	495292,3398
358	703,322427	494662,4363
359	247,3835065	61198,59927
360	213,6217154	45634,2373
361	696,0904754	484541,9499
362	702,8751758	494033,5128
363	826,5667854	683212,6508
364	480,2445463	230634,8242
365	706,5451277	499206,0175
366	2043,166088	4174527,663
367	532,1144481	283145,7859
368	1003,691473	1007396,573
369	704,7948631	496735,7991
370	622,7231332	387784,1007
371	228,0710826	52016,41871
372	534,5884363	285784,7962
373	702,4921465	493495,2159
374	701,9819157	492778,61
375	246,3787691	60702,49785
TOTAL WCV		36360347,23

Berdasarkan tabel di atas dapat diketahui bahwa Total BCV adalah 5022248,232 dan total WCV adalah 36360347,23. Dari data tersebut dapat diketahui besar rasio menggunakan rumus (3.4), sehingga didapatkan rasio pada Iterasi ke-III sebesar

0,13812432. Dari perbandingan antara rasio pada Iterasi ke-II dengan rasio pada Iterasi ke-III data diketahui bahwa Rasio III = Rasio II (0,13812432= 0,13812432). Karena rasio sekarang tidak lebih besar dari rasio sebelumnya, maka proses iterasi dihentikan, sehingga didapatkan keanggotaan akhir sebagaimana terlihat pada Tabel 4.19.

Tabel 4.19 Hasil Akhir Keanggotaan dari Pengelompokan Menggunakan Algoritma *K-means*

<i>Cluster</i>	ID Reklame	Jumlah
<i>Cluster 1</i>	330, 333, 334, 337, 338, 339, 343, 345, 347, 349, 350, 351, 352, 356, 357, 358, 360, 361, 362, 363, 364, 365, 366, 369, 373, 374	26
<i>Cluster 2</i>	331, 332, 344, 353, 354, 359, 367, 368, 371, 372, 375	11
<i>Cluster 3</i>	335, 336, 340, 346, 348, 355, 370	7

Berdasarkan Tabel 4.19, dapat diketahui bahwa *cluster* yang beranggotakan paling banyak adalah Cluster 1 yang terdiri dari 26 reklame yaitu reklame dengan id 330, 333, 334, 337, 338, 339, 343, 345, 347, 349, 350, 351, 352, 356, 357, 358, 360, 361, 362, 363, 364, 365, 366, 369, 373, 374, yang selanjutnya yang urutan kedua adalah *cluster 2* (dua) yang terdiri dari 11 reklame dengan id 331, 332, 344, 353, 354, 359, 367, 368, 371, 372, 375. Untuk *cluster* nomor 3 (tiga) berada pada urutan nomor 3 (tiga) yang terdiri dari 8 (delapan) reklame yaitu reklame dengan id 335, 336, 340, 346, 348, 355, 370. Berikut analisis dari masing masing *cluster*.

4.9 Analisis Pada Cluster I

Dari hasil *Cluster K-means* pada Reklame berdasar variabel Jarak dengan Pasar, Jarak dengan *Traffic light* dan Volume Kendaraan dihasilkan anggota dengan rincian sebagaimana terlihat pada Tabel 4.20.

Tabel 4.20 Hasil *Cluster K-means* Pada Cluster 1

No.	Id Reklame	Luas (m)	Jumlah Pajak Reklame (Rp) / Tahun	Kelas Wilayah	Kawasan Jalan	Jarak dengan <i>Traffic light</i>	Jarak dengan Pasar	Volume Kendaraan	Kecamatan
1	330	24	4,276,800	Wilayah I	Jalan Utama	0,26	10,100	1105,900	Kajen
2	333	24	4,276,800	Wilayah I	Jalan Utama	0,078	765,400	2351,900	Kedungwuni
3	334	24	4,276,800	Wilayah I	Jalan Utama	1,006	162,200	2351,900	Kedungwuni
4	337	24	4,276,800	Wilayah I	Jalan Utama	0,108	98,200	1794,500	Karang Anyar
5	338	24	4,276,800	Wilayah II	Jalan Utama	0,441	1435,400	1105,900	Kesesi
6	339	24	4,276,800	Wilayah II	Jalan Utama	0,238	2489,200	1647,600	Sragi
7	343	24	3,888,000	Wilayah I	Jalan Utama	0,02	250,300	1105,900	Kajen

No.	Id Reklame	Luas (m)	Jumlah Pajak Reklame (Rp) / Tahun	Kelas Wilayah	Kawasan Jalan	Jarak dengan Traffic light	Jarak dengan Pasar	Volume Kendaraan	Kecamatan
8	345	24	3,888,000	Wilayah I	Jalan Utama	0,134	404,200	1105,900	Kajen
9	347	24	4,276,800	Wilayah I	Jalan Utama	0,878	34,200	2351,900	Kedungwuni
10	349	24	3,888,000	Wilayah I	Jalan Utama	0,644	117,200	1647,600	Bojong
11	350	19	3,118,500	Wilayah I	Jalan Utama	0,287	17,100	1105,900	Kajen
12	351	24	4,276,800	Wilayah I	Jalan Utama	0,612	149,700	1647,600	Bojong
13	352	24	5,760,000	Wilayah I	Jalan Utama	2,547	1785,600	1647,600	Bojong
14	356	24	4,276,800	Wilayah I	Jalan Utama	1,695	851,600	2351,900	Kedungwuni
15	357	24	4,276,800	Wilayah I	Jalan Utama	0,842	2,000	2351,900	Kedungwuni
16	358	50	8,100,000	Wilayah I	Jalan Utama	0,841	2,700	2351,900	Kedungwuni
17	360	15	2,430,000	Wilayah I	Jalan Utama	0,229	239,300	1794,500	Karang Anyar
18	361	5	729,000	Wilayah I	Jalan Utama	0,829	14,100	2351,900	Kedungwuni
19	362	24	3,888,000	Wilayah I	Jalan Utama	0,84	3,400	2351,900	Kedungwuni
20	363	24	4,276,800	Wilayah I	Jalan Utama	0,249	21,000	1105,900	Kajen
21	364	24	4,276,800	Wilayah I	Jalan Utama	0,762	0,700	1647,600	Bojong
22	365	24	8,553,600	Wilayah I	Jalan Utama	0,138	407,600	1105,900	Kajen
23	366	24	8,553,600	Wilayah II	Jalan Utama	0,238	2488,900	1647,600	Sragi
24	369	14	270,000	Wilayah I	Jalan Utama	0,844	0,400	2351,900	Kedungwuni
25	373	24	3,888,000	Wilayah I	Jalan Utama	0,84	4,000	2351,900	Kedungwuni
26	374	24	3,888,000	Wilayah I	Jalan Utama	0,848	4,800	2351,900	Kedungwuni
Jumlah		607	112,164,300			5259,2	11759,3	47086,8	
Mean		23,34615	4,314,012			202,2769	452,2808	1811,031	
Median		24	4,276,800			0,628	107,7	1721,05	
Max		50	8,553,600			2547	2489,2	2351,9	
Min		5	270,000			0,02	0,4	1105,9	
Wilayah I		7							
Wilayah II		3							
Wilayah III		0							
Jalan Utama		10							
Bukan Jalan Utama		0							

Berdasarkan Tabel 4.20, dapat diketahui bahwa jumlah anggota pada *cluster* 1 adalah 26 reklame, sedangkan rata-rata penerimaan pajak pada *cluster* 1 adalah Rp4,314,012,-. Pada *cluster* 1 jumlah reklame yang berada pada kelas wilayah 1 adalah sejumlah 23 reklame, sedangkan pada kelas wilayah II ada 3 (tiga) reklame dan pada kelas wilayah III ada 0 reklame. Berdasarkan kawasan jalan, semua reklame pada *cluster* 1 berada pada jalan utama. Berdasarkan kecamatannya pada *cluster* 1 yang paling banyak adalah Kecamatan Kedungwuni, dimana jumlah reklame pada *cluster* 1 yang berada pada Kecamatan Kedungwuni ada 11 (sebelas) reklame, kemudian di kecamatan Kajen ada 5 (lima) reklame, di Kecamatan Bojong ada 4 (empat) reklame, di Kecamatan Karanganyar

ada 2 (dua) reklame, di Kecamatan Sragi ada 2 (dua) reklame dan di Kecamatan Kesesi ada 1 (satu) reklame. Besaran pajak reklame tertinggi pada *Cluster 1*, dimana reklame tersebut berada pada kecamatan Kajen dan Sragi yang mempunyai kedekatan dengan *traffic light* yang cukup dekat.

4.10 Analisis Pada *Cluster 2*

Dari hasil *Cluster K-means* pada Reklame berdasar variabel Jarak dengan Pasar, Jarak dengan *Traffic light* dan Volume Kendaraan dihasilkan anggota dengan rincian sebagaimana terlihat pada Tabel 4.21.

Tabel 4.21 Hasil *Cluster K-means* Pada *Cluster 2*

No.	Id Reklame	Luas (m)	Jumlah Pajak Reklame (Rp) / Tahun	Kelas Wilayah	Kawasan Jalan	Jarak dengan <i>Traffic light</i>	Jarak dengan Pasar	Volume Kendaraan	Kecamatan
1	331	32	5,184,000	Wilayah I	Jalan Utama	0,073	9,200	3261,400	Wiradesa
2	332	15	2,430,000	Wilayah I	Jalan Utama	0,049	14,100	3261,400	Wiradesa
3	344	18	2,916,000	Wilayah I	Jalan Utama	0,051	12,200	3261,400	Wiradesa
4	353	50	8,910,000	Wilayah I	Jalan Utama	0,168	104,500	3261,400	Wiradesa
5	354	24	4,276,800	Wilayah I	Jalan Utama	0,99	123,000	3991,600	Tirto
6	359	50	16,200,000	Wilayah I	Jalan Utama	0,066	3,000	3261,400	Wiradesa
7	367	50	17,820,000	Wilayah I	Jalan Utama	0,997	116,200	3991,600	Tirto
8	368	15	2,430,000	Wilayah I	Jalan Utama	1,07	1133,500	3261,400	Wiradesa
9	371	32	5,184,000	Wilayah I	Jalan Utama	0,025	38,600	3261,400	Wiradesa
10	372	24	3,888,000	Wilayah I	Jalan Utama	1,024	88,400	3991,600	Tirto
11	375	24	4,276,800	Wilayah I	Jalan Utama	0,059	4,700	3261,400	Wiradesa
Jumlah		334	73,515,600			1027,548	1647,4	38066	
Mean		30,36364	6,683,236			93,41345	149,7636	3460,545	
Median		24	4,276,800			0,073	38,6	3261,4	
Max		50	17,820,000			1024	1133,5	3991,6	
Min		15	2,430,000			0,025	3	3261,4	
Wilayah I		11							
Wilayah II		0							
Wilayah II		0							
Jalan Utama		11							
Bukan Jalan Utama		0							

Berdasarkan Tabel 4.21 Hasil *clustering* menggunakan algoritma *k-means* pada *cluster 2*, dapat diketahui bahwa Jumlah Anggota Pada *cluster 2* adalah 11 reklame,

sedangkan rata-rata penerimaan pajak pada *cluster 2* adalah Rp6,683,236,-. Pada *cluster 2* jumlah reklame yang berada pada wilayah 1 (satu) adalah sejumlah 11 (sebelas) reklame, sedangkan wilayah II ada 0 reklame dan di wilayah III ada 0 reklame. Berdasarkan kawasan jalan semua reklame di *cluster 2* berada pada jalan utama. Berdasarkan kecamatannya pada *cluster 2* hanya berada pada 2 (dua) kecamatan yaitu 8 (delapan) di Kecamatan Wiradesa dan 3 (tiga) di Kecamatan Tirto. *Cluster 2* merupakan *cluster* yang mempunyai rata-rata penerimaan pajak reklame yang paling tinggi daripada *cluster* yang lain. Reklame yang ada pada *cluster* ini berada di Kecamatan Wiradesa dan Tirto, dimana keduanya merupakan kecamatan di Kabupaten Pekalongan yang dilewati Jalur Pantura, sehingga kecamatan yang dilewati jalur pantura mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan. Pada *cluster 2* merupakan *cluster* yang mempunyai rata-rata kedekatan dengan pasar dan *traffic light* yang paling dekat, sehingga dapat disimpulkan bahwa kedekatan dengan pasar dan *traffic light* mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan. Pada *cluster 2* mempunyai rata-rata volume kendaraan yang paling tinggi dibandingkan dengan *cluster 1* ataupun *cluster 3*, dimana *cluster 1* mempunyai rata-rata volume kendaraan sebesar 3460,545455 sedangkan *cluster 3* sebesar 1953,757143 dan *cluster 1* sebesar 1811,030769, sehingga bisa disimpulkan bahwa volume kendaraan mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan.

4.11 Analisis Pada Cluster 3

Dari hasil *clustering* menggunakan algoritma *k-means* pada reklame berdasar variabel jarak dengan pasar, jarak dengan *traffic light* dan volume kendaraan dihasilkan anggota dengan rincian sebagaimana terlihat pada Tabel 4.22.

Tabel 4.22 Hasil *Cluster K-means* pada *Cluster 3*

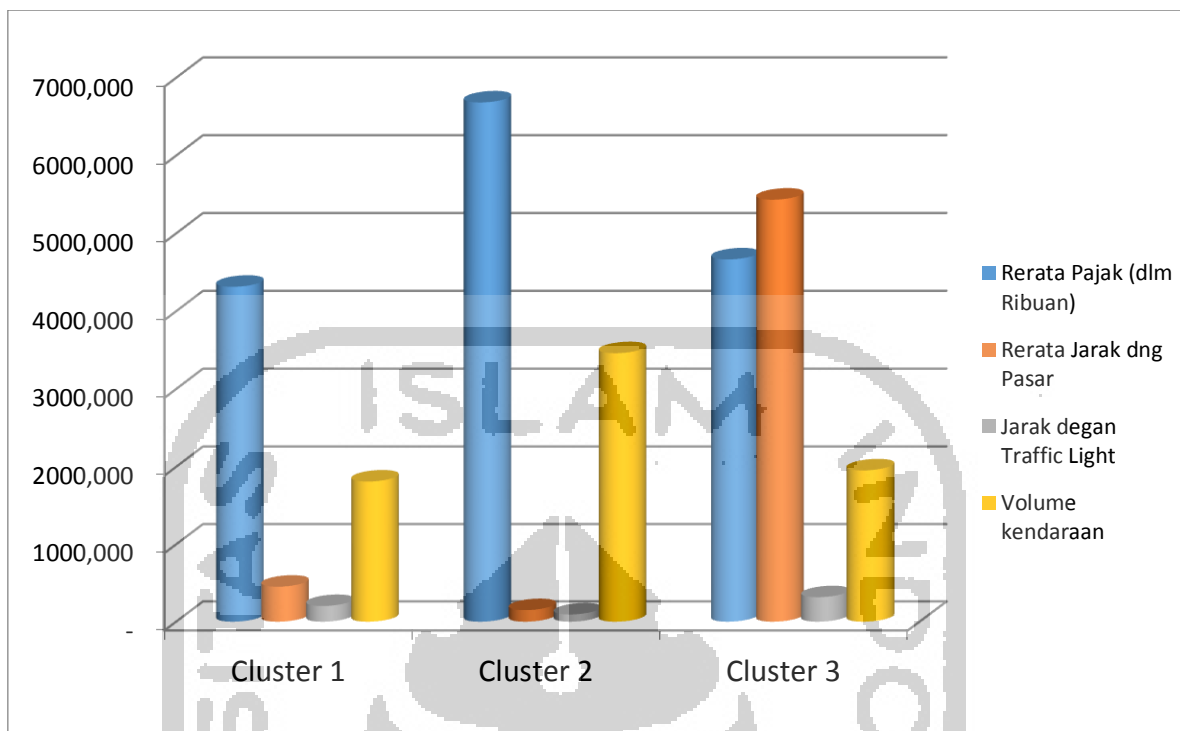
No.	Id Reklame	Luas (m)	Jumlah Pajak Reklame (Rp) / Tahun	Kelas Wilayah	Kawasan Jalan	Jarak dengan Traffic light	Jarak dengan Pasar	Volume Kendaraan	Kecamatan
1	335	24	4,276,800	Wilayah I	Jalan Utama	2,213	7429,800	1794,500	Wonopringgo
2	336	24	4,276,800	Wilayah I	Jalan Utama	0,322	4894,500	1794,500	Wonopringgo
3	340	24	4,276,800	Wilayah II	Jalan Utama	0,025	7435,000	2351,900	Doro
4	346	24	3,888,000	Wilayah I	Jalan Utama	0,203	5013,800	1794,500	Wonopringgo
5	348	24	3,888,000	Wilayah I	Jalan Utama	0,344	4872,000	1794,500	Wonopringgo
6	355	24	4,276,800	Wilayah I	Jalan Utama	0,959	3558,600	2351,900	Buaran
7	370	24	7,776,000	Wilayah I	Jalan Utama	0,385	4831,600	1794,500	Wonopringgo

No.	Id Reklame	Luas (m)	Jumlah Pajak Reklame (Rp) / Tahun	Kelas Wilayah	Kawasan Jalan	Jarak dengan Traffic light	Jarak dengan Pasar	Volume Kendaraan	Kecamatan
Jumlah		168	32,659,200			2215,238	38035,3	13676,3	
Mean		24	4,665,600			316,4626	5433,614	1953,757	
Median		24	4,276,800			0,344	4894,5	1794,5	
Max		24	7,776,000			2213	7435	2351,9	
Min		24	3,888,000			0,025	3558,6	1794,5	
Wilayah I		6							
Wilayah II		1							
Wilayah III		0							
Jalan Utama		7							
Bukan Jalan Utama		0							

Berdasarkan Tabel 4.23 Hasil *clustering* menggunakan algoritma *k-means* pada *cluster* 3, dapat diketahui bahwa jumlah anggota pada *cluster* 3 adalah 7 reklame, sedangkan rata-rata penerimaan pajak pada *cluster* 3 adalah Rp4,665,600,-. Pada *cluster* 3 jumlah reklame yang berada pada wilayah 1 adalah sejumlah 6 reklame, sedangkan wilayah II ada 1 reklame dan di Wilayah III ada 0 reklame. Berdasarkan kawasan jalan semua reklame di *cluster* 3 berada pada jalan utama. Berdasarkan kecamatannya pada *cluster* 3 yang paling banyak adalah Kecamatan Wonopringgo dimana jumlah reklame pada *cluster* 3 yang berada pada Kecamatan Wonopringgo ada 5 reklame, kemudian di kecamatan Doro ada 1 reklame, dan di Kecamatan Buaran ada 1 reklame.

4.12 Pembahasan

Penelitian ini telah dikembangkan sebuah SIG yang mampu menyajikan data-data spasial, serta mampu melakukan analisis *clustering* menggunakan algoritma *k-means*. Dari hasil pembangunan sistem kemudian diimplementasikan untuk melakukan analisis dengan menggunakan algoritma *k-means*. Namun, sistem yang dibangun belum bisa melakukan analisis terkait penentuan jumlah *cluster* yang optimal. Untuk menentukan jumlah *cluster* yang optimal, digunakan RStudio dengan memanfaatkan package dari library NBClust menggunakan semua indeks. Berdasarkan perhitungan pada package NBClust jumlah *cluster* yang optimal adalah 3. Jumlah *cluster* yang paling banyak berada pada *cluster* 1 terdiri dari 26 reklame, kemudian *cluster* 2 terdiri dari 11 reklame dan *cluster* 3 terdiri dari 9 reklame. Gambar 4.52 menunjukkan grafik hasil *clustering* menggunakan algoritma *k-means* berdasarkan rata-rata besaran pajaknya.

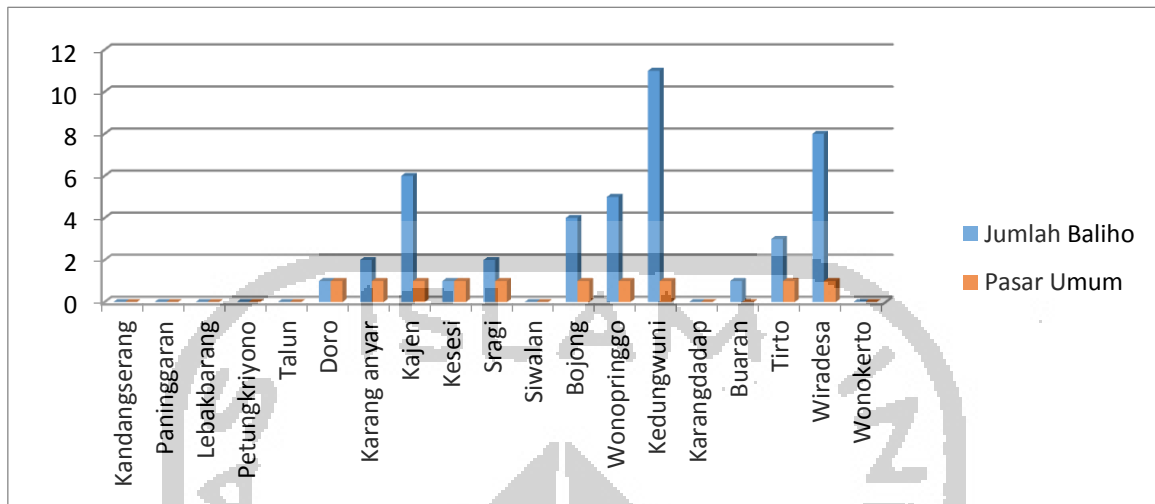


Gambar 4.52 Grafik Hasil *Clustering* menggunakan Algoritma *K-means* berdasarkan Rata-Rata Besaran Pajak

Berdasarkan Gambar 4.52, *cluster 2* merupakan *cluster* yang mempunyai rata-rata penerimaan pajak reklame yang paling tinggi daripada *cluster* yang lain. Reklame yang ada pada *cluster* ini berada di Kecamatan Wiradesa dan Tirto, dimana keduanya merupakan kecamatan di Kabupaten Pekalongan yang dilewati Jalur Pantura. Sehingga Kecamatan yang dilewati jalur pantura mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan. *Cluster 2* merupakan *cluster* yang mempunyai rata-rata kedekatan dengan pasar dan *traffic light* yang paling dekat, sehingga dapat disimpulkan bahwa kedekatan dengan pasar dan *traffic light* mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan. Begitu pula untuk volume kendaraan, yang paling tinggi adalah *cluster 2*, sehingga bisa disimpulkan bahwa volume kendaraan mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan.

Berdasarkan penelitian dapat diketahui bahwa di Kabupaten Pekalongan terdiri dari 19 Kecamatan, dimana dari 19 kecamatan tersebut ada 8 kecamatan yang tidak ada pasar dengan jenis pasar umum. Berdasarkan pengolahan data, reklame dengan jenis baliho

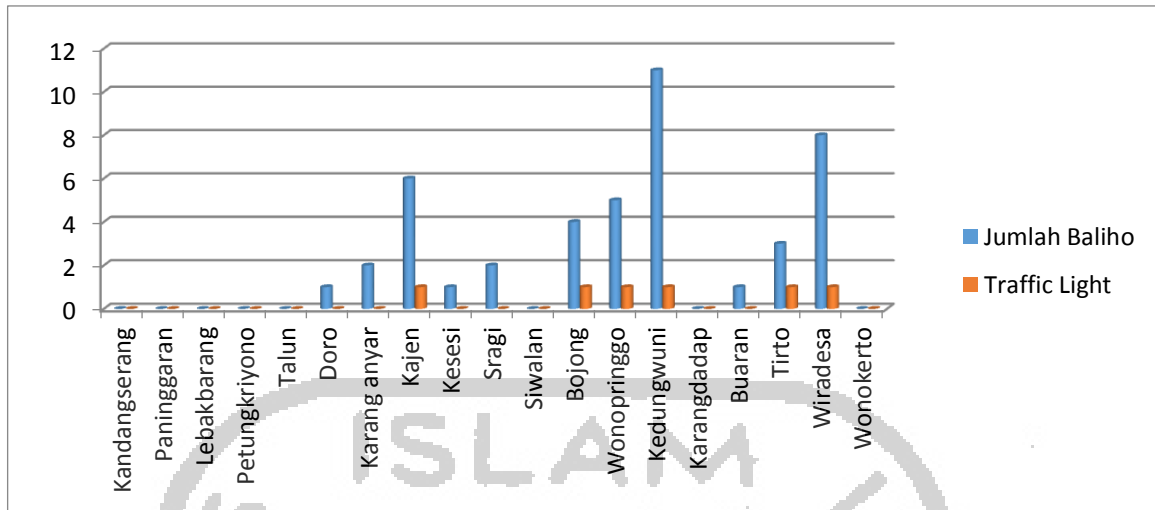
tersebar pada kecamatan yang mempunyai pasar umum. Gambar 4.53 menunjukkan grafik dari persebaran reklame baliho per kecamatan :



Gambar 4.53 Grafik Persebaran Reklame dan Pasar Per-Kecamatan

Berdasarkan Gambar 4.53, dapat terlihat bahwa reklame tersebar pada kecamatan yang ada pasar umumnya. Ada 1 kecamatan dimana pada pasar tersebut tidak ada pasar umumnya namun ada reklame balihonya. Berdasarkan penelitian di lapangan bahwa di Kecamatan Buaran merupakan perbatasan antara antara Kabupaten Pekalongan dan Kota Pekalongan. Pada perbatasan tersebut ada salah satu pasar yang berada di kawasan Pekalongan Kota yaitu “Pasar Pringlangu” yang berdekatan dengan Kecamatan Buaran. Berdasarkan kondisi tersebut dapat disimpulkan bahwa persebaran reklame baliho dapat dikaitkan dengan adanya pasar umum pada Kecamatan.

Berdasarkan pengolahan data persebaran reklame juga berkaitan dengan persebaran *traffic light* yang ada pada suatu kecamatan. Gambar 4.52 menunjukkan grafik dari persebaran reklame dan *traffic light* per-kecamatan.



Gambar 4.54 Grafik Persebaran Reklame dan *Traffic light* Per-Kecamatan

Berdasarkan Gambar 4.54, dapat terlihat bahwa sebagian besar reklame tersebar pada kecamatan yang ada *traffic light*nya. Ada beberapa kecamatan dimana pada kecamatan tidak ada *traffic light*nya namun ada reklamenya. Pada kecamatan yang tidak ada *traffic light*nya namun ada reklamenya, hal itu terjadi karena pada kecamatan tersebut ada pasar umumnya.

4.13 Ringkasan Penelitian dan Rekomendasi

Analisis spasial dengan menggunakan metode *clustering* terhadap data reklame dilakukan dengan menambahkan variabel spasial yaitu jarak dengan pasar, jarak dengan *traffic light*, dan volume kendaraan. Data-data spasial diinput kedalam basisdata SIG melalui aplikasi berbasis web yang dibangun dengan memanfaatkan google map. Data yang telah diinput kedalam basisdata SIG, kemudian diseragamkan dan dilakukan *cleaning*. Setelah data siap kemudian data diolah menggunakan menggunakan metode *clustering*, untuk mendapatkan pengetahuan baru terhadap data reklame tersebut.

Berdasarkan hasil *clustering* pada *cluster 2* berada pada jalan utama. Berdasarkan kecamatannya pada *cluster 2* hanya berada pada 2 kecamatan yaitu 8 di Kecamatan Wiradesa dan 3 di Kecamatan Tirto. *Cluster 2* merupakan *cluster* yang mempunyai rata-rata penerimaan pajak reklame yang paling tinggi daripada *cluster* yang lain. Reklame yang ada pada *cluster* ini berada di Kecamatan Wiradesa dan Tirto, dimana keduanya merupakan kecamatan di Kabupaten Pekalongan yang dilewati Jalur Pantura. Sehingga Kecamatan yang dilewati jalur pantura mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan.

Cluster 2 merupakan *cluster* yang mempunyai rata-rata kedekatan dengan pasar dan *traffic light* yang paling dekat, sehingga dapat disimpulkan bahwa kedekatan dengan pasar dan *traffic light* mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan. Pada *Cluster 2* mempunyai rata-rata Volume kendaraan yang paling tinggi dibandingkan dengan *cluster 1* ataupun *cluster 3*, dimana *cluster 1* mempunyai rata-rata volume kendaraan sebesar 3460,545455 sedangkan *cluster 3* sebesar 1953,757143 dan *cluster 1* sebesar 1811,030769, sehingga bisa disimpulkan bahwa volume kendaraan mempunyai korelasi dengan tingginya besaran pajak reklame di Kabupaten Pekalongan.

Berdasarkan hasil analisis, rata-rata reklame tersebar pada kecamatan yang ada pasar umumnya. Ada 1 kecamatan dimana pada pasar tersebut tidak ada pasar umumnya namun ada reklame balihonya. Berdasarkan penelitian di lapangan bahwa di Kecamatan Buaran merupakan perbatasan antara antara Kabupaten Pekalongan dan Kota Pekalongan. Pada perbatasan tersebut ada salah satu pasar yang berada di kawasan Pekalongan Kota yaitu “Pasar Pringlangu” yang berdekatan dengan Kecamatan Buaran. Berdasarkan kondisi tersebut dapat disimpulkan bahwa persebaran reklame baliho dapat dikaitkan dengan adanya pasar umum pada Kecamatan.

Ada beberapa kecamatan dimana pada kecamatan tidak ada *traffic light*nya namun ada reklamenya. Pada kecamatan yang tidak ada *traffic light*nya namun ada reklamenya, hal itu terjadi karena pada kecamatan tersebut ada pasar umumnya.

Dari beberapa temuan di atas dapat dijadikan rekomendasi bagi Pemerintah Daerah Kabupaten Pekalongan, khususnya BPKD yang merupakan OPD (Organisasi Perangkat Daerah) yang menangani pajak reklame. Rekomendasi tersebut diantaranya adalah :

1. Hasil analisis persebaran reklame di Kabupaten Pekalongan dapat dijadikan sebagai acuan bagi petugas pengelola pajak reklame khususnya petugas pendataan reklame pada Bidang Perencanaan dan Penetapan BPKD Kabupaten Pekalongan dalam mengetahui potensi-potensi reklame yang tersebar di wilayah Kabupaten Pekalongan.
2. Hasil analisis persebaran reklame dapat dijadikan sebagai acuan bagi petugas pemeriksa pajak dalam mengetahui titik-titik reklame yang tersebar di wilayah Kabupaten Pekalongan.
3. Peta spasial persebaran reklame dapat dijadikan acuan untuk OPD yang menerbitkan IMB (Ijin Mendirikan Bangunan) dalam hal ini reklame, sehingga dapat memudahkan petugas dalam mengetahui penyimpangan-penyimpangan dalam penempatan titik reklame.

4. Hasil analisis persebaran reklame menggunakan *clustering k-means* dapat dijadikan sebagai acuan bagi pejabat yang berwenang dalam pengambilan keputusan, karena dapat menentukan kelompok kecamatan yang mempunyai besaran pajak yang tinggi.

