

BAB 4 PENGOLAHAN DATA, ANALISIS KEBUTUHAN, DAN PERANCANGAN SISTEM

4.1 Pengolahan Data

4.1.1 Data Kecelakaan Kota Batu Tahun 2013

Data kecelakaan tahun 2013 ini didapat dari Unit Laka Polres Kota Batu. Sedangkan data koordinat didapat dari *google maps*. Pada tabel 4.1 dapat diketahui terdapat 6 titik kecelakaan pada tahun 2013. Pada tahun 2013 terjadi 179 kecelakaan. Titik kecelakaan yang paling tinggi terjadi di Jalan Suropati No. 1 dengan total kejadian 69. Dimana T adalah total kecelakaan, M adalah total meninggal, LB adalah total luka berat, dan LR adalah total luka ringan.

Table 4.1 Data Kecelakaan Tahun 2013

No	Longitude	Latitude	Nama Jalan	Alamat	T	M	LB	LR
1	112.57833	-7.906944	Jl. Raya Dadaprejo No. 1	Junrejo	25	11	0	36
2	112.520972	-7.874738	Jl. Suropati No. 43-45	Batu	69	15	0	63
3	112.48684	-7.852967	Jl. Rajekwesi	Pujon	32	9	0	28
4	112.390759	-7.877026	Jl. Brigjend Abd Mana Wijaya	Ngantang	25	10	0	43
5	112.35241	-7.801518	Jl. Raya Waturejo No. 16	Kasembon	14	6	0	40
6	112.298409	-7.774564	Jl. Raya Kasembon No. 16	Kasembon	11	3	0	25

4.1.2 Data Kecelakaan Kota Batu Tahun 2014

Data kecelakaan tahun 2014 ini didapat dari Unit Laka Polres Kota Batu. Sedangkan data koordinat didapat dari *google maps*. Pada tabel 4.2 dapat diketahui terdapat 6 titik kecelakaan pada tahun 2014. Pada tahun 2014 terjadi 109 kecelakaan, angka ini turun dari tahun 2013. Titik kecelakaan yang paling tinggi terjadi di Jalan Brigjend Abd Manan Wijaya dengan total kejadian 30. Dimana T adalah total kecelakaan, M adalah total meninggal, LB adalah total luka berat, dan LR adalah total luka ringan.

Table 4.2 Data Kecelakaan Tahun 2014

No	Longitude	Latitude	Nama Jalan	Alamat	T	M	LB	LR
1	112.575875	-7.902226	Jl. Drs. Moh. Hatta No. 4	Junrejo	21	6	2	35

2	112.543051	-7.906495	Jl. Brigjend Abd Manan Wijaya	Junrejo	30	14	3	53
3	112.528689	-7.835729	Jl. Salam	Bumiaji	16	5	0	33
4	112.451425	-7.841097	Jl. Brigjend Abd Manan Wijaya No. 18	Pujon	18	6	0	37
5	112.370684	-7.845474	Jl. Hasanuddin No. 267	Ngantang	14	3	0	30
6	112.308344	-7.780632	Jl. Raya Kasembon No. 16	Kasembon	10	3	0	11

4.1.3 Data Kecelakaan Kota Batu Tahun 2015

Data kecelakaan tahun 2015 ini didapat dari Unit Laka Polres Kota Batu. Sedangkan data koordinat didapat dari *google maps*. Pada tabel 4.3 dapat diketahui terdapat 7 titik kecelakaan pada tahun 2014, lebih banyak dibandingkan tahun-tahun sebelumnya yaitu tahun 2013 dan 2014. Pada tahun 2015 terjadi 170 kecelakaan. Titik kecelakaan yang paling tinggi terjadi di Jalan Raya Oro-oro Ombo No. 200 dengan total kejadian 66. Dimana T adalah total kecelakaan, M adalah total meninggal, LB adalah total luka berat, dan LR adalah total luka ringan.

Table 4.3 Data Kecelakaan Tahun 2015

No	Longitude	Latitude	Nama Jalan	Alamat	T	M	LB	LR
1	112.569432	-7.905099	Jl. Raya Dadaprejo No. 136	Junrejo	24	3	7	17
2	112.535172	-7.899766	Jl. Oro-oro Ombo No. 200	Junrejo	66	17	23	54
3	112.489032	-7.865777	Jl. Trunojoyo No. 93	Batu	21	4	8	24
4	112.43095	-7.857578	Jl. Brigjend Abd Manan Wijaya	Pujon	11	0	6	23
5	112.380927	-7.852966	Jl. Hasanuddin No. 267	Ngantang	16	2	8	22
6	112.365568	-7.82051	Jl. Raya Waturejo No. 16	Ngantang	10	3	6	13
7	112.291463	-7.769773	Jl. Raya Kasembon No. 16	Kasembon	22	4	11	23

4.1.4 Data Kecelakaan Kota Batu Tahun 2016

Data kecelakaan tahun 2016 ini didapat dari Unit Laka Polres Kota Batu. Sedangkan data koordinat didapat dari *google maps*. Pada tabel 4.4 dapat diketahui terdapat 7 titik kecelakaan pada tahun 2016. Titik kecelakaan yang paling tinggi terjadi di Jalan Panglima Sudirman No.507 dengan total kejadian 45. Dimana T adalah total kecelakaan, M adalah total meninggal, LB adalah total luka berat, dan LR adalah total luka ringan.

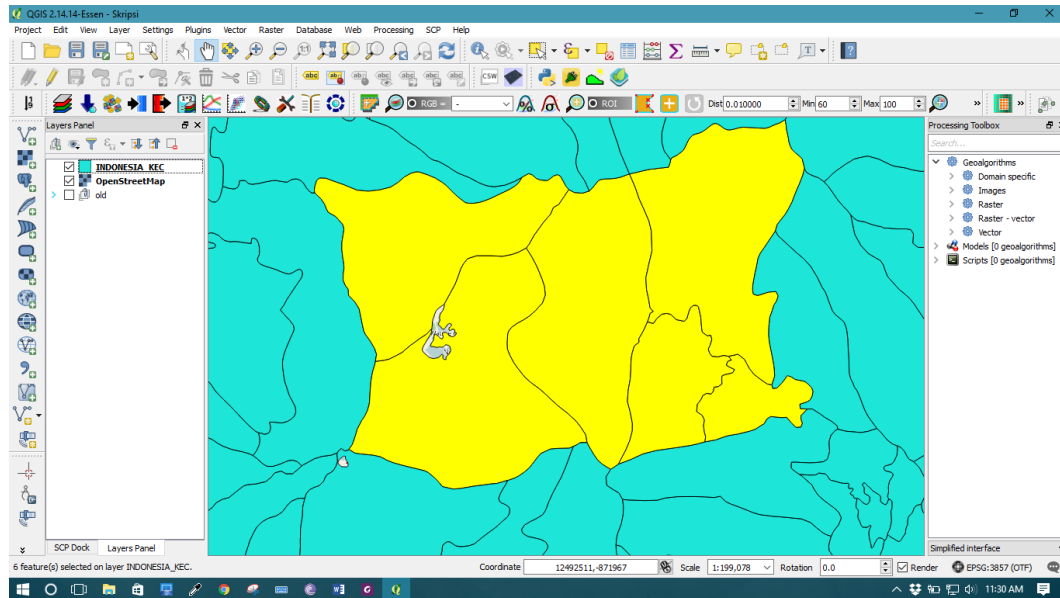
Table 4.4 Data Kecelakaan Tahun 2016

No	Longitude	Latitude	Nama Jalan	Alamat	T	M	LB	LR
1	112.555021	-7.909111	Jl. Hasanuddin No.145	Junrejo	38	8	4	40
2	112.533284	-7.863353	Jl. Raya Pandanrejo No.2	Bumiaji	11	1	0	25
3	112.511631	-7.866714	Jl. Panglima Sudirman No.507	Batu	45	11	8	31
4	112.441181	-7.852417	Jl. Brigjend Abd Manan Wijaya No.703	Pujon	22	8	1	29
5	112.380886	-7.860677	Jl. Brigjend Abd Manan Wijaya	Pujon	13	1	2	26
6	112.345654	-7.790677	Jl. Raya Waturejo No.16	Ngantang	15	4	2	22
7	112.311707	-7.78416	Jl. Raya Waturejo No.16	Ngantang	14	2	3	27

4.2 Tahapan Pada QGIS

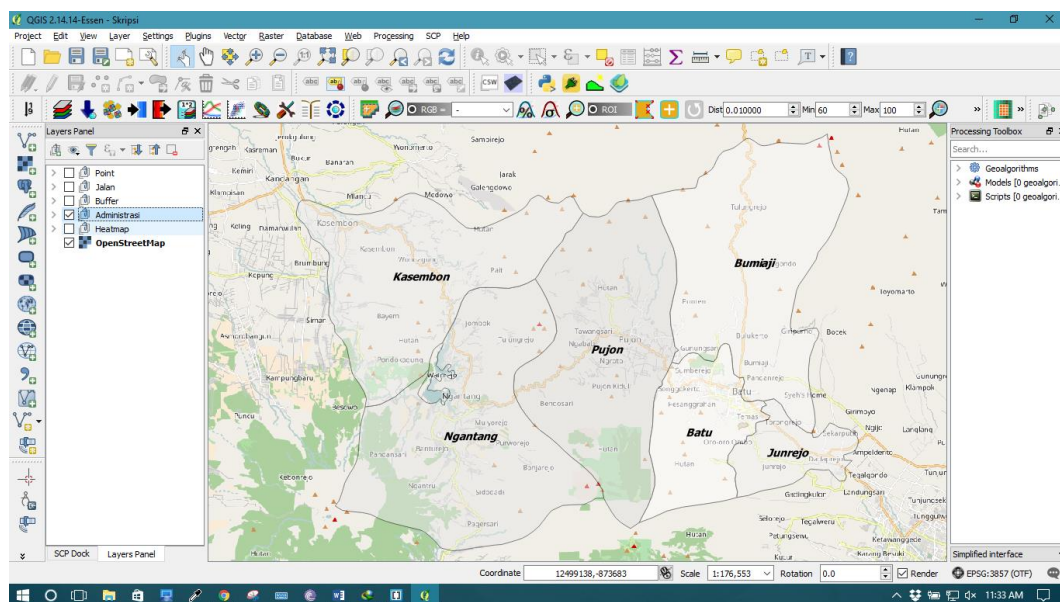
Pada bagian ini akan dijelaskan mengenai cara mengolah data yang sudah dikumpulkan. Data akan diolah menggunakan *tools* yang tersedia pada aplikasi QGIS.

4.2.1 Pengolahan Data Menggunakan QGIS



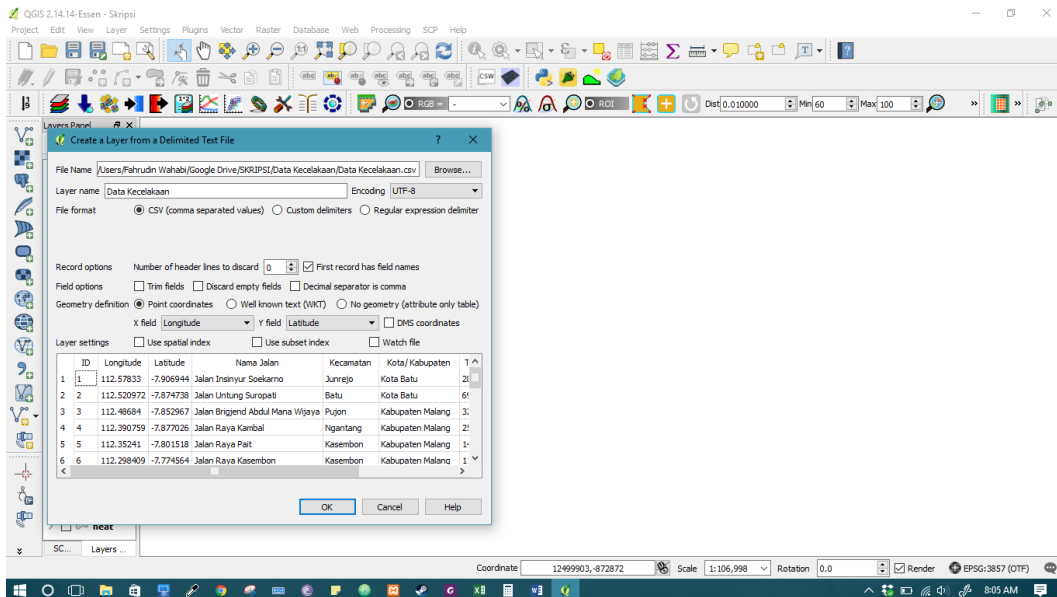
Gambar 4.1 Seleksi Wilayah Administrasi

Pada gambar 4.1 ditunjukkan wilayah administrasi yang telah diseleksi dari *shapefile* batas wilayah kecamatan seluruh Indonesia. Wilayah yang diseleksi merupakan wilayah operasional Polres Kota Batu.



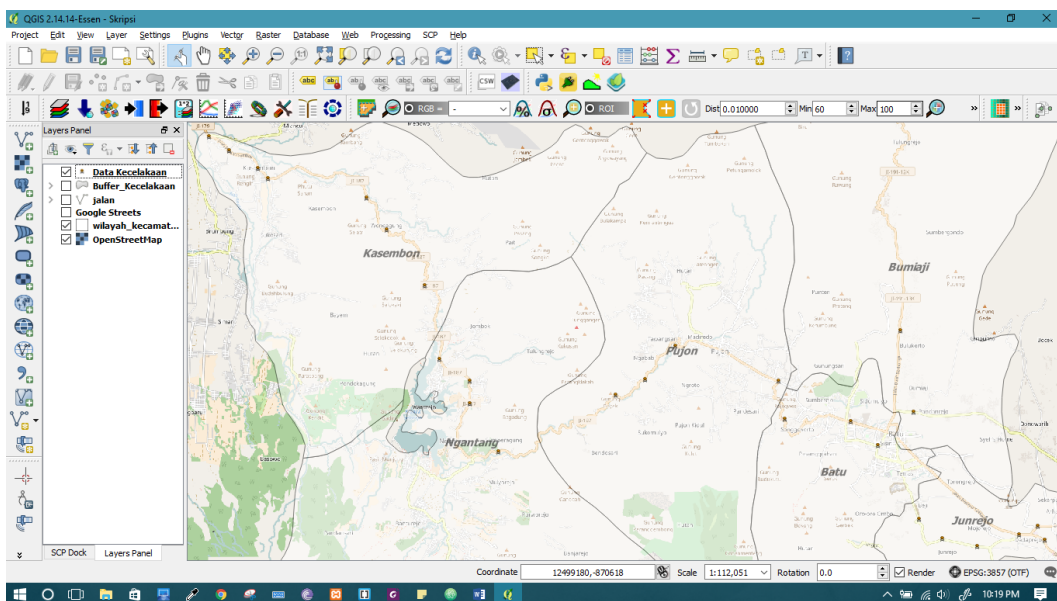
Gambar 4.2 Hasil Seleksi Wilayah

Pada gambar 4.2 ditunjukkan hasil dari seleksi wilayah administrasi yang dilakukan pada gambar 4.1. Pada Kota Batu sendiri terdapat 3 kecamatan yaitu, Kecamatan Junrejo, Kecamatan Bumiaji, dan Kecamatan Batu. Sedangkan pada Kabupaten Malang terdapat 3 kecamatan, yaitu Kecamatan Pujon, Kecamatan Ngantang, dan Kecamatan Kasembon.



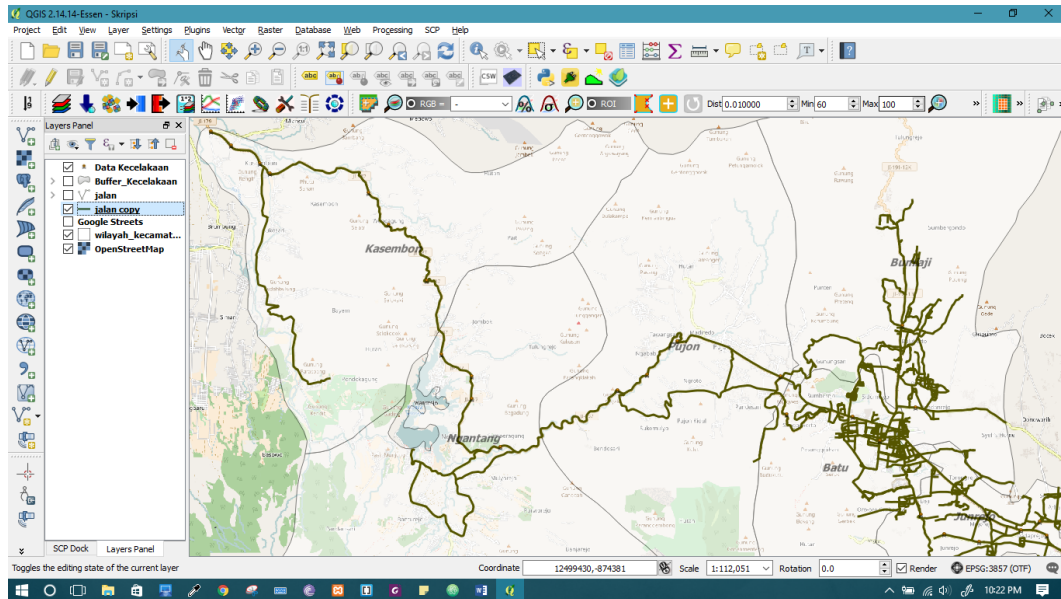
Gambar 4.3 Memasukkan Data Kecelakaan

Pada gambar 4.3 ditunjukkan proses untuk memasukkan data lokasi kecelakaan beserta informasi kecelakaan. Sebelum dimasukkan ke *QGIS*, data kecelakaan disimpan pada file dengan format *csv (comma separated values)*. Data yang dimasukkan berisi koordinat lokasi dan informasi kecelakaan pada lokasi tersebut.



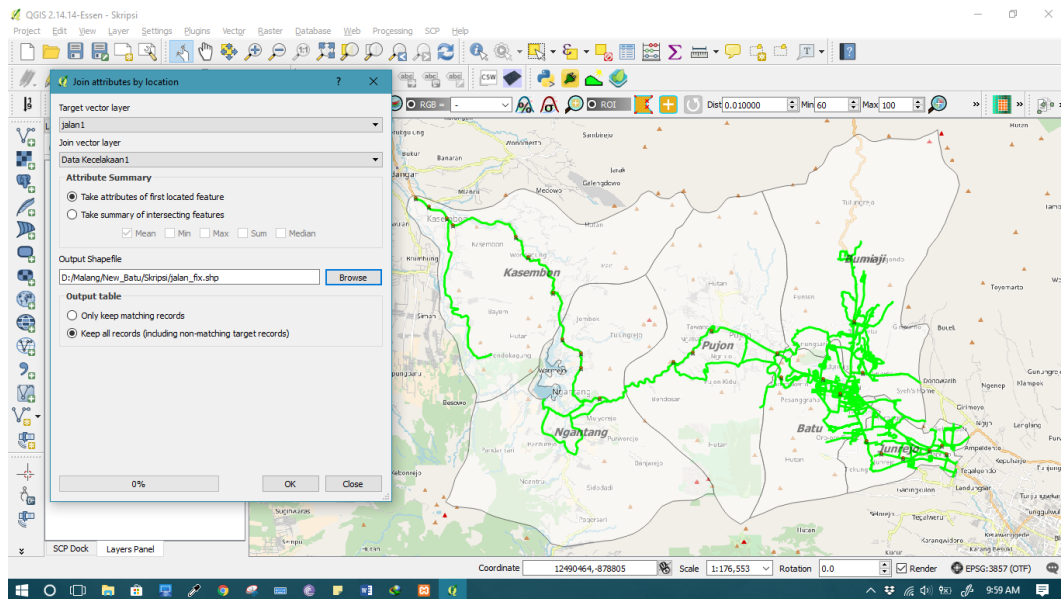
Gambar 4.4 Lokasi Kecelakaan di Kota Batu dan Sekitarnya

Pada gambar 4.4 ditunjukkan lokasi titik-titik kecelakaan yang pernah terjadi di Kota Batu dan sekitarnya. Pada saat proses memasukkan data kecelakaan ke *QGIS*, dipilih *WGS84 (World Geodetic System of 1984)* sebagai referensi koordinat.



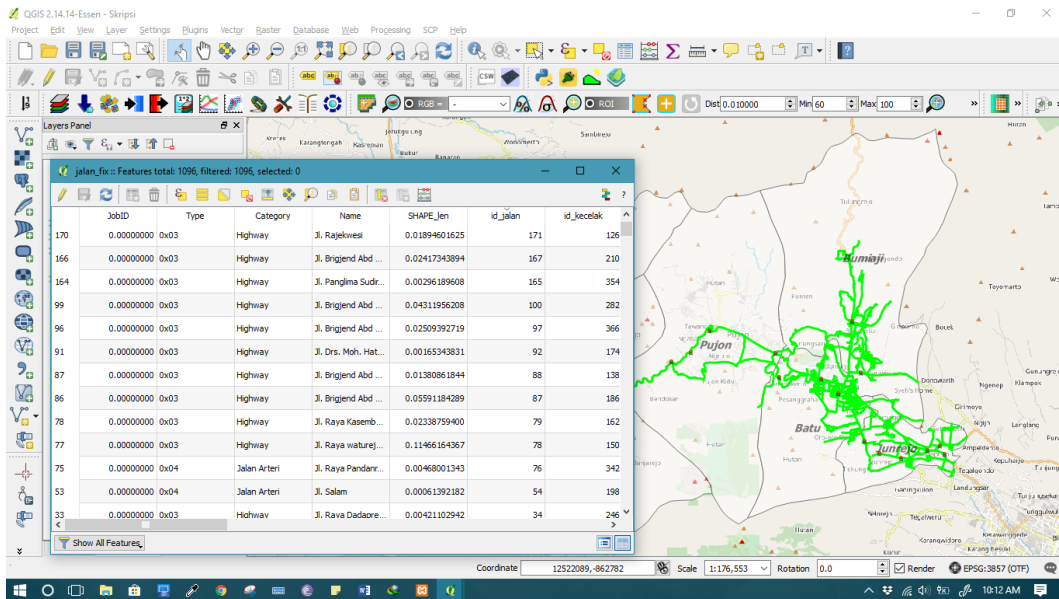
Gambar 4.5 Jaringan Jalan Berhasil Ditambahkan

Pada gambar 4.5 ditunjukkan jaringan jalan yang telah berhasil dimasukkan ke dalam *QGIS*. Jaringan jalan ini digunakan sebagai objek untuk melakukan analisis selanjutnya. Jaringan jalan ini merupakan jenis data spasial vektor.



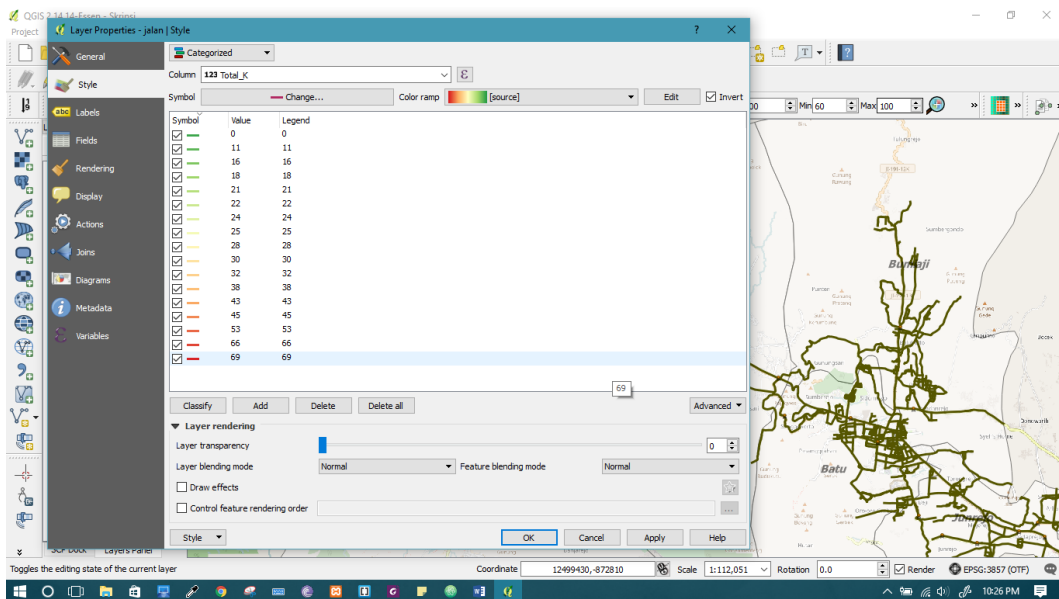
Gambar 4.6 Join Data Atribut

Gambar 4.6 merupakan proses *join* atribut antara data kecelakaan dengan data jaringan jalan. Proses ini digunakan untuk menggabungkan atribut dari kedua data tersebut. Hasil proses ini akan digunakan dalam proses selanjutnya.



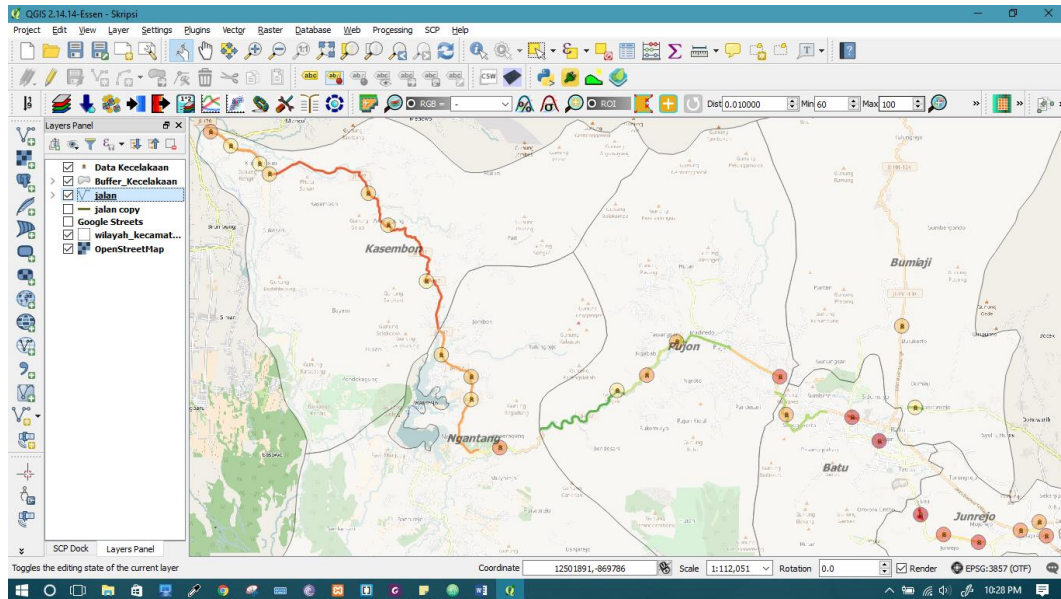
Gambar 4.7 Hasil Join Atribut

Pada gambar 4.7 ditunjukkan proses hasil proses *join* atribut dimana data atribut dari 2 data berhasil dikombinasikan. Di dalam jaringan jalan sebenarnya tidak terdapat atribut *id_kecelakaan*. Oleh karena itu perlu ditambahkan kolom baru untuk menyimpan data *id_kecelakaan*.



Gambar 4.8 Penggunaan Tools Klasifikasi

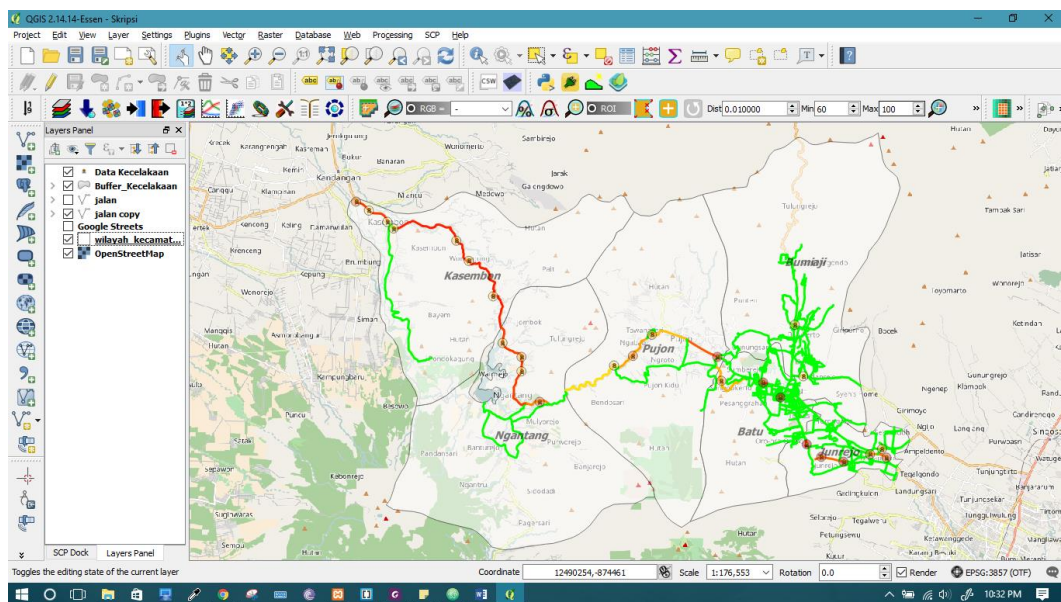
Pada gambar 4.8 ditunjukkan proses pewarnaan jaringan jalan. Pada analisis ini dilakukan klasifikasi dengan warna berdasarkan jumlah kecelakaan di setiap segmen jalan. Pengurutan data menggunakan *kernel density analysis*, yaitu diurutkan berdasarkan total kecelakaan paling banyak ke total kecelakaan paling sedikit. Hasilnya jalan dengan total kecelakaan paling tinggi berwarna merah dan paling rendah berwarna hijau.



Gambar 4.9 Hasil Pewarnaan Menggunakan *Tools* Klasifikasi

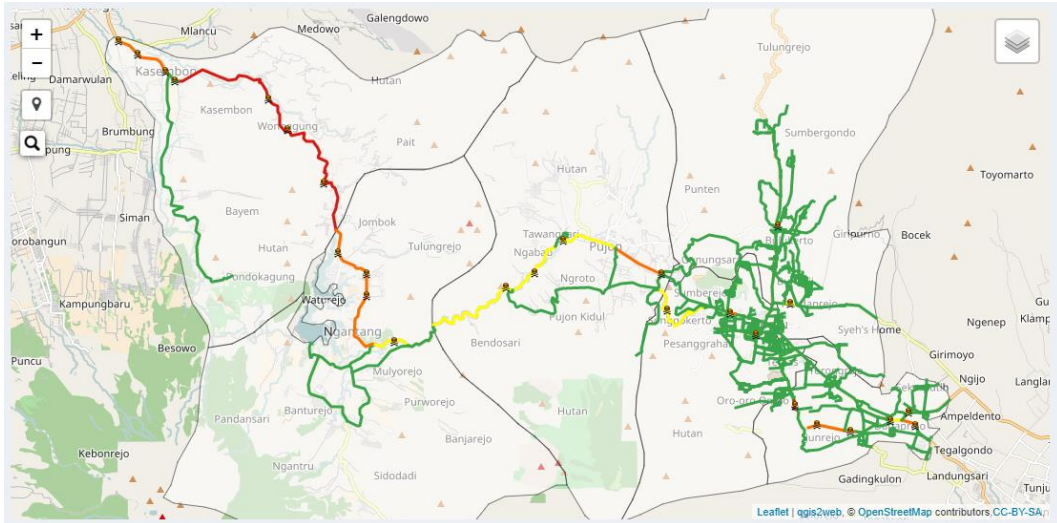
Pada gambar 4.9 ditunjukkan hasil dari pewarnaan jaringan jalan. Jaringan jalan yang berwarna merah merupakan jalan yang memiliki jumlah kecelakaan terbanyak.

4.2.2 Hasil Pemetaan Lokasi Kecelakaan



Gambar 4.10 Hasil Pemetaan Menggunakan *QGIS*

Pada gambar 4.10 ditunjukkan hasil pemetaan menggunakan *QGIS*. Pada peta tersebut sudah terdapat batas wilayah administrasi yang ditangani oleh Polres Kota Batu. Terdapat juga lokasi-lokasi kecelakaan yang pernah terjadi. Ditunjukkan pula hasil dari penggunaan *tools* klasifikasi berupa jalan yang berwarna sesuai jumlah kecelakaan pada setiap segmen jalan.

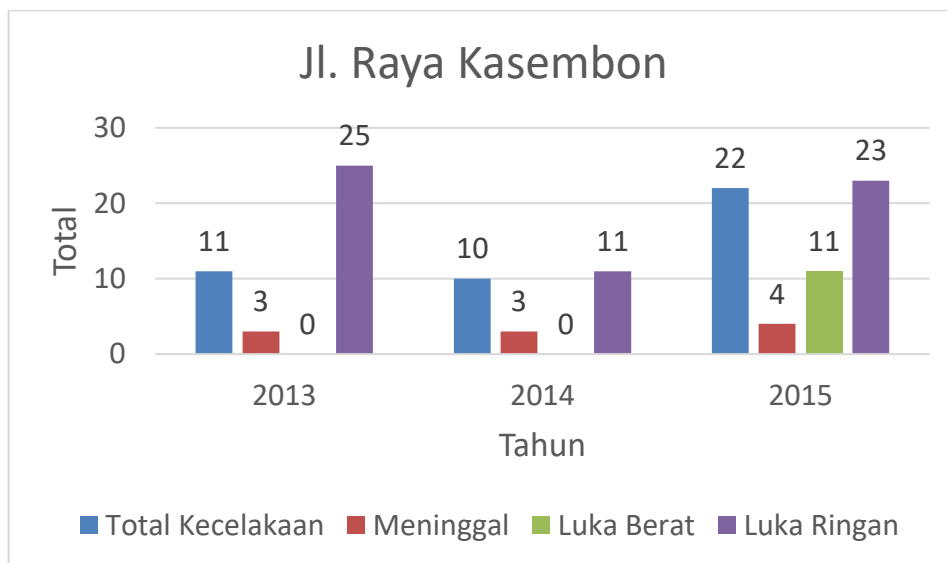


Gambar 4.11 Hasil Export WebGIS

Pada gambar 4.11 ditunjukkan hasil *export* dari QGIS menjadi WebGIS. Lokasi-lokasi kecelakaan sudah berhasil dipetakan. Selanjutnya hasil *export* ini akan dilengkapi dengan kode program pada tahap implementasi.

4.3 Analisis Street Profile

Pada tahap ini dilakukan analisis *Street Profile* untuk melihat pola perkembangan kecelakaan pada jalan tertentu. Tahap ini diawali dengan mengumpulkan data-data kecelakaan pada masing-masing jalan pada sebuah tabel. Data-data yang sudah dikumpulkan bisa dilihat pada Tabel 4.1 sampai Tabel 4.4. Setelah itu data tersebut dibuatkan sebuah grafik menggunakan bantuan aplikasi *Microsoft Excel*. Pada Gambar 4.12 merupakan contoh salah satu grafik analisis *Street Profile*.



Gambar 4.12 Contoh Grafik Street Profile

4.4 Perhitungan EAN

Pada tahap ini dilakukan perhitungan menggunakan EAN. Pada perhitungan ini digunakan Rumus 2.1. Jalan yang akan dilakukan perhitungan ini hanya diambil beberapa saja sebagai contoh. Pada Tabel 4.5 merupakan hasil perhitungan EAN.

Table 4.5 Hasil Perhitungan EAN

No	Nama Jalan	Weighted Accident Number				Total
		10 × FAT	4,25 × SVI	2,33 × MNI	1 × PDO	
1	Jl. Raya waturejo No.16	15	11	102	0	434.41
2	Jl. Oro-Oro Ombo No.200	17	23	54	0	393.57
3	Jl. Brigjend Abd Manan Wijaya No. 1	15	5	79	0	355.32
4	Jl. Suropati No.43-45	15	0	63	0	296.79
5	Jl. Raya Kasembon No.16	10	11	59	0	284.22
6	Jl. Panglima Sudirman No.507	11	8	31	0	216.23
7	Jl. Hasanuddin No.267	5	8	52	0	205.16
8	Jl. Brigjend Abd Manan Wijaya	10	0	43	0	200.19
9	Jl. Raya Dadaprejo No.1	11	0	36	0	193.88
10	Jl. Hasanuddin No.145	8	4	40	0	190.2
dst.						

Hasil dari perhitungan ini didapatkan bahwa Jl. Raya Waturejo memiliki total nilai tertinggi. Dengan demikian dapat disimpulkan bahwa tingkat kecelakaan tertinggi terdapat pada Jl. Raya Waturejo.

4.5 Analisis Kebutuhan Sistem

4.5.1 Identifikasi Aktor

Identifikasi aktor adalah seseorang atau sistem lain yang terlibat dengan sistem secara langsung, berikut ini aktor yang terkait:

1. Masyarakat Umum

Masyarakat umum merupakan semua masyarakat yang berkenan untuk melihat hasil dari pemetaan lokasi kecelakaan di Kota Batu. Pada aktor masyarakat umum ini hanya dapat melihat informasi yang ada di sistem dan tidak dapat menambah, mengubah, dan menghapus data.

2. Admin

Admin merupakan orang yang diberikan hak akses penuh untuk mengelola sistem. Aktor ini dapat melakukan banyak hal pada sistem, contohnya adalah menambah, mengubah, dan menghapus data.

4.5.2 Kebutuhan Fungsional

Kebutuhan fungsional sistem merupakan pernyataan layanan sistem yang harus disediakan. Pada kebutuhan fungsional sistem dapat dibagi berdasarkan aktor yang terlibat langsung dengan sistem. Berikut merupakan kebutuhan fungsional dari Sistem Informasi Pemetaan Lokasi Kecelakaan Kota Batu Berbasis *WebGIS* berdasarkan fungsionalitas yang dapat dilakukan oleh aktor. Pada Tabel 4.6 merupakan daftar kebutuhan fungsional dari masyarakat umum. Sedangkan pada Tabel 4.7 merupakan daftar kebutuhan fungsional dari admin.

1. Masyarakat Umum

Table 4.6 Tabel Kebutuhan Fungsional Pengguna Umum

No.	Kode Fungsi	Nama Fungsi	Keterangan
1	SIPK_01_01	Melihat peta kecelakaan	Dengan fungsi ini Pengguna Umum dapat melihat dimana saja lokasi kecelakaan yang pernah terjadi
2	SIPK_01_02	Melihat detail lokasi kecelakaan	Dengan fungsi ini Pengguna Umum dapat melihat grafik perkembangan kecelakaan pada titik kecelakaan yang dipilih
3	SIPK_01_03	Melihat detail jalan	Dengan fungsi ini Pengguna Umum dapat melihat grafik perkembangan kecelakaan pada ruas jalan yang dipilih

4	SIPK_01_04	Melihat statistik	Dengan fungsi ini Pengguna Umum dapat melihat statistik kecelakaan dalam banyak model
5	SIPK_01_05	Mencari Jalan	Dengan fungsi ini Pengguna Umum dapat mencari jalan berdasarkan nama jalan
6	SIPK_01_06	Melihat Perbandingan	Dengan fungsi ini Pengguna Umum dapat membandingkan informasi kecelakaan (total kecelakaan, total luka ringan, total luka berat, dan total meninggal) dari 2 wilayah kecamatan
7	SIPK_01_07	Mencetak Statistik	Dengan fungsi ini Pengguna Umum dapat mencetak statistik yang ada di halaman statistik

2. Admin

Table 4.7 Tabel Kebutuhan Fungsional Admin

No.	Kode Fungsi	Nama Fungsi	Keterangan
1	SIPK_01_08	<i>Login</i>	Dengan fungsi ini admin bisa mendapatkan hak akses untuk keseluruhan sistem
2	SIPK_01_09	<i>Logout</i>	Dengan fungsi ini admin dapat keluar dari sistem
3	SIPK_01_10	Melihat peta kecelakaan	Dengan fungsi ini admin dapat melihat dimana saja lokasi kecelakaan yang pernah terjadi
4	SIPK_01_11	Melihat detail titik kecelakaan	Dengan fungsi ini admin dapat melihat grafik perkembangan kecelakaan pada titik kecelakaan yang dipilih
5	SIPK_01_12	Melihat detail jalan	Dengan fungsi ini admin dapat melihat grafik perkembangan kecelakaan pada ruas jalan yang dipilih
6	SIPK_01_13	Melihat statistik	Dengan fungsi ini admin dapat melihat statistik kecelakaan dalam banyak model

7	SIPK_01_14	Melihat koordinat	data	Dengan fungsi ini admin bisa melihat data koordinat dalam bentuk tabel
8	SIPK_01_15	Menambah koordinat	data	Dengan fungsi ini admin dapat menambah koordinat lokasi kecelakaan. Data yang disimpan adalah latitude, longitude, dan kode lokasi
9	SIPK_01_16	Mengubah koordinat	data	Dengan fungsi ini admin dapat mengubah koordinat lokasi kecelakaan. Data yang disimpan adalah latitude, longitude, dan kode lokasi
10	SIPK_01_17	Menghapus koordinat	data	Dengan fungsi ini admin dapat menghapus data koordinat
11	SIPK_01_18	Melihat kecelakaan	data	Dengan fungsi ini admin bisa melihat data kecelakaan dalam bentuk tabel
12	SIPK_01_19	Menambah kecelakaan	data	Dengan fungsi ini admin dapat menambah data kecelakaan. Data yang dapat disimpan adalah kode titik, nama jalan, kecamatan, total kecelakaan, total korban meninggal, total korban luka berat, total korban luka ringan, dan waktu kejadian.
13	SIPK_01_20	Mengubah kecelakaan	data	Dengan fungsi ini admin dapat mengubah data kecelakaan yang sudah ada dengan data kecelakaan yang baru
14	SIPK_01_21	Menghapus kecelakaan	data	Dengan fungsi ini admin dapat menghapus data kecelakaan
15	SIPK_01_22	Mencari Jalan		Dengan fungsi ini admin dapat mencari jalan berdasarkan nama jalan

16	SIPK_01_23	Melihat Perbandingan	Dengan fungsi ini admin dapat membandingkan informasi kecelakaan (total kecelakaan, total luka ringan, total luka berat, dan total meninggal) dari 2 wilayah kecamatan
17	SIPK_01_24	Mencetak Statistik	Dengan fungsi ini Admin dapat mencetak statistik yang ada di halaman statistik sebagai laporan

4.5.3 Kebutuhan Non-Fungsional

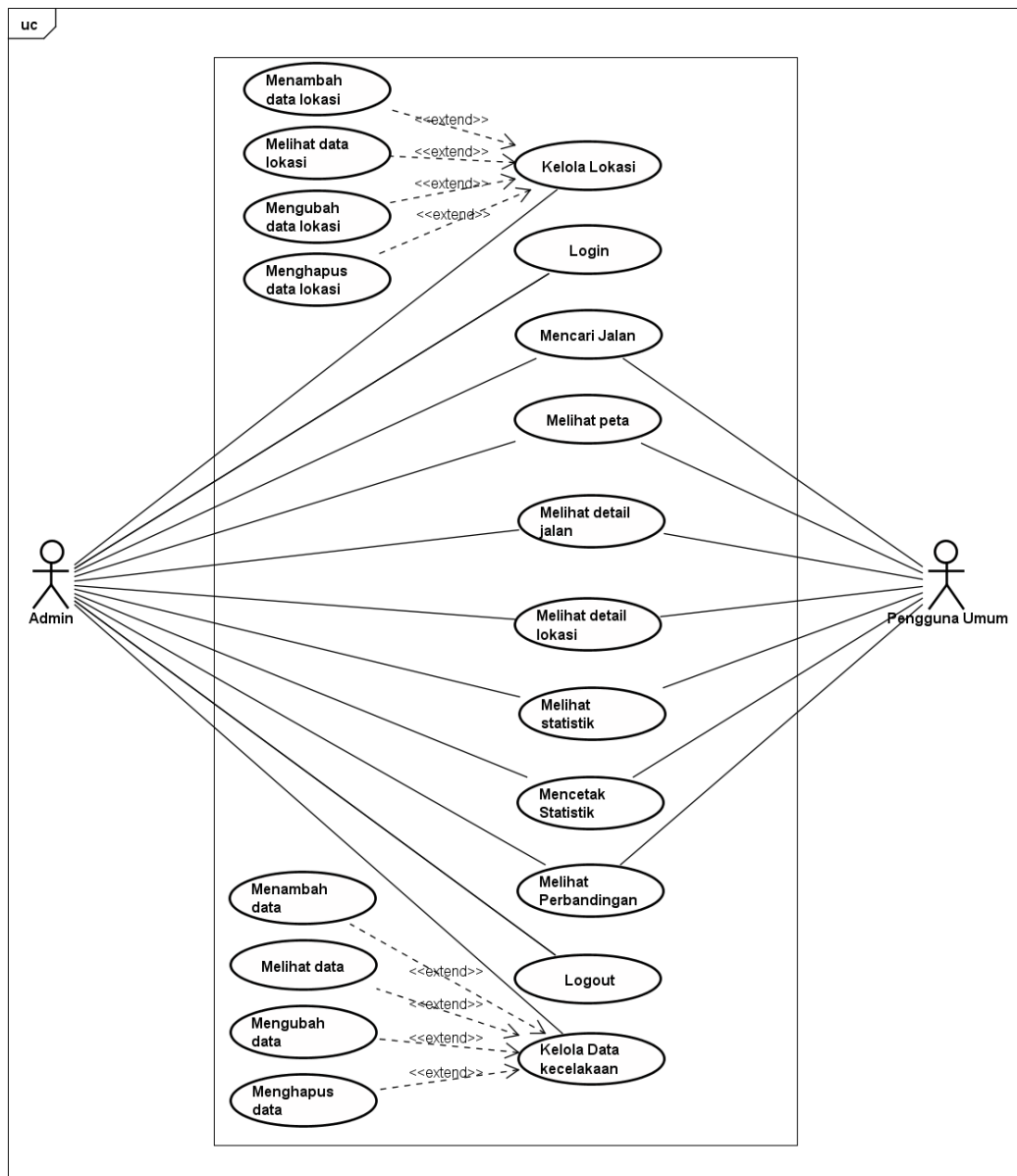
Kebutuhan non-fungsional sistem merupakan suatu batasan layanan atau standarisasi dan fungsi yang ditawarkan oleh sistem. Kebutuhan non-fungsional Sistem Informasi Pemetaan Lokasi Kecelakaan Kota Batu berbasis *WebGIS* dapat dilihat pada Tabel 4.8.

Table 4.8 Tabel Kebutuhan Non-Fungsional

No.	Kode Fungsi	Paramater	Keterangan
1	SIK_02_01	<i>Compatibility</i>	Antarmuka sistem untuk pengguna umum dapat dibuka dengan berbagai macam <i>platform</i> .

4.5.4 Pemodelan Use Case

Use case diagram pada gambar 4.13 digunakan untuk menggambarkan *behavior* sistem dan interaksi antar aktor. *Use case* pada gambar 4.13 terdiri dari 2 aktor yaitu admin dan pengguna umum. Admin memiliki 17 fungsional, sedangkan masyarakat umum memiliki 7 fungsional.



Gambar 4.13 Use case Diagram

4.5.5 Skenario Use Case

Use case scenario merupakan penjabaran alur kinerja atau step-step setiap use case. Pada *use case scenario* yang akan dijelaskan hanya mengambil 4 kebutuhan fungsional. Fungsi yang dijelaskan merupakan fungsi utama pada sistem yang dikembangkan.

4.5.5.1 Skenario Use Case Menambah Data Koordinat

Pada Tabel 4.9 dijelaskan mengenai tahapan alur kerja pada sistem untuk kebutuhan fungsional menambah data koordinat. Fungsi menambah data kecelakaan dapat dikerjakan oleh 1 aktor, yaitu admin sistem. Kondisi awal dari alur kerja ini dimulai dari sistem menampilkan halaman kelola koordinat.

Sedangkan kondisi akhir dari alur kerja ini adalah aktor berhasil menambah data koordinat.

Table 4.9 Use Case Scenario Menambah Data Koordinat

Skenario Kasus	
Nomor Use Case	SIPK_01_15
Nama	Menambah Data Koordinat
Tujuan	Admin berhasil menambahkan koordinat lokasi kecelakaan baru
Deskripsi	Use case ini menjelaskan bagaimana admin menambahkan koordinat lokasi kecelakaan baru
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan halaman kelola koordinat
Aksi Aktor	Reaksi Sistem
1. Admin memilih tombol tambah lokasi	2. Sistem menjalankan fungsi untuk menampilkan peta
	3. Sistem mengambil data dari <i>database</i>
	4. Sistem mengembalikan data dari <i>database</i>
	5. Sistem menampilkan halaman tambah lokasi
6. Admin mengarahkan kursor kedalam peta dan memilih lokasi yang diinginkan. Kemudian mengisi kode lokasi kedalam form. Admin memilih tombol tambah	7. Sistem menjalankan fungsi
	8. Sistem mengirimkan data ke <i>database</i> untuk disimpan
	9. Sistem menampilkan halaman kelola koordinat
10. Admin melihat halaman kelola koordinat	
Skenario Alternatif 1: Jika form Kode kosong	

	11. Tombol Tambah tidak dapat ditekan
Skenario Alternatif 2: Jika form Latitude kosong	
	12. Sistem akan menampilkan pesan <i>"Please fill out this field"</i>
Skenario Alternatif 3: Jika form Longitude kosong	
	13. Sistem akan menampilkan pesan <i>"Please fill out this field"</i>
Kondisi Akhir	Aktor dapat menambah koordinat lokasi kecelakaan dan sistem menampilkan halaman kelola koordinat.

4.5.5.2 Skenario *Use Case* Menambah Data Kecelakaan

Pada Tabel 4.10 dijelaskan mengenai tahapan alur kerja pada sistem untuk kebutuhan fungsional menambah data kecelakaan. Fungsi menambah data kecelakaan dapat dikerjakan oleh 1 aktor, yaitu admin sistem. Kondisi awal dari alur kerja ini dimulai dari sistem menampilkan halaman kelola data. Sedangkan kondisi akhir dari alur kerja ini adalah aktor berhasil menambah data kecelakaan.

Table 4.10 *Use Case Scenario* Menambah Data Kecelakaan

Skenario Kasus	
Nomor Use Case	SIPK_01_19
Nama	Menambah data kecelakaan
Tujuan	Admin berhasil menambahkan data kecelakaan baru
Deskripsi	Use case ini menjelaskan bagaimana admin menambahkan data kecelakaan baru
Aktor	Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan kelola data
Aksi Aktor	Reaksi Sistem
1. Admin memilih menu tambah data	2. Sistem menampilkan halaman tambah data kecelakaan

3. Admin mengisi form data kecelakaan. Kemudian memilih tombol tambah	4. Sistem menjalankan fungsi
	5. Sistem mengirimkan data ke <i>database</i> untuk disimpan
	6. Sistem menampilkan halaman kelola data
7. Admin melihat halaman kelola data	
Skenario Alternatif 1: Jika form Kode Titik kosong	
	8. Sistem akan menampilkan pesan <i>"Please select an item in the list"</i>
Skenario Alternatif 2: Jika form Kecamatan kosong	
	9. Sistem akan menampilkan pesan <i>"Please select an item in the list"</i>
Skenario Alternatif 3: Jika form Total kosong	
	10. Sistem akan menampilkan pesan <i>"Please fill out this field"</i>
Skenario Alternatif 4: Jika form Meninggal kosong	
	11. Sistem akan menampilkan pesan <i>"Please fill out this field"</i>
Skenario Alternatif 5: Jika form Luka Berat kosong	
	12. Sistem akan menampilkan pesan <i>"Please fill out this field"</i>
Skenario Alternatif 6: Jika form Luka Ringan kosong	
	13. Sistem akan menampilkan pesan <i>"Please fill out this field"</i>
Skenario Alternatif 7: Jika form Waktu kosong	
	14. Sistem akan menampilkan pesan <i>"Please fill out this field"</i>

Kondisi Akhir	Admin berhasil menambah data kecelakaan dan sistem menampilkan halaman data kecelakaan
----------------------	--

4.5.5.3 Skenario *Use Case* Melihat Peta Kecelakaan

Pada Tabel 4.11 dijelaskan mengenai tahapan alur kerja pada sistem untuk kebutuhan fungsional melihat peta kecelakaan. Fungsi melihat peta kecelakaan dapat dikerjakan oleh 2 aktor, yaitu pengguna umum atau admin sistem. Kondisi awal dari alur kerja ini dimulai dari sistem menampilkan halaman selamat datang. Sedangkan kondisi akhir dari alur kerja ini adalah aktor berhasil melihat peta kecelakaan.

Table 4.11 Use Case Scenario Melihat Peta Kecelakaan

Skenario Kasus	
Nomor Use Case	SIPK_01_01 atau SIPK_01_10
Nama	Melihat Peta Kecelakaan
Tujuan	Pengguna Umum atau Admin berhasil melihat peta kecelakaan Kota Batu
Deskripsi	Use case ini menjelaskan bagaimana aksi aktor dan reaksi sistem sehingga aktor dapat melihat peta kecelakaan Kota Batu
Aktor	Pengguna Umum atau Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan halama selamat datang sistem
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu peta	2. Sistem menjalankan fungsi
	3. Sistem mengambil data dari <i>database</i>
	4. Sistem mengembalikan data dari <i>database</i>
	5. Sistem menampilkan data yang telah diambil dalam bentuk peta
6. Aktor melihat peta	
Kondisi Akhir	Aktor berhasil melihat peta kecelakaan daerah operasional Kota Batu.

4.5.5.4 Skenario Use Case Melihat Statistik

Pada Tabel 4.12 dijelaskan mengenai tahapan alur kerja pada sistem untuk kebutuhan fungsional melihat statistik. Fungsi melihat statistik dapat dikerjakan oleh 2 aktor, yaitu pengguna umum atau admin sistem. Kondisi awal dari alur kerja ini dimulai dari sistem menampilkan halaman selamat datang. Sedangkan kondisi akhir dari alur kerja ini adalah aktor berhasil melihat statistik kecelakaan.

Table 4.12 Use Case Scenario Melihat Statistik

Skenario Kasus	
Nomor Use Case	SIPK_01_04 atau SIPK_01_10
Nama	Melihat Statistik
Tujuan	Pengguna Umum atau Admin berhasil melihat statistik kecelakaan dalam banyak model
Deskripsi	Use case ini menjelaskan bagaimana cara melihat melihat statistik kecelakaan dalam banyak model
Aktor	Pengguna Umum atau Admin
Skenario Utama	
Kondisi Awal	Sistem menampilkan halaman selamat datang
Aksi Aktor	Reaksi Sistem
1. Aktor memilih menu statistik	2. Sistem menjalankan fungsi
	3. Sistem mengambil data dari <i>database</i>
	4. Sistem mengembalikan fungsi
	5. Sistem menampilkan data yang telah diambil dari <i>database</i> dalam beberapa bentuk grafik.
6. Aktor melihat halaman statistik	
Kondisi Akhir	Aktor dapat melihat statistik kecelakaan

4.6 Perancangan Sistem

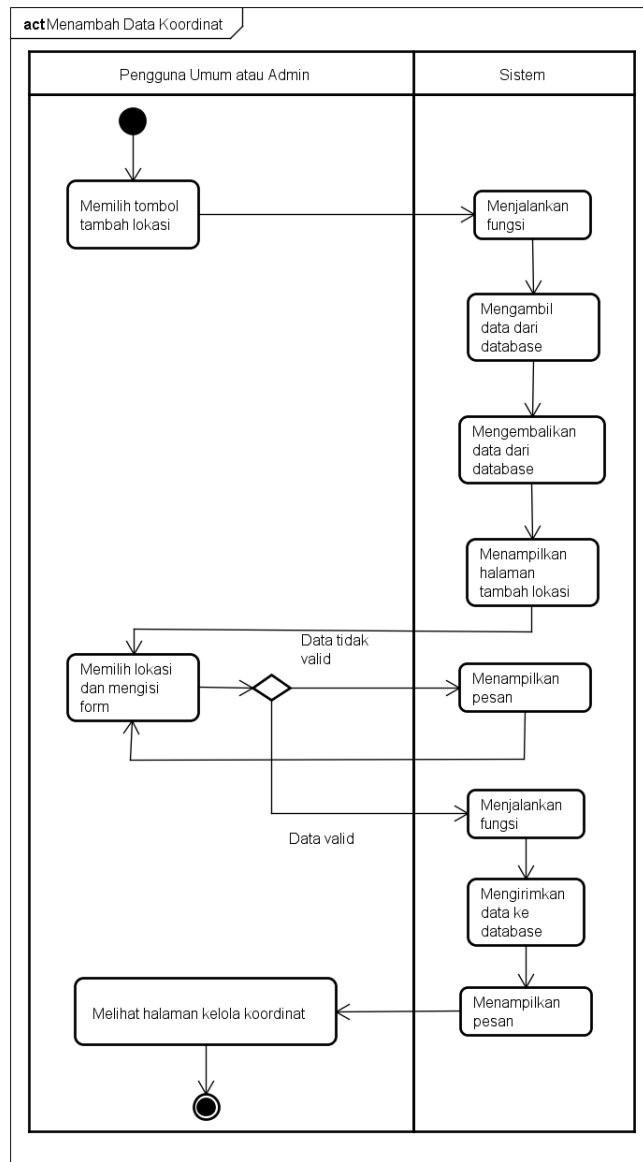
Pada bagian ini akan dijelaskan mengenai pemodelan sistem. Kebutuhan fungsional yang sudah didefinisikan pada tabel 4.6 dan tabel 4.7 akan dimodelkan menjadi beberapa diagram, yaitu *use case diagram*, *sequence diagram*, dan *class*

diagram. Perancangan *database* dan perancangan antarmuka juga akan dijelaskan pada bagian ini.

4.6.1 Perancangan *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan alur dari interaksi sistem dengan pengguna. Pada diagram ini biasanya digunakan bahasa-bahasa yang lebih mudah untuk dipahami oleh orang umum. Diagram ini ditujukan kepada pengembang yang tidak terlalu mengerti bahasa pemrograman, namun mengerti alur dari sebuah sistem.

4.6.1.1 *Activity Diagram* Menambah Lokasi Kecelakaan

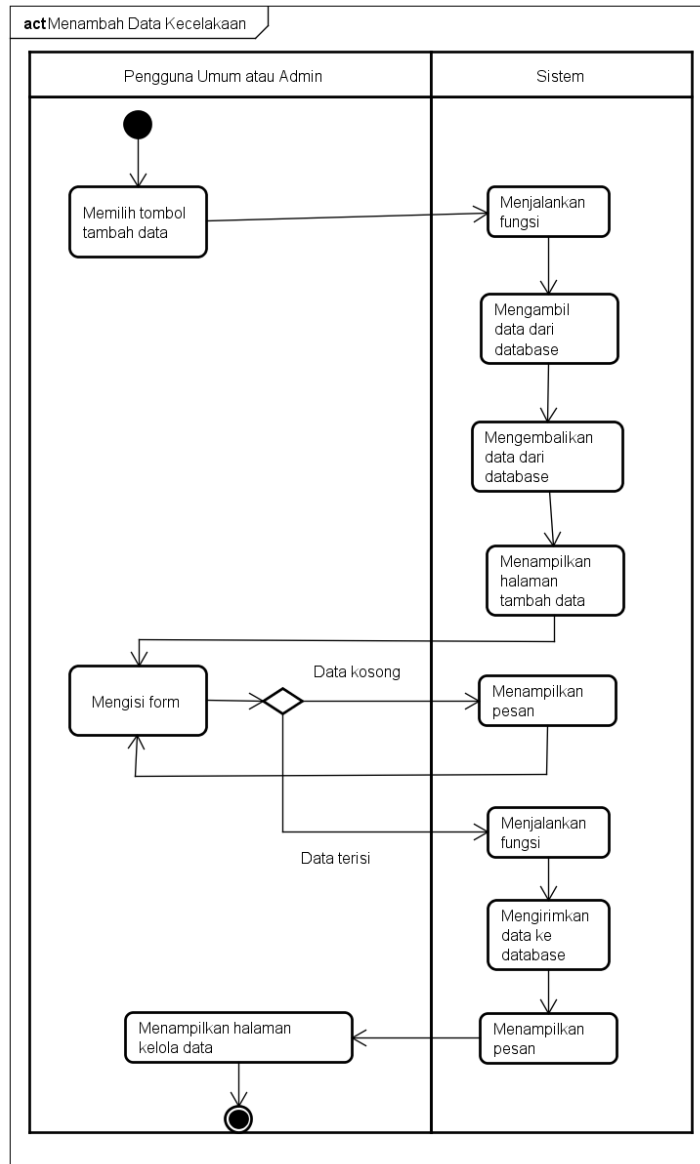


Gambar 4.14 *Activity Diagram* Menambah Lokasi Kecelakaan

Pada Gambar 4.14 merupakan *activity diagram* menambah lokasi kecelakaan yang menggambarkan berbagai aliran aktivitas dalam sistem ketika pengguna

melakukan proses menambah lokasi kecelakaan, aktivitas ini dimulai dengan menekan menu data kecelakaan kemudian menekan tombol kelola koordinat dan aktivitas akhir sistem akan menampilkan halaman kelola koordinat beserta data yang telah dimasukkan.

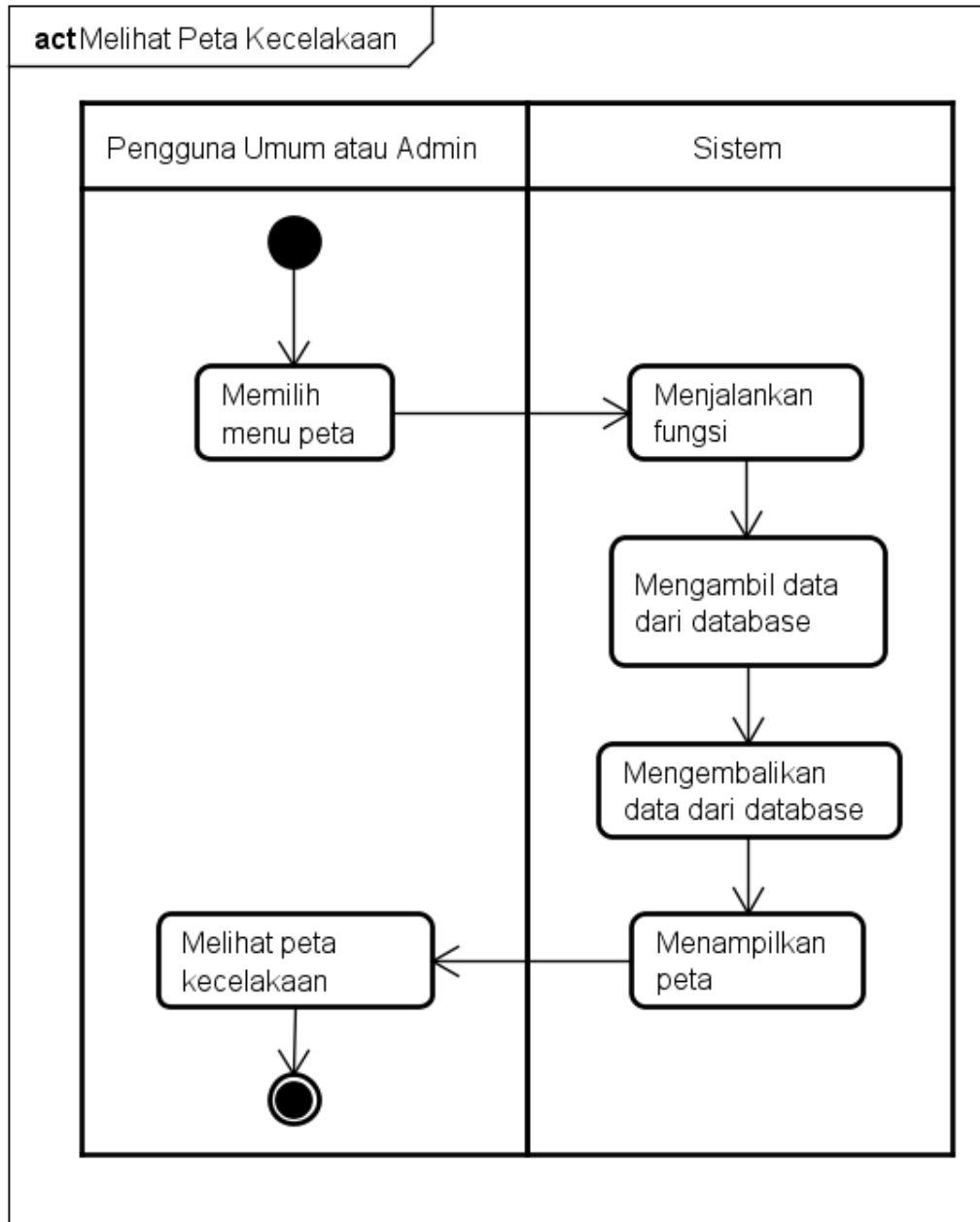
4.6.1.2 Activity Diagram Menambah Data Kecelakaan



Gambar 4.15 Activity Diagram Menambah Data Kecelakaan

Pada Gambar 4.15 merupakan *activity diagram* menambah data kecelakaan yang menggambarkan berbagai aliran aktivitas dalam sistem ketika pengguna melakukan proses menambah data kecelakaan, aktivitas ini dimulai dengan menekan menu data kecelakaan kemudian menekan tombol kelola data dan aktivitas akhir sistem akan menampilkan halaman kelola data beserta data yang telah dimasukkan.

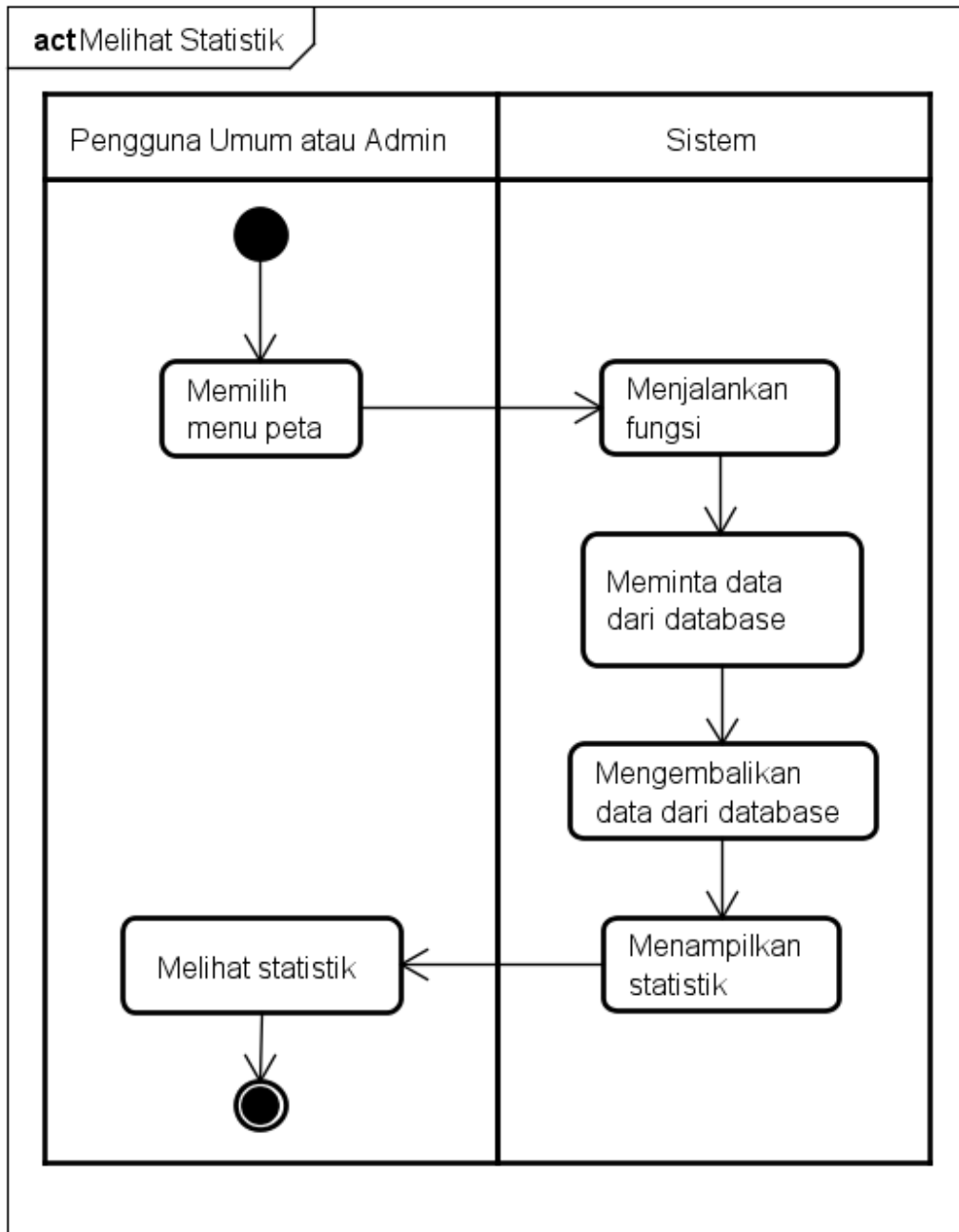
4.6.1.3 Activity Diagram Melihat Peta Kecelakaan



Gambar 4.16 Activity Diagram Melihat Peta Kecelakaan

Pada Gambar 4.16 ditunjukkan *activity diagram* untuk menggambarkan berbagai aliran aktivitas dalam sistem ketika pengguna melakukan proses melihat peta kecelakaan, aktivitas ini dimulai dengan menekan menu peta dan aktivitas akhir sistem akan menampilkan halaman peta kecelakaan.

4.6.1.4 Activity Diagram Melihat Statistik



Gambar 4.17 Activity Diagram Melihat Statistik

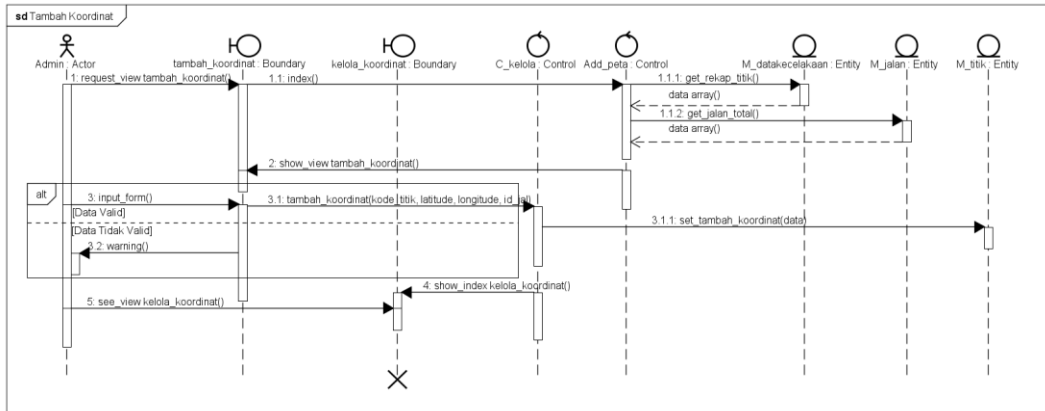
Pada Gambar 4.17 ditunjukkan *activity diagram* untuk menggambarkan berbagai aliran aktivitas dalam sistem ketika pengguna melakukan proses melihat statistik kecelakaan, aktivitas ini dimulai dengan menekan menu statistik dan aktivitas akhir sistem akan menampilkan halaman statistik kecelakaan.

4.6.2 Perancangan *Sequence Diagram*

Perancangan *sequence diagram* digunakan untuk menggambarkan interaksi antar objek dan menunjukkan serangkaian pesan yang dipertukarkan antar objek

saat melakukan tugas tertentu. Diagram ini biasanya menggunakan bahasa-bahasa mesin yang lebih mudah dipahami oleh *programmer*. Pada perancangan diagram kali ini hanya digambarkan 4 kebutuhan yang paling utama saja. Kebutuhan tersebut antara lain melihat peta kecelakaan, melihat statistik kecelakaan, menambah data koordinat, dan menambah data kecelakaan.

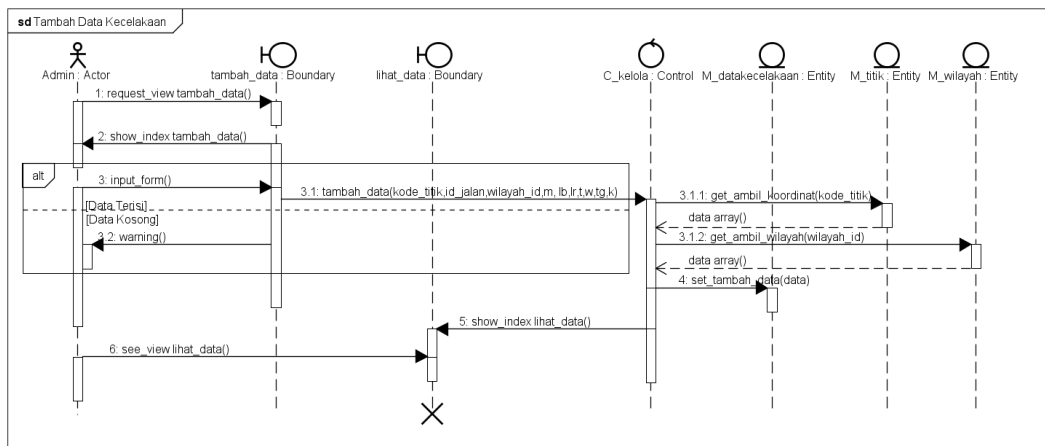
4.6.2.1 Sequence Diagram Menambah Data Koordinat



Gambar 4.18 Sequence Diagram Menambah Data Koordinat

Pada gambar 4.18 digambarkan hubungan antar objek saat melakukan fungsi menambah data kecelakaan. Dimulai dari aktor berinteraksi dengan *view* tambah_lokasi untuk menampilkan halaman tambah lokasi kecelakaan. Aktor mengisi form pada halaman tambah data kecelakaan dan memilih tombol tambah. Jika data kosong, maka muncul pesan gagal. Jika data terisi, maka dilanjutkan dengan objek *c_kelola* menjalankan fungsi tambah_data untuk memasukkan data baru ke *database*. Setelah itu muncul pesan sukses.

4.6.2.2 Sequence Diagram Menambah Data Kecelakaan

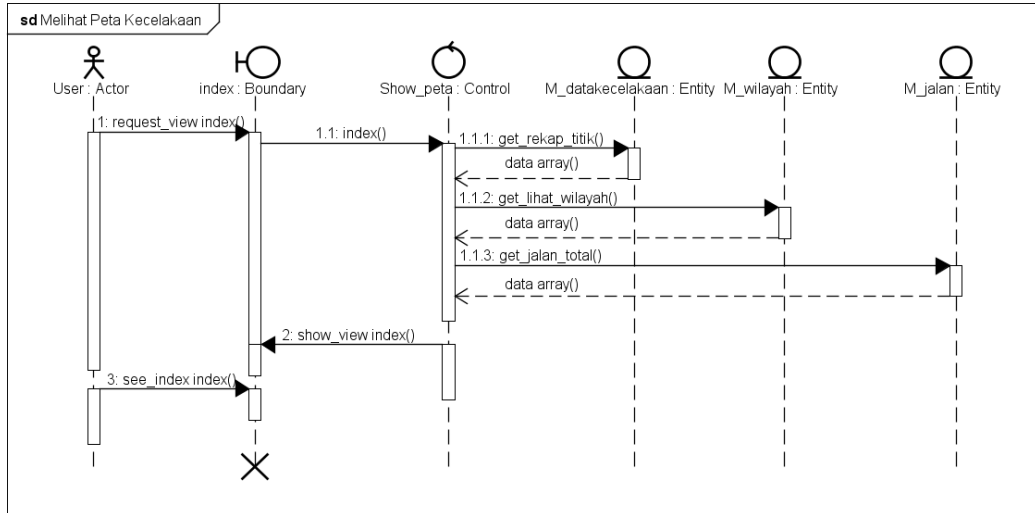


Gambar 4.19 Sequence Diagram Menambah Data Kecelakaan

Pada gambar 4.19 digambarkan hubungan antar objek saat melakukan fungsi menambah data kecelakaan. Dimulai dari aktor berinteraksi dengan *view* tambah_data untuk menampilkan halaman tambah data kecelakaan. Aktor

mengisi form pada halaman tambah data kecelakaan dan memilih tombol tambah. Jika data kosong, maka muncul pesan gagal. Jika data terisi, maka dilanjutkan dengan objek `c_kelola` menjalankan fungsi `tambah_data` untuk memasukkan data baru ke *database*. Setelah itu muncul pesan sukses.

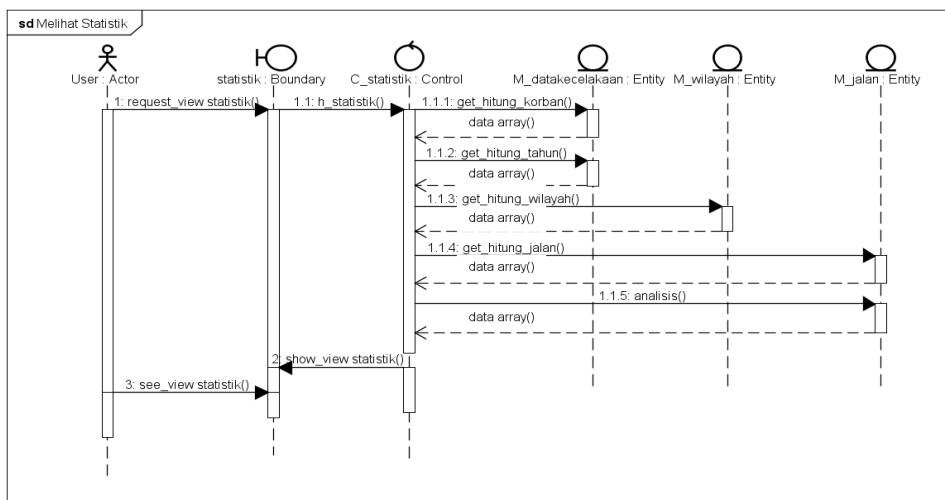
4.6.2.3 Sequence Diagram Melihat Peta Kecelakaan



Gambar 4.20 Sequence Diagram Melihat Peta Kecelakaan

Pada gambar 4.20 digambarkan hubungan antar objek saat melakukan fungsi melihat peta kecelakaan. Dimulai dari aktor berinteraksi dengan `view index` untuk menjalankan fungsi pada objek peta. Diteruskan dengan objek peta menjalankan fungsi `rekap titik` untuk mendapatkan data koordinat lokasi dan data kecelakaan dari *database* melalui `m_kelola`. Data yang sudah didapatkan dikirim kembali ke objek peta dan diteruskan ke `view index` untuk ditampilkan dalam bentuk peta.

4.6.2.4 Sequence Diagram Melihat Statistik

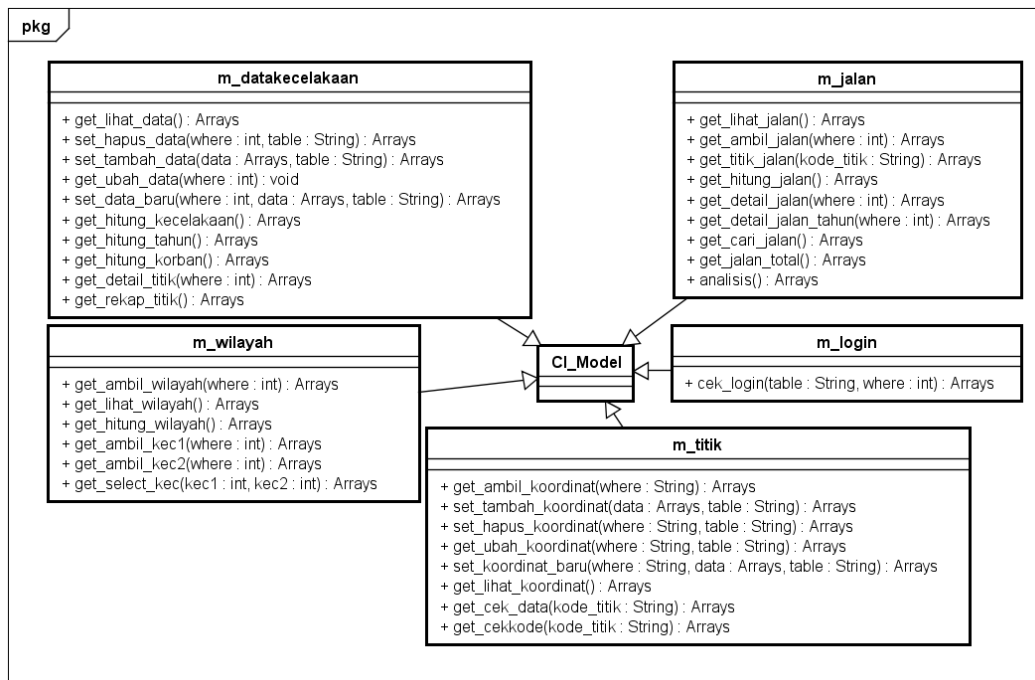


Gambar 4.21 Sequence Diagram Melihat Statistik

Pada gambar 4.21 digambarkan hubungan antar objek saat melakukan fungsi melihat statistik. Dimulai dari aktor berinteraksi dengan *view* statistik untuk menjalankan fungsi pada objek *c_statistik*. Diteruskan dengan objek *c_statistik* menjalankan fungsi hitung untuk mendapatkan data statistik kecelakaan dari *database* melalui *m_statistik*. Data yang sudah didapatkan dikirim kembali ke objek peta dan diteruskan ke *view* index untuk ditampilkan dalam bentuk peta.

4.6.3 Perancangan *Class Diagram*

Pada perancangan *class diagram* digambarkan mengenai struktur sistem dalam bentuk pendefinisian kelas-kelas yang digunakan dalam membangun sistem.

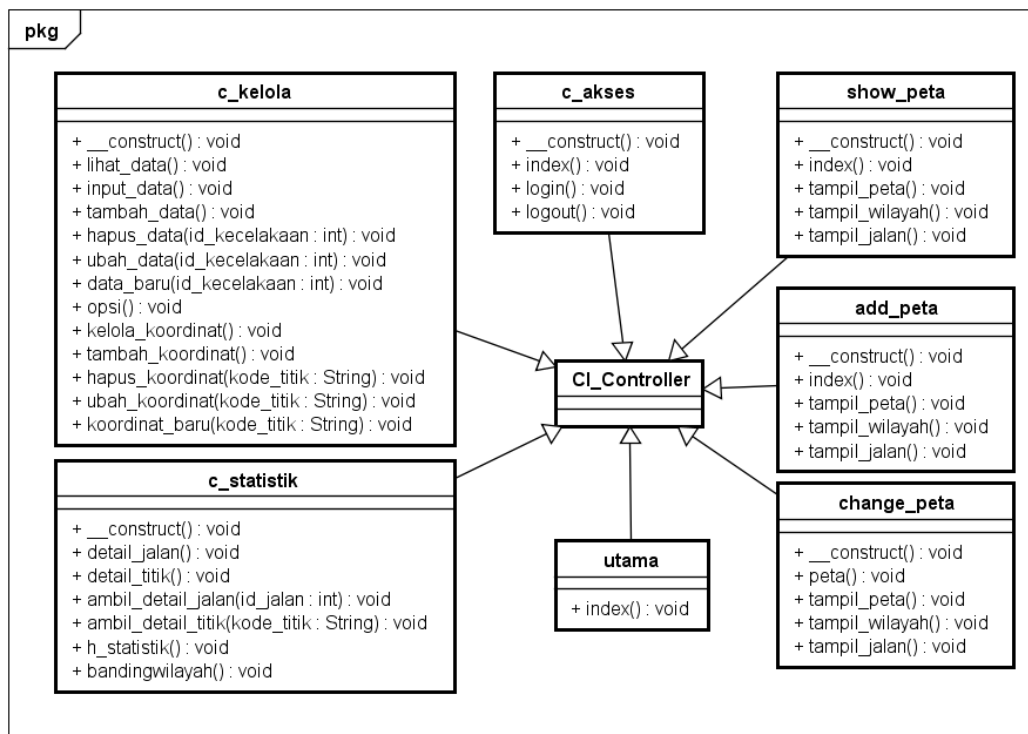


Gambar 4.22 Class Diagram Model

Pada gambar 4.22 adalah komponen *class diagram controller* dari sistem yang akan dibangun. Komponen tersebut terdiri dari:

1. *m_login*
2. *m_jalan*
3. *m_titik*
4. *m_wilayah*
5. *m_datakecelakaan*

Kelima komponen tersebut melakukan *extends* ke *ci_model* agar dapat menjalankan fungsi-fungsi pada *model* yang disediakan oleh *code igniter*. Semua *model* ini digunakan untuk menghubungkan *controller* ke *database* untuk mendapatkan data yang dibutuhkan. Masing-masing *model* ini mempunyai hubungan ke *controller* dengan dan tanpa membawa parameter tertentu.



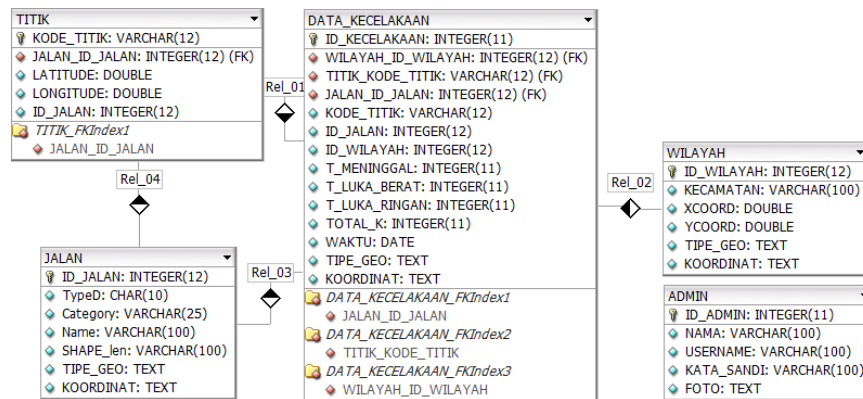
Gambar 4.23 Class Diagram Controller

Pada gambar 4.23 adalah komponen *class diagram controller* dari sistem yang akan dibangun. Komponen tersebut terdiri dari:

1. c_kelola
2. c_akses
3. show_peta
4. add_peta
5. change_peta
6. utama
7. c_statistik

Ketujuh komponen tersebut melakukan *extends* ke *ci_controller* agar dapat menjalankan fungsi-fungsi pada *controller* yang disediakan oleh *code igniter*. *Controller* ini digunakan untuk menyimpan beberapa *method* untuk menjalankan fungsi pada sistem. Untuk mendapatkan data, *controller* ini akan melakukan koneksi ke *model* dengan dan tanpa membawa parameter tertentu.

4.6.4 Perancangan Data Model



Gambar 4.24 Perancangan Data Model

Mengacu pada gambar 4.24, bahwa terdapat 5 tabel pada rancangan *database*. Tabel tersebut adalah tabel Admin, tabel Data Kecelakaan, tabel Jalan, dan tabel Wilayah, dan tabel Titik. Tabel-tabel ini didapat dari beberapa cara, antara lain dari studi dokumen yang ada di Polres Kota Batu dan beberapa kebutuhan yang harus ada saat mengembangkan sistem berbasis *WebGIS*.

Tabel 4.13 merupakan rincian dari Tabel Admin. Tabel ini digunakan untuk menyimpan data admin pada *database*. Terdapat 5 kolom pada tabel ini.

4.6.4.1 Tabel Admin

- Nama tabel : Admin
- Jumlah field : 5
- Fungsi : Untuk menyimpan data admin

Table 4.13 Tabel Admin

No	Nama Field	Tipe	Lebar	Keterangan
1	ID_ADMIN	INT	11	ID Admin
2	NAMA	VARCHAR	100	Nama Admin
3	USERNAME	VARCHAR	100	Username Admin
4	KATA_SANDI	VARCHAR	100	Kata Sandi Admin
5	FOTO	TEXT	-	Foto Admin

Tabel 4.14 merupakan rincian dari Tabel Data Kecelakaan. Tabel ini digunakan untuk menyimpan data -data kecelakaan yang akan digunakan pada sistem. Terdapat 12 kolom pada tabel ini.

4.6.4.2 Tabel Data Kecelakaan

- Nama tabel : Data Kecelakaan
- Jumlah field : 12

Fungsi : Untuk menyimpan data kecelakaan

Table 4.14 Tabel Data Kecelakaan

No	Nama Field	Tipe	Lebar	Keterangan
1	ID_DATA KECELAKAAN	INT	11	ID Data Kecelakaan
2	KODE_TITIK	VARCHAR	12	Kode Titik sebagai <i>foreign key</i> dari tabel titik
3	ID_JALAN	INT	12	ID Jalan sebagai <i>foreign key</i> dari tabel jalan
4	ID_WILAYAH	INT	12	ID Wilayah sebagai <i>foreign key</i> dari tabel wilayah
5	K_KECAMATAN	VARCHAR	100	Wilayah Kecelakaan
6	T_MENINGGAL	INT	11	Total Korban Meninggal
7	T_LUKA_BERAT	INT	11	Total Korban Luka Berat
8	T_LUKA_RINGAN	INT	11	Total Korban Luka Ringan
9	TOTAL_K	INT	12	Total Kejadian Kecelakaan
10	WAKTU	DATE	-	Waktu Kejadian
11	TIPE_GEO	TEXT	-	Tipe geometry pada peta
12	KOORDINAT	TEXT	-	Koordinat Lokasi Kecelakaan

Tabel 4.15 merupakan rincian dari Tabel Wilayah. Tabel ini digunakan untuk menyimpan data spasial wilayah pada *database*. Terdapat 6 kolom pada tabel ini.

4.6.4.3 Tabel Wilayah

Nama tabel : Wilayah

Jumlah field : 6

Fungsi : Untuk meyimpan data wilayah

Table 4.15 Tabel Wilayah

No	Nama Field	Tipe	Lebar	Keterangan
1	ID_WILAYAH	INT	12	ID Wilayah
2	KECAMATAN	VARCHAR	100	Nama Kecamatan
3	XCOORD	DOUBLE	-	Latitude Wilayah

4	YCOORD	DOUBLE	-	Longitude Wilayah
5	TIPE_GEO	TEXT	-	Tipe geometry pada peta
6	KOORDINAT	TEXT	-	Koordinat Wilayah

Tabel 4.16 merupakan rincian dari Tabel Titik. Tabel ini digunakan untuk menyimpan data spasial titik pada *database*. Terdapat 3 kolom pada tabel ini.

4.6.4.4 Tabel Titik

Nama tabel : Titik

Jumlah field : 3

Fungsi : Untuk menyimpan data koordinat lokasi kecelakaan

Table 4.16 Tabel Titik

No	Nama Field	Tipe	Lebar	Keterangan
1	KODE_TITIK	VARCHAR	12	Kode untuk setiap titik
2	LATITUDE	DOUBLE	-	Koordinat latitude
3	LONGITUDE	DOUBLE	-	Koordinat longitude
4	ID_JALAN	INT	12	ID Jalan sebagai <i>foreign key</i> dari tabel jalan

Tabel 4.17 merupakan rincian dari Tabel Jalan. Tabel ini digunakan untuk menyimpan data spasial jalan pada *database*. Terdapat 7 kolom pada tabel ini.

4.6.4.5 Tabel Jalan

Nama tabel : Jalan

Jumlah field : 7

Fungsi : Untuk menyimpan data jaringan jalan

Table 4.17 Tabel Jalan

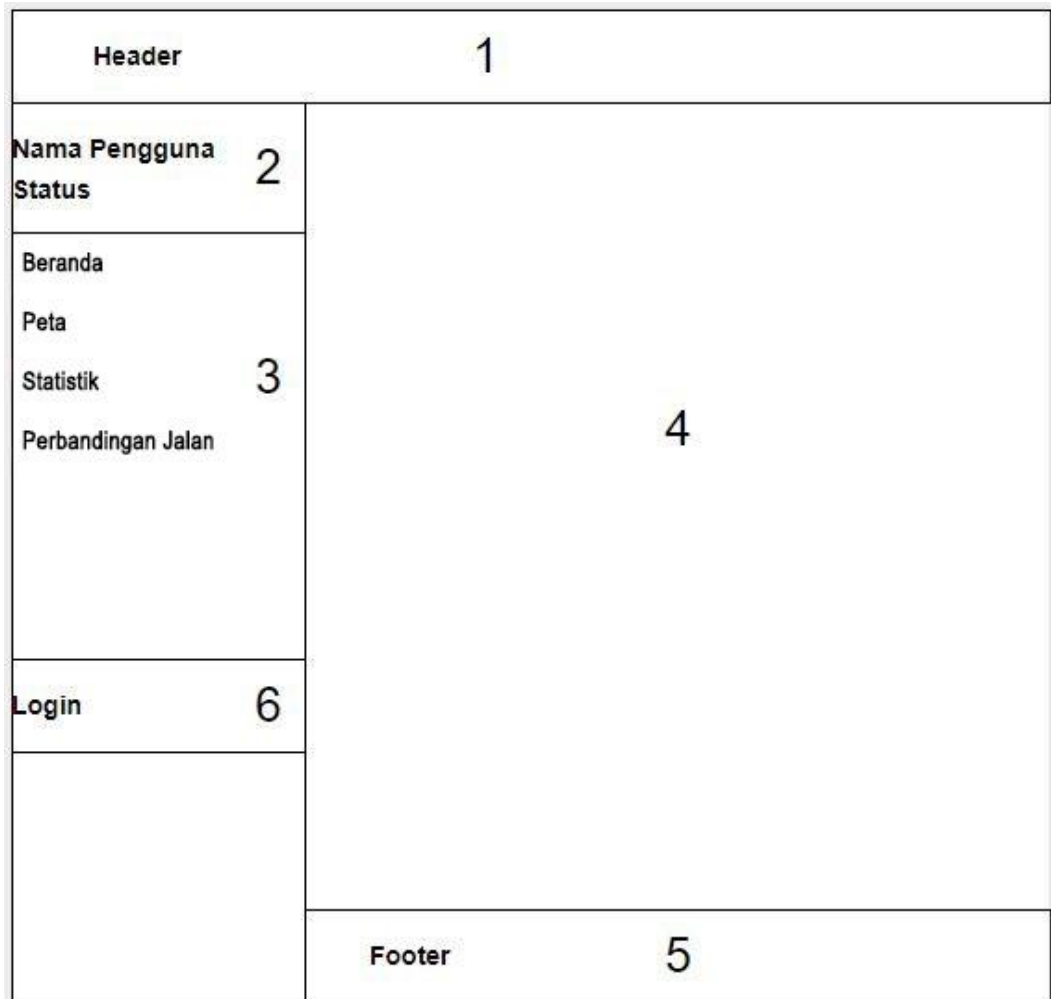
No	Nama Field	Tipe	Lebar	Keterangan
1	ID_JALAN	INT	12	ID Jalan
2	TypeD	CHAR	10	-
3	Category	VARCHAR	25	Kategori Jalan
4	Name	VARCHAR	100	Nama Jalan
5	SHAPE_len	VARCHAR	100	Panjang Jalan
6	TIPE_GEO	TEXT	-	Tipe geometry pada peta
7	KOORDINAT	TEXT	-	Koordinat Jalan

4.7 Perancangan Antarmuka

Perancangan antarmuka digunakan untuk menggambarkan tampilan antarmuka yang akan dibangun. Pada perancangan ini hanya digambarkan dalam bentuk *mockup* dan tata letak tombol, menu, *field* yang akan digunakan.

4.7.1 Perancangan Antarmuka

4.7.1.1 Halaman Utama

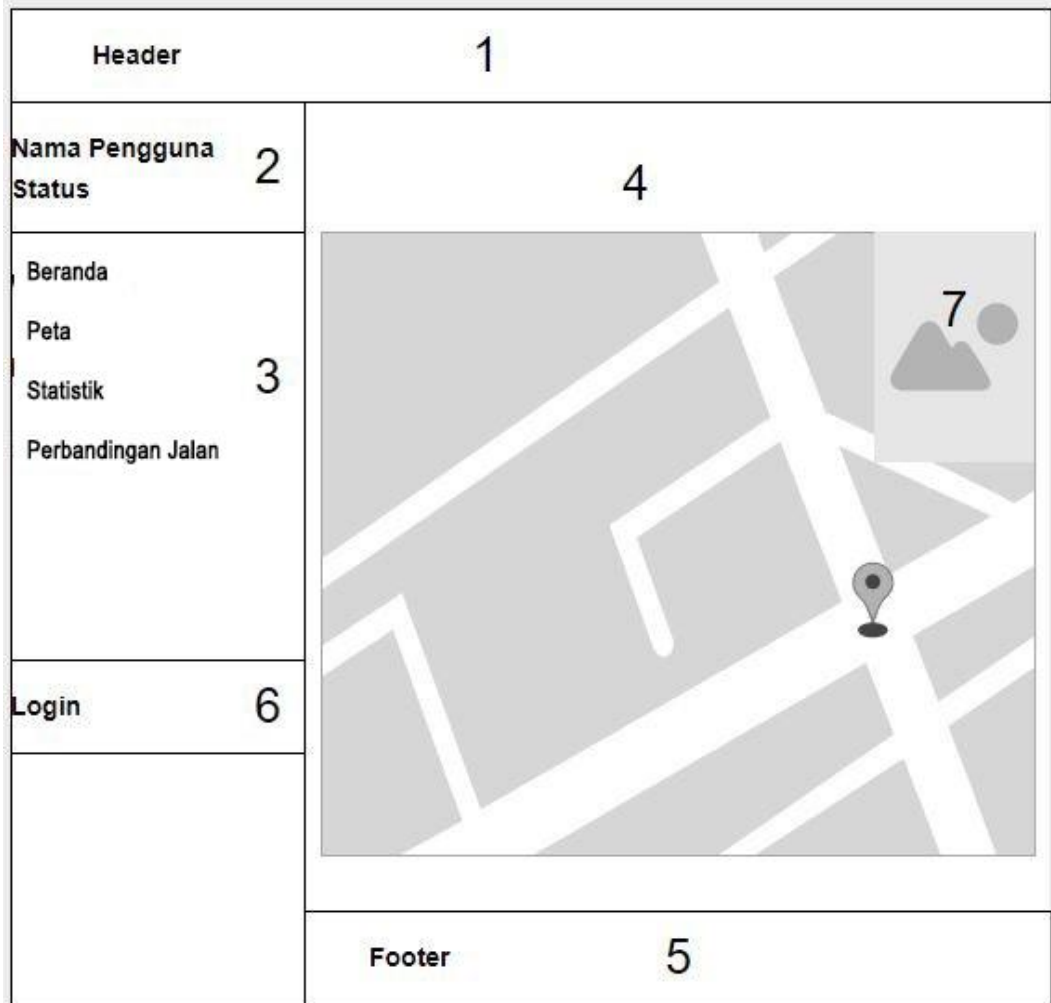


Gambar 4.25 Halaman Utama

Keterangan gambar 4.25 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi deskripsi singkat sistem.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol login untuk mengarahkan ke halaman *login*

4.7.1.2 Halaman Peta

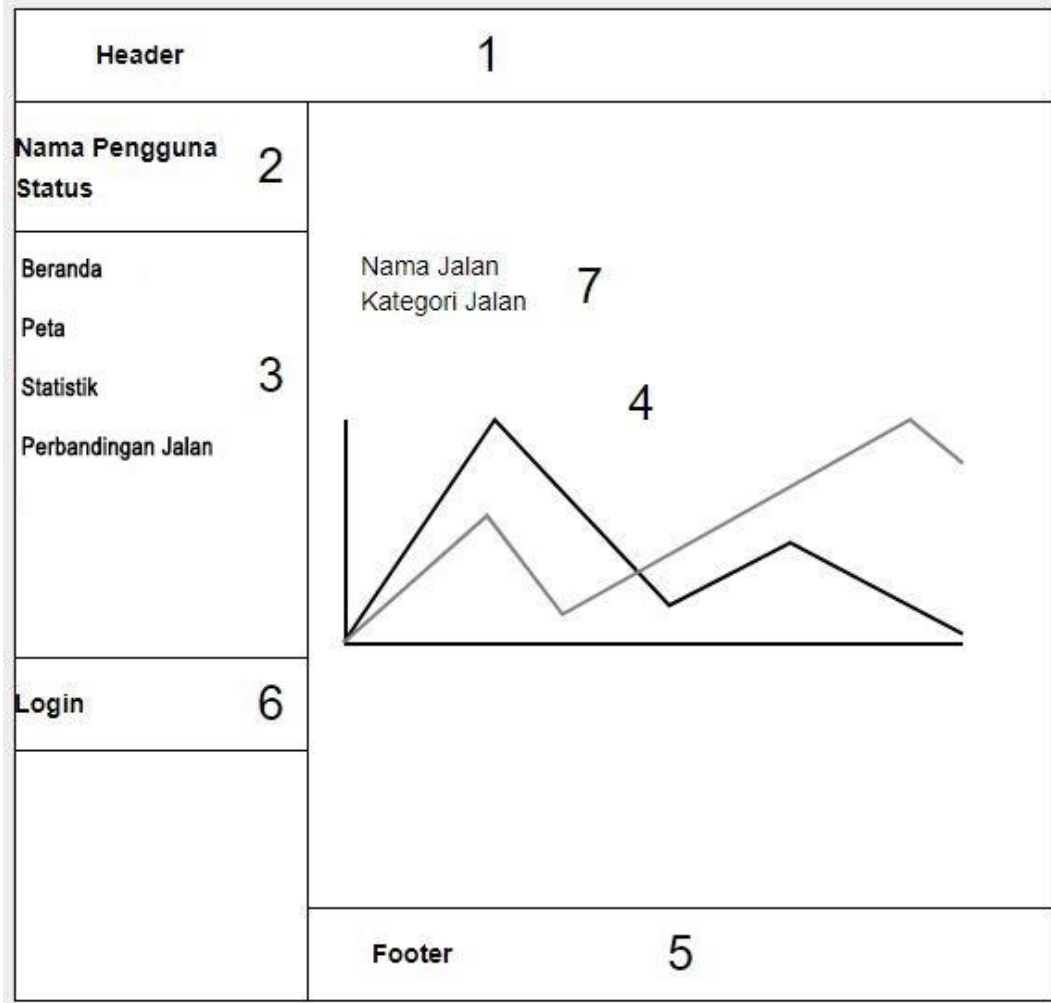


Gambar 4.26 Halaman Peta

Keterangan gambar 4.26 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi peta kecelakaan.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol *login* untuk mengarahkan ke halaman *login*.
7. *Legend* peta kecelakaan.

4.7.1.3 Halaman Detail Jalan

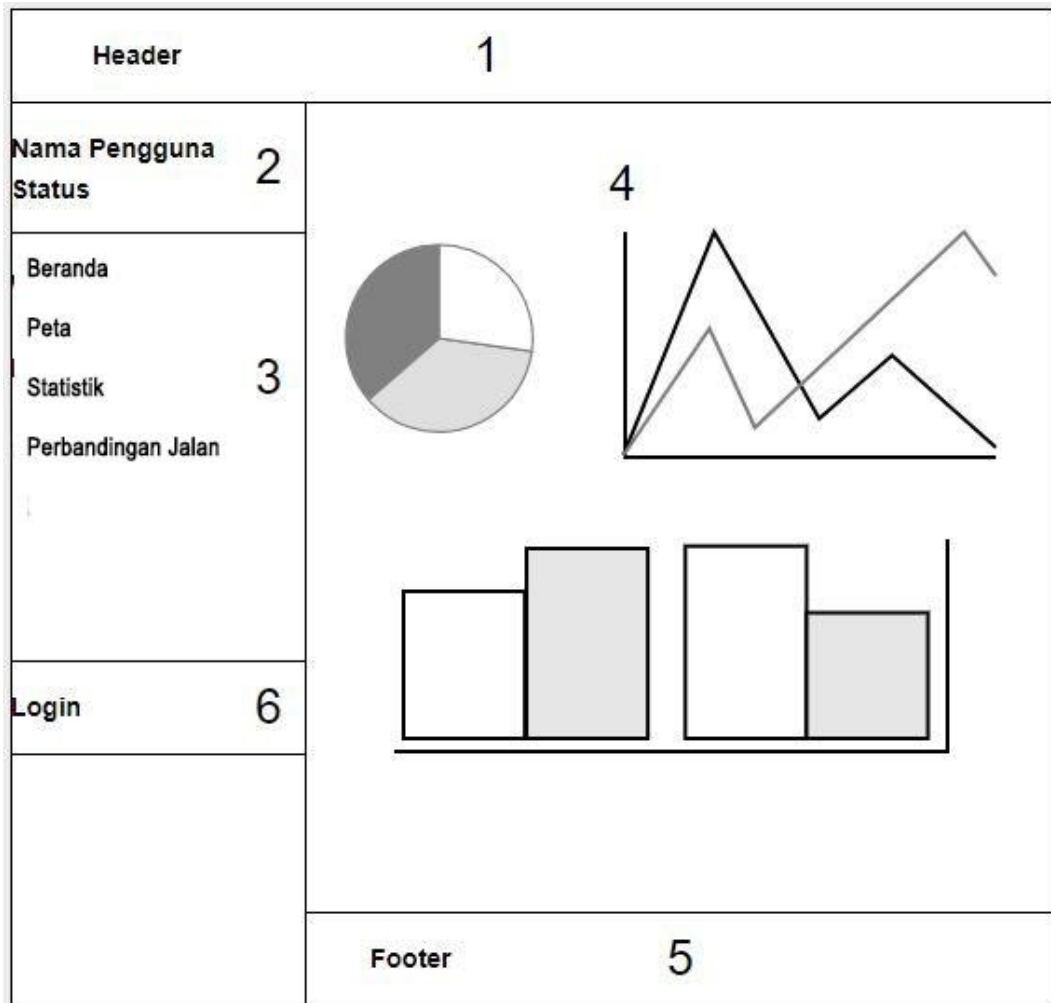


Gambar 4.27 Halaman Detail Jalan

Keterangan gambar 4.27 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi grafik perkembangan kecelakaan pada jalan yang dipilih.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol *login* untuk mengarahkan ke halaman *login*.
7. Nama jalan dan kategori jalan yang dipilih.

4.7.1.4 Halaman Statistik



Gambar 4.28 Halaman Statistik

Keterangan gambar 4.28 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi berbagai grafik kecelakaan.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol *login* untuk mengarahkan ke halaman *login*.

4.7.1.5 Halaman *Login*



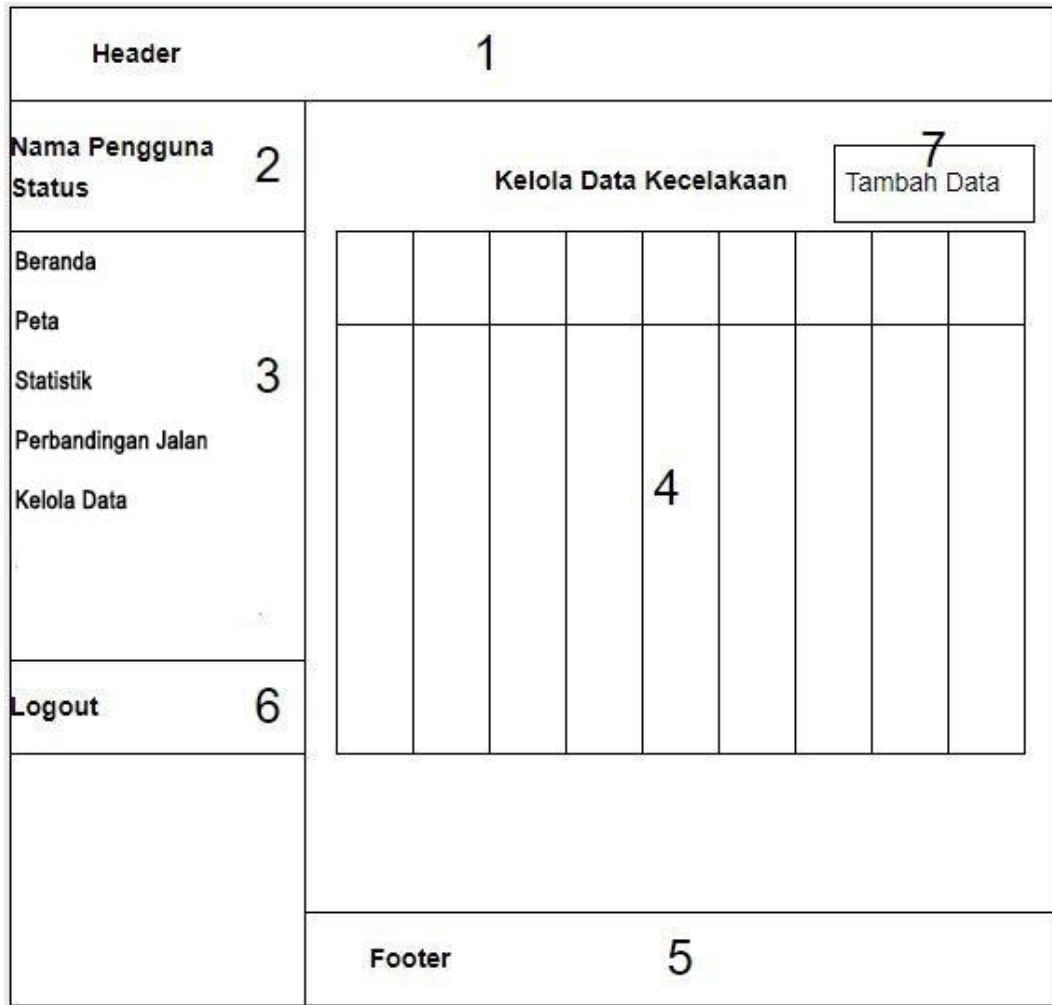
The image shows a login interface for an administrator. At the top center is a logo (1) depicting a landscape with mountains and a sun. Below the logo, the word "Admin" is displayed in a large font. Underneath "Admin" is the system name "Sistem Informasi Pemetaan Kecelakaan Kota Batu" (2). A message reads "Silahkan Login Untuk Mendapatkan Hak Akses Sistem". There are two input fields: "Username" (3) and "Password" (4). At the bottom left is a "Login" button (5).

Gambar 4.29 Halaman *Login*

Keterangan gambar 4.29 sebagai berikut:

1. Logo sistem.
2. Nama Sistem.
3. *Field* untuk memasukkan *username*.
4. *Field* untuk memasukkan *password*.
5. Tombol *login* untuk menjalankan fungsi *login*.

4.7.1.6 Halaman Kelola Data Kecelakaan

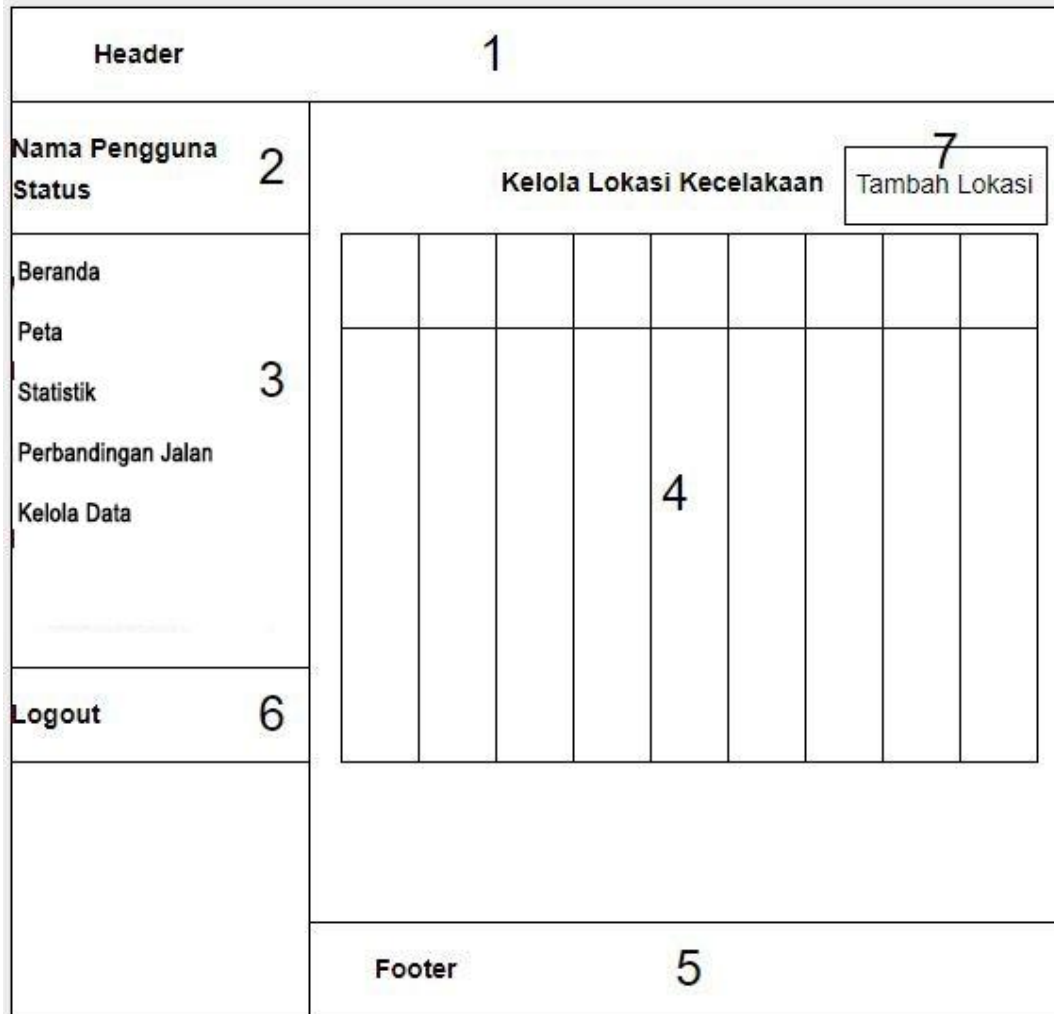


Gambar 4.30 Halaman Kelola Data

Keterangan gambar 4.30 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi semua data kecelakaan yang ditampilkan dalam bentuk tabel.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol *logout* untuk mengarahkan keluar dari hak akses sistem.
7. Tombol untuk mengarahkan ke halaman tambah data.

4.7.1.7 Halaman Kelola Lokasi Kecelakaan



Gambar 4.31 Halaman Kelola Lokasi

Keterangan gambar 4.31 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi semua data koordinat yang ditampilkan dalam bentuk tabel.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol *logout* untuk mengarahkan keluar dari hak akses sistem.
7. Tombol untuk mengarahkan ke halaman tambah koordinat lokasi.

4.7.1.8 Halaman Tambah Data Kecelakaan

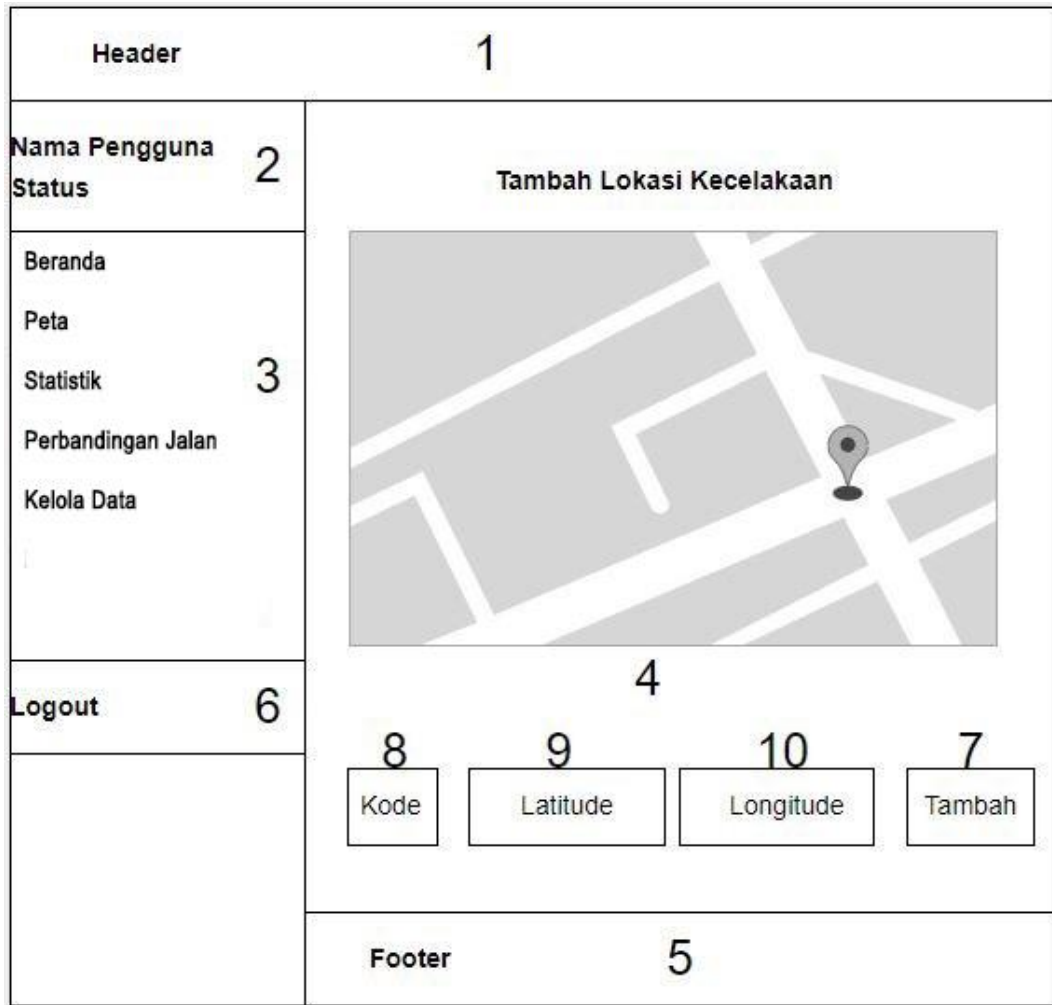
Header		1		
Nama Pengguna	2	Tambah Data Kecelakaan		
Status				
Beranda	3		Kode Titik	<input type="text"/>
Peta			Nama Jalan	<input type="text"/>
Statistik			Kecamatan	<input type="text"/>
Perbandingan Jalan		Total	<input type="text"/>	
Kelola Data		Meninggal	<input type="text"/>	
		Luka Berat	<input type="text"/>	
		Luka Ringan	<input type="text"/>	
		Waktu	<input type="text"/>	
Logout	6			
			7 Tambah	
Footer		5		

Gambar 4.32 Halaman Tambah Data

Keterangan gambar 4.32 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi beberapa *field* yang harus diisi untuk menambahkan data kecelakaan.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol *logout* untuk mengarahkan keluar dari hak akses sistem.
7. Tombol untuk menajalankan fungsi tambah data.

4.7.1.9 Halaman Tambah Lokasi Kecelakaan



Gambar 4.33 Halaman Tambah Lokasi

Keterangan gambar 4.33 sebagai berikut:

1. *Header* sistem informasi pemetaan kecelakaan.
2. Menampilkan nama pengguna dan status pengguna sistem.
3. *Side* menu yang berisi menu-menu yang terdapat pada sistem.
4. Konten yang berisi peta Kota Batu dan sekitarnya.
5. *Footer* sistem informasi pemetaan kecelakaan.
6. Tombol *logout* untuk mengarahkan keluar dari hak akses sistem.
7. Tombol untuk menjalankan fungsi tambah lokasi.
8. *Field* untuk memasukkan kode lokasi.
9. *Field* untuk memasukkan koordinat latitude
10. *Field* untuk memasukkan koordinat longitude