

**SISTEM INFORMASI PENGELOLAAN PELAPORAN DAN
PENANGANAN BENCANA ASAP BERBASIS
*CITIZEN SCIENCE***



Disusun Oleh:

N a m a : Bambang Irawan

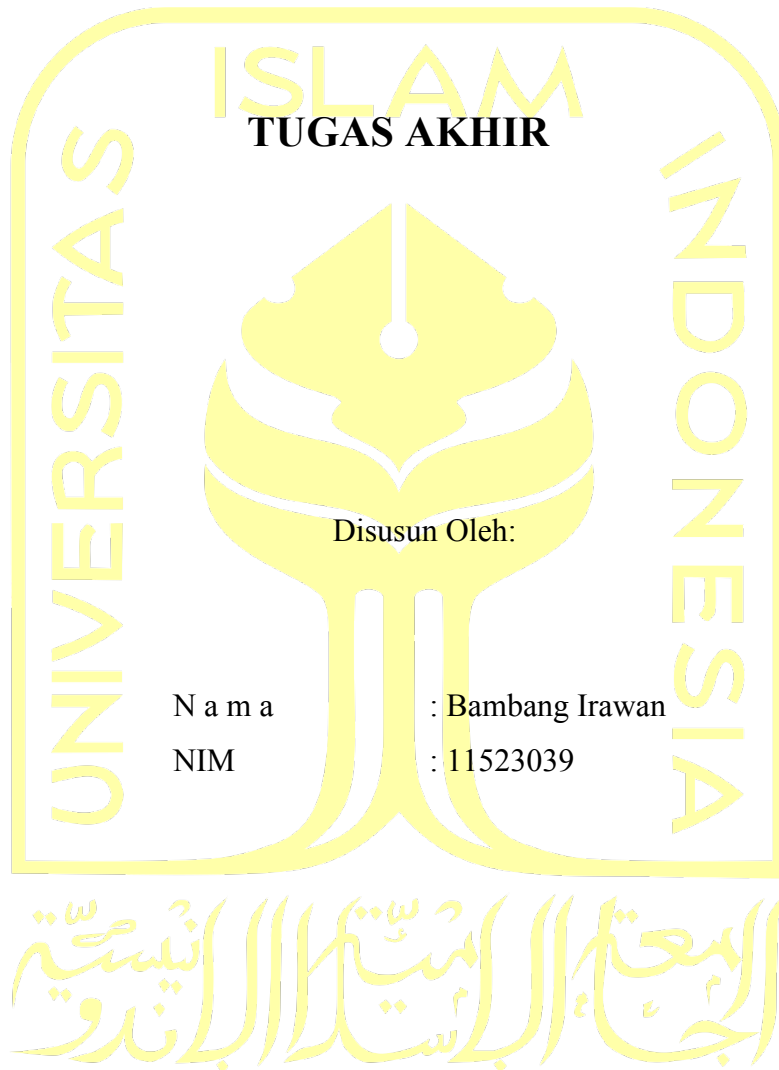
NIM : 11523039

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2018

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**SISTEM INFORMASI PENGELOLAAN PELAPORAN DAN
PENANGANAN BENCANA ASAP BERBASIS
*CITIZEN SCIENCE***



Yogyakarta, 1 Agustus 2018

Pembimbing,

(Kholid Haryono, S.T, M.Kom)

HALAMAN PENGESAHAN DOSEN PENGUJI

**SISTEM INFORMASI PENGELOLAAN PELAPORAN DAN
PENANGANAN BENCANA ASAP BERBASIS
*CITIZEN SCIENCE***

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk
memperoleh gelar Sarjana Teknik Informatika
di Fakultas Teknologi Industri Universitas Islam Indonesia
Yogyakarta, 1 Agustus 2018

Tim Penguji

Kholid Haryono, S.T, M.Kom. _____

Ketua

Rahadian Kurniawan, S.Kom., M.Kom. _____

Anggota 1

Galang P Mahardika, S.Kom., M.Kom. _____

Anggota 2

Mengetahui,

Ketua Jurusan Teknik Informatika
Fakultas Teknologi Industri
Universitas Islam Indonesia

(Hendrik, S.T., M.Eng.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Bambang Irawan

NIM : 11523039

Tugas akhir dengan judul:

SISTEM INFORMASI PENGELOLAAN PELAPORAN DAN PENANGANAN BENCANA ASAP BERBASIS *CITIZEN* *SCIENCE*

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 1 Agustus 2018

(Bambang Irawan)

HALAMAN PERSEMBAHAN

Alhamdulillah Robbil 'Alamin puji syukur atas segala nikmat dan karunia yang Allah SWT berikan kepada saya sehingga tugas akhir ini dapat terselesaikan dengan baik. Atas semua dukungan dan bantuan yang telah diberikan, saya persembahkan tugas akhir ini untuk, yang selalu bertanya **“Kapan Kuliah Selesai?”**

Terlambat lulus atau lulus tidak dengan waktu yang cepat bukanlah sebuah kejahatan dan juga bukan tergolong tindakan yang melanggar norma serta aturan yang berlaku, apalagi dianggapnya sebuah aib.

Setiap Mahasiswa, pada lubuk hatinya inginlah cepat lulus, namun percayalah bahwasannya setiap perjalanan Mahasiswa dalam perjuangan menyelesaikan pendidikannya berbeda-beda.

Alangkah kerdilnya jika kita menjadikan waktu kelulusan sebagai parameter dalam mengukur kemampuan seorang Mahasiswa dan menentukan kesuksesannya kelak di masa depan. Bukankah sebaik-baiknya skripsi adalah skripsi yang selesai? Baik itu selesai dengan waktu yang cepat atau memakan waktu yang cukup lama.

HALAMAN MOTO

“Sebaik-baiknya manusia adalah manusia yang bermanfaat bagi orang lain.”

(H.R. Ahmad, ath-Thabrani, ad-Daruqutni)

“Hai orang-orang yang beriman, mintalah pertolongan (kepada Allah) dengan sabar dan (mengerjakan) shalat, sesungguhnya Allah beserta orang-orang yang sabar.”

(Al-Baqarah : 153)

“Barangsiapa yang menapaki suatu jalan dalam rangka mencari ilmu maka Allah akan memudahkan baginya jalan ke Surga.

(H.R. Ibnu Majah & Abu Dawud)

“Maka sesungguhnya beserta kesulitan itu ada kemudahan. Sesungguhnya beserta kesulitan itu ada kemudahan.”

(QS. Al-Insyirah : 5-6)

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillah, penulis panjatkan kehadiran Allah SWT yang telah memberikan rahmat, hidayah, dan karunia-Nya, sehingga laporan Tugas Akhir dapat penulis selesaikan. Tak lupa shalawat dan salam kami haturkan kepada junjungan kita Nabi Muhammad SAW, yang telah membawa kita dari zaman jahiliyah menuju jaman terang benderang.

Tugas Akhir ini dibuat sebagai salah satu syarat yang harus dipenuhi untuk memperoleh gelar sarjana di Jurusan Teknik Informatika Universitas Islam Indonesia. Adapun Tugas Akhir penulis mengenai Sistem Informasi Pengelolaan Pelaporan dan Penanganan Bencana Asap Berbasis *Citizen Science*.

Pelaksanaan Tugas Akhir ini merupakan salah satu mata kuliah wajib dari jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia dan juga merupakan sarana bagi penulis untuk menambah wawasan serta pengalaman dalam menerapkan keilmuan yang telah diampu selama ini sesuai dengan yang diambil di bangku perkuliahan.

Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada:

1. Prof. Dr. Ir. Hari Purnomo, selaku Dekan Fakulras Teknologi Industri Universitas Islam Indonesia.
2. Hendrik, S.T., M.Eng, selaku Ketua Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.
3. Kholid Haryono, S.T., M.Kom., selaku Dosen Pembimbing Tugas Akhir penulis yang senantiasa membimbing dengan sabar, memberi saran dan solusi hingga penulis mampu menyelesaikan tugas akhir ini dengan sebaik-baiknya. Terima kasih banyak, Pak.
4. Orang tua dan keluarga penulis atas segala doa dan dukungan selama penulis melakukan tugas akhir. Terutama bagi kedua orang tua tercinta, Ayah Aiptu Jumari, Ibu Cicin Yuliani dan Kakak Rini Susanti, Amd.Keb., yang selalu memberi dukungan baik moril maupun materi untuk terus berjuang menyelesaikan kuliah tanpa pernah membuat anaknya merasa tertekan hingga pada akhirnya tugas akhir ini selesai.

5. Segenap keluarga besar teman-teman di Fakultas Teknologi Industri Universitas Islam Indonesia terutama dari Jurusan Teknik Informatika 2011 “DEFINE” yang telah memberikan bantuan dan dukungannya.
6. Teman-teman “Semesta Café” yang senantiasa selalu menemani penulis dalam menyelesaikan tugas akhir ini, membantu saran dan dukungan, terima kasih Ari Azhari, Syafiul Hadi, Beny Prasetya, Imam Rianto, Yanuardhi Mazid, Iman, Danu, Miftahul Khair, Yudha Pratama, Alfian Wiralodra, Cak Kholis, Aris, Yosi, Shafrina.
7. Semua pihak yang telah banyak membantu penulis dalam pelaksanaan Tugas Akhir yang tidak dapat disebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih belum sempurna, karena keterbatasan kemampuan dan pengalaman di lapangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi kesempurnaan Laporan Tugas Akhir ini. Akhir kata, penulis berharap agar laporan ini dapat bermanfaat bagi semua pihak.

Wassalamu’alaikum Warahmatullahi Wabarakatuh

Yogyakarta, 1 Agustus 2018

(Bambang Irawan)

SARI

Penulisan Tugas Akhir ini bertujuan untuk membantu pengelolaan pelaporan dan penanganan bencana asap di Indonesia. Adapun yang menjadi latar belakang tugas akhir ini karena Indonesia merupakan salah satu negara yang memiliki wilayah atau daerah-daerah yang rentan mengalami kejadian kebakaran hutan dan bencana asap. Meskipun pemerintah memiliki Badan Nasional Penanggulangan Bencana (BNPB) yang sudah tersebar di berbagai wilayah, namun kejadian kebakaran hutan dan bencana asap masih sering terjadi karena sulitnya mendapatkan informasi sesuai yang di butuhkan oleh pihak BNPB dan para relawan.

Informasi yang belum terpusat menyebabkan sering terjadinya keterlambatan dalam melakukan penanganan, baik penanganan dalam bentuk pemadaman wilayah yang mengalami kebakaran hutan ataupun penanganan terhadap dampak dari bencana kabut asap yang menyebabkan jatuhnya banyak korban dan kerugian materi oleh pemerintah dari sektor riil dan non riil.

Solusi atas uraian masalah dan isu dari bencana asap maka penulis mengusulkan sebuah sistem informasi guna membantu menyelesaikan masalah tersebut dan meningkatkan efektivitas dalam penanganan kebakaran hutan dan bencana asap. Sistem informasi ini merupakan suatu sistem informasi yang bersifat keruangan dan membantu memberikan sarana pelaporan dan menyajikan informasi mengenai kejadian bencana asap. Sistem informasi ini disertai data pendukung yang berguna untuk menyajikan informasi yang dibutuhkan untuk keperluan proses penanganan bencana. Informasi tersebut dihasilkan dari pengolahan data-data kejadian oleh citizen science mengenai kejadian bencana asap yang terjadi. Sebagai bentuk penyelesaiannya maka terbentuk sebuah prototype Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis Citizen Science hasil dari pengembangan sebuah platform bernama Ushahidi. Prototype dari sistem informasi ini berupa web aplikasi dan aplikasi smarthphone berbasis android.

Kata kunci: metode, prototype, bencana asap, sistem pelaporan, citizen science.

GLOSARIUM

Glosarium memuat daftar kata tertentu yang digunakan dalam laporan dan membutuhkan penjelasan, misalnya kata serapan yang belum lazim digunakan. Contoh penulisannya seperti di bawah ini:

<i>Citizen Science</i>	proses untuk mengubah berkas kode program dengan berkas lain yang terkait menjadi berkas yang siap untuk dieksekusi oleh sistem operasi secara langsung.
<i>Netizen</i>	gabungan dari kata internet dan <i>citizen</i> atau pengguna internet aktif.
<i>Graphic</i>	grafis atau gambar.
<i>Workflow</i>	alir kerja adalah otomatisasi proses bisnis yang meliputi perpindahan dokumen melalui suatu prosedur kerja.
<i>Interface</i>	sebuah antarmuka, penghubung.
<i>Prototype</i>	Purwarupa (bahasa Inggris: <i>prototype</i>) atau arketipe adalah bentuk awal (contoh).
<i>Waterfall</i>	metode pengembangan perangkat lunak.
<i>Citizen science</i>	relawan peneliti.
<i>Developer</i>	programmer yang sudah terlatih, mereka tidak hanya menyelesaikan masalah, namun juga membangun suatu produk..
<i>Script</i>	bahasa yang digunakan untuk menerjemahkan setiap perintah dalam situs yang pada saat di akses. Sebuah skrip pemrograman adalah kode yang dapat meningkatkan fungsionalitas dari sebuah situs web.
<i>Platform</i>	istilah dalam teknologi informasi mengenai sebuah kerangka perangkat lunak
<i>Open-Source</i>	sistem terbuka dalam <i>software</i> , sistem pengembangan yang tidak dikoordinasi oleh suatu individu / lembaga pusat dan tersedia bebas.
<i>Crowd</i>	orang banyak.
<i>Blueprint</i>	cetak biru atau kerangka kerja terperinci (arsitektur) sebagai landasan dalam pembuatan kebijakan yang meliputi penetapan tujuan.
<i>Mobile</i>	sebuah antarmuka, penghubung.
<i>Development</i>	sebuah antarmuka, penghubung.
<i>User</i>	pengguna (yang menggunakan sebuah sistem).

<i>Concurrency</i>	control akan mekanisme sebuah sistem ketika terjadi akses atau transaksi yang bersamaan agar tidak saling mengganggu transaksi yang sedang berjalan.
<i>Element</i>	bagian-bagian dasar yang mendasari sesuatu.
<i>Mashup</i>	penggabungan dua hal atau lebih.
<i>Waterfall</i>	metode pengembangan perangkat lunak.
<i>Diagram Activity</i>	diagram atau aktifitas yang berjalan pada sistem.
<i>Back-End</i>	halaman pada belakang sistem yang mengatur semua informasi pada situs halaman depan yang terlihat oleh Pengguna.
API	sekumpulan perintah, fungsi, dan protokol yang dapat digunakan oleh programmer saat membangun perangkat lunak.
<i>Front-End</i>	halaman atau bagian depan yang dilihat oleh Pengguna.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI.....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI	ix
GLOSARIUM.....	x
DAFTAR ISI.....	xii
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR	xvi
BAB I PENDAHULUAN.....	18
1.1 Latar Belakang.....	18
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Metode Penelitian.....	3
1.7 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI.....	6
2.1 Bencana Asap	6
2.2 <i>Citizen Science</i>	7
2.3 Ushahidi.....	7
2.4 OpenStreetMap.....	8
2.5 <i>Unified Modelling Language</i>	8
2.5.1 <i>Use Case Diagram</i>	9
BAB III ANALISIS DAN PERANCANGAN	10
3.1 <i>Planning</i>	11
3.1.1 Pemilihan Metode <i>Prototype</i>	11
3.1.1 Pemilihan Metode Pengembangan.....	13

3.2	Analisis	15
3.2.1	Analisis Kebutuhan Proses	15
3.2.2	Analisis Kebutuhan Perangkat Keras	16
3.2.3	Analisis Kebutuhan Perangkat Lunak	16
3.3	Perancangan Desain Sistem	17
3.3.1	Perancangan <i>Undefined Model</i>	18
3.3.2	Perancangan <i>Activity Diagram</i>	19
3.3.3	Perancangan Struktur Basisdata	27
3.3.4	Perancangan Relasi Tabel Basisdata Web Aplikasi	50
3.3.5	Perancangan Antar Muka	50
BAB IV IMPLEMENTASI DAN PENGUJIAN		60
4.1	Implementasi	60
4.1.1	Implementasi <i>Web-Aplikasi</i>	61
4.1.2	Implementasi Aplikasi Android	66
4.2	Pengujian	68
4.2.1	Pengujian Validitas	68
4.2.2	Pengujian Reliabilitas	69
4.2.3	Hasil Deskripsi Efektivitas Sistem Oleh Pengguna	69
BAB V KESIMPULAN DAN SARAN		74
5.1	Kesimpulan	74
5.2	Saran	74
DAFTAR PUSTAKA		75
LAMPIRAN		77

DAFTAR TABEL

Tabel 3.1 Perbandingan Metode	12
Tabel 3.2 Tabel Kebutuhan Proses	15
Tabel 3.3 Tabel Kebutuhan Perangkat Keras	16
Tabel 3.4 Kebutuhan Perangkat Lunak.....	16
Tabel 3.5 Tabel actions	28
Tabel 3.6 Tabel action_log	29
Tabel 3.7 Tabel alert	29
Tabel 3.8 Tabel alert_category	29
Tabel 3.9 Tabel alert_sent.....	30
Tabel 3.10 Tabel api_banned.....	30
Tabel 3.11 Tabel api_log	30
Tabel 3.12 Tabel api_setting.....	31
Tabel 3.13 Tabel badge.....	31
Tabel 3.14 Tabel badge_user	31
Tabel 3.15 Tabel category.....	32
Tabel 3.16 Tabel city	32
Tabel 3.17 Tabel comment	32
Tabel 3.18 Tabel cluster.....	33
Tabel 3.19 Tabel country	34
Tabel 3.20 Tabel extrenal_app.....	34
Tabel 3.21 Tabel form.....	34
Tabel 3.22 Tabel form_field	35
Tabel 3.23 Tabel field_option.....	35
Tabel 3.24 Tabel form_response	36
Tabel 3.25 Tabel geometry	36
Tabel 3.26 Tabel incident	36
Tabel 3.27 Tabel incident_category.....	37
Tabel 3.28 Tabel incident_lang.....	37
Tabel 3.29 Tabel incident_lang.....	38
Tabel 3.30 Tabel incident_person.....	38

Tabel 3.31 Tabel layer	38
Tabel 3.32 Tabel layer	39
Tabel 3.33 Tabel maintenance	39
Tabel 3.34 Tabel media.....	39
Tabel 3.35 Tabel message.....	40
Tabel 3.36 Tabel permissions	40
Tabel 3.37 Tabel permissions_roles	41
Tabel 3.38 Tabel plugins.....	41
Tabel 3.39 Tabel private_message.....	41
Tabel 3.40 Tabel rating	42
Tabel 3.41 Tabel reporter.....	42
Tabel 3.42 Tabel roles.....	43
Tabel 3.43 Tabel roles_users	43
Tabel 3.44 Tabel scheduler	43
Tabel 3.45 Tabel scheduler_log.....	44
Tabel 3.46 Tabel service	44
Tabel 3.47 Tabel sessions	44
Tabel 3.48 Tabel settings	45
Tabel 3.49 Tabel sharing.....	45
Tabel 3.50 Tabel sharing_incident.....	45
Tabel 3.51 Tabel sharing_log	46
Tabel 3.52 Tabel smssync_message	46
Tabel 3.53 Tabel smssync_settings.....	46
Tabel 3.54 Tabel socialmedia_asset	47
Tabel 3.55 Tabel socialmedia_keywords.....	47
Tabel 3.56 Tabel socialmedia_messages_data	47
Tabel 3.57 Tabel socialmedia_settings	48
Tabel 3.58 Tabel users	48
Tabel 3.59 Tabel user_tokens	48
Tabel 3.60 Tabel verified.....	49
Tabel 4.1 Hasil Uji Validitas Instrumen	69
Tabel 4.2 Nilai Persepsi	69
Tabel 4.3 Distribusi Frekuensi Skor Persepsi Pengguna Terhadap	70
Tabel 4.4 Jawab Responden Terhadap.....	71

Tabel 4.5 Tabel Responden.....	72
Tabel 4.6 Hasil Kuesioner Pengukuran Efektivitas Sistem Informasi.....	73

DAFTAR GAMBAR

Gambar 3.1 Metode <i>Prototype</i>	12
Gambar 3.2 Skema Kerja Sistem.....	17
Gambar 3.3 Perancangan <i>Use Case Diagram</i>	18
Gambar 3.4 <i>Activity Diagram Login</i>	19
Gambar 3.5 <i>Activity Diagram Pendaftaran</i>	20
Gambar 3.6 <i>Activity Diagram Laporkan</i>	20
Gambar 3.7 <i>Activity Diagram Lihat Informasi Kejadia</i>	21
Gambar 3.8 <i>Activity Diagram Pemberitahuan</i>	22
Gambar 3.9 <i>Activity Download Laporan</i>	22
Gambar 3.10 <i>Activity Diagram Hubungi Kami</i>	23
Gambar 3.11 <i>Activity Diagram Manajemen Pengguna</i>	24
Gambar 3.12 <i>Activity Diagram Verifikasi Laporan</i>	24
Gambar 3.13 <i>Activity Diagram Verifikasi Komentar</i>	25
Gambar 3.14 <i>Activity Diagram Kelola Pesan Masuk</i>	26
Gambar 3.15 <i>Activity Diagram Hapus Semua Laporan</i>	26
Gambar 3.16 <i>Activity Diagram Hapus Lupa Password</i>	27
Gambar 3.17 Halaman Beranda.....	50
Gambar 3.18 Halaman Laporkan.....	51
Gambar 3.19 Halaman Daftar Informasi Kejadian.....	51
Gambar 3.20 Halaman Hubungi Kami.....	52
Gambar 3.21 Halaman Pemberitahuan.....	52
Gambar 3.22 Halaman Dashboard Admin.....	53
Gambar 3.23 Halaman Laporan Admin.....	53
Gambar 3.24 Halaman Input dan Edit Laporan Admin.....	54
Gambar 3.25 Halaman Hapus Semua Laporan.....	54
Gambar 3.26 Halaman Pesan Admin.....	55
Gambar 3.27 Halaman Pesan Admin.....	55
Gambar 3.28 Halaman Manajemen Pengguna.....	56
Gambar 3.29 Halaman Awal Aplikasi Android.....	57

Gambar 3.30 Halaman Utama Aplikasi Android.....	57
Gambar 3.31 Halaman Input Laporan	58
Gambar 3.32 Rancangan Halaman Notifikasi Sukses Input Laporan.....	58
Gambar 3.33 Halaman Lihat Daftar Laporan	59
Gambar 3.34 Halaman Lihat Daftar Laporan	59
Gambar 4.1 Impelementasi Halaman Beranda	61
Gambar 4.2 Impelementasi Halaman Laporkan	61
Gambar 4.3 Impelementasi Halaman Informasi Kejadian.....	62
Gambar 4.4 Impelementasi Halaman Download Laporan.....	62
Gambar 4.5 Impelementasi Halaman Hubungi Kami.....	63
Gambar 4.6 Impelementasi Halaman Dashboard <i>Admin</i>	63
Gambar 4.7 Impelementasi Halaman Laporan Masuk <i>Admin</i>	64
Gambar 4.8 Impelementasi Halaman Pesan Masuk <i>Admin</i>	64
Gambar 4.9 Impelementasi Halaman Manajemen Pengguna	65
Gambar 4.10 Impelementasi Halaman Manajemen <i>Web-Aplikasi</i>	65
Gambar 4.11 Impelementasi Halaman <i>Loading</i> Aplikasi Android	66
Gambar 4.12 Impelementasi Halaman Utama Aplikasi Android	67
Gambar 4.13 Impelementasi Halaman Input Laporan Aplikasi Android	67
Gambar 4.14 Impelementasi Halaman Daftar Laporan Aplikasi Android	68

BAB I

PENDAHULUAN

1.1 Latar Belakang

Indonesia memiliki sejumlah wilayah yang rentan mengalami kebakaran hutan dan bencana asap. Menurut data yang dirilis oleh Badan Nasional Penanggulangan Bencana (BNPB) setidaknya ada enam provinsi di Indonesia yang rawan mengalami bencana kebakaran hutan, lahan, dan kabut asap. Provinsi itu antara lain; Provinsi Riau, Jambi, Kalimantan Barat, Kalimantan Tengah dan Kalimantan Selatan.

Kebakaran hutan, lahan dan bencana kabut asap yang terjadi di sejumlah wilayah di Indonesia memberikan ancaman serius dan harus menelan kerugian baik secara materi maupun non-materi. Menurut data yang dihimpun oleh kementerian sosial menyebutkan bencana kabut asap di wilayah Sumatera dan Kalimantan pada periode tahun 2015 menelan korban kurang lebih 19 orang. Sementara menurut Dinas Kesehatan Provinsi Riau, dalam periode 29 Juni- 27 September 2015, sebanyak 44.871 orang terkena risiko asap. Dari angka itu, sebanyak 37.396 orang lainnya menderita ISPA, 656 orang menderita pneumonia, 1.702 orang terkena asma, 2.207 orang menderita penyakit mata, dan penyakit kulit sebanyak 2.911 orang (Maharani, 2015).

Sedangkan kerugian secara materi dari data yang dihitung oleh Bank Dunia, total kerugian Indonesia akibat bencana kabut asap pada tahun 2015 telah melampaui 16 miliar dolar AS, angka ini jauh lebih besar dari kerugian yang dialami Indonesia saat dilanda tsunami di Aceh pada tahun 2004 silam. Kerugian itu dihimpun dari berbagai sektor riil dan non riil, diantaranya; sektor pertanian, kehutanan, transportasi, perdagangan, industri, pariwisata dan sektor-sektor lainnya seperti kerusakan lingkungan karena punahnya keanekaragaman hayati yang diperkirakan bernilai 295 juta dolar AS (Isyana, 2015).

Dampak dari kerugian secara materi maupun non-materi dalam skala besar yang dialami Indonesia membuat pemerintah melakukan evaluasi besar terhadap penanggulangan bencana kebakaran hutan dan kabut asap. Pemerintah pusat dan daerah bersama *stakeholder* yang terlibat melakukan upaya pengawasan ekstra untuk beberapa wilayah yang rentan mengalami bencana kebakaran hutan dan kabut asap. Sejauh ini pemerintah hanya mengandalkan peraturan gubernur dan daerah dalam upaya mencegah terjadinya kebakaran hutan dan bencana kabut asap, namun ketika bencana sudah terjadi penanggulangan bencana asap secara

garis besar terpaku pada hasil laporan para relawan dengan jumlah yang terbatas, masyarakat seakan tidak memiliki media untuk memberikan informasi agar penanggulangan dan penanganan para korban oleh pihak BNPB ataupun relawan dapat berjalan secara efektif, cepat dan tepat.

BNPB selama ini hanya mengandalkan data dan informasi yang dikumpulkan oleh para petugas BNPB atau relawan ketika telah terjun langsung ke lokasi kejadian bencana untuk memberikan tindakan lebih lanjut secara efektif. Proses seperti itu tentu dirasa semakin kurang efektif dan memakan waktu, sedangkan penanggulangan dan penanganan yang dibutuhkan para korban harus segera dilakukan. Untuk itu diperlukan sistem informasi yang mampu meningkatkan efektifitas pengelolaan pelaporan dalam penanganan bencana asap.

Penelitian ini mengusulkan pembangunan sistem informasi pengelolaan dan pelaporan *citizen science* dalam penanganan bencana asap. Sistem informasi demikian diperlukan agar relawan dan pihak-pihak yang terlibat dapat dengan cepat, tepat dan efektif untuk melakukan penyebaran dalam menanggulangi bencana dan memberikan penanganan secara tepat yang dibutuhkan para korban bencana asap.

Citizen Science atau dalam pengertian lain adalah ilmu pengetahuan masyarakat dalam penggalan ilmu pengetahuan dan teknologi. Sedangkan menurut The Oxford English Dictionary memasukan pengertian *citizen science* adalah karya ilmiah yang dilakukan oleh anggota masyarakat umum, umumnya bekerjasama dengan ilmuwan profesional atau dibawah arahan ilmuwan profesional maupun lembaga-lembaga ilmiah. Pelaku *citizen science* dikenal sebagai *citizen scientist* atau ilmuwan warga (Simpson, 2013). Fokus penelitian adalah membuat *tools* pusat informasi pengelolaan dan pelaporan bencana asap dengan melibatkan masyarakat khususnya relawan dalam pengambilan data dan sumber informasi peneliti.

1.2 Rumusan Masalah

Mengembangkan (*Development*) sistem informasi berbasis *web* aplikasi dan aplikasi *mobile* berbasis android yang dapat digunakan sebagai sarana penyebaran dan pengumpulan informasi terkait dengan penanganan bencana asap dan kebakaran hutan, maka dalam menyelesaikan masalah berikut didapat kesimpulan sebagai berikut:

1. Bagaimana mengukur efektivitas pengelolaan pelaporan dan penanganan bencana asap dengan menggunakan sistem informasi?

2. Bagaimana menangani bencana asap secara efektif dan tepat sasaran dengan teknik sistem informasi terpusat?

1.3 Batasan Masalah

Batasan masalah pada sistem informasi ini sebagai berikut:

1. Studi kasus hanya meliputi wilayah Indonesia.
2. Sistem tidak menangani pengelolaan distribusi bantuan tetapi melakukan pengelolaan informasi bantuan

1.4 Tujuan Penelitian

Tujuan penelitian adalah merancang dan membuat sebuah sistem informasi pengelolaan pelaporan dan penanganan bencana asap berbasis *citizence science* atau dalam pengertian ilmu pengetahuan warga dan bekerja sama dengan ilmuwan profesional ataupun amatir yang mampu membantu masyarakat dalam penanganan bencana asap dan permintaan bantuan serta kebutuhan lainnya yang bersifat *mobile*, cepat dan tanggap.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini antara lain:

- a. Hasil uji sistem informasi pengelolaan pelaporan dan penanganan bencana asap dapat diusulkan kepada Badan Nasional Penanggulangan Bencana (BNPB) untuk dimanfaatkan, dikembangkan lebih jauh dan menjadi *prototype* sebagai masukan dalam pemanfaatan sistem informasi tersebut.
- b. *Prototype* sistem informasi akan diberikan kepada relawan TIK sebagai organisasi Lembaga Swadaya Masyarakat (LSM) yang bergerak dalam bidang TIK agar dapat dimanfaatkan dalam gerak dan aktivitasnya terhadap pengelolaan pelaporan dan penanganan bencana asap.
- c. Menambah hasanah wawasan dalam bidang pemanfaatan IT untuk penanggulangan berbagai bencana, khususnya bencana kebakaran hutan dan bencana asap.

1.6 Metode Penelitian

Dalam pengerjaan penelitian metode penelitian yang akan digunakan adalah metode *Prototype*. Metode ini meliputi beberapa tahapan sebagai berikut:

1. *Planning*

Tahapan perencanaan langkah – langkah penilitan, pemilihan metode, dan pemilihan penggunaan alat dalam pembangunan sistem informasi.

2. Analisis

Analisis adalah aktivitas pengumpulan data, menyusun kebutuhan sistem, menerjemahkan dan mengevaluasi data sebelum sistem di desain.

3. Perancangan

Pembuatan perancangan desain sistem dan perancangan purwarupa sistem informasi yang akan dibangun.

4. Implementasi

Pada tahap ini memberikan *prototype* dari sistem yang telah dirancang sebelumnya. *Prototype* dari *sistem* informasi pengelolaan pelaporan dan penanganan bencana asap berbasis *citizen science* berbentuk keluaran berupa *web* aplikasi dan aplikasi android.

5. Pengujian

Pengujian terhadap *prototype* sistem informasi pengelolaan pelaporan dan penanganan bencana asap berbasis *citizen science* dengan menggunakan metode pengujian efektivitas melalui kuesioner terhadap beberapa responden untuk mengukur tingkat efektivitas sistem informasi dan kinerja keseluruhan sistem informasi.

1.7 Sistematika Penulisan

Sistematika penulisan penelitian ini disusun untuk memberikan gambaran umum tentang penelitian yang dijalankan. Sistematika penelitian ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab pendahuluan merupakan bab yang menjelaskan tentang hal paling dasarb mengenai penelitian yang dilakukan secara umum. Isi dari bab pendahuluan meliputi latar belakang masalah, rumusan masalah, batasan masalah, tujuan serta manfaat dari penelitian, dan metodologi yang akan diterapkan.

BAB II LANDASAN TEORI

Landasan teori membahas mengenai dasar teori yang digunakan pada penelitian atau pembuatan sistem. Teori tersebut dapat membantu penentuan konsep pada sistem yang akan dibangun.

BAB III ANALISIS DAN PERANCANGAN

Bab metodologi membahas analisis serta perancangan sistem. Analisis sistem menjabarkan tentang identifikasi masalah, gambaran umum sistem serta analisis kebutuhan yang diperlukan sepanjang penelitian. Perancangan sistem menjabarkan tentang perancangan sistem yang dibutuhkan seperti *Use Case Diagram*, *Diagram Activity* , perancangan basisdata serta perancangan antarmuka.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Hasil penelitian membahas mengenai implementasi atau penerapan yang dilakukan dari rancangan yang telah dibuat sebelumnya. Implementasi didapat dari perancangan sesuai dengan kebutuhan sistem yang dibuat

BAB V KESIMPULAN DAN SARAN

Kesimpulan berisi tentang ringkasan dari proses - proses sistem yang telah dilakukan serta mengemukakan saran-saran selama pengembangan sistem.

BAB II

LANDASAN TEORI

2.1 Bencana Asap

Bencana asap adalah bencana yang berbentuk kabut asap atau yang lebih dikenal dengan istilah kabut asap adalah kumpulan asap dan kabut yang bercampur menjadi satu kesatuan yang mencemari udara oleh asap. Bencana asap terjadi akibat kebakaran hutan atau lahan. Kebakaran yang terjadi bisa karena proses alamiah seperti kombinasi antara kebakaran hutan dan musim kemarau yang panjang dan juga bisa terjadi karena ulah tangan manusia seperti pembakaran yang dengan sengaja dilakukan untuk pembukaan lahan peladangan.

Bencana kabut asap sangat berbahaya bagi kesehatan karena dapat mengganggu sistem pernafasan dan memicu penyakit jantung kronis. Selain itu bencana kabut asap juga sangat berbahaya bagi dunia lalu lintas yang bisa menyebabkan kecelakaan baik pada lalu lintas darat, laut atau sungai dan udara, karena kabut asap membuat jarak pandang hanya beberapa meter saja. Semua pihak perlu diingatkan tentang betapa bahayanya kabut asap bagi kegiatan transportasi udara, darat, bahkan laut. Kabut asap dapat menyebabkan terbatasnya jarak pandang pilot, nakhoda, masinis, supir (pengendara mobil), atau pengendara sepeda motor, sehingga sangat berpotensi menimbulkan kecelakaan transportasi, baik di udara, darat, sungai, maupun laut. Kecelakaan bukan hanya dipicu oleh tabrakan, tapi juga oleh kemungkinan tidak terkontrolnya (*out of control*) armada atau alat transportasi (Biro Komunikasi Dan Informasi Publik, 2012).

Selain dampak buruk yang terjadi pada dunia kesehatan maupun lalu lintas, bencana kabut asap juga memberikan dampak buruk pada kegiatan keseharian masyarakat seperti terganggunya kegiatan bekerja dan kegiatan belajar-mengajar di sekolah.

Banyak upaya yang sudah dilakukan pemerintah untuk mencegah dan menanggulangi terjadinya bencana kabut asap dan dampak buruk yang ditimbulkannya, seperti membuat hujan buatan saat terjadi kebakaran hutan. Namun upaya tersebut masih dirasa kurang. Oleh karena itu, bencana kabut asap ini merupakan bencana yang sangat serius untuk diperhatikan dalam upaya penanggulangannya, sehingga kita semua berperan penuh dalam membantu menanggulangnya (Wardana, 2015).

2.2 Citizen Science

Citizen science atau dalam pengertian lain adalah ilmu pengetahuan masyarakat dalam penggalian ilmu pengetahuan dan teknologi. The Oxford English Dictionary memasukan pengertian *citizen science* pada bulan Juni 2014 yang menjelaskan *citizen science* sebagai karya ilmiah yang dilakukan oleh anggota masyarakat umum, umumnya bekerjasama dengan ilmuwan profesional atau dibawah arahan ilmuwan profesional maupun lembaga-lembaga ilmiah.

Pelaku *citizen science* dikenal sebagai *citizen scientist* atau ilmuwan warga. *Citizen science* juga dapat diartikan sebagai berikut:

- Ilmuwan yang karyanya ditandai dengan tanggung jawab untuk melayani kepentingan terbaik dari warga masyarakat luas;
- Anggota masyarakat umum yang terlibat dalam pengerjaan karya ilmiah yang dapat bekerja sendiri maupun bekerjasama dengan ilmuwan profesional dan lembaga-lembaga ilmiah;
- Ilmuwan amatir.

Perkembangan *citizen science* di Eropa ditandai dengan terbitnya *Green Paper on Citizen Science: Citizen Science For Europa* yang memberi definisi tentang *citizen science*. Paper ini mendefinisikan *citizen science* sebagai; ”keterlibatan warga negara / masyarakat umum dalam kegiatan penelitian ilmiah”, ketika warga secara aktif berkontribusi pada ilmu pengetahuan baik dengan upaya intelektualnya maupun lingkungan sekitar atau dengan sumberdaya yang dimiliki warga itu sendiri (Scientize Project, 2014).

Kemampuan *citizen science* yaitu kemampuan warga negara memberikan data yang layak. *Citizen science* memberi peluang penyelesaian masalah yang baru serta menciptakan budaya ilmiah yang baru dan komperhensif. Keuntungan bagi relawan / warga yang terlibat dalam *citizen science* adalah memperoleh pengetahuan, pengalaman dan keterampilan yang berkaitan dengan proses karya ilmiah dengan cara yang menyenangkan. *Citizen science* membuka luas pergaulan dan jaringan dengan para relawan, jaringan antar bidang dalam berbagai disiplin ilmu maupun pekerjaan (Diolah Dari Berbagai Sumber).

2.3 Ushahidi

Ushahidi merupakan nama atau *brand* yang dari sejenis *platform web* aplikasi yang digunakan untuk mengumpulkan banyak data, presentasi atau fungsi dari dua sumber atau

lebih atau disebut juga dengan *mashup* untuk menciptakan layanan jasa baru dalam pengumpulan data.

Ushahidi diciptakan pada akhir tahun 2007 sebagai respon dari huru hara yang terjadi setelah pemilu di Kenya. Tidak lama setelah kejadian tersebut Ushahidi dikenal sebagai perusahaan yang menciptakan sebuah *platform* sumber terbuka atau dikenal dengan istilah *open-sources* (Prasetya, 2010).

Hasil pengamatan penulis tentang penggunaan Ushahidi akhir-akhir ini lebih banyak diimplementasikan sebagai sebuah *web* aplikasi pelaporan tentang sesuatu dalam hal pelaporan masalah publik atau bencana, baik itu terjadi dalam skala yang kecil seperti dalam ruang lingkup sebuah kota atau provinsi, atau dalam skala besar seperti dalam ruang lingkup suatu negara.

2.4 OpenStreetMap

OpenStreetMap (OMS) adalah proyeksi kolaborasi pembuatan peta dunia yang bebas disunting. Dua penggerak utama dibalik perintisan dan pembuatan OMS adalah karena terbatasnya penggunaan atau ketersediaan informasi peta di sebagian besar wilayah dunia serta maraknya perangkat navigasi satelit portabel yang terjangkau. Data yang didapat OMS dihasilkan dari perangkat *Global Positioning System* (GPS), survei mandiri, fotografi udara dan sumber lainnya (Openstreep Map Indonesia, 2014).

2.5 Unified Modelling Language

Unidied modelling language (UML) adalah suatu metode permodelan secara *visual* untuk sarana perancangan sistem berorientasi objek, atau pada definisi lain dari UML adalah sebagai suatu bahasa yang sudah menjadi standar visualisasi, perancangan dan juga pendokumentasian sistem *software*. Saat ini UML sudah menjadi bahasa standar dalam penulisan *blue print software* (Sora, 2015).

UML memiliki banyak jenis pemodelan sesuai dengan kebutuhan penggunaannya seperti *Use case diagram*, *activity diagram*, *squence diagram*, *class diagram* dan lainnya. Dalam penelitian ini penulis menggunakan dua jenis pemodelan UML, pemodelan dari UML yang digunakan adalah pemodelan *use case diagram* dan *activity diagram* (Sora, 2015).

2.5.1 *Use Case Diagram*

Use case diagram adalah gambaran *graphical* dari beberapa atau semua *Actor*, *use case* dan interaksi diantaranya memperkenalkan suatu sistem. *Use case diagram* tidak menjelaskan secara detail tentang penggunaan *use case*, tetapi hanya memberi gambaran singkat hubungan antara *usecase*, *Actor* dan sistem. Didalam *use case* ini kita akan mengetahui fungsi – fungsi apa saja yang berada pada sistem yang dibuat (Rahmatullah, 2016).

BAB III

ANALISIS DAN PERANCANGAN

Pada tahapan analisis dan perancangan, tahapan pertama yang dilakukan penulis adalah melakukan pengumpulan data. Tahapan ini nantinya akan menjadi acuan dalam proses pengerjaan tahapan selanjutnya dan menjawab rumusan masalah. Metode pengumpulan data yang digunakan yaitu metode studi literatur atau bisa disebut juga dengan kajian pustaka, yaitu teknik pengumpulan data melalui teks-teks tertulis maupun *soft-copy edition*, seperti buku, *ebook*, artikel-artikel, surat kabar, buletin, jurnal, laporan atau arsip-arsip organisasi, makalah, publikasi, dan lain sebagainya. Bahan pustaka dalam bentuk *soft-copy edition* secara *general* diperoleh penulis dari hasil pencarian di internet dan khususnya didapat dari media daring yang mengumpulkan sumber – sumber kajian bersifat kredibel.

Pemilihan metode studi literatur atau kajian pustaka dalam proses pengumpulan data dikarenakan sifat utama dari data yang didapat tidak terbatas pada ruang dan waktu sehingga memberi peluang kepada penulis untuk dapat mengetahui hal – hal yang terjadi pada waktu silam. Meskipun begitu, pengumpulan data dengan metode studi literatur ini sangat tergantung pada penulisan hasil laporan atau fenomena yang terjadi diungkapkan melalui teks tertulis '*printed phenomenons*'. Atas alasan pemilihan metode pengumpulan data diatas, penulis dapat melakukan penelitian dalam rentang waktu yang singkat karena data yang diperlukan mudah didapat oleh penulis dan penulis tidak perlu secara langsung terjun ke lapangan dalam proses pengumpulan data.

Hasil dari pengolahan data yang di dapat menjadi acuan untuk penentuan langkah peneliti dalam membangun sistem informasi ini dan kemudian di kelompokkan kedalam beberapa tahapan sesuai dengan metode pembangunan sistem informasi yang digunakan. Metode pembangunan sistem informasi ini menggunakan metode *prototype*, metode *prototype* memiliki beberapa pengelompokan tahapan, dimulai dari tahapan *Planning* yaitu melakukan perencanaan pembangunan sistem dan jika perencanaan sudah matang maka akan masuk pada tahap *Analysis*, yaitu tahapan dimana menganalisa segala kebutuhan yang diperlukan dalam pembangunan sistem baik dalam kebutuhan data atau alat, jika tahapan analisis sudah terpenuhi maka akan masuk pada tahapan *Design*, yaitu tahapan yang membuat rancangan dari sitem yang akan dibuat, baik rancangan skema kerja sistem dan memberikan rancangan *prototype* sistem, jika ada perubahan akan kembali ke tahapan *design* namun jika tidak ada perubahan maka berlanjut pada tahap terakhir yaitu *Implementation*,

tahapan dimana semua tahapan sebelumnya di konversi menjadi sebuah sistem yang diinginkan.

3.1 *Planning*

Perencanaan (*planning*) adalah pemilihan atau penetapan tujuan organisasi dan penentuan strategi, kebijaksanaan, proyek, program, prosedur, metode, sistem, anggaran dan standar yang dibutuhkan untuk mencapai tujuan. Pembuatan keputusan banyak terlibat dalam fungsi ini (Dimas, 2012). *Planning* dalam pembangunan sistem informasi ini memiliki beberapa penetapan metode yang akan digunakan sebagai langkah tujuan, diantaranya:

3.1.1 *Pemilihan Metode Prototype*

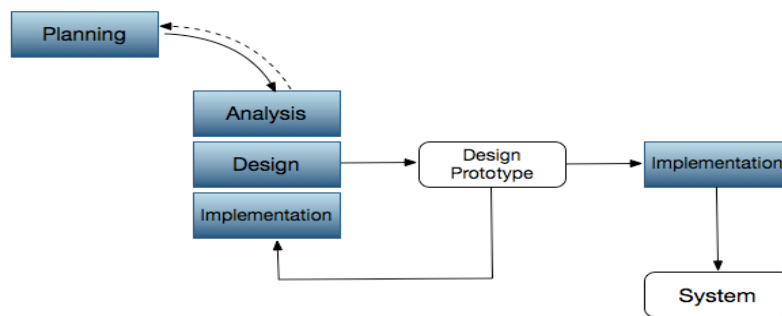
Prototyping adalah salah satu metode siklus hidup sistem yang didasarkan pada konsep model bekerja (*working model*). Tujuannya adalah mengembangkan model menjadi sistem final. Artinya sistem akan dikembangkan lebih cepat dari pada metode tradisional dan biayanya menjadi lebih rendah. Ada banyak jenis metode *prototyping*, begitu pula dengan penggunaannya (Wanda, 2015).

Pada metode ini, Pengembang dan *User* bertemu untuk melakukan identifikasi kebutuhan dari *software*, mengidentifikasi segala kebutuhan yang diketahui, menganalisa sistem dan melakukan studi kelayakan serta studi kebutuhan pemakai, meliputi model *interface*, teknik prosedural dan teknik pengembangan yang akan dipakai (Ummiati, 2011). Metode *prototyping* juga melibatkan pihak pengembang dan *User* untuk berkordinasi dalam melakukan desain, pengembang dan *User* saling bantu dalam mengidentifikasi format seluruh perangkat lunak, menentukan keperluan pengembangan, dan menentukan garis besar sistem yang akan dirancang.

Pemilihan metode ini dikarenakan penulis tidak memiliki acuan *User* yang spesifik untuk mendefinisikan kebutuhannya sehingga dengan penggunaan metode ini nantinya mampu mengurangi risiko kesalah pahaman ketika kebutuhan yang diinginkan *User* tidak dapat terdefinisi dengan baik oleh penulis, sederhananya tema yang diangkat dalam pembangunan sistem informasi ini nantinya akan melibatkan banyak *User* dalam penggunaannya. Selain itu, pemilihan metode ini karena memiliki pengembangan yang cepat karena menyederhanakan dan mempercepat desain sistem.

Penulis melakukan perbandingan metode *prototyping* dengan metode yang lainnya, apabila dibandingkan dengan metode *Waterfall* atau *Extreme Programmer (XP)*, perbedaannya terletak dengan adanya pembuatan *prototype* dari sebuah sistem yang akan

dibangun. Dalam fase ini, *prototype* yang telah dirancang oleh *developer* akan diberikan kepada *User* untuk dapat dievaluasi (Priyambudi, 2017). Tahap ini akan terus menerus diulang sampai kedua belah pihak benar-benar mengerti tentang *requirement* dari aplikasi yang akan dikembangkan. skema gambaran umum metode *prototyping* bisa dilihat seperti pada Gambar 3.1.



Gambar 3.1 Metode *Prototype*

Sumber: (Wixom, Barbara Harley, 2014)

Pembangunan sistem informasi pengelolaan pelaporan dan penanganan bencana asap berbasis *citizen science* ini memilih menggunakan metodologi *prototyping* setelah melakukan perbandingan keseluruhan dengan metode lainnya, perbandingan tersebut bisa dilihat pada Tabel 3.1.

Tabel 3.1 Perbandingan Metode

<i>Waterfall</i>	<i>Prototype</i>	<i>Extreme Programmer</i>	<i>Agile</i>
Kelebihan: <ul style="list-style-type: none"> • Merupakan model pengembangan paling handal dan paling lama digunakan. • Cocok untuk system software berskala besar. • Cocok untuk system software yang bersifat generic. • Pengerjaan project system akan terjadwal dengan baik dan mudah dikontrol. (www.xprorammin) 	Kelebihan: <ul style="list-style-type: none"> • <i>Prototype</i> melibatkan user dalam analisa dan desain. • Punya kemampuan menangkap requirement secara konkret daripada secara abstrak. • Untuk digunakan secara standalone. • Digunakan untuk memperluas SDLC. • Mempersingkat waktu pengembangan Sistem Informasi. • (www.xprorammin.com) 	Kelebihan: <ul style="list-style-type: none"> • Menjalin komunikasi yang baik dengan klien. (Planning Phase) • Menurunkan biaya pengembangan (Implementation Phase) • Meningkatkan komunikasi dan sifat saling menghargai antar developer. (Implementation Phase) • XP merupakan metodologi yang <u>semi formal</u>. (Planning Developer harus selalu siap dengan perubahan 	Kelebihan: <ul style="list-style-type: none"> • Meningkatkan kepuasan kepada klien • Pembangunan system dibuat lebih cepat • Mengurangi resiko kegagalan implementasi software dari segi non-teknis • Jika pada saat pembangunan system terjadi kegagalan, kerugian dar segi materi

g.com)		karena perubahan akan selalu diterima, atau dengan kata lain eksibel. (Maintenance Phase) (www.xproramming.com)	relative kecil. (www.xproramming.com)
<p>Kekurangan:</p> <ul style="list-style-type: none"> • Persyaratan system harus digambarkan dengan jelas. • Rincian proses harus benar-benar jelas dan tidak boleh berubah-ubah. • Sulit untuk mengadaptasi jika terjadi perubahan spesifikasi pada suatu tahapan pengembangan. (www.xproramming.com) 	<p>Kekurangan:</p> <ul style="list-style-type: none"> • Proses analisis dan perancangan terlalu singkat. • Mengesampingkan alternatif pemecahan masalah. • Bisanya kurang fleksible dalam menghadapi perubahan. • Protitype yang dihasilkan tidak selamanya mudah dirubah • Prototype terlalu cepat selesai. (www.xproramming.com) 	<p>Kekurangan:</p> <ul style="list-style-type: none"> • Tidak bisa membuat kode yang detail di awal (prinsip simplicity dan juga anjuran untuk melakukan apa yang diperlukan hari itu juga). • XP tidak memiliki dokumentasi formal yang dibuat selama pengembangan. Satu-satunya dokumentasi adalah dokumentasi awal yang dilakukan oleh user. (www.xproramming.com) 	<p>Kekurangan :</p> <ul style="list-style-type: none"> • <i>Developer</i> harus selalu siap dengan perubahan karena perubahan akan selalu diterima. (www.xproramming.com)

3.1.1 Pemilihan Metode Pengembangan

Sistem informasi yang penulis bangun tidak dibangun dari nol, melainkan melakukan pengembangan atau modifikasi terhadap suatu *platform* yang sudah ada dan bersifat *open-sources* sehingga dapat terbentuk suatu sistem baru dengan peningkatan fungsi dan *usability* dari sistem tersebut. Metode ini pengembangan ini dipilih dikarenakan beberapa alasan seperti dibutuhkan kerangka sebuah sistem yang sudah terbentuk sehingga bisa membantu penulis dalam membangun sistem pengelolaan pelaporan dan penanganan bencana asap ini, selain itu beberapa *platform open-sources* memiliki fitur yang dapat di modifikasi dan dikembangkan sesuai dengan fitur-fitur yang akan di bentuk pada sistem informasi yang akan penulis buat dan dengan tujuan semoga kelak sistem ini dapat di kembangkan lebih baik lagi dikarenakan memiliki kerangka yang bersifat *open-sources* sehingga banyak *developer* lainnya yang bisa berkontribusi dalam pengembangan *script* maupun fitur yang ada pada

aplikasi ini.

Sistem informasi pengelolaan pelaporan dan penanganan bencana asap ini dikembangkan tidak terlepas dari banyaknya perangkat lunak yang bersifat *open-source* seperti Ushahidi yang dapat digunakan maupun di modifikasi sesuai dengan kebutuhan yang diperlukan dalam pembangunan sistem informasi

Pemilihan *platform* Ushahidi ini sendiri di dapatkan dari hasil melakukan kajian dan perbandingan terhadap beberapa platform yang serupa dengan Ushahidi, yaitu *platform* Humanitarian OpenStreetMap Teams atau *Platform* besutan Google Crisis Mapping. Pemilihan Ushahidi ini dikarenakan kedua platform tersebut hanya fokus kepada pemetaan saja, selain itu Ushahdi menyediakan beberapa versi yang sesuai dengan kebutuhan pengembangan, sedangkan untuk kedua *platform* lainnya memiliki versi yang terbatas, misalnya pada Google Crisis Mapping tidak memiliki fitur selain menampilkan suatu kejadian sehingga jika menggunakan google crisis mapping peneliti akan membutuhkan waktu yang cukup lama dalam melakukan pengembangan, karena hanya fitur peta saja yang tersedia.

Dengan perbandingan tersebut tidak berarti peneliti mengadopsi 100% fitur yang ada pada Ushahidi, penggunaan Ushahidi ini dijadikan kerangka dalam pengembangan sistem informasi pengelolaan pelaporan dan penanganan bencana asap ini, serta melakukan modifikasi atau pembuatan ulang *script-script* dan perbaikan yang ada pada Ushahidi ini. Modifikasi yang di lakukan dari *platform* ini diantaranya :

- Modifikasi *script* CSS agar sesuai dengan rancangan yang dibutuhkan oleh aplikasi bencana asap.
- Pembuatan ulang *script* fitur pelaporan melalui Twitter, dikarenakan pada *platform* asli Ushahidi fitur ini tidak berjalan atau berfungsi karena masalah API.
- Pembuatan ulang *script* API email, dikarenakan pada ushahidi fitur ini tidak berjalan atau berfungsi.
- Modifikasi pada sistem instalasi atau perbaikan bug dikarenakan pengembangan Ushahidi ini sendiri tidak di rekomendasikan untuk di kembangkan pada sistem operasi MacOS dan hanya di rekomenadasikan untuk di kembangan pada perangkat lunak dengan sistem operasi Windows dan Linux, sedangkan peneliti melakukan pengembangan menggunakan perangkat keras dengan Sistem Operas MacOS.
- Melakukan modifikasi pada penggunaan OpenStreetMap (OMS) dikarenakan Ushahidi hanya menggunakan OMS Negara kenya untuk *mapping default* sedangkan fokus

pembangunan aplikasi ini ditujukan untuk kawasan atau teritorial Indonesia.

- Melakukan modifikasi *script* fitur SMS, dikarenakan fitur pelaporan SMS pada ushahidi ini tidak dapat berjalan atau berfungsi.
- Melakukan modifikasi pada tabel basis data.
- Melakukan modifikasi pada penggunaan *script* lainnya dan fungsi pada *framework codeigneter* yang di adopsi oleh Ushahidi.
- Penambahan fitur penyebaran Relawan.
- Konversi platform Ushahidi kedalam aplikasi android.

3.2 Analisis

Berdasarkan analisis yang telah dilakukan pada sistem informasi pengelolaan pelaporan dan penanganan bencana asap berbasis *citizen science* ini, penulis mengetahui hal apa saja yang akan menjadi masukan sistem *input*, keluaran sistem *output*, kebutuhan perangkat lunak dan antarmuka sistem yang akan di bentuk, sehingga sistem dapat di bentuk sesuai dengan harapan. Analisis kebutuhan adalah tahap awal yang dibutuhkan untuk mengetahui kebutuhan penulis dalam melakukan pengembangan perangkat lunak, meliputi kebutuhan masukan, proses dan keluaran, dan kebutuhan perangkat. Metode yang digunakan adalah kajian literatur terkait teori-teori yang dibutuhkan, dan pengumpulan data.

3.2.1 Analisis Kebutuhan Proses

Analisis kebutuhan proses ini akan menjadi data yang diolah oleh sistem, data tersebut berasal dari data masukan yang kemudian diproses oleh sistem dan menjadi keluaran berupa informasi. Analisis kebutuhan proses diterangkan pada Tabel 3.2.

Tabel 3.2 Tabel Kebutuhan Proses

Input	Proses	Output
Input login <i>user</i> (Username & Password).	Verifikasi username dan password.	Masuk ke dalam halaman admin.
Input kejadian (Nama Lokasi, Deskripsi Kejadian, Data Pelapor, Kategori Kejadian).	Pembuatan data pelaporan dan merekam data pelaporan.	Laporan kejadian.
Verifikasi Laporan (Login).	Update file laporan baru.	Menambahkan data laporan.
Manajemen User .	Update data dan status user.	Menambah User.

3.2.2 Analisis Kebutuhan Perangkat Keras

Kebutuhan perangkat keras disini merupakan alat yang digunakan dalam pembangunan sistem maupun untuk mengolah dan menampilkan hasil keluaran sistem. Untuk keluaran sistem perangkat keras yang digunakan harus memiliki spesifikasi minimal yang sudah ditentukan agar sistem dapat berjalan optimal, spesifikasi minimal tersebut dapat dilihat pada Tabel 3.3.

Tabel 3.3 Tabel Kebutuhan Perangkat Keras

Kebutuhan Perangkat Keras Yang digunakan Peneliti	Perangkat Keras yang dibutuhkan Untuk Pengoperasian Sistem
Komputer: <ul style="list-style-type: none"> Komputer MacBook Pro Core i5 2,5 Ghz. Sistem Operasi MacOS. Memory 4GB. Smartphone: <ul style="list-style-type: none"> Android Nugget. Memory 3GB 	Komputer: <ul style="list-style-type: none"> Sistem operasi Windows, Linux dan MacOS. Processor Intel Pentium 4. Memory 2GB. Smartphone: <ul style="list-style-type: none"> Android Jelly Bean. Memory 1GB.

3.2.3 Analisis Kebutuhan Perangkat Lunak

Kebutuhan perangkat lunak disini adalah komponen perangkat lunak yang digunakan sebagai alat dalam mendukung penelitian yang dilakukan, alat ini yang akan dijadikan sebagai alat pengolahan data dan pembuatan sistem yang akan dirancang dan keluaran sistem. Perangkat lunak yang harus digunakan dapat dilihat pada Tabel 3.4.

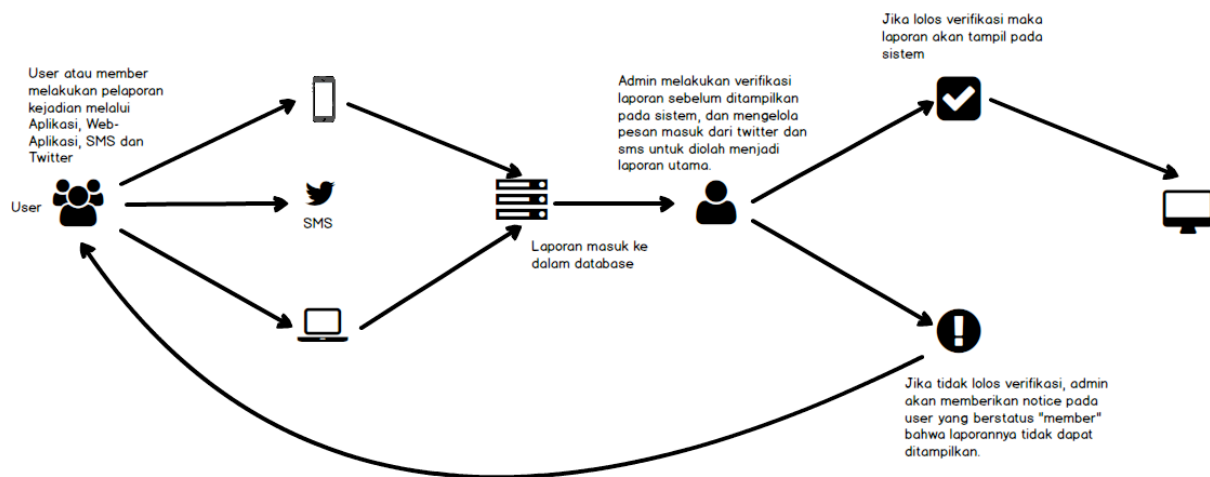
Tabel 3.4 Kebutuhan Perangkat Lunak

Software yang digunakan untuk membangun sistem	Software yang digunakan untuk pengoperasian sistem
<ul style="list-style-type: none"> Ushahidi (Platform Kerangka Sistem Informasi) PHPmyAdmin (Database server) SublimeTextEditor (Script editor PHP, SQL, JAVA, ETC) Manager-OSX (Log database) 	<ul style="list-style-type: none"> <i>Browser machine</i> (Safari, Google chrome, Mozila, etc). Aplikasi android Sistem Informasi Pengelolaan pelaporan dan penanganan bencana asap berbasis citizen science.

<ul style="list-style-type: none"> • Astah UML (Perancangan UML, UCS dan Activity Diagram) • CoedIgniter (Framework) • Android Studio (Pembuatan aplikasi android) • Bilsamic Mockups 3 (Pembuatan rancangan interface android aplikasi) 	
--	--

3.3 Perancangan Desain Sistem

Perancangan dilakukan untuk membuat penggambaran model sistem, aspek teknis dan implementasi dari sebuah sistem yang akan dibangun pada sistem informasi pengelolaan pelaporan dan penanganan bencana asap berbasis *citizence science* ini. Perancangan dilakukan melalui beberapa tahapan perancangan, seperti: perancangan *undefined model language (UML)*, perancangan activity diagram, perancangan basis data, perancangan antarmuka *web* aplikasi dan perancangan antarmuka aplikasi android. Peneliti membuat gambaran skema kerja sistem yang dijadikan acuan dalam menentukan langkah – langkah dalam perancangan yang dibutuhkan, skema kerja sistem tersebut seperti terlihat pada Gambar 3.2.

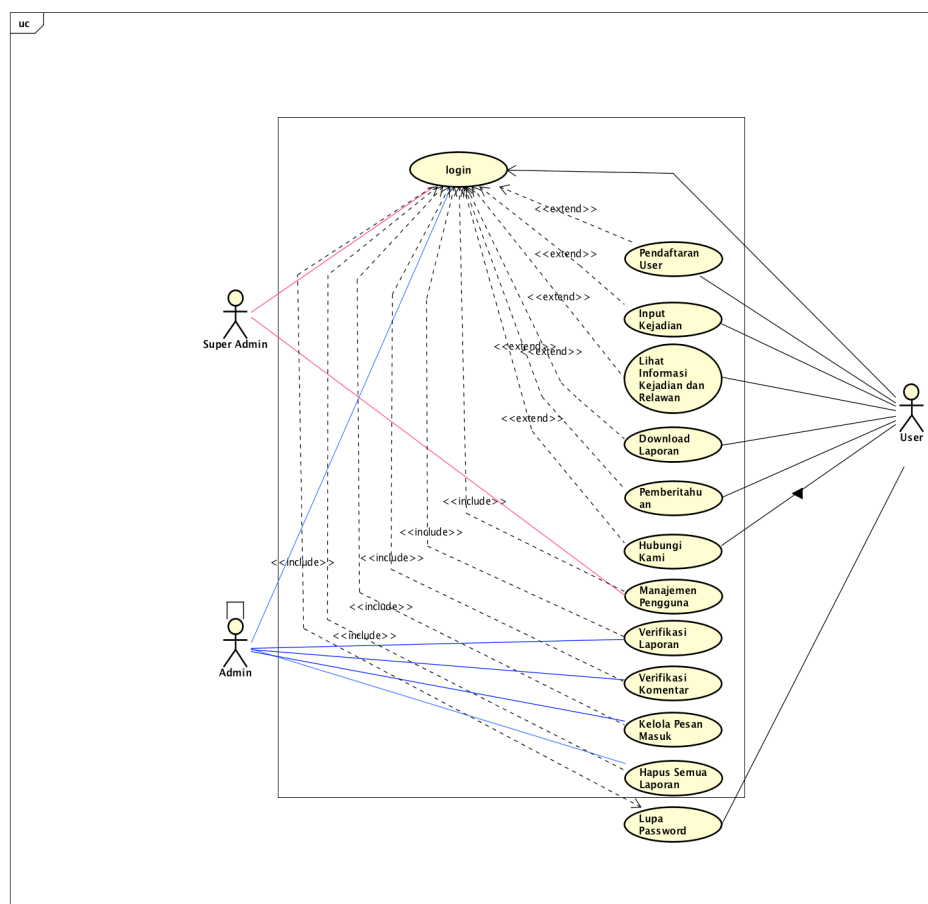


Gambar 3.2 Skema Kerja Sistem

3.3.1 Perancangan *Undefined Model*

Tahapan perancangan sistem akan dibuat dengan menggunakan *Undefined Model Language (UML)*. *Unified Modeling Language (UML)* adalah alat bantu analisis serta perancangan perangkat lunak berbasis objek”. Dalam pengerjaannya, UML mempunyai simbol atau diagram yang mempunyai fungsi tersendiri. Untuk mempresentasikannya, penulis menggunakan *Use-case Diagram* dan *Activity Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem, misalnya *login* ke sistem, meng-*create* sebuah daftar proses, dan sebagainya. Seorang / sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Usecase* diagram menjelaskan fungsionalitas sistem yang akan diimplementasikan pada bab selanjutnya. Terdapat 3 aktor yaitu admin, super admin dan user. Ada beberapa proses pengelolaan pada sistem informasi pengelolaan pelaporan dan penanganan bencana asap yang terdapat pada *use-case diagram* tersebut. *Use-case Diagram* dijelaskan pada Gambar 3.3.

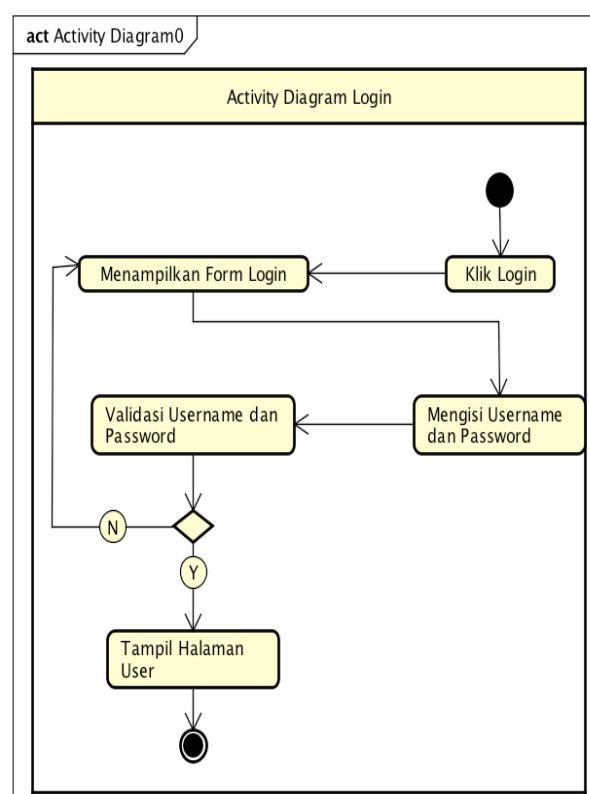


Gambar 3.3 Perancangan *Use Case Diagram*

3.3.2 Perancangan *Activity Diagram*

A. *Activity Diagram Login*

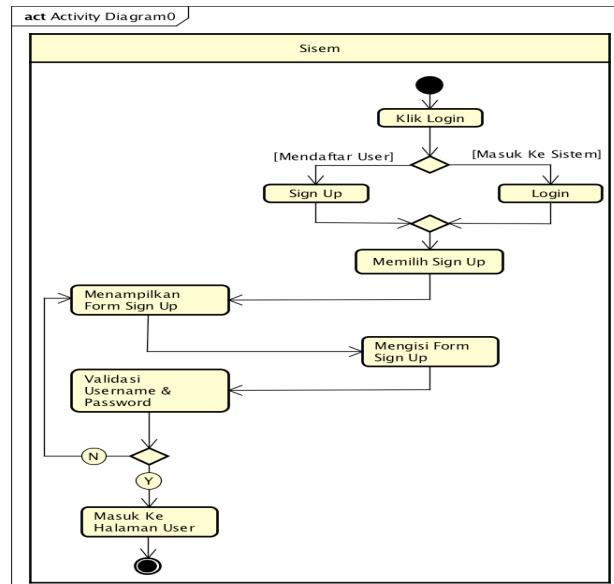
Activity diagram ini menjelaskan proses *login User*. *User* mengakses *web-aplikasi* kemudian klik menu *login*, maka sistem akan menampilkan *form username* dan *password* yang harus diisi oleh *User*. Jika *username* dan *password* yang dimasukan *User* benar maka sistem akan menampilkan halaman utama, namun jika salah maka sistem akan kembali menampilkan *form login* yang harus diisi *User* dengan benar. Penjelasan *activity diagram* dapat dilihat pada Gambar 3.4.



Gambar 3.4 *Activity Diagram Login*

B. *Activity Diagram Pendaftaran*

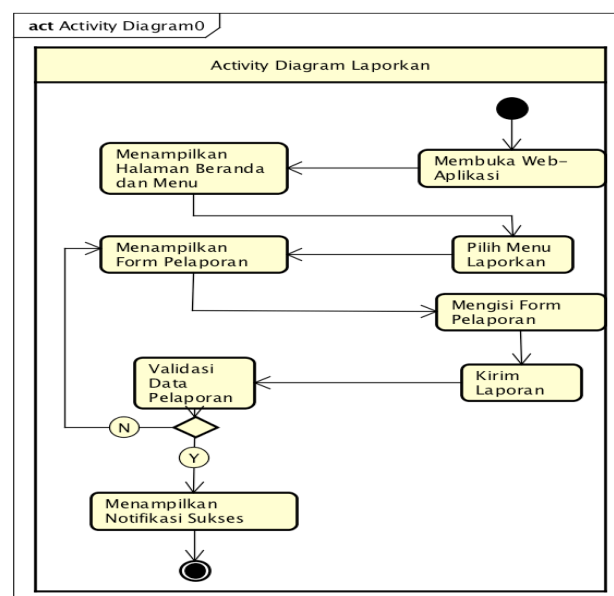
Activity diagram pendaftaran menjeaskan tentang aktivitas pendaftaran *User* menjadi *member*. *User* mengakses *web-aplikasi*, lalu memilih menu *login*, dalam menu *login* terdapat *button sign up* yang harus dipilih, kemudian sistem akan menampilkan *form* yang harus diisi oleh *User*. *Form* yang diisi berupa nama, email, *username* dan *password*. *Activity diagram* pendaftaran dapat dilihat pada Gambar 3.5.



Gambar 3.5 Activity Diagram Pendaftaran

C. Activity Diagram Laporkan

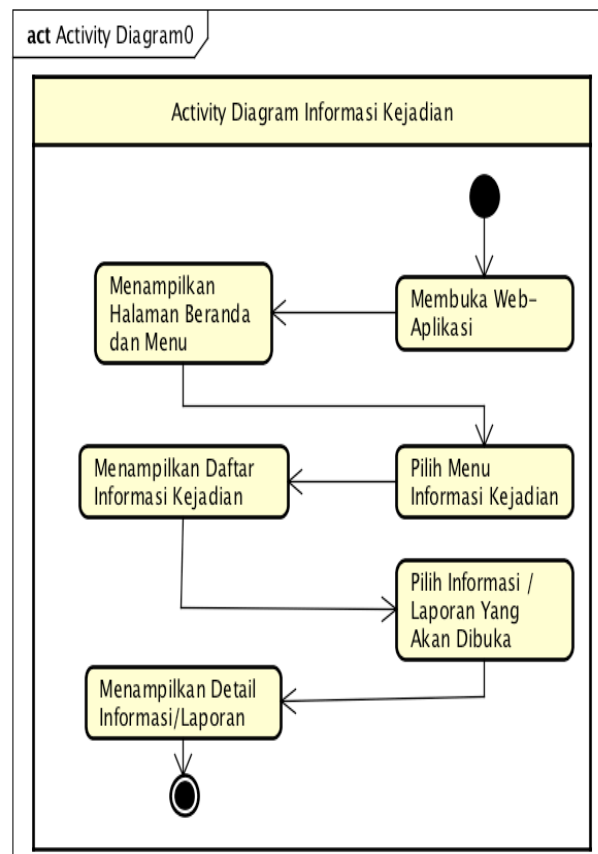
Activity diagram ini menjelaskan proses pelaporan kejadian atau pelaporan bencana yang dilakukan oleh *User* dan dapat dilakukan tanpa harus *login* atau menjadi *member*. *User* mengakses *web-aplikasi* kemudian memilih menu Lapoeakan!, lalu sistem akan menampilkan *form* pelaporan yang harus diisi oleh *User*, *form* tersebut berisi judul, deskripsi kejadian, kategori kejadian, waktu, informasi pelapor, lokasi, video dan foto kejadian. Jika *form* yang sudah diisi benar dan lengkap, maka sistem akan kembali ke halaman utama setelah laporan dikirim, namun jika kurang lengkap maka sistem akan otomatis menampilkan kembali *form* isian laporkan. *Activity diagram* dapat dilihat pada Gambar 3.6.



Gambar 3.6 Activity Diagram Laporkan

D. Activity Diagram Informasi Kejadian

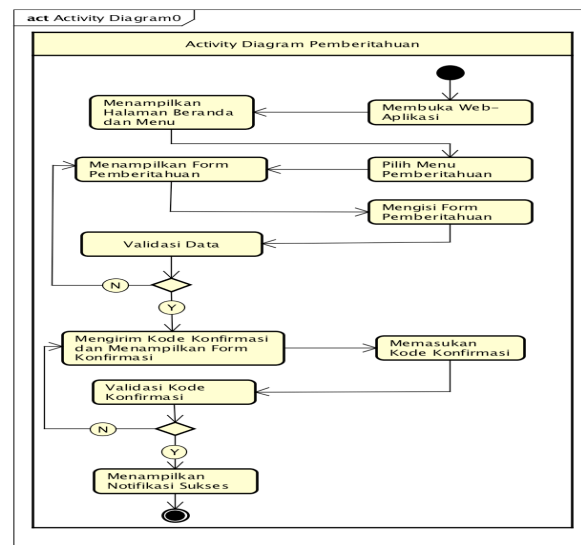
Activity diagram informasi kejadian menjelaskan tentang proses yang dilakukan *User* untuk melihat pelaporan kejadian yang masuk serta penyebaran relawan yang tersedia. *User* mengakses *web-aplikasi* dengan *login* maupun tanpa *login* kemudian memilih menu Informasi Kejadian, maka sistem akan menampilkan daftar laporan kejadian dan relawan. *Activity diagram* dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Activity Diagram* Lihat Informasi Kejadian

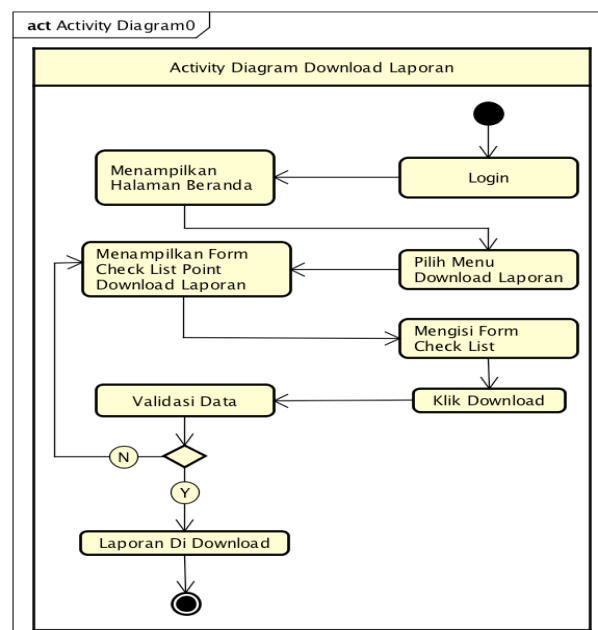
E. Activity Diagram Pemberitahuan

Activity diagram ini menjelaskan tentang proses permintaan pemberitahuan laporan masuk oleh *User*. Proses pemberitahuan dimulai dari *User* memilih menu pemberitahuan, kemudian mengisi *form* yang disediakan, *form* yang harus diisi adalah alamat email yang akan dikirim pemberitahuan dan pilihan kategori kejadian. Setelah mengisi *form* sistem akan mengirim kode konfirmasi kepada email yang dimasukan. Setelah kode diterima, *User* harus melakukan verifikasi dengan mengisi kode tersebut dan permintaan pemberitahuan sukses dijalankan. *Activity diagram* dapat dilihat pada Gambar 3.8.

Gambar 3.8 *Activity Diagram Pemberitahuan*

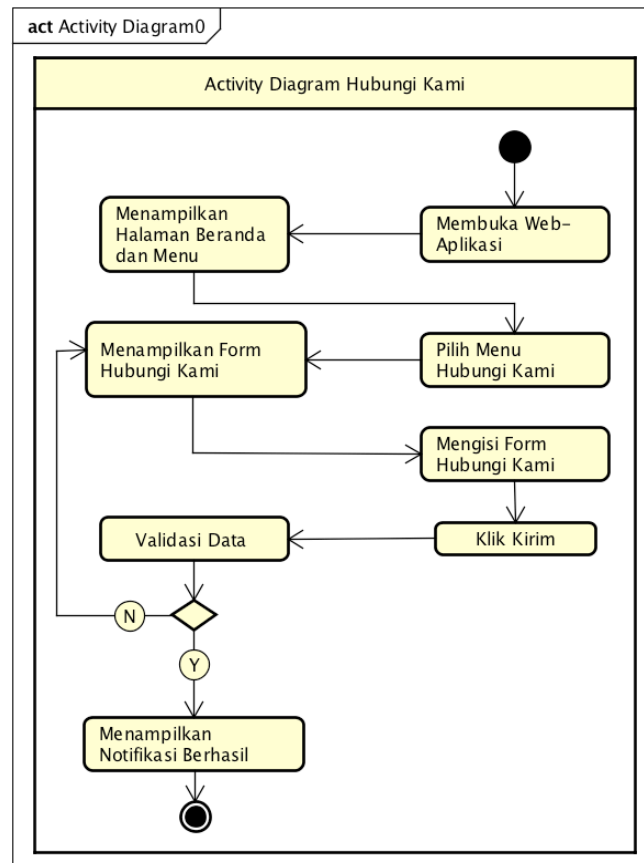
F. *Activity Diagram Download Laporan*

Activity diagram ini menjelaskan proses unduh laporan yang dilakukan oleh *User*. Laporan yang dapat diunduh tidak hanya laporan yang sudah diverifikasi atau diterima, yang belum diverifikasi ataupun belum diterima dapat diunduh. Proses unduh dimulai dengan dari *User* mengakses *web-aplikasi* kemudian memilih menu *Download*, lalu *User* dapat memilih salah satu laporan atau semua laporan dan menentukan interval waktu laporan yang akan diunduh beserta memilih format laporan yang akan diunduh. *Activity Diagram* dapat dilihat pada Gambar 3.9.

Gambar 3.9 *Activity Download Laporan*

G. Activity Diagram Hubungi Kami

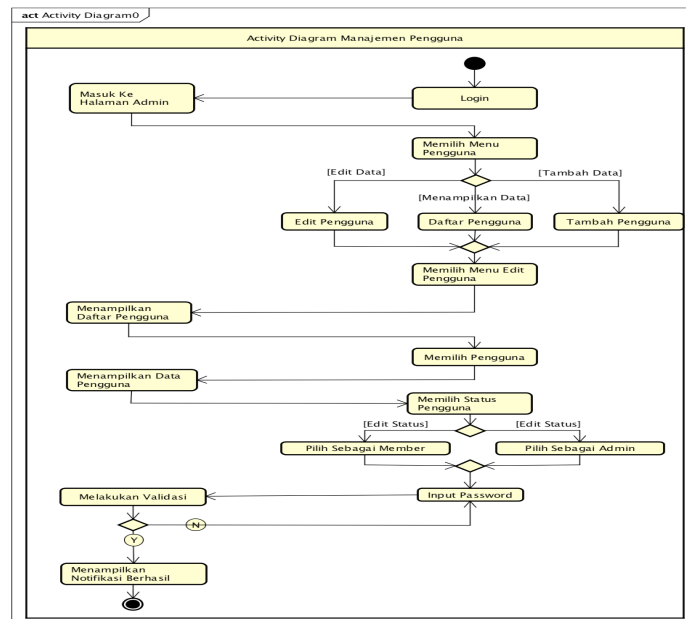
Activity diagram ini menjelaskan proses mengirimkan kritik dan saran oleh *User*. Setelah membuka *web-aplikasi*, *User* memilih menu kritik dan saran, kemudian mengisi form yang telah disediakan. Jika tidak ada kesalahan dalam pengisian maka *User* akan menerima notifikasi sukses setelah di klik “*irim*”. *Activity Diagram* dapat dilihat pada Gambar 3.10.



Gambar 3.10 Activity Diagram Hubungi Kami

H. Activity Diagram Manajemen Pengguna

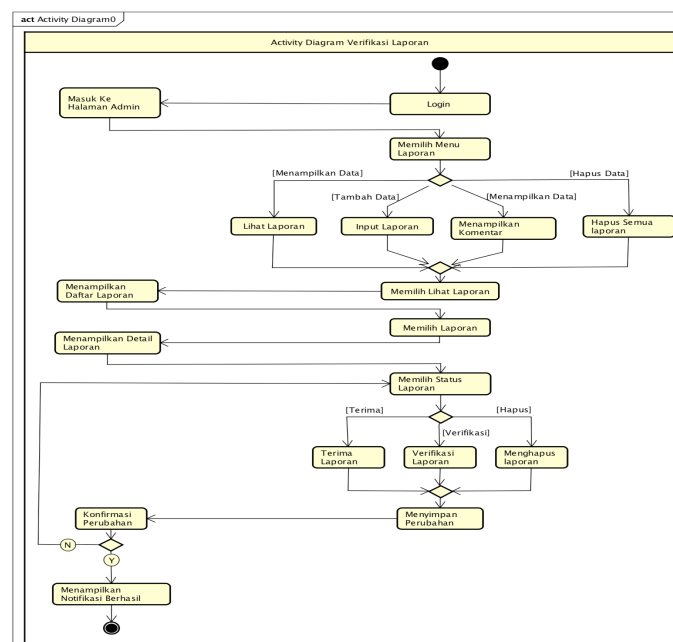
Activity diagram ini menjelaskan tentang proses yang dilakukan oleh *Super Admin* dalam merubah status *member* menjadi *Admin* atau *Member* biasa. *Super Admin* perlu melakukan *login terlebih dahulu*, setelah *login* akan masuk ke halaman *Admin* dan memilih menu pengguna. Maka akan tampil daftar pengguna dan *Admin*, lalu *super admin* memilih Pengguna atau *Member* mana yang akan dirubah statusnya dengan memilihnya dan melakukan *edit*. Setelah itu *Super Admin* harus memasukan *password* sebagai verifikasi sebelum perubahan dapat tersimpan, jika *password* yang dimasukan salah maka perubahan tidak akan tersimpan. *Activity diagram* dapat dilihat pada Gambar 3.11.



Gambar 3.11 Activity Diagram Manajemen Pengguna

I. Activity Diagram Verifikasi Pelaporan

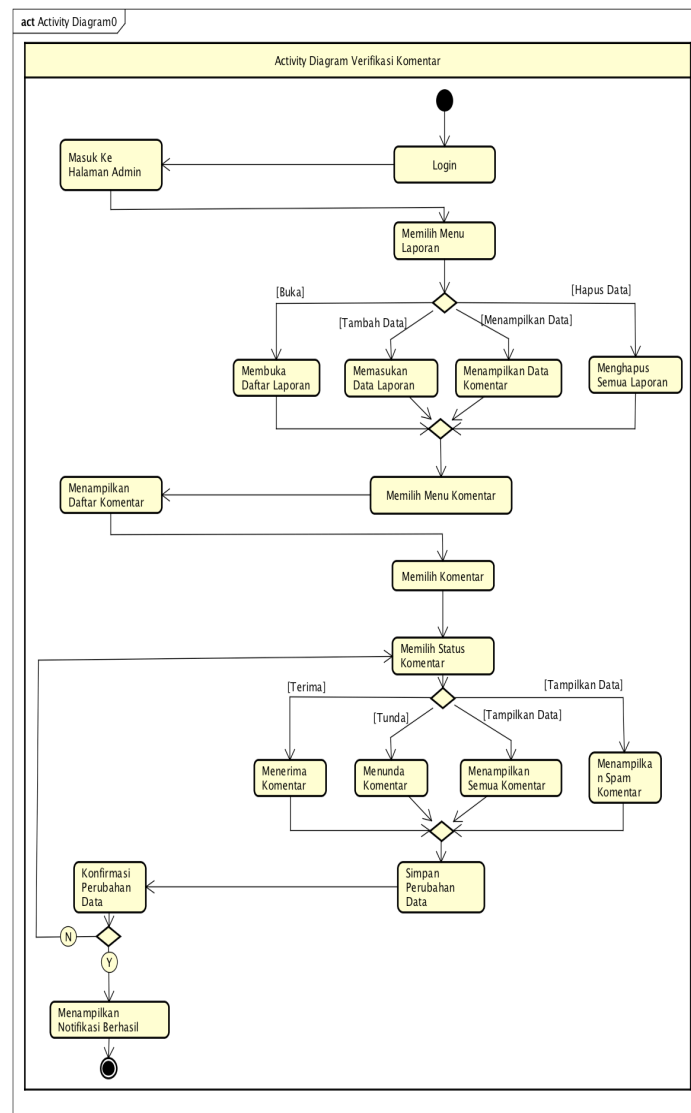
Activity diagram ini menjelaskan proses verifikasi laporan yang masuk oleh Admin. Selain untuk verifikasi, activity diagram ini juga digunakan untuk menampilkan laporan yang masuk ke dalam bagian front-end web-aplikasi. Admin harus login terlebih dahulu, setelah login, admin masuk ke menu laporan masuk lalu memilih laporan mana yang akan diverifikasi dan ditampilkan dengan klik button verifikasi dan tampilkan yang ada pada bagian bawah laporan. Activity diagram dapat dilihat pada Gambar 3.12.



Gambar 3.12 Activity Diagram Verifikasi Laporan

J. Activity Diagram Verifikasi Komentar

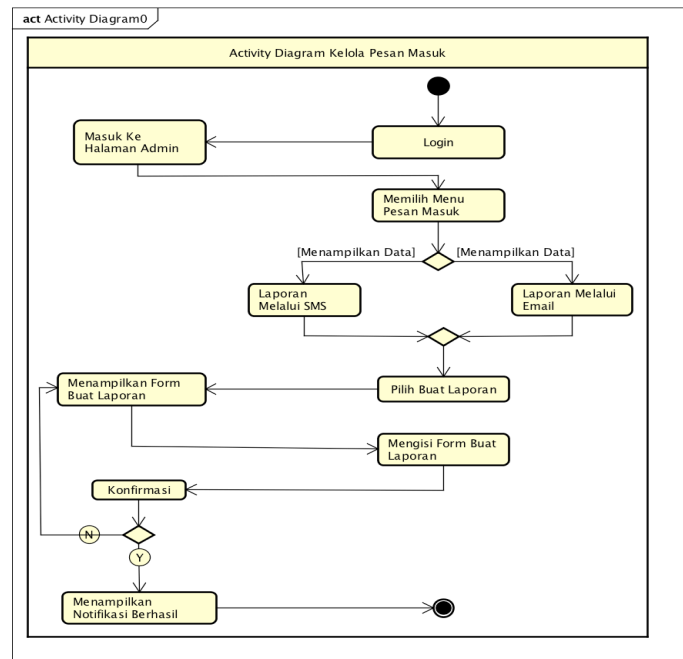
Activity diagram ini menjelaskan proses verifikasi komentar yang masuk, verifikasi dilakukan oleh *Admin* dan proses verifikasinya sama dengan proses verifikasi laporan, perbedaannya terletak pada pemilihan menu, pada proses verifikasi komentar *Admin* memilih menu komentar kemudian memilih komentar mana yang akan ditampilkan. *Activity diagram* dapat dilihat pada Gambar 3.13.



Gambar 3.13 Activity Diagram Verifikasi Komentar

K. Activity Diagram Kelola Pesan Masuk

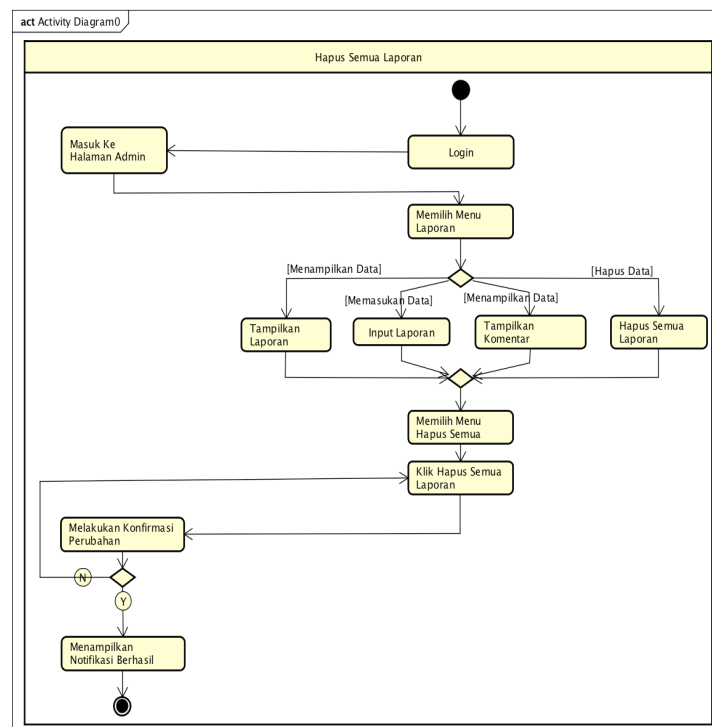
Activity diagram ini menjelaskan proses pengelolaan pesan masuk dari SMS atau Twitter yang kemudian dijadikan laporan oleh admin. Setelah *login*, *Admin* memilih menu pesan masuk, kemudian *Admin* memilih pesan masuk mana yang akan diedit untuk dijadikan laporan. *Activity diagram* dapat dilihat pada Gambar 3.14.



Gambar 3.14 Activity Diagram Kelola Pesan Masuk

L. Activity Diagram Hapus Semua Laporan

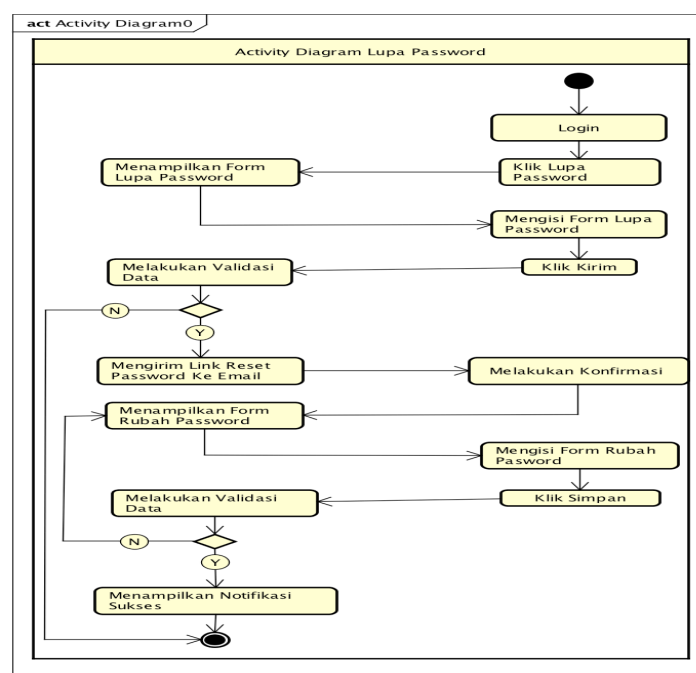
Activity diagram ini menjelaskan proses menghapus seluruh laporan yang masuk pada sistem, baik melalui menu laporan maupun pesan masuk. Setelah *login*, Admin memilih menu “Laporan Masuk”, kemudian Admin memilih menu “Hapus Semua”. Activity diagram dapat dilihat pada Gambar 3.15.



Gambar 3.15 Activity Diagram Hapus Semua Laporan

M. Activity Diagram Lupa Password

Activity diagram lupa password menjelaskan cara untuk melakukan reset ulang password bagi User yang sudah terdaftar sebagai Member. *User* membuka mengakses *web-aplikasi*, kemudian memilih menu *login*, kemudian memilih menu *lupa password*. Setelah itu sistem akan menampilkan *form* alamat email yang harus diisi *User*, email yang dimasukan harus alamat email yang didaftarkan sebelumnya, setelah alamat email dimasukan dan klik *reset*, maka *User* akan menerima URL reset *password* pada email yang dimasukan tadi untuk melakukan reset password dan memasukan *password* baru. *Activity diagram lupa password* dapat dilihat pada Gambar 3.16.



Gambar 3.16 Activity Diagram Hapus Lupa Password

3.3.3 Perancangan Struktur Basisdata

Pada sistem informasi pengelolaan pelaporan dan penanganan bencana asap berbasis citizen science ini dalam perancangan basisdata menggunakan phpMyAdmin yang merupakan paketan dari Xampp sebagai penyedia layanan basisdata. Karena sistem informasi ini dibangun dengan sifat mengembangkan suatu *platform* Ushahidi dan meskipun Ushahidi menyediakan beberapa basis data bawaan, tapi penulis tidak menggunakan semua basisdata yang disediakan Ushahidi, bahkan penulis melakukan beberapa tambahan struktur basisdata, yaitu; Tabel Twitter, Tabel smssync, Tabel incident_person.

Sedangkan untuk basisdata yang tidak digunakan penulis karena tidak sesuai dengan menu yang akan diimplementasikan dalam sistem ini yaitu sebagai berikut:

1. Tabel openid
2. Tabel page
3. Tabel maintenance
4. Tabel layer
5. Tabel incident lang
6. Tabel form_field
7. Tabel form_field_option
8. Tabel form_response
9. Tabel feed
10. Tabel feed_item
11. Tabel feed_item_category

Sedangkan untuk untuk struktrur tabel basisdata yang digunakan pada *web*-aplikasi dan aplikasi android akan diterangkan pada tabel – tabel berikut:

A. Perancangan Struktur Basisdata *Web* Aplikasi

1. Tabel Actions

Tabel actions berfungsi untuk menyimpan aksi yang dilakukan oleh *User* terhadap sistem, aksi yang disimpan berupa kualifikasi yang dilakukan *User* aktif. Struktur tabel actions dapat dilihat pada Tabel 3.5.

Tabel 3.5 Tabel actions

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Action_id	Int(11)	Primary key	
2.	Action	Varchar(75)		
3.	Qualifiers	Text		
4.	Response	Varchar(75)		
5.	Responses	Text		
6.	active	tinyint		

2. Tabel action_log

Tabel actions_log ini sebagai tempat untuk menyimpan aksi atau sebuah tindakan dari *User* atau pengguna dan dicatat dengan waktu selama melakukan tindakannya tersebut. Struktur tabel actions_log dapat dilihat pada Tabel 3.6.

Tabel 3.6 Tabel action_log

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(2)	Primary key	
2.	Acrion_id	Int(11)		
3.	User_id	Int(11)		
4.	time	Int(10)		

3. Tabel alert

Tabel alert ini sebagai tempat untuk menyimpan data peringatan kejadian yang masuk pada sistem. Struktur tabel alert dapat dilihat pada Tabel 3.7.

Tabel 3.7 Tabel alert

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	User_id	Int(11)	Foreign key	
3.	Alert_type	Tinyint(4)		
4.	Alert_recepient	Varchar(200)		
5.	Alert_code	Varchar(30)	Foreign key	
6.	Alert_confirmed	Tinyint(4)		
7.	Alert_lat	Varchar(150)		
8.	Alert_lon	Varchar(150)		
9.	Alert_radius	Tinyint(4)		
10.	Alert_ip	Varchar(100)		

4. Tabel alert_category

Tabel alert_category berfungsi untuk menyimpan peringatan kategori yang sudah ditentukan. Struktur tabel alert_category dapat dilihat pada Tabel 3.8.

Tabel 3.8 Tabel alert_category

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Alert_id	Bigint(20)	Foreign key	
3.	Category_id	Int(11)	Foreign key	

5. Tabel alert_sent

Tabel alert_sent berfungsi untuk menyimpan data peringatan yang sudah dikirimkan sistem terhadap *User* sesuai dengan permintaan. Sturuktur tabel dapat dilihat pada Tabel 3.9.

Tabel 3.9 Tabel alert_sent

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	
3.	Alert_id	Bigint(20)	Foreign key	
4.	Alert_date	datetime		

6. Tabel api_banned

Tabel ini berfungsi untuk menyimpan daftar api yang diblokir oleh sistem agar tidak mengganggu fungsi sistem. Sturuktur tabel api_banned dapat dilihat pada Tabel 3.10.

Tabel 3.10 Tabel api_banned

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Banned_ipaddress	Varchar(50)		
3.	Banned_date	datetime		

7. Tabel api_log

Tabel ini berfungsi untuk menyimpan data api yang digunakan oleh sistem. Stuktur tabel api_log dapat dilihat pada Tabel 3.11.

Tabel 3.11 Tabel api_log

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Api_task	Varchar(10)		
3.	Api_parameters	Varchar(100)		
4.	Api_records	Tinyint(11)		
5.	Api_ipaddress	Varchar(50)		
6.	Api_date	datetime		

8. Tabel api_setting

Tabel ini berfungsi untuk menyimpan pengaturan api yang digunakan pada sistem. Struktur tabel api_setting dapat dilihat pada Tabel 3.12.

Tabel 3.12 Tabel api_setting

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Default_record_limit	Int(11)		
3.	Max_record_limit	Int(11)		
4.	Max_request_per_ip_address	Int(11)		
5.	Max_resuest_quota_basis	Int(11)		
6.	Modification_date	datetime		

9. Tabel badge

Tabel ini berfungsi untuk menyimpan data status *User* yang masuk ke dalam sistem. Struktur tabel ini dapat dilihat pada Tabel 3.13.

Tabel 3.13 Tabel badge

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Name	Varchar(250)		
3.	description	text		

10. Tabel badge_user

Tabel ini berfungsi untuk mnyimpan status *User*. Struktur tabel dapat dilihat pada Tabel 3.14.

Tabel 3.14 Tabel badge_user

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	User_id	Int(11)	Primary key	
2.	Badge_id	Int(11)	Primary key	

11. Tabel category

Tabel ini berfungsi untuk menyimpan kategori kejadian yang dibuat oleh Admin. Struktur tabel dapat dilihat pada Tabel 3.15.

Tabel 3.15 Tabel category

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Parent_id	Int(11)	Foreign key	
3.	Locale	Varchar(10)		
4.	Category_position	Tinyint(4)		
5.	Category_title	Varchar(255)		
6.	Category_description	Text		
7.	Category_color	Varchar(20)		
8.	Category_image	Varchar(255)		
9.	Category_image_thumb	Varchar(255)		
10.	Category_visible	Tinyint(4)	Foreign key	
11.	Category_trusted	Tinyint(4)		

12. Tabel city

Tabel ini berfungsi untuk menyimpan daftar kota yang ada pada OpenStreetMap, daftar kota yang kemudian dipilih *user* ketika membuat pelaporan kejadian bencana asap. Struktur tabel dapat dilihat pada Tabel 3.16.

Tabel 3.16 Tabel city

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Country_id	Int(11)	Foreign key	
3.	City	Varchar(200)		
4.	City_lat	Varchar(150)		
5.	City_lan	Varchar(200)		

13. Tabel comment

Tabel ini berfungsi menyimpan data komentar yang diberikan oleh *User*. Struktur tabel dapat dilihat pada Tabel 3.17.

Tabel 3.17 Tabel comment

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	

3.	User_id	Int(11)	Foreign key	
4.	Comment_uthor	Varchar(100)		
5.	Comment_email	Varchar(120)		
6.	Comment_description	Text		
7.	Comment_ip	Varchar(100)		
8.	Comment_spam	Tinyint(4)		
9.	Comment_active	Tinyint(4)		
10.	Comment_date	Datetime		
11.	Comment_date_gmt	Datetime		

14. Tabel cluster

Tabel ini berfungsi untuk menyimpan data geografis dari lokasi yang dimasukan oleh *User* dalam input pelaporan kejadian. Struktur tabel dapat dilihat pada Tabel 3.18.

Tabel 3.18 Tabel cluster

No.	Nama Kolom	Type Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Location_id	Bigint(20)	Foreign key	
3.	Latitude	Double		
4.	Longitude	Double		
5.	Latitude_min	Double		
6.	Longitude_min	Double		
7.	Latitude_max	Double		
8.	Longitude_max	Double		
9.	Child_count	Int(11)		
10.	Parent_id	Int(11)		
11.	Left_side	Int(11)		
12.	Ridght_side	Int(11)		
13.	Level	Int(11)		
14.	Incident_id	Bigint(20)	Foreign key	
15.	Category_id	Int(11)	Foreign key	

15. Tabel country

Tabel ini berfungsi untuk menyimpan data negara yang ada pada sistem, data negara yang disediakan oleh OpenStreetMap yang kemudian bisa digunakan sistem untuk menampilkan map yang dipilih. Struktur tabel dapat dilihat pada tabel 3.19.

Tabel 3.19 Tabel country

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Iso	Varchar(10)		
3.	Country	Varchar(100)		
4.	Capital	Varchar(100)		
5.	cities	Tinyint(4)		

16. Tabel external_app

Tabel ini berfungsi untuk menyimpan data alamat *external* aplikasi seperti url download aplikasi android atau lainnya yang akan ditampilkan pada *Web-Aplikasi*. Struktur tabel dapat dilihat pada Tabel.3.20.

Tabel 3.20 Tabel extrenal_app

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Name	Varchar(255)		
3.	url	Varchar(255)		

17. Tabel form

Tabel ini berfungsi untuk menyimpan data *form* yang akan diisi oleh *User*. Struktur tabel *form* dapat dilihat pada Tabel 3.21.

Tabel 3.21 Tabel form

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Form_title	Varchar(200)	Foreign key	
3.	Form_description	Text		
4.	Form_active	Tinyint(4)		

18. Tabel form_field

Tabel ini berfungsi untuk menyimpan detail data pada *form* yang dipilih dan diisi oleh *User* sesuai keperluannya. Struktur tabel form_field dapat dilihat pada Tabel 3.22.

Tabel 3.22 Tabel form_field

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Form_id	Int(11)	Foreign key	
3.	Field_name	Varchar(200)	Foreign key	
4.	Field_type	Tinyint(4)		
5.	Field_required	Tinyint(4)		
6.	Field_position	Tinyint(4)		
7.	Field_default	Text		
8.	Field_maxlength	Int(11)		
9.	Field_width	Smallint(6)		
10.	Field_height	Tinyint(4)		
11.	Field_isdate	Tinyint(4)		
12.	Field_ispublic_visible	Tinyint(4)		
13.	Field_ispublic_submit	Tinyint(4)		

19. Tabel field_option

Tabel ini berfungsi untuk membuat *form* baru yang akan digunakan pada sistem. Struktur tabel dapat dilihat pada Tabel 3.23.

Tabel 3.23 Tabel field_option

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Form_field_option	Int(11)	Foreign key	
3.	Option_name	Varchar(200)		
4.	Option_value	text		

20. Tabel form_response

Tabel ini berfungsi untuk menyimpan data respon dari *form* yang diisi *User*. Struktur tabel dapat dilihat pada Tabel 3.24.

Tabel 3.24 Tabel form_response

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Form_field_id	Int(11)	Foreign key	
3.	Incident_id	Bigint(20)	Foreign key	
4.	Form_response	text		

21. Tabel geometry

Tabel ini berfungsi untuk menyimpan data geografis dari lokasi yang dipilih *User* pada peta. Struktur tabel dapat dilihat pada Tabel 3.25.

Tabel 3.25 Tabel geometry

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	
3.	Geometry	Geometry	Foreign key	
4.	Geometry_label	Varchar(155)		
5.	Geometry_comment	Varchar(255)		
6.	Geometry_color	Varchar(25)		
7.	Geometry_strokewidth	Varchar(5)		

22. Tabel incident

Tabel ini berfungsi untuk menyimpan data pelaporan kejadian yang dimasukan oleh *User*. Struktur tabel dapat dilihat pada Tabel 3.26.

Tabel 3.26 Tabel incident

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Location_id	Bigint(20)	Foreign key	
3.	Form_id	Int(11)	Foreign key	
4.	Locale	Varchar(10)		
5.	User_id	Int(11)	Foreign key	
6.	Incident_title	Varchar(25)		
7.	Incident_description	Longtext		
8.	Incident_date	Datetime	Foreign key	

9.	Incident_mode	Tinyint(4)	Foreign key	
10.	Incident_active	Tinyint(4)	Foreign key	
11.	Incident_veriified	Tinyint(4)	Foreign key	
12.	Incident_dateadd	Datetime		
13.	Incident_dateadd_gmt	Datetime		
14.	Incident_datemodify	Datetime		
15.	Incident_alert_status	Tinyint(4)		
16.	Incident_zoom	Tinyint(4)		

23. Tabel incident_category

Tabel ini berfungsi untuk menyimpan data kategori pelaporan kejadian yang dimasukan oleh *User*. Struktur tabel dapat dilihat pada Tabel 3.27.

Tabel 3.27 Tabel incident_category

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	
3.	Category_id	Int(11)	Foreign key	

24. Tabel incident_lang

Tabel ini berfungsi untuk menyimpan data kejadian yang dimasukna oleh *User*. Struktur tabel dapat dilihat pada Tabel 3.28.

Tabel 3.28 Tabel incident_lang

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	
3.	Locale	Varchar(10)		
4.	Incident_title	Varchar(255)		
5.	Incident_description	longtext		

25. Tabel incident_lang

Tabel ini berfungsi untuk menyimpan deta deskripsi kejadiin yang dimasukan oleh *User*. Tabel dapat dilihat pada Tabel 3.29.

Tabel 3.29 Tabel incident_lang

No	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	
3.	Locale	Varchar(10)		
4.	Incident_title	Varchar(255)		
5.	Incident_description	longtext		

26. Tabel incident_person

Tabel ini berfungsi untuk menyimpan data pelapor kejadian. Struktur tabel dapat dilihat pada Tabel 3.30.

Tabel 3.30 Tabel incident_person

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	
3.	Person_first	Varchar(200)		
4.	Person_last	Varchar(200)		
5.	Person_email	Varchar(120)		
6.	Person_phone	Varchar(60)		
7.	Person_phonnee	Varchar(15)		
8.	Person_ip	Varchar(50)		
9.	Person_date	datetime		

27. Tabel level

Tabel ini berfungsi untuk menyimpan data tingkatan. struktur tabel dapat dilihat pada Tabel 3.31.

Tabel 3.31 Tabel layer

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Level_title	Varchar(200)		
3.	Level_description	Varchar(200)		
4.	Level_weight	Tinyint(4)		

28. Tabel location

Tabel ini berfungsi untuk menyimpan data lokasi yang dipilih *User*. saat membuat laporan kejadian. Struktur tabel dapat dilihat pada Tabel 3.32.

Tabel 3.32 Tabel layer

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Location_name	Varchar(255)		
3.	Country_id	Int(11)	Foreign key	
4.	Latitude	Double	Foreign key	
5.	Longitude	Double	Foreign key	
6.	Location_visible	Tinyint(4)		
7.	Location_date	Datetime		

29. Tabel maintenance

Tabel ini berfungsi untuk memperbaiki ip pada sistem. Struktur tabel dapat dilihat pada Tabel 3.33.

Tabel 3.33 Tabel maintenance

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Allowed_ip	Varchar(15)	Primary key	

30. Tabel media

Tabel ini berfungsi untuk menyimpan data media yang masuk. Struktur tabel dapat dilihat pada tabel 3.34.

Tabel 3.34 Tabel media

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Location_id	Bigint(20)	Foreign key	
3.	Incident_id	Bigint(20)	Foreign key	
4.	Message_id	Bigint(20)	Foreign key	
5.	Badge_id	Int(11)	Foreign key	
6.	Media_type	Tinyint(4)		
7.	Media_title	Varchar(255)		
8.	Media_description	Longtext		

9.	Media_link	Varchar(255)		
10.	Media_medium	Varchar(255)		
11.	Media_thumb	Varchar(255)		
12.	Media_date	Datetime		
13.	Media_active	Tinyint(4)		

31. Tabel message

Tabel ini berfungsi untuk menyimpan data pesan yang masuk pada sistem. Struktur tabel dapat dilihat pada Tabel 3.35.

Tabel 3.35 Tabel message

No.	Nama Kolom	Type Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Parent_id	Bigint(20)		
3.	Incident_id	Bigint(20)	Foreign key	
4.	User_id	Int(11)	Foreign key	
5.	Reporter_id	Bigint(20)	Foreign key	
6.	Service_messageid	Varchar(100)		
7.	Message_from	Varchar(100)		
8.	Message_to	Varchar(100)		
9.	Message	Text		
10.	Message_detail	Text		
11.	Message_type	Tinyint(4)		
12.	Message_date	Datetime		
13.	Message_level	Tinyint(4)		
14.	Latitude	Double		
15.	longitude	double		

32. Tabel permissions

Tabel ini berfungsi untuk menyimpan data permissions. Struktur tabel dapat dilihat pada Tabel 3.36.

Tabel 3.36 Tabel permissions

No.	Nama Kolom	Type Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	

2.	name	Varchar(32)	Foreign key	
----	------	-------------	-------------	--

33. Tabel permissions_roles

Tabel ini berfungsi untuk menyimpan data permissions roles dari pengguna. Struktur tabel dapat dilihat pada Tabel 3.37.

Tabel 3.37 Tabel permissions_roles

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Role_id	Int(11)	Primary key	
2.	Permissions_id	Int(11)	Primary key	

34. Tabel plugins

Tabel ini berfungsi untuk menyimpan data *plugins* sistem. Struktur tabel dapat dilihat pada Tabel 3.38.

Tabel 3.38 Tabel plugins

No.	Nama Kolom	Tipe Data	Constrau=in	Keterangan
1.	Id	Int(11)	Primary key	
2.	Plugin_name	Varchar(100)	Foreign key	
3.	Plugin_url	Varchar(250)		
4.	Plugin_description	Text		
5.	Plugin_priority	Tinyint(4)		
6.	Plugin_active	Tinyint(4)		
7.	Plugin_installed	Tinyint(4)		

35. Tabel private_message

Tabel ini berfungsi untuk menyimpan data pesan yang dikirim oleh admin pada *User*. Struktur tabel dapat dilihat pada Tabel 3.39.

Tabel 3.39 Tabel private_message

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Parent_id	Int(11)		
3.	User_id	Int(11)	Foreign key	
4.	From_user_id	Int(11)		

5.	Private_subject	Varchar(255)		
6.	Private_message	Text		
7.	Private_message_date	Datetime		
8.	Private_message_new	Tinyint(4)		

36. Tabel rating

Tabel ini berfungsi untuk menyimpan data rating sistem. Struktur tabel dapat dilihat pada Tabel 3.40.

Tabel 3.40 Tabel rating

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	User_id	Int(11)	Foreign key	
3.	Incident_id	Bigint(20)	Foreign key	
4.	Comment_id	Bigint(20)	Foreign key	
5.	Rating	Tinyint(4)		
6.	Rating_ip	Varchar(100)		
7.	Rating_date	datetime		

37. Tabel reporter

Tabel ini berfungsi untuk menyimpan data pelaporan setiap *User*. Struktur tabel dapat dilihat pada Tabel 3.41.

Tabel 3.41 Tabel reporter

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Location_id	Bigint(20)	Foreign key	
3.	User_id	Int(11)	Foreign key	
4.	Service_id	Int(10)	Foreign key	
5.	Level_id	Int(11)	Foreign key	
6.	Service_account	Varchar(255)		
7.	Reporter_first	Varchar(200)		
8.	Reporter_last	Varchar(200)		
9.	Reporter_email	Varchar(120)		
10.	Reporter_phone	Varchar(60)		

11.	Reporter_ip	Varchar(50)		
12.	Reporter_date	datetime		

38. Tabel roles

Tabel ini berfungsi untuk menyimpan data roles yang digunakan *User*. Struktur tabel dapat dilihat pada Tabel 3.42.

Tabel 3.42 Tabel roles

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Name	Varchar(32)	Foreign key	
3.	Description	Varchar(255)		
4.	Access_level	Tinyint(4)		

39. Tabel roles_users

Tabel ini berfungsi untuk menyimpan roles *user*. Struktur tabel dapat dilihat pada Tabel 3.43.

Tabel 3.43 Tabel roles_users

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	User_id	Int(11)	Primary key	
2.	Role_id	Int(11)	Foreign key	

40. Tabel scheduler

Tabel ini berfungsi untuk menyimpan data *scheduler*. Struktur tabel dapat dilihat pada Tabel 3.44.

Tabel 3.44 Tabel scheduler

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(10)	Primary key	
2.	Scheduler_name	Varchar(100)		
3.	Scheduler_last	Int(10)		
4.	Scheduler_weekday	Smallint(6)		
5.	Scheduler_day	Smallint(6)		
6.	Scheduler_hour	Smallint(6)		
7.	Scheduler_minute	Smallint(6)		

8.	Scheduler_controller	Varchar(100)		
9.	Scheduler_active	Tinyint(4)		

41. Tabel scheduler_log

Tabel ini untuk menyimpan data scheduler yang pernah dibuat. Struktur tabel dapat dilihat pada Tabel 3.45.

Tabel 3.45 Tabel scheduler_log

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Scheduler_id	Int(10)	Foreign key	
3.	Scheduler_status	Varchar(20)		
4.	Scheduler_date	Int(10)		

42. Tabel service

Tabel ini berfungsi untuk menyimpan data service sistem. Struktur tabel dapat dilihat pada Tabel 3.46.

Tabel 3.46 Tabel service

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(10)	Primary key	
2.	Service_name	Varchar(100)		
3.	Service_description	Varchar(255)		
4.	Service_url	Varchar(255)		
5.	Service_api	Varchar(255)		

43. Tabel sessions

Tabel ini berfungsi untuk menyimpan data sesi masuk pengguna terhadap sistem. Struktur tabel dapat dilihat pada Tabel 3.47.

Tabel 3.47 Tabel sessions

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Session_id	Varchar(127)	Primary key	
2.	Last_activity	Int(10)		
3.	data	text		

44. Tabel settings

Tabel ini berfungsi untuk menyimpan data pengaturan sistem. Struktur tabel dapat dilihat pada Tabel 3.48.

Tabel 3.48 Tabel settings

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Key	Varchar(100)	Foreign key	
3.	value	text		

45. Tabel sharing

Tabel ini berfungsi untuk menyimpan data sharing *User*. Struktur tabel dapat dilihat pada Tabel 3.49.

Tabel 3.49 Tabel sharing

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(10)	Primary key	
2.	Sharing_name	Varchar(150)		
3.	Sharing_url	Varchar(255)		
4.	Sharing_color	Varchar(20)		
5.	Sharing_active	Tinyint(4)		
6.	Sharing_date	datetime		

46. Tabel sharing_incident

Tabel ini berfungsi untuk menyimpan data kejadian yang sudah di sharing. Struktur tabel dapat dilihat pada Tabel 3.50.

Tabel 3.50 Tabel sharing_incident

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(10)	Primary key	
2.	Sharing_id	Int(10)		
3.	Incident_id	Int(10)		
4.	Incident_title	Varchar(255)		
5.	Latitude	Double		
6.	Longitude	Double		
7.	Incident_date	datetime		

47. Tabel sharing_log

Tabel ini berfungsi untuk menyimpan log dari data kejadian yang sudah di sharing. Struktur tabel dapat dilihat pada Tabel 3.51.

Tabel 3.51 Tabel sharing_log

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(10)	Primary key	
2.	Sharing_id	Int(11)		
3.	Sharing_log_date	Int(10)		

48. Tabel smssync_message

Tabel ini berfungsi untuk menyimpan url aplikasi pihak kedua yang digunakan untuk menerima pelaporan lewat sms. Struktur tabel dapat dilihat pada Tabel 3.52.

Tabel 3.52 Tabel smssync_message

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Smssync_to	Varchar(100)		
3.	Smssync_from	Varchar(100)		
4.	Smssync_message	Text		
5.	Smssync_message_date	Datetime		
6.	Smssync_sent	Tinyint(4)		
7.	Smssync_sent_date	datetime		

49. Tabel smssync_settings

Tabe ini berfungsi untuk menyimpan pengaturan smsync. Struktur tabel dapat dilihat pada Tabel 3.53.

Tabel 3.53 Tabel smssync_settings

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Smssync_secret	Varchar(100)		

50. Tabel **socialmedia_asset**

Tabel ini berfungsi untuk menyimpan data pengaturan sosial media yang digunakan. Struktur tabel dapat dilihat pada Tabel 3.54.

Tabel 3.54 Tabel **socialmedia_asset**

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Type	Varchar(45)		
3.	url	Text		
4.	Message_id	Int(11)		

51. Tabel **socialmedia_keywords**

Tabel ini berfungsi untuk menyimpan keywords yang akan digunakan dalam pengaturan pelaporan melalui sosial media. Struktur tabel dapat dilihat pada Tabel 3.55.

Tabel 3.55 Tabel **socialmedia_keywords**

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	id	Int(11)	Primary key	
2.	Keyword	Varchar(40)		
3.	disabled	Tinyint(1)		

52. Tabel **socialmedia_messages_data**

Tabel ini berfungsi untuk menyimpan data pelaporan melalui sosial media. Struktur tabel dapat dilihat pada Tabel 3.56.

Tabel 3.56 Tabel **socialmedia_messages_data**

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Field_name	Varchar(45)		
3.	Value	Text		
4.	Message_id	Int(11)	Foreign key	

53. Tabel **socialmedia_settings**

Tabel ini berfungsi untuk menyimpan pengaturan penggunaan sosial media. Struktur tabel dapat dilihat pada Tabel 3.57.

Tabel 3.57 Tabel socialmedia_settings

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Setting	Varchar(40)	Foreign key	
3.	value	text		

54. Tabel users

Tabel ini berfungsi untuk menyimpan data *User* yang terdaftar pada sistem. Struktur tabel dapat dilihat pada Tabel 3.58.

Tabel 3.58 Tabel users

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Int(11)	Primary key	
2.	Riverid	Varchar(128)		
3.	Name	Varchar(200)		
4.	Email	Varchar(127)	Foreign key	
5.	Username	Varchar(100)	Foreign key	
6.	Password	Cahr(50)		
7.	Logins	Int(10)		
8.	Last_logins	Int(10)		
9.	Notify	Tinyint(1)		
10.	Updated	Timestap		
11.	Color	Varchar(6)		
12.	Code	Varchar(30)		
13.	Confirmed	Tinyint(1)		
14.	Public_profile	Tinyint(1)		
15.	Approved	Tinyint(1)		
16.	Needinfo	Tinyint(1)		

55. Tabel user_tokens

Tabel ini berfungsi untuk menyimpan data tokens yang digunakan *User*. Struktur tabel dapat dilihat pada Tabel 3.59.

Tabel 3.59 Tabel user_tokens

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
-----	------------	-----------	-----------	------------

1.	Id	Int(11)	Primary key	
2.	User_id	Int(11)	Foreign key	
3.	User_agent	Varchar(40)		
4.	Token	Varchar(64)	Foreign key	
5.	Created	Int(10)		
6.	expires	Int(10)		

56. Tabel verified

Tabel ini berfungsi untuk menyimpan data pelaporan yang sudah diverifikasi oleh admin. Struktur tabel dapat dilihat pada Tabel 3.60.

Tabel 3.60 Tabel verified

No.	Nama Kolom	Tipe Data	Constrain	Keterangan
1.	Id	Bigint(20)	Primary key	
2.	Incident_id	Bigint(20)	Foreign key	
3.	User_id	Int(11)		
4.	Verified_id	Datetime		
5.	Verified_status	Tinyint(4)		

B. Perancangan Struktur Basisdata Aplikasi Android

Pada perancangan struktur tabel basisdata yang akan digunakan pada aplikasi android secara keseluruhan menggunakan struktur basisdata yang sama dengan yang dipakai pada *web* aplikasi, namun karena menu yang diimplementasikan pada aplikasi android mengalami efisiensi maka penggunaan basisdata yang digunakan lebih sederhana, diantaranya sebagai berikut:

1. Tabel category.
2. Tabel incident.
3. Tabel location.
4. Tabel country.
5. Tabel incident_category.
6. Tabel incident_person.

3.3.4 Perancangan Relasi Tabel Basisdata Web Aplikasi

Relasi antar tabel dalam database mempunyai pengertian hubungan antara tabel yang satu dengan yang lain pada database, seperti hubungan dengan jenis “Many to One”, “One to Many” atau “One to One”.

Relasi ini sangat penting karena dapat meminimalisir adanya kesalahan dalam mendesain database. Untuk relasi tabel ada pada bagian “LAMPIRAN” laporan.

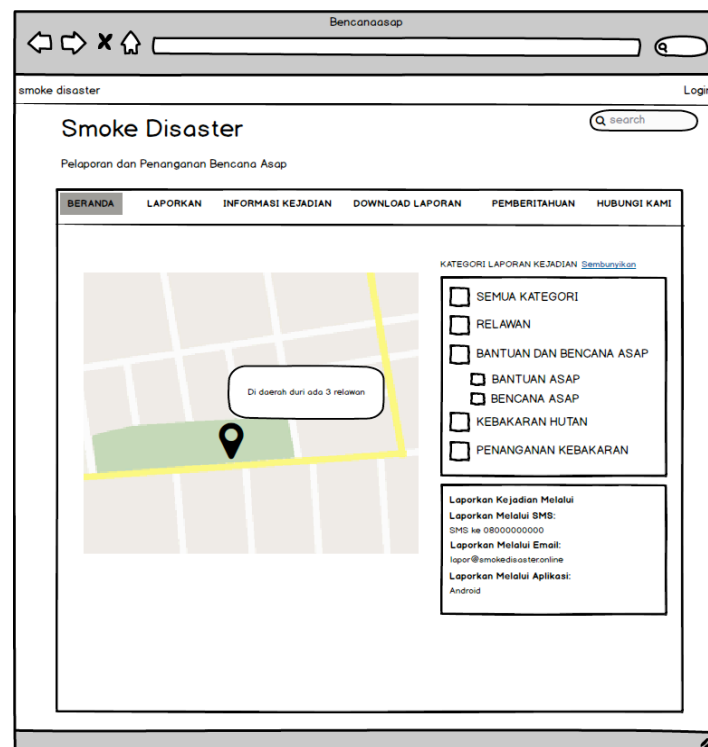
3.3.5 Perancangan Antar Muka

Interface atau antarmuka merupakan salah satu hal penting yang dapat mempermudah pengguna untuk berinteraksi dengan sistem. Pengguna akan mendapatkan informasi yang diinginkan pengguna dengan mudah sesuai dengan *input* interaksi yang diberikan pengguna ke dalam sistem. Untuk perancangan antarmuka sistem dibagi menjadi 2 yaitu perancangan, yaitu perancangan antarmuka *web* aplikasi dan perancangan antarmuka aplikasi android.

A. Perancangan Antarmuka Web Aplikasi

1. Halaman Beranda

Halaman antar muka utama ketika pertama kali situs dibuka. Dalam halaman beranda ini menampilkan semua menu utama yang ada pada web aplikasi. Rancangan halaman beranda dapat dilihat pada Gambar 3.20.



Gambar 3.17 Halaman Beranda

2. Halaman Laporkan

Merupakan antarmuka halaman yang digunakan oleh pengguna untuk memmbuat laporan kejadian seputar bencana asap, permintaan bantaun, dan informasi lokasi kebakaran hutan. Rancangan antarmuka dari halamn ini dapat dilihat pada Gambar 3.21.

Gambar 3.18 Halaman Laporkan

3. Halaman Daftar Informasi Kejadian

Merupakan antarmuka halaman yang digunakan oleh *User* untuk melihat daftar laporan kejadian seputar bencana asap yang masuk dan sudah di verifikasi oleh Admin. Rancangan antarmuka dapat dilihat pada Gambar 3.22.

Gambar 3.19 Halaman Daftar Informasi Kejadian..

4. Halaman Hubungi Kami.

Merupakan antarmuka halaman yang digunakan oleh pengguna untuk mengirim pesan atau kritik tentang sistem informasi ini. Rancangan antarmuka dapat dilihat pada Gambar 3.23.

Gambar 3.20 Halaman Hubungi Kami

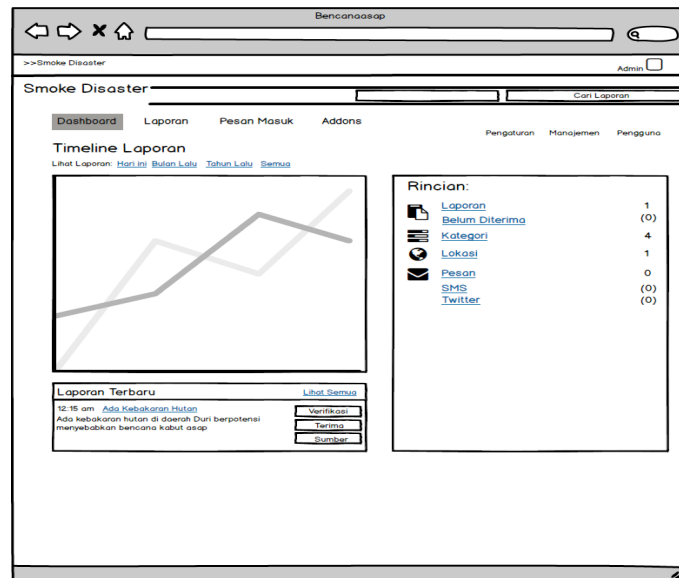
5. Halaman Pemberitahuan

Merupakan antarmuka halaman yang digunakan oleh *User* untuk mendapatkan pemberitahuan laporan masuk sesuai dengan kategori yang diinginkan. Rancangan antarmuka dapat dilihat pada Gambar 3.24.

Gambar 3.21 Halaman Pemberitahuan

6. Halaman Admin

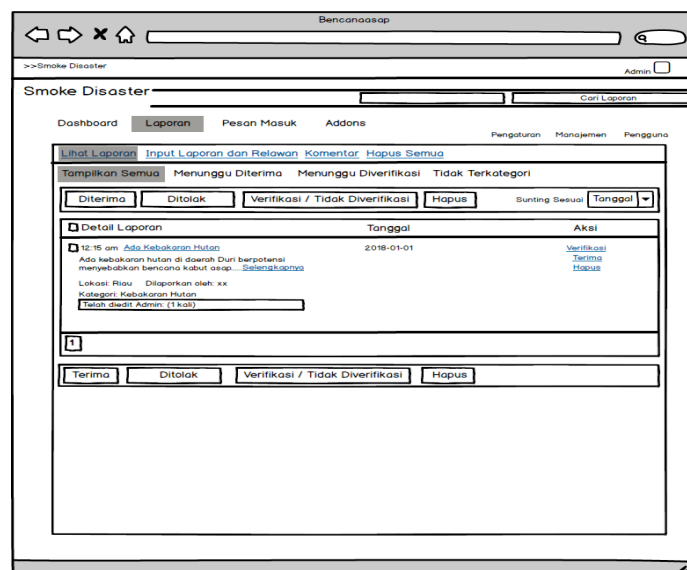
Merupakan antarmuka halaman utama yang digunakan oleh admin untuk mengedit laporan masuk, membuat laporan, mengelola pesan masuk menjadi laporan, manajemen pengguna dan lainnya. Rancangan antarmuka dapat dilihat pada Gambar 3.25.



Gambar 3.22 Halaman Dashboard Admin

7. Halaman Laporan Admin.

Merupakan antarmuka halaman laporan yang digunakan oleh admin untuk mengelola laporan yang dikirim oleh *User*. Pada halaman ini admin dapat menerima laporan untuk ditampilkan verifikasi laporan atau menghapus laporan. Rancangan antarmuka dapat dilihat pada Gambar 3.26.



Gambar 3.23 Halaman Laporan Admin

8. Halaman Edit dan Input Laporan Admin.

Merupakan antarmuka yang digunakan oleh admin untuk membuat laporan baru tentang kejadian bencana asap. Rancangan antarmuka dapat dilihat pada Gambar 3.27.

The screenshot shows a web browser window titled 'Bencanaasap'. The page header includes a search bar and a user profile icon. The main navigation bar has tabs for 'Dashboard', 'Laporan', 'Pesan Masuk', and 'Addons'. The 'Laporan' tab is active, showing a sub-menu with 'Lihat Laporan', 'Input Laporan dan Input Relawan', 'Komentar', and 'Hapus Semua'. The 'Input Laporan dan Input Relawan' sub-tab is selected, displaying a form titled 'Laporan Baru'. The form includes fields for 'Judul Laporan *', 'Deskripsi Laporan *', 'Tanggal dan Waktu' (set to 08:06 am on 11/09/2020), 'Kategori Laporan *' (with checkboxes for 'Kebakaran Hutan', 'Bantuan dan Bencana Asap', 'Penanganan Kebakaran', and 'Relawan'), 'Lokasi Kejadian*', 'Link Video', and 'Masukan Foto' (with a 'Choose File' button and a 'No File Selected' message). A 'Kirim Laporan' button is at the bottom right. The footer contains 'Informasi Pelapor' (Nama Depan, Email, Nama Belakang) and 'Evaluasi Informasi' (Tampilkan Laporan Ini, Verifikasi Laporan Ini) with 'Yes' and 'No' radio buttons.

Gambar 3.24 Halaman Input dan Edit Laporan Admin

9. Halaman Hapus Semua Laporan.

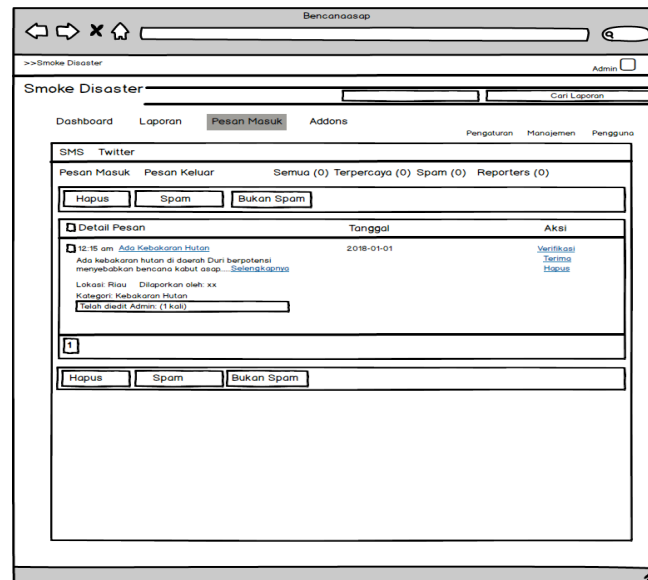
Merupakan antarmuka yang digunakan admin untuk menghapus semua laporan yang ada pada sistem, baik berasal dari laporan masuk ataupun pesan masuk via SMS atau email. Rancangan antarmuka dapat dilihat pada Gambar 3.28.

The screenshot shows the same web browser window as Gambar 3.24, but the 'Hapus Semua' sub-tab is selected. The main content area displays a large red 'HAPUS' button, followed by the text 'Hapus Semua Intruksi', 'Hapus Semua Backup', and a 'Hapus' button.

Gambar 3.25 Halaman Hapus Semua Laporan

10. Halaman Pesan Admin.

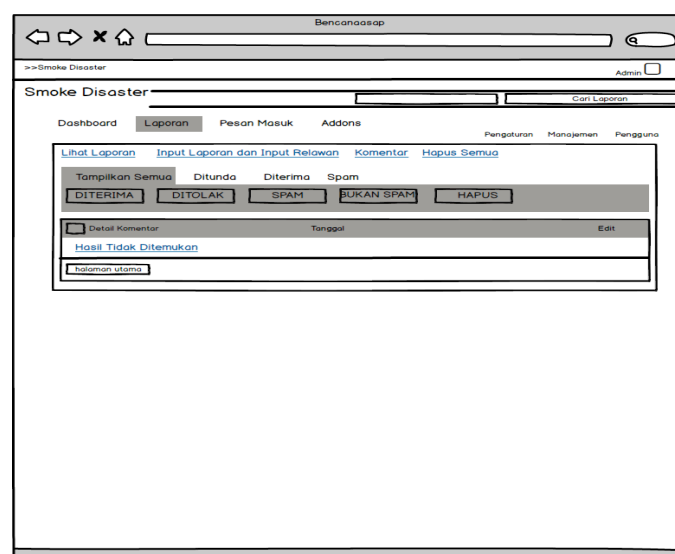
Merupakan antarmuka yang digunakan admin untuk mengelola pesan atau laporan masuk bersumber dari SMS ataupun Twitter untuk dijadikan laporan utama. Selain itu pada halaman ini juga admin dapat menghapus atau mengkategorikan pesan masuk yang dianggap tidak sesuai kriteria atau spam. Rancangan antarmuka dapat dilihat pada Gambar 3.29.



Gambar 3.26 Halaman Pesan Admin

11. Halaman Komentar Admin.

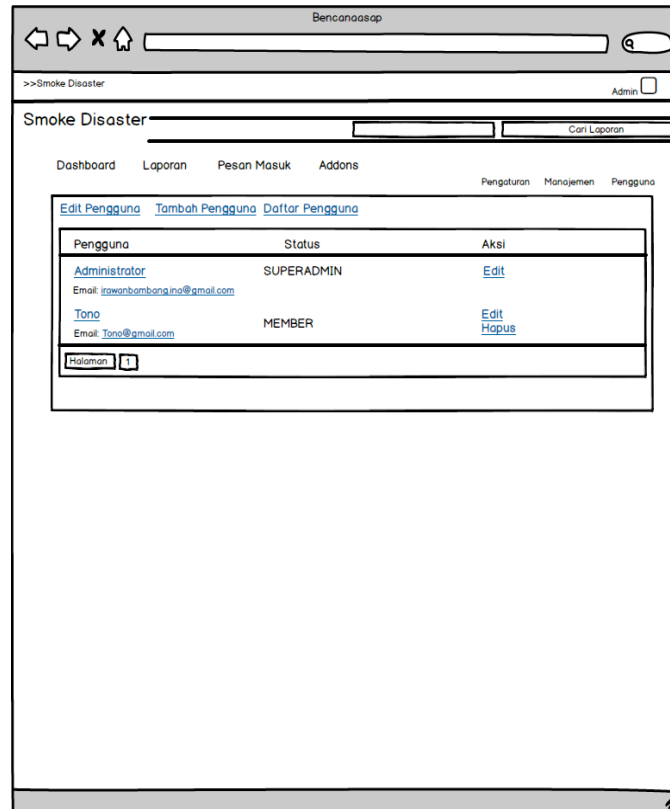
Merupakan antarmuka yang digunakan admin untuk melakukan filter terhadap komentar-komentar yang dikirim *User* sebelum ditampilkan pada sistem. Rancangan dapat dilihat pada Gambar 3.30.



Gambar 3.27 Halaman Pesan Admin

12. Halaman Mananajemen Pengguna

Merupakan antarmuka yang digunakan oleh super admin untuk mengatur atau melakukan manajemen *User* sistem. Super Admin dapat merubah status member menjadi admin, atau menambah dan menghapus *User* yang dipilih. Halaman ini hanya bisa diakses oleh super admin. Rancangan antarmuka dapat dilihat pada Gambar 3.31.



Gambar 3.28 Halaman Manajemen Pengguna

B. Perancangan Antar Muka Aplikasi Android

Berikut ini adalah perancangan antarmuka sistem informasi pengelolaan dan penanganan bencana asap pada aplikasi android. Perancangan antarmuka sistem informasi pada aplikasi android terdiri dari beberapa rancangan, yaitu :

1. Halaman Awal

Halaman antar muka utama ketika pertama kali aplikasi dibuka. Dalam halaman ini menampilkan logo dan nama aplikasi. Rancangan dapat dilihat pada Gambar 3.32.



Gambar 3.29 Halaman Awal Aplikasi Android

2. Halaman Utama

Halaman utama aplikasi ini dapat diakses tanpa atau dengan login, fitur-fitur yang ada pada aplikasi juga akan tampil pada saat halaman ini diakses. Pada halaman utama, *User* dapat melakukan input laporan kejadian dan melihat daftar laporan. Rancangan halaman ini dapat dilihat pada Gambar 3.33.



Gambar 3.30 Halaman Utama Aplikasi Android

3. Halaman Input Laporan

Halaman ini berfungsi untuk melaporkan dan memberikan informasi kejadian bencana asap oleh *user*. Rancangan halaman ini dapat dilihat pada Gambar 2.34.



Gambar 3.31 Halaman Input Laporan

4. Halaman Notifikasi Sukses Input Laporan

Halaman ini berfungsi untuk memberikan notifikasi sukses ketika input kejadian bencana asap oleh *user*. Rancangan halaman ini dapat dilihat pada Gambar 2.35.



Gambar 3.32 Rancangan Halaman Notifikasi Sukses Input Laporan

5. Halaman Lihat Daftar Laporan

Halaman ini berfungsi untuk melihat daftar kejadian dan informasi tentang bencana asap. Rancangan halaman ini dapat dilihat pada Gambar 3.36.



Gambar 3.33 Halaman Lihat Daftar Laporan

6. Halaman Deskripsi Daftar Laporan

Halaman ini berfungsi untuk melihat deskripsi daftar kejadian dan informasi tentang bencana asap. Rancangan halaman ini dapat dilihat pada Gambar 3.37.



Gambar 3.34 Halaman Lihat Daftar Laporan

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Implementasi sistem menjelaskan tentang hasil yang diperoleh dari tahapan sebelumnya yaitu tahap analisis dan perancangan. Pada tahap implementasi sistem dijelaskan mengenai sistem yang telah dirancang bekerja dalam bentuk *prototype*. Implementasi sistem ini sendiri dibentuk menjadi dua bagian, yaitu implementasi pada *prototype web*-aplikasi dan implementasi pada *prototype* aplikasi *mobile* berbasis android.

Setelah tahapan implementasi selesai, tahapan selanjutnya akan dilakukan pengujian, pengujian merupakan tahapan akhir sebelum *prototype* sistem benar-benar digunakan sesuai manfaat dan untuk keperluan penyelesaian masalah serta tujuan yang sudah ditentukan. Pengujian juga dilakukan untuk mengetahui apakah masih terdapat malfungsi atau *error* dari kinerja keseluruhan komponen yang terdapat pada *prototype* sistem.

Pengujian sistem dilakukan dengan melakukan uji coba *prototype* dengan cara melakukan simulasi pelaporan suatu kejadian tentang bencana asap atau kebakaran hutan oleh beberapa *User*. Setelah uji coba *prototype* sistem dilakukan, penulis melakukan uji efektivitas sistem sebagai tahapan akhir dari keseluruhan tahapan yang dilakukan dalam pembangunan sistem informasi pengelolaan pelaporan dan penanganan bencana asap berbasis *citizen science*.

Uji efektivitas dilakukan untuk mendapatkan parameter ukur dari tingkat efektivitas sistem dalam skala persen serta mampu menjawab rumusan masalah dan mencapai tujuan dari pembangunannya. Metode uji efektivitas ini dilakukan dengan menggunakan metode wawancara terhadap salah satu narasumber yang kredibel dan menyebarkan kuesioner terhadap beberapa responden yang telah melakukan uji coba dan simulasi pelaporan menggunakan *prototype* sistem.

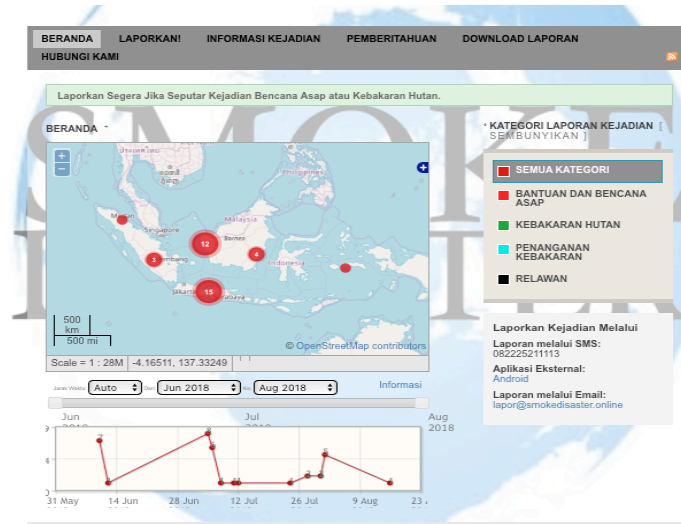
4.1 Implementasi

Implementasi merupakan bagian dari hasil nyata sistem. Pada bagian implementasi ini terdapat dua bagian implementasi, yaitu; 1. Implementasi *Web*-Aplikasi, 2. Implementasi *mobile* aplikasi berbasis android.

4.1.1 Implementasi *Web-Aplikasi*

A. Impelementasi Halaman Beranda

Halaman ini merupakan halaman pertama yang akan tampil ketika *web-aplikasi* diakses, pada halaman ini juga menu-menu utama pada *web-aplikasi* ditampilkan dan dapat digunakan oleh *User*. Implementasi halaman beranda dapat dilihat pada Gambar 4.1.



Gambar 4.1 Impelementasi Halaman Beranda

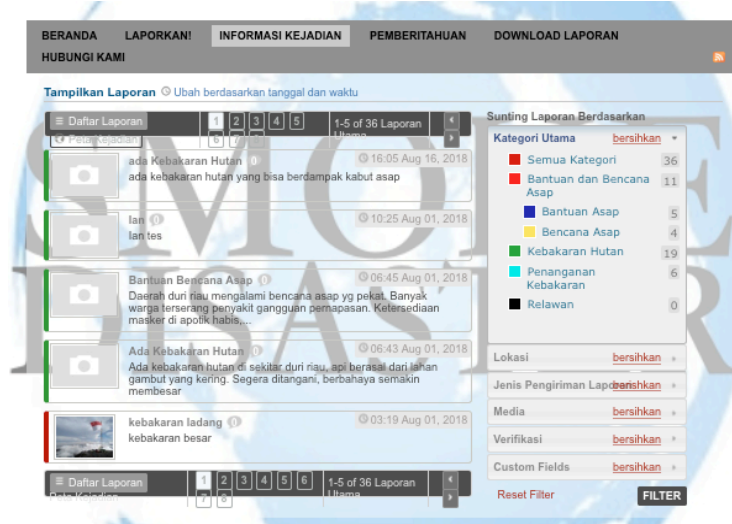
B. Implementasi Halaman Laporkan

Halaman ini berfungsi untuk melakukan pelaporan kejadian oleh *User*. Halaman ini akan menampilkan form yang harus diisi oleh *User*, form yang harus diisi diantaranya Judul, Deskripsi Kejadian, Lokasi dan Waktu Kejadian, Kategori Kejadian, dan Data Pelapor. Implementasi Halaman Laporkan dapat dilihat pada Gambar 4.2

Gambar 4.2 Impelementasi Halaman Laporkan

C. Implementasi Halaman Informasi Kejadian

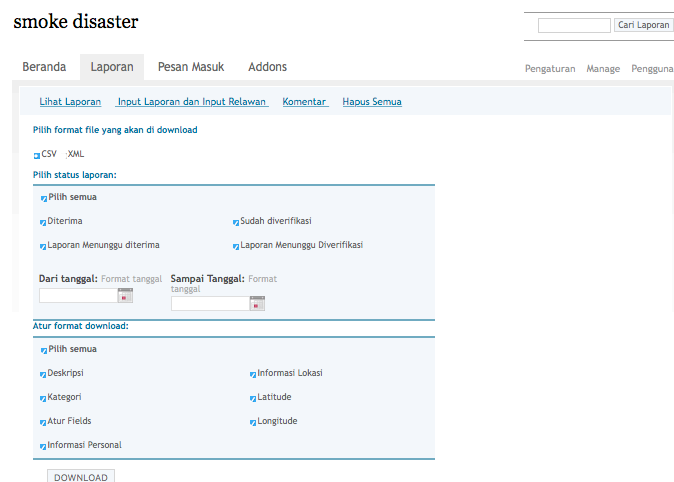
Halaman ini berfungsi untuk menampilkan laporan-laporan yang masuk. Pada halaman ini laporan yang ditampilkan hanya laporan-laporan yang telah disetujui untuk ditampilkan oleh Admin sebelumnya. Implementasi halaman informasi kejadian dapat dilihat pada Gambar 4.3.



Gambar 4.3 Impelementasi Halaman Informasi Kejadian

D. Implementasi Halaman Download Laporan

Halaman ini berfungsi untuk mengunduh laporan – laporan yang telah dikirim oleh *User*. Laporan yang dapat diunduh tidak hanya laporan yang sudah diterima atau diverifikasi oleh Admin, maupun yang belum diterima atau belum diverifikasi. Implementasi halaman informasi kejadian dapat dilihat pada Gambar 4.3.



Gambar 4.4 Impelementasi Halaman Download Laporan

E. Implementasi Halaman Hubungi Kami

Halaman ini berfungsi untuk mengirimkan seputar pertanyaan, kritik, ataupun saran terhadap sistem oleh User. Pada halaman User cukup mengisi form yang telah disediakan oleh sistem. Implementasi halaman hubungi kami dapat dilihat pada Gambar 4.4.

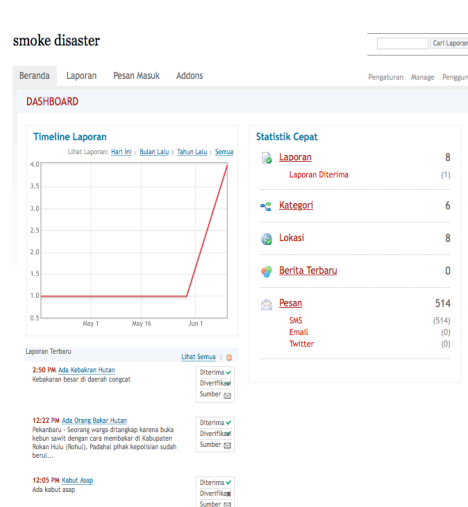
The screenshot shows a web interface for contacting the system. At the top, there is a navigation bar with links: BERANDA, LAPORKANI, INFORMASI KEJADIAN, PEMBERITAHUAN, and DOWNLOAD LAPORAN. Below this is a section titled 'HUBUNGI KAMI'. The form contains the following fields:

- Nama : (text input)
- Email : (text input)
- Telepon : (text input)
- Subjek Pesan : (text input)
- Deskripsi Pesan : (large text area)
- Masukan Kode : 19 + 9 = (CAPTCHA input)
- Kirim Pesan (green button)

Gambar 4.5 Impelementasi Halaman Hubungi Kami

F. Implementasi Halaman *Dashboard Admin*

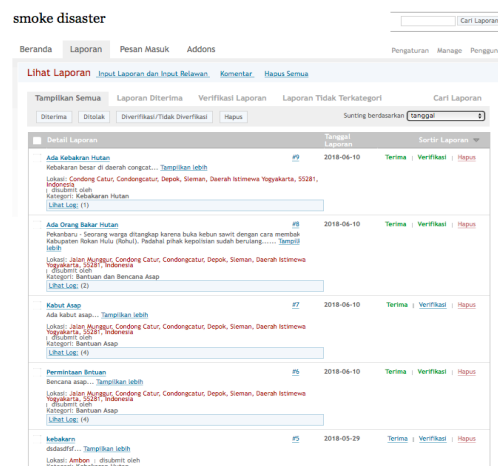
Halaman ini menjadi halaman yang pertama tampil ketika *Admin* sukses melakukan *login*. Pada halaman ini juga menu-menu utama yang dapat diakses oleh *Admin* ditampilkan, selain itu pada halaman ini juga akan ditampilkan grafik tentang aktivitas sistem. Implementasi halaman *dashboard admin* dapat dilihat pada Gambar 4.6.



Gambar 4.6 Impelementasi Halaman Dashboard *Admin*

G. Implementasi Halaman Laporan Masuk *Admin*

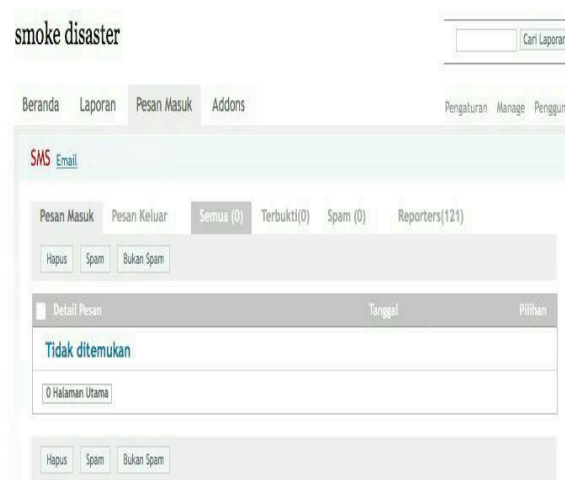
Halaman ini berfungsi untuk mengelola laporan yang masuk ke dalam sistem. Laporan-laporan tersebut dipilih untuk ditampilkan atau diverifikasi oleh *Admin*. Pada halaman ini juga *Admin* juga dapat melakukan perubahan laporan sesuai dengan yang diperlukan atau menghapus laporan. Implementasi halaman laporan masuk admin dapat dilihat pada Gambar 4.7.



Gambar 4.7 Impelementasi Halaman Laporan Masuk *Admin*

H. Implementasi Halaman Pesan Masuk *Admin*

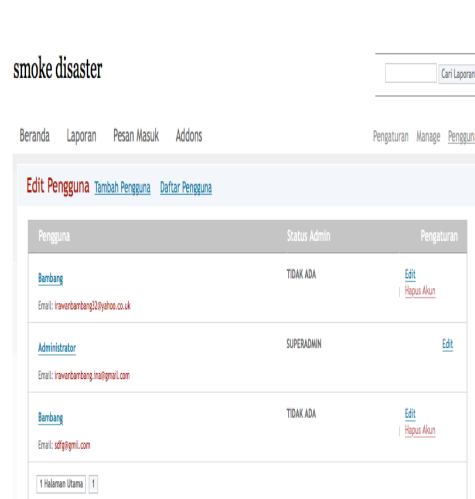
Halaman ini berfungsi untuk menampilkan laporan masuk yang bersumber dari SMS atau Email dan kemudian dikelola oleh *Admin* menjadi laporan. Implementasi halaman pesan masuk ini dapat dilihat pada Gambar 4.8



Gambar 4.8 Impelementasi Halaman Pesan Masuk *Admin*

I. Implementasi Halaman Manajemen Pengguna

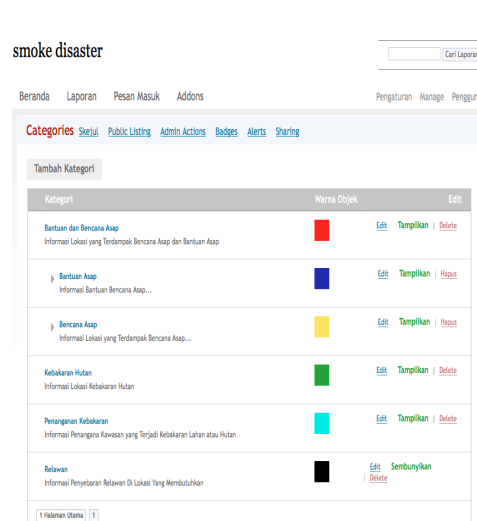
Halaman ini berfungsi untuk melakukan manajemen pada *User* yang terdaftar atau berstatus sebagai *Member*. Manajemen yang dilakukan berupa perubahan status *User* misalnya dari hanya *Member* menjadi *Admin* serta sebaliknya. Manajemen pengguna ini hanya dapat dilakukan oleh *Super Admin*. Implementasi halaman manajemen pengguna dapat dilihat pada Gambar 4.9.



Gambar 4.9 Impelementasi Halaman Manajemen Pengguna

J. Implementasi Halaman Manajemen Web-Aplikasi

Halaman ini berfungsi untuk melakukan perubahan-perubahan atau pengaturan yang diperlukan terhadap sistem atau *web-aplikasi* oleh *Admin*. Salah satu perubahan yaitu manajemen kategori. Implementasi halaman manajemen *web-aplikasi* ini dapat dilihat pada Gambar 4.10.



Gambar 4.10 Impelementasi Halaman Manajemen *Web-Aplikasi*

4.1.2 Implementasi Aplikasi Android

Implementasi pada aplikasi *smartphone* berbasis sistem operasi android ini mengalami efisiensi pada penggunaan menu, hal ini dilakukan demi menunjang kemudahan kepada *User* dalam proses pelaporan, dari keseluruhan menu yang digunakan pada *web*-aplikasi, hanya menu “Input Laporan” dan “Daftar Laporan” yang digunakan. Sifat “*mobile*” pada aplikasi android ini tentunya menjadi pertimbangan penulis dalam melakukan efisiensi penggunaa menu pada aplikasi android. Proses pelaporan yang mungkin terjadi mendadak atau serba cepat karena kejadian bencana asap tidak dapat terprediksi memungkinkan frekuensi penggunaan menu “Input Laporan” dan menu “Daftar Kejadian” akan jauh lebih sering diakses atau digunakan *User*.

A. Implementasi Halaman *Loading* Aplikasi Android

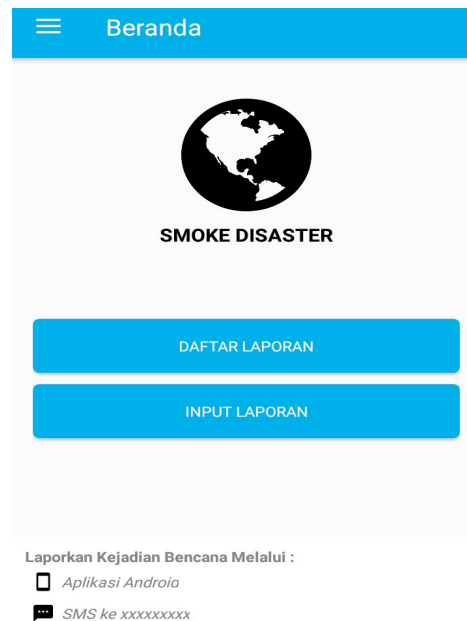
Halaman ini merupan tampilan *loading* saat aplikasi android dibuka atau dijalankan. Implementasi halaman *loading* aplikasi android dapat dilihat pada Gambar 4.11.



Gambar 4.11 Impelementasi Halaman *Loading* Aplikasi Android

B. Implementasi Halaman Utama Aplikasi Android

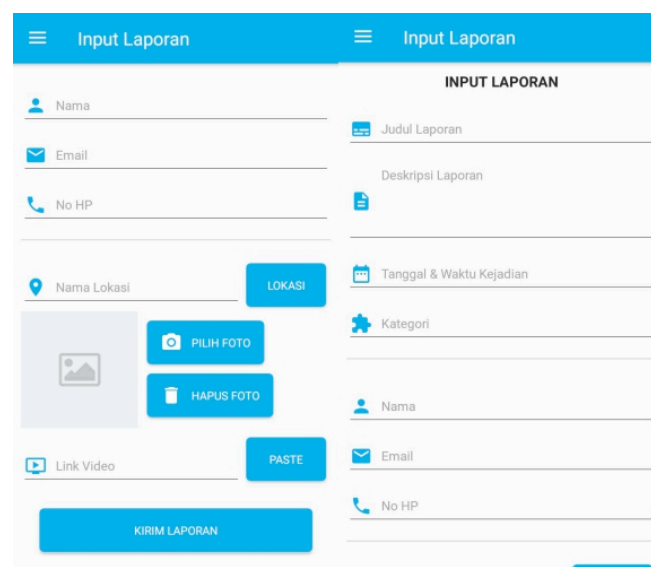
Pada halaman ini menu utama aplikasi akan ditampilkan dan dapat diakses oleh *User*. menu utama yang ditampilkan adalah menu “Input Laporan” dan “Daftar Laporan”. Implementasi halaman utama aplikasi android dapat dilihat pada Gambar 4.12.



Gambar 4.12 Impelementasi Halaman Utama Aplikasi Android

C. Implementasi Halaman Input Laporan Aplikasi Android

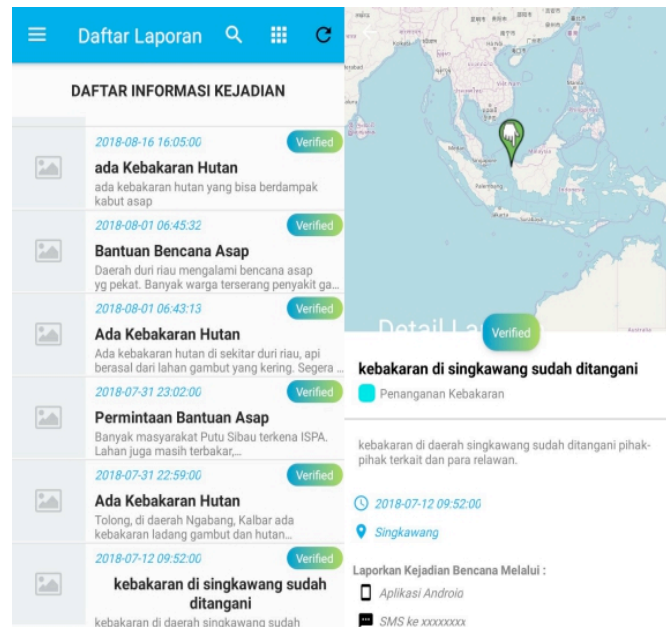
Halaman ini berfungsi untuk melakukan pelaporan melalui aplikasi android. Pada halaman aplikasi ini *User* akan mengisi *form* pelaporan yang disediakan seperti Judul, Deskripsi Kejadian, Tanggal dan Waktu Kejadian, Kategori Kejadian, Data Pelapor, Foto atau *Link* Video Kejadian, dan Lokasi Kejadian. Implementasi halaman ini dapat dilihat pada Gambar 4.13.



Gambar 4.13 Impelementasi Halaman Input Laporan Aplikasi Android

D. Implementasi Halaman Daftar Kejadian Aplikasi Android

Halaman ini berfungsi untuk menampilkan daftar dan deskripsi laporan-laporan yang masuk dan telah diverifikasi oleh *Admin*. Pada halaman ini juga *User* dapat melihat laporan sesuai dengan kategori yang dipilih. Implementasi halaman daftar kejadian dapat dilihat pada Gambar 4.14.



Gambar 4.14 Implementasi Halaman Daftar Laporan Aplikasi Android

4.2 Pengujian

4.2.1 Pengujian Validitas

Uji Validitas dilakukan untuk mengungkapkan kemampuan instrumen dalam mengukur kebenaran hasil penelitian secara cermat dan tepat, yang sekaligus membuktikan bahwa suatu instrumen dinyatakan salih/valid.

Uji Validitas ini dilakukan dengan mengambil 20 sampel dari pengguna Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science* sebagai sampel penelitian. Setelah diketahui bahwa kuesioner tersebut valid maka kuesioner tersebut dapat digunakan untuk penelitian. Dari hasil uji validitas terhadap butir – butir pertanyaan yang telah tercantum dalam kuesioner ternyata semua butir pertanyaan mengenai persepsi pengguna terhadap efektivitas sistem dinyatakan valid. Hasil uji validitas dapat dilihat pada Tabel 4.1.

Tabel 4.1 Hasil Uji Validitas Instrumen

Item	Koefisien Validitas	Keterangan
1	0.6528	Valid
2	0.6676	Valid
3	0.7212	Valid
4	0.5904	Valid

4.2.2 Pengujian Reliabilitas

Kualitas kuesioner tidak hanya ditentukan oleh kemampuan item – itemnya dalam mengukur atau mengungkapkan pendapat subyek, namun juga harus mampu memberikan konsistensi (keandalan) maksud, sehingga dapat menghindari terjadinya pemaknaan ganda oleh sesama subyek pada kuesioner. Evaluasi terhadap keandalan kuesioner dilakukan dengan uji reabilitas, hasil uji menemukan kuesioner yang digunakan memiliki koefisien keandalan (Sperman Brown) sebesar 0.7713. Nilai ini lebih besar dari r-tabel sebesar 0.135 sehingga dinyatakan reliabel.

4.2.3 Hasil Deskripsi Efektivitas Sistem Oleh Pengguna

Untuk mengetahui efektivitas sistem menurut pengguna umum dapat diketahui melalui pengujian statistik diskriptif. Pengujian ini memberitahukan skor penilaian dalam bentuk rata – rata maupun per subyek, tinggi rendahnya skor penilaian ini merupakan persepsi pengguna secara umum terhadap efektif atau tidaknya Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science*.

Nilai rata – rata persepsi pengguna terhadap pertanyaan menggunakan skala likert terkategori (Sugiyono, 1999:98) dapat dilihat pada Tabel 4.2 :

Tabel 4.2 Nilai Persepsi

No	Persepsi	Skor
1	Sangat Efektif	5
2	Efektif	4
3	Netral	3
4	Tidak Efektif	2
5	Sangat Tidak Efektif	1

Dengan demikian skor tertinggi akan tercapai bila semua pertanyaan ditanggapi setuju atau diberi skor 5 oleh responden, sehingga diperoleh skor tertinggi sebesar $5 \times 20 = 100$, sebaliknya skor terendah diberi skor 1 oleh responden, sehingga diperoleh skor $1 \times 20 = 20$. Dalam rentang skor 20 (terendah) sampai dengan 100 (tertinggi) sapat ditentukan interval untuk membuat 5 kategorisasi penelitian sebagai berikut :

$$\begin{aligned}\text{Interval} &= (100 - 20) / 5 \\ &= 16\end{aligned}$$

$$\text{Interval rata - rata} = 16/20 = 0.8$$

Sehingga diperoleh katagorisasi :

Interval	Katagori
1.0 – 1.8	Sangat Tidak Efektif
➤ 1.8 – 2.6	Tidak Efektif
➤ 2.6 – 3.4	Netral
➤ 3.4 – 4.2	Efektif
➤ 4.2 – 5.0	Sangat Efektif

Data skor dan hasil perhitungan data kuesioner tentang persepsi pengguna terhadap efektivitas Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science* dari sejumlah 20 responden diperoleh skor maksimum 5.0, skor minimum 1.0 dan skor rata – rata 3.8350 dari 4 butir pertanyaan kesioner. Data kuesioner ini dapat dideskripsikan dalam bentuk distribusi frekuensi skor seperti pada Tabel 4.2.

Tabel 4.3 Distribusi Frekuensi Skor Persepsi Pengguna Terhadap Efektivitas Sistem

No	Kategori	Interval Nilai	Frekuensi	Valid Percent
1	STE	1.0 – 1.8	0	0
2	TE	1.8 – 2.6	1	5.0
3	N	2.6 – 3.4	2	10.0
4	E	3.4 – 4.2	9	45.0
5	SE	4.2 – 5.0	8	40.0
Hasil			100	100

Berdasarkan tabel diatas, diperoleh data bahwa, terdapat 8 orang atau (40%) responden memiliki persepsi terhadap efektivitas Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap “sangat efektif”, dan terdapat 9 orang atau (45%) responden memiliki persepsi “efektif”, dan terdapat 2 orang memiliki persepsi “netral”, dan 1 orang atau (5%) memiliki persepsi tidak efektif;

Berdasarkan distribusi penelitian ini berarti mayoritas konsumen menilai bahwa Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science* sudah efektif.

Tabel 4.4 Jawab Responden Terhadap Efektivitas Sistem

Item	SE	E	N	TE	STE
	Jumlah	Jumlah	Jumlah	Jumlah	Jumlah
1	4	14	2	0	0
2	6	12	2	0	0
3	7	10	3	0	0
4	6	7	5	1	0

Berdasarkan pertanyaan no.1 sampai no.4 yaitu tentang efektivitas Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science* menurut persepsi pengguna diperoleh jawaban responden sebagai berikut:

1. Untuk pertanyaan no.1, (Apakah dengan adanya aplikasi ini, pelaporan kejadian bencana asap dapat lebih mudah dilakukan) dari 20 responden diperoleh 4 responden memberikan jawaban “sangat efektif”, 14 responden memberikan jawaban “efektif” dan 2 responden memberikan jawaban “netral”.
2. Untuk pertanyaan no.2, (Apakah aplikasi ini dirasa efektif dalam pengelolaan pelaporan bencana asap dan kebakaran hutan.) dari 20 responden diperoleh 6 responden memberikan jawaban “sangat efektif”, 12 responden memberikan jawaban “efektif” dan 2 responden memberikan jawaban “netral”.
3. Untuk pertanyaan no.3, (Apakah aplikasi ini dapat membantu pihak-pihak terkait dan para relawan agar dapat menyebar memberikan bantuan atau penanganan ketika terjadi bencana asap maupun kebakaran hutan.) dari 20 responden diperoleh 7 responden

memberikan jawaban “sangat efektif”, 10 responden memberikan jawaban “efektif” dan 3 responden memberikan jawaban “netral”.

4. Untuk pertanyaan no.4, (Apakah dengan adanya aplikasi ini akan membantu pihak-pihak yang terkait untuk segera memberikan bantuan yang tepat dan cepat terhadap daerah dan korban yang terdampak bencana asap.) dari 20 responden diperoleh 6 responden memberikan jawaban “sangat efektif”, 6 responden memberikan jawaban “efektif”, 5 responden memberikan jawaban “netral” dan 1 respondedn memberikan jawaban tidak efektif.

Penulis menyebarkan kuesioner kepada 20 responden dengan rentang usia antara 19 tahun sampai 30 tahun. Analisis tabel responden dapat dilihat pada tabel 4.1.

Tabel 4.5 Tabel Responden

No.	Nama	Kota Asal	Pekerjaan
1.	Rizky Febriansyah	Cirebon	Mahasiswa
2.	Haryanus S. K	Pontianak	Mahasiswa
3.	Pramudanu	Pontianak	Mahasiswa
4.	Syafiul Hadi	Pontianak	Jurnalis
5.	Beny	Yogyakarta	Jurnalis
6.	Kholisudin	Sumenep	Swasta
7.	Arie Azhari D P	Pontianak	Mahasiswa
8.	Ricky D Pintabar	Pontianak	Mahasiswa
9.	Syukron Baharis	Madura	Mahasiswa
10.	Yanuardhi Mazid P	Yogyakarta	Mahasiswa
12.	Yudha Pratama	Pontianak	Swasta
13.	Adam	Solo	Jurnalis
14.	Jule	Solo	Jurnalis
15.	Jumali	Semarang	Jurnalis
16.	Fatah	Yogyakarta	Swasta
17.	Nizar Ahmad	Yogyakarta	Swasta
18.	Resti	Yogyakarta	Mahasiswa
19.	Imam Rianto	Banjar	Mahasiswa
20.	Ellanda Pratama	Yogyakarta	Mahasiswa

Pengujian dilakukan secara objektif, dimana responden melakukan uji coba *prototype* sistem berupa *web*-aplikasi dan aplikasi *mobile* berbasis android dengan melakukan simulasi pelaporan sebuah kejadian dalam ruang lingkup seputar bencana asap maupun kebakaran hutan. Setelah ujicoba dilakukan responden, penulis meminta responden untuk mengisi tanggapan dengan memberikan penilaian terhadap pertanyaan atau pernyataan yang sudah disediakan penulis dalam bentuk kuesioner. Detail daftar pernyataan atau pertanyaan pada kuesioner dapat dilihat pada Tabel 4.3.

Tabel 4.6 Hasil Kuesioner Pengukuran Efektivitas Sistem Informasi

NO	Pertanyaan / Pernyataan	Frekuensi					Skor
		STE	TE	N	E	SE	
1.	Apakah dengan adanya aplikasi ini, pelaporan kejadian bencana asap dapat lebih mudah dilakukan	0	0	2	14	4	76 %
2.	Apakah aplikasi ini dirasa efektif dalam pengelolaan pelaporan bencana asap dan kebakaran hutan.	0	0	2	12	6	80 %
3.	Apakah aplikasi ini dapat membantu pihak-pihak terkait dan para relawan agar dapat menyebar memberikan bantuan atau penanganan ketika terjadi bencana asap maupun kebakaran hutan.	0	0	4	11	6	80.00 %
4.	Apakah dengan adanya aplikasi ini akan membantu pihak-pihak yang terkait untuk segera memberikan bantuan yang tepat dan cepat terhadap daerah dan korban yang terdampak bencana asap.	0	1	6	8	5	80.00 %

Pada penelitian ini jika dilihat secara rata – rata sebagian responden memberikan tanggapan yang positif, yaitu ada 8 responden atau (40%) yang menyatakan bahwa sistem informasi ini “sangat efektif”, ada 9 responden atau (45%) yang menyatakan bahwa sistem informasi ini “efektif “ dan 2 responden atau (10%) menyatakan “netral”. Berdasarkan hasil tersebut, berarti mayoritas konsumen menilai bahwa Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science* sudah efektif.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil – hasil yang telah diuraikan dalam analisis data dan pembahasan, menunjukkan bahwa Sistem Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science* telah berhasil mencapai tujuannya dan menjawab rumusan masalah yaitu meningkatkan efektivitas pengelolaan pelaporan dan sebagai media pemusatan informasi bagi para relawan RTIK dan pihak – pihak terkait demi keperluan membantu menangani bencana asap secara efektif, efisien dan tepat sasaran dengan teknik sistem informasi terpusat.

Jika dilihat per-item pertanyaan dapat disimpulkan bahwa item ke-1 (Apakah dengan adanya aplikasi ini, pelaporan kejadian bencana asap dapat lebih mudah dilakukan) merupakan item yang paling efektif, yaitu 14 responden menyatakan efektif. Dan ada satu reponden menyatakan tidak efektif pada item ke-4 (Apakah dengan adanya aplikasi ini akan membantu pihak-pihak yang terkait untuk segera memberikan bantuan yang tepat dan cepat terhadap daerah dan korban yang terdampak bencana asap).

Secara keseluruhan, dari analisi statistik hasil diperoleh bawah persepsi konsumen terhadap efektifitas Informasi Pengelolaan Pelaporan Dan Penanganan Bencana Asap Berbasis *Citizen Science* sudah “efektif”.

5.2 Saran

Karena sistem ini dibangun dengan sifat mengembangkan, semoga kelak para *developer* lainnya diharapkan berkenan ikut serta dalam mengembangkan sistem ini menjadi lebih baik lagi. Serta harapan penulis semoga sistem ini kedepannya akan dibuat dalam bentuk aplikasi smartphone berbasis IOS dan tidak hanya tersedia dalam bentuk aplikasi *smarthphone* berbasis Android.

DAFTAR PUSTAKA

- Biro Komunikasi Dan Informasi Publik. (2012). Ancaman Kabut Asap Terhadap Keselamatan Transportasi. Retrieved August 7, 2018, from <http://dephub.go.id/post/read/ancaman-kabut-asap-terhadap-keselamatan-transportasi-14678>
- Dimas, S. (2012). Definisi Perencanaan. Retrieved May 6, 2018, from <http://definisimu.blogspot.com/2012/08/definisi-perencanaan.html>
- Isyana, A. (2015). Dampak ekonomi kabut asap 'lebih dari Rp20 triliun. Retrieved May 8, 2018, from http://www.bbc.com/indonesia/berita_indonesia/2015/09/150917_indonesia_kerugian_kabut_asap
- Maharani, D. (2015). Bahaya Kabut Asap Bagi Kesehatan. Retrieved May 7, 2018, from <https://lifestyle.kompas.com/read/2015/09/07/100657423/Bahaya.Kabut.Asap.Bagi.Kesehatan>
- Openstreep Map Indonesia. (2014). OpenStreetMap adalah sebuah proyek kolaboratif untuk membuat peta dunia gratis yang dapat diubah. Retrieved April 20, 2018, from <https://openstreetmap.id/about/tentang-openstreetmap/>
- Prasetya, H. (2010). Ushahidi Rebut Best Weblog. Retrieved June 9, 2018, from <http://www.dw.com/id/ushahidi-rebut-best-weblog-blogombal-terbaik-bahasa-indonesia/a-5474496>
- Priyambudi, H. (2017). Pengertian Metode Prototype, Tahapan dan Kelebihan Metode Prototype. Retrieved June 9, 2018, from <https://androidunik.com/pengetahuan/pengertian-metode-prototype-dan-kelebihannya/>
- Rahmatullah, A. (2016). Belajar UML – Use Case Diagram. Retrieved May 8, 2018, from <https://www.codepolitan.com/mengenal-uml-diagram-use-case>
- Scientize Project. (2014). Call for Contributors to Citizen Science Green Paper. Retrieved May 18, 2018, from <http://www.socientize.eu/?q=eu/content/call-contributors-citizen-science-green-paper>
- Simpson, R. (2013). Explainer: what is citizen science. Retrieved May 6, 2018, from <http://theconversation.com/explainer-what-is-citizen-science-16487>
- Sora, N. (2015). Pengertian UML dan Jenis-Jenisnya Serta Contoh Diagramnya. Retrieved March 20, 2018, from <http://www.pengertianku.net/2015/09/pengertian-uml-dan-jenis->

jenisnya- serta-contoh-diagramnya.html

- Ummiati, S. (2011). Penjelasan Antara SDLC, Prototype dan Waterfall. Retrieved May 15, 2018, from <https://sitiumiati.wordpress.com/2011/04/28/penjelasan-antara-sdlc-prototype-dan-waterfall/>
- Wanda, R. L. (2015). Pengertian Prototyping Model: Model Prototyping. Retrieved May 12, 2018, from <http://rizalloa.ilearning.me/?p=132>
- Wardana, M. (2015). Penanggulangan Dampak Bencana Kabut Asap. Retrieved March 20, 2018, from <http://nationalgeographic.co.id/berita/2015/10/penanggulangan-dampak-kabut-asap>
- Wixom, Barbara Harley, R. R. (2014). System Analys And Design. Retrieved May 1, 2018, from http://www.uoitc.edu.iq/images/documents/informatics-institute/Competitive_exam/Systemanalysisanddesign.pdf

LAMPIRAN

- A. Relasi Tabel Web-Aplikasi
- B. Makalah Karya Tulis

HALAMAN PENGESAHAN DOSEN PENGUJI

**SISTEM INFORMASI PENGELOLAAN PELAPORAN DAN
PENANGANAN BENCANA ASAP BERBASIS
CITIZEN SCIENCE
TUGAS AKHIR**

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk
memperoleh gelar Sarjana Teknik Informatika
di Fakultas Teknologi Industri Universitas Islam Indonesia
Yogyakarta, 1 Agustus 2018

Tim Penguji

Kholid Haryono, S.T, M.Kom.

Ketua

Rahadian Kurniawan, S.Kom., M.Kom.

Anggota 1

Galang P Mahardhika, S.Kom., M.Kom.

Anggota 2

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Teknologi Industri
Universitas Islam Indonesia



(Hendrik, S.T., M.Eng.)

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**SISTEM INFORMASI PENGELOLAAN PELAPORAN DAN
PENANGANAN BENCANA ASAP BERBASIS
*CITIZEN SCIENCE***



الجامعة الإسلامية
الاستاذة الأستاذة
الاستاذة الأستاذة

Yogyakarta, 1 Agustus 2018

Pembimbing,

(Kholid Haryono, S.T, M.Kom)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Bambang Irawan

NIM : 11523039

Tugas akhir dengan judul:

**SISTEM INFORMASI PENGELOLAAN PELAPORAN DAN
PENANGANAN BENCANA ASAP BERBASIS *CITIZEN*
*SCIENCE***

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 1 Agustus 2018



(Bambang Irawan)