ICM Indonesian Journal of Computing and Modeling

e-ISSN : 2598-9421

Pemetaan Potensi Bencana di Jawa Tengah Menggunakan Google Maps API dan KML dengan Metode *K-Means*

Muhammad Fahmi Haryono Adi [1], Yerymia Alfa Susetyo [2]
[1,2,3)] Program Studi Sarjana Teknik Informatika Universitas Kristen Satya Wacana
Email : 672015043@student.uksw.edu[1], yeremia.alfa@uksw.edu [2]

Abstrak — Sistem Informasi Geografis (SIG) merupakan sistem informasi berbasis komputer yang digunakan untuk mengolah, dan menghasilkan data geografis untuk pengambilan keputusan dalam suatu perencanaan. Indonesia memiliki iklim dengan curah hujan tinggi dan kemarau yang panjang akan menambah potensi terjadinya bencana seperti banjir, tanah longsor, dan kekeringan. Penelitian ini bertujuan untuk pemetaan potensi bencana di Jawa Tengah menggunakan Google Maps API dan KML dengan metode *K-Means*. Pengolahan data bencana alam dilakukan menggunakan metode *K-Means*, karena metode ini mampu menjadi solusi atas penanggulangan bencana alam. Penelitian ini membahas tentang pengelompokan jumlah data bencana yang terjadi di Jawa Tengah. Data yang digunakan dalam penelitan ini dengan pengumpulan data peta wilayah Kota dan Kabupaten di Jawa Tengah menggunakan Google Maps API dan pengumpulan data titik koordinat. *Keyhole Markup Language* (KML) adalah format (XML) untuk visualisasi data geospatial. Dengan menggunakan Google Maps API dan KML dapat menghasilkan suatu peta yang memberikan informasi secara lengkap dan tepat, karena sistem yang berbasis web akan lebih mudah dan cepat diakses oleh pengguna dari berbagai tempat hanya dengan menggunakan web-browser.

Kata kunci: *Bencana, Google Maps API, SIG, K-Means, KML*

I. PENDAHULUAN

Secara geografis Indonesia terletak di daerah khatulistiwa dengan morfologi beragam dari daratan sampai pegunungan tinggi. Morfologi dipengaruhi oleh faktor geologi dengan adanya aktivitas pergerakan lempeng tektonik aktif di sekitar perairan Indonesia di antaranya lempeng Eurasia, Australia dan lempeng dasar Samudera Pasifik. Pergerakan lempeng-lempeng tektonik menyebabkan terbentuknya jalur gempa bumi, rangkaian gunung api aktif serta patahan geologi yang merupakan zona rawan bencana. Beberapa potensi bencana antara lain adalah bencana alam seperti gempa bumi, banjir, dan lain-lain. Potensi bencana yang ada di Indonesia dapat dikelompokkan menjadi dua kelompok utama, yaitu bahaya utama (*main hazard*) dan bahaya ikutan (*collateral hazard*). Berdasarkan indikator tersebut dapat disimpulkan bahwa Indonesia memiliki potensi bahaya utama (*main hazard potency*) yang tinggi [1].

Indonesia merupakan negara yang memiliki jumlah terbanyak gunung api aktif di dunia. Selain itu iklim dan cuaca dengan curah hujan tinggi dan kemarau yang panjang akan menambah potensi terjadinya bencana seperti banjir, tanah longsor, dan kekeringan. Bencana yang terjadi akan memberikan dampak dan traumatis bagi masyarakat serta konflik sosial dalam masyarakat apabila tidak dilakukan penanganan yang tepat saat bencana terjadi.

Sistem Informasi Geografis (SIG) merupakan sistem informasi berbasis komputer yang digunakan untuk mengolah, menganalisa, menyimpan, dan menghasilkan data bereferensi geografis untuk mendukung pengambilan

2598-9421 ©2018 Pusat Studi Sistem Informasi Pemodelan dan Mitigasi Tropis (*Simitro*) Universitas Kristen Satya Wacana.
This is an open access article under the CC BY license (https://creativecommons.org/licenses/by/4.0/)

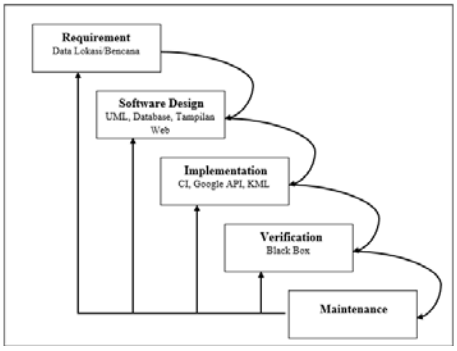
keputusan dalam suatu perencanaan. SIG merupakan teknologi mengenai geografis yang memiliki kemampuan menvisualisasi data spasial dengan atribut-atributnya dan menyajikan informasi dalam bentuk grafis menggunakan peta. Penelitian SIG ini telah banyak dilakukan oleh peneliti lain, beberapa penelitian SIG yang mencakup bencana di suatu daerah dengan studi kasus yang berbeda. *Application Programming Interface* (API) adalah sekumpulan komponen yang dibuat dalam kelas-kelas yang memiliki berbagai fungsi tertentu. Google Maps API merupakan sekumpulan komponen untuk melakukan berbagai fungsi terkait aplikasi pemetaan, Google Maps API dapat memudahkan pengembang sistem dalam merancang dan mengimplementasikan sebuah media informasi. Google Maps API menyediakan banyak fasilitas dan utilitas untuk memanipulasi peta dan menambahkan konten ke peta melalui berbagai layanan. *Keyhole Markup Language* (KML) adalah format *Extensible Markup Language* (XML) untuk visualisasi data geospatial. KML bisa menyimpan obyek utama seperti titik, garis dan luasan dalam format tertentu sehingga bisa ditampilkan dengan Google Maps [2].

K-Means merupakan salah satu metode data *clustering non hirarki* yang berusaha mempartisi data yang ada ke dalam bentuk satu atau lebih *cluster*/kelompok. Metode ini mempartisi data ke dalam *cluster*/kelompok sehingga data yang memiliki karakteristik yang sama dikelompokkan ke dalam satu *cluster* yang sama dan yang mempunyai karakteristik yang berbeda dikelompokkan ke dalam kelompok yang lain. Di antara semua algoritma berdasarkan pengelompokan, *K-Means* merupakan salah satu metode yang paling terkenal untuk menerapkan dataset ke dalam kelompok pola [3].

Berdasarkan latar belakang maka akan dilakukan perancangan Pemetaan Potensi Bencana di Jawa Tengah Menggunakan Google Maps API dan KML dengan Metode *K-Means*.

II. METODE PENELITIAN

Tahapan penelitian yang akan digunakan dalam penerapan implementasi Pemetaan Potensi Bencana di Jawa Tengah Menggunakan Google Maps API dan KML dengan Metode *K-Means* menggunakan konsep *waterfall* yaitu dengan melalui lima tahap yang berkesinambungan membentuk suatu siklus daur hidup. Pada Gambar 1 ditunjukkan konsep model *waterfall*.



Gambar 1 Model Waterfall

Tahap-tahap dalam tahapan penelitian pada Gambar 1, dijelaskan sebagai berikut: *Requirements*: Yaitu dengan menentukan kebutuhan sistem secara keseluruhan, antara lain dengan menentukan komponen-komponen sistem (*Entity*), atribut komponen dan hubungan antara

komponen. Secara umum *Entity* dibedakan atas data, algoritma dan *interface*. *Software Design*: Desain sistem merupakan tahap penyusunan proses, data, aliran proses, dan hubungan antar data yang paling optimal untuk menjalankan proses bisnis dan memenuhi kebutuhan sesuai dengan hasil analisis kebutuhan. Di tahap ini, ditentukan dan dibuat desain sistem seperti UML, database, dan tampilan web. *Implementation*: mengimplementasikan rancangan atau desain dengan menuliskan kode program sesuai bahasa pemrograman *Codeigneter*. *Verification*: melakukan pengujian program aplikasi yang telah selesai dibuat dengan memperhatikan konsep logika untuk mengetahui kinerja aplikasi apakah sesuai dengan kebutuhan sistem dan melakukan pencegahan terjadinya kesalahan seminimal mungkin, pengujian dilakukan dengan pengujian *black box*. *Maintenance*: memungkinkan terjadinya perubahan data, lingkungan sistem dan kebutuhan penggunaan agar aplikasi tetap bisa dikembangkan sesuai perubahan yang terjadi.

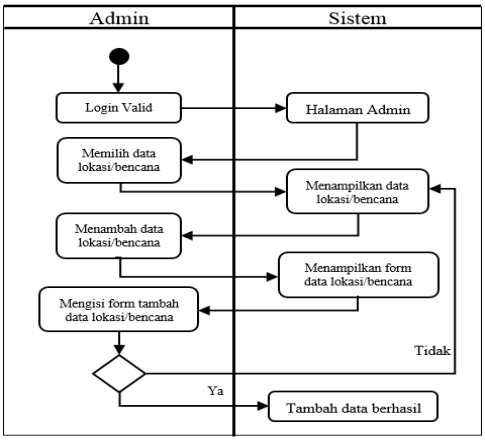
Diagram alur data dan informasi perancangan Pemetaan Potensi Bencana di Jawa Tengah Menggunakan Google Maps API dan KML dengan Metode *K-Means*, terdiri dari *Use case* diagram, *Activity* diagram, dan *Class* diagram.



Gambar 2 *Use Case Diagram*

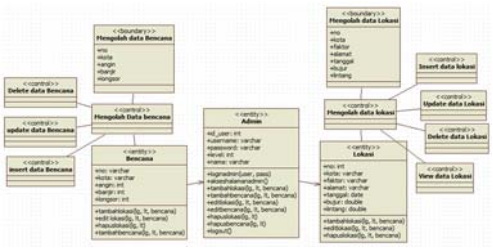
Use case merepresentasikan interaksi antara *actor* dengan sistem dan menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. *Use case* pada Gambar 2 terdiri dari dua aktor, yaitu *Admin* dan *User*. *Admin* adalah pengelola dari sistem informasi geografis, sedangkan *User* adalah masyarakat umum selaku pengguna sistem. *Admin* sebelum melakukan manipulasi data SIG terlebih dahulu melakukan *login*. Manipulasi data SIG disini terdiri dari *input* dan *update* data, *input* dan *update* letak peta, *input* dan *update* data perkembangan. Sementara *user* dalam hal ini adalah masyarakat pengguna sistem berinteraksi dengan sistem melalui informasi potensi bencana di Jawa Tengah.

Activity Diagram menggambarkan berbagai alur aktivitas secara umum dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity* Diagram tidak menggambarkan *behaviour* internal sebuah sistem. Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.



Gambar 3 *Activity Diagram* Mengolah Data

Gambar 3 menggambarkan aktivitas yang dilakukan *admin* dalam *login* ke sistem, dimana *admin* dapat memasukkan *username* dan *password* yang sudah terdaftar ke sistem tersebut. *Admin* dalam pengolahan sistem, dimana *admin* dapat melakukan penambahan, perubahan serta penghapusan data tentang sistem informasi tersebut.



Gambar 4 *Class Diagram*

Gambar 4 merupakan *class diagram* yang telah disusun berdasarkan daftar kandidat kelas yang terdiri dari tiga kelas yaitu bencana, *admin* dan lokasi.

K-Means clustering adalah metode untuk mengelompokkan item ke dalam k kelompok (dimana k adalah jumlah kelompok yang diinginkan). Kelompok/*cluster* dibentuk dengan meminimalkan jumlah dari *Euclidean distances*) di antara data dengan titik pusat (*centroid*) yang berkorespondensi. *Centroid* adalah titik pusat data, dalam hal ini diasumsikan rata-rata *vector* sebagai *centroid*.

Inisialisasi *centroid* dapat dilakukan dengan beberapa cara, contohnya tiga cara berikut:

- Dipilih secara dinamik: Metode ini tepat digunakan jika data baru ditambahkan secara cepat dan banyak. Untuk menyederhanakan persoalan, inisial *cluster* dipilih dari beberapa data baru, misal jika data dikelompokkan menjadi tiga *clusters*, maka inisial *cluster* berarti tiga item pertama dari data.
- Dipilih secara *random*: Paling banyak digunakan, dimana inisial *cluster* dipilih secara *random* dengan *range* data antara nilai terendah sampai nilai tertinggi.
- Memilih dari batasan nilai tinggi dan rendah: tergantung pada tipe datanya, nilai data tertinggi dan terendah dipilih sebagai inisial *cluster*.

Pengelompokan data dengan metode *K-Means* dilakukan dengan algoritma:

a. Tentukan jumlah kelompok.
b. Alokasikan data ke dalam kelompok secara acak.
c. Hitung pusat kelompok (*centroid*/rata-rata) dari data yang ada di masing-masing kelompok. Lokasi *centroid* setiap kelompok diambil dari rata-rata (*mean*) semua nilai data pada setiap fiturnya. Jika M menyatakan jumlah data dalam sebuah kelompok, i menyatakan fitur ke-i dalam sebuah

kelompok, dan p menyatakan dimensi data, maka persamaan untuk menghitung *centroid* fitur ke-i digunakan Persamaan 1.

$$\text{Ci} = \frac{1}{M}\sum_{j=1}^{M} xj \quad (1)$$

Keterangan:
Ci = pusat kelompok *centroid*M = Jumlah data sebuah kelompok
j = Kelompok data pada ke-j
Xj = nilai kelompok data ke-j
persamaan 1 dilakukan sebanyak p dimensi dari i=1 sampai dengan i=p.

$$\text{Ci} = \frac{1}{M}\sum_{j=1}^{M} xj$$
$$= \frac{15+13+35+5+13+14+9+1+6+5+14+3}{12}$$
$$= \frac{133}{12} = 11.083$$

d. Alokasikan masing-masing data ke *centroid*/jarak minumum. Ada beberapa cara yang dapat dilakukan untuk mengukur jarak data ke pusat kelompok, di antaranya adalah *Euclidean* [9]. Pengukuran jarak pada ruang jarak (*distance space*) *Euclidean* dapat dicari menggunakan Persamaan 2.

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (2)$$

Keterangan :
d = jarak pada ruang / *Euclidean*
$x1$ = nilai dari obyek x ke 1 $x2$ = nilai dari obyek x ke 2
$y1$ = nilai dari obyek y ke 1 $y2$ = nilai dari obyek y ke 2

$d1$
$= \sqrt{(5-81)^2 + (1-65)^2 + (70-65)^2}$
$= \sqrt{5776 + 4096 + 25}$
$= \sqrt{9897}$
$= 99.484$

$d2$
$= \sqrt{(5-65)^2 + (1-81) + (70-65)^2}$
$= \sqrt{3600 + 6400 + 25}$
$= \sqrt{10025}$
$= 100.129$

$d3$
$= \sqrt{(5-65)^2 + (1-65)^2 + (70-81)^2}$
$= \sqrt{3600 + 4096 + 121}$
$= \sqrt{7817}$
$= 88.413$

Maka, jarak minimum untuk Kota Banjarnegara pada iterasi pertama terdapat pada cluster 3, namun proses masih belum berhenti hingga tidak ada lagi perpindahan objek cluster.

e. Kembali ke langkah 3, apabila masih ada data yang berpindah kelompok atau apabila ada perubahan nilai *centroid* di atas nilai ambang yang ditentukan, atau apabila perubahan nilai pada fungsi objektif yang digunakan masih di atas nilai ambang yang ditentukan.

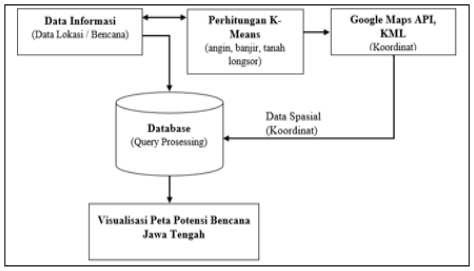
III. HASIL DAN PEMBAHASAN

Sistem informasi geografis yang dibangun berbasis web, dimana terdapat dua *user* yang dapat mengakses website yaitu: *Admin* dan *User* Umum. *Admin* dan *user* umum akan mengunjungi halaman utama *website* ketika pertama kali mengakses url *website* SIG, dimana *website* menampilkan halaman *login*. *User* umum dapat langsung mengakses *website* tanpa harus *login* terlebih dahulu dengan memilih menu data daerah. Terdapat juga menu lain seperti data daerah, peta lokasi dan tentang *website*.



Gambar 5 Tampilan Data Daerah Sistem Informasi Pemetaan Potensi Bencana di Jawa Tengah

Pada Gambar 5 halaman utama terdapat menu-menu yang ada pada web sistem informasi geografis pemetaan potensi bencana di Jawa Tengah. Menu data daerah berisi data lokasi dan data bencana, di halaman data daerah *user* bisa melakukan *view* peta bencana di Jawa Tengah dan melihat detail peta. Halaman ini dapat diakses oleh *user* tanpa harus *login*, sedangkan *admin* harus *login* terlebih dahulu untuk mengolah sistem. *Admin* dapat melakukan tambah data lokasi, hapus data lokasi, menghapus data lokasi, tambah data bencana, hapus data bencana, dan menghapus data bencana.



Gambar 6 Kerangka Sistem *Web GIS Application*

Pada Gambar 6 sistem dibangun memiliki administrator web sistem, bagian ini akan memasukkan data informasi bencana yang telah dihitung dengan metode *K-Means* ke dalam database *server*, sementara itu *google maps* API *Key* di-*embedded* ke dalam *administrator web* sistem dan data spasial dari *google maps* dimasukkan ke dalam database hasil *query processing* akan menampilkan peta beserta data informasi bencana di Jawa Tengah.



Gambar 7 Peta Jawa Tengah

Pada Gambar 7 halaman ini menampilkan peta Provinsi Jawa Tengah dengan batasan pada setiap kota atau kabupaten. Setiap kota atau kabupaten disimbolkan dengan warna hijau untuk bencana banjir, warna merah untuk bencana angin, dan warna kuning untuk bencana tanah longsor.

Kode Program 1 Menampilkan Peta

```
1.   var layers = [];
2.
3.  layers[0]= new
    google.maps.KmlLayer('
    https://drive.google.com/open?id=1p
    Nl1JgaD0Fh583iDdtsoOEG1wGJ, {
4.  preserveViewport: true});
5.  layers[1]= new
    google.maps.KmlLayer('
    https://drive.google.com/open?id=1p
    Nl1JgaD0Fh583iDdtsoOEG1wGJ, {
6.    preserveViewport: true});
7.
8.  function initialize()
9.   {
10.  var myLatlng = new
    google.maps.LatLng (-6.967354,
    109.020553);
11.
12.     var mapOptions = {
13.
14.        zoom: 12,
15.         center: myLatlng,
16. mapTypeId:
    google.maps.MapTypeId.SATELLITE
    };

17. Var map = new
    google.maps.Map(document.getElement
    ById('map'), mapOptions);
```

Kode Program 1 merupakan fungsi *initialize* () yang bertujuan untuk menyiapkan peta. Di dalam fungsi *initialize* () terdapat definisi properti yang dibutuhkan peta. Ada tiga properti yang dibutuhkan yaitu: *center*, *zoom*, dan *mapTypeId*. Fungsi *initialize* () dieksekusi dalam *event window load* saat halaman web dibuka. KML dipanggil dengan *link google drive* kemudian peta dipusatkan di koordinat Provinsi Jawa Tengah. KML yang menunjukkan posisi Provinsi Jawa Tengah dalam bentuk obyek. Ketika *file* ini dibuka dengan Google Maps maka akan tampil sebuah titik tepat pada koordinat yang sudah ditentukan.

Clustering data dikelompokkan menjadi tiga kelompok berdasarkan data yang tersedia. Masing masing kelompok ditentukan nilai *centroid* yang didapatkan dari nilai minimum, nilai *maximum* dan nilai tengah dari data yang tersedia. Proses *cluster* dilakukan dengan cara memperhitungkan jarak terdekat dengan menggunakan rumus *Euclidean.*

Proses *K-Means clustering* akan terus melakukan iterasi hingga data hasil *clustering* akan sama dengan hasil iterasi sebelumnya. Proses ini dilakukan berulang- ulang dengan menggunakan metode penentuan *centroid* hingga hasil *clustering* pada iterasi tersebut akan menghasilkan *clustering* yang persis sama dengan hasil *clustering* pada iterasi sebelumnya. Proses selanjutnya adalah *clustering* pada iterasi kedua, yang hasil *cluster* belum sama dengan iterasi satu. Selanjutnya dilakukan iterasi ketiga yang hasilnya masih ada perbedaan pada iterasi kedua dan begitu seterusnya hingga didapatkan hasil yang sesuai. *Centroid* satu, dua dan tiga menunjukkan titik pusat pada variabel bencana (angin, banjir dan tanah longsor), sedangkan C1 menunjukkan potensi bencana angin yang dilambangkan dengan warna merah pada peta, C2 menunjukkan potensi bencana banjir dilambangkan dengan warna hijau pada peta dan C3 menunjukkan bencana tanah longsor dilambangkan dengan warna kuning pada peta.

Kode Program 2 *Function Lanjut*

```
1.  public function lanjut()  {
2.  $data['bencana'] = $this->db-
    >get('bencana');
3.  $id = $this->db->query('select
    max(nomor) as m from
    hasil_centroid');
4.  foreach($id->result() as $i)
        { $id = $i->m; }
5.  $this->db->where('nomor', $id);
6.  $data['centroid'] = $this->db-
    >get('hasil_centroid');
7.  $it = $this->db->query('select
    max(iterasi) as it from
    centroid_temp');
8.  foreach($it->result() as $i){
9.  $it = $i->it;      }     $it_temp
    = $it-1;
10. $this->db->where('iterasi',
    $it_temp);
11. $no=0;
12. foreach($it_sebelum->result() as
    $it_prev){
13. $c1_sebelum[$no] = $it_prev->c1;
14. $c2_sebelum[$no] = $it_prev->c2;
15. $c3_sebelum[$no] = $it_prev->c3;
```

Kode Program 2 merupakan proses *K-Means clustering* yang terus melakukan iterasi hingga data hasil *clustering* akan sama dengan hasil iterasi sebelumnya. Kode Program 2 melakukan proses secara berulang-ulang dengan menggunakan metode penentuan *centroid* hingga hasil *clustering* pada iterasi tersebut akan menghasilkan *clustering* yang sama dengan hasil *clustering* pada iterasi sebelumnya.

Kode Program 3 *View Hasil*

```
1.  $this->db->query('truncate table
    centroid_temp');
2.  $this->db->query('truncate table
    hasil_centroid');
3.  $hc1 = sqrt(pow(($s['angin']-
    $c1a),2) +pow(($s['banjir']-
    $c1b),2) +pow(($s['longsor']-
    $c1c),2));
4.              echo $hc1;     ?></td>
5.      <td colspan="3"><?php
6.  $hc2 = sqrt(pow(($s['angin']-
    $c2a),2) +pow(($s['banjir']-
    $c2b),2) +pow(($s['longsor']-
    $c2c),2));
7.  echo $hc2;?></td>
8.
9.  <td colspan="3"><?php
10. $hc3 = sqrt(pow(($s['angin']-
    $c3a),2) +pow(($s['banjir']-
    $c3b),2) +pow(($s['longsor']-
    $c3c),2)); echo $hc3;
```

Kode Program 3 merupakan hasil dari perkumpulan beberapa iterasi yang diproses secara berulang-ulang dengan menggunakan metode penentuan centroid hingga hasil *clustering* pada iterasi tersebut akan menghasilkan *clustering* yang persis begitu seterusnya hingga didapatkan hasil yang sesuai. Pow merupakan operator aritmatika perkalian suatu nilai dengan dirinya sendiri sebayak jumlah pemangkatnya. Sedangkan Sqrt merupakan kebalikan dari pow yaitu suatu operator aritmatika yang memiliki kemampuan untuk mencari akar dari suatu nilai/hasil bilangan berpangkat.

Tabel 2 Hasil pengujian *black-box*

Fungsi yang di Uji	Kondisi	*Output* yang diharapkan
Login	*Username* dan *password* benar	Sukses *Login*
	Username dan *password* salah / tidak diisi	Gagal *Login*
Tambah data lokasi / bencana	*Form* diisi dengan benar	Sukses Tambah data lokasi / bencana
***Edit* data lokasi / bencana**	*Form* diisi dengan benar	Sukses *edit* data lokasi / bencana
Hapus data lokasi / bencana	Pilih salah satu data klik *button* hapus	Sukses hapus data lokasi / bencana
***Edit* data artikel / berita**	*Form* diisi dengan benar	Sukses *edit* artikel / berita
Menampilkan peta	Klik *button view*	Menampilkan Peta
Menampilkan *K-Means*	Klik menu analisa data	Menampilkan perhitungan *K-Means*

Berdasarkan pengujian yang dilakukan pada aplikasi sistem informasi geografis ini dapat dilihat dari setiap pengujian yaitu *valid,* maka dapat disimpulkan bahwa aplikasi ini berjalan dengan baik dan sesuai dengan yang diharapkan.

IV. KESIMPULAN

Berdasarkan penelitian yang dilakukan, maka dapat disimpulkan bahwa perancangan Pemetaan Potensi Bencana di Jawa Tengah Menggunakan Google Maps API dan KML dengan Metode *K-Means* dapat memberikan cara baru pada masyarakat untuk memenuhi kesiapan yang cukup sebelum bencana terjadi, sehingga efek bencana dapat diminimalisir. Lebih dari itu, dapat menjadi solusi untuk memutuskan langkah yang tepat untuk menangani dan menganalisis risiko baik jangka pendek maupun jangka panjang termasuk rencana pemulihan wilayah bencana. Dengan adanya sistem informasi geografis yang

berbasis web menggunakan bahasa pemrograman PHP berbasis *Framework Codeigniter* dan tampilan peta Google Maps API, akan mempermudah pengguna dalam mencari informasi potensi bencana di Jawa Tengah. Saran sistem informasi geografis yang dibuat masih dalam tahap awal, sehingga data yang ditampilkan masih relatif sedikit, oleh sebab itu perlu dikembangkan selanjutnya agar menjadi lebih baik, yaitu penambahan fitur sehingga informasi yang disampaikan kepada pengguna lebih lengkap. Perlu data yang rinci mengenai jumlah bencana. Perlu menggunakan algoritma lain untuk melakukan pengelompokan status bencana sehinggga pengelompokan yang dilakukan memiliki nilai akurasi yang lebih baik.

DAFTAR PUSTAKA

[1] Hpli. 2017, Potensi Bencana. *http://www.hpli.org/bencana.php/*. Diakses tanggal 22 Januari 2019.

[2] Aini, A. (n.d.). Sistem Informasi Geografis Pengertian Dan Aplikasinya. Diakses tanggal 5 Februari 2019.

[3] Khomarudin, A. N. (2016). Teknik Data Mining : Algoritma *K-Means* Clustering, 1–12. Diakses tanggal 22 Januari 2019.

[4] Putra, R. P. C., Santoso, N., Studi, P., Informatika, T., Informasi, J. T., & Malang, P. N. (2011). Pemetaan Daerah Rawan Banjir Berbasis Geographic Information System (GIS) Menggunakan Metode Bayes Studi Kasus : BPBD Kabupaten Malang. Diakses tanggal 22 Januari 2019.

[5] Purnomo, A. S. (2018). Pemanfaatan Google Maps API Dalam Sistem Informasi Kerawanan Bencana Berbasis Web, (November). Diakses tanggal 10 April 2019.

[6] Hasmin, E., & Aini, N. (2017). Integrasi Google Maps Sebagai Platform Pengelolaan Air Limbah, 55–60. Diakses tanggal 23 Januari 2019.

[7] Ramadhan, M. I. (n.d.). Penerapan data mining untuk analisis data bencana milik bnpb menggunakan algoritma, 22(1). Diakses tanggal 24 Januari 2019.

[8] Sosial, F. I. (n.d.). Pemetaan daerah rawan bencana gerakan tanah di wilayah Grabag Kabupaten Magelang Propinsi Jawa Tengah, 1–8. Diakses tanggal 24 Januari 2019.

[9] Saraswati, F. (2016). Sistem Informasi Geografis Pemetaan Desa Pengrajin Batik d i Kabupaten Bantul Berbasis Web, 1–17. Diakses tanggal 24 Februari 2019.

[10] Eddy, P. 2009:119*, Sistem Informasi Geografis: Konsep-konsep Dasar (Perspektif Geodesi & Geomatika).* Bandung: Penerbit Informatika Bandung.