

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kebakaran adalah api dalam skala besar yang tidak bisa dikendalikan sehingga diperlukan suatu alat pemadam kebakaran untuk memadamkan api. Kebakaran terjadi melalui proses kimiawi antara uap bahan bakar dengan oksigen dan bantuan panas. Teori ini dikenal sebagai segitiga api. Kebakaran dapat disebabkan oleh beberapa faktor, faktor yang secara sengaja maupun tidak disengaja. Faktor yang disebabkan secara sengaja seperti membakar hutan secara besar-besaran atau membakar rumah dengan bensin. Sedangkan faktor secara tidak sengaja yang dapat menyebabkan terjadinya kebakaran antara lain seperti korsleting listrik, sambaran petir, ledakan tabung gas *LPG*, dan lain-lain. Peristiwa kebakaran dapat menimbulkan kerugian dari segi terhadap kesehatan, aset, bahkan kebakaran dapat merenggut nyawa manusia.

Di Indonesia, pihak yang bertugas untuk memadamkan kebakaran disebut Petugas Pemadam Kebakaran atau Damkar. Pemadam kebakaran ini akan bertugas atau bekerja apabila ada laporan kebakaran yang diterima dari masyarakat. Dalam hal ini laporan masyarakat merupakan aspek penting dalam sistem kerja yang dimiliki oleh petugas pemadam kebakaran. Oleh karena itu masyarakat perlu difasilitasi agar dapat lebih efektif dan efisien dalam menyampaikan laporan mengenai kebakaran mengingat kecepatan penyampaian informasi sangat berpengaruh dalam keberhasilan petugas pemadam kebakaran dalam melakukan tugasnya untuk memadamkan api.

Mekanisme pelaporan kebakaran secara konvensional yang berjalan di lapangan saat ini masih kurang efektif dan optimal. Masyarakat biasanya melapor kepada pihak kepolisian atau menghubungi pihak Pemadam Kebakaran melalui kontak Pemadam Kebakaran yang tersimpan pada *smartphone* masyarakat. Selain itu, berdasarkan observasi yang telah dilakukan, ditemukan fakta di lapangan bahwa masyarakat cenderung menghubungi pihak Pemadam Kebakaran tanpa mengetahui apakah lokasi Pemadam Kebakaran tersebut dekat atau jauh dari lokasi kebakaran. Hal ini mengakibatkan pemadam kebakaran harus menempuh jarak yang cukup jauh untuk sampai ke lokasi kebakaran. Bentuk pelaporan

seperti ini tentunya akan membuat kinerja Pemadam Kebakaran menjadi kurang optimal.

Setelah melakukan analisa terhadap permasalahan tersebut ditemukan sebuah solusi yang dapat menyelesaikan permasalahan tersebut yaitu dengan merancang dan membangun sebuah aplikasi yang dapat mengoptimalkan proses pelaporan kebakaran dari masyarakat dengan menggunakan teknologi *geotagging* berbasis *Android* serta memanfaatkan layanan *Google Maps* untuk menentukan rute dari lokasi pemadam kebakaran menuju titik lokasi kebakaran.

Teknologi *geotagging* dipilih sebagai solusi karena *geotagging* sistem pelaporan yang dikembangkan pada aplikasi menggunakan sebuah foto. Di mana foto tersebut sudah disematkan informasi berupa data *geotag*. Pengguna hanya perlu mengambil foto kebakaran kemudian mengirimkan laporan tersebut. Dengan sistem ini dapat dipastikan bahwa waktu pelaporan dapat lebih dipersingkat.

Dengan adanya aplikasi pelaporan kebakaran ini, diharapkan masyarakat dapat melaporkan kebakaran sesuai titik koordinat kebakaran yang diambil dari foto *geotag* dan masyarakat dapat melaporkan kepada Pemadam Kebakaran yang terdekat dari lokasi terjadinya kebakaran. Aplikasi yang dibangun juga memanfaatkan layanan *Google Maps*, sehingga Pemadam Kebakaran dapat menentukan petunjuk arah secara otomatis berdasarkan lokasi kebakaran menggunakan layanan *Google Maps* agar Pemadam Kebakaran dapat mengetahui rute menuju lokasi kebakaran.

1.2 Perumusan Masalah

Informasi untuk mengembangkan pelaporan terdapat beberapa masalah yang harus diatasi antara lain mendeteksi lokasi foto yang dikirimkan oleh user ke pihak Pemadam Kebakaran. Dari informasi tersebut bisa dihasilkan koordinat yang nanti dimunculkan sebagai tujuan lokasi kebakaran yang menjadi rute petugas pemadam kebakaran mengarah ke titik lokasi kebakaran.

Untuk mengembangkan suatu laporan diperlukan suatu ide untuk membuat laporan secara konvensional di ubah menjadi digitalisasi berupa aplikasi. Dalam hal ini masyarakat memberikan laporan juga menyertakan sebuah

foto kebakaran, sehingga didapatlah koordinatnya untuk menemukan titik lokasi kebakaran.

Berdasarkan rumusan tersebut maka penelitian ini akan menghasilkan aplikasi pelaporan yang bisa mendeteksi posisi kebakaran dengan memanfaatkan fitur *geotag* dari sebuah foto dan pemadam kebakaran dapat memanfaatkan fitur *Google Maps* untuk menentukan rute pemadam kebakaran untuk menuju ke titik lokasi kebakaran.

1.3 Tujuan Penelitian

Tujuan dari penelitian ini adalah menghasilkan aplikasi yang memanfaatkan teknologi *geotagging* untuk menentukan lokasi titik koordinat serta memanfaatkan layanan *Google Maps* untuk menentukan rute pemadam kebakaran ke titik lokasi kebakaran dari sebuah laporan yang diterima.

1.4 Pembatasan Masalah

Pembatasan masalah pada penelitian ini adalah sebagai berikut.

1. Masyarakat memilih posisi Pemadam Kebakaran yang terdekat dari titik lokasi kebakaran.
2. Aplikasi ini menggunakan *geotag* dengan fitur kamera untuk menentukan titik koordinat secara *real time*.
3. Penentuan rute menggunakan layanan *Google Maps*.
4. Perhitungan jarak pelapor dengan Damkar menggunakan layanan *Google Maps* menggunakan *direction*.
5. Hanya menangani kebakaran di wilayah Kota Pontianak
6. Masyarakat yang memberikan laporan hanya masyarakat di sekitar lokasi kebakaran berdasarkan lokasi *GPS*.
7. Tidak menangani kebakaran hutan, hanya menangani kebakaran perumahan, dan gedung tinggi.

1.5 Sistematika Penulisan Skripsi

Adapun sistematika penulisan skripsi ini disusun dalam lima bab yang terdiri dari Bab I Pendahuluan, Bab II Tinjauan Pustaka, Bab III Metodologi

Penelitian dan Perancangan Aplikasi, Bab IV Hasil Perancangan dan Pengujian Aplikasi, serta Bab V Kesimpulan dan Saran.

Bab I Pendahuluan adalah bab yang berisi latar belakang, perumusan masalah, tujuan penelitian, pembatasan masalah, dan sistematika penulisan skripsi.

Bab II Tinjauan Pustaka adalah bab yang berisi landasan teori yang ada hubungannya dengan penelitian yang akan dilakukan dan prinsip-prinsip penunjang yang berguna dalam pengerjaan tugas akhir. Landasan teori yang berhubungan dengan penelitian ini tentang pelaporan, pemadam kebakaran, *android*, aplikasi, *Google Maps*, *UML* serta pengujian perangkat lunak.

Bab III Metodologi Penelitian dan Perancangan Aplikasi adalah bab yang membahas tentang perancangan aplikasi, berisi Bahan Penelitian, Alat yang Dipergunakan, Metode Penelitian, Variabel atau Daya, Diagram Alir Penelitian, Perancangan Aplikasi, Perancangan *UML*, Perancangan Antarmuka, serta Rencana Pengujian.

Bab IV Hasil Perancangan dan Pengujian Aplikasi adalah bab ini dijelaskan implementasi dari perangkat lunak yang dibangun. Implementasi perangkat lunak dilakukan berdasarkan kebutuhan perancangan perangkat lunak yang sudah dilakukan. Dari hasil implementasi kemudian dilakukan pengujian sistem berdasarkan pada analisis kebutuhan perangkat lunak yang menjelaskan apakah sudah benar-benar sesuai dengan analisis dan perancangan yang telah dilakukan.

Bab V Kesimpulan dan Saran adalah bab yang berisi kesimpulan dari penelitian yang telah dilakukan dan saran/rekomendasi untuk penulis berikutnya dalam pengembangan penelitian yang telah dilakukan.

BAB II

TINJAUAN PUSTAKA

2.1 Pelaporan

Pelaporan menurut Luther M. Gullick (1960) dalam bukunya *Papers on the Science of Administration* merupakan salah satu fungsi manajemen berupa penyampaian perkembangan atau hasil kegiatan atau pemberian keterangan mengenai segala hal yang bertalian dengan tugas dan fungsi-fungsi kepada pejabat yang lebih tinggi, baik secara lisan maupun tertulis sehingga dalam penerimaan laporan dapat memperoleh gambaran bagaimana pelaksanaan tugas orang yang memberi laporan.

2.2 Kebakaran

Kebakaran adalah suatu peristiwa oksidasi dengan ketiga unsur (bahan bakar, oksigen dan panas) yang berakibat menimbulkan kerugian harta benda atau cidera bahkan sampai kematian (Karla, 2007; NFPA, 1986). Menurut Dewan Keselamatan dan Kesehatan Kerja Nasional (DK3N), kebakaran adalah suatu peristiwa bencana yang berasal dari api yang tidak dikehendaki yang dapat menimbulkan kerugian, baik kerugian materi (rasa takut, *shock*, ketakutan, dan lain-lain) hingga kehilangan nyawa atau cacat tubuh yang ditimbulkan akibat kebakaran tersebut.

Memadamkan kebakaran adalah suatu teknik menghentikan reaksi pembakaran/nyala api. Memadamkan kebakaran dapat dilakukan dengan prinsip menghilangkan salah satu atau beberapa unsur dalam proses nyala api (Depnakertrans, 2008). Untuk memadamkan kebakaran diperlukan pasukan khusus untuk memadamkan kebakaran yang biasa disebut pemadam kebakaran.

Pemadam kebakaran disingkat Damkar adalah orang atau pasukan yang bertugas untuk memadamkan kebakaran, melakukan penyelamatan, dan menanggulangi bencana atau kejadian lainnya. Petugas pemadam kebakaran selain terlatih untuk menyelamatkan korban dari kebakaran atau melakukan pemadaman, juga dilatih untuk menyelamatkan korban-korban bencana seperti kecelakaan lalu lintas, gedung runtuh, banjir, gempa bumi, dan lain-lain. Di lain hal, mereka juga

ditugaskan untuk melakukan tugas-tugas penyelamatan yang tidak menyangkut adanya kebakaran seperti pengevakasian sarang tawon, menyelamatkan korban bunuh diri, menyelamatkan orang atau hewan yang terjebak, menanggulangi pohon tumbang, dan lain-lain. Pemadam kebakaran juga terkadang ditugaskan untuk memberi sosialisasi dan pendidikan kepada rakyat sipil tentang kebakaran dan cara menanggapinya. Pemadam kebakaran mempunyai kendaraan sebagai unit gawat darurat. Tipe kendaraan ini biasanya truk yang bagian belakang merupakan penyimpanan air, dan kendaraan ini umumnya berwarna merah. Ada beberapa tipe kendaraan yang digunakan di kesatuan pemadam kebakaran seperti :

1. Mobil *pick-up double cabin* atau *SUV* yang digunakan untuk membawa perwira/komando pemadam kebakaran.
2. Truk pemadam kebakaran dengan ukuran kecil dan besar sebagai unit pembawa air (unit tanker).
3. Truk pemompa dan penyimpan air (biasanya dapat memompa air dari *Hidrant* dan sumber air lainnya) disebut *Pump Unit*.
4. Truk dan mobil pembawa alat-alat dan perlengkapan (selang, palu, gergaji, P3K, lampu, dan lain-lain) pemadam kebakaran.
5. Truk pembawa tangga (unit *ladder*), serta kendaraan pembantu operasional lainnya seperti: ambulans milik pemadam kebakaran.

Pada kondisi darurat atau menanggapi suatu kebakaran, kendaraan ini wajib diberi laluan dan jalan di lalulintas agar sampai di lokasi dengan cepat. Pada kondisi darurat atau menanggapi suatu kebakaran, kendaraan ini akan membunyikan sirene dan menyalakan lampu-lampu darurat yang umumnya berwarna merah atau biru maupun kuning, jika pengemudi melihat ini di jalan raya atau lalulintas, maka seluruh kendaraan wajib memberi laluan atau minggir untuk memprioritaskan tugas penyelamatan nyawa tersebut. Kemudian jika ada pengemudi yang mengabaikan, membiarkan, atau mengganggu perjalanan kendaraan darurat yang sedang menjalankan tugas, maka itu merupakan tindakan pelanggaran lalu lintas dan sangat dilarang dalam peraturan lalu-lintas Indonesia maupun seluruh dunia.

2.2.1 Visi

Terciptanya rasa aman masyarakat dari kebakaran dan bencana lain.

2.2.2 Misi

1. Memberikan pelayanan prima dalam bidang pencegahan dan pemadaman kebakaran, serta penyelamatan jiwa pada kebakaran dan kejadian bencana.
2. Meningkatkan ketahanan lingkungan bersama masyarakat.
3. Meningkatkan kerjasama dengan instansi terkait.

2.2.3 Daftar Nama-Nama Pemadam Kebakaran Beserta Nomor *Handphone* di Kota Pontianak

- a. Kecamatan Pontianak Barat
 1. Pemadam Mitra Jawi, Jln. Hasanudin (0561 – 7527434)
 2. Pemadam PPKJ, Jln. Komyos Sudarso (0813 4546 5527)
 3. Bhakti 3 Serumpun, Jl. Komyos Sudarso Perum II (0813 5266 2287)
 4. Bhakti Sei. Beliung, Jl. Komyos Sudarso (0561 – 7171066)
 5. Swadesi Borneo, Jl. Tabrani Ahmad (0561 – 795569)
 6. Redam 21, Jl. Komyos Sudarso Gg. Karya Tani (0812 5604 5454)
 7. Pemadam Kota Baru, Jl. Komyos Sudarso (0823 6017 2578)
- b. Kecamatan Pontianak Selatan dan Tenggara
 1. Pemadam Panca Bakti, Jl. Suprpto No. 30 (0561 – 740916)
 2. Pemadam Budi Pekerti, Jl. Gajahmada No. 147 (0561 – 739200)
 3. Pemadam Mitra Bhakti, Jl. Siam (0561 – 747349)
 4. Pemadam YPKK, Jl. Kedah (0561 – 7511899)
 5. Pemadam Pandu Siaga, Jl. Wonobaru (0813 4529 7347)
 6. Pemadam Join Fitrah, Jl. Adisucipto Gg. Fitrah (0821 5928 0006)
 7. Pemadam Purnama, Jl. Purnama (0812 5678 5544)
 8. Pemadam Kapuas Bhakti (0852 4525 6444)
 9. Pemadam Armed (0812 5676 6669)
- c. Kecamatan Pontianak Timur
 1. Pemadam Bintang Timur, Jl. Saman Hadi Perum III (0813 4594 5401)
 2. Pemadam Hikmah, Jl. Tanjung Raya Gg. Alhuda (0561 – 571420)

3. Pemadam PKPA 86, Jl. Ya' M. Sabran No. 86 (0812 5604 5454)
- d. Kecamatan Pontianak Utara
 1. Pemadam BPAS, Jl. Situt Mahmud No. 16 A (0561 – 883030)
 2. Pemadam UPKGR, Gg. Selat Sunda (0561 – 887450)
 3. Pemadam CSK, Jl. Situt Mahmud Gg. Selat Madura (0821 5945 5685)
 4. Pemadam Tri Tura, Jl. Tanjung Raya I (0853 4606 1331)

2.3 *GPS Photo Tagging (Geotagging)*

Geotagging merupakan proses penambahan informasi geospasial pada berbagai media digital. Media yang telah mengalami proses *geotagging* akan memiliki informasi koordinat berupa *longitude* (bujur), *latitude* (lintang) dan *altitude* (ketinggian). Hal tersebut memungkinkan media dapat diposisikan secara tepat pada peta (Joshi, et al., 2010). Dengan menggunakan *geotagging*, pengguna dapat menemukan suatu lokasi dengan informasi yang lebih spesifik. Kita dapat menemukan gambar pada lokasi tertentu dengan memasukkan arah lintang dan bujur. Dengan teknik *geotagging* ini dapat disisipkan berupa gambar atau media yang dapat menunjukkan lokasi tersebut. Basis dari *geotagging* adalah posisi, pada kebanyakan posisi ini diketahui dari *GPS*. Di samping *longitude* dan *latitude* sistem menunjukkan tiap lokasi dari bumi 180 derajat barat sampai 180 derajat timur sepanjang garis *equator* dan 90 derajat utara sampai 90 derajat selatan meridian (bujurutama). Ada 2 pilihan melakukan *geotagging* yaitu, dengan mengambil informasi posisi dari *GPS* atau melampirkan petanya setelah gambar tersebut diambil.

Ada dua opsi utama untuk melakukan *geotagging* foto; mengambil informasi *GPS* pada saat foto diambil atau "melampirkan" foto ke peta setelah foto diambil. Untuk mengambil data *GPS* pada saat foto diambil, pengguna harus memiliki kamera dengan *GPS* bawaan atau *GPS* mandiri bersama dengan kamera digital. Karena persyaratan penyedia layanan nirkabel di Amerika Serikat untuk menyediakan informasi lokasi yang lebih tepat untuk panggilan 911 pada 11 September 2012, semakin banyak *smartphone* yang memiliki chip *GPS* bawaan. Sebagian besar *smartphone* sudah menggunakan chip *GPS* bersama dengan kamera *built-in* untuk memungkinkan pengguna untuk *geotag* foto secara

otomatis. Orang lain mungkin memiliki chip *GPS* dan kamera tetapi tidak memiliki perangkat lunak internal yang diperlukan untuk menanamkan informasi *GPS* di dalam gambar. Beberapa kamera digital juga memiliki *GPS* bawaan atau *built-in* yang memungkinkan untuk *geotagging* otomatis. Perangkat menggunakan *GPS*, *A-GPS* atau keduanya. *A-GPS* bisa lebih cepat mendapatkan perbaikan awal jika dalam jangkauan menara jaringan seluler, dan dapat bekerja lebih baik di dalam gedung. *GPS* tradisional tidak memerlukan menara jaringan seluler dan menggunakan sinyal *GPS* standar di luar daerah perkotaan.

Data *geotag* ini kemudian dapat menuliskan informasi lokasi pada foto dengan bentuk data *EXIF HEADER*. Berikut contoh foto yang mengandung informasi lokasi atau foto *geotagging* :

Rincian	
Nama :	IMG20190108201118
Waktu :	08/01/2019 20.11
Lokasi :	Kalimantan Barat, Kota Pontianak, Pontianak, Sungai Jawi, Gang Sepakat 6, 78113, Indonesia
Dimensi :	4160 * 3120
Ukuran :	4,53 MB
Jalur :	Kartu SD/DCIM/Camera/IMG20190108201118.jpg
Perangkat :	A1601
Jarak fokus :	3,5 mm
Bukaan :	f/2,2
ISO :	125
Lampu kilat :	Tanpa lampu kilat
Keseimbangan pth :	Otomatis

Gambar 2.1 foto yang berisi informasi lokasi berupa data *EXIF Header*

2.4 Sistem Operasi

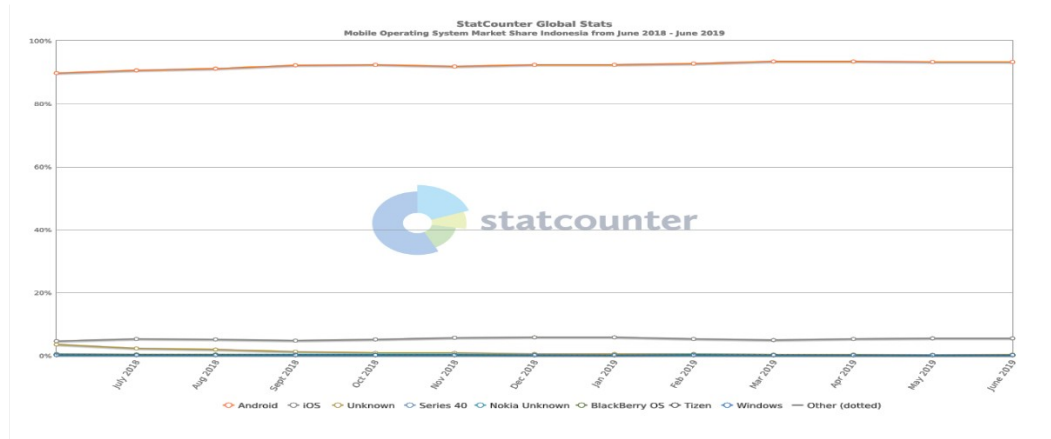
2.4.1 Android

Menurut Safaat (2015) *Android* adalah sebuah sistem operasi untuk perangkat mobile berbasis *Linux*. *Android* menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang akan digunakan untuk bermacam peranti bergerak. Awalnya, *Google Inc.* membeli *Android Inc.*, pendatang baru yang membuat peranti lunak untuk *smartphone*. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *T-Mobile*, dan *Nvidia*. Pada saat perilisan perdana *Android*, 5 November 2007, *Android* bersama *Open Handset Alliance* menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, *Google* merilis kode-kode *Android* di bawah *lisensi Apache*, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Banyak *smarthphone* dan *tablet* yang menggunakan sistem operasi dengan versi yang berbeda. Semakin versi tinggi fiturnya, semakin canggih *smartphone* atau *tablet* yang dipakai. *Android* dipuji sebagai *platform mobile* pertama yang lengkap, terbuka, dan bebas. Lengkap dalam artian *android* menyediakan *tools* dalam membangun *software* dan memungkinkan peluang untuk membangun aplikasi, terbuka bagi pengembang untuk mengembangkan aplikasi *android*, dan bebas biaya keanggotaan, bebas biaya pengujian, tidak ada kontrak yang diperlukan, dan aplikasi *android* dapat didistribusikan dan diperdagangkan dalam bentuk apapun.

Di Indonesia pengguna *smartphone* yang menggunakan sistem operasi *Android* adalah pengguna terbesar yang menempati peringkat pertama. Menurut data dari *GlobalState Counter* dalam rentang waktu juni 2018 hingga juni 2019, pengguna *smartphone* dengan sistem operasi *Android* di Indonesia sekitar 93,22%, diikuti sistem operasi *IOS* di peringkat kedua dengan 5,52%. Hal ini yang kemudian menjadi pertimbangan penggunaan basis aplikasi *Android* pada

penelitian ini. Lampiran gambar data statistik pengguna *OS Android* di Indonesia dalam rentang waktu 1 tahun mulai juni 2018 hingga juni 2019 dapat dilihat pada gambar 2.2.



Gambar 2.2 Data statistik pengguna *OS Android* di Indonesia

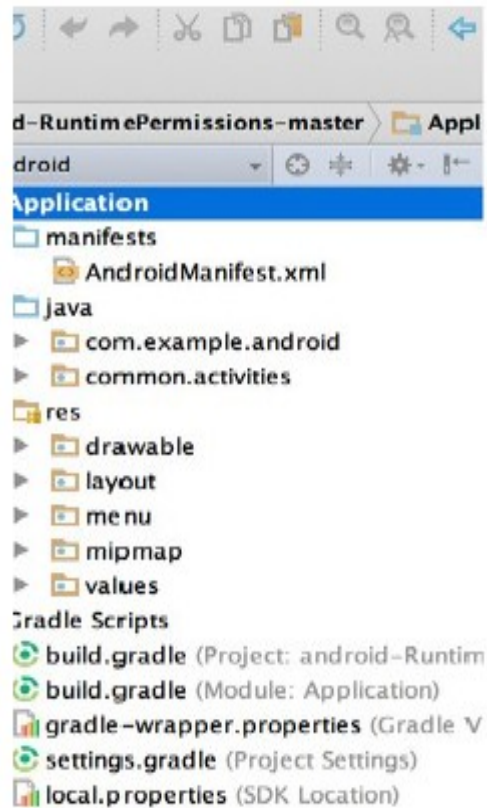
(Sumber : <http://gs.statcounter.com/os-market-share/mobile/indonesia>)

2.4.2 *Android Studio*

Dikutip dari situs resminya <https://developer.android.com/studio/intro/index.html?hl=id>, *Android Studio* merupakan sebuah *Integrated Development Environment (IDE)* khusus untuk membangun aplikasi yang berjalan pada *platform android*. *Android Studio* ini berbasis pada *IntelliJ IDEA*, sebuah *IDE* untuk bahasa pemrograman *Java*. Bahasa pemrograman utama yang digunakan adalah *Java*, sedangkan untuk membuat tampilan atau *layout*, digunakan bahasa *XML*. *Android Studio* juga terintegrasi dengan *Android Software Development Kit (SDK)* untuk *deploy* ke perangkat *android*. *Android Studio* juga merupakan pengembangan dari *eclipse*, dikembangkan menjadi lebih kompleks dan profesional yang telah tersedia di dalamnya *Android Studio IDE*, *Android SDK tools*. Setiap proyek di *Android Studio* berisi satu atau beberapa modul dengan file kode sumber dan file sumber daya. Jenis - jenis modul mencakup :

- Modul aplikasi *Android*
- Modul Pustaka
- Modul *Google App Engine*

Secara *default*, *Android Studio* akan menampilkan file proyek Anda dalam tampilan proyek *Android*, seperti yang ditampilkan dalam gambar 2.3

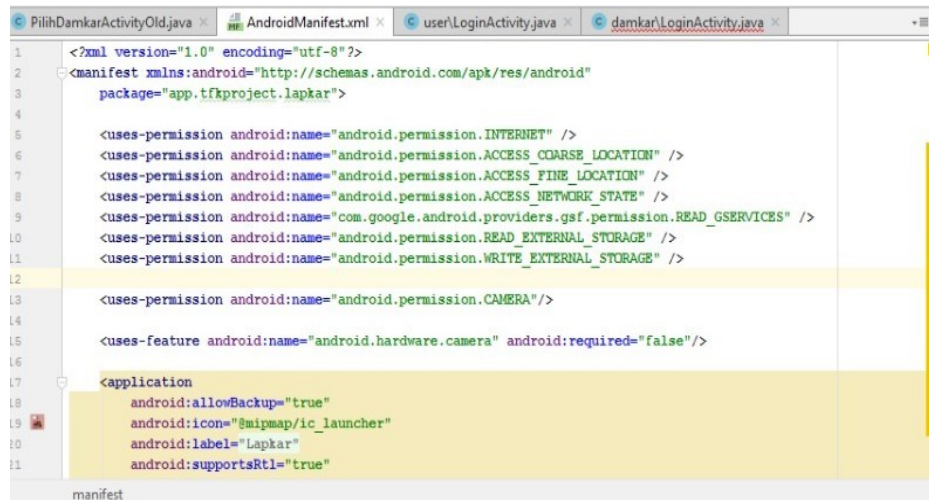


Gambar 2.3 File Proyek Tampilan *Android Studio*

(Sumber : <https://developer.android.com/studio/intro/index.html?hl=id>)

Tampilan disusun berdasarkan modul untuk memberikan akses cepat ke file sumber utama proyek Anda. Semua file versi terlihat di bagian atas di bawah *Gradle Scripts* dan masing-masing modul aplikasi berisi *folder* berikut :

1. *Manifests* : Berisi file *AndroidManifest.xml*, dalam pengembangan aplikasi pada penelitian ini, di dalam file *manifest* terdapat kode yang berfungsi untuk meminta izin hak akses antara lain, *internet*, pembacaan memori, akses *GPS* untuk mendapatkan lokasi, konfigurasi layanan *google*, serta akses kamera. Berikut dilampirkan *source code* pada file *AndroidManifest.xml* yang berfungsi untuk memberlakukan izin aplikasi pada penelitian ini.



Gambar 2.4 Source Code Android Manifest.xml pada Aplikasi

2. *Java* : Berisi file kode sumber *Java*, termasuk kode pengujian *JUnit*.
- Res* : Berisi semua sumber daya bukan kode, seperti tata letak *XML*, *string* user interface, dan gambar *bitmap*.

2.5 Aplikasi

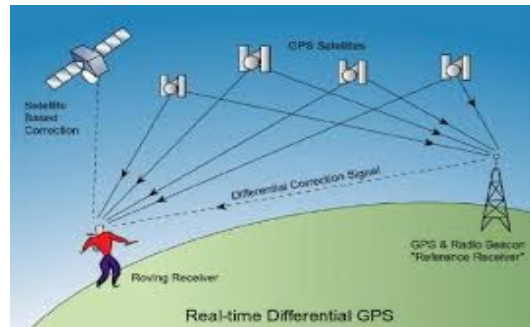
Perangkat lunak aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya dibandingkan dengan perangkat lunak sistem yang mengintegrasikan berbagai kemampuan komputer, tapi tidak secara langsung menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna. Contoh utama perangkat lunak aplikasi adalah pengolah kata, lembar kerja, dan pemutar media. Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket atau suite aplikasi (*application suite*) (Safaat H, 2012).

2.6 Metode Pencarian Lokasi

Ada beberapa metode pencarian lokasi yang umum digunakan, di mana masing-masing berbeda karena memiliki kelebihan dan kelemahan masing-masing tergantung penggunaannya. Sebagai contoh, metode yang satu bisa saja memerlukan biaya lebih besar dari yang lain namun memiliki tingkat akurasi yang tinggi. Adapun metode yang sering digunakan, (Deitel et al, 2002), yaitu:

1. *Device-based*

metode ini memerlukan perangkat tambahan berupa modul *GPS* yang didasarkan pada sistem satelit global.



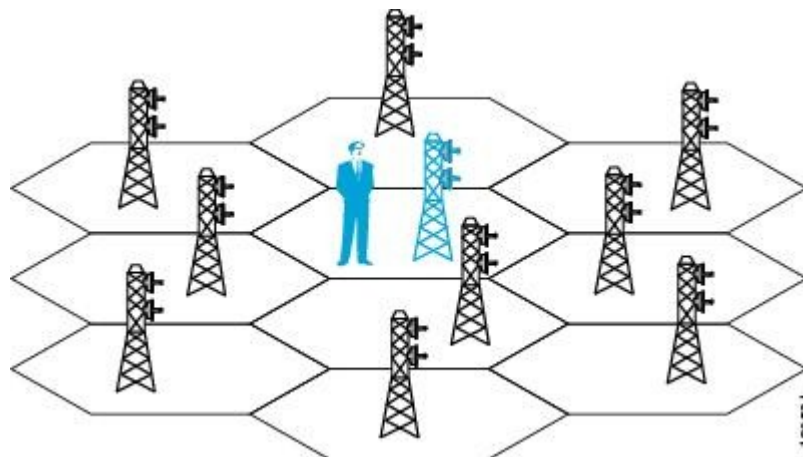
Gambar 2.5 Penentuan Lokasi dengan *GPS* (HnD Computer)

2. *Network-based*

metode ini didasarkan pada sistem jaringan seluler. Ada beberapa jenis mekanisme pada metode ini, yakni:

a. *Cell of Origin*

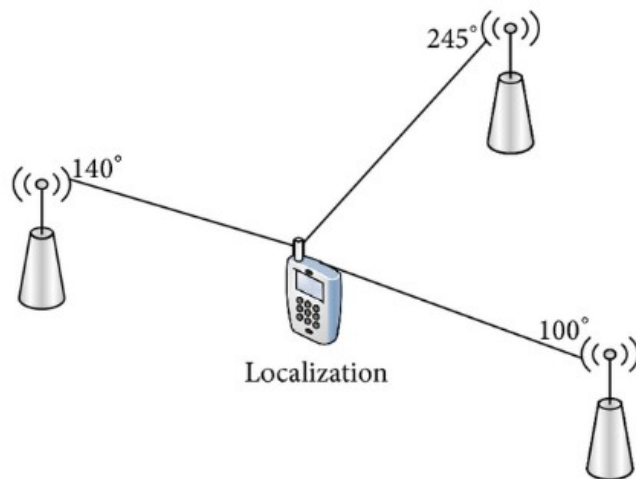
Pada metode ini penentuan lokasi didasarkan pada *cell* jaringan di mana posisi ponsel pengguna berada, di mana masing-masing *cell* dijangkau oleh satu BTS (*Base Tower Station*). Informasi lokasi ditentukan oleh BTS terdekat yang menjangkau posisi ponsel *user*. Metode ini merupakan metode paling sederhana dan memiliki tingkat akurasi yang rendah.



Gambar 2.6 *Cell of Origin* (Cisco)

b. *Angle of Arrival*

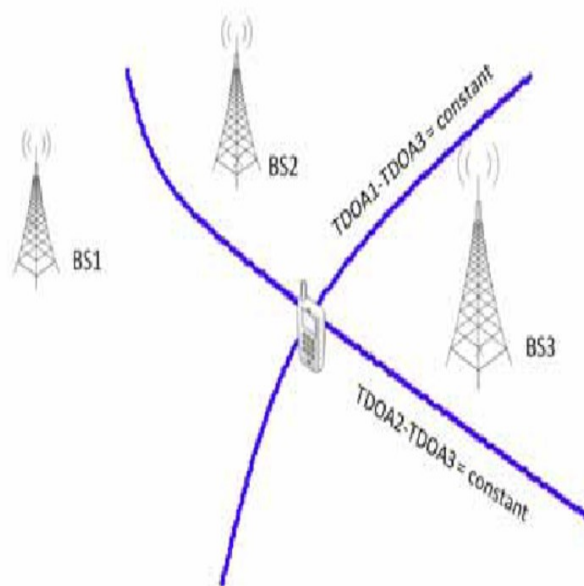
Metode ini memiliki tingkat akurasi yang lebih baik. Penentuan lokasi didasarkan pada sudut yang dibentuk antara posisi ponsel dengan dua atau lebih BTS yang menjangkaunya. *Angle of Arrival* menentukan informasi lokasi dengan *triangulasi* (pembentukan segitiga). Metode ini mudah terkena gangguan pada daerah perkotaan. Oleh karena itu baik digunakan pada daerah dengan populasi yang sedikit.



Gambar 2.7 *Angle of Arrival* (cs.odu.edu)

c. *Time Difference of Arrival*

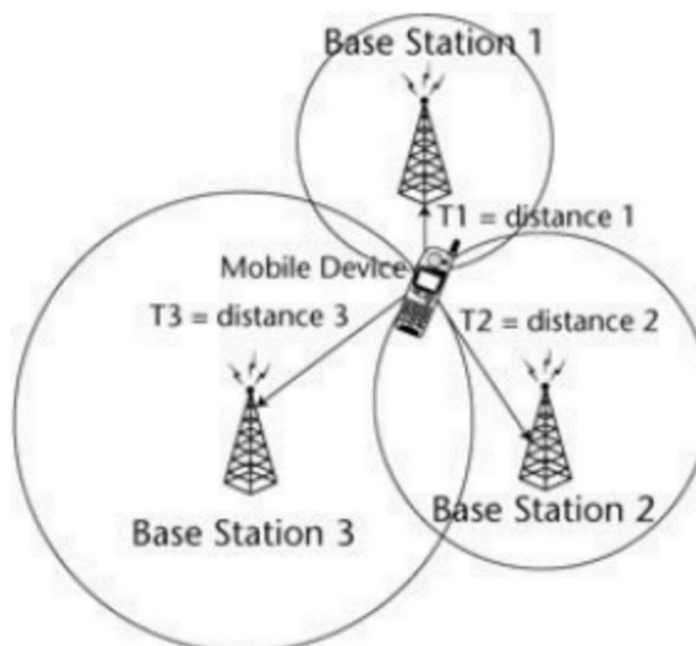
Metode ini menentukan jarak dengan menggunakan *triangulasi* dari sinyal perjalanan. Informasi lokasi ditentukan dari perpotongan jarak busur lingkaran yang dibentuk atas perbedaan waktu kedatangan yang dihasilkan.



Gambar 2.8 Time Difference of Arrival (ietr.fr)

d. *Enhanced Observed Time Difference*

Pada metode ini, tiga BTS akan mengamati perbedaan sinyal waktu perjalanan ponsel *user*, dengan mentransmisikan sinyal ke ponsel. Metode ini menyerupai TDOA, namun membutuhkan perangkat tambahan pada ponsel.



Gambar 2.9 Enhanced Observed Time Difference (cs.odu.edu)

e. *Location-Pattern Matching*

Pada metode ini, penentuan lokasi didasarkan pada kondisi daerah di mana ponsel *user* berada. Faktor penting yang digunakan adalah tekstur area yang didapat dari pemantulan gelombang radio ponsel terhadap bangunan atau penghalang yang dilewatinya. Pola pemantulan sinyal akan dianalisis oleh BTS untuk dicocokkan dengan pola area yang tersimpan di *database*. Metode ini tidak membutuhkan perangkat tambahan, namun membutuhkan pengelolaan *database* dan cocok digunakan di area perkotaan.

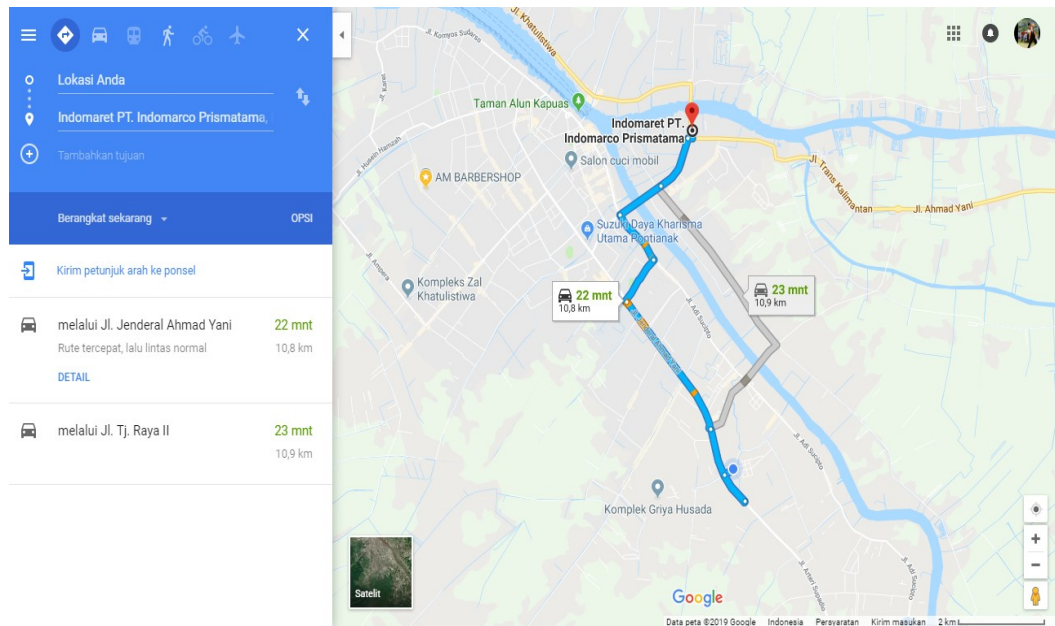
3. *Hybrid method,*

metode ini mengkombinasikan metode *GPS* dengan *Network-based* untuk menghasilkan informasi yang cepat dan akurat.

2.7 *Google Maps*

Menurut (Juwairiah, 2013:302) *Google Maps* merupakan sebuah layanan peta dunia *virtual* berbasis *web* yang disediakan oleh *Google*. *Browsing* peta dapat dilakukan dengan mudah, dan data peta selalu diperbaharui secara berkala. Setiap orang bebas memperbaharui konten, jika belum pernah melalui jalan yang ada di peta, dapat menambah dengan bebas. Tidak sedikit orang dan perusahaan komersial telah ikut menempatkan data untuk keperluan masing-masing (Riyanto, 2010). Layanan ini dapat diakses melalui situs <http://maps.google.com>. Pada situs tersebut kita dapat melihat informasi geografis pada hampir semua wilayah di bumi. Layanan ini interaktif, karena di dalamnya peta dapat digeser sesuai keinginan pengguna, mengubah tingkat *zoom*, serta mengubah tampilan peta.

Tampilan yang akan muncul pada situs *Google Maps* adalah sebagai berikut.



Gambar 2.10 Tampilan *Google Maps*

Fasilitas yang terdapat pada *Google Maps* antara lain adalah menjelajah peta, dan mencari lokasi tertentu seperti hotel, tempat hiburan, lokasi bisnis, dan menghitung rute dalam berkendara. *Google Maps* mempunyai fasilitas *zoom* yang berfungsi untuk mengubah fokus. Terdapat 18 tingkatan *zoom* yang dapat dipilih dengan cara menekan tombol *plus/minus*, atau dengan menggeser tombol yang terdapat di antara tanda *plus/minus*. Tampilan *zoom* pada *Google Maps* dari yang paling rendah hingga yang paling tinggi dapat dilihat pada gambar 2.11 (tertinggi) dan gambar 2.12 (terendah).



Gambar 2.11 Tingkat *Zoom Level* Tertinggi



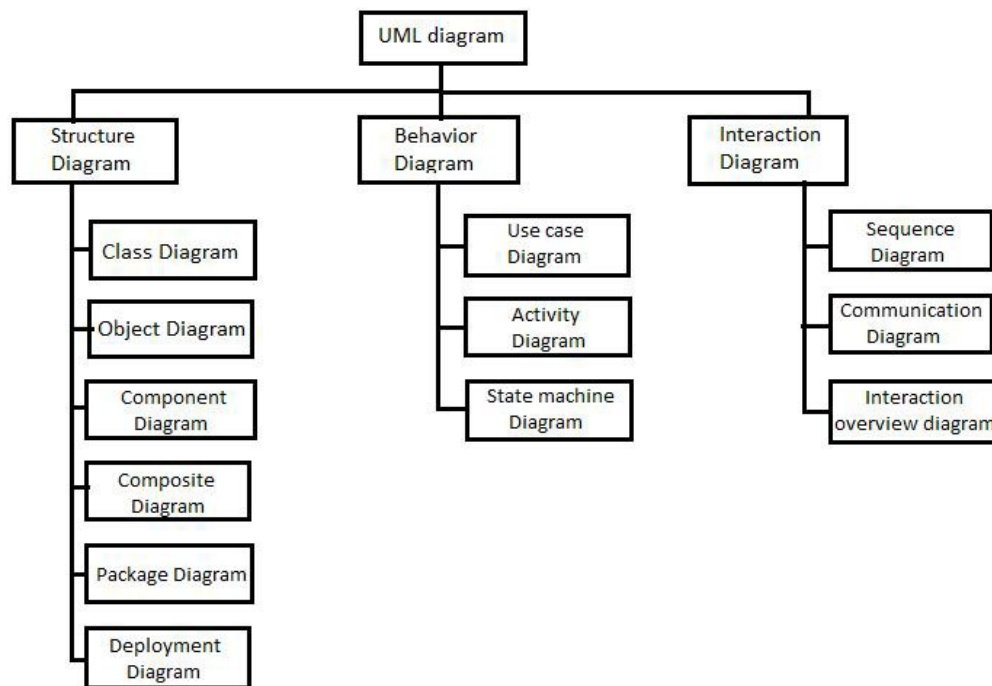
Gambar 2.12 Tingkat *Zoom Level* Terendah

Google Maps menyediakan beberapa mode pada tampilan petanya. Pada gambar berikut, dapat dilihat tampilan dengan *mode map*. *Mode map* merupakan bentuk peta dasar, yang di dalamnya terdapat informasi mengenai nama jalanan, sungai, danau, dan lain-lain. Namun untuk wilayah negara Indonesia fitur ini belum tersedia karena terbatasnya *database*. Selain itu juga terdapat beberapa mode lain, yaitu:

1. *Sattelite*, menampilkan gambar muka bumi di seluruh lokasi di dunia yang diambil dari satelit atau pesawat udara
2. *Terrain*, menampilkan citra topografi dari muka bumi.
3. *Traffic*, menampilkan informasi mengenai keadaan lalu lintas dengan indikator warna merah, kuning, dan hijau (hanya tersedia di beberapa kota di negara Amerika Serikat).
4. *Street*, merupakan tampilan yang serupa dengan mode *sattelite*, namun dapat dilihat dari berbagai sudut (hingga 360°) (hanya tersedia di beberapa kota di Amerika Serikat).

2.8 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah standard pemodelan dalam pembuatan aplikasi berorientasi obyek yang diajukan oleh *Object Management Group (OMG)* pada tahun 1996. Dalam UML terdapat 13 diagram yang dikelompokkan dalam 3 kategori



Gambar 2.13 Klasifikasi Diagram UML

Sumber : (Rosa A.S., 2013)

Gambar 2.13 dapat dijelaskan sebagai berikut:

1. *Structure Diagram*, yaitu kumpulan diagram-diagram yang menggambarkan struktur statis dari sistem yang dimodelkan.
2. *Behavior Diagram*, yaitu kumpulan diagram-diagram yang menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi di dalam sistem.
3. *Interaction Diagram*, yaitu kumpulan diagram-diagram yang menggambarkan interaksi sistem dengan sistem lain ataupun interaksi antar subsistem dalam sebuah sistem.




Secara garis besar, beberapa diagram utama sudah dapat menggambarkan keseluruhan sistem. Diagram tersebut antara lain *use case diagram*, *class diagram*, *sequence diagram*, dan *activity diagram*.

2.9.1 Use case Diagram

Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Dengan kata lain, *use case diagram* digunakan untuk mengetahui fungsi-fungsi apa saja yang terdapat di dalam sistem dan siapa saja yang berhak mengakses fungsi tersebut. (Rosa A.S., 2013).

Berikut merupakan simbol-simbol notasi yang digunakan pada *use case diagram*.

Tabel 2.1 Tipe Relasi pada *Use Case Diagram*

No	Notasi	Nama	Deskripsi
1		<i>Use Case</i>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2		<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
3		<i>Association</i>	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau <i>use case</i> memiliki interaksi dengan aktor.
4	5 <<extend>> - - - - ->	<i>Extended</i> (Ekstensi)	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.
5	6 <<include>> < - - - - -	<i>Include</i> (Menggunakan)	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.




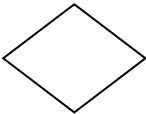

			<i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan.
--	--	--	---

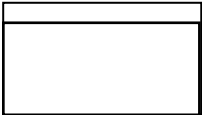
2.9.2 Activity diagram

Activity Diagram menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem, bukan apa yang dilakukan oleh aktor. (Rosa A.S., 2013)

Berikut ini adalah simbol-simbol notasi yang digunakan pada diagram aktivitas:

Tabel 2.2 Tipe Relasi pada *Activity Diagram*

No	Notasi	Nama	Deskripsi
1		Status Awal (<i>initial node</i>)	Status awal aktivitas, sebuah diagram aktivitas memiliki status awal.
2		Status Akhir (<i>final node</i>)	Status akhir yang dilakukan sistem.
3		Aktivitas (<i>activity</i>)	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja..
4		Percabangan (<i>decision</i>)	Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu.
5		Penggabungan (<i>join</i>)	Asosiasi penggabungan di mana lebih dari satu aktivitas digabungkan menjadi satu.


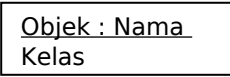

6		<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
---	---	-----------------	--

2.9.3 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Dalam menggambarkan *sequence diagram* perlu memperhatikan objek-objek yang terlibat di dalam *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. (Rosa A.S.,2013)

Berikut ini adalah simbol-simbol notasi yang digunakan pada *sequence diagram*:

Tabel 2.3 Deskripsi Notasi pada *Sequence Diagram*

No	Notasi	Nama	Deskripsi
1		<i>Actor</i>	Orang Proses, atau Sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat. Biasanya dinyatakan dengan menggunakan kata benda di nama aktor.
2		<i>Object</i>	Menyatakan objek yang berinteraksi pesan.
3		Garis Hidup	Menyatakan kehidupan suatu objek.

2.9 Database

Database atau basis data merupakan sekumpulan data yang berisi informasi yang saling berhubungan (Yuswanto, 2005). Basis data sebuah penyimpanan data yang besar yang bisa digunakan oleh banyak pengguna dan departemen. Kumpulan dari data yang berhubungan dan merupakan deskripsi dari data-data tersebut yang didesain untuk menemukan informasi yang dibutuhkan suatu perusahaan. (Connolly, 2010). Proses memasukkan dan mengambil data dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (*Database Management System / DBMS*). Tujuan utama dari *DBMS* adalah untuk memberikan tinjauan abstrak data kepada *user* (Marlinda, 2004).

2.10 Pengujian Perangkat Lunak

Pengujian perangkat lunak merupakan suatu teknik yang digunakan menguji apakah sebuah perangkat lunak yang dihasilkan telah sesuai dengan yang diharapkan atau belum. Menurut Pressman (2002), pengujian adalah proses eksekusi suatu program untuk menemukan kesalahan sebelum digunakan oleh pengguna akhir (*end-user*). Dalam pengujian, kualitas dari data pengujian lebih penting dari pada jumlah datanya. ada beberapa metode yang dapat digunakan untuk memilih data pengujian, yaitu :

2.10.1 Teknik User Acceptance Test (UAT)

User acceptance test (UAT) merupakan pengujian yang dilakukan oleh *end-user*, pihak yang langsung berinteraksi dengan sistem dan dilakukan verifikasi apakah fungsi yang ada telah berjalan sesuai dengan kebutuhan (Sugiyono, 2008).

Acceptance testing biasanya berusaha menunjukkan bahwa sistem telah memenuhi persyaratan-persyaratan tertentu. Pada pengembangan *software* dan *hardware* komersial, *acceptance test* biasanya disebut juga "*alpha tests*" (yang dilakukan oleh pengguna *in-house*) dan "*beta tests*" (yang dilakukan oleh pengguna yang sedang menggunakan atau akan menggunakan sistem tersebut). *Alpha* dan *beta test* biasanya juga menunjukkan bahwa produk sudah siap untuk dijual atau dipasarkan. *Acceptance testing* mencakup data, *environment* dan

skenario yang sama atau hampir sama pada saat *live* yang biasanya berfokus pada skenario penggunaan produk tertentu (Black , 2002).

2.11 Kajian Terkait

Beberapa peneliti terdahulu yang telah melakukan penelitian tentang sistem informasi geografis wisata berbasis *android* yang salah satunya adalah Muhammad Iqbal Asyrafi (2018) dengan penelitian berjudul “Rancang Bangun Aplikasi Pelaporan TAMAT (Tanya Camat) untuk Wilayah Kecamatan Pontianak Barat Berbasis *Android*”. Aplikasi pelaporan ini dapat memfasilitasi masyarakat dalam menyampaikan keluhan, kritik dan saran ke pihak Camat Pontianak. Pada penelitian tersebut yang dibuat adalah masyarakat di Kota Pontianak dapat menyampaikan laporan kepada Camat untuk menyampaikan keluhan, kritik, dan saran. Perbedaan dengan penelitian ini adalah penelitian terdahulu hanya membuat aplikasi laporan untuk bertanya kepada Camat dan hanya menampilkan teks yang disampaikan dari masyarakat. Sedangkan pada penelitian ini adalah masyarakat dapat membuat laporan yang akurat mengenai kasus kebakaran menggunakan metode *geotag* dan Pemadam Kebakaran dapat memanfaatkan layanan Google Maps untuk rute yang dilaporkan masyarakat/pelapor.

Sedangkan pada Agustina melakukan penelitian tentang “Sistem Informasi Geografis Kuliner Berbasis *Android* Menggunakan Haversine Formula di Kota Yogyakarta. (Agustina, 2014). Penelitian ini dilakukan oleh Dika Agustina tahun 2014 jurusan Informatika, Fakultas Sains dan Teknologi, UIN Sunan Kalijaga Yogyakarta. Berdasarkan penelitian ini digunakan metode Haversine Formula sebagai metode untuk menentukan jarak terdekat dari lokasi pengguna ke lokasi kuliner. Selain itu juga sistem ini menerapkan konsep client-server yang memanfaatkan jaringan internet dengan menggunakan *GPS (Global Positioning System)* di atas *Android* platform. Sistem ini menggunakan *library* pada *Google Maps API* sebagai layanan peta visual. Metode pengembangan sistem yang digunakan dalam pembuatan sistem ini adalah metode SDLC (*Software Development Life Cycle*). Sistem ini dapat menampilkan lokasi kuliner yang memiliki jarak terdekat dengan pengguna yang dilengkapi dengan rute menuju lokasi tersebut. Perbedaan dengan penelitian yang akan penulis teliti, aplikasi

yang dibangun menggunakan fitur kamera untuk membuat laporan yang berisi informasi data *geotag*.

Kemudian pada Ary Mardani (2014) melakukan penelitian tentang “Sistem Informasi Geografis Pelaporan Masyarakat (Sigma) Berbasis Foto *Geotag*”. Penelitian ini dilakukan oleh Ary Mardani tahun 2014 jurusan Teknik Informatika Universitas Tanjungpura Pontianak. Berdasarkan penelitian ini diperoleh bahwa Sistem Informasi Geografis Pelaporan Masyarakat (SIGMA) merupakan sebuah sistem yang menampilkan informasi lokasi sebaran masalah berdasarkan posisi geografis pengambilan sebuah foto. Penelitian tersebut memiliki kesamaan dengan penelitian ini yaitu menggunakan metode *geotag*. Sedangkan perbedaan penelitian tersebut dengan penelitian ini yaitu tersedia fitur penentuan rute dari sebuah laporan foto *geotag* menggunakan layanan Google Maps, dan adanya informasi berupa notifikasi mengenai status laporan secara bertahap. Kemudian adanya informasi jarak dari posisi foto kebakaran dengan pihak Damkar.

#ASUS#####A#S#U#S#####

#b#y# #a#d#g#u#a#r#d###C#o#u#r#i#e#r# #N#e#w#####r#t#f#####

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN APLIKASI

3.1 Metodologi Penelitian

3.1.1 Bahan Penelitian

Bahan penelitian yang digunakan berupa data-data yang berdasarkan hasil wawancara dengan Kasip Peralatan dan Pemadaman di Kota Pontianak. Adapun data-data yang diperlukan untuk melakukan pengujian *skala likert's* menggunakan borang kuisioner dan pengujian kompatibilitas.

3.1.2 Alat yang Digunakan

3.1.2.1 Alat Penelitian

Alat penelitian yang digunakan dalam penelitian ini adalah *Unified Modelling Language (UML)*. Diagram yang digunakan dalam pengembangan peranti lunak ini adalah *use case diagram*, *activity diagram*, dan *sequence diagram*.

3.1.2.2 Perangkat Keras

Perangkat keras yang digunakan dalam penelitian ini adalah:

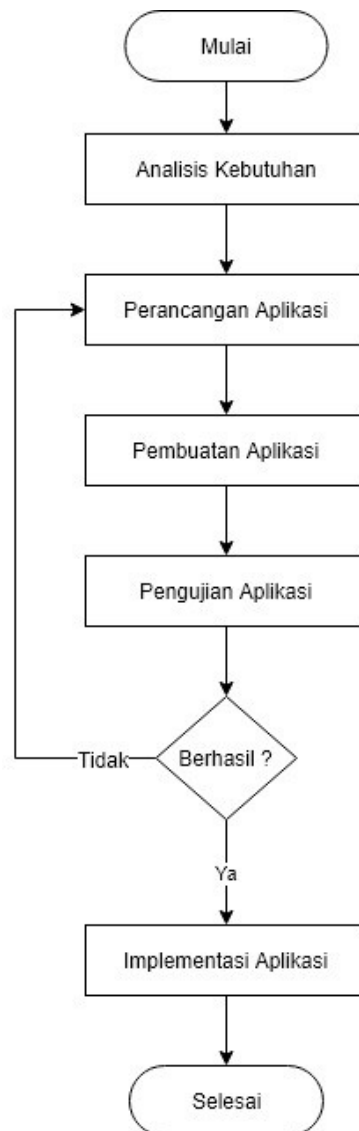
- a. Laptop Asus A456U dengan spesifikasi *Intel (R) Core(TM) i5-62000 CPU @ 2.30GHz 2.40 GHz, RAM 12 GB, HardDisk 500 GB*
- b. *Smartphone Realme 128 GB*
- c. *Smartphone Vivo V5 64 GB*

3.1.2.3 Perangkat Lunak

- a. *Android Studio 3.4.1*
- b. *Android SDK*
- c. *MySQL*
- d. *Web Browser*
- e. *Adobe Photoshop CS6*
- f. *Photoscape*
- g. *Star UML*
- h. *Draw IO*
- i. *Sistem Operasi Windows 10*

3.1.3 Diagram Alir Penelitian

Adapun langkah-langkah yang digunakan dalam penelitian ini dapat diuraikan melalui diagram alir sebagai berikut :



Gambar 3.1 Diagram Alir Penelitian

Adapun tahapan penelitian yang dilaksanakan adalah sebagai berikut:

1. Analisis Kebutuhan

Analisis kebutuhan bertujuan untuk melihat kebutuhan pengguna terhadap sistem yang akan dibuat. Hasil analisis kebutuhan ini menjadi referensi untuk pemodelan aplikasi.

2. Perancangan Aplikasi

Melakukan desain terhadap aplikasi yang akan dibuat agar aplikasi berjalan sesuai kebutuhan pengguna, melalui perancangan, analisa, wawancara, observasi perancangan *UML*, dan merancang antarmuka sistem untuk memodelkan bentuk keseluruhan dari aplikasi.

3. Pembuatan Perangkat Lunak

Tahap ini merupakan tahap di mana perangkat lunak akan dibuat meliputi pembuatan antarmuka pengguna, proses pengolahan *input*, dan penampilan *output*.

4. Pengujian Aplikasi

Pengujian aplikasi dilakukan untuk mengetahui apakah antar subsistem sudah dapat terkoneksi sesuai rancangan. Pengujian aplikasi ini dilakukan menggunakan menggunakan *skala likert* sebagai pengujian tingkat kepuasan pengguna dan menggunakan pengujian kompatibilitas sebagai dapat mengetahui *device* apa aja yang bisa di gunakan.

5. Hasil Pengujian

Analisis hasil pengujian digunakan untuk mengetahui dan mengidentifikasi kesalahan pada sistem yang dibuat sehingga sistem dapat diperbaiki. Hasil analisis pengujian ini juga dapat digunakan untuk pengembangan sistem agar dapat berjalan lebih baik.

3.2 Analisis Kebutuhan

Analisis kebutuhan aplikasi dilakukan dengan melakukan analisis terhadap sistem kerja yang saat ini sedang berjalan. Analisis ini dilakukan melalui wawancara terhadap narasumber baik dari pihak masyarakat sebagai pihak pelapor maupun dari pihak Damkar sebagai pihak terkait yang akan mengambil tindakan atas adanya laporan tentang kebakaran dari masyarakat. Berdasarkan hasil wawancara kepada narasumber tersebut didapatkan *informasi* mengenai tata cara penanganan laporan kebakaran dimulai dari adanya laporan kebakaran yang masuk kepada pihak Damkar hingga selesai proses pemadaman oleh pihak Damkar.

Kemudian dari hasil tersebut maka diperoleh kebutuhan aplikasi yang dibangun pada penelitian ini. Analisis terhadap kebutuhan tersebut dibagi menjadi dua, yaitu analisis kebutuhan fungsional dan analisis kebutuhan non-fungsional.

3.2.1 Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional pada aplikasi yang dibangun dalam penelitian ini diuraikan seperti pada tabel 3.1

Tabel 3.1 Analisis Kebutuhan Fungsional Aplikasi

No.	Kebutuhan Fungsional	Kebutuhan Pengguna	
		Masyarakat	Damkar
1	Pengguna dapat melakukan registrasi akun	✓	✓
2	Pengguna dapat <i>login</i> ke dalam aplikasi	✓	✓
3	Pengguna dapat melakukan prosedur <i>login</i> dengan melakukan verifikasi ketika pengguna lupa <i>password</i>	✓	✓
4	Pengguna dapat membuat laporan kebakaran	✓	
5	Pengguna dapat mengambil foto <i>geotag</i>	✓	
6	Pengguna dapat melihat laporan kebakaran	✓	✓
7	Pengguna dapat melihat status laporan	✓	✓
8	Pengguna dapat memilih Damkar tujuan pada saat membuat laporan	✓	
9	Pengguna dapat melihat pengirim laporan		✓
10	Pengguna mendapatkan notifikasi ketika ada laporan baru yang masuk		✓
11	Pengguna mendapatkan notifikasi ketika status laporan berubah	✓	
12	Pengguna dapat mengubah status laporan		✓
13	Pengguna dapat melihat laporan saya yang telah dibuat		✓

14	Pengguna dapat melihat posisi titik kebakaran di dalam peta pada laporan yang ada	✓	✓
15	Pengguna dapat melihat laporan <i>spam</i>		✓
16	Pengguna dapat melihat rekap laporan		✓
17	Pengguna dapat melihat rekap laporan dalam bentuk titik lokasi pada peta		✓
18	Pengguna dapat melihat alamat pada laporan kebakaran	✓	✓
19	Pengguna dapat melihat rute terdekat menuju lokasi kebakaran		✓
20	Pengguna dapat melihat estimasi waktu untuk sampai ke titik lokasi kebakaran pada peta		✓
21	Pengguna dapat menggunakan sistem navigasi pada peta untuk sampai ke lokasi kebakaran		✓
24	Pengguna dapat <i>logout</i> dari aplikasi	✓	✓

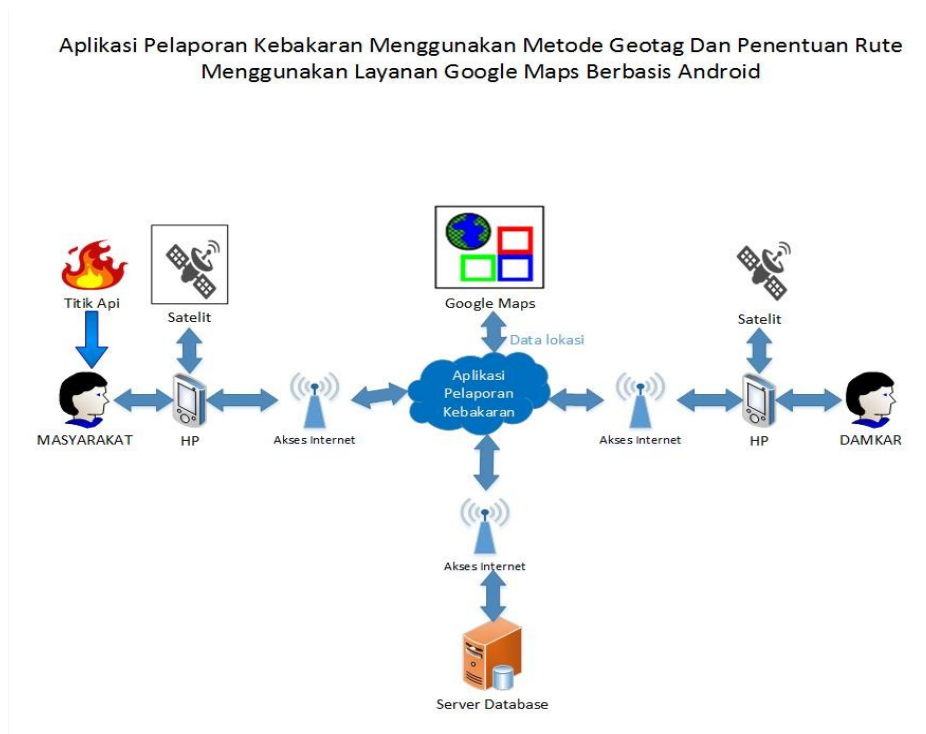
3.2.2 Analisis Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional pada aplikasi dalam penelitian ini dapat diuraikan sebagai berikut :

1. Aplikasi dijalankan pada perangkat *Smartphone* dengan Sistem Operasi *Android* versi 4.2 ke atas.
2. *Smartphone android* yang digunakan harus memiliki perangkat *GPS*.
3. Aplikasi berjalan menggunakan jaringan *internet*.
4. Minimum memori *Smartphone* yang dibutuhkan minimal 100 *MegaByte*.

3.3 Perancangan Sistem

3.3.1 Arsitektur Sistem

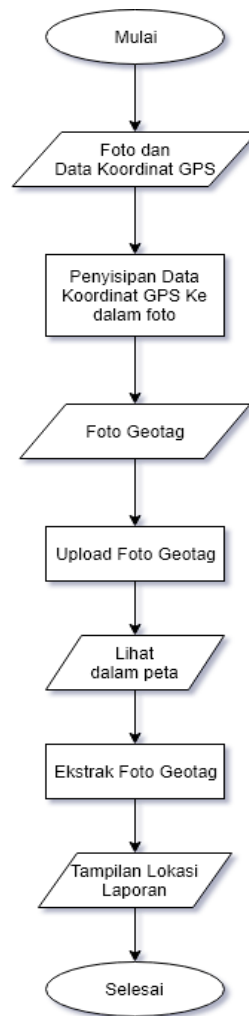


Gambar 3.2 Rancangan Arsitektur Sistem

Alur kerja pada aplikasi yang dibangun dimulai ketika masyarakat melihat adanya kebakaran atau titik api, kemudian masyarakat dapat melaporkan situasi yang dilihatnya melalui aplikasi ini menggunakan *smartphone* yang terhubung dengan jaringan *internet* serta akses satelit sebagai pemancar data lokasi masyarakat tersebut melalui *chip GPS* yang terdapat pada *smartphone* yang digunakan. Kemudian masyarakat dapat melaporkan melalui aplikasi dengan cara mengisi *form* laporan serta melampirkan foto yang didalamnya sudah terdapat data *geotag*. Selanjutnya pada foto *geotag* tersebut dapat diambil data lokasi dari masyarakat dengan cara diekstrak ke dalam bentuk data *latitude* dan *longitude* yang kemudian dapat diolah oleh aplikasi dengan bantuan *API Google Maps* untuk menentukan letak koordinat lokasi masyarakat tersebut pada peta.

Selain itu masyarakat juga harus memilih Damkar dan masyarakat dapat mengetahui jarak dari lokasi masyarakat dengan lokasi Damkar pada aplikasi ini. Pada proses ini data lokasi Damkar diambil dari *database* yang telah disimpan sebelumnya pada saat Damkar melakukan registrasi pada aplikasi. Data lokasi Damkar dan masyarakat sebagai pelapor kemudian diambil dan dikirimkan kepada *Google Maps API* untuk diproses agar diketahui jarak antara Damkar dan lokasi kebakaran, serta dapat diketahui rute terdekat untuk menuju ke lokasi kebakaran.

Kemudian data laporan tersebut disimpan ke dalam *database* pada aplikasi ini. Tahap selanjutnya akan muncul notifikasi pada pihak Damkar sebagai pihak terkait yang akan menangani laporan tersebut. Dalam proses ini aplikasi akan mengambil data lokasi dari sebuah foto yang dikirimkan sebelumnya oleh pelapor. Tahap selanjutnya Damkar dapat melihat titik lokasi kebakaran serta rute menuju lokasi kebakaran dari sebuah foto yang berisi data *geotag*. Selanjutnya Damkar dapat segera menuju ke lokasi kebakaran sesuai dengan titik lokasi yang telah ditentukan.



Gambar 3.3 Data Flow Diagram Mekanisme Foto Geotag

Proses penyisipan foto *geotag* dimulai ketika *user* mengambil foto, disaat yang bersamaan juga diambil data koordinat lokasi menggunakan *GPS* yang tertanam pada *smartphone* pengguna. Setelah foto berhasil diambil kemudian dilanjutkan ke tahap penyisipan data koordinat lokasi yang diperoleh dari data *GPS smartphone* digabungkan dengan foto yang diambil oleh *user*.

Selanjutnya dari proses tersebut diperoleh sebuah foto *geotag*. Foto tersebut kemudian di *upload* sebagai laporan yang ditujukan kepada pihak Damkar. Kemudian Damkar akan membuka foto tersebut untuk melihat lokasi kebakaran. Untuk mendapatkan lokasi kebakaran dari sebuah foto *geotag*, diperlukan proses ekstrak data dari foto *geotag* tersebut.

Proses ekstrak foto dilakukan dengan mengambil data *exif* dari foto *geotag* yang berisikan data *latitude* dan *longitude*. Data *EXIF* (*Exchangeable*

Image File Format) adalah data yang disimpan oleh kamera digital pada sebuah image yang berisi informasi mengenai kondisi dan *setting* kamera digital pada waktu dilakukan pemotretan. Data tersebut kemudian digunakan untuk menampilkan titik koordinat lokasi kebakaran serta digunakan untuk menentukan petunjuk arah yang akan digunakan oleh Damkar untuk sampai ke lokasi kebakaran.

3.3.1.1 Pseudocode Set Foto Geotag

Deklarasi :

ExifInterface, *SimpleDateFormat*, *LATITUDE*, *LATITUDE_REF*, *LONGITUDE*, *LONGITUDE_REF* : *String*

Deskripsi :

```
ExifInterface exif = new ExifInterface(imagePath);
exif.setAttribute(ExifInterface.TAG_GPS_LATITUDE,
GPS.convert(location.getLatitude()));
exif.setAttribute(ExifInterface.TAG_GPS_LATITUDE_REF,
GPS.latitudeRef(location.getLatitude()));
exif.setAttribute(ExifInterface.TAG_GPS_LONGITUDE,
GPS.convert(location.getLongitude()));
exif.setAttribute(ExifInterface.TAG_GPS_LONGITUDE_REF,
GPS.longitudeRef(location.getLongitude()));

SimpleDateFormat fmt_Exif = new SimpleDateFormat("yyyy:MM:dd
HH:mm:ss", Locale.getDefault());
exif.setAttribute(ExifInterface.TAG_DATETIME,fmt_Exif.format(new
Date(location.getTime())));
exif.saveAttributes();
```

Penjelasan : Deklarasi variabel dan *type* data antara lain *ExifInterface*, *SimpleDateFormat*, *LATITUDE*, *LATITUDE_REF*, *LONGITUDE*, *LONGITUDE_REF* : *String*. Selanjutnya deskripsikan *ExifInterface* sebagai *method* untuk menangkap dan menyimpan atribut koordinat lokasi dan waktu pengambilan foto dalam bentuk data *geotag*. Jalankan fungsi untuk menyimpan data *geotag* dalam format *EXIF*.

3.3.1.2 Pseudocode Ekstrak Foto Geotag

Deklarasi :

LATITUDE, LATITUDE_REF, LONGITUDE, LONGITUDE_REF : String

Deskripsi :

```

Intent intent = new Intent(context, MapTkpActivity.class);
intent.putExtra("key_link_img", itemList.get(position).getUrl_gambar());
intent.putExtra("key_nama_tujuan", itemList.get(position).getJudul());
intent.putExtra("key_nama_tempat", itemList.get(position).getTempat());
intent.putExtra("key_lokasi_tujuan", itemList.get(position).getJalan());

if ((LATITUDE != null) && (LATITUDE_REF != null) && (LONGITUDE != null) &&
    (LONGITUDE_REF != null)) {
    if (LATITUDE_REF.equals("N")) {
        Latitude = convertToDegree(LATITUDE);
    } else {
        Latitude = 0 - convertToDegree(LATITUDE);
    }

    if (LONGITUDE_REF.equals("E")) {
        Longitude = convertToDegree(LONGITUDE);
    } else {
        Longitude = 0 - convertToDegree(LONGITUDE);
    }
}

intent.putExtra("key_lat_tujuan", Latitude);
intent.putExtra("key_long_tujuan", Longitude);
context.startActivity(intent);

```

Penjelasan : Deklarasi variabel dan type data antara lain *LATITUDE, LATITUDE_REF, LONGITUDE, LONGITUDE_REF : String*. Selanjutnya deskripsikan *intent* sebagai *method* untuk mengambil file gambar yang ada pada aplikasi yang akan di ekstrak data *geotag*nya. Jika variabel *latitude* tidak sama dengan kosong dan variabel *latitude reff* tidak sama dengan kosong dan variabel *longitude* tidak sama dengan kosong dan variabel *longitude reff* tidak sama dengan kosong maka *check* apakah nilai *latitude reff* mengarah ke utara, jika iya maka nilai *latitude* positif, jika ke selatan maka nilai *latitude* sama dengan negatif. Selanjutnya *check* apakah nilai *longitude reff* mengarah ke timur jika iya, maka nilai *longitude* positif. Sedangkan apabila mengarah ke barat maka nilai *longitude* negatif. Apabila nilai *latitude* dan *longitude* telah didapatkan maka *method intent* dijalankan untuk menyimpan data *latitude* dan *longitude* kedalam *database* untuk membuat laporan.

3.3.1.3 Pseudocode Convert Data Koordinat Degrees, Minutes, Seconds To Decimal Degrees

Deklarasi :

DMS, stringD, stringM, stringS : String

Deskripsi :

```
Float result = null;
String[] DMS = stringDMS.split(",", 3);

String[] stringD = DMS[0].split("/", 2);
Double D0 = new Double(stringD[0]);
Double D1 = new Double(stringD[1]);
Double FloatD = D0/D1;

String[] stringM = DMS[1].split("/", 2);
Double M0 = new Double(stringM[0]);
Double M1 = new Double(stringM[1]);
Double FloatM = M0/M1;

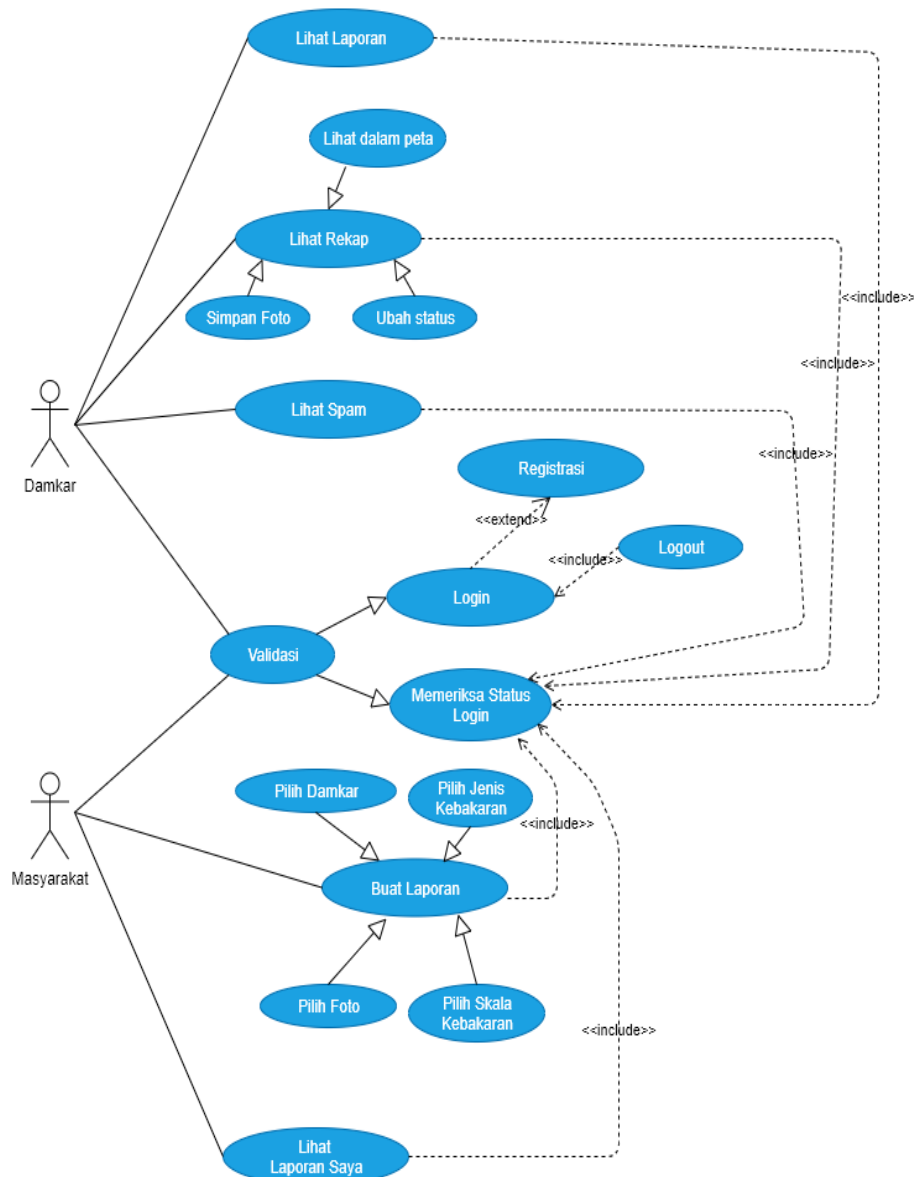
String[] stringS = DMS[2].split("/", 2);
Double S0 = new Double(stringS[0]);
Double S1 = new Double(stringS[1]);
Double FloatS = S0/S1;
result = new Float(FloatD + (FloatM/60) + (FloatS/3600));
return result;
```

Penjelasan : Deklarasi variabel dan *type* data antara lain *DMS, stringD, stringM, stringS : String*. Definisikan *float* sebagai *null*. Kemudian *string DMS* dipecah dengan pemisah tanda “,” sebanyak 3 bagian. Jalankan rumus untuk mengubah data *DMS* kedalam bentuk *Decimal Degree*. Setelah itu kirimkan kembali hasilnya.

3.3.2 Perancangan Unified Modeling Language (UML)

3.3.2.1 Use Case Diagram

**APLIKASI PELAPORAN KEBAKARAN MENGGUNAKAN METODE GEOTAG
DAN PENENTUAN RUTE MENGGUNAKAN LAYANAN GOOGLE MAPS BERBASIS ANDROID**



Gambar 3.4 Use Case Diagram Laporan Kebakaran

Tabel 3.2 Penjelasan Use Case Diagram untuk User

Nama	Deskripsi
Pilih Damkar	Use case ini merupakan proses untuk memilih Damkar yang terdekat dari titik lokasi

	berada
Buat Laporan	<i>Use case</i> ini merupakan proses untuk membuat laporan kebakaran
Lihat Laporan Saya	<i>Use case</i> ini merupakan proses untuk melihat laporan yang ditangani oleh Damkar

a. Skenario *Use Case* untuk *User/Masyarakat*

1. Nama *Use Case* : Pilih Damkar
Aktor : *User/Masyarakat*
Tujuan : Memilih Damkar yang terdekat dari posisi *user* berada

Tabel 3.3 Skenario *Use Case* Pilih Damkar

Aktor	Sistem
1. Membuka aplikasi yang akan digunakan	
	2. Menampilkan tampilan <i>splash screen</i> (gambar awal)
3. Melakukan <i>login</i>	
	4. Menampilkan halaman utama
5. Memilih tombol buat laporan	
	6. Menampilkan halaman pilih Damkar
7. Memilih Damkar sesuai jarak terdekat	

2. Nama *Use Case* : Buat laporan
Aktor : *User/Masyarakat*
Tujuan : Membuat laporan kebakaran

Tabel 3.4 Skenario *Use Case* Buat Laporan

Aktor	Sistem
1. Membuka aplikasi yang	

akan digunakan	
	2. Menampilkan tampilan <i>splash screen</i> (gambar awal)
3. Melakukan <i>login</i>	
	4. Menampilkan halaman utama
5. Memilih tombol buat laporan	
	6. Menampilkan halaman pilih Damkar
7. Memilih Damkar sesuai jarak terdekat	
	8. Menampilkan halaman buat laporan
9. Mengambil foto dan mengisi <i>form</i> selanjutnya <i>submit</i> laporan	
	10. Menampilkan halaman beranda

3. Nama *Use Case* : Lihat Laporan Saya
Aktor : User/Masyarakat
Tujuan : Mengetahui proses laporan yang ditangani oleh Damkar

Tabel 3.5 Skenario *Use Case* Lihat Laporan Saya

Aktor	Sistem
1. Membuka aplikasi yang akan digunakan	
	2. Menampilkan tampilan <i>splash screen</i> (gambar awal)
3. Melakukan <i>login</i>	
	4. Menampilkan halaman utama
5. Memilih menu laporan saya	
	6. Menampilkan halaman laporan saya

Tabel 3.6 Penjelasan *Use Case* Diagram untuk *Admin*/Damkar

Nama	Deskripsi
Lihat laporan	<i>Use case</i> ini merupakan proses untuk melihat laporan masuk
Ubah status laporan	<i>Use case</i> ini merupakan proses untuk mengubah status laporan
Ubah status Damkar	<i>Use case</i> ini merupakan proses untuk mengubah status pada Damkar
Melihat peta lokasi titik kebakaran	<i>Use case</i> ini merupakan proses untuk melihat pada peta titik kebakaran
Menghubungi pelapor	<i>Use case</i> ini merupakan proses untuk menghubungi nomor <i>handphone</i> pelapor
Buat rekap laporan	<i>Use case</i> ini merupakan proses untuk rekap laporan

b. Skenario *Use Case* untuk *Admin/Damkar*

Skenario *Use Case* untuk *Admin/Damkar* berfungsi untuk menerima laporan yang masuk, melakukan eksekusi pada laporan yang ditangani, dan merekap laporan.

- Nama *Use Case* : Lihat Laporan

Aktor : *Admin/Damkar*

Tujuan : Melihat laporan yang masuk

Tabel 3.7 Skenario *Use Case* Lihat Laporan

Aktor	Sistem
1. Membuka aplikasi yang akan digunakan	

	2. Menampilkan tampilan <i>splash screen</i> (gambar awal)
3. Memilih <i>login</i> sebagai Damkar	
	4. Menampilkan halaman <i>login</i> Damkar
5. Mengisi <i>email</i> dan <i>password</i>	
	6. Menampilkan halaman beranda pada Damkar
7. Memilih lihat laporan	
	8. Menampilkan laporan yang masuk

2. Nama *Use Case* : Ubah Status Laporan

Aktor : *Admin/Damkar*

Tujuan : Mengubah status penanganan pada laporan yang masuk

Tabel 3.8 Skenario *Use Case* Ubah Status Laporan

Aktor	Sistem
1. Memilih menu lihat rekap	
	2. Menampilkan daftar laporan yang masuk
3. Memilih laporan yang akan di ubah status	
	4. Menampilkan <i>dialog box</i>
5. Memilih menu ubah status	
	6. Menampilkan pilihan status terhadap laporan
7. Memilih status terhadap laporan	
	8. Melakukan <i>update</i> status laporan ke dalam <i>database</i>

3. Nama *Use Case* : Melihat peta lokasi titik kebakaran
 Aktor : *Admin/Damkar*
 Tujuan : Mengetahui titik lokasi kebakaran pada peta

Tabel 3.9 Skenario *Use Case* Melihat Peta Lokasi Titik Kebakaran

Aktor	Sistem
1. Memilih menu lihat laporan	
	2. Menampilkan titik lokasi kebakaran pada peta

4. Nama *Use Case* : Menghubungi pelapor
 Aktor : *Admin/Damkar*
 Tujuan : Mengecek nomor *handphone* dan menghubungi nomor *handphone* pelapor

Tabel 3.10 Skenario *Use Case* Menghubungi Pelapor

Aktor	Sistem
1. Memilih menu lihat rekap	
	2. Menampilkan daftar laporan yang masuk
3. Memilih laporan yang akan di ubah status	
	4. Menampilkan <i>dialog box</i>
5. Memilih menu ubah status	
	6. Menampilkan pilihan status terhadap laporan
7. Memilih tombol hubungi pelapor	
	8. Menampilkan nomor <i>handphone</i> pelapor

5. Nama *Use Case* : Buat rekap laporan

Aktor : Admin/Damkar
 Tujuan : mengeksekusi laporan dan merekap laporan

Tabel 3.11 Skenario *Use Case* Buat Rekap Laporan

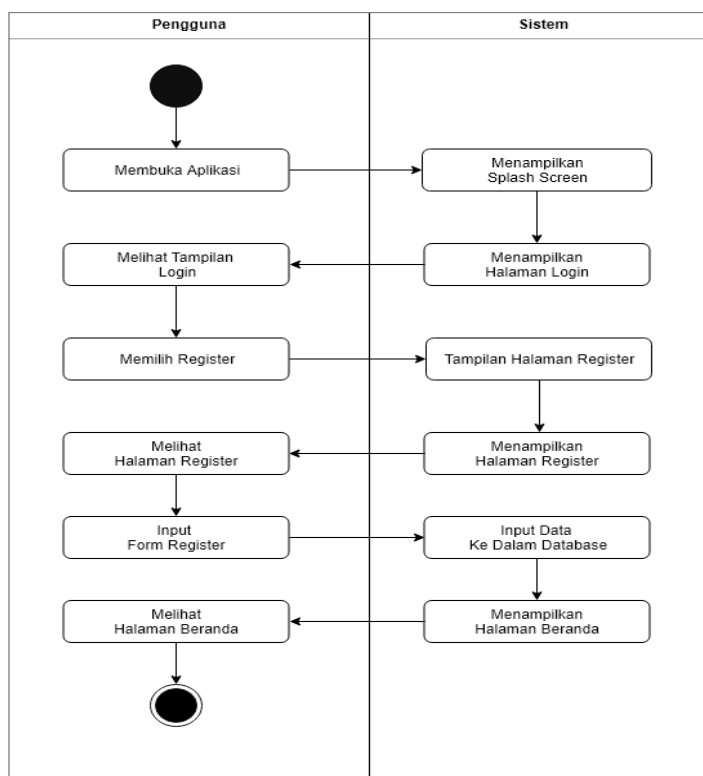
Aktor	Sistem
1. Memilih menu lihat rekap	
	2. Menampilkan daftar laporan yang masuk
3. Memilih menu print	
	4. Menampilkan web rekapitulasi

3.3.2.2 Activity diagram

Activity diagram menggambarkan berbagai alur aktivitas dalam sistem aplikasi yang sedang dirancang dan tentang bagaimana masing-masing alur dari aplikasi ini berawal. Berikut merupakan alur *activity diagram* dari Aplikasi Pelaporan Kebakaran Menggunakan Metode *Geotag* dan Penentuan Rute Menggunakan Layanan *Google Maps*.

1. Activity Diagram Aplikasi untuk User Register

Activity diagram untuk akses menu *user register* adalah seperti yang ditunjukkan pada gambar 3.5.



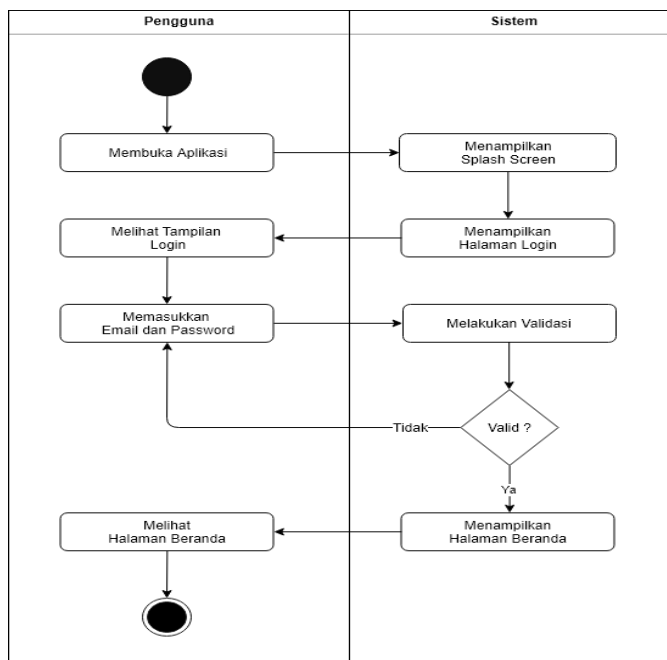
Gambar 3.5 Activity Diagram Aplikasi untuk User Register

Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

- a. Aktivitas dimulai dari pengguna membuka aplikasi Lapkar.
- b. Selanjutnya pengguna masuk ke halaman *login*.
- c. Kemudian pengguna memilih halaman *register* untuk membuat akun pengguna.
- d. Selanjutnya pengguna mengisi *form* yang telah disediakan di aplikasi.

2. Activity diagram Aplikasi untuk User Login

Activity diagram untuk akses menu *user login* adalah seperti yang ditunjukkan pada gambar 3.6.



Gambar 3.6 Activity Diagram Aplikasi Untuk User Login

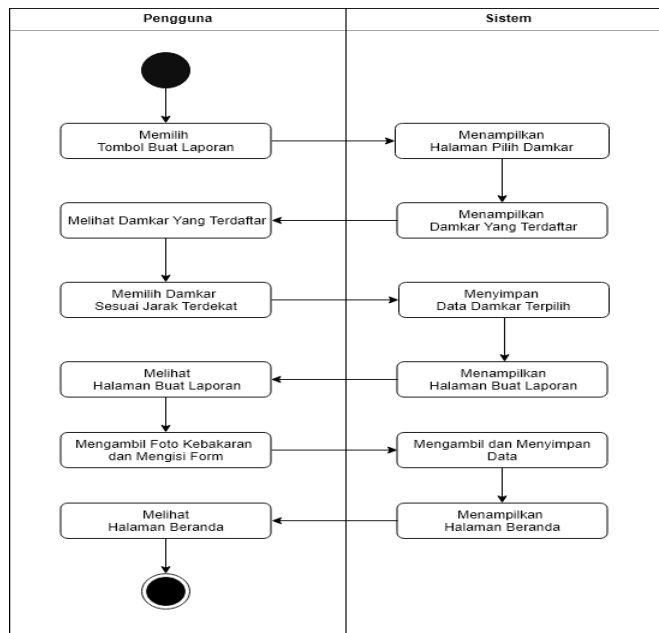
Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

- a. Aktivitas dimulai dari pengguna membuka aplikasi Lapkar.
- b. Selanjutnya pengguna masuk ke halaman *login*.

- c. Kemudian pengguna mengisi *email* dan *password* dengan benar.
- d. Selanjutnya pengguna masuk ke halaman *beranda*.

3. Activity diagram Aplikasi untuk User Membuat Laporan Kebakaran

Activity diagram untuk akses menu user membuat laporan kebakaran adalah seperti yang ditunjukkan pada gambar 3.7.



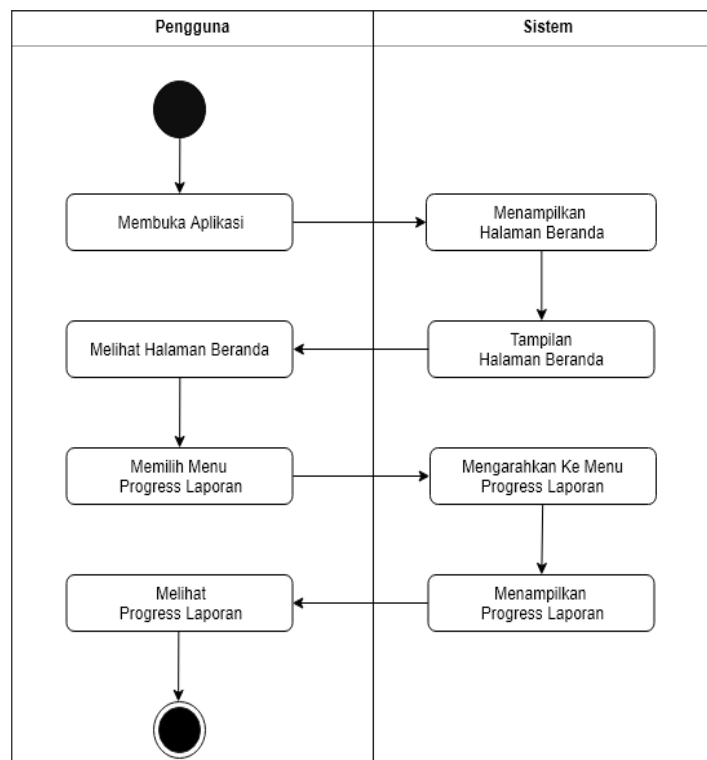
Gambar 3.7 Activity Diagram Aplikasi User Membuat Laporan Kebakaran

Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

- a. Aktivitas dimulai dari pengguna masuk di halaman beranda dan memilih tombol + untuk membuat laporan baru.
- b. Selanjutnya pengguna memilih Damkar yang terdekat dari posisi pengguna berada.
- c. Kemudian pengguna membuat laporan dengan mengambil objek kebakaran dan mengisi *field* yang disediakan di aplikasi.

4. Activity diagram Aplikasi untuk User Melihat Proses Laporan

Activity diagram untuk akses menu user melihat proses kebakaran adalah seperti yang ditunjukkan pada gambar 3.8.



Gambar 3.8 Activity

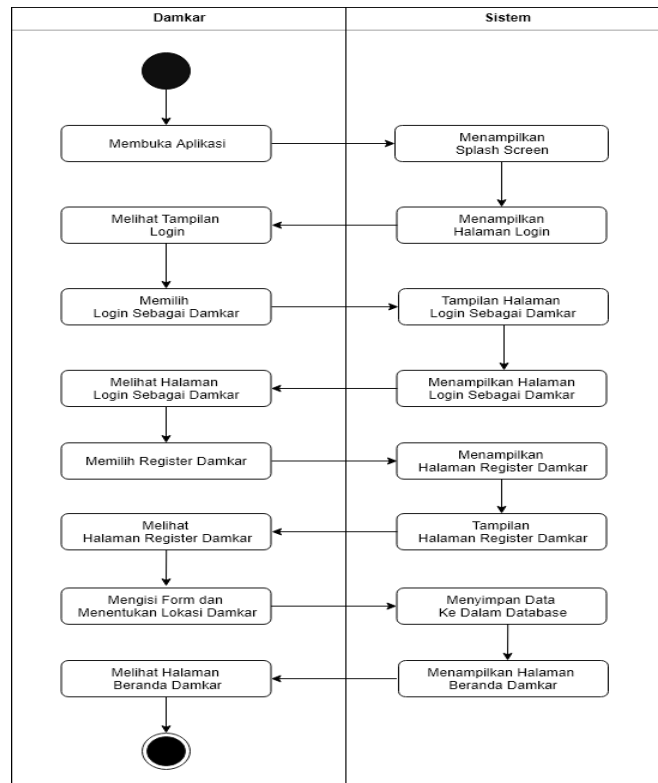
Diagram Aplikasi User Melihat Proses Laporan

Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

- a. Aktivitas dimulai dari pengguna membuka aplikasi Lapkar
- b. Selanjutnya pengguna masuk ke halaman beranda aplikasi.
- c. Kemudian pengguna memilih menu lihat laporan saya.
- d. Selanjutnya pengguna dapat melihat proses laporan yang telah dilaporkan ke Damkar pilihan pengguna.

5. *Activity diagram Aplikasi untuk Damkar Register*

Activity diagram untuk akses menu Damkar *register* adalah seperti yang ditunjukkan pada gambar 3.9.



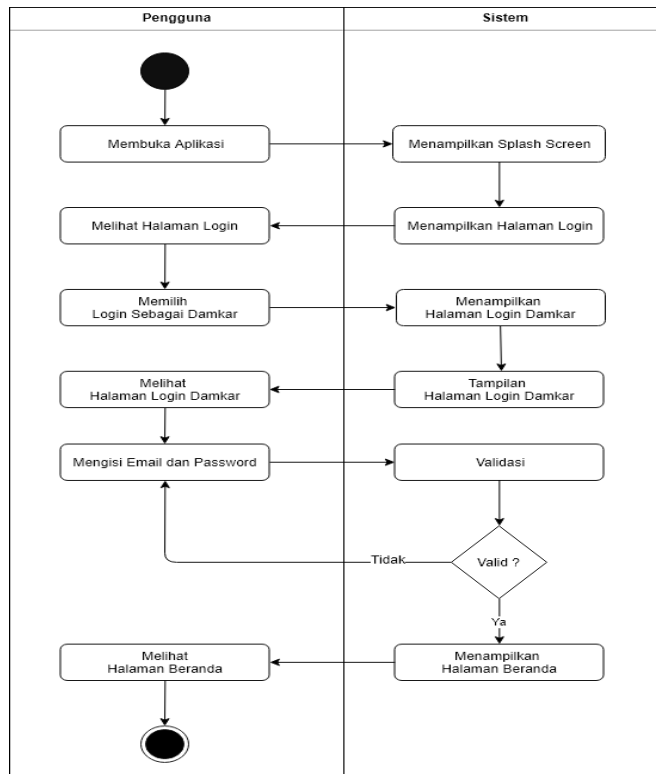
Gambar 3.9 Activity Diagram Aplikasi Damkar Register

Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

- a. Aktivitas dimulai dari Damkar membuka aplikasi Lapkar
- b. Selanjutnya Damkar masuk ke halaman *login*.
- c. Kemudian Damkar memilih menu *register*.
- d. Selanjutnya Damkar mengisi *form* di halaman *register*.

6. Activity diagram Aplikasi untuk Damkar Login

Activity diagram untuk akses menu Damkar *login* adalah seperti yang ditunjukkan pada gambar 3.10.



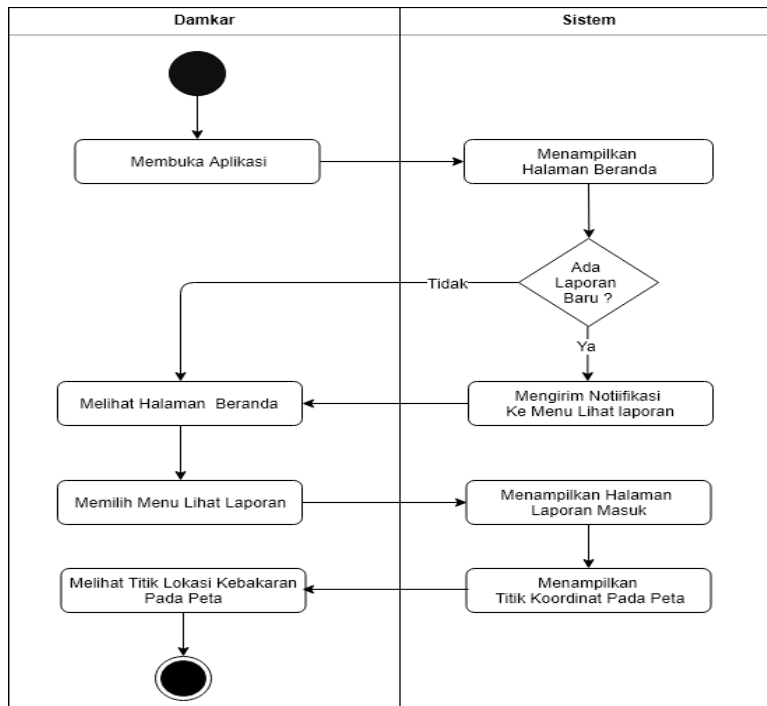
Gambar 3.10 Activity Diagram Aplikasi Damkar Login

Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

- Aktivitas dimulai dari Damkar membuka aplikasi Lapkar.
- Selanjutnya Damkar masuk ke halaman *login*.
- Kemudian Damkar mengisi *email* dan *password* dengan benar.
- Selanjutnya Damkar masuk ke halaman *beranda*.

7. Activity diagram Aplikasi untuk Damkar Notifikasi Dan Melihat Laporan Masuk

Activity diagram untuk akses menu Damkar menerima notifikasi dan melihat laporan masuk adalah seperti yang ditunjukkan pada gambar 3.11.



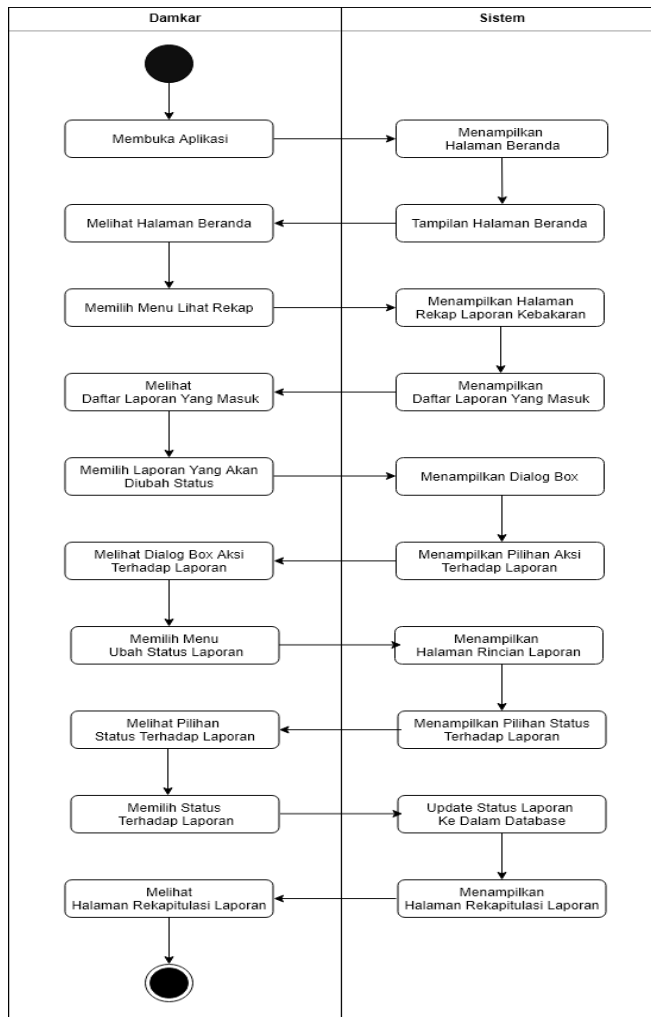
Gambar 3.11 Activity Diagram Aplikasi Menerima Notifikasi dan Melihat Laporan Masuk

Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

- a. Aktivitas dimulai dari Damkar membuka aplikasi Lapkar.
- b. Selanjutnya Damkar masuk ke halaman beranda.
- c. Kemudian Damkar menerima notifikasi laporan masuk.
- d. Selanjutnya Damkar melihat laporan masuk pada menu lihat laporan.

8. Activity diagram Aplikasi untuk Damkar Ubah Status Laporan

Activity diagram untuk akses Damkar ubah status laporan adalah seperti yang ditunjukkan pada gambar 3.12.



Gambar 3.12 Activity Diagram Damkar Mengubah Status Laporan

Alur aktivitas yang terjadi akan dijelaskan sebagai berikut :

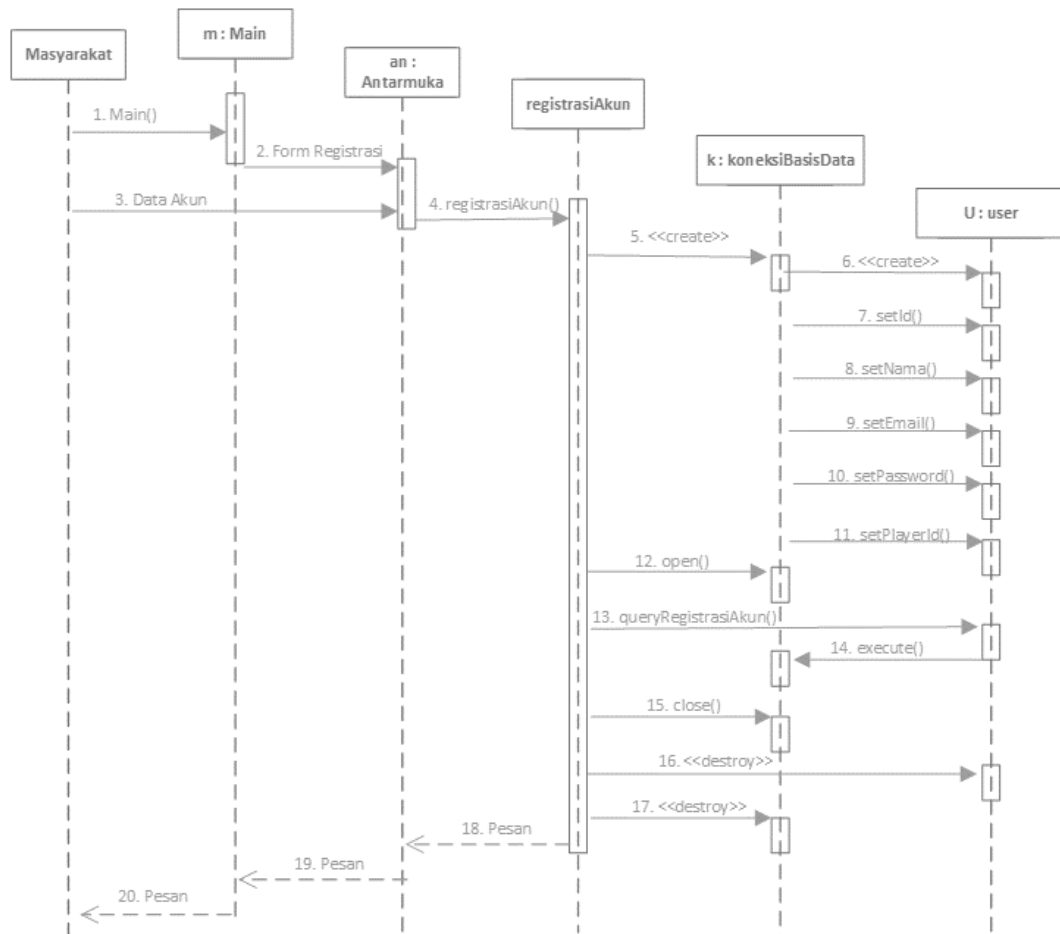
- Aktivitas dimulai dari Damkar membuka aplikasi Lapkar.
- Selanjutnya Damkar masuk ke halaman beranda.
- Kemudian Damkar memilih menu lihat rekap dan memilih laporan yang akan diubah statusnya
- Selanjutnya Damkar memilih status terhadap laporan.

3.3.2.3 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan interaksi antar objek dalam waktu yang berurutan. *Sequence diagram* juga menunjukkan alur yang digunakan oleh suatu objek. Berikut merupakan alur *sequence diagram* dari

aplikasi Pelaporan Kebakaran Menggunakan Fitur *Geotag* Dan Penentuan Rute Menggunakan Layanan *Google Maps*.

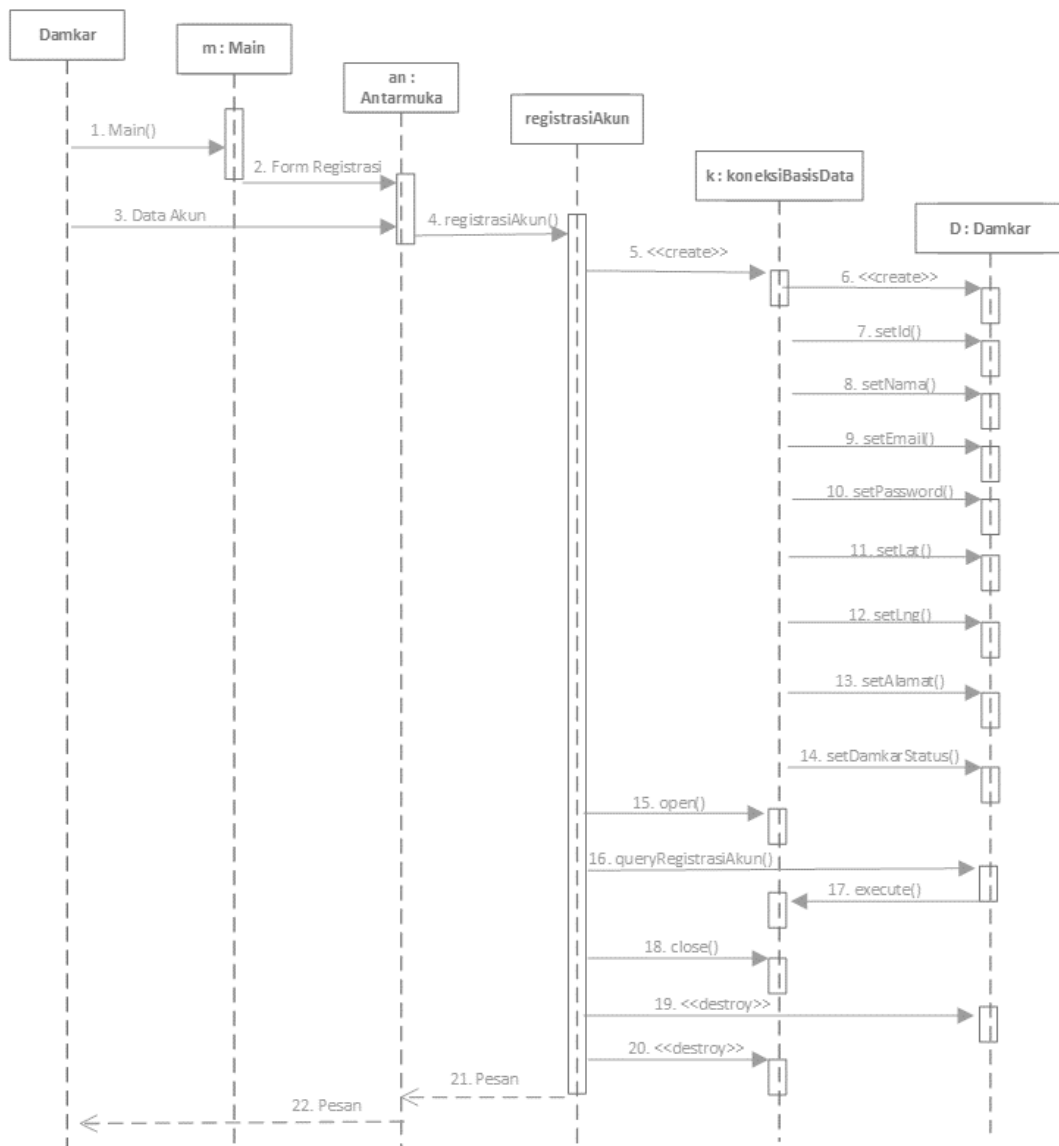
3.3.2.3.1 Use Case : Registrasi User



Gambar 3.13 Sequence Diagram Use Case Registrasi User

Sequence diagram pada gambar 3.13 ini adalah proses *user*/masyarakat dalam membuat akun dengan cara registrasi *user*. Proses registrasi berisi untuk masyarakat dapat membuat akun pada aplikasi dengan mengisi form2 yang disediakan pada aplikasi.

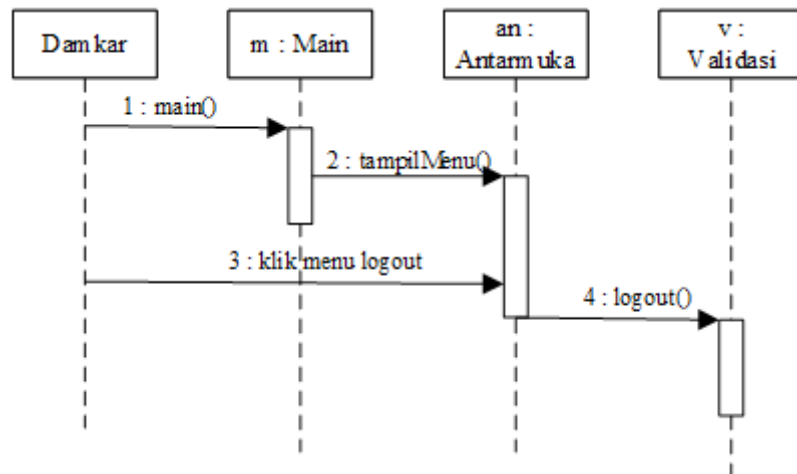
3.3.2.3.2 Use Case : Registrasi Damkar



Gambar 3.14 Sequence Diagram Use Case Registrasi Damkar

Sequence diagram pada gambar 3.14 ini adalah proses Damkar dalam membuat akun dengan cara registrasi Damkar. Proses registrasi berisi untuk masyarakat dapat membuat akun pada aplikasi dengan mengisi form2 yang disediakan pada aplikasi.

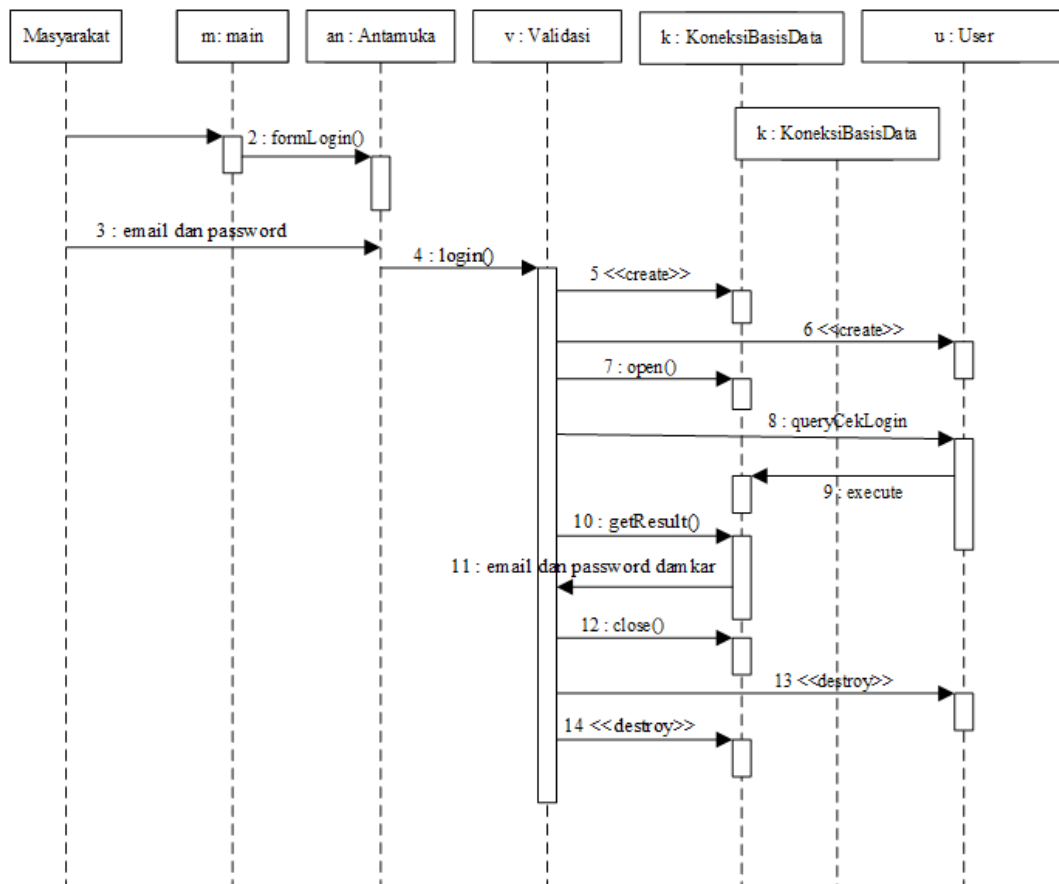
3.3.2.3.3 Use Case : logout



Gambar 3.15 Sequence Diagram Use Case Logout

Proses *logout* hanya mengosongkan *session status login*

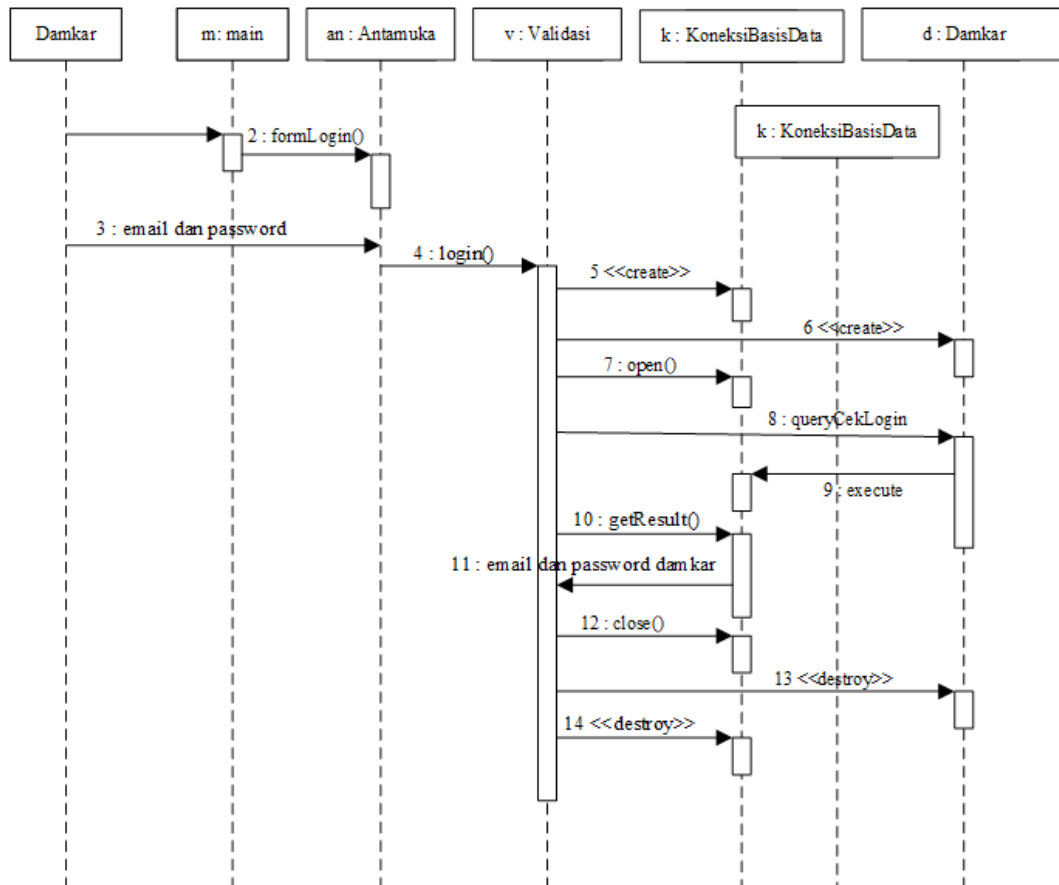
3.3.2.3.4 Use Case : Memeriksa Status Login User



Gambar 3.16 Sequence Diagram Use Case Memeriksa Status Login User

Proses memeriksa status *login user* berisi untuk memeriksa apakah pengguna perangkat lunak sudah melakukan *login*. Proses ini digunakan oleh *use case* lain sehingga akan menjadi bagian diagram *sequence* dari *use case* lain yang menggunakannya.

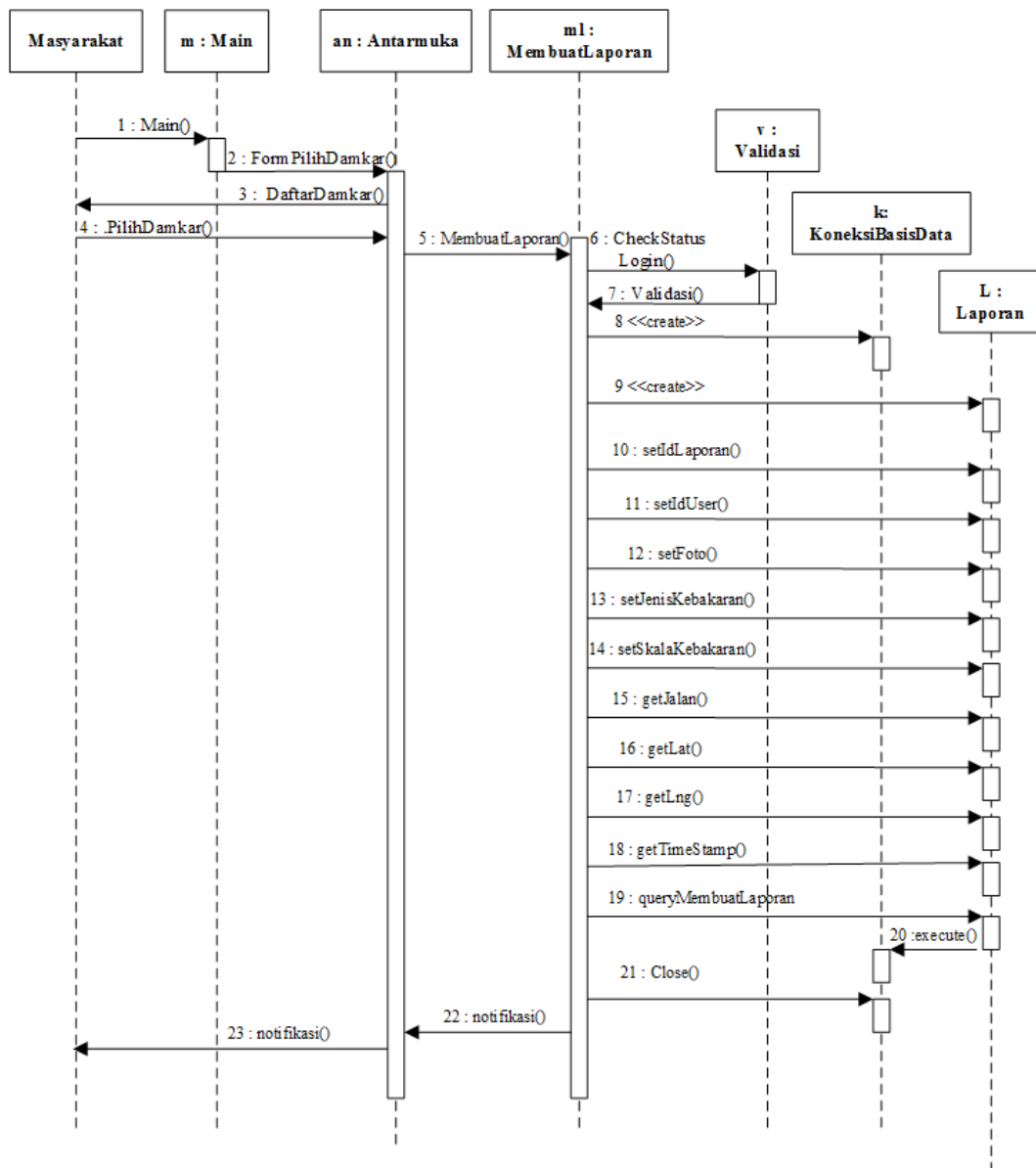
3.3.2.3.5 Use Case : Memeriksa Status Login Damkar



Gambar 3.17 Sequence Diagram Use Case Memeriksa Status Login Damkar

Proses memeriksa status *login Damkar* berisi untuk memeriksa apakah pengguna perangkat lunak sudah melakukan *login*. Proses ini digunakan oleh *use case* lain sehingga akan menjadi bagian diagram *sequence* dari *use case* lain yang menggunakannya.

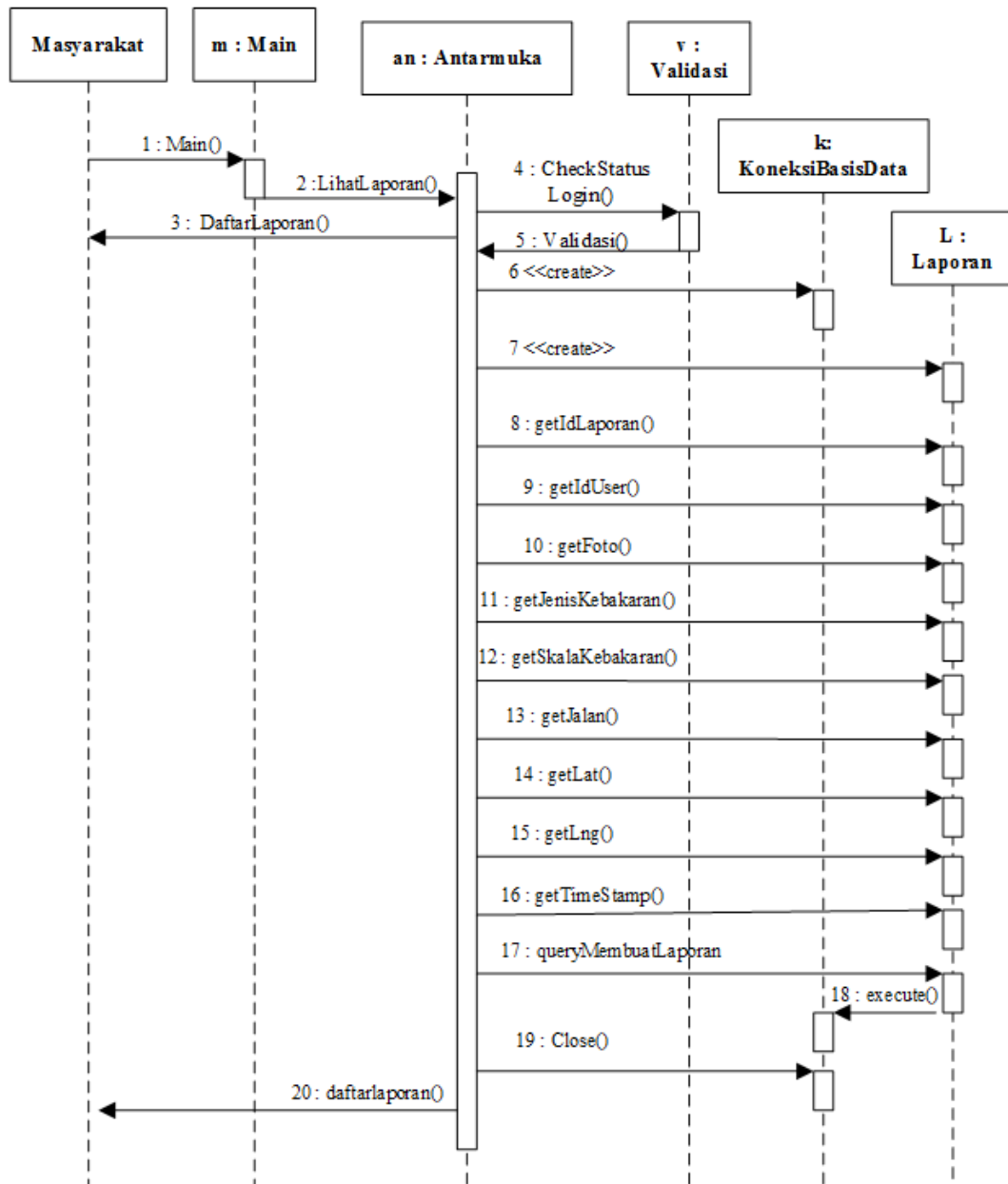
3.3.2.3.6 Use Case : Membuat Laporan Kebakaran Pada Masyarakat



Gambar 3.18 Sequence Diagram Use Case Membuat Laporan Kebakaran Pada Masyarakat

Proses Membuat Laporan Kebakaran diawali dengan proses memilih damkar yang terdekat dari posisi pelapor. Selanjutnya Pelapor dapat membuat laporan kebakaran dengan cara memasukkan foto *geotag*, memilih objek kebakaran dan memilih skala kebakaran.

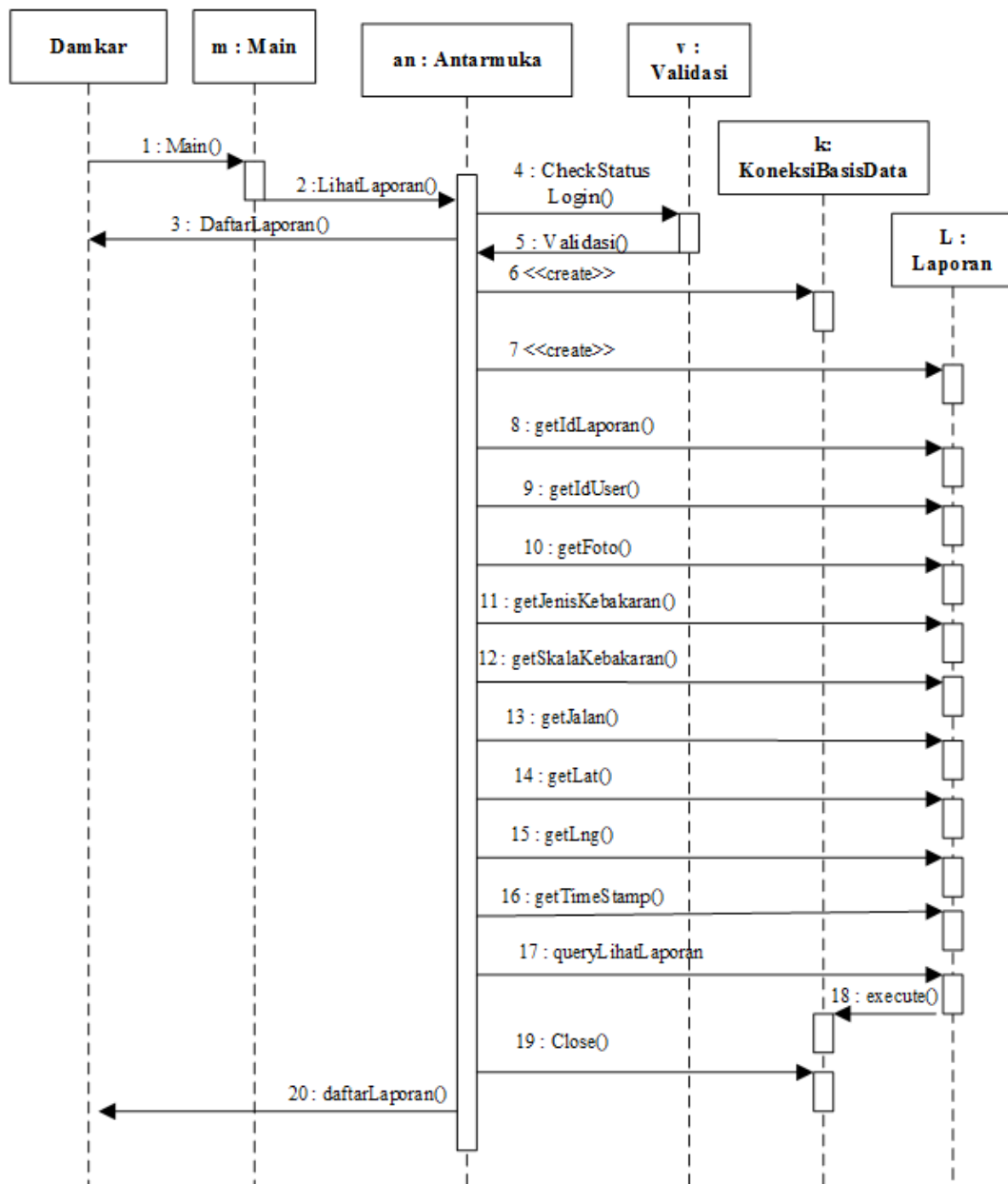
3.3.2.3.7 Use Case : Lihat Laporan Saya Pada Masyarakat



Gambar 3.19 Sequence Diagram Use Case Lihat Laporan Kebakaran Pada Masyarakat

Proses melihat laporan diawali dengan memilih menu lihat laporan yang dapat dilihat pada diagram *sequence* lihat laporan.

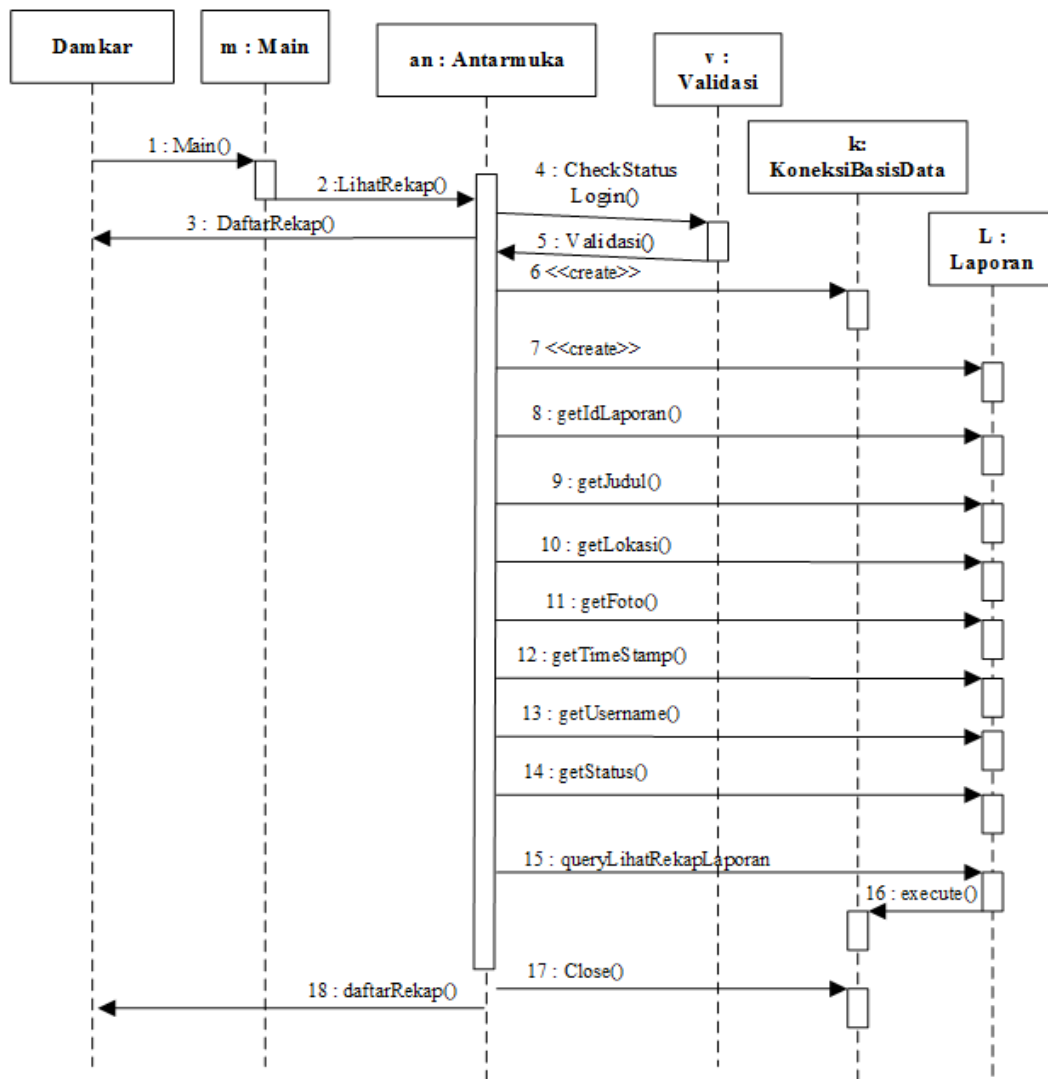
3.3.2.3.8 Use Case : Lihat Laporan Pada Damkar



Gambar 3.20 Sequence Diagram Use Case lihat Laporan Pada Damkar

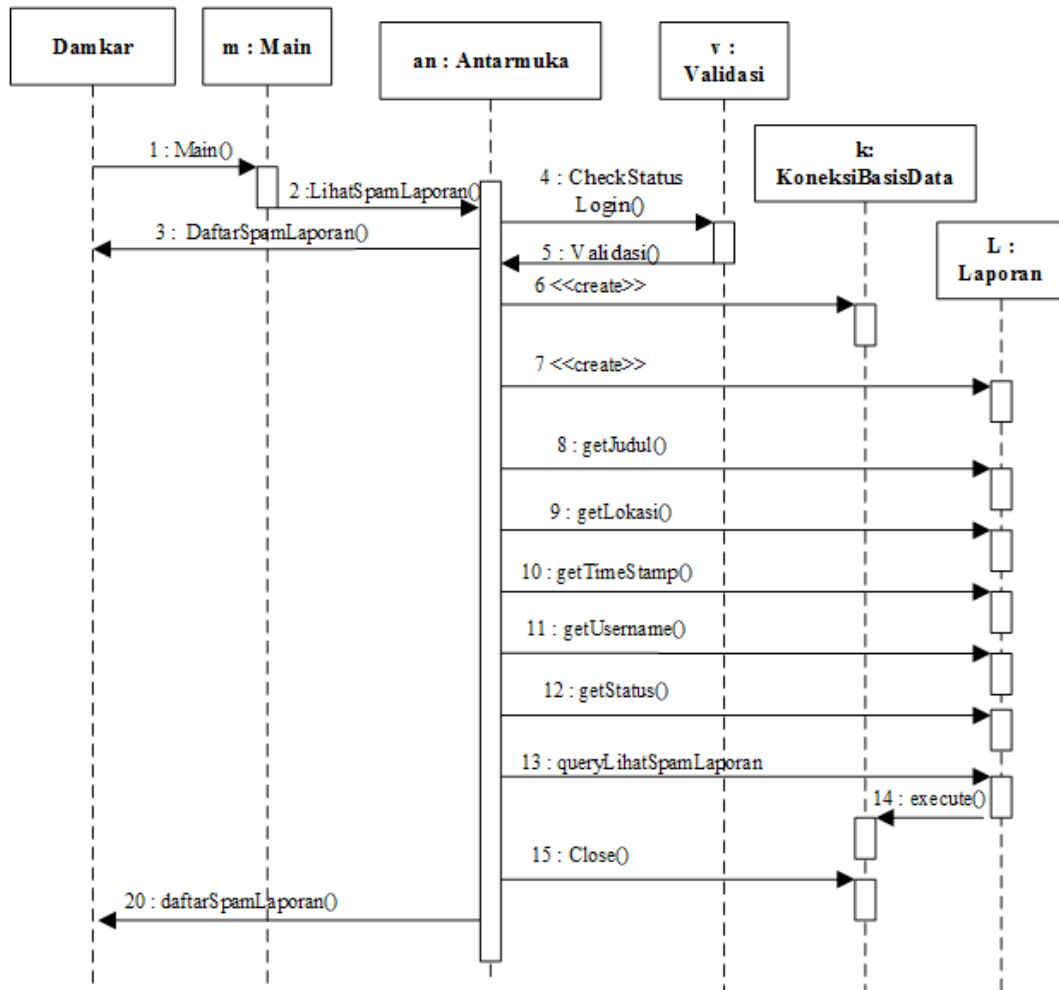
Proses melihat laporan masuk dapat diawali dengan memilih menu lihat laporan yang dapat dilihat pada diagram *sequence* lihat laporan pada damkar.

3.3.2.3.9 Use Case : Lihat Rekap Pada Damkar



Gambar 3.21 Sequence Diagram Use Case Rekap Laporan Pada Damkar

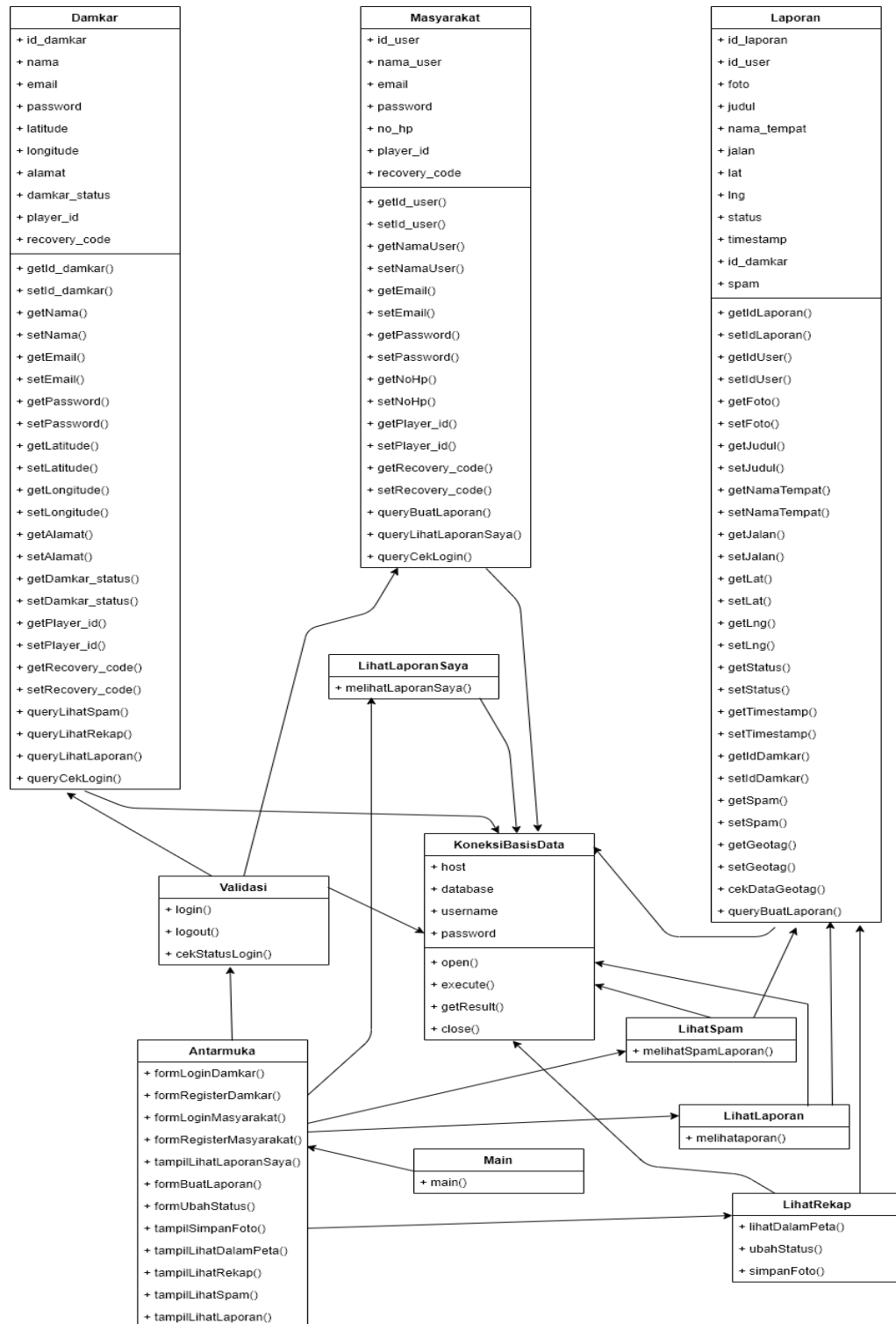
3.3.2.3.10 Use Case : Lihat Spam Laporan



Gambar 3.22 Sequence Diagram Use Case Lihat Spam Laporan Pada Damkar

3.3.2.4 Class Diagram

Class diagram yang digunakan untuk melakukan perancangan struktur kelas-kelas yang terdapat dalam aplikasi yang dibuat ditunjukkan pada Gambar 3.21.



Gambar 3.23 Class Diagram Aplikasi Lapkar

3.3.2.5 Perancangan Basis Data

Berikut merupakan spesifikasi tabel basis data yang digunakan :

1. Tabel *User*

Tabel 3.12 Struktur Tabel *User*

Nama Field	Tipe Data	Key	Keterangan	Extra
<i>Id_user</i>	<i>Integer(11)</i>	<i>Primary Key</i>	<i>Not Null</i>	<i>Auto Increment</i>
<i>nama_user</i>	<i>Text</i>		<i>Not Null</i>	
<i>Email</i>	<i>Text</i>		<i>Not Null</i>	
<i>Password</i>	<i>Text</i>		<i>Not Null</i>	
<i>no_hp</i>	<i>Varchar(13)</i>		<i>Not Null</i>	
<i>player_id</i>	<i>Text</i>		<i>Not Null</i>	

2. Tabel Laporan

Tabel 3.13 Struktur Tabel Laporan

Nama Field	Tipe Data	Key	Keterangan	Extra
<i>id_laporan</i>	<i>Integer(11)</i>	<i>Primary Key</i>	<i>Not Null</i>	<i>Auto Increment</i>
<i>id_user</i>	<i>Integer(11)</i>	<i>Foreign Key</i>	<i>Not Null</i>	
<i>Foto</i>	<i>Text</i>		<i>Not Null</i>	
<i>Judul</i>	<i>Text</i>		<i>Not Null</i>	
<i>nama_tempat</i>	<i>Text</i>		<i>Not Null</i>	
<i>Jalan</i>	<i>Text</i>		<i>Not Null</i>	
<i>Lat</i>	<i>Text</i>		<i>Not Null</i>	
<i>Lng</i>	<i>Text</i>		<i>Not Null</i>	
<i>status</i>	<i>Varchar(2)</i>		<i>Not Null</i>	
<i>timestamp</i>	<i>Timestamp</i>		<i>Not Null</i>	
<i>id_Damkar</i>	<i>Integer(11)</i>	<i>Foreign Key</i>	<i>Not Null</i>	
<i>spam</i>	<i>Varchar(1)</i>		<i>Not Null</i>	

3. Tabel Damkar

Tabel 3.14 Struktur Tabel Damkar

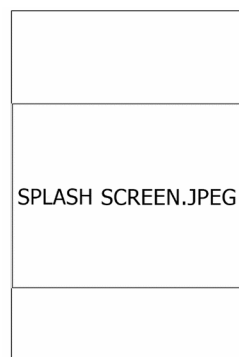
Nama Field	Tipe Data	Key	Keterangan	Extra
id_Damkar	Integer(11)	<i>Primary Key</i>	<i>Not Null</i>	<i>Auto Increment</i>
nama	<i>Text</i>		<i>Not Null</i>	
<i>email</i>	<i>Text</i>		<i>Not Null</i>	
<i>password</i>	<i>Text</i>		<i>Not Null</i>	
<i>lat</i>	<i>Text</i>		<i>Not Null</i>	
<i>lng</i>	<i>Text</i>		<i>Not Null</i>	
alamat	<i>Text</i>		<i>Not Null</i>	
Damkar_status	<i>Varchar(1)</i>		<i>Not Null</i>	

3.4 Pembuatan Perangkat Lunak

Antarmuka merupakan tampilan dari suatu program aplikasi yang berperan sebagai media komunikasi yang digunakan sebagai sarana berdialog antara aplikasi dengan pengguna. Sistem yang akan dibangun diharapkan menyediakan *interface* yang mudah dipahami dan digunakan oleh *user*. Berikut ini perancangan antarmuka dari *user/masyarakat*:

1. Perancangan Aktivitas Pilihan Menu Utama

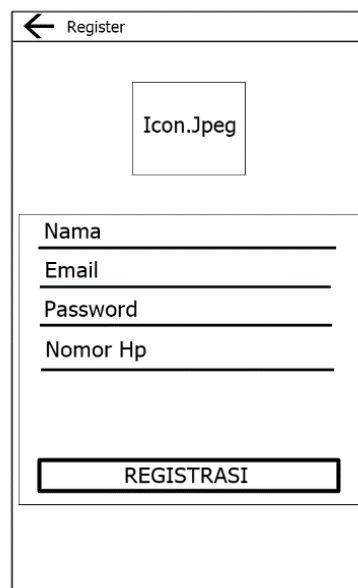
Pada aktivitas ini pengguna dihadapkan pada tampilan awal setelah aplikasi dijalankan atau dieksekusi. Berikut adalah rancangan aktivitas menu utama dapat dilihat pada gambar 3.24.

**Gambar 3.24** Rancangan Tampilan Halaman *Splash Screen* Aplikasi

Pada tampilan awal aplikasi ini pengguna akan dibawa pada tampilan awal aplikasi yang berupa *splash screen* dari aplikasi yang dirancang. Dapat dilihat beberapa *highlight* dari aplikasi yang berjalan di menu utama ini.

2. Perancangan Aktivitas Menu *Register*

Pada aktivitas menu *register* ini, pengguna harus mendaftarkan terlebih dahulu akun-nya supaya pengguna dapat melakukan *login*. Berikut adalah rancangan aktivitas menu *register* dapat dilihat pada Gambar 3.25.



The image shows a wireframe of a mobile application's 'Register' screen. At the top left is a back arrow icon, and next to it is the title 'Register'. Below the title is a square placeholder for a profile picture labeled 'Icon.Jpeg'. Underneath the profile picture is a group of four input fields, each with a label above it: 'Nama', 'Email', 'Password', and 'Nomor Hp'. At the bottom of the form is a rectangular button with the text 'REGISTRASI'.

Gambar 3.25 Rancangan Tampilan Halaman Menu *Register*

3. Perancangan Aktivitas Menu *Login*

Pada aktivitas menu *login* ini, pengguna dapat memasukkan *email* dan *password* pada *interface* yang dirancang. Pengguna harus memasukkan *email* dan *password* dengan benar, jika pengguna memasukkan *email/password* dengan salah maka pengguna tidak dapat melakukan *login*. Berikut adalah rancangan aktivitas menu *login* dapat dilihat pada gambar 3.26.

Lapkar	LOGIN DAMKAR
	
<div>Email</div> <div>Password</div>	
<div>LOGIN</div>	
<p>Belum punya akun?</p> <p>Registrasi User</p> <p>Lupa Password?</p>	

Gambar 3.26 Rancangan Tampilan Halaman Menu *Login*

4. Perancangan Aktivitas Menu Beranda

Pada aktivitas menu beranda ini, pengguna dapat melihat laporan-laporan yang telah dilaporkan oleh masyarakat. Selain itu pengguna dapat melihat proses laporan yang ada di menu laporan saya dan di *layout* ini juga pengguna dapat membuat laporan dengan cara memilih tombol + yang telah disediakan pada *layout* ini. Berikut adalah perancangan aktivitas menu beranda dapat dilihat pada gambar 3.27

Lapkar	Laporan Saya	Logout
<p>Nama : alex Email : alex@mail.com Nomor : 085858585858 Hp</p>		
		
<p>kebakaran di supadio</p> <p>bandara supadio pontianak Jl. Bandara Supadio No.1, Limbung, Kec. Sungai Raya, Kabupaten Kubu Raya, Kalimantan Barat 78391, Indonesia</p> <p>Dituju ke: Damkar B</p> <p style="text-align: right;">Dari : tuah, 2019-05-14 07:40:43</p>		
		

Gambar 3.27 Rancangan Tampilan Halaman Menu Beranda

5. Perancangan Aktivitas Pilih Damkar

Pada aktivitas menu pilih Damkar ini, pengguna dapat memilih Damkar yang terdekat dari posisi pengguna berada. Selain itu pengguna dapat mengetahui nama Damkar yang tersedia di dalam aplikasi dan dapat membuat laporan ke Damkar yang akan dipilih oleh pengguna untuk membuat laporan kebakaran. Berikut adalah perancangan aktivitas menu pilih Damkar dapat dilihat pada gambar 3.28

← Pilih Damkar	
Nama Damkar	: Kapuas Bhakti
Lokasi	: Jl. Dr. Wahidin. S No. 40, Sungai Jawi, Pontianak Kota, Kota Pontianak, Kalimantan Barat 78113, Indonesia
Jarak	: 3.50 Km dari posisi anda
Nama Damkar	: Panca Bhakti
Lokasi	: Jl. Danau Sentarum No. 64, Sungai Jawi, Pontianak Kota, Kota Pontianak, Kalimantan Barat 78119, Indonesia
Jarak	: 2.20 Km dari posisi anda
Nama Damkar	: Karya Baru
Lokasi	: Jl. Dr. Sutomo No. 8, Sungai Jawi, Pontianak Kota, Kota Pontianak, Kalimantan Barat 78116, Indonesia
Jarak	: 1.20 Km dari posisi anda

Gambar 3.28 Rancangan Tampilan Halaman Menu Pilih Damkar

6. Perancangan Aktivitas Buat Laporan Kebakaran

Pada aktivitas menu buat laporan ini, pengguna membuat laporan tentang kebakaran kepada Damkar yang dituju. Pada *layout* ini pengguna mengambil gambar atau objek kebakaran, ketika pengguna mengambil foto kebakaran maka sudah otomatis data alamat sudah terisi. Selanjutnya pengguna memilih objek apa yang kebakaran dan pengguna dapat memilih kebakaran dalam skala kecil, sedang, dan besar sehingga Damkar dapat mengetahui objek kebakaran, skala kebakaran dan alamat yang diketahui dari sebuah laporan pengguna. Berikut adalah perancangan aktivitas menu buat laporan dapat dilihat pada gambar 3.29

Gambar 3.29 Rancangan Tampilan Halaman Menu Buat Laporan

7. Perancangan Aktivitas Laporan Saya

Pada aktivitas laporan saya ini, pengguna dapat mengetahui tanggapan laporan yang telah dibuat oleh pengguna. Pengguna juga dapat mengetahui status proses laporannya dan dapat mengetahui proses laporan secara *realtime* dapat dilihat pada gambar 3.30

Gambar 3.30 Rancangan Tampilan Halaman Proses Laporan Saya

3.5 Pengujian Aplikasi

Pada tahap ini, aplikasi yang telah dibuat akan diuji untuk mengetahui apakah aplikasi yang dibuat sudah berjalan sesuai dengan yang diharapkan. Pengujian yang dilakukan adalah dengan menggunakan metode pengujian *Skala Likert*, dan Pengujian Kompatibilitas.

3.5.1 Pengujian *Skala Likert*

Kuesioner berisi 15 pertanyaan yang dikelompokkan menjadi 3 aspek yang digunakan dalam pengujian aplikasi tersebut, yaitu aspek rekayasa perangkat lunak, aspek fungsionalitas dan aspek komunikasi visual. Kuesioner dibagikan kepada 10 responden yang merupakan pengguna *smartphone* dengan sistem operasi *Android*, terdiri dari masyarakat sekitar. Hasil kuesioner dirangkum berdasarkan tiga aspek yang diujikan.

3.5.1.1 Aspek Rekayasa Perangkat Lunak

Hasil kuesioner aspek rekayasa perangkat lunak dirangkum dalam sebuah tabel seperti pada tabel 3.15.

Tabel 3.15 Tabel Hasil Kuesioner Aspek Rekayasa Perangkat Lunak

No	Aspek Rekayasa Perangkat Lunak	Tanggapan					Total
		1	2	3	4	5	
1	Berjalannya aplikasi pada <i>smartphone</i>						
2	Kemudahan menjalankan aplikasi pada <i>smartphone</i>						
3	Kelancaran menjalankan aplikasi pada <i>smartphone</i>						
4	Kemudahan mengakses menu-menu halaman pada aplikasi						
5	Kenyamanan dalam menggunakan aplikasi secara keseluruhan						
Jumlah							
Persentase (%)							

Keterangan : 1 = Sangat Buruk 3 = Cukup baik 5 = Sangat Baik
 2 = Buruk 4 = Baik

3.5.1.2 Aspek Fungsionalitas

Hasil kuesioner aspek fungsionalitas dirangkum dalam sebuah tabel seperti pada Tabel 3.16.

Tabel 3.16 Tabel Hasil Kuesioner Aspek Fungsionalitas

No	Aspek Rekayasa Perangkat Lunak	Tanggapan					Total
		1	2	3	4	5	
1	Kinerja aplikasi saat memuat data						
2	Kinerja aplikasi saat menampilkan data						
3	Kinerja aplikasi pada sistem notifikasi						
4	Kinerja aplikasi saat membuat laporan kebakaran						
5	Kinerja aplikasi saat menggunakan layanan <i>Google Maps</i> untuk menunjukkan rute perjalanan ke titik koordinat pelapor						
Jumlah							
Persentase (%)							

Keterangan : 1 = Sangat Buruk 3 = Cukup baik 5 = Sangat Baik
 2 = Buruk 4 = Baik

3.5.1.3 Aspek Komunikasi Visual

Hasil kuesioner aspek komunikasi visual dirangkum dalam sebuah tabel seperti pada Tabel 3.17.

Tabel 3.17 Tabel Hasil Kuesioner Aspek Komunikasi Visual

No	Aspek Rekayasa Perangkat Lunak	Tanggapan					Total
		1	2	3	4	5	
1	Tampilan antarmuka aplikasi						
2	Tampilan menu aplikasi						
3	Kemudahan membaca jenis dan ukuran huruf pada aplikasi						
4	Kombinasi warna pada tampilan aplikasi						
5	Respon (<i>feedback</i>) aplikasi ketika mengklik tombol atau memilih						

BAB IV

HASIL PERANCANGAN DAN PENGUJIAN APLIKASI

Pada awal bab ini dijelaskan bagaimana hasil perancangan terhadap aplikasi yang telah selesai dibuat dan dapat diimplementasikan. Proses implementasi pada aplikasi ini dilakukan dengan melihat spesifikasi kebutuhan aplikasi dan memberi petunjuk kepada pengguna tentang cara mengakses aplikasi, sehingga memudahkan pengguna dalam mengakses aplikasi.

Selanjutnya hasil dari perancangan dan pembuatan aplikasi dipaparkan melalui gambar atau *screenshoot* aplikasi yang disertai dengan penjelasannya. Langkah berikutnya dijelaskan bagaimana hasil dari pengujian aplikasi. Pengujian aplikasi dalam penelitian ini menggunakan beberapa metode antara lain dengan menggunakan metode skala *likert*, di mana diberikan *input* berupa suatu set data yang didapat menggunakan borang. Borang tersebut kemudian diisi oleh 10 responden dengan pembagian 5 responden mengisi borang dengan akses pelapor atau masyarakat dan 5 responden mengisi borang dengan hak akses Damkar.

4.1 Hasil Perancangan

4.1.1 Proses Akses Aplikasi

Setelah semua spesifikasi kebutuhan aplikasi terpenuhi oleh pengguna, selanjutnya akan dijelaskan proses akses aplikasi pelaporan kebakaran sudah terpasang atau ter-*install* pada *smartphone*. Dalam proses ini dikondisikan bahwa aplikasi sudah berhasil dipasang dan dapat diakses.

Langkah-langkah untuk melakukan akses aplikasi dari sisi pelapor dapat diuraikan sebagai berikut :

1. Pengguna membuka aplikasi pelaporan kebakaran dan memastikan bahwa *GPS* pada *smartphone* yang digunakan telah aktif.
2. Kemudian akan muncul tampilan *splash screen* sebagai halaman utama setelah pengguna membuka aplikasi.
3. Selanjutnya pengguna akan diarahkan menuju halaman *login*, pada halaman ini pengguna harus memiliki akun untuk dapat masuk ke

dalam aplikasi. Jika pengguna belum memiliki akun maka pengguna dapat memilih menu registrasi akun dan mengisi *form* registrasi.

4. Setelah memiliki akun, pengguna dapat melakukan *login* dengan cara memasukkan *email* dan *password* yang dimiliki dengan benar. Jika pengguna memasukkan data yang salah, maka pengguna akan mendapatkan *toast login* anda salah. Kemudian pengguna harus memasukkan kembali data dengan benar agar dapat masuk ke halaman beranda pada aplikasi.
5. Setelah melakukan *login*, pengguna akan diarahkan menuju halaman beranda. Pada halaman beranda, pengguna dapat membuat laporan dengan cara memilih tombol (+) untuk membuat laporan. Setelah itu pengguna dapat memilih Damkar yang tersedia dan yang terdekat dari posisi pengguna saat membuat laporan.
6. Setelah memilih Damkar, pengguna akan diarahkan menuju ke halaman buat laporan untuk mengisi *form* laporan tentang kebakaran berupa foto *geotag*, pilih objek kebakaran, pilih skala kebakaran, serta nama jalan. Pada kolom nama jalan, pengguna tidak perlu mengisi secara manual karena data nama jalan telah otomatis terisi ketika foto *geotag* diambil. Selanjutnya jika semua kolom sudah terisi, pengguna dapat memilih tombol lapor untuk mengirimkan laporan.
7. Kemudian pengguna dapat melihat laporan yang pernah dibuat pada halaman laporan saya. Jika terdapat laporan yang sudah diubah status laporannya, maka pengguna akan mendapatkan notifikasi dan pengguna dapat melihat tanggapan laporan dari Damkar secara *realtime*.
8. Pengguna dapat keluar atau *logout* dari aplikasi dengan cara memilih tombol *logout*.

Selanjutnya langkah-langkah untuk melakukan akses aplikasi dari sisi Damkar dapat diuraikan sebagai berikut :

1. Damkar membuka aplikasi pelaporan kebakaran dan memastikan bahwa *GPS* pada *smartphone* yang digunakan telah aktif.
2. Setelah membuka aplikasi, pihak Damkar akan muncul halaman *splash screen*.
3. Kemudian pihak Damkar akan diarahkan ke halaman *login*, pada tahap ini pihak Damkar dapat memilih tombol *login* sebagai Damkar untuk dapat masuk ke dalam aplikasi dengan hak akses sebagai Damkar.
4. Setelah itu pihak Damkar harus membuat akun Damkar terlebih dahulu dengan memilih menu *register* kemudian mengisi *form* registrasi.
5. Setelah memiliki akun, pihak Damkar dapat melakukan *login* dengan mengisi *email* dan *password* pada halaman *login*.
6. Setelah berhasil *login*, terdapat beberapa menu yang akan tampil pada halaman beranda di sisi Damkar. Menu-menu tersebut antara lain diuraikan sebagai berikut :
 - *Logout*
Pada menu *logout*, Damkar dapat memilih *logout* jika Damkar ingin keluar atau mengakhiri sesi aplikasi.
 - *Lihat Laporan*
Pada menu lihat laporan, Damkar dapat mengetahui laporan yang masuk pada menu lihat laporan
 - *Lihat Spam*
Pada menu *spam*, Damkar dapat membuang laporan ke dalam menu *spam* jika ada laporan yang palsu.

4.1.2 Tampilan Antarmuka Aplikasi Pengguna / Masyarakat

4.1.2.1 Tampilan Antarmuka Halaman *Splash Screen* Aplikasi *Android* untuk Pengguna / Masyarakat

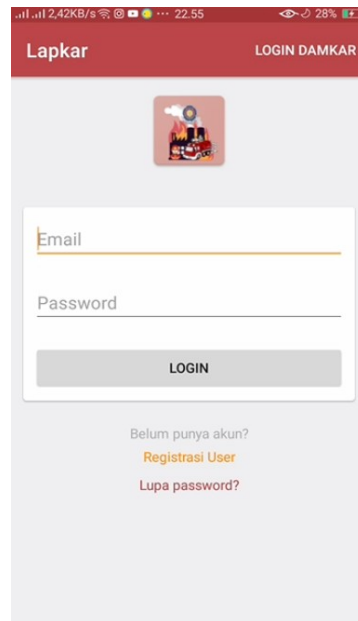
Splash screen adalah tampilan pertama program yang muncul sementara sebelum masuk ke halaman *login*. Tampilan *splash screen* pada aplikasi yang dibangun dapat dilihat pada gambar 4.1.



Gambar 4.1 Tampilan Antarmuka Halaman *Splash Screen*

4.1.2.2 Tampilan Antarmuka Halaman *Login Pengguna/Masyarakat*

Setelah tampilan halaman *splash screen*, selanjutnya pengguna akan diarahkan ke halaman *login*. Sebagaimana diperlihatkan pada gambar 4.2, halaman *login* merupakan halaman yang digunakan pengguna untuk masuk dalam halaman utama aplikasi. Apabila pengguna sudah melakukan proses registrasi, maka pengguna tersebut dapat masuk ke aplikasi dengan cara memasukkan *email* pada kolom *email*. Kemudian pengguna juga perlu memasukkan *password* pada kolom *password*. Tapi jika pengguna memasukkan *email/password* dengan salah maka akan muncul *toast login* salah sebagai peringatan dan pengguna harus memasukkan *email* dan *password* dengan benar agar dapat masuk ke dalam aplikasi. Tampilan halaman *login* pengguna pada aplikasi yang dibangun dapat dilihat pada gambar 4.2.

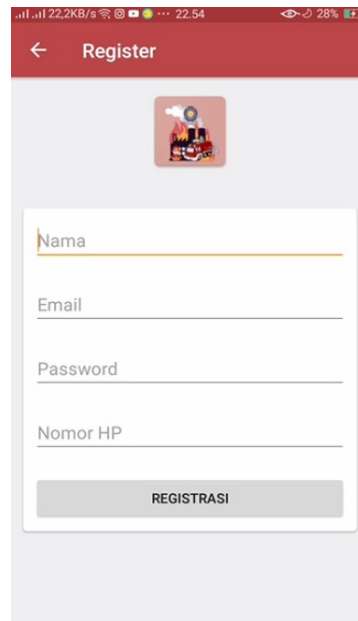


Gambar 4.2 Tampilan Antarmuka Halaman *Login* Pengguna

Pada tampilan antarmuka *login*, terdapat dua fungsi atau fitur yaitu registrasi *user* dan lupa *password*. Kedua fungsi atau fitur tersebut akan dijelaskan pada bagian selanjutnya.

4.1.2.3 Tampilan Halaman *Register* Pengguna/Masyarakat

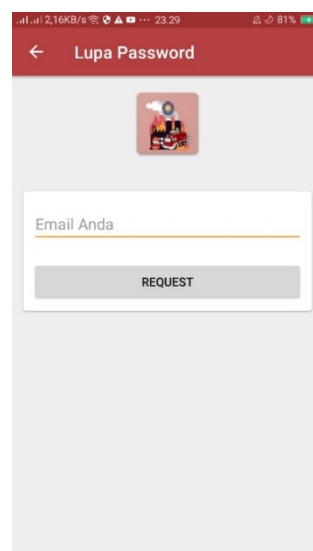
Register merupakan menu yang digunakan untuk mendaftar menjadi anggota ataupun *member* dari sebuah aplikasi. Dalam proses pendaftaran ini, pengguna diharuskan untuk mengisi sebuah *formulir* mengenai identitas pengguna secara benar. Adapun identitas tersebut seperti halnya nama, *email*, *password*, dan nomor *handphone*. Apabila data identitas yang pengguna masukkan terbukti benar, maka proses *register* yang pengguna lakukan akan cepat selesai. Setelah pengguna melakukan *register*, pengguna bisa *login* dengan *email* dan *password* yang pengguna daftarkan sebelumnya. Tampilan halaman *register* pengguna pada aplikasi yang dibangun dapat dilihat pada gambar 4.3.



Gambar 4.3 Tampilan Antarmuka Halaman *Register* Pengguna

4.1.2.4 Tampilan Halaman Lupa *Password* Pengguna / Masyarakat

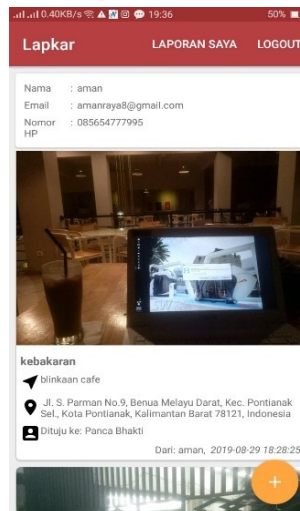
Halaman lupa *password* merupakan halaman yang berisi *form* untuk melakukan *reset password* bagi pengguna yang lupa *password*. Pengguna dapat mengisi *email* yang dimiliki atau telah terdaftar ke dalam aplikasi sebelumnya. Kemudian pengguna akan menerima notifikasi berupa *email* yang di dalamnya terdapat kode verifikasi untuk melakukan *reset password* pada alamat *email* yang diisikan sebelumnya. Selain itu pengguna juga akan menerima *link website* untuk melakukan *reset password*. Selanjutnya pengguna dapat memasukkan kode verifikasi tersebut pada *form* yang ada pada website untuk melakukan *reset password* serta *login* kembali ke dalam aplikasi. Tampilan halaman lupa *password* dapat dilihat pada gambar 4.4.



Gambar 4.4 Tampilan Antarmuka Halaman Lupa Password Pengguna

4.1.2.5 Tampilan Halaman *Beranda* Pengguna/masyarakat

Halaman *beranda* adalah halaman utama dari aplikasi pelaporan kebakaran apabila pengguna berhasil *login*. Pengguna juga dapat melihat profil pengguna itu di halaman *beranda*. Halaman utama ini terdiri dari laporan saya, *logout*, dan tombol (+) untuk membuat sebuah laporan tentang kebakaran. Pada halaman *beranda*, pengguna juga bisa melihat laporan dari pengguna lain yang belum ditangani oleh Damkar yang terpilih. Tampilan halaman beranda dapat dilihat pada gambar 4.5.



Gambar 4.5 Tampilan Antarmuka Halaman *Beranda* pada Pengguna

4.1.2.6 Tampilan Halaman Pilih Damkar pada Pengguna/masyarakat

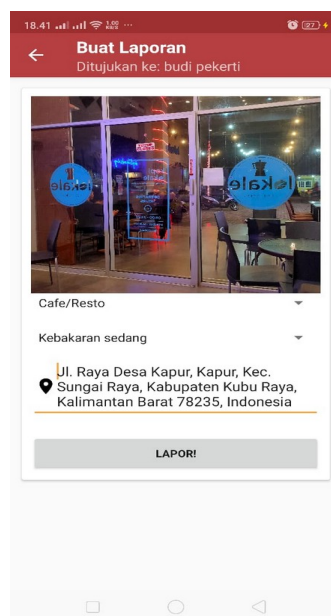
Pada halaman pilih Damkar ini, pengguna dapat memilih Damkar yang tersedia dan pengguna juga bisa memilih Damkar yang terdekat dari posisi pengguna berada. Tampilan halaman pilih Damkar dapat dilihat pada gambar 4.6.



Gambar 4.6 Tampilan Antarmuka Halaman Pilih Damkar pada Pengguna

4.1.2.7 Tampilan Halaman Buat Laporan pada Pengguna/masyarakat

Pada tampilan halaman buat laporan ini, pengguna dapat membuat laporan tentang kebakaran dengan menggunakan fitur kamera untuk mengambil sebuah objek kebakaran. *Field-field* yang disediakan di dalam halaman buat laporan harus diisi semua kecuali nama jalan. *Field* pada nama jalan sudah di *setting* otomatis sesuai pada posisi *smarthpone* pengguna berada. Tampilan halaman buat laporan dapat dilihat pada gambar 4.7.



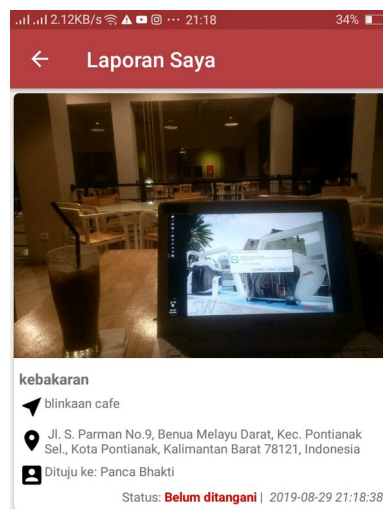
Gambar 4.7 Tampilan Antarmuka Halaman Buat Laporan pada Pengguna

4.1.2.8 Tampilan Halaman Laporan Saya pada Pengguna/masyarakat

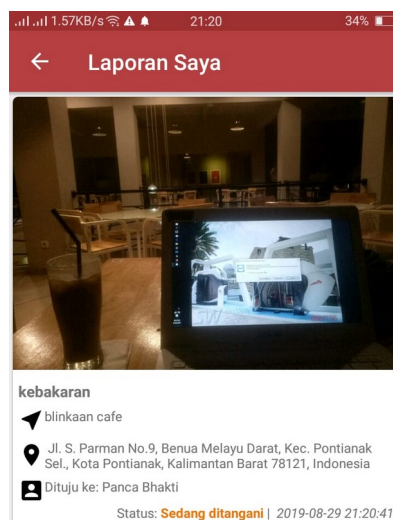
Pada halaman laporan saya, pengguna dapat melihat sebuah laporan yang sedang ditangani oleh petugas Damkar. Pengguna juga dapat mengetahui status sebuah laporan yang dilapor secara *realtime* oleh Damkar. Pada halaman ini status terbagi 3 keterangan yaitu :

- Belum ditangani
- Sedang ditangani
- Sudah Ditangani

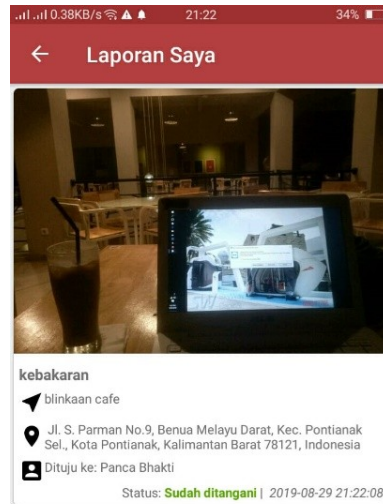
Tampilan halaman buat laporan saya yang statusnya belum ditangani, sedang ditangani dan sudah ditangani dapat dilihat pada gambar 4.8, 4.9 dan 4.10



Gambar 4.8 Tampilan Antarmuka Halaman Laporan Saya Status Belum Ditangani pada Pengguna



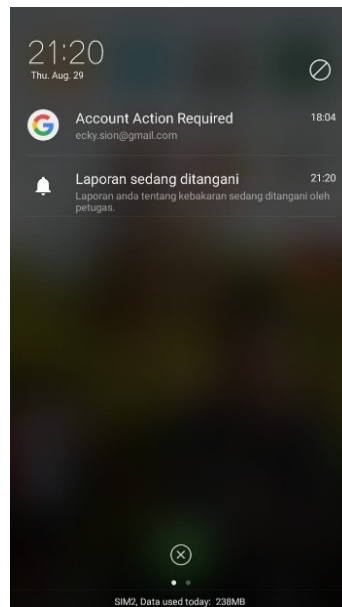
Gambar 4.9 Tampilan Antarmuka Halaman Laporan Saya Status Sedang Ditangani pada Pengguna



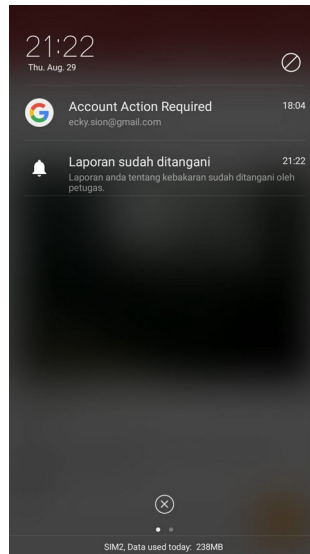
Gambar 4.10 Tampilan Antarmuka Halaman Laporan Saya Status Sudah Ditangani pada Pengguna

4.1.2.9 Tampilan *Notifikasi* pada Pengguna/masyarakat

Pada halaman ini jika laporan yang dilaporkan oleh pengguna sudah diterima oleh Damkar, maka pengguna mendapatkan notifikasi laporan sedang di proses oleh petugas, dan apabila Damkar sudah selesai menanggulangi kebakaran maka pengguna mendapatkan notifikasi laporan sudah ditangani oleh petugas. Tampilan halaman notifikasi dapat dilihat pada gambar 4.11, dan 4.12.



Gambar 4.11 Tampilan Antarmuka Halaman *Notifikasi* Status Laporan Sedang Ditangani pada Pengguna



Gambar 4.12 Tampilan Antarmuka Halaman *Notifikasi* Status Laporan Sudah Ditangani pada Pengguna

4.1.3 Tampilan Antarmuka Aplikasi Damkar

4.1.3.1 Tampilan Antarmuka Halaman *Splash Screen* Aplikasi *Android* pada Damkar

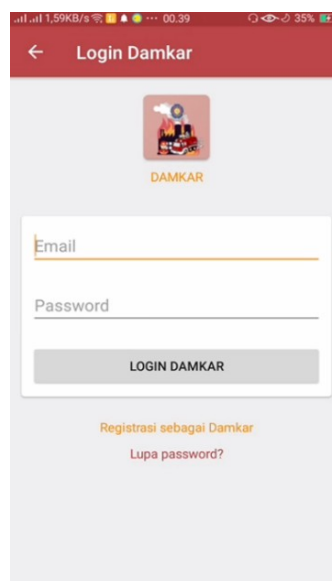
Splash screen adalah tampilan pertama program yang muncul sementara sebelum masuk ke menu utama. Tampilan halaman *splash screen* dapat dilihat pada gambar 4.13.



Gambar 4.13 Tampilan Antarmuka Halaman *Splash Screen*

4.1.3.2 Tampilan Antarmuka Halaman *Login* pada Damkar

Setelah tampilan halaman *splash screen*, selanjutnya Damkar masuk ke halaman *login* Damkar. *Login* adalah menu yang digunakan untuk masuk dalam suatu aplikasi atau web dan lain sebagainya. Apabila Damkar sudah melakukan proses registrasi, Damkar bisa masuk ke aplikasi dengan cara memasukkan *email* di kolom *email*. Selain itu, Damkar juga perlu menuliskan *password* di kolom *password*. Tapi jika Damkar memasukkan *email/password* dengan salah maka akan ada *toast login* salah dan Damkar harus memasukkan *email* dan *password* dengan benar agar dapat melakukan *login*. Tampilan halaman *login* Damkar dapat dilihat pada gambar 4.14.



Gambar 4.14 Tampilan Antarmuka Halaman *Login* Damkar

4.1.3.3 Tampilan Antarmuka Halaman *Register* pada Damkar

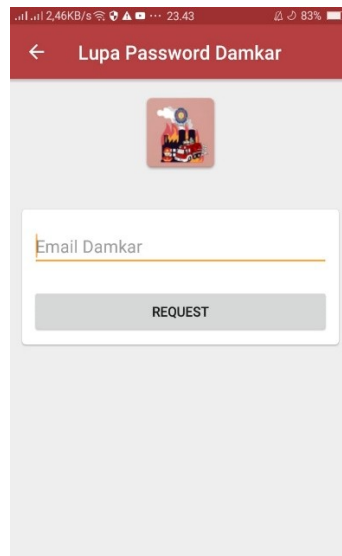
Register merupakan menu yang digunakan untuk mendaftar menjadi anggota ataupun *member* dari sebuah aplikasi. Dalam proses pendaftaran ini, Damkar diharuskan untuk mengisi sebuah formulir mengenai identitas Damkar secara benar. Adapun identitas tersebut seperti halnya nama, *email*, *password*, tentukan lokasi, dan alamat. Apabila data identitas yang pengguna masukkan terbukti benar, maka proses *register* yang Damkar lakukan akan cepat selesai. Setelah Damkar melakukan *register*, maka Damkar bisa *login* dengan *email* dan *password* yang Damkar daftarkan sebelumnya. Tampilan halaman *register* Damkar dapat dilihat pada gambar 4.15.

Gambar 4.15 Tampilan Antarmuka Halaman *Register Damkar*

4.1.3.4 Tampilan Halaman Lupa *Password Damkar*

Halaman lupa *password* merupakan halaman yang berisi *form* untuk melakukan *reset password* bagi Damkar yang lupa *password*. Damkar dapat mengisi *email* yang dimiliki atau telah terdaftar ke dalam aplikasi sebelumnya. Kemudian Damkar akan menerima notifikasi berupa *email* yang di dalamnya terdapat kode verifikasi untuk melakukan *reset password* pada alamat *email* yang diisikan sebelumnya. Selain itu Damkar juga akan menerima *link website* untuk melakukan *reset password*.

Selanjutnya Damkar dapat memasukkan kode verifikasi tersebut pada *form* yang ada pada *website* untuk melakukan *reset password* serta *login* kembali ke dalam aplikasi. Tampilan halaman lupa *password* dapat dilihat pada gambar 4.16



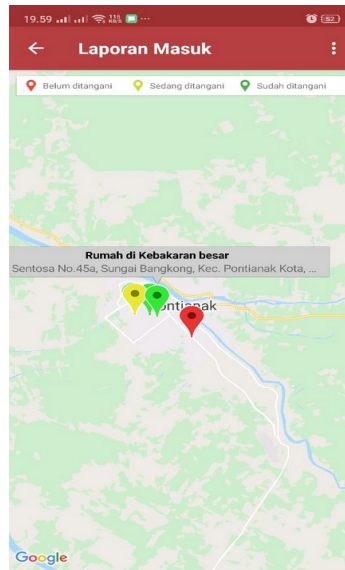
Gambar 4.16 Tampilan Antarmuka Halaman Lupa *Password* Damkar

4.1.3.5 Tampilan Halaman Beranda Damkar

Halaman beranda adalah halaman utama dari aplikasi pelaporan kebakaran apabila berhasil *login*. Halaman utama ini terdiri dari status, lihat laporan, lihat rekap, and lihat *spam*. Damkar juga dapat mengetahui profil Damkar itu sendiri. Pada halaman *beranda*, Damkar dapat melihat laporan yang masuk pada menu lihat laporan. Tampilan halaman *beranda* Damkar dapat dilihat pada gambar 4.17 dan tampilan halaman laporan masuk dapat dilihat pada gambar 4.18.



Gambar 4.17 Tampilan Antarmuka Halaman Beranda Damkar



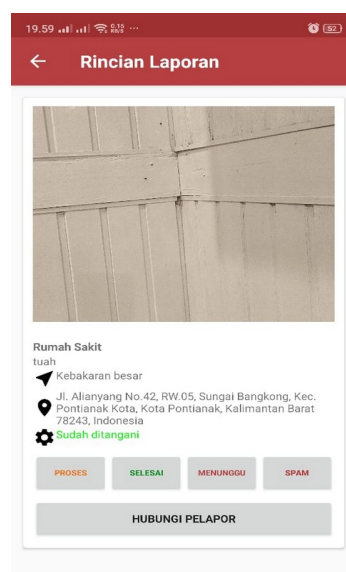
Gambar 4.18 Tampilan Antarmuka Halaman Lihat Laporan pada Damkar

4.1.3.6 Tampilan Halaman Rekapitulasi Laporan dan Rincian Laporan Damkar

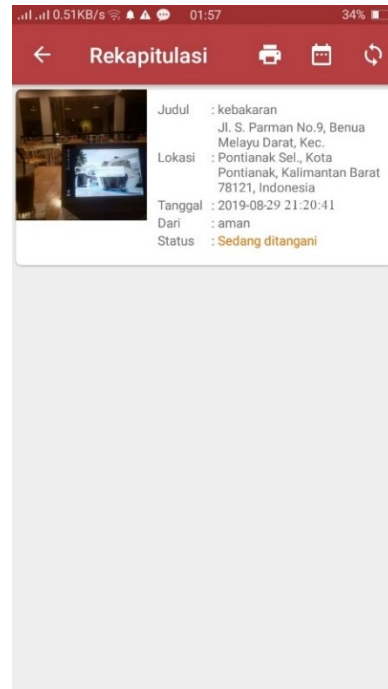
Pada halaman rekapitulasi laporan, Damkar dapat melihat proses laporan-laporan yang masuk pada laporan rekap. Laporan yang masuk dan belum ditangani oleh Damkar, maka laporan tersebut harus diselesaikan sampai tuntas agar pengguna dapat mengetahui tanggapan laporan dari Damkar. Damkar juga dapat mengubah status laporan yang masuk untuk memberikan tanggapan bahwa laporan yang masuk sedang diproses. Selanjutnya setelah laporan selesai ditangani, Damkar dapat memberitahukan kepada pelapor bahwa laporan telah selesai ditangani. Pada halaman ini Damkar dapat menghubungi pelapor dan apabila laporan nya itu tidak benar atau *hoax* maka laporan dari pelapor dibuang ke dalam *spam*. Tampilan halaman rekapitulasi laporan masuk Damkar dapat dilihat pada gambar 4.19, 4.20, 4.21, 4.22, dan tampilan halaman spam dapat dilihat pada gambar 4.23.



Gambar 4.19 Tampilan Antarmuka Halaman Rekapitulasi Laporan Belum Ditangani pada Damkar



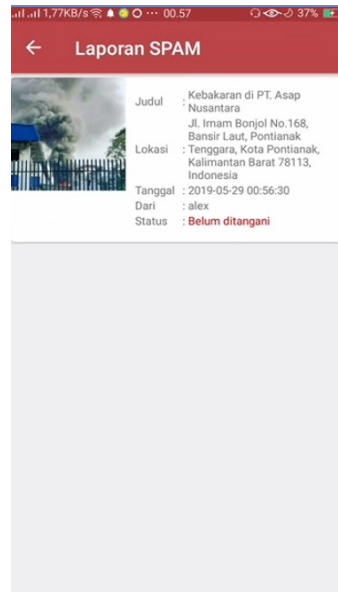
Gambar 4.20 Tampilan Antarmuka Halaman Rincian Laporan pada Damkar



Gambar 4.21 Tampilan Antarmuka Halaman Rekapitulasi Laporan Status Sedang Ditangani pada Damkar



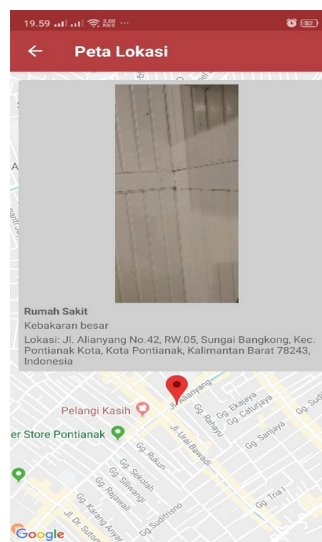
Gambar 4.22 Tampilan Antarmuka Halaman Rekapitulasi Laporan Status Sudah Ditangani pada Damkar



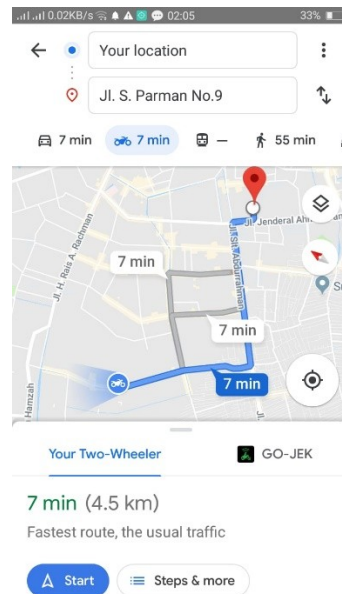
Gambar 4.23 Tampilan Antarmuka Halaman Laporan Spam pada Damkar

4.1.3.7 Tampilan Halaman Peta Lokasi Laporan dan *Direction* (Petunjuk Arah) di *Google Maps* pada Damkar

Pada tampilan halaman peta lokasi laporan, Damkar dapat melihat dan mengetahui sebuah laporan titik lokasi pada peta yang dilaporkan oleh pengguna. Damkar juga bisa mengeksekusi laporan serta meminta petunjuk arah titik lokasi kebakaran dengan memanfaatkan layanan *Google Maps*. Tampilan halaman peta lokasi pada Damkar dapat dilihat pada gambar 4.24, dan tampilan halaman *direction* pada Damkar dapat dilihat pada gambar 4.25.



Gambar 4.24 Tampilan Antarmuka Halaman Peta Lokasi pada Damkar



Gambar 4.25 Tampilan Antarmuka Halaman *Direction* di *Google Maps* pada Damkar

4.2 Pengujian Kuesioner Terhadap Aplikasi

Kuesioner berisi 15 pertanyaan yang dikelompokkan menjadi 3 aspek yang digunakan dalam pengujian aplikasi tersebut, yaitu aspek rekayasa perangkat lunak, aspek fungsionalitas dan aspek komunikasi visual.

4.2.1 Aspek Rekayasa Perangkat Lunak

Hasil kuesioner aspek rekayasa perangkat lunak dirangkum dalam sebuah tabel seperti pada tabel 4.1.

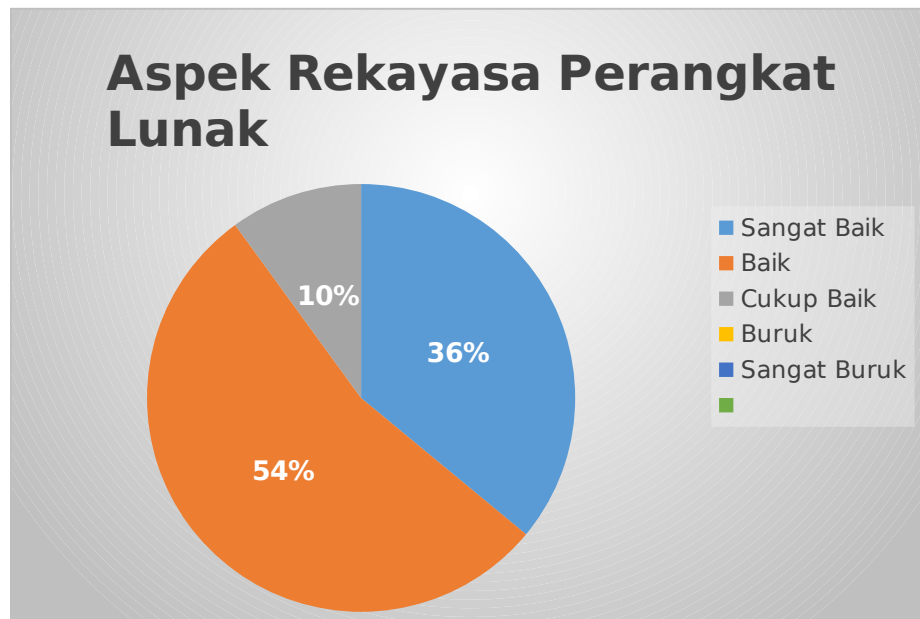
Tabel 4.1 Tabel Hasil Kuesioner Aspek Rekayasa Perangkat Lunak

No	Aspek Rekayasa Perangkat Lunak	Tanggapan					Total
		1	2	3	4	5	
1	Berjalannya aplikasi pada <i>smartphone</i>	0	0	1	6	3	10
2	Kemudahan menjalankan aplikasi pada <i>smartphone</i>	0	0	1	7	2	10
3	Kelancaran menjalankan aplikasi pada <i>smartphone</i>	0	0	1	4	5	10
4	Kemudahan mengakses menu-menu halaman pada aplikasi	0	0	1	4	5	10
5	Kenyamanan dalam menggunakan aplikasi secara keseluruhan	0	0	1	6	3	10
Jumlah		0	0	5	27	18	50

Persentase (%)	0%	0%	10%	54%	36%	100%

Keterangan : 1 = Sangat Buruk 3 = Cukup baik 5 = Sangat Baik
 2 = Buruk 4 = Baik

Berdasarkan hasil kuesioner pada aspek rekayasa perangkat lunak tersebut dapat diketahui bahwa sebagian besar responden menanggapi dengan aspek sangat baik dengan persentase 36%, baik dengan persentase 54%, dan cukup baik sebanyak 10%. *Chart* aspek rekayasa perangkat lunak dapat lihat pada Gambar 4.26.



Gambar 4.26 *Chart* Pengujian Pada Aspek Rekayasa Perangkat Lunak

4.2.2 Aspek Fungsionalitas

Hasil kuesioner aspek fungsionalitas dirangkum dalam sebuah tabel seperti pada tabel 4.2.

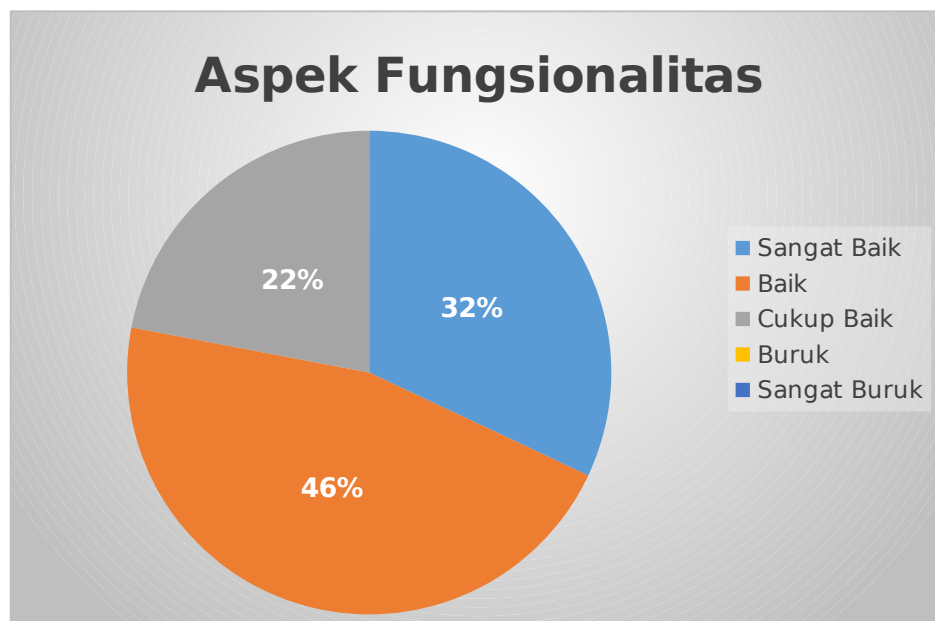
Tabel 4.2 Tabel Hasil Kuesioner Aspek Fungsionalitas

No	Aspek Rekayasa Perangkat Lunak	Tanggapan					Total
		1	2	3	4	5	
1	Kinerja aplikasi saat memuat data	0	0	3	5	2	10

2	Kinerja aplikasi saat menampilkan data	0	0	4	3	3	10
3	Kinerja aplikasi pada sistem notifikasi	0	0	2	6	2	10
4	Kinerja aplikasi saat membuat laporan kebakaran	0	0	2	4	4	10
5	Kinerja aplikasi saat menggunakan layanan <i>Google Maps</i> untuk menunjukkan rute perjalanan ke titik koordinat pelapor	0	0	0	5	5	10
Jumlah		0	0	11	23	16	50
Persentase (%)		0%	0%	22%	46%	32%	100%

Keterangan : 1 = Sangat Buruk 3 = Cukup baik 5 = Sangat Baik
 2 = Buruk 4 = Baik

Berdasarkan hasil kuesioner pada aspek fungsionalitas tersebut, diketahui bahwa sebagian besar responden menanggapi dengan nilai sangat baik pada aspek ini dengan persentase 32%, baik dengan persentase 46%, dan cukup baik sebanyak 22%. *Chart* aspek rekayasa perangkat lunak dapat lihat pada Gambar 4.27.



Gambar 4.27 *Chart* Aplikasi Pada Aspek Fungsionalitas

4.2.3 Aspek Komunikasi Visual

Hasil kuesioner aspek komunikasi visual dirangkum dalam sebuah tabel seperti pada tabel 4.3.

Tabel 4.3 Tabel Hasil Kuesioner Aspek Komunikasi Visual

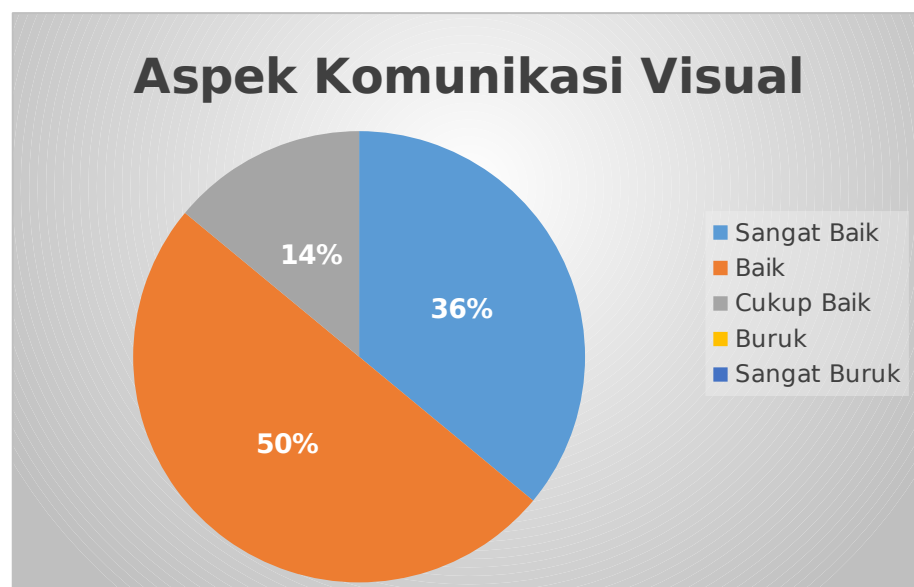
No	Aspek Rekayasa Perangkat Lunak	Tanggapan					Total
		1	2	3	4	5	
1	Tampilan antarmuka aplikasi	0	0	1	7	2	10
2	Tampilan menu aplikasi	0	0	3	6	1	10
3	Kemudahan membaca jenis dan ukuran huruf pada aplikasi	0	0	1	5	4	10
4	Kombinasi warna pada tampilan aplikasi	0	0	2	1	7	10
5	Respon (<i>feedback</i>) aplikasi ketika mengklik tombol atau memilih konten pada aplikasi	0	0	0	6	4	10
Jumlah		0	0	7	25	18	50
Persentase (%)		0%	0%	14%	50%	36%	100%

Keterangan : 1 = Sangat Buruk 3 = Cukup baik 5 = Sangat Baik

2 = Buruk

4 = Baik

Berdasarkan hasil kuesioner pada aspek komunikasi visual tersebut dapat diketahui bahwa sebagian besar responden menanggapi dengan aspek sangat baik dengan persentase 36%, baik dengan persentase 50%, dan cukup baik sebanyak 14%. *Chart* aspek rekayasa perangkat lunak dapat lihat pada Gambar 4.28.



Gambar 4.28 Chart Aplikasi Pada Aspek Komunikasi Visual

4.2.4 Likert's Summated Rating (LSR)

Untuk melihat skor terbesar dan terkecil dari satu orang responden dan total semua responden dari kuesioner terlihat pada tabel 4.4 berikut :

Tabel 4.4 Total Skor Responden dari Kuesioner

Responden	Item															Total
	1	2	3	4	5	6	7	8	9	10	11	12	1	14	1	

													3		5	
A	5	5	4	4	4	5	3	4	5	4	5	4	4	5	5	66
B	4	4	5	5	5	4	3	3	4	4	4	5	4	5	4	63
C	4	4	4	5	5	3	4	3	4	5	5	4	5	5	5	65
D	4	4	3	5	5	4	3	4	5	5	4	3	5	4	5	63
E	3	4	4	5	4	4	5	5	4	4	4	3	4	5	4	63
F	4	4	5	3	4	4	3	4	5	4	4	3	4	5	4	60
G	4	4	5	4	4	3	4	4	5	5	4	4	3	3	4	60
H	4	3	5	5	4	5	5	4	3	5	4	4	5	5	4	65
I	5	4	4	4	3	3	5	4	4	5	4	4	5	3	5	62
J	5	5	5	4	4	5	3	4	3	4	4	4	4	5	4	63
Total																630

Data yang diperoleh dari hasil pengujian dengan kuesioner kemudian diukur dengan metode *Likert's Summated Rating* (LSR).

1. Jumlah skor untuk setiap responden:

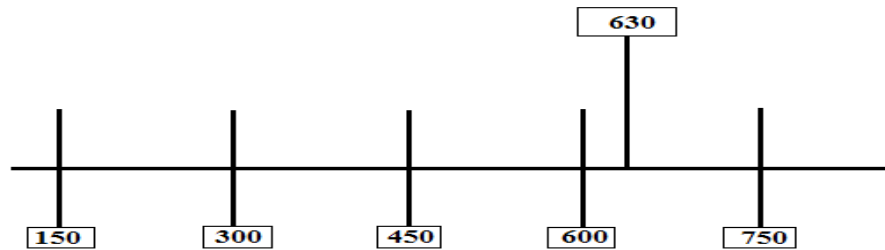
- skor maksimal = 75 (5 x 15 item)
- skor minimal = 15 (1 x 15 item)
- skor median = 45 (3 x 15 item)
- skor kuartil I = 30 (2 x 15 item)
- skor kuartil III = 60 (4 x 15 item)

2. Jumlah skor untuk seluruh responden:

- Maksimal = 750 (10 x 75)
- Minimal = 150 (10 x 15)
- Median = 450 (10 x 45)
- Kuartil I = 300 (10 x 30)
- Kuartil III = 600 (10 x 60)

3. Interpretasi jumlah skor tersebut adalah:

- $600 < \text{Skor} < 750$, artinya sangat positif (program dinilai berhasil)
- $450 < \text{Skor} < 600$, artinya positif (program dinilai cukup berhasil)
- $300 < \text{Skor} < 450$, artinya negatif (program dinilai kurang berhasil)
- $150 < \text{Skor} < 300$, artinya sangat negatif (program dinilai tidak berhasil)



Gambar 4.29 Hasil Kuesioner Pada Interpretasi LSR.

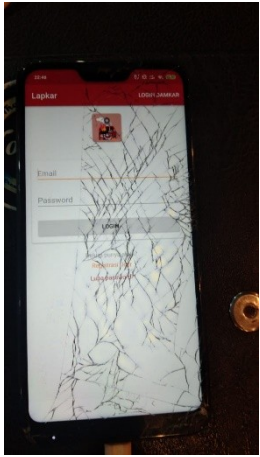
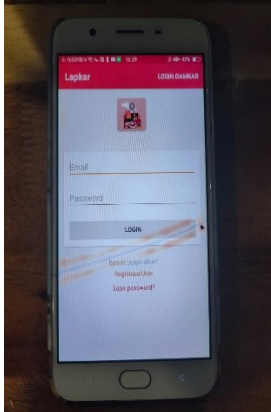
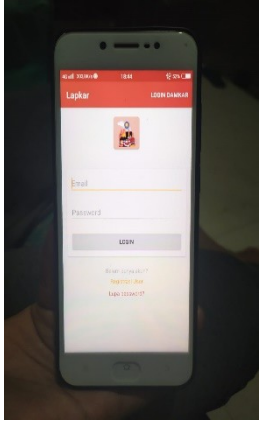
Gambar 4.31 menunjukkan bahwa hasil perhitungan kuisisioner pada interpretasi *Likert's Sumated Rating* (LSR) adalah 630 yang di antara skor 600 dan 750.

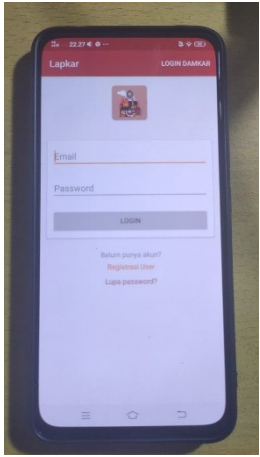
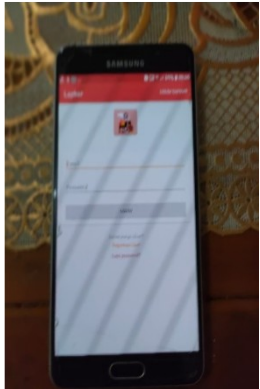
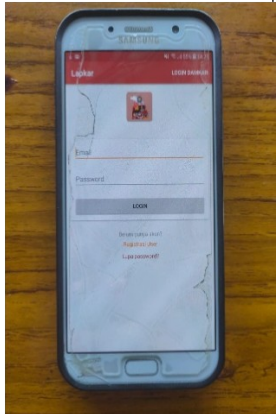
4.3 Pengujian Kompatibilitas Aplikasi

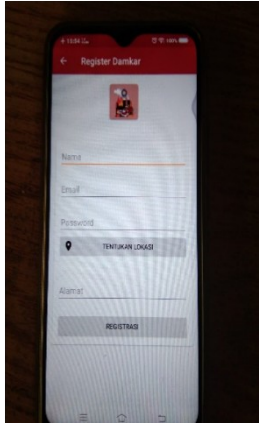
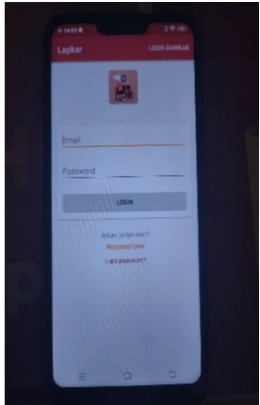
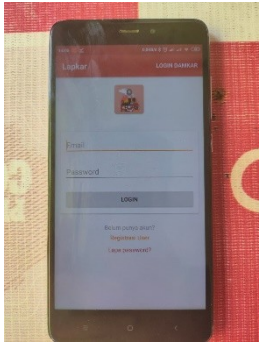
Pengujian ini dilakukan bertujuan untuk melihat kompatibilitas perangkat saat mengeksekusi aplikasi. Pengujian ini dilakukan dengan meng-*install* aplikasi pada 10 perangkat *android* yang berbeda. Hasil pengujian kompatibilitas aplikasi dapat dilihat pada tabel 4.5

Tabel 4.5 Tabel Pengujian Kompatibilitas Aplikasi

No	Merk Perangkat Android	Tipe	Versi Sistem Operasi	Keterangan Hasil	Screenshot Perangkat Android
1	Realme	Realme 5 pro	9	1) <i>Instalasi</i> : berhasil 2) <i>Layout</i> menu : sesuai 3) <i>Pop up</i> notifikasi : berhasil 4) Pencarian Damkar tersedia dan terdekat : sesuai 5) Penandaan lokasi : sesuai 6) Membuat laporan foto <i>geotag</i> : berhasil 7) Menerima laporan masuk : berhasil 8) Penentuan rute terdekat : sesuai	

2	Xiaomi	Redmi 6 pro	8.1	<p>1) <i>Instalasi</i> : berhasil</p> <p>2) <i>Layout</i> menu : sesuai</p> <p>3) <i>Pop up</i> notifikasi : berhasil</p> <p>4) Pencarian Damkar tersedia dan terdekat : sesuai</p> <p>5) Penandaan lokasi : sesuai</p> <p>6) Membuat laporan foto <i>geotag</i> : berhasil</p> <p>7) Menerima laporan masuk : berhasil</p> <p>8) Penentuan rute terdekat : sesuai</p>	
3	Oppo	F1S	5.1	<p>1) <i>Instalasi</i> : berhasil</p> <p>2) <i>Layout</i> menu : sesuai</p> <p>3) <i>Pop up</i> notifikasi : berhasil</p> <p>4) Pencarian Damkar tersedia dan terdekat : sesuai</p> <p>5) Penandaan lokasi : sesuai</p> <p>6) Membuat laporan foto <i>geotag</i> : berhasil</p> <p>7) Menerima laporan masuk : berhasil</p> <p>8) Penentuan rute terdekat : sesuai</p>	
4	Vivo	V5	6.0	<p>1) <i>Instalasi</i> : berhasil</p> <p>2) <i>Layout</i> menu : sesuai</p> <p>3) <i>Pop up</i> notifikasi : berhasil</p> <p>4) Pencarian Damkar tersedia dan terdekat : sesuai</p> <p>5) Penandaan lokasi : sesuai</p> <p>6) Membuat laporan foto <i>geotag</i> : berhasil</p> <p>7) Menerima laporan masuk : berhasil</p> <p>8) Penentuan rute terdekat : sesuai</p>	

5	Vivo	V15	9.0	<p>1) <i>Instalasi</i> : berhasil</p> <p>2) <i>Layout</i> menu : sesuai</p> <p>3) <i>Pop up</i> notifikasi : berhasil</p> <p>4) Pencarian Damkar tersedia dan terdekat : sesuai</p> <p>5) Penandaan lokasi : sesuai</p> <p>6) Membuat laporan foto <i>geotag</i> : berhasil</p> <p>7) Menerima laporan masuk : berhasil</p> <p>8) Penentuan rute terdekat : sesuai</p>	
6	Samsung	A5	7.0	<p>1) <i>Instalasi</i> : berhasil</p> <p>2) <i>Layout</i> menu : sesuai</p> <p>3) <i>Pop up</i> notifikasi : berhasil</p> <p>4) Pencarian Damkar tersedia dan terdekat : sesuai</p> <p>5) Penandaan lokasi : sesuai</p> <p>6) Membuat laporan foto <i>geotag</i> : berhasil</p> <p>7) Menerima laporan masuk : berhasil</p> <p>8) Penentuan rute terdekat : sesuai</p>	
7	Samsung	A7	9	<p>1) <i>Instalasi</i> : berhasil</p> <p>2) <i>Layout</i> menu : sesuai</p> <p>3) <i>Pop up</i> notifikasi : berhasil</p> <p>4) Pencarian Damkar tersedia dan terdekat : sesuai</p> <p>5) Penandaan lokasi : sesuai</p> <p>6) Membuat laporan foto <i>geotag</i> : berhasil</p> <p>7) Menerima laporan masuk : berhasil</p> <p>8) Penentuan rute terdekat : sesuai</p>	

8	Vivo	Y15	9	1) <i>Instalasi</i> : berhasil 2) <i>Layout</i> menu : sesuai 3) <i>Pop up</i> notifikasi : berhasil 4) Pencarian Damkar tersedia dan terdekat : sesuai 5) Penandaan lokasi : sesuai 6) Membuat laporan foto <i>geotag</i> : berhasil 7) Menerima laporan masuk : berhasil 8) Penentuan rute terdekat : sesuai	
9	Vivo	Y83	9	1) <i>Instalasi</i> : berhasil 2) <i>Layout</i> menu : sesuai 3) <i>Pop up</i> notifikasi : berhasil 4) Pencarian Damkar tersedia dan terdekat : sesuai 5) Penandaan lokasi : sesuai 6) Membuat laporan foto <i>geotag</i> : berhasil 7) Menerima laporan masuk : berhasil 8) Penentuan rute terdekat : sesuai	
10	Xiaomi	Redmi 4	6.0.1	1) <i>Instalasi</i> : berhasil 2) <i>Layout</i> menu : sesuai 3) <i>Pop up</i> notifikasi : berhasil 4) Pencarian Damkar tersedia dan terdekat : sesuai 5) Penandaan lokasi : sesuai 6) Membuat laporan foto <i>geotag</i> : berhasil 7) Menerima laporan masuk : berhasil 8) Penentuan rute terdekat : sesuai	

4.4 Analisis Hasil Pengujian

Dari hasil pengujian aplikasi terhadap aspek rekayasa perangkat lunak, aspek fungsionalitas, dan aspek komunikasi visual dilakukan analisis sebagai berikut.

Berdasarkan hasil kuesioner pada aspek rekayasa perangkat lunak tersebut dapat diketahui bahwa sebagian besar responden menanggapi dengan aspek sangat baik dengan persentase 36%, baik dengan persentase 54%, dan cukup baik sebanyak 10%.

Berdasarkan hasil kuesioner pada aspek fungsionalitas tersebut, diketahui bahwa sebagian besar responden menanggapi dengan nilai sangat baik pada aspek ini dengan persentase 32%, baik dengan persentase 46%, dan cukup baik sebanyak 22%.

Berdasarkan hasil kuesioner pada aspek komunikasi visual tersebut dapat diketahui bahwa sebagian besar responden menanggapi dengan aspek sangat baik dengan persentase 36%, baik dengan persentase 50%, dan cukup baik sebanyak 14%.

Berdasarkan hasil pengujian kompatibilitas dapat disimpulkan bahwa semua perangkat dapat menggunakan aplikasi.

ASUSASUSHC;
by adguard=NC<wbk

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah dilakukan pengujian dan analisis aplikasi pelaporan kebakaran menggunakan metode *geotag* dan penentuan rute menggunakan layanan *Google Maps* berbasis *android*, dapat disimpulkan bahwa:

1. Masyarakat dapat membuat laporan kebakaran menggunakan fitur kamera dan alamat sudah otomatis terisi pada saat pengambilan gambar.
2. Damkar dapat mengetahui lokasi kebakaran dengan teknologi fitur foto *geotag* dan dapat memanfaatkan *Google Maps* untuk penentuan rute ke lokasi kebakaran.
3. Hasil pengujian kuesioner yang diukur dengan metode *Likert's Summated Rating* (LSR) menunjukkan bahwa responden menilai sistem sangat baik dengan skor 630 dari 750.

5.2 Saran

Adapun beberapa hal yang dapat ditambahkan dalam pengembangan sistem ini adalah :

1. Penelitian selanjutnya dapat menambahkan fitur *chatting* pada aplikasi.
2. Penelitian selanjutnya dapat menambahkan fitur *rewards point* bagi pengguna yang sudah membuat laporan secara *real* dilapangan.
3. Penelitian selanjutnya dapat menambahkan fitur nomor whatsapp Damkar ketika sudah melapor.
4. Penelitian selanjutnya dapat menambahkan fitur *tracking* pada aplikasi.
5. Penelitian selanjutnya dapat menambahkan fitur nomor *WhatsApp*.

#ASUS#####A#S#U#S#####e
r###
#b#y# #a#d#g#u#a#r#d###c#r# #C#o#u#r#i#e#r# #N#e#w#a#r#!#d#o#c#####