

**PERANGKAT LUNAK TANGGAP BENCANA KEBAKARAN BERBASIS
ANDROID PT. PELABUHAN INDONESIA II (PERSERO) CABANG
PANJANG DENGAN METODE *WATERFALL***

(Skripsi)

Oleh:

**STEFANY WULANDARI
1517051050**



**S1 ILMU KOMPUTER
JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
2022**

ABSTRAK

Perangkat Lunak Tanggap Bencana Kebakaran Berbasis Android PT. Pelabuhan Indonesia II (Persero) Cabang Panjang dengan Metode *Waterfall*

Oleh

STEFANY WULANDARI

Beberapa perusahaan baik swasta maupun badan milik negara pada dasarnya sudah memiliki standar operasional prosedur mitigasi kebencanaan. Pada PT. Pelabuhan Indonesia II (Persero) Cabang Panjang diterapkan standar mitigasi kebencanaan melalui *Port Facility Security Plan* (PFSP), ditambah dengan peraturan terkait dengan Keselamatan dan Kesehatan Kerja (K3). Kekurangan stok Alat Pemadam Api Ringan (APAR) dan sulitnya akses terhadap bantuan dan informasi tanggap bencana menjadikan suatu gedung khususnya badan pemerintahan menjadi rentan akan terjadinya kebakaran. Padahal potensi terjadi kebakaran disana cukup tinggi mengingat fokus PT. Pelabuhan Indonesia II (Persero) Cabang Panjang tidak hanya di pelabuhan namun juga di gedung administratif yang memiliki potensi hubungan arus pendek listrik, kelalaian pegawai, dan sebagainya. Berdasarkan permasalahan tersebut, pengembangan perangkat lunak tanggap bencana kebakaran dapat menjadi solusi karena pengguna dapat melakukan akses informasi laporan kebakaran dan lokasi alat-alat pemadam kebakaran ketika berada di area gedung maupun area pelabuhan. Selain itu, perangkat lunak tanggap bencana kebakaran menyediakan informasi jalur evakuasi yang ada. Perangkat lunak tanggap bencana kebakaran juga dapat mempermudah pegawai dalam melakukan *monitoring* dan memanjamen alat pemadam kebakaran yang tersedia. Pengujian perangkat lunak dilakukan oleh para pegawai PT. Pelabuhan Indonesia II (Persero) Cabang Panjang. Secara keseluruhan, perangkat lunak yang dikembangkan dapat mencapai tujuannya.

Kata Kunci : Kebakaran, Laporan Kebakaran, Tanggap Bencana, Manajemen Alat Pemadam Kebakaran

ABSTRACT

Android Based Fire Disaster Response Software PT. Pelabuhan Indonesia II (Persero) Panjang Branch with Waterfall Method

By

STEFANY WULANDARI

Some companies, both private and state-owned corporation, basically already have standard operating procedures for disaster mitigation. On PT. Pelabuhan Indonesia II (Persero) Panjang Branch applies disaster mitigation standards through the Port Facility Security Plan (PFSP), coupled with the regulations related to Occupational Health and Safety (OHS). The shortage of stocks of fire extinguishers gear and the difficulty to access disaster response assistance and information make the building especially the government building is vulnerable to fires. Even though the potential for a wildfire there is quite high considering the focus of PT. Pelabuhan Indonesia II (Persero) Panjang Branch is not only the port but also in administrative buildings that have the potential for electrical short circuits, employee negligence, and so on. Based on these problems, the development of fire disaster response software can be a solution because users can access fire report information and the location of fire extinguishers gears in the building and port area. In addition, fire disaster response software provides information on existing evacuation routes. Fire disaster response software can also make it easier for employees to monitor and manage available fire extinguishers. Software testing is carried out by the employees of PT. Pelabuhan Indonesia II (Persero) Panjang Branch. All in all, the developed software was able to achieve its goals.

Keywords : Fire, Fire Reports, Disaster Response, Fire Extinguisher Management

**PERANGKAT LUNAK TANGGAP BENCANA KEBAKARAN BERBASIS
ANDROID PT. PELABUHAN INDONESIA II (PERSERO) CABANG
PANJANG DENGAN METODE *WATERFALL***

Oleh

STEFANY WULANDARI

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA KOMPUTER**

Pada

**Jurusan Ilmu Komputer
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDARLAMPUNG
2022**

Judul Skripsi : **PERANGKAT LUNAK TANGGAP BENCANA
KEBAKARAN BERBASIS ANDROID
PT. PELABUHAN INDONESIA II (PERSERO)
CABANG PANJANG DENGAN METODE
WATERFALL**

Nama Mahasiswa : **Stefany Wulandari**

Nomor Pokok Mahasiswa : 1517051214

Jurusan : Ilmu Komputer

Fakultas : Matematika dan Ilmu Pengetahuan Alam



Aristoteles, S.Si., M.Si.
NIP 19810521 200604 1 002

Ardiansyah, S.Kom., M.Kom.
NIP 19870128 201803 1 001

2. Ketua Jurusan Ilmu Komputer

Didik Kurniawan, S.Si., M.T.
NIP 19800419 200501 1 004

MENGESAHKAN

1. Tim Penguji

Ketua : Aristoteles, S.Si., M.Si.



**Penguji I
Sekretaris**

: Ardiansyah, S.Kom., M.Kom.



**Penguji II
Bukan Pembimbing**

: Dr. Eng. Admi Syarif



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Surtipto Dwi Yuwono, M.T.
NIP 197407052000031001

Tanggal Lulus Ujian Skripsi: 12 Januari 2022

PERNYATAAN

Saya yang bertanda tangan di bawah ini, menyatakan bahwa skripsi saya yang berjudul “Perangkat Lunak Tanggap Bencana Kebakaran Berbasis Android PT. Pelabuhan Indonesia II (Persero) Cabang Panjang dengan Metode *Waterfall*” merupakan karya saya sendiri dan bukan karya orang lain. Semua tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila dikemudian hari terbukti skripsi saya merupakan hasil penjiplakan atau dibuat orang lain, maka bersedia menerima sanksi berupa pencabutan gelar yang saya terima.

Bandar Lampung, 12 Januari 2022



Handwritten signature of Stefany Wulandari.

Stefany Wulandari
NPM. 1517051050

RIWAYAT HIDUP



Penulis dilahirkan di Bandung pada tanggal 6 April 1997, Sebagai anak pertama dari tiga bersaudara dari ayah bernama Muhammad Rudtiadji Budiono dan Sitti Nurhayati S.E. Penulis memiliki dua orang adik yang bernama Stefina Noor Widjoyo dan Stephen Suryo Widjoyo.

Penulis menyelesaikan beberapa pendidikannya di Taman Kanak-Kanak (TK) pada tahun 2013 di TK Xaverius 2 Panjang. Sekolah Dasar di SD Pardomuan Bandung yang selesai pada tahun 2009. Pendidikan Sekolah Menengah Pertama di SMP Negeri 12 Bandar Lampung yang selesai pada tahun 2012, dan pendidikan Sekolah Menengah Atas di SMA Yayasan Pembina Universitas Lampung yang selesai pada tahun 2015.

Penulis terdaftar sebagai mahasiswi jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur SNMPTN pada tahun 2015. Selama menjadi mahasiswa beberapa kegiatan yang dilakukan penulis antara lain

1. Pada bulan Januari 2016 penulis mengikuti Karya Wisata Ilmiah di Desa Batutegi, Kecamatan Air Nanningan, Kabupaten Tanggamus.
2. Pada Bulan Juli 2018 Penulis melaksanakan Kuliah Kerja Nyata selama 40 hari di Desa Peniangan, Kecamatan Marga Sekampung, Kabupaten Lampung Timur.
3. Pada bulan Januari 2019 penulis melaksanakan Kerja Praktik di PT. Pelabuhan Indonesia II (Persero) Cabang Panjang

PERSEMBAHAN

Puji dan syukur saya panjatkan kepada Allah SWT, atas segala rahmat dan karunia yang telah diberikan-Nya sehingga skripsi saya yang berjudul “Perangkat Lunak Tanggap Bencana Kebakaran Berbasis Android PT. Pelabuhan Indonesia II (Persero) Cabang Panjang dengan Metode *Waterfall*” dapat terselesaikan.

Namun, saya tahu keberhasilan bukanlah akhir dari perjuangan
Tapi awal dari sebuah harapan dan cita-cita yang besar.

Kupersembahkan karya ini kepada:

Kedua orang tua yang tidak pernah berhenti dalam memberikan doa, nasihat, semangat, motivasi, dan kasih sayangnya, serta kepada kedua adik saya yang juga tidak pernah berhenti mendukung saya untuk menyelesaikan skripsi ini. Terima kasih selama ini selalu menjaga, melindungi, memberikan kasih sayang, perhatian, dan pengorbanan yang tak pernah bisa terbalaskan.

Sahabat dan teman-teman seperjuangan yang tidak pernah meninggalkan dan selalu ada untuk saya. Selalu menemani, memberikan bantuan, dan memotivasi. Terima kasih atas semua hal yang telah kita lalui bersama baik dalam cerita suka maupun duka. Terima kasih untuk canda dan tawa yang telah kalian berikan.

Keluarga Besar Ilmu Komputer 2015,
Serta Almamater Tercinta, Universitas Lampung.

MOTO

*“Books! And cleverness! There are more important things — friendship and
bravery.”*

– J.K. Rowling

“There is always a way out for those clever enough to find it.”

– Rick Riordan

“May it be a light to you in dark places, when all other lights go out.”

– J.R.R. Tolkien

“If you don't know where you are going, any road can take you there.”

– Lewis Carroll

“Life is only worth living if you have something to strive for. To aim at.”

– P.B. Kerr

SANWACANA

Puji syukur kehadirat Allah SWT atas berkat, rahmat, dan karunia yang diberikan sehingga penulis dapat menyelesaikan penelitian skripsi yang berjudul “Perangkat Lunak Tanggap Bencana Kebakaran Berbasis Android PT. Pelabuhan Indonesia II (Persero) Cabang Panjang dengan Metode Waterfall”. Penelitian dilakukan sebagai syarat untuk memperoleh gelar Sarjana Ilmu Komputer Universitas Lampung.

Terima kasih penulis ucapkan kepada semua pihak yang telah berperan besar dan membantu dalam penyusunan skripsi ini, antara lain.

1. Kedua orang tua yang selalu memberikan kasih sayang, cinta, doa, dukungan, bantuan, dan perhatian, serta memfasilitasi segala kebutuhan yang diperlukan dalam menyelesaikan skripsi ini.
2. Adik-adikku Stefina Noor Widjoyo dan Stephen Suryo Widjoyo yang selalu menemani serta memberikan doa, dukungan, dan bantuan.
3. Bapak Aristoteles, S.Si., M.Si. sebagai pembimbing utama yang telah memberi bimbingan, ilmu, kritik, saran, bantuan, dan nasihat sehingga skripsi ini dapat diselesaikan.
4. Bapak Ardiansyah, S.Kom., M.Kom. sebagai pembimbing kedua yang telah memberi bimbingan, ilmu, kritik, saran, bantuan, dan nasihat sehingga skripsi ini dapat diselesaikan.
5. Bapak Dr. Eng. Admi Syarif sebagai pembahas yang telah memberi kritik dan saran yang bermanfaat dalam perbaikan skripsi ini.
6. Bapak Didik Kurniawan, S.Si., M.T. selaku Ketua Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung
7. Bapak dan Ibu Dosen Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang telah senantiasa memberikan

ilmu dan pengalaman dalam hidup penulis selama menjadi mahasiswa.

8. Ibu Ade Nora Maela, Kak Zainuddin, dan kak Ardi Novalia yang membantu dalam segala urusan administrasi penulis di Jurusan Ilmu Komputer.
9. Joshua Evan Anthony, yang selalu ada di sisi penulis serta memberikan doa, motivasi, dan semangat dalam penyusunan skripsi ini.
10. Garinda Linggar Nasansia, sebagai sahabat terbaik yang selalu ada dan tidak pernah meninggalkan saya dalam suka maupun duka, serta selalu memberikan dukungan dan canda tawa.
11. Friaji, Ria, Yunita dan Anisa, sebagai sahabat yang telah mewarnai hari-hari penulis, serta memberikan dukungan dan doa.
12. Sahabat “Abnormal” Putri, Elin, Dito, Ronaldy, Wawai, Fikri, Shendy, Raxy, Ibrahim, Afrido sebagai sahabat yang selalu ada dan telah memberikan banyak kenangan serta pengalaman selama masa perkuliahan.
13. Keluarga besar Ilmu Komputer 2015 yang telah berjuang bersama-sama dalam menjalankan studi di Jurusan Ilmu Komputer.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, akan tetapi semoga skripsi ini dapat bermanfaat bagi perkembangan ilmu pengetahuan serta dapat menjadi bantuan bagi rekan-rekan Ilmu Komputer Universitas Lampung.

Bandar Lampung, 12 Januari 2022

Stefany Wulandari

NPM. 1517051050

DAFTAR ISI

	Halaman
DAFTAR ISI	v
DAFTAR GAMBAR	vii
DAFTAR TABEL	x
I. PENDAHULUAN	11
A. Latar Belakang	11
B. Rumusan Masalah	4
C. Batasan Masalah.....	5
D. Tujuan.....	5
E. Manfaat.....	5
II. LANDASAN TEORI	6
A. Kebakaran.....	6
B. Android.....	7
C. Android Studio	8
D. <i>Unified Modelling Language</i> (UML).....	10
E. Metode <i>Waterfall</i>	16
F. Testing	17
III. METODE PENELITIAN	18
A. Tempat dan Waktu Penelitian	18
B. Alat Pendukung	18
C. Tahapan Penelitian	19
V. KESIMPULAN DAN SARAN	68
A. Kesimpulan.....	68
B. Saran.....	68

DAFTAR PUSTAKA.....69

DAFTAR GAMBAR

Gambar	Halaman
1. Logo Android (Satyaputra, 2014)	7
2. Berbagai Versi Ikon Android (Firly, 2019)	8
3. Contoh ilustrasi <i>Usecase Diagram</i> (Safaat, 2018).....	11
4. Contoh Ilustrasi <i>Activity Diagram</i> (Safaat, 2018).....	13
5. Contoh Ilustrasi <i>Class Diagram</i> (Alsaad dan Hussein, 2020)	14
6. Metode <i>Waterfall</i> (Ramadhan dan Wahab, 2019)	16
7. Diagram Alir Tahapan Penelitian	20
8. <i>Usecase Diagram</i> Perangkat Lunak Tanggap Bencana Kebakaran.....	23
9. <i>Activity Diagram Login Admin</i>	24
10. <i>Activity Diagram Lihat Daftar Gedung</i>	25
11. <i>Activity Diagram Tambah Gedung</i>	26
12. <i>Activity Diagram Edit Gedung</i>	27
13. <i>Activity Diagram Hapus Gedung</i>	27
14. <i>Activity Diagram Lihat Alat Pemadam Api Ringan</i>	28
15. <i>Activity Diagram Tambah Alat Pemadam Api Ringan</i>	29
16. <i>Activity Diagram Edit Alat Pemadam Api Ringan</i>	30
17. <i>Activity Diagram Hapus Alat Pemadam Api Ringan</i>	31
18. <i>Activity Diagram Lihat Hydrant</i>	31
19. <i>Activity Diagram Lihat Mobil Pemadam Kebakaran</i>	32
20. <i>Activity Diagram Lihat Jalur Evakuasi</i>	33
21. <i>Activity Diagram Lihat Pertolongan Pertama pada Kecelakaan</i>	33
22. <i>Activity Diagram Bantuan</i>	34
23. <i>Activity Diagram Pelaporan Kebakaran</i>	34

24. <i>Activity Diagram</i> Lihat Notifikasi.....	35
25. <i>Activity Diagram</i> Lihat Notifikasi.....	36
26. <i>Activity Diagram</i> Telepon Nomor Darurat	36
27. <i>Activity Diagram</i> Ubah <i>Password</i>	37
28. <i>Activity Diagram</i> Unduh Data	38
29. <i>Entity Relationship Diagram</i> Tanggap Bencana Kebakaran PT. Pelabuhan Indonesia Cabang Panjang	39
30. Rancangan <i>Splash Screen</i>	40
31. Rancangan Beranda.....	41
32. Rancangan <i>Login Admin</i>	41
33. Rancangan Halaman Admin	42
34. Rancangan <i>Form</i> Ubah <i>Password</i>	43
35. Rancangan Halaman Notifikasi Admin	44
36. Rancangan Halaman Sebarkan Lokasi (Admin).....	44
37. Rancangan Halaman Notifikasi Pengguna.....	45
38. Rancangan Halaman Lokasi Notifikasi Pengguna.....	45
39. Rancangan Menu Navigasi Pemadam.....	46
40. Rancangan Halaman Daftar Gedung.....	47
41. Rancangan Halaman Alat Pemadam Api Ringan	47
42. Rancangan Halaman <i>Pop-up</i> Alat Pemadam Api Ringan	48
43. Rancangan <i>Form</i> Tambah Gedung	49
44. Rancangan <i>Form Edit</i> Gedung.....	50
45. Rancangan <i>Form</i> Hapus Gedung	50
46. Rancangan Konfirmasi Hapus Gedung.....	51
47. Rancangan <i>Form</i> Tambah Alat Pemadam Api Ringan.....	52
48. Rancangan <i>Form Edit</i> Alat Pemadam Api Ringan	53
49. Rancangan <i>Form</i> Hapus Alat Pemadam Api Ringan.....	54
50. Rancangan Konfirmasi Hapus Alat Pemadam Api Ringan	54
51. Rancangan Halaman <i>Hydrant</i>	55
52. Rancangan Halaman Mobil Pemadam.....	56
53. Rancangan Halaman Gedung Evakuasi	56
54. Rancangan Halaman Jalur Evakuasi	57

55. Rancangan Halaman Gedung Alat P3K.....	58
56. Rancangan Halaman Daftar Alat P3K	58
57. Rancangan Halaman <i>Pop-up</i> Alat P3K.....	59
58. Rancangan Halaman Menu Navigasi Bantuan.....	59
59. Rancangan Halaman Menu Navigasi Tentang.....	60
60. Rancangan Halaman Laporkan Kebakaran.....	61

DAFTAR TABEL

Tabel	Halaman
1. Simbol-Simbol dalam <i>Usecase Diagram</i> (Sugiarti, 2018).	12
2. Simbol-Simbol dalam <i>Activity Diagram</i> (Rusmawan, 2019)	13
3. Simbol-Simbol dalam <i>Class Diagram</i> (Sugiarti, 2018).....	15
4. Tabel Pengujian Oleh Admin.....	63
5. Tabel Pengujian Oleh Pengguna	66

I. PENDAHULUAN

A. Latar Belakang

Kebakaran merupakan salah satu bencana yang dapat terjadi baik karena unsur alam maupun akibat kelalaian manusia. Terdapat total kejadian kebakaran sebanyak 352 kebakaran terjadi di lingkungan masyarakat di seluruh Indonesia, 42 di antaranya merupakan kebakaran hutan (BNPB, 2018). Kebakaran lebih banyak terjadi akibat unsur kelalaian manusia dengan berbagai pemicunya, seperti misalnya membuang puntung rokok sembarangan, merokok atau memantik api di tempat yang rawan akan kebakaran, arus pendek listrik, ledakan gas, hingga pembukaan lahan untuk perkebunan dan pemukiman warga (Faisal dan HT, 2019). Kebakaran yang terjadi di pemukiman cenderung lebih berisiko jatuh korban karena berada dekat dengan aktivitas masyarakat sehari-hari. Belum lagi, bencana kebakaran seringkali berakhir dengan kerugian besar yang harus dialami masyarakat, mulai dari kehilangan seluruh materi hingga nyawa. Salah satu contoh kebakaran fatal yang pernah terjadi di Indonesia yaitu kebakaran pabrik kembang api di Tangerang yang menewaskan sebanyak 49 orang (Supriatin, 2017). Hal tersebut menjadikan bencana kebakaran menjadi salah satu bencana yang paling berbahaya terutama jika kebakaran tersebut terjadi di lingkungan masyarakat.

Kejadian kebakaran di lingkungan masyarakat terutama di lingkungan kerja sering mengalami kesulitan dalam mendapatkan akses kebencanaan. Alasannya adalah

kurangnya akses informasi tanggap bencana, pemukiman padat penduduk, jauh dari sumber air, hingga tidak tersedianya alat-alat mitigasi kebencanaan (ILO 2018).

Pada beberapa kejadian, terjadi kesulitan evakuasi korban hingga menelan korban jiwa yang tidak sedikit. Kebakaran yang terjadi di gedung Kemenhub pada 2018 menewaskan tiga orang. Kebakaran ini diakibatkan oleh hubungan arus pendek listrik di gedung yang sedang direnovasi, jalur evakuasi menjadi sulit dicapai karena terlalu banyak asap (Detik 2018). Kebakaran pada gedung bertingkat seringkali mengalami kesulitan dalam mengevakuasi korban karena seluruh akses gedung tertutup, mulai dari jaringan listrik, *lift*, jaringan internet, hingga akses ke inventaris alat mitigasi kebakaran sehingga potensi jatuhnya korban menjadi lebih tinggi. Maka dari itu, gedung kantor memerlukan media tanggap kebencanaan yang dapat diakses dengan mudah apabila terjadi kebakaran.

Beberapa perusahaan baik swasta maupun badan milik negara pada dasarnya sudah memiliki standar operasional prosedur terkait mitigasi kebencanaan. Pada PT. Pelabuhan Indonesia II (Persero) Cabang Panjang yang telah menerapkan standar mitigasi kebencanaan melalui *Port Facility Security Plan* (PFSP) ditambah dengan peraturan terkait dengan Keselamatan dan Kesehatan Kerja (K3) yang diatur dalam Peraturan Menteri Perhubungan Republik Indonesia No. PM 134 tahun 2016 tentang Manajemen Keamanan Kapal dan Fasilitas Pelabuhan (KEMENHUB 2016).

Standar operasional prosedur keamanan terhadap bencana cenderung seringkali diabaikan. Padahal potensi kejadian bencana tersebut cukup tinggi mengingat fokus PT. Pelabuhan Indonesia II (Persero) Cabang Panjang bukan hanya di pelabuhan namun juga di gedung administratif yang memiliki potensi hubungan arus pendek listrik, kelalaian pegawai, dan sebagainya. Pada dasarnya setiap perusahaan harus memiliki standar mitigasi kebencanaan yang baik, dan hal ini diatur berdasarkan peraturan pemerintah melalui Peraturan Menteri Pekerjaan Umum pasal 3 dan pasal 6 tentang Pedoman Teknis Manajemen Proteksi Kebakaran di Perkotaan, terkait upaya mitigasi dan kewajiban tanggap kebencanaan (Kemen PU 2009). Namun, pada gedung perkantoran biasanya hanya ditemukan inventaris alat pemadam api ringan (APAR), padahal bencana kebakaran memiliki pola penanggulangan yang berbeda-beda tergantung jenis pemicu kebakarannya.

Kekurangan stok APAR dan sulitnya akses terhadap bantuan dan informasi tanggap bencana kebakaran menjadikan suatu gedung khususnya yang terdapat pada badan pemerintahan menjadi rentan akan terjadinya kebakaran. Selain itu, tidak adanya perangkat lunak yang dapat mempermudah proses persebaran informasi dan respon bantuan kebencanaan menjadikan perusahaan semakin berpotensi kesulitan ketika terjadinya bencana. Dalam beberapa penelitian, tersedia beberapa upaya mitigasi kebencanaan melalui sistem berbasis web, namun cenderung kurang tanggap.

Penelitian yang dilakukan Pradiga dkk adalah membuat identifikasi resiko dan perencanaan proteksi kebakaran berbasis android untuk mencoba mendeteksi asap dalam suatu gedung dengan kenaikan suhu yang mengidentifikasi kebakaran dalam gedung (Pradiga dkk, 2018). Penelitian lain yang diteliti oleh Aquino dkk, adalah membuat perangkat lunak *emergency* berdasarkan suara. Suara yang dikeluarkan dari *smartphone* memberikan arah evakuasi yang terdekat atau pintu keluar ketika terjadi bencana (Aquino et al., 2019). Sedangkan penelitian yang dilakukan Amores dkk adalah untuk memaksimalkan kemampuan *smartphone* dalam keadaan darurat termasuk pada saat terjadi bencana kebakaran. Manajemen ketika terjadi bencana dapat digunakan pada *smartphone* sehingga Amores merancang fitur-fitur untuk membuat perangkat lunak keadaan darurat pada *smartphone* (Amores et al., 2019). Penelitian ini dilakukan untuk membuat sebuah perangkat lunak tanggap bencana kebakaran pada PT. Pelabuhan Indonesia II (Persero) Cabang Panjang, agar dapat bermanfaat dapat memberikan informasi dalam penanganan kebakaran, manajemen alat pemadam api ringan (APAR), dan informasi mengenai jalur-jalur evakuasi.

B. Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan sebelumnya, maka rumusan masalah dalam penelitian ini adalah bagaimana sistem dapat membantu dalam melakukan upaya dalam membantu memberikan informasi ketika terjadi bencana kebakaran dan manajemen alat pemadam api ringan (APAR) pada PT. Pelabuhan Indonesia II (Persero) Cabang Panjang.

C. Batasan Masalah

Pada penelitian ini diadakan pembatasan masalah agar ruang lingkup yang dibahas dapat tersusun dan terarah dengan baik sesuai dengan judul yang diajukan pada penelitian ini. Maka dari itu, batasan masalah dalam menyusun penelitian ini adalah sebagai berikut:

1. Penelitian hanya berfokus pada bencana kebakaran pada kantor PT. Pelabuhan Indonesia II (Persero) Cabang Panjang.
2. Penelitian hanya membuat perangkat lunak yang dikhususkan untuk digunakan pada internal kantor PT. Pelabuhan Indonesia II (Persero) Cabang Panjang.

D. Tujuan

Adapun tujuan dilakukannya penelitian di PT. Pelabuhan Indonesia II (Persero) Cabang Panjang adalah untuk mengembangkan perangkat lunak tanggap bencana kebakaran pada PT. Pelabuhan Indonesia II (Persero) Cabang Panjang.

E. Manfaat

Manfaat yang dapat diperoleh dari penelitian yang dilakukan ini adalah sebagai berikut:

1. Dapat membantu karyawan PT. Pelabuhan Indonesia II (Persero) Cabang Panjang dalam mendapatkan bantuan dan informasi apabila terjadi bencana kebakaran.
2. Dapat mempermudah manajemen alat pemadam api ringan (APAR) di PT. Pelabuhan Indonesia II (Persero) Cabang Panjang.

II. LANDASAN TEORI

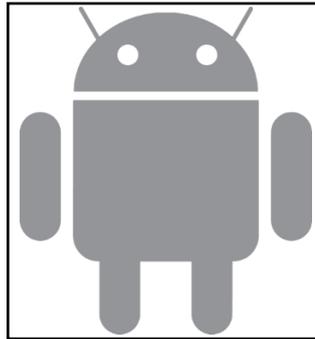
A. Kebakaran

Kebakaran adalah bencana yang menjadi ancaman bagi kehidupan manusia, harta benda maupun lingkungan hidup sekitar (Faisal dan HT, 2019). Kebakaran menjadi bencana yang tidak dapat terdeteksi dan berujung kerugian yang besar. Bencana kebakaran adalah suatu bencana yang dapat mengakibatkan jatuhnya korban dalam jumlah besar (Aquino et al., 2019). Tujuan utama manajemen darurat mengalami kemajuan. Awalnya manajemen ini bertujuan untuk pengurangan risiko, kemudian menjadi pengurangan kerusakan, karena bencana yang dipengaruhi oleh bencana alam tidak dapat dihindari. Ada dua metode utama untuk mengurangi kerusakan: mensimulasikan suatu skenario bencana untuk memperkirakan skala bencana, dan mengumpulkan informasi *real-time* tentang skala bencana untuk mengoreksi serta menyesuaikan hasil simulasi (Mori et al., 2017).

Cara yang dapat diterapkan saat terjadi bencana kebakaran adalah merencanakan jalan keluar darurat dan menyiapkan kelengkapan alat (Ramawangsa dkk, 2018). Upaya pencegahan kebakaran agar perusahaan mampu untuk menurunkan risiko kecelakaan yang akan terjadi akibat adanya kebakaran. Desain sistem perlindungan yang baik diperlukan agar bangunan memiliki kesiapan menghadapi kebakaran dan meminimalkan kerugian yang disebabkan oleh kebakaran, terutama jika terjadi di gedung fasilitas umum dan diharapkan agar dapat mengakomodasi kerumunan. Sistem proteksi kebakaran pasif adalah sistem yang dapat diterapkan agar dapat mengatasi kebakaran pada bangunan (Rahman and Stephanie, 2017).

B. Android

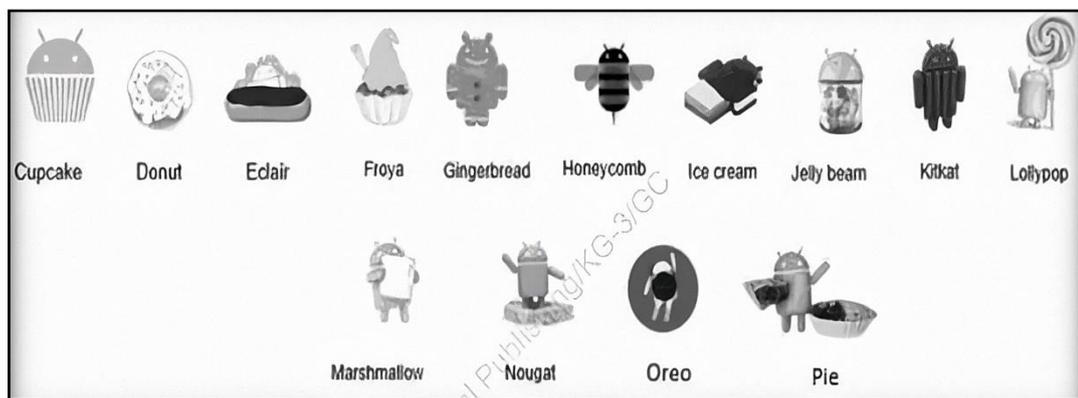
Android merupakan istilah dalam bahasa Inggris yang berarti “robot yang menyerupai manusia”. Logo dari Android seperti yang terdapat pada Gambar 1 dicerminkan seperti arti katanya (Satyaputra, 2014).



Gambar 1. Logo Android (Satyaputra, 2014)

Android adalah salah satu sistem operasi yang berkembang menjadi bahasa pemrograman. Pada dasarnya, Android merupakan sistem operasi berbasis Linux. Perusahaan Android yang awalnya berada di bawah bendera Android Inc adalah pengembang pertama dari sistem operasi ini. Android Inc didirikan pertama kali di Kota Alto, California Amerika Serikat, pada bulan Oktober tahun 2003. Pendirinya terdiri Andy Rubin, Rich Miner, dan Chris White. Kemudian pada tahun 2005, Android dibeli oleh Google dan selanjutnya digunakan oleh khalayak ramai. Pada tahun 2007 secara resmi Google meluncurkan Android sebagai sebuah sistem operasi baru, khususnya untuk digunakan pada *smartphone*. Pengembangan aplikasi berbasis Android yang didukung oleh Google Inc. membuat Android cepat tersebar ke seluruh dunia. Sifat kode sumbernya yang terbuka membuat banyak orang tertarik untuk ikut mengembangkan sistem operasi ini dan menjadikan Android sebagai sistem operasi yang menawarkan banyak variasi dan juga aplikasi Android dengan menyediakan panduan, dan kode sumber untuk *Application Programming Interface (API)* yang cukup lengkap (Fachrul dan Gianto, 2015).

Android menggunakan sistem operasi *open source* yang kodenya dirilis oleh Google di bawah Lisensi Apache. Kode *open source* dan lisensi perizinan yang ada pada Android memungkinkan perangkat lunak ini dimodifikasi dengan bebas dan dapat didistribusikan secara luas oleh para pembuat perangkat, operator nirkabel, dan pengembang aplikasi. Selain itu, Android juga memiliki sebuah komunitas pengembang aplikasi dalam memperluas fungsionalitas perangkat, pada umumnya ditulis dalam versi kustomisasi bahasa pemrograman Java (Jubilee Enterprise, 2015). Selain itu, sistem operasi Android juga dapat diilustrasikan sebagai jembatan antara perangkat dan penggunanya, sehingga pengguna dapat melakukan interaksi dengan perangkatnya dengan menjalankan berbagai perangkat lunak yang tersedia pada perangkat tersebut yang dapat dipergunakan dengan bebas, hal ini dikarenakan sistem operasi Android bersifat *open source*. Paradigma *open source* memudahkan pengembangan teknologi Android. Hal ini dapat terjadi karena semua pihak yang tertarik dapat memberikan kontribusi, baik pada pengembangan sistem operasi maupun pada aplikasi (Satyaputra, 2014). Gambar 2 merupakan ikon Android dari berbagai versinya.



Gambar 2. Berbagai Versi Ikon Android (Firly, 2019)

C. Android Studio

Integrated Development Environment (IDE) untuk pengembangan aplikasi Android yang didukung oleh Google adalah IntelliJ IDEA yang dirangkai dalam Android

Studio. Hal ini terjadi karena, IDE Eclipse sudah tidak didukung secara resmi oleh Google sebagai IDE untuk membuat aplikasi Android (Safaat, 2018). Android Studio adalah IDE resmi untuk membangun aplikasi Android berdasarkan IntelliJ IDEA. IntelliJ IDEA sendiri merupakan Java *Integrated Development Environment* (IDE) yang dikembangkan oleh JetBrains, untuk mengembangkan perangkat lunak komputer. IntelliJ IDE berfungsi dalam membantu proses *coding*, baik dari segi navigasi, penyokong produktivitas, hingga *code editor* (Satyaputra dan Aritonang, 2016).

Beberapa fitur yang terdapat pada Android Studio (Satyaputra dan Aritonang, 2016) adalah sebagai berikut:

- *Instant run*, ketika menekan tombol *run* atau *debug*, maka fitur *instant run* akan mendorong *code* dan perubahan *resource* aplikasi yang sedang berjalan. Hal ini dapat membantu dalam melakukan perubahan dan menyajikan perubahan tersebut tanpa harus melakukan *restart* pada aplikasi maupun membangun ulang APK, sehingga efek perubahan dapat terlihat secara langsung.
- *Emulator*, Android *emulator* melakukan instalasi dan menjalankan aplikasi lebih cepat dibandingkan perangkat asli. *Emulator* ini juga memungkinkan untuk membuat sebuah *prototype* dan melakukan pengujian aplikasi pada berbagai perangkat Android seperti ponsel, tablet, perangkat Android, dan Android TV. Dapat juga untuk melakukan simulasi pada berbagai fitur seperti GPS, *network latency*, sensor gerak, dan *input multi-touch*.
- *Smart code editor*, *smart code editor* membantu dalam menuliskan barisan kode yang lebih baik, cepat, dan produktif. Program dapat bekerja lebih cepat dan lebih produktif dengan adanya penawaran pelengkapan kode, pemfaktoran ulang, dan analisis kode. *Smart code editor* Android Studio menawarkan penyelesaian kode, *refactoring*, dan analisis kode yang canggih. Saat menuliskan kode, Android Studio akan memberikan saran kode dalam daftar *dropdown* yang baik.
- *Code templates and sample app*, Android Studio meliputi *template code* dan proyek yang memudahkan untuk menambah pola mapan seperti *navigation*

drawer dan *view pager* sehingga dapat memulai *coding* dari awal ataupun menggunakan kode contoh.

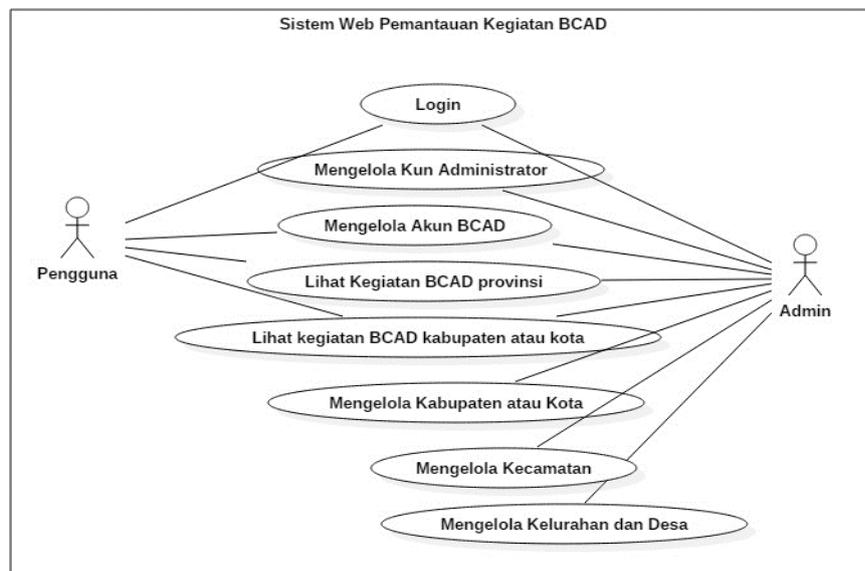
- *Testing tools* dan *framework*, Android Studio menyediakan *tools* yang membantu dalam menguji aplikasi menggunakan Junit 4 dan *functional UI test framework*. Bahkan, dapat menggunakan *Espresso Test recorder* yang dapat melakukan *generate code* tes uji dengan merekam interaksi antara pengguna dan aplikasi pada *device/emulator*.
- *Lintelligence*, Android Studio menyediakan kerangka kerja analisis statis yang kuat dan mencakup 280 *lints checks* yang berbeda pada keseluruhan aplikasi. Android Studio juga membantu dalam perbaikan untuk mengatasi masalah kinerja, keamanan, dan kebenaran.

D. Unified Modelling Language (UML)

Unified Modelling Language (UML) merupakan bahasa pemodelan visual yang biasa digunakan untuk memodelkan, menspesifikasikan, dan mendokumentasikan rancangan suatu sistem perangkat lunak. Pemodelan memberikan gambaran yang jelas tentang sistem yang akan dibangun baik dari sisi struktural maupun fungsional (Akil, 2016). Menurut sumber lain, *Unified Modelling Language* (UML) merupakan salah satu bentuk notasi atau bahasa yang digunakan oleh profesional di bidang *software* untuk menggambarkan atau memodelkan sebuah sistem *software*. Sebelumnya, ada bahasa lain dalam memodelkan sistem seperti *Data Flow Diagram* (DFD) dan *Booch Diagram*. Namun sejak populernya teknologi pemrograman, perancangan, dan analisis berbasis objek *Unified Modelling Language* (UML) menjadi *de facto standar language*. Selain itu, *Unified Modelling Language* (UML) juga berfungsi sebagai jembatan dalam mengkomunikasikan beberapa aspek sistem melalui elemen-elemen grafis yang dapat dikombinasikan menjadi diagram (Sugiarti, 2018). *Unified Modelling Language* (UML) juga dapat diartikan sebagai bahasa yang menggunakan grafik/gambar untuk memvisualisasi, menspesifikasikan, dan membangun suatu rancangan sistem untuk pengembangan *software* berbasis OO (*Object Oriented*) (Zufria 2013).

1. *Usecase Diagram*

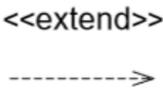
Dalam membuat sistem hal pertama yang harus dilakukan adalah menentukan kebutuhan. Terdapat 2 jenis kebutuhan, yaitu kebutuhan fungsional dan kebutuhan non-fungsional. Kebutuhan fungsional akan digambarkan melalui sebuah diagram yang dinamakan *Usecase Diagram*. *Usecase Diagram* merupakan permodelan untuk menggambarkan perilaku dari sistem yang akan dibuat. *Usecase Diagram* digunakan untuk mendeskripsikan interaksi antara satu atau lebih aktor dengan sistem dan digunakan untuk mengetahui fungsi apa saja yang ada dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut (Sugiarti, 2018). Selain itu, *Usecase Diagram* adalah suatu aktivitas yang menggambarkan urutan interaksi antar satu atau lebih aktor dan sistem (Safaat, 2018). Gambar 3 menjelaskan interaksi antara *Usecase* dengan aktor terhadap sistem Pemantauan Kegiatan Bakal Calon Anggota Dewan (BCAD).



Gambar 3. Contoh ilustrasi *Usecase Diagram* (Safaat, 2018)

Usecase Diagram memiliki 5 simbol yang perlu dipahami yaitu simbol *Usecase*, simbol aktor, simbol asosiasi, simbol *extend*, dan simbol *include* (Sugiarti, 2018). Penjelasan mengenai simbol-simbol yang ada pada *Usecase Diagram* dapat dilihat pada Tabel 1.

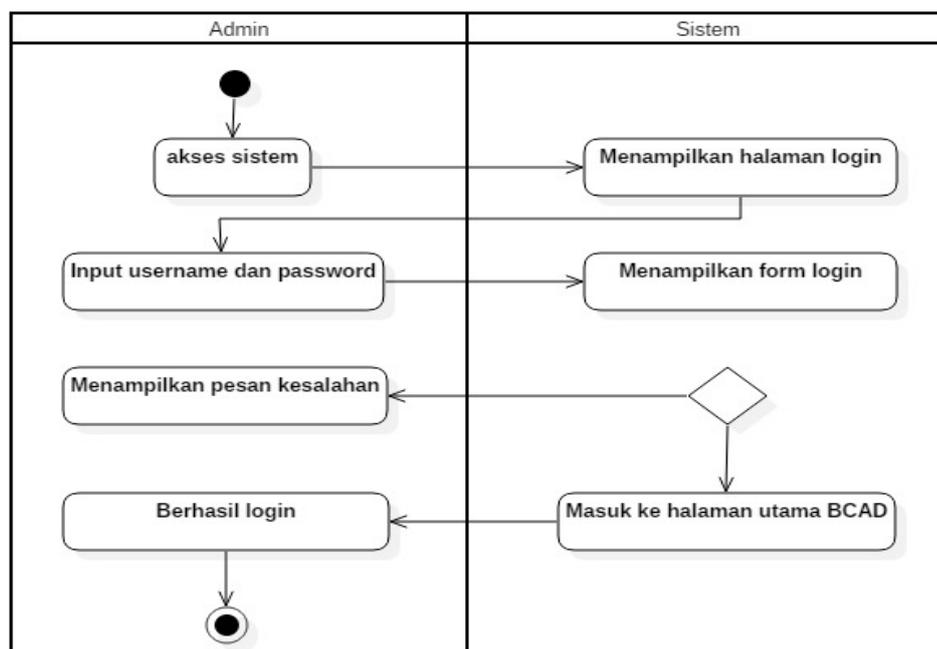
Tabel 1. Simbol-Simbol dalam *Usecase Diagram* (Sugiarti, 2018).

<i>Construct</i>	<i>Description</i>	<i>Symbol</i>
<i>Usecase</i>	Fungsi yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.	
Aktor	Berupa orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat.	
Asosiasi	Komunikasi antara aktor dan <i>Usecase</i> yang berpartisipasi pada <i>Usecase</i> atau <i>Usecase</i> memiliki interaksi dengan aktor.	
<i>Extend</i>	Relasi <i>Usecase</i> tambahan pada sebuah <i>Usecase</i> . <i>Usecase</i> yang ditambahkan bisa berdiri sendiri tanpa <i>Usecase</i> tambahan.	
<i>Include</i>	Relasi <i>Usecase</i> tambahan pada sebuah <i>Usecase</i> . Dan <i>Usecase</i> yang ditambahkan akan memerlukan <i>Usecase</i> ini untuk menjalankan fungsi atau sebagai syarat untuk menjalankannya.	

2. Activity Diagram

Activity Diagram merupakan *state diagram* khusus, dimana sebagian besar *state* adalah aksi dan transisinya dipicu oleh *state* yang telah selesai. Oleh karena itu, *Activity Diagram* tidak menggambarkan perilaku internal sebuah sistem dan interaksi antar sub-sistemnya secara detail, tetapi lebih menggambarkan proses dan jalur *Activity* dari tahap atas secara umum (Sugiarti, 2018). *Activity Diagram* menggambarkan alur kerja pada setiap *Usecase*. *Activity Diagram* pada analisa ini mencakup *Activity Diagram* setiap *Usecase* (Safaat, 2018). *Activity Diagram*

merupakan *Unified Modelling Language* (UML) yang paling banyak digunakan di dalam *software development*. Model ini memungkinkan ada operasi *flow* di setiap kelasnya, *Activity Diagram* juga mengizinkan hirarki dekomposisi, dengan status sub aktivitas sehingga dapat memodelkan beberapa kelas dengan *class aggregation* (Sulaiman et al., 2019). Gambar 4 merupakan contoh *Activity Diagram* mengenai *login* ke dalam sistem Pemantauan Kegiatan Bakal Calon Anggota Dewan (BCAD).



Gambar 4. Contoh Ilustrasi *Activity Diagram* (Safaat, 2018)

Tabel 2 menjelaskan tentang simbol-simbol notasi inti pada *Activity Diagram*. Menurut Rusmawan, 5 simbol utama pada *Activity Diagram* adalah simbol *start state*, simbol *end state*, simbol *Activity*, simbol *decision*, dan simbol *interaction*.

Tabel 2. Simbol-Simbol dalam *Activity Diagram* (Rusmawan, 2019)

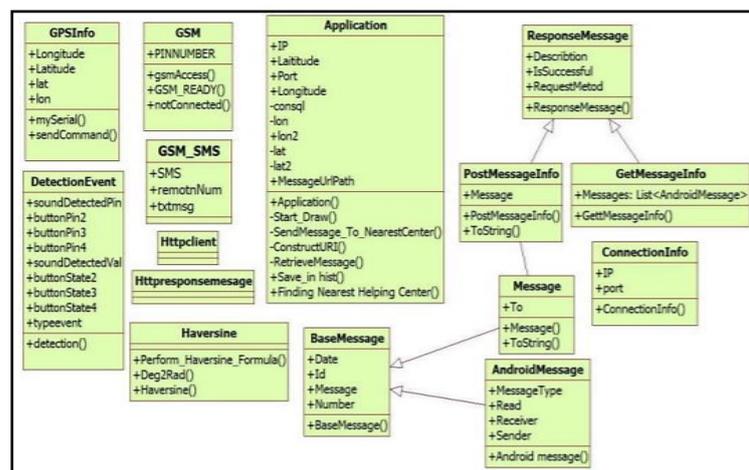
<i>Construct</i>	<i>Description</i>	<i>Symbol</i>
<i>Start state</i>	Titik awal atau permulaan.	●
<i>End state</i>	Titik akhir atau akhir dari aktivitas.	◎

Tabel 2. Simbol-Simbol dalam *Activity Diagram* (lanjutan)

<i>Construct</i>	<i>Description</i>	<i>Symbol</i>
<i>Activity</i>	<i>Activity</i> atau aktivitas yang dilakukan oleh <i>actor</i> .	
<i>Decision</i>	Pilihan untuk mengambil keputusan.	
<i>Interaction</i>	Alur.	

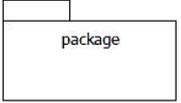
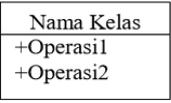
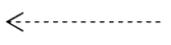
3. *Class Diagram*

Diagram kelas atau *Class Diagram* menggambarkan sistem dari segi pendefinisian berdasarkan *class-class* yang dibuat untuk membangun sistem. Dalam *Class Diagram* dideskripsikan jenis-jenis objek dalam sistem dan berbagai hubungan statis yang terdapat di antara mereka. Tiga area pokok yang terdapat pada *Class Diagram* adalah nama, atribut, dan operasi (Sugiarti, 2018). Contoh ilustrasi *Class Diagram* dapat dilihat pada Gambar 5.

**Gambar 5.** Contoh Ilustrasi *Class Diagram* (Alsaad dan Hussein, 2020)

Penjelasan mengenai simbol-simbol yang ada pada *Class Diagram* dapat dilihat pada Tabel 3.

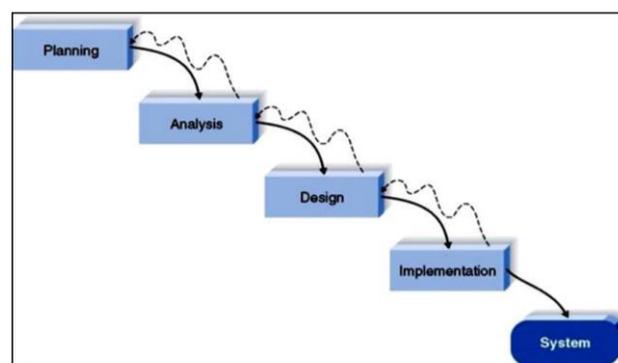
Tabel 3. Simbol-Simbol dalam *Class Diagram* (Sugiarti, 2018)

<i>Construct</i>	<i>Description</i>	<i>Symbol</i>
<i>Package</i>	<i>Package</i> merupakan bungkus dari satu atau lebih <i>class</i> .	
Operasi	<i>Class</i> pada struktur sistem.	
Antarmuka / <i>Interface</i>	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.	
Asosiasi	Relasi antar <i>Class</i> dengan makna umum, asosiasi biasanya disertai dengan <i>multiplicity</i> .	
Asosiasi berarah / <i>directed</i> asosiasi	Relasi antar <i>Class</i> dengan makna yang <i>Class</i> satu digunakan oleh <i>Class</i> yang lain, biasanya disertai <i>multiplicity</i> .	
Generalisasi	Relasi antar <i>Class</i> dengan makna generalisasi-spesialisasi (umum-khusus).	
Kebergantungan / <i>Dependency</i>	Relasi antar <i>Class</i> dengan makna kebergantungan antar <i>Class</i> .	
<i>Agregasi</i>	Relasi antar <i>Class</i> dengan makna semua bagian (<i>Whole-part</i>).	

Class Diagram menggambarkan struktur dan deskripsi *class*, *package*, dan objek yang saling terhubung (Safaat, 2018). Pada sistem OO (*Object Oriented*) dengan menggunakan kelas, atribut, dan relasi. Model *Class Diagram* memiliki isi seperti atribut dan operasi yang terjadi pada sistem. *Class Diagram* memiliki fungsi seperti hubungan antara kelas dan dapat digunakan pada *database* (Erik et al., 2019).

E. Metode *Waterfall*

Waterfall merupakan salah satu metode dalam *Software Development Life Cycle* (SDLC) yang mempunyai ciri khas pengerjaan yaitu setiap fase dalam metode *Waterfall* harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. Artinya fokus terhadap masing-masing fase dapat dilakukan secara lebih maksimal karena jarang terdapat pengerjaan yang sifatnya paralel walaupun dapat saja terjadi paralelisme dalam *Waterfall* (Yurindra, 2017). Pada metode *waterfall* pendekatan pengembangan dilakukan berurutan mengalir kebawah seperti air terjun. Melalui fase analisis persyaratan, desain, implementasi, pengujian (validasi), integrasi, dan pemeliharaan. Metode *waterfall* adalah pendekatan teknik tradisional diterapkan pada rekayasa perangkat lunak (Bishan et al., 2017). Metode ini melakukan pendekatan secara sistematis dan urut mulai dari tahapan kebutuhan sistem lalu ke tahap analisis, *desain*, *coding*, *testing/verification*, dan *maintenance*. Dengan metode *waterfall* bermanfaat dalam mengidentifikasi persyaratan sebelum sistem dikembangkan (Gerhard et al., 2017). Metode *Waterfall* sebenarnya merupakan *Linear Sequential Model* (LSM), sering juga disebut dengan “*classic life cycle*”. Metode ini muncul pertama kali sekitar tahun 1970 sehingga dianggap kuno, tetapi model/metode ini paling banyak dipakai dalam *Software Engineering* (SE). Metode ini melakukan pendekatan secara sistematis dan berurutan mulai dari tahapan perencanaan kebutuhan sistem lalu ke tahap analisis, desain, *coding*, *testing* atau *verification*, dan *maintenance*. Contoh gambar metode *Waterfall* terdapat pada Gambar 6.



Gambar 6. Metode *Waterfall* (Ramadhan dan Wahab, 2019)

F. Testing

Software testing adalah suatu proses yang penting dalam pengembangan perangkat lunak untuk memverifikasi apakah sistem memenuhi persyaratan pengguna dan spesifikasi. Pemeriksaan perangkat lunak berfungsi pemenuhan persyaratan pengguna dan untuk menyampaikan aplikatif fungsional pada perangkat lunak sehingga membuat pengujian perangkat lunak menjadi proses yang kritis. Pengujian perangkat lunak juga dapat dilakukan dengan manual atau otomatis menggunakan alat pengujian perangkat lunak (Gamido *and* Gamido, 2019). Berikut adalah 2 macam testing yang digunakan dalam penelitian ini:

1. *Blackbox Testing*

Blackbox testing merupakan salah satu bentuk pengujian pada perangkat lunak. *Blackbox testing* digunakan untuk mengetahui seberapa baik fungsionalitas sebuah perangkat lunak yang telah dibuat. *Blackbox testing* tidak mempertimbangkan struktur internal dan implementasi *case* yang sedang dalam proses pengujian, tetapi difokuskan pada *input* dan *output* pengujian. Pendekatan pengujian menggunakan *blackbox* berguna mengamati perilaku dari perangkat lunak terhadap *input* tertentu (Hanza *and* Hammad, 2020).

Blackbox testing sering digunakan untuk melakukan pengujian aplikasi *mobile* dan menggunakan beberapa *tools* dalam pengujiannya seperti *Mobile-Test* (Bo et al., 2007). Dalam pengujian diperlukan pengujian otomatis, tujuannya adalah dapat mempermudah *blackbox testing*, saat ini *Auto-Black-Test* yang digunakan dalam *blackbox testing* untuk melakukan pengujian otomatis (Mariani et al., 2011).

III. METODE PENELITIAN

A. Tempat dan Waktu Penelitian

Penelitian dilaksanakan di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berada di Jalan Soemantri Brojonegoro No.1 Gedung Meneng, Bandar Lampung. Penelitian ini dilakukan pada semester ganjil tahun ajaran 2019-2020.

B. Alat Pendukung

Pada penelitian ini dibutuhkan alat pendukung berupa perangkat keras dan perangkat lunak, yaitu:

1. Perangkat Keras

Perangkat keras yang digunakan pada penelitian ini adalah dengan menggunakan laptop dan *smartphone*.

- Laptop dengan spesifikasi:

Processor : CPU 2.80 GHz, 4 Cores, 8 Threads, Cache 6MB

Memory : 16 GB DDR4 2400Mhz

Storage : 512 GB M.2 SATA SSD, 1 TB 5200rpm HDD

Graphics : 6GB GDDR5, 192.2 GB/s Bandwidth

- *Smartphone* Android dengan spesifikasi:

Processor : Snapdragon 626 2.2GHz octa-core

RAM : 4 GB

Internal : 64 GB

2. Perangkat Lunak

Perangkat lunak yang digunakan untuk melakukan pada penelitian ini adalah sebagai berikut:

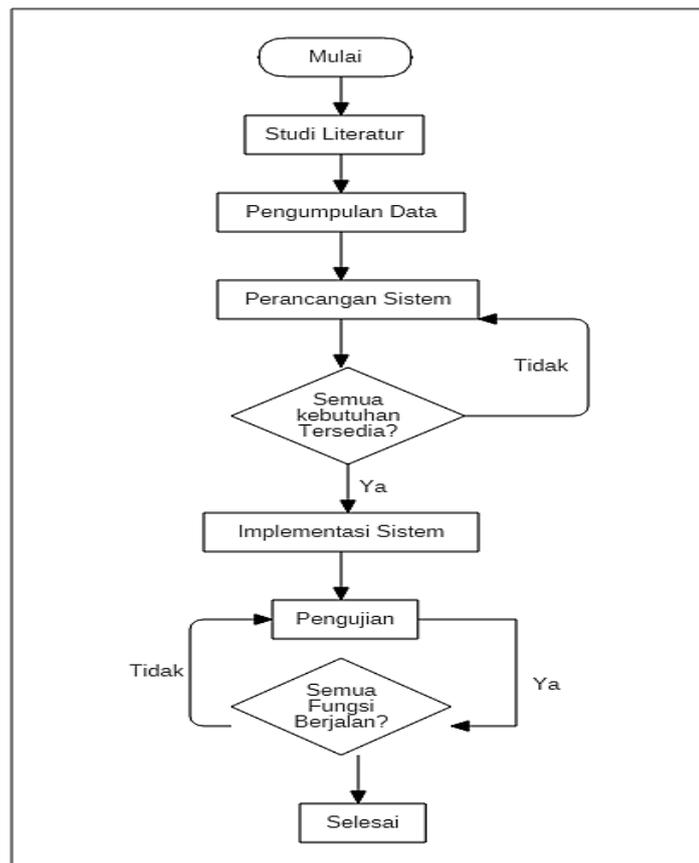
- Sistem Operasi Windows 10 Home Single Language 64-bit
- Android Studio v4.1.2
- Database Server MariaDB v10.3.31
- Web Server PHP v7.3.30
- Balsamiq Mockups v3.5.16
- StarUML v3.0.2
- Postman v9.0.8
- Text Editor Notepad++

C. Tahapan Penelitian

Tahapan penelitian yang dilakukan dalam pembuatan perangkat lunak dikerjakan berdasarkan diagram alir penelitian yang terdapat pada Gambar 7. Tahapan ini, merupakan implementasi dari metode pengembangan perangkat lunak model *waterfall*. Dalam pengembangannya metode ini dilakukan secara terstruktur dari tahapan awal sampai dengan selesai. Tahapan pada metode ini, dimulai dari analisis, desain, implementasi, dan pengujian. Metode ini disebut *waterfall* dikarenakan tahapan dari metode ini dilakukan secara berurutan dan berkelanjutan.

Pada metode penelitian dengan model *waterfall* terdapat beberapa tahapan, dimulai dari mempelajari studi literatur mengenai sistem, kemudian mengumpulkan data yang dibutuhkan untuk keperluan sistem, melakukan perancangan sistem baik

Unified Modelling Language (UML) maupun *design user interface* dari sistem, mengimplementasikan sistem, dan menguji kelayakan sistem. Berikut adalah penjelasan dari masing-masing tahapan sesuai dengan tahapan yang ada pada gambar diagram alir.



Gambar 7. Diagram Alir Tahapan Penelitian

1. Studi Literatur

Studi literatur merupakan tahapan awal dan sebagai hal yang mendasar untuk dilakukan dalam memulai pembuatan perangkat lunak. Pada tahap ini, dilakukan pengumpulan bahan studi yang berhubungan dengan penelitian mengenai perangkat lunak tanggap bencana kebakaran, dan juga mempelajari berbagai hal yang berkaitan mengenai penelitian-penelitian serupa dengan sistem yang akan dibuat. Terdapat beberapa sumber referensi yang akan digunakan untuk melakukan

pembuatan sistem ini, seperti buku, *e-book*, skripsi, dan jurnal yang berkaitan dengan tema penelitian.

2. Pengumpulan Data

Pengumpulan data dilakukan untuk mendapatkan informasi yang dibutuhkan untuk pembuatan perangkat lunak. Salah satu cara untuk mendapatkan data adalah dengan cara diperoleh secara langsung dari narasumber. Data tersebut diperoleh dari hasil wawancara dan konsultasi dengan petugas PFSO dan pegawai yang berwenang untuk menyiapkan data dan kebutuhan yang diperlukan untuk pembuatan perangkat lunak ini. Data-data yang diperlukan contohnya seperti alur kerja, model data, kebutuhan informasi pada sistem, dan rancangan tampilan dari perangkat lunak yang akan dibuat.

Selain itu, beberapa data dapat diperoleh dan dikumpulkan dari sumber yang tersedia, seperti dokumen dan bahan-bahan lain yang bisa didapatkan di PT. Pelabuhan Indonesia II Cabang Panjang. Data yang dapat diperoleh di antaranya denah lokasi Alat Pemadam Api Ringan (APAR), tabel rincian informasi Alat Pemadam Api Ringan (APAR), jalur-jalur evakuasi, pemetaan letak *hydrant*, denah lokasi alat Pertolongan Pertama pada Kecelakaan (P3K), rincian isi serta informasi alat Pertolongan Pertama pada Kecelakaan (P3K), data informasi mengenai mobil pemadam, dan data personel pemadam yang sedang bertugas.

Observasi juga dilakukan pada saat melakukan pengumpulan data, observasi yang dilakukan adalah pengamatan secara langsung tentang berbagai hal yang dianggap penting guna melengkapi data-data yang diperlukan. Pada proses observasi, hal yang dilakukan ialah melihat informasi dan permasalahan pada mengawasi pengelolaan alat pemadam kebakaran seperti Alat Pemadam Api Ringan (APAR), *hydrant*, mobil pemadam serta personel pemadam yang bertugas, dan alat Pertolongan Pertama pada Kecelakaan (P3K) yang ada. Selain itu juga, melihat dan

menentukan jalur evakuasi yang tepat untuk digunakan korban ketika terjadi kebakaran.

3. Perancangan Sistem

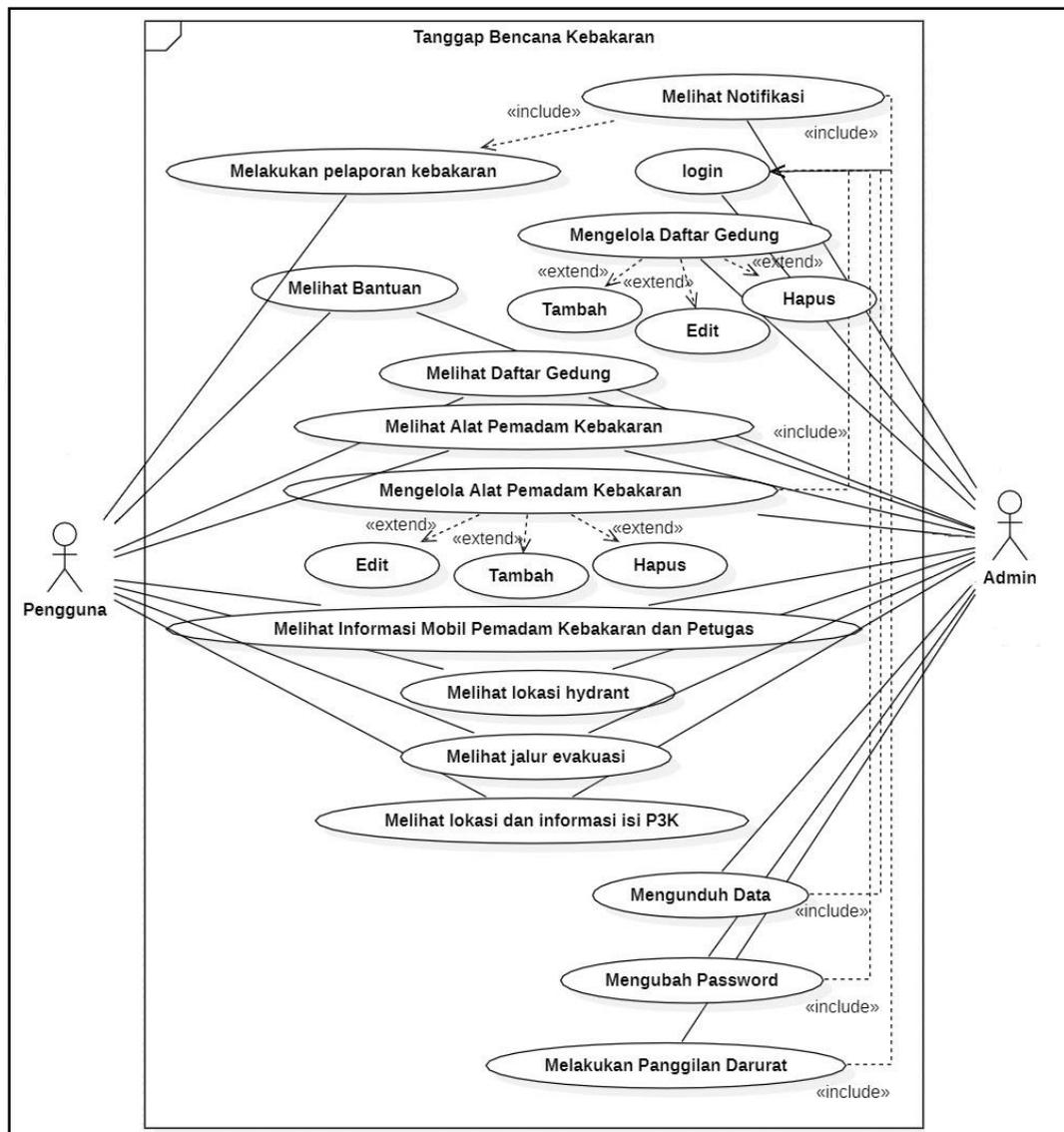
Desain rancangan perangkat lunak dapat dijelaskan dengan alur diagram agar pengguna dapat memahami proses dari sistem. Maka dari itu, dibuat beberapa desain rancangan sistem yang diilustrasikan dengan *Usecase Diagram*, *Activity Diagram*, *Class Diagram*, dan *User Interface*.

3.1 *Usecase Diagram*

Usecase Diagram adalah diagram yang dapat menggambarkan, melakukan dokumentasi serta membantu melihat spesifikasi dari sistem agar mudah dipahami. *Usecase Diagram* terdiri dari *Actor*, *Usecase*, dan hubungan yang terjadi antara *Usecase* dan *Actor*.

Gambar 8 merupakan *Usecase Diagram* dari perangkat lunak tanggap bencana kebakaran PT. Pelabuhan Indonesia II berbasis Android. Pada *Usecase* tersebut dijelaskan bahwa terdapat 2 aktor, pengguna dan admin. Pengguna dapat melakukan akses terhadap aplikasi tanpa melakukan *login*. Pengguna dapat melakukan pelaporan kebakaran, melihat bantuan, melihat daftar gedung, melihat alat pemadam kebakaran, melihat informasi mobil pemadam kebakaran serta petugasnya, melihat lokasi *hydrant*, melihat jalur evakuasi, dan melihat lokasi serta informasi mengenai alat Pertolongan Pertama pada Kecelakaan (P3K). Admin dapat melakukan panggilan darurat, melihat dan mengelola daftar gedung, melihat dan mengelola alat pemadam kebakaran, melihat informasi mobil pemadam kebakaran dan petugas, melihat lokasi *hydrant*, melihat jalur evakuasi, melihat informasi serta isi alat Pertolongan Pertama pada Kecelakaan (P3K), mengubah *password*, melihat notifikasi laporan, dan mengunduh data. Untuk melakukan

pengelolaan seperti menambah data, *edit* data, menghapus data, mengubah *password*, melihat notifikasi laporan, mengunduh data, dan melakukan panggilan ke nomor darurat maka admin diharuskan untuk *login*.



Gambar 8. Usecase Diagram Perangkat Lunak Tanggap Bencana Kebakaran

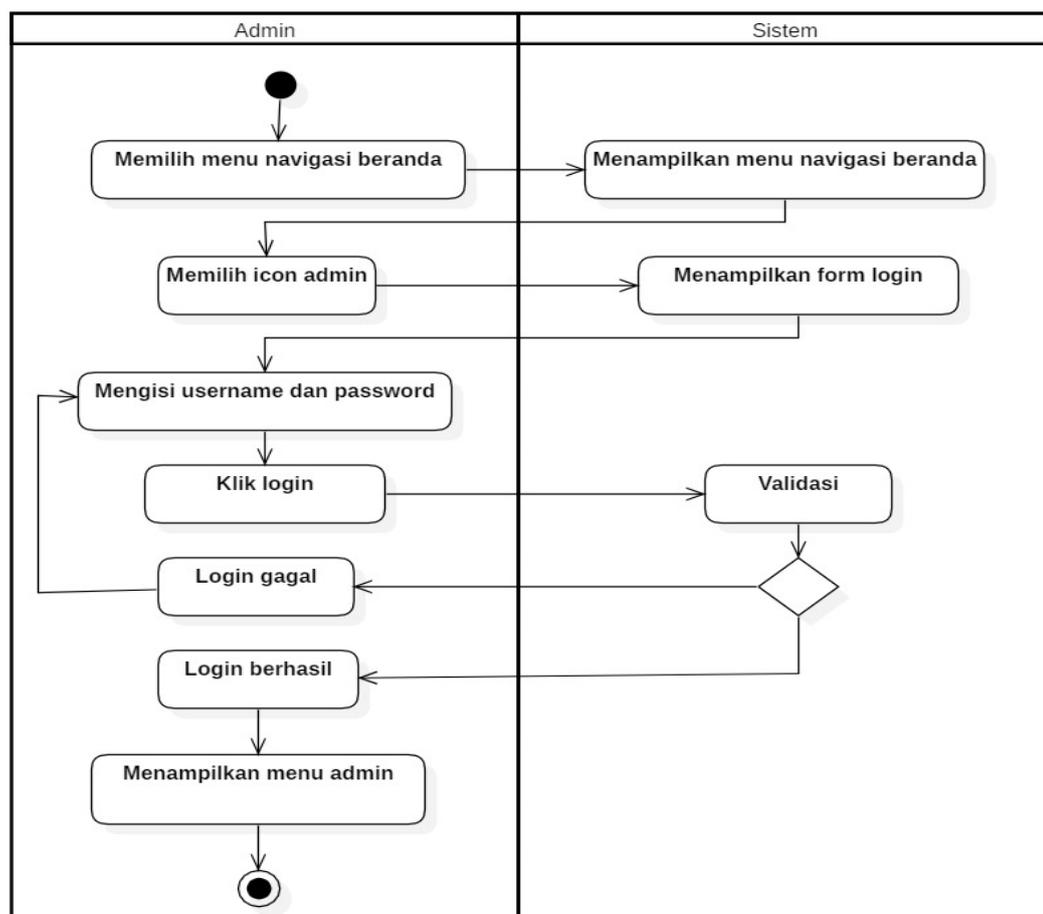
3.2 Activity Diagram

Activity Diagram menggambarkan fungsionalitas pada alur kerja sistem. Activity Diagram juga dapat membantu dalam merepresentasikan Usecase Diagram yang

telah dibuat. *Activity Diagram* berisi langkah-langkah cara kerja pada sistem. Pada pengembangan sistem, *Activity Diagram* digunakan untuk memudahkan penggambaran alur atau proses yang terjadi dalam sistem perangkat lunak. *Activity Diagram* akan menjelaskan setiap proses yang berjalan. Contohnya adalah seperti fitur tambah, *edit*, hapus, dan lihat detail dari setiap data yang ada pada perangkat lunak tanggap bencana kebakaran PT. Pelabuhan Indonesia II Cabang Panjang.

Berikut ini adalah *Activity Diagram* yang dibuat untuk menggambarkan fungsi-fungsi yang ada pada perangkat lunak tanggap bencana kebakaran PT. Pelabuhan Indonesia II berbasis Android.

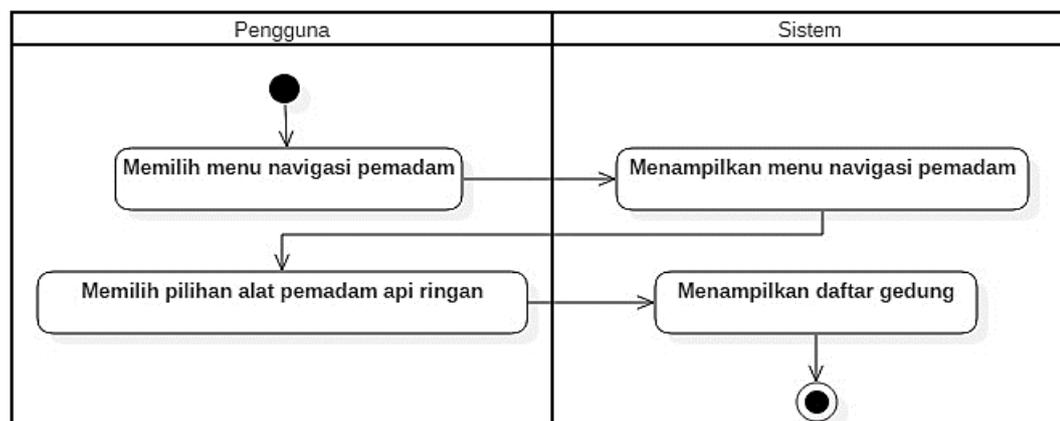
3.2.1 *Activity Diagram Login Admin*



Gambar 9. *Activity Diagram Login Admin*

Activity Diagram dari *login* admin dapat dilihat pada Gambar 9. Berdasarkan diagram tersebut, pertama admin memilih menu navigasi beranda. Kemudian melakukan akses ke bagian *login* dengan cara memilih ikon admin. Sistem akan menampilkan halaman admin, halaman tersebut berisi *form login*. Admin memasukan *username* dan *password* yang dimiliki, sistem akan melakukan validasi terhadap *username* dan *password*, jika salah maka sistem kembali ke halaman sebelumnya dan jika benar maka *login* berhasil akan masuk ke menu admin.

3.2.2 *Activity Diagram* Lihat Daftar Gedung



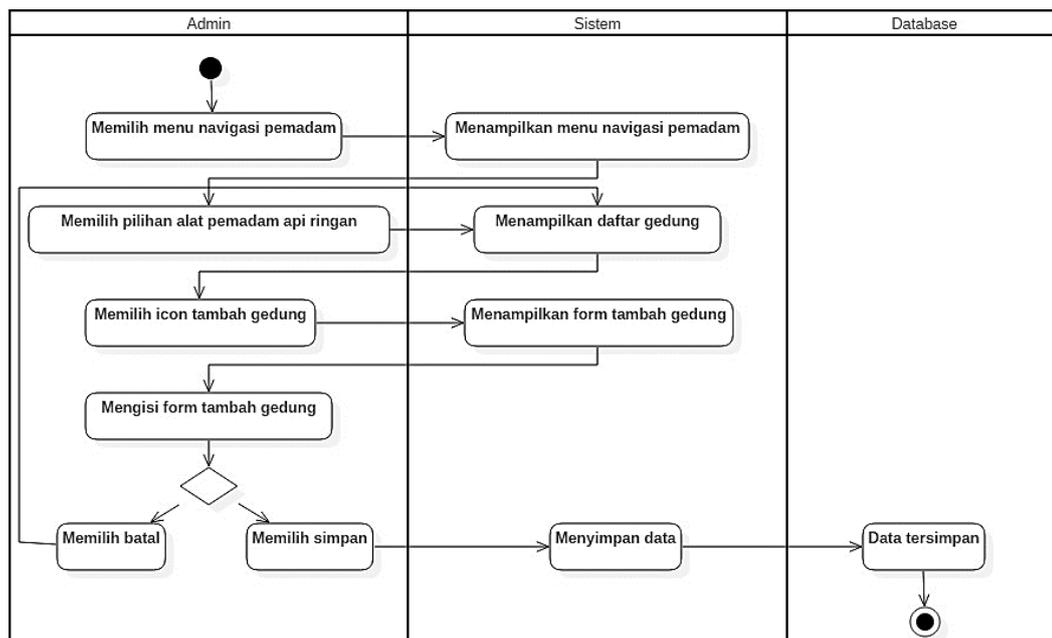
Gambar 10. *Activity Diagram* Lihat Daftar Gedung

Gambar 10 merupakan *Activity Diagram* tentang alur untuk melihat daftar gedung. Diagram ini menjelaskan tentang proses melihat data gedung yang sudah ada. Pengguna melakukan akses ke menu navigasi pemadam. Sistem akan menampilkan halaman tersebut, Kemudian pengguna dapat memilih pilihan alat pemadam api ringan dan sistem akan menampilkan daftar gedung dari *database*.

3.2.3 *Activity Diagram* Tambah Gedung

Gambar 11 menjelaskan alur dari *Activity Diagram* tambah daftar gedung, untuk melakukan penambahan gedung, admin diharuskan untuk sudah *login* sebelumnya.

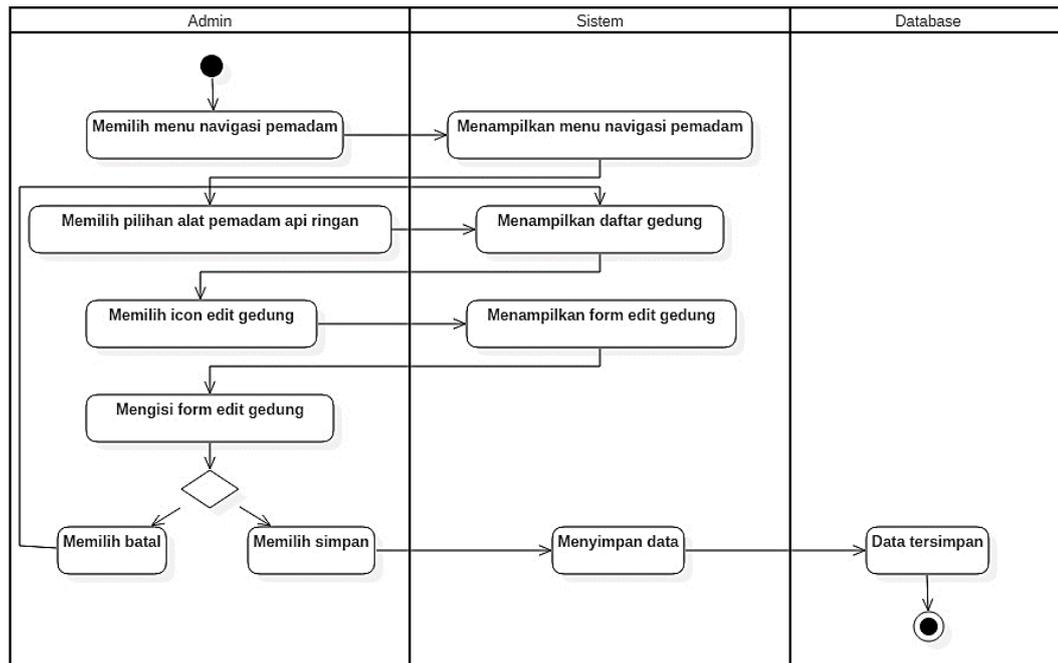
Pilih menu navigasi pemadam, kemudian pilih alat pemadam api ringan dan halaman daftar gedung akan ditampilkan, pilih ikon tambah gedung. Sistem akan ditampilkan *form* tambah gedung. Setelah admin selesai mengisi *form* tambah gedung dan menekan tombol simpan, maka sistem akan melakukan proses simpan data ke dalam *database*. Admin dapat memilih tombol batal jika tidak ingin menambahkan gedung baru.



Gambar 11. *Activity Diagram* Tambah Gedung

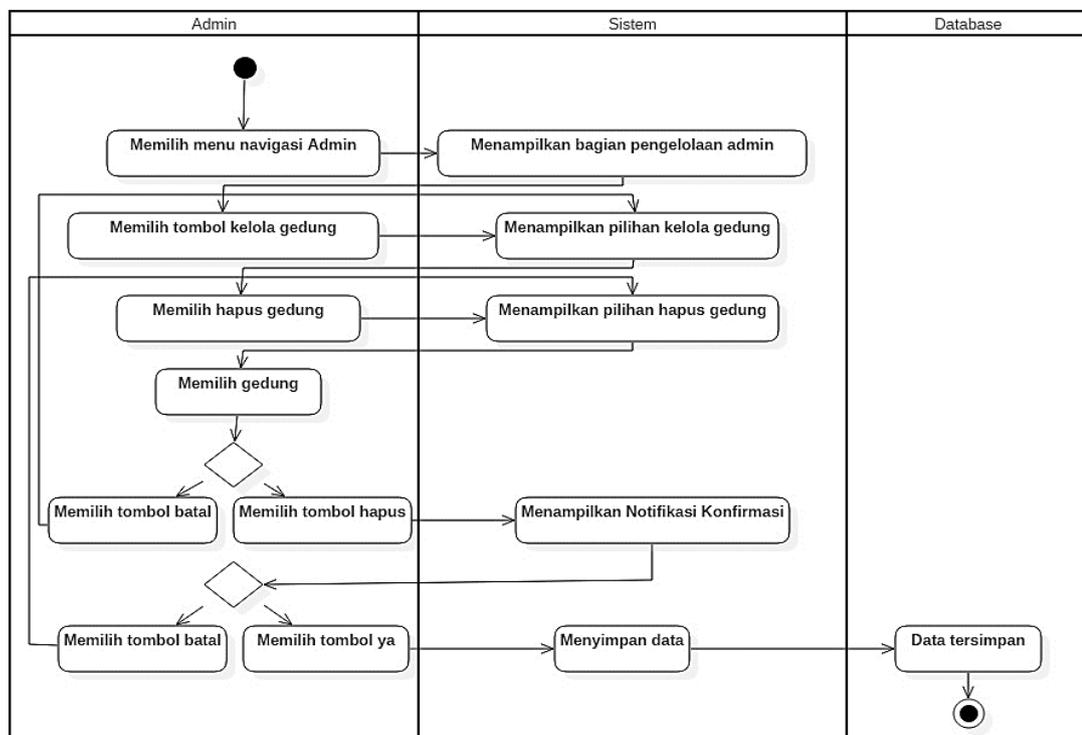
3.2.4 *Activity Diagram* Edit Gedung

Activity Diagram pada Gambar 12 menjelaskan alur untuk melakukan *edit* pada daftar gedung yang dilakukan oleh admin. Fitur ini hanya dapat dilakukan oleh admin yang telah melakukan *login*. Pilih menu navigasi pemadam, kemudian pilih pemadam api ringan. Pada halaman daftar gedung, pilih ikon *edit* gedung, sistem akan ditampilkan *form edit* gedung. Setelah admin mengisi *form edit* gedung dan menekan tombol simpan, maka sistem akan melakukan proses simpan data yang telah diperbaiki ke dalam *database*. Jika tidak ingin melakukan *edit* maka pilih tombol batal.



Gambar 12. Activity Diagram Edit Gedung

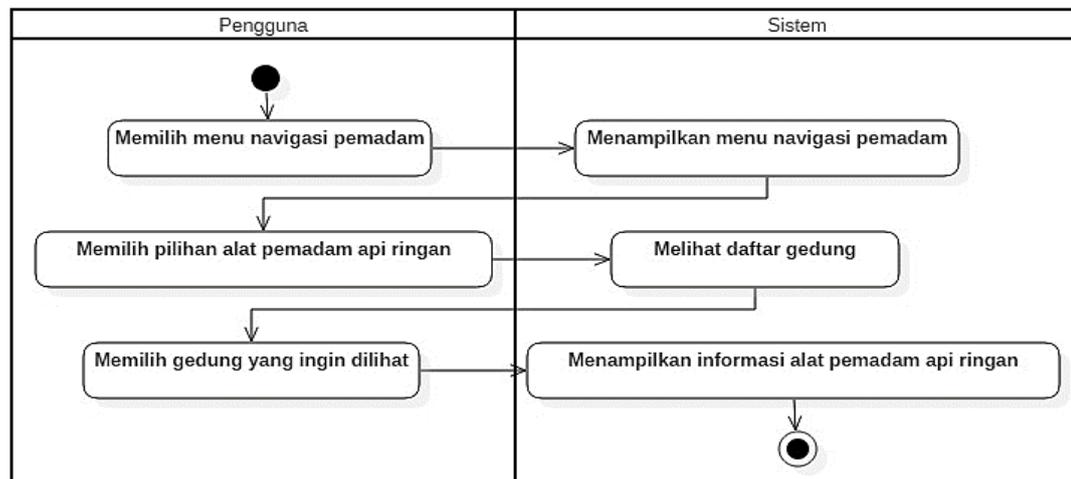
3.2.5 Activity Diagram Hapus Gedung



Gambar 13. Activity Diagram Hapus Gedung

Gambar 13 merupakan proses dari *Activity Diagram* hapus gedung. Setelah *login* admin mengakses menu navigasi pemadam, pilih alat pemadam api ringan. Pilih ikon hapus gedung, sistem akan menampilkan pilihan gedung yang akan dihapus. Setelah admin memilih gedung yang akan dihapus, akan tampil pesan konfirmasi “Ya” dan “Batal”. Jika memilih “Batal” maka sistem akan kembali ke halaman daftar gedung, jika memilih “Ya” maka sistem akan melakukan proses hapus data ke dalam *database*.

3.2.6 *Activity Diagram* Lihat Alat Pemadam Api Ringan

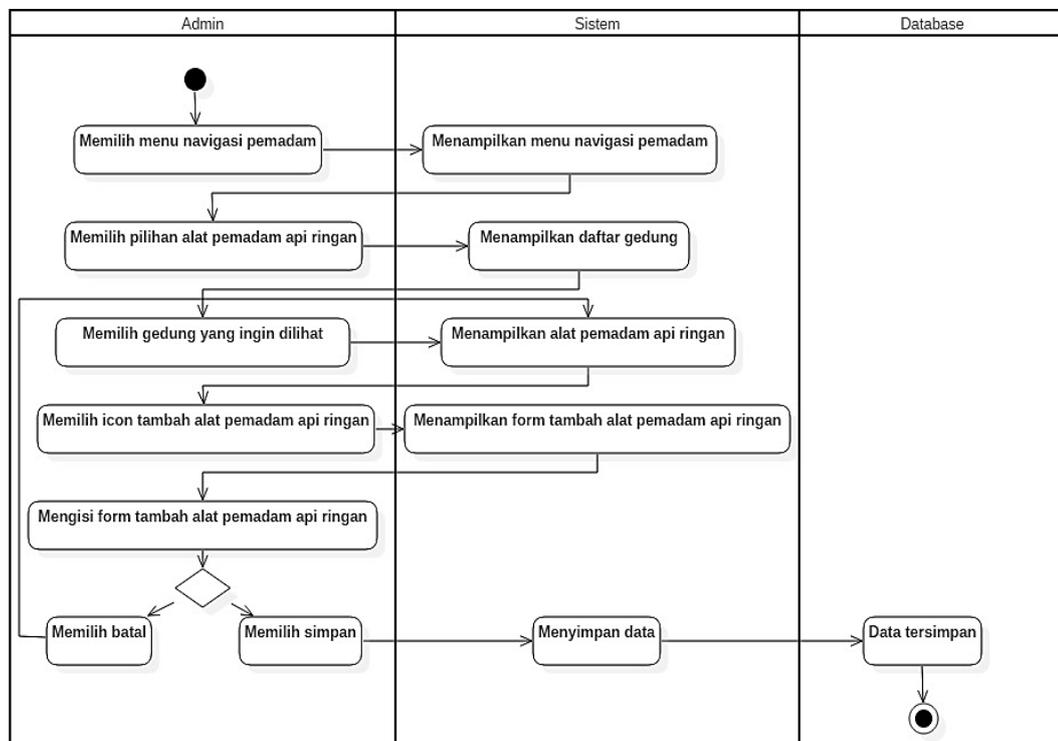


Gambar 14. *Activity Diagram* Lihat Alat Pemadam Api Ringan

Proses *Activity Diagram* untuk melihat alat pemadam kebakaran dapat dilihat pada Gambar 14. Pertama, pengguna melakukan akses ke menu navigasi pemadam dan sistem akan menampilkan halaman tersebut. Pilih alat pemadam api ringan dan sistem akan menampilkan halaman berisi daftar gedung yang ada. Pengguna memilih gedung yang ingin dilihat alat pemadam api ringannya, kemudian sistem akan menampilkan informasi mengenai alat pemadam api yang ada di gedung tersebut.

3.2.7 Activity Diagram Tambah Alat Pemadam Api Ringan

Gambar 15 merupakan *Activity Diagram* untuk menambahkan alat pemadam api ringan. Proses penambahan dilakukan oleh admin yang telah *login*. Admin melakukan akses ke menu navigasi pemadam, pilih alat pemadam api ringan. Pilih gedung yang diinginkan. Selanjutnya, admin memilih ikon tambah alat pemadam api ringan, sistem akan ditampilkan *form* tambah alat. Setelah admin mengisi *form* tambah alat, tekan tombol “simpan” agar sistem akan melakukan proses simpan ke dalam *database*. Jika tidak ingin melakukan penambahan data maka klik “batal”.

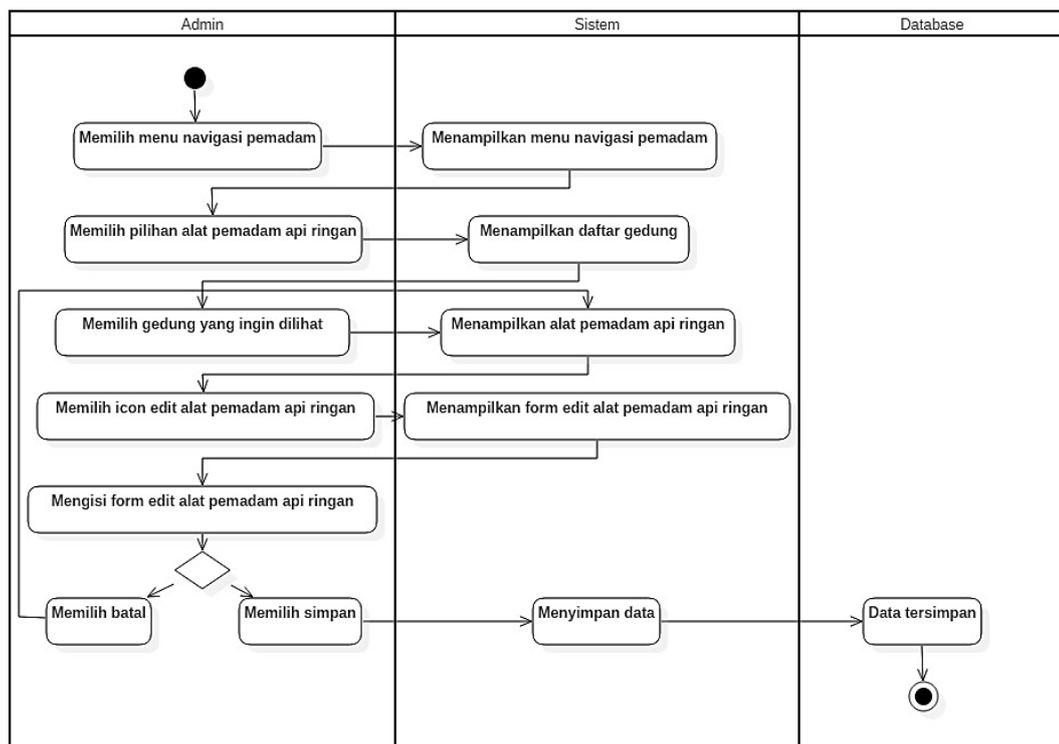


Gambar 15. Activity Diagram Tambah Alat Pemadam Api Ringan

3.2.8 Activity Diagram Edit Alat Pemadam Api Ringan

Gambar 16 menjelaskan proses *edit* alat pemadam api ringan. Admin yang sudah *login* melakukan akses ke menu navigasi pemadam, pilih pemadam api ringan. Selanjutnya, pilih gedung yang diinginkan. Kemudian admin memilih ikon *edit*.

Setelah admin mengisi *form edit*, admin dapat menekan tombol “simpan”, maka sistem akan melakukan proses penyimpanan data ke dalam *database*. Jika tidak ada data yang ingin diubah dan ingin kembali ke halaman pemadam api ringan maka admin dapat memilih tombol “batal”.

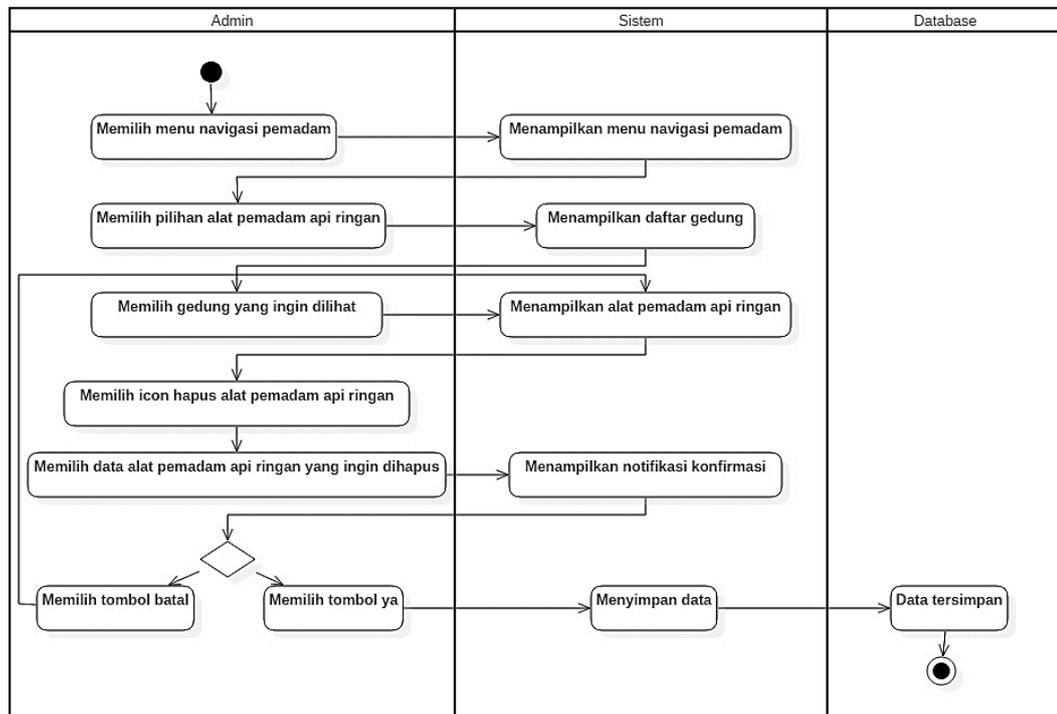


Gambar 16. *Activity Diagram Edit Alat Pemadam Api Ringan*

3.2.9 *Activity Diagram Hapus Alat Pemadam Api Ringan*

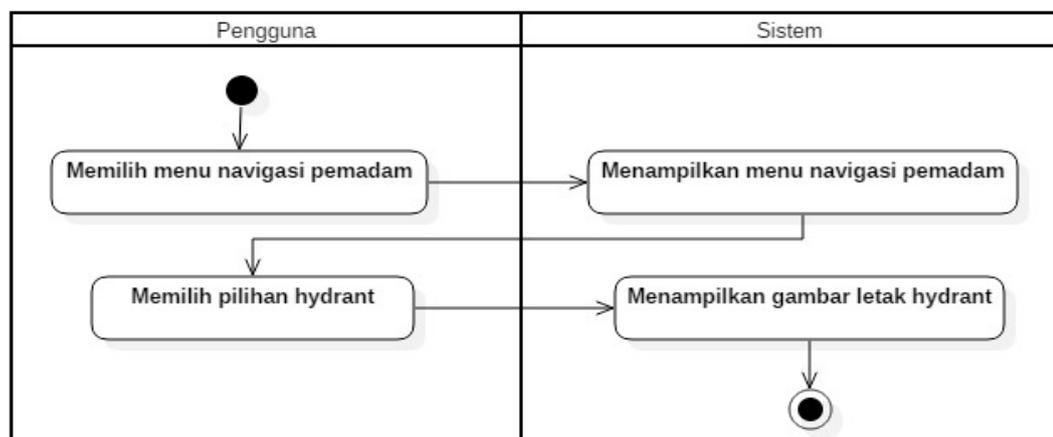
Gambar 17 merupakan *Activity Diagram* dari proses hapus alat pemadam api ringan. Proses dilakukan oleh admin, admin harus sudah melakukan *login* terlebih dahulu. Pertama, admin memilih menu navigasi pemadam. Selanjutnya, admin memilih alat pemadam api ringan. Sistem akan menampilkan daftar gedung. Admin memilih gedung yang ingin dilihat alatnya, kemudian pada halaman alat pemadam api ringan pilih ikon hapus, sistem akan ditampilkan pilihan alat pemadam kebakaran yang akan dihapus. Setelah memilih akan tampil pesan konfirmasi “Ya” dan “Batal”. Jika admin memilih “Batal” maka sistem akan kembali ke halaman

alat pemadam api ringan, jika memilih “Ya” maka sistem akan melakukan proses hapus data dari *database*. Apabila tidak ada yang akan dihapus maka admin dapat memilih tombol pilih batal.



Gambar 17. *Activity Diagram* Hapus Alat Pemadam Api Ringan

3.2.10 *Activity Diagram* Lihat Hydrant



Gambar 18. *Activity Diagram* Lihat Hydrant

Activity Diagram lihat *hydrant* dapat dilihat pada Gambar 18. Diagram ini menjelaskan alur melihat letak *hydrant* yang ada pada PT. Pelabuhan Indonesia Cabang Panjang. Pengguna melakukan akses ke menu navigasi pemadam. Sistem akan menampilkan halaman pemadam, kemudian pilih *hydrant*.

3.2.11 *Activity Diagram* Lihat Mobil Pemadam Kebakaran

Gambar 19 menjelaskan alur *Activity Diagram* untuk melihat informasi mobil pemadam yang dimiliki oleh PT. Pelabuhan Indonesia Cabang Panjang beserta data informasi mengenai petugas pemadam kebakaran. Untuk mengakses halaman ini, pilih menu navigasi pemadam. Sistem akan menampilkan pilihan alat pemadam yang ada. Kemudian, pilih mobil pemadam kebakaran, maka halaman informasi mengenai mobil pemadam kebakaran serta petugasnya akan ditampilkan.

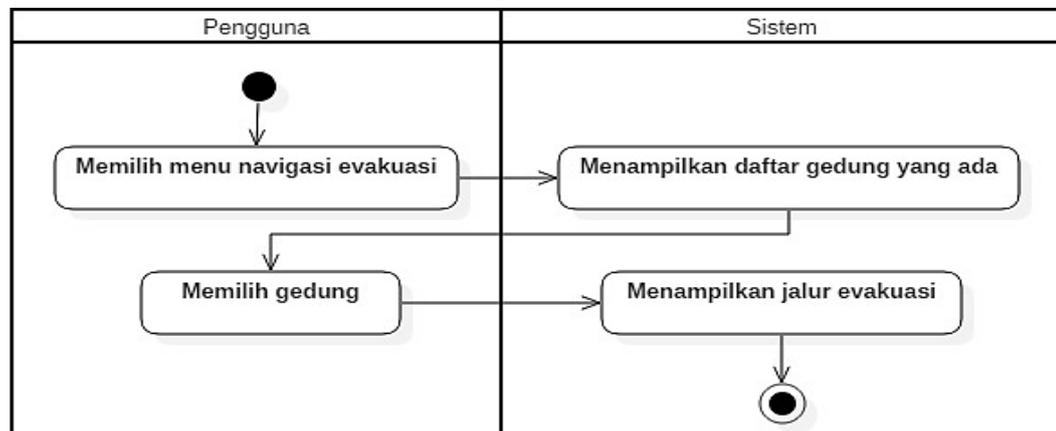


Gambar 19. *Activity Diagram* Lihat Mobil Pemadam Kebakaran

3.2.12 *Activity Diagram* Lihat Jalur Evakuasi

Activity Diagram yang menjelaskan alur untuk melihat jalur evakuasi pada aplikasi tanggap bencana kebakaran PT. Pelabuhan Indonesia Cabang Panjang terdapat pada Gambar 20. Cara mengakses menu ini, pada menu navigasi, pilih evakuasi.

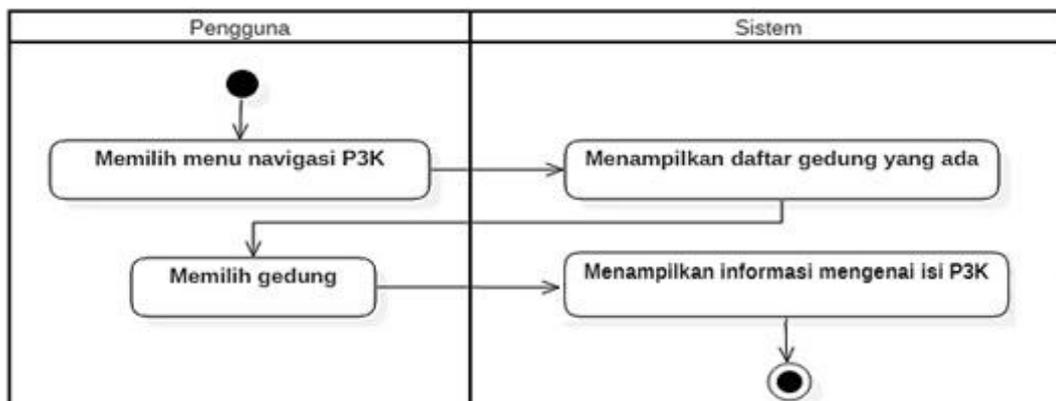
Kemudian akan ditampilkan gedung yang ada pada PT. Pelabuhan Indonesia Cabang Panjang. Pilih gedung yang ingin dilihat jalur evakuasinya dan sistem akan menampilkan jalur evakuasi pada gedung tersebut.



Gambar 20. Activity Diagram Lihat Jalur Evakuasi

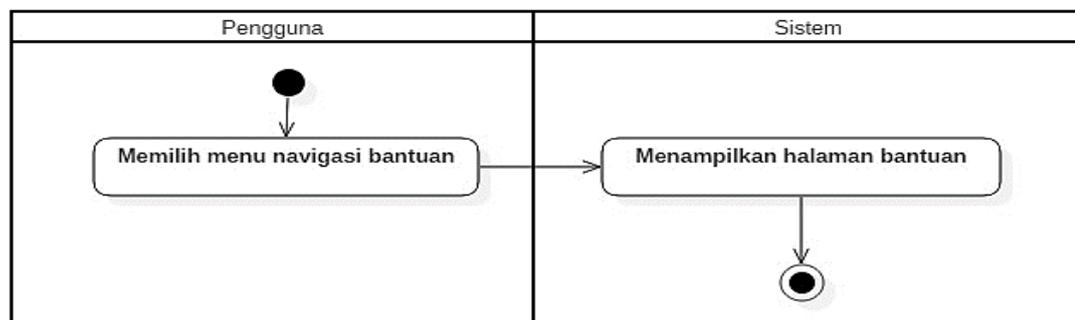
3.2.13 Activity Diagram Lihat Alat Pertolongan Pertama pada Kecelakaan

Gambar 21 adalah Activity Diagram yang menjelaskan alur untuk melihat jalur alat Pertolongan Pertama pada Kecelakaan (P3K) di PT. Pelabuhan Indonesia Cabang Panjang. Untuk melakukan akses ke halaman ini, pilih menu navigasi evakuasi, pilih gedung dan informasi mengenai P3K di gedung tersebut akan ditampilkan.



Gambar 21. Activity Diagram Lihat Pertolongan Pertama pada Kecelakaan

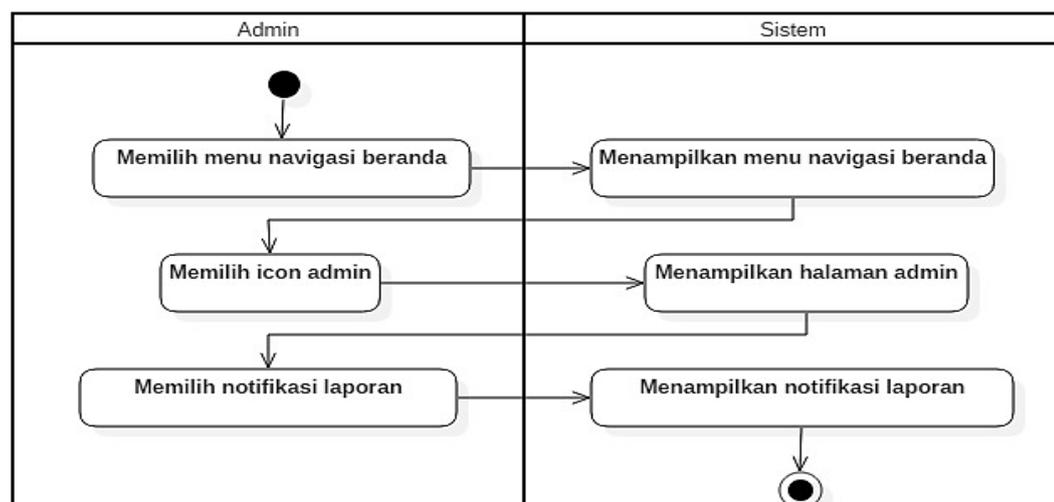
3.2.14 Activity Diagram Bantuan



Gambar 22. Activity Diagram Bantuan

Gambar 22 menjelaskan mengenai *Activity Diagram* untuk melihat bantuan tata cara penggunaan perangkat lunak Tanggap Bencana Kebakaran PT. Pelabuhan Indonesia Cabang Panjang. Untuk mengakses halaman ini, dari halaman beranda pengguna atau admin melakukan akses ke menu navigasi bantuan. Setelah itu, sistem akan menampilkan halaman bantuan.

3.2.15 Activity Diagram Pelaporan Kebakaran

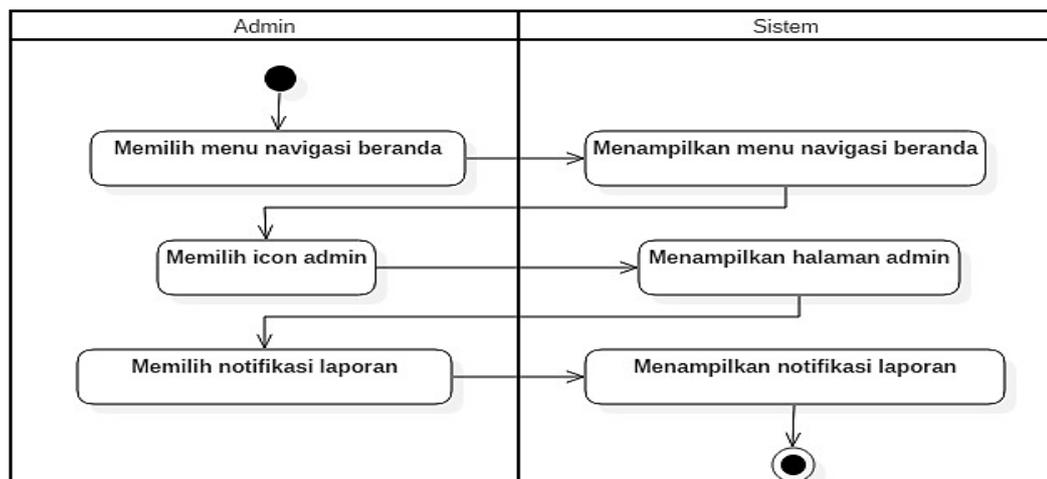


Gambar 23. Activity Diagram Pelaporan Kebakaran

Gambar 23 menggambarkan alur dari *Activity Diagram* pada saat melaporkan terjadinya kebakaran di PT. Pelabuhan Indonesia II (Persero) Cabang Panjang.

Untuk melaporkan kebakaran, pengguna melakukan akses pada menu navigasi beranda, kemudian pilih tombol laporkan kebakaran. Maka sistem akan menampilkan peta lokasi, pilih sebarikan untuk melaporkan terjadinya kebakaran kepada admin, atau tutup jika ingin kembali ke halaman beranda.

3.2.16 Activity Diagram Lihat Notifikasi Pengguna



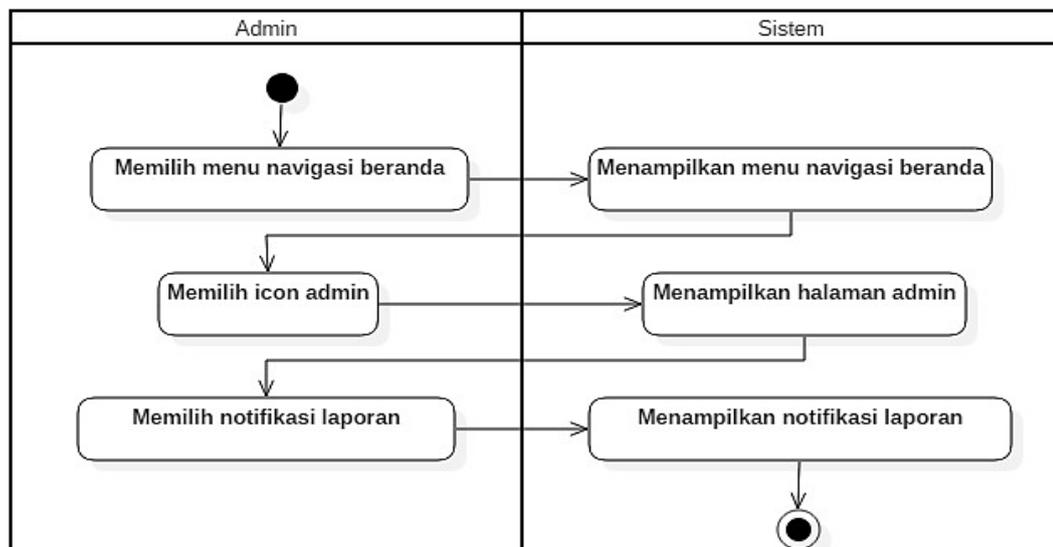
Gambar 24. Activity Diagram Lihat Notifikasi

Gambar 24 adalah tata cara melihat notifikasi pemberitahuan laporan kebakaran yang dapat diakses oleh pengguna. Untuk mengakses *Activity Diagram* ini pada halaman beranda pilih ikon notifikasi, sistem akan menampilkan notifikasi dari admin jika terjadi kebakaran.

3.2.17 Activity Diagram Lihat Notifikasi Admin

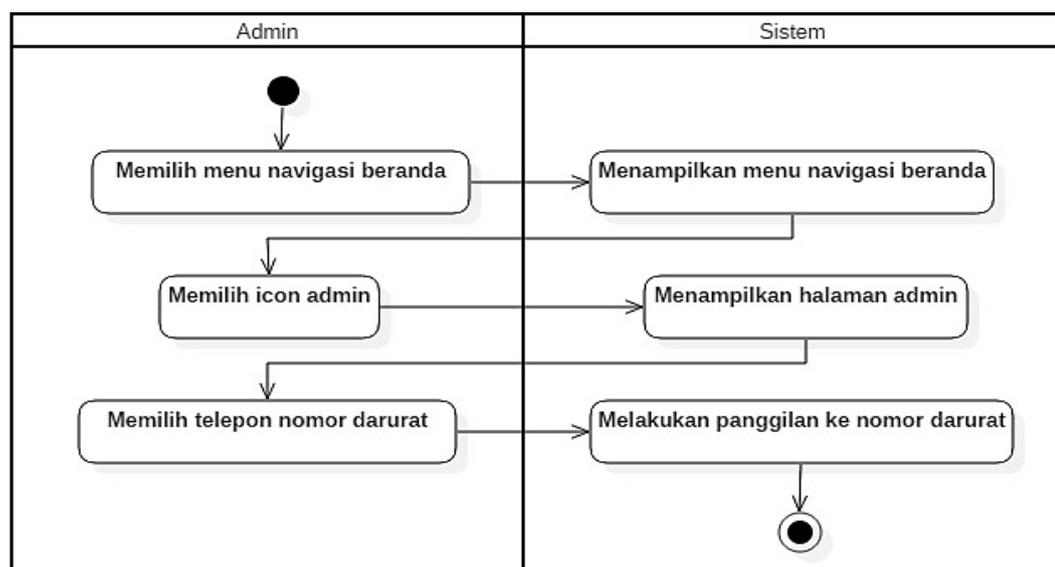
Activity Diagram pada Gambar 25 menggambarkan tata cara melihat notifikasi pemberitahuan laporan kebakaran. Untuk mengakses *Activity Diagram* ini admin diharuskan telah melakukan *login*. Dari halaman branda, pilih ikon admin. Setelah itu, akan ditampilkan halaman admin, kemudian pilih notifikasi. Sistem akan

menampilkan notifikasi dari laporan pengguna ketika terjadi kebakaran, admin dapat memilih untuk menyebarkan lokasi kebakaran keseluruhan pengguna dengan cara memilih tombol “sebar” atau mengabaikan notifikasi tersebut dengan memilih tombol “tutup”.



Gambar 25. Activity Diagram Lihat Notifikasi

3.2.18 Activity Diagram Telepon Nomor Darurat

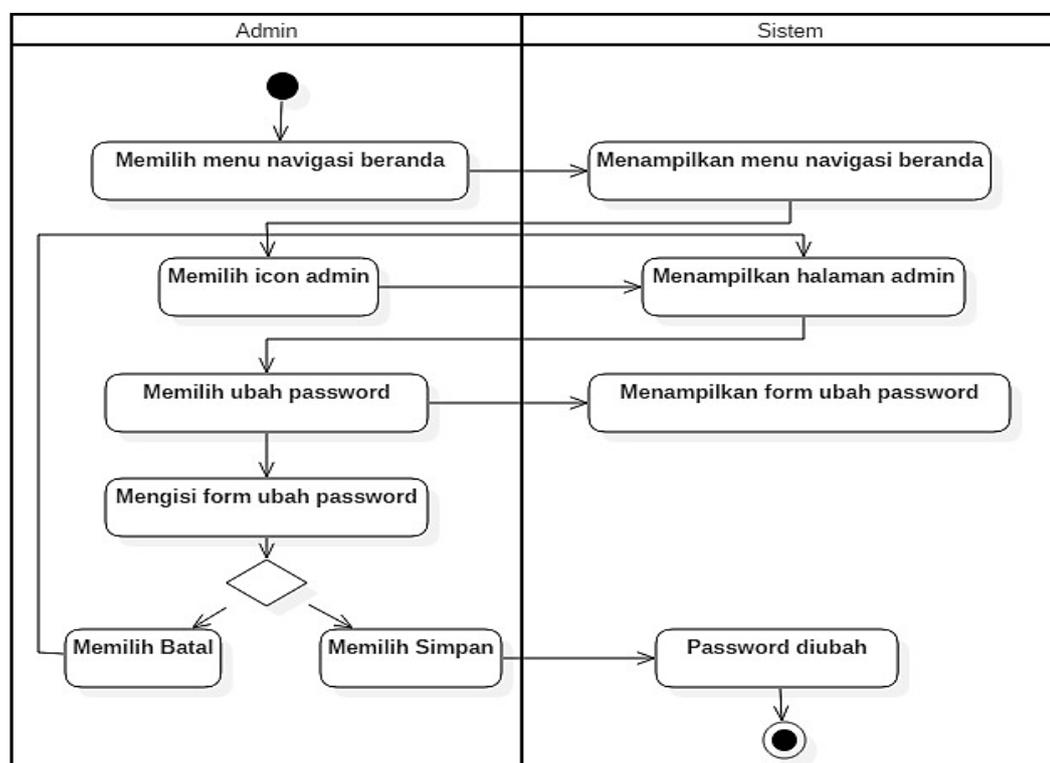


Gambar 26. Activity Diagram Telepon Nomor Darurat

Activity Diagram telepon nomor darurat pada Gambar 26 menjelaskan tentang alur dalam melakukan panggilan ke nomor darurat yang dilakukan oleh admin. Dari menu navigasi beranda, pilih ikon admin. Kemudian, halaman admin akan ditampilkan. Pilih nomor darurat yang ingin dihubungi. Sistem akan melakukan panggilan ke nomor tersebut.

3.2.19 *Activity Diagram* Ubah Password

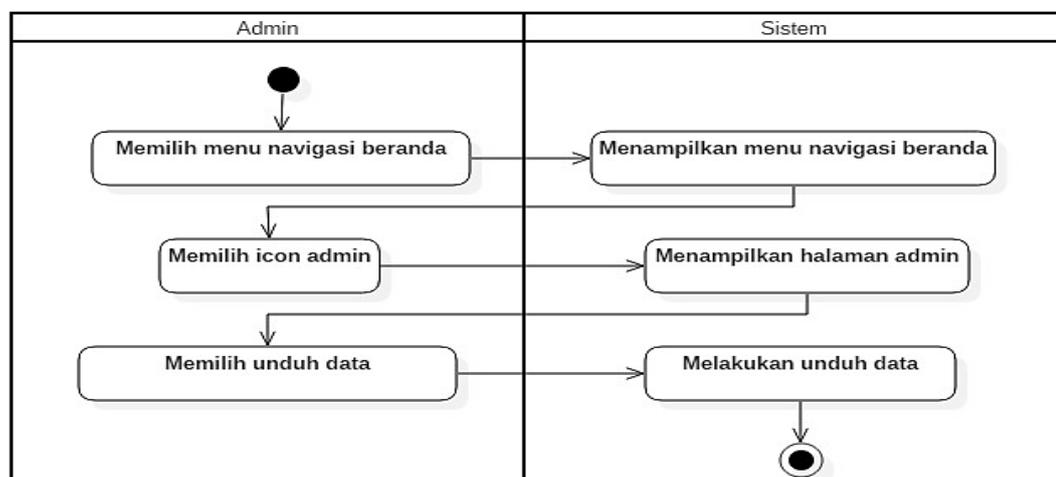
Activity Diagram pada Gambar 27 menjelaskan cara mengubah *password* admin. Pada menu navigasi beranda, pilih ikon admin. Setelah itu, akan ditampilkan halaman admin, kemudian pilih ubah *password* untuk mengubah *password* yang ada. Sistem akan menampilkan *form* ubah *password*. Setelah mengisi *form* ubah *password* tekan tombol simpan, dan data akan tersimpan pada *database*. Untuk membatalkan perubahan maka admin dapat memilih tombol batal, sehingga akan kembali ke halaman admin.



Gambar 27. *Activity Diagram* Ubah Password

3.2.20 Activity Diagram Unduh Data

Gambar 28 merupakan *Activity Diagram* yang menjelaskan cara melakukan unduh data yang ada pada perangkat lunak Tanggap Bencana Kebakaran pada PT. Pelabuhan Indonesia II Cabang Panjang. Untuk mengaksesnya, pada menu navigasi beranda pilih ikon admin. Kemudian, sistem akan menampilkan halaman admin. Pada halaman admin pilih unduh data, setelah itu sistem akan melakukan unduh data yang ada dari *database*.



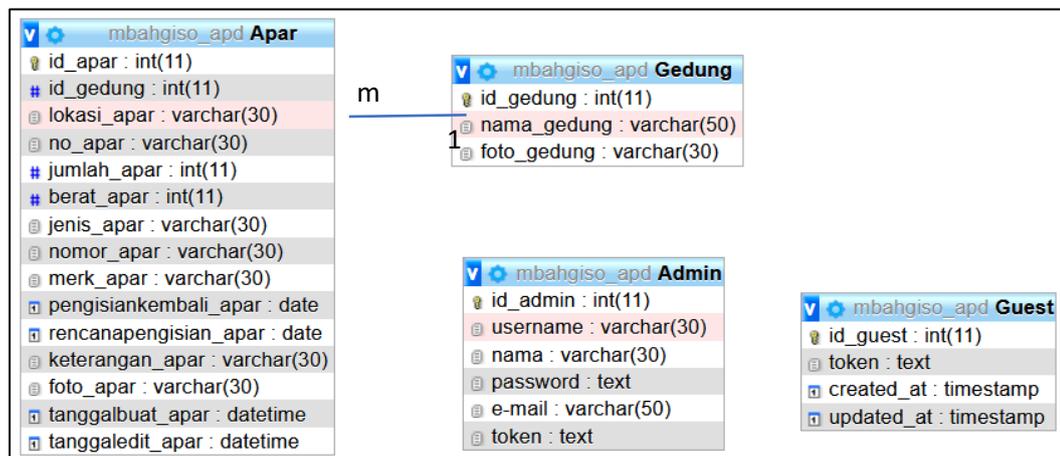
Gambar 28. Activity Diagram Unduh Data

3.3 Class Diagram

Class Diagram adalah diagram yang digunakan untuk merepresentasikan atau menggambarkan sebuah objek pada sistem dan merupakan inti dari pengembangan serta desain sistem berorientasi objek. *Class Diagram* menggambarkan atribut dari suatu sistem sekaligus menawarkan layanan yang dapat dipakai untuk melakukan manipulasi pada keadaan tersebut.

Gambar 29 merupakan *Class Diagram* dalam pengembangan Perangkat Lunak Tanggap Bencana Kebakaran PT. Pelabuhan Indonesia Cabang Panjang. Terdapat 4 *Class* yaitu, *Class* Gedung, *Class* Apar, *Class* Admin, dan *Class* Guest. *Class* Gedung memiliki relasi relasi *dependency* (kebergantungan) dengan *Class* Apar.

Class Gedung terdapat 3 atribut, yaitu *id_gedung*, *nama_gedung*, dan *foto_gedung*. *Primary key* pada *Class* ini adalah *id_gedung*. Sedangkan pada *Class* Apar terdapat 10 atribut, yaitu atribut *id_apar*, *lokasi_apar*, *jumlah_apar*, *berat_apar*, *jenis_apar*, *no_apar*, *merk_apar*, *pengisiankembali_apar*, *rencanapengisian_apar*, dan *keterangan_apar*. *Primary key* dari *Class* Apar adalah *id_apar*. Selanjutnya, *Class* Admin, terdapat 5 atribut, atributnya adalah *id_admin*, *username*, *nama*, *password*, dan *e-mail*. *Class* ini menggunakan *id_admin* sebagai *primary key*. Pada *Class* Guest terdapat 4 atribut yaitu atribut *id_guest*, *token*, *created_at*, dan *updated_at*. *Primary key* pada *Class* Guest adalah *id_guest*.



Gambar 29. Entity Relationship Diagram Tanggap Bencana Kebakaran PT. Pelabuhan Indonesia Cabang Panjang

3.4 User Interface System

User Interface adalah mekanisme komunikasi antara pengguna dan sistem. *User Interface* juga merupakan serangkaian tampilan grafis yang dapat dimengerti oleh pengguna atau *user* dan diprogram sehingga dapat berjalan dengan semestinya. Maka dari itu, *user interface* dibuat agar *user* dapat lebih mengerti bagaimana menavigasi perangkat lunak yang akan dibuat dengan lebih mudah. Desain dari *user interface* memang sangat penting untuk dikembangkan, karena desain dari sistem akan menentukan bagaimana *user* dapat melakukan interaksi dengan perangkat lunak tersebut.

3.4.1 Rancangan Tampilan *Splash Screen*

Gambar 30 tampilan *splash screen* merupakan tampilan awal ketika *user* akan menjalankan perangkat lunak Tanggap Bencana Kebakaran PT. Pelabuhan Indonesia Cabang Panjang. *Splash Screen* merupakan kondisi yang digunakan untuk memberikan *feedback* atau menggambarkan bahwa perangkat lunak dalam proses *loading* dengan menggunakan *image* atau gambar. Pada aplikasi ini logo PT. Pelabuhan Indonesia II dipilih menjadi gambar pada *splash screen*. Di bawah logo terdapat nama perangkat lunak yang dibuat yaitu “Tanggap Bencana Kebakaran PT. Pelabuhan Indonesia Cabang Panjang”.

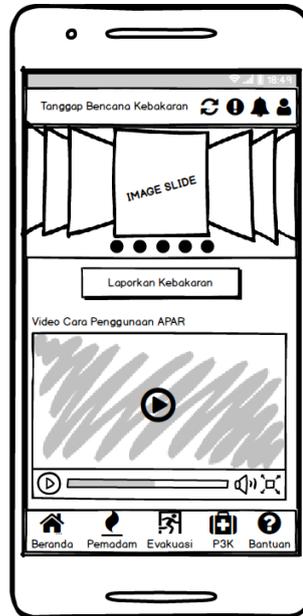


Gambar 30. Rancangan *Splash Screen*

3.4.2 Rancangan Tampilan Beranda

Gambar 31 tampilan beranda merupakan menu utama dari perangkat lunak yang akan dibuat. Halaman ini akan otomatis terbuka setelah tampilan *splash screen*. Pada halaman beranda terdapat *slide show* gambar, tombol pelaporan kebakaran,

dan terdapat berbagai macam video informatif yang dapat membantu jika terjadi kebakaran.



Gambar 31. Rancangan Beranda

3.4.3 Rancangan Halaman *Login Admin*

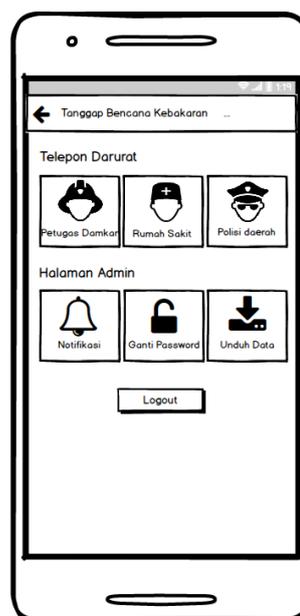


Gambar 32. Rancangan *Login Admin*

Gambar 32 adalah halaman *login* admin. Pada halaman ini terdapat sebuah *form login*, masukkan *username* dan *password* dengan benar untuk masuk ke halaman admin. Jika salah memasukkan *username* atau *password* maka tidak dapat masuk ke menu admin.

3.4.4 Rancangan Halaman Admin

Gambar 33 merupakan halaman admin yang hanya dapat diakses oleh admin, karena diharuskan untuk *login* terlebih dahulu. Pada halaman admin terdapat 3 ikon pada bagian telepon darurat yang berisi ikon Rumah Sakit Umum Daerah (RSUD), Pemadam Kebakaran, dan Polisi Daerah (Polda). Jika admin memilih salah satu dari ikon telepon darurat maka sistem akan langsung melakukan panggilan terhadap nomor tersebut. Kemudian, pada bagian halaman admin terdapat 3 ikon yaitu ikon notifikasi, ikon ubah password, dan ikon unduh data yang akan mengunduh secara otomatis data-data Alat Pemadam Api Ringan (APAR) dalam bentuk pdf ketika tombol unduh data dipilih. Di bagian bawah terdapat tombol *logout* untuk keluar dari akun admin.



Gambar 33. Rancangan Halaman Admin

3.4.5 Rancangan Halaman Ubah *Password*

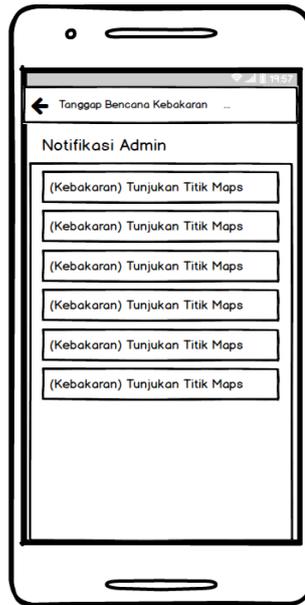
The image shows a mobile application interface for changing a password. At the top, there is a navigation bar with a back arrow and the text 'Tanggap Bencana Kebakaran'. Below this, a form titled 'Form Ganti Password' is displayed. The form contains two input fields: 'PASSWORD LAMA :' and 'PASSWORD BARU :'. At the bottom of the form, there are two buttons: 'Batal' and 'Simpan'.

Gambar 34. Rancangan *Form* Ubah *Password*

Gambar 34 merupakan halaman berisi *form* untuk merubah *password*. Untuk merubah *password*, admin diminta untuk mengisi *password* lama dan *password* baru. Setelah selesai mengisi *password* lama dan baru, pilih tombol “Simpan” agar *password* yang ingin diganti tersimpan atau pilih tombol “Batal” untuk kembali ke halaman admin.

3.4.6 Rancangan Halaman Notifikasi Admin

Pada Gambar 35 terdapat halaman notifikasi admin. Pada halaman ini akan ditampilkan notifikasi dari laporan pengguna berupa titik lokasi ketika terjadi kebakaran. Ketika pengguna melaporkan terjadinya kebakaran maka laporan tersebut akan dikirimkan ke dalam notifikasi admin terlebih dahulu. Jika admin membuka notifikasi tersebut, maka akan ditampilkan peta lokasi kebakaran yang dilaporkan oleh pengguna (Gambar 36).



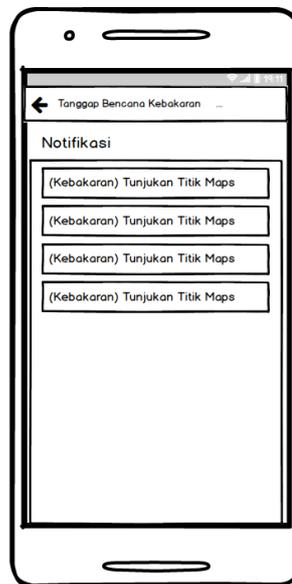
Gambar 35. Rancangan Halaman Notifikasi Admin

Gambar 36 akan ditampilkan ketika admin membuka notifikasi kebakaran, admin dapat memeriksa informasi yang masuk apakah benar atau tidak. Jika informasi yang dilaporkan oleh pengguna benar, maka admin dapat memilih “sebarikan” untuk menyebarkan lokasi kebakaran yang terjadi kepada seluruh pengguna atau memilih tombol tutup untuk mengabaikan notifikasi/laporan palsu oleh pengguna.



Gambar 36. Rancangan Halaman Sebarikan Lokasi (Admin)

3.4.7 Rancangan Halaman Notifikasi Pengguna



Gambar 37. Rancangan Halaman Notifikasi Pengguna

Gambar 37 adalah gambar halaman notifikasi pengguna, pengguna akan menerima pesan notifikasi yang disebarkan oleh admin. Pesan notifikasi ini berisi lokasi kebakaran yang terjadi seperti pada Gambar 38.



Gambar 38. Rancangan Halaman Lokasi Notifikasi Pengguna

Gambar 38 merupakan gambar lokasi titik kebakaran yang terjadi, gambar ini akan muncul apabila pengguna memilih salah satu pesan notifikasi yang ada.

3.4.8 Rancangan Halaman Menu Navigasi Pemadam

Gambar 39 adalah rancangan dari menu navigasi pemadam. Cara mengaksesnya adalah dari beranda pilih menu navigasi pemadam pada bagian bawah aplikasi. Pada menu ini terdapat 3 pilihan alat pemadam kebakaran yaitu Alat Pemadam Api Ringan (APAR), Mobil Pemadam Kebakaran, dan *Hydrant*.

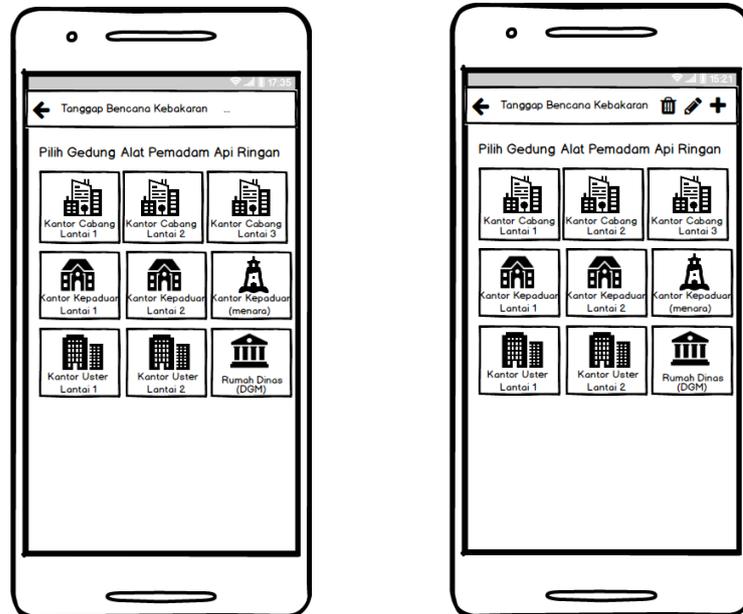


Gambar 39. Rancangan Menu Navigasi Pemadam

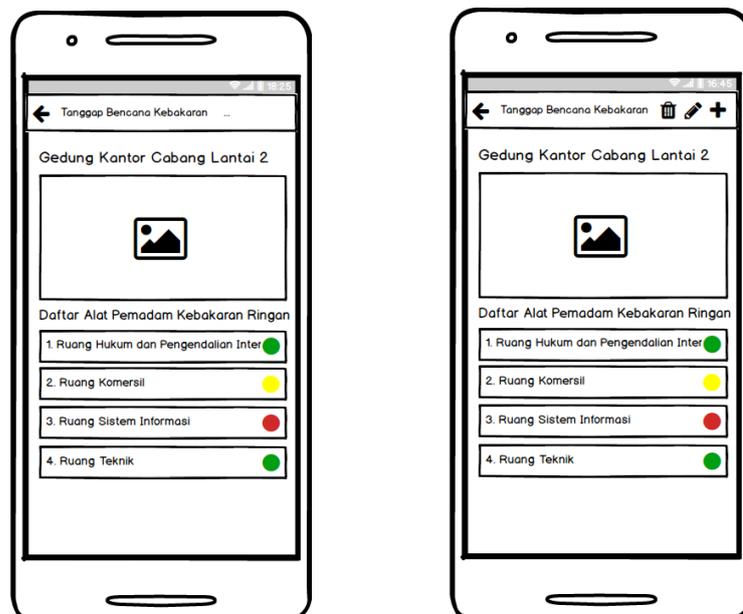
3.4.9 Rancangan Halaman Alat Pemadam Api Ringan

Gambar 40 akan ditampilkan ketika pengguna memilih Alat Pemadam Api Ringan pada halaman pemadam (lihat Gambar 40). Gambar sebelah kiri merupakan tampilan pada pengguna dan sebelah kanan adalah tampilan admin. Admin dapat melihat ikon tambah, *edit* dan hapus. Halaman ini berisi daftar gedung yang ada

pada PT. Pelabuhan Indonesia Cabang Panjang. Untuk melihat Alat Pemadam Api Ringan (APAR) yang ada, maka pengguna dapat memilih gedung dan lantai dari lokasi yang ingin dilihat posisi dan alat pemadam kebakarannya.

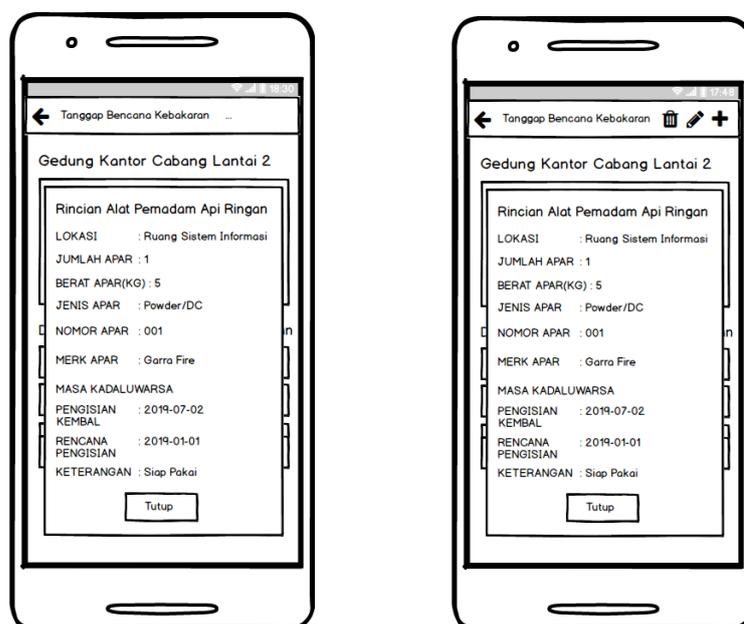


Gambar 40. Rancangan Halaman Daftar Gedung (kiri pengguna, kanan admin)



Gambar 41. Rancangan Halaman Alat Pemadam Api Ringan (kiri pengguna, kanan admin)

Pada Gambar 41 merupakan halaman informasi Alat Pemadam Api Ringan (APAR). Halaman ini berisi gambar denah lokasi berisi titik-titik keberadaan Alat Pemadam Api Ringan (APAR) yang ada pada ruangan tersebut, *list* alat pemadam kebakaran yang ada, dan indikator masa kadaluwarsa. Indikatornya terdiri dari 3 warna yaitu hijau jika kadaluwarsa alat masih di atas 3 bulan, kuning kadaluwarsa alat di bawah 3 bulan, dan merah berarti alat sudah memasuki masa kadaluwarsa. Jika alat dipilih, maka akan tampil *pop-up* detail alat pemadam seperti Gambar 42.



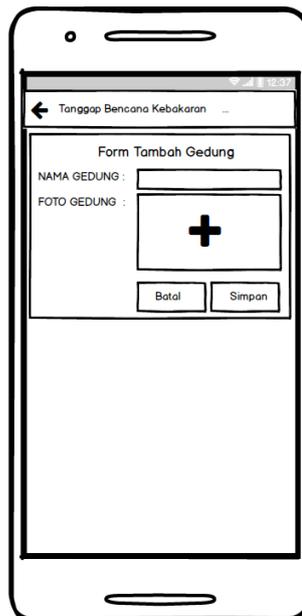
Gambar 42. Rancangan Halaman *Pop-up* Alat Pemadam Api Ringan

Gambar 42 merupakan *pop-up* detail informasi mengenai Alat Pemadam Api Ringan (APAR) yang ditampilkan ketika pengguna memilih salah satu lokasi. Pada *pop-up* tersebut, terdapat informasi lokasi, jumlah, berat, jenis, nomor, merk, dan masa kadaluwarsa keterangan mengenai Alat Pemadam Api Ringan (APAR).

3.4.10 Rancangan *Form* Tambah Gedung

Gambar 43 merupakan *form* untuk melakukan penambahan data gedung. Untuk mengakses *form* ini, pada halaman daftar gedung (lihat Gambar 40) pilih ikon

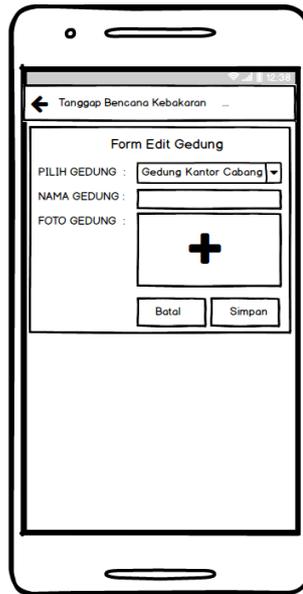
tambah. Kemudian, sistem akan menampilkan *form* untuk menambah data gedung. *Form* ini berisi nama gedung dan foto gedung. Terdapat 2 tombol yang dapat dipilih. Jika memilih tombol “Simpan” maka sistem akan menyimpan data yang telah diisi sebelumnya ke dalam *database*, dan jika memilih tombol “Batal” maka akan kembali ke halaman sebelumnya.



Gambar 43. Rancangan *Form* Tambah Gedung

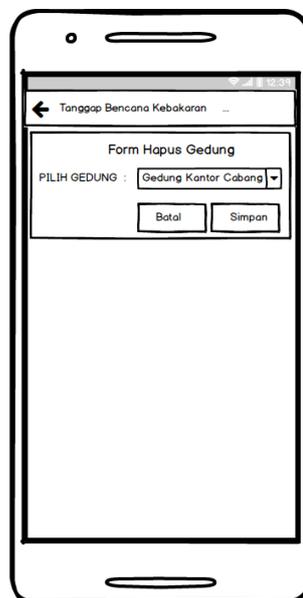
3.4.11 Rancangan *Form* Edit Gedung

Gambar 44 merupakan *form* untuk melakukan *edit* pada data gedung yang sudah ada. *Form* ini dapat diakses dengan cara memilih ikon *edit* pada halaman daftar gedung (lihat Gambar 40). *Form* untuk melakukan *edit* pada data gedung akan ditampilkan oleh sistem. Sebelum melakukan *edit* data, pilih data gedung yang ingin disunting. Kemudian isi nama gedung dan foto gedung dengan data yang baru. Setelah selesai mengisi *form* dengan data yang baru, maka dapat memilih tombol “Simpan” untuk menyimpan data yang sudah diperbarui ke *database* atau dapat memilih tombol “Batal” untuk kembali ke halaman daftar gedung.



Gambar 44. Rancangan *Form Edit Gedung*

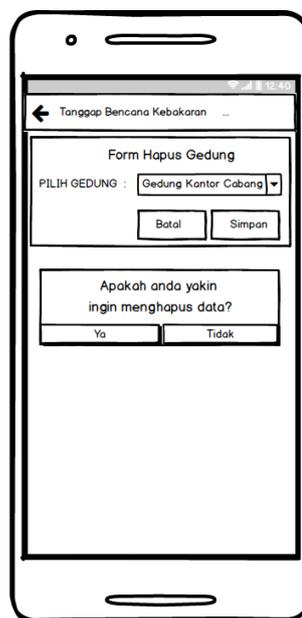
3.4.12 Rancangan *Form Hapus Gedung*



Gambar 45. Rancangan *Form Hapus Gedung*

Gambar 45 merupakan *design User Interface* dari *form* untuk melakukan penghapusan pada data gedung yang sudah ada pada *database*. Untuk melakukan

akses pada *form* hapus gedung dapat dilakukan adalah dengan cara memilih ikon hapus pada halaman daftar gedung (lihat Gambar 40). *Form* untuk menghapus data gedung akan ditampilkan oleh sistem. Untuk melakukan penghapusan data gedung, pilih data gedung yang akan dihapus. Kemudian, pilih tombol “Hapus” untuk menghapus data yang diinginkan dari *database* atau pilih tombol “Batal” untuk kembali ke halaman daftar gedung. Jika memilih tombol “Hapus” maka akan tampil pesan konfirmasi.



Gambar 46. Rancangan Konfirmasi Hapus Gedung

Pada Gambar 46 ditampilkan pesan konfirmasi jika memilih tombol hapus pada *form* hapus gedung (lihat Gambar 45). Pesan konfirmasi berfungsi agar data lebih aman karena tombol ini dapat digunakan untuk mencegah data terhapus ketika *user* secara tidak sengaja memilih tombol hapus.

3.4.13 Rancangan *Form* Tambah Alat Pemadam Api Ringan

Form untuk melakukan penambahan data Alat Pemadam Api Ringan (APAR) dapat dilihat pada Gambar 47. *Form* ini, dapat diakses saat memilih ikon tambah pada

halaman alat pemadam api ringan (lihat Gambar 41). Sistem akan menampilkan *form* untuk menambah data. *Form* ini berisi lokasi, jumlah, berat, jenis, nomor, merk, tanggal kadaluwarsa seperti tanggal pengisian kembali serta tanggal rencana pengisian, keterangan layak pakai, dan foto. Pada bagian masa kadaluwarsa yang berisi tanggal pengisian kembali. Tanggal rencana pengisian hari bulan dan tahun dipilih menggunakan *date picker*. Terdapat 2 tombol yaitu “Simpan” dan “Batal”. Pilih tombol “Simpan” untuk menyimpan data yang telah diisi ke dalam *database*, dan pilih tombol “Batal” untuk kembali ke halaman alat pemadam api ringan.

Gambar 47. Rancangan *Form* Tambah Alat Pemadam Api Ringan

3.4.14 Rancangan *Form Edit* Alat Pemadam Api Ringan

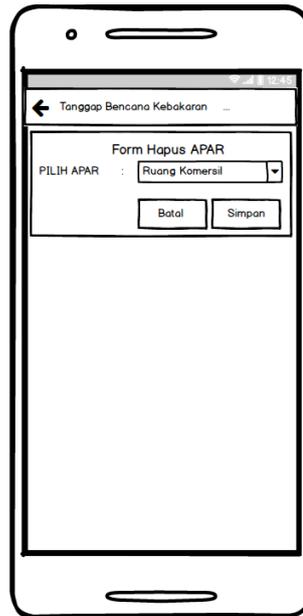
Pada Gambar 48 dibuat rancangan *User Interface* berupa *form* untuk melakukan *edit* data Alat Pemadam Api Ringan (APAR). Untuk melakukan akses terhadap *form* ini, caranya adalah dengan memilih ikon *edit* pada halaman alat pemadam api ringan (lihat Gambar 41). *Form* ini berfungsi untuk melakukan *edit* data Alat Pemadam Api Ringan (APAR). Sebelum melakukan *edit* data, pilih data alat yang ingin disunting. Setelah itu, isi kembali nama lokasi, jumlah, berat, jenis, nomor,

merk, masa kadaluwarsa seperti tanggal pengisian kembali serta tanggal rencana pengisian, keterangan, dan foto dengan data yang baru. Tanggal pengisian kembali dan tanggal rencana pengisian dipilih menggunakan *date picker*. Setelah selesai mengisi *form* dengan data yang baru, maka dapat memilih tombol “Simpan” untuk menyimpan data yang sudah diperbarui ke *database* atau dapat memilih tombol “Batal” untuk kembali ke halaman sebelumnya.

Gambar 48. Rancangan *Form Edit* Alat Pemadam Api Ringan

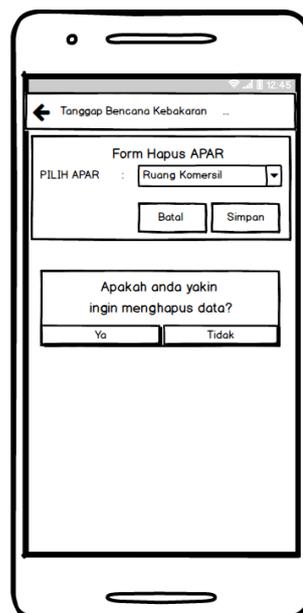
3.4.15 Rancangan *Form Hapus* Alat Pemadam Api Ringan

Rancangan *User Interface* pada Gambar 49 adalah berupa sebuah *form* untuk melakukan penghapusan data Alat Pemadam Api Ringan (APAR) yang ada pada *database*. Akses *form* hapus dilakukan dengan cara memilih ikon hapus. Setelah itu, *form* untuk menghapus data alat akan ditampilkan oleh sistem. Pilih alat yang ingin dihapus dan tekan tombol “Hapus” untuk menghapus data dari *database* atau dapat pilih tombol “Batal” untuk kembali ke halaman sebelumnya. Jika tombol “Hapus” dipilih maka sistem akan memberikan *pop-up* berupa pesan konfirmasi.



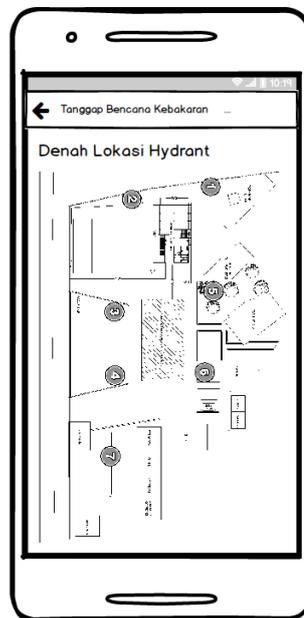
Gambar 49. Rancangan *Form Hapus* Alat Pemadam Api Ringan

Gambar 50 merupakan *pop-up* konfirmasi yang ditampilkan ketika memilih tombol hapus pada halaman alat pemadam api ringan. Pesan konfirmasi sebelum menghapus data berfungsi sebagai pengaman data sekaligus untuk mencegah data yang ada dapat terhapus ketika *user* secara tidak sengaja memilih tombol hapus.



Gambar 50. Rancangan Konfirmasi Hapus Alat Pemadam Api Ringan

3.4.16 Rancangan Halaman *Hydrant*

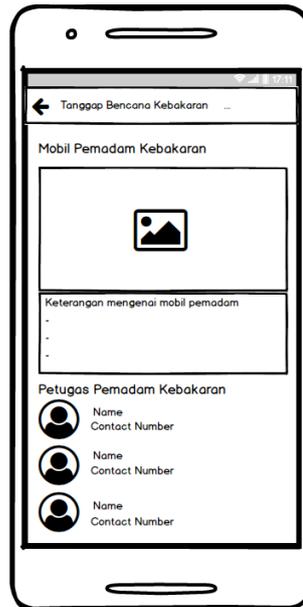


Gambar 51. Rancangan Halaman *Hydrant*

Gambar 51 merupakan halaman *hydrant*. akses ke halaman *hydrant* dilakukan dengan cara memilih *hydrant* pada halaman pemadam (lihat Gambar 39). Halaman ini berisi gambar denah dari lokasi-lokasi *hydrant* yang dimiliki PT. Pelabuhan Indonesia Cabang Panjang. Lokasi *hydrant* tersebut akan digambarkan dengan titik merah.

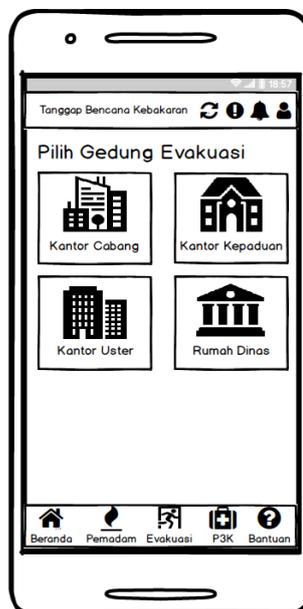
3.4.17 Rancangan Halaman Mobil Pemadam Kebakaran

Gambar 52 merupakan rancangan dari *User Interface* untuk melihat halaman mobil pemadam. Untuk melakukan akses ke halaman ini, dari halaman pemadam (lihat Gambar 39) pengguna pilih ikon mobil pemadam. Halaman ini berisi berbagai informasi dari mobil pemadam yang digunakan oleh pada PT. Pelabuhan Indonesia Cabang Panjang. Dan pada bagian bawah terdapat data informasi dari petugas pemadam kebakaran yang ada pada PT. Pelabuhan Indonesia Cabang Panjang.



Gambar 52. Rancangan Halaman Mobil Pemadam

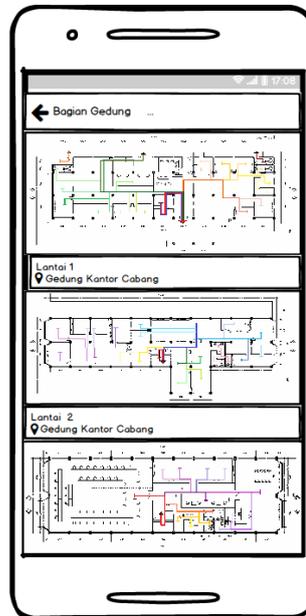
3.4.18 Rancangan Halaman Evakuasi



Gambar 53. Rancangan Halaman Gedung Evakuasi

Gambar 53 merupakan halaman dari menu navigasi evakuasi. Pada halaman ini berisi daftar gedung yang ada pada PT. Pelabuhan Indonesia Cabang Panjang.

Untuk melihat jalur evakuasi maka pengguna dapat memilih gedung yang ingin dilihat jalur evakuasinya. Kemudian sistem akan menampilkan halaman jalur evakuasi seperti pada gambar 54.

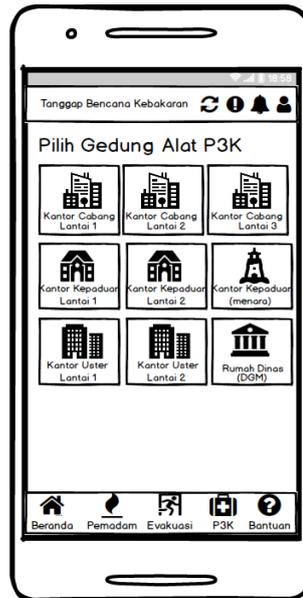


Gambar 54. Rancangan Halaman Jalur Evakuasi

Gambar 54 merupakan denah dari jalur-jalur evakuasi yang ada pada setiap ruangan di semua gedung yang ada di PT. Pelabuhan Indonesia Cabang Panjang. Pada halaman ini, terdapat sebuah gambar denah dengan ikon bulat berisi nomor untuk mewakili suatu ruangan. Pengguna dapat memilih jalur berdasarkan nomor.

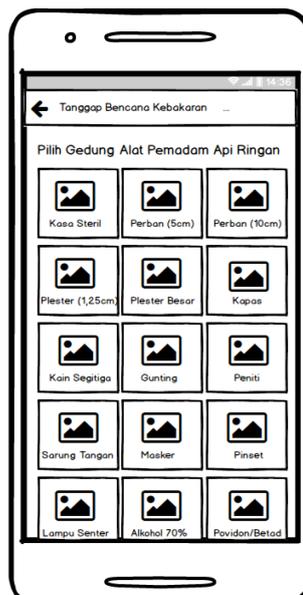
3.4.19 Rancangan Halaman Pertolongan Pertama pada Kecelakaan

Pada Gambar 55 terdapat rancangan dari menu navigasi Pertolongan Pertama pada Kecelakaan (P3K). Untuk mengakses menu ini adalah dengan cara memilih menu navigasi P3K pada bagian bawah aplikasi. Pada halaman ini, terdapat daftar gedung dan lantai yang ada di PT. Pelabuhan Indonesia Cabang Panjang. Untuk melihat alat Pertolongan Pertama pada Kecelakaan (P3K) yang ada pengguna harus memilih gedung dan lantai dari lokasi yang ingin dilihat posisi dan alatnya.



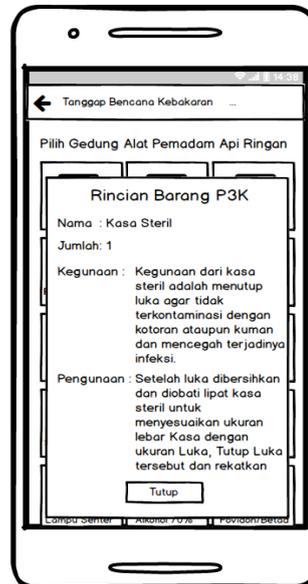
Gambar 55. Rancangan Halaman Gedung Alat P3K

Gambar 56 merupakan halaman yang ditampilkan oleh sistem ketika pengguna memilih salah satu pilihan gedung yang ada pada halaman menu navigasi pertolongan pertama pada kecelakaan (lihat Gambar 50). Halaman ini berisi daftar Alat Pertolongan Pertama pada Kecelakaan (P3K). Terdapat daftar berisi nama dan foto alat. Jika alat telah dipilih maka sistem akan menampilkan *pop-up* detail alat.



Gambar 56. Rancangan Halaman Daftar Alat P3K

Gambar 57 rancangan halaman *pop-up* detail informasi mengenai Alat Pertolongan Pertama pada Kecelakaan (P3K). *Pop-up* tersebut, terdapat informasi mengenai nama alat, jumlah yang ada, kegunaan dari alat tersebut, dan cara penggunaannya.



Gambar 57. Rancangan Halaman *Pop-up* Alat P3K

3.4.20 Rancangan Halaman Menu Navigasi Bantuan



Gambar 58. Rancangan Halaman Menu Navigasi Bantuan

Gambar 58 adalah rancangan dari menu navigasi bantuan. Pada halaman menu navigasi ini, terdapat berbagai macam informasi yang dapat membantu pengguna dalam menjalankan perangkat lunak Tanggap Bencana Kebakaran pada PT. Pelabuhan Indonesia Cabang Panjang.

3.4.21 Rancangan Halaman Tentang

Gambar 59 merupakan rancangan dari halaman tentang pada perangkat lunak. Pada halaman ini terdapat logo, penjelasan mengenai perangkat lunak seperti tentang apa tujuan dan kegunaan dari perangkat lunak ini dibuat, penjelasan singkat mengenai PT. Pelabuhan Indonesia Cabang Panjang, dan informasi tentang perangkat lunak.



Gambar 59. Rancangan Halaman Menu Navigasi Tentang

3.4.22 Rancangan Laporan Kebakaran

Gambar 60 merupakan halaman untuk melaporkan terjadinya kebakaran. Untuk mengakses halaman ini, pada beranda pilih tombol “laporkan kebakaran”. Setelah

itu, akan muncul halaman untuk melaporkan kebakaran berupa sebuah *maps* yang langsung menitikkan lokasi di tempat pengguna. Selain itu, pengguna juga dapat menggeser lokasi agar lebih akurat. Pilih tombol “Laporkan” untuk mengirim laporan kebakaran kepada admin, atau pilih tombol “Batal” untuk kembali ke halaman sebelumnya.



Gambar 60. Rancangan Halaman Laporkan Kebakaran

4. Implementasi Sistem

Setelah selesai membuat perancangan sistem perangkat lunak, maka tahapan selanjutnya adalah tahapan implementasi. Pada tahap implementasi, sistem akan mulai dibangun. Sistem yang akan dibangun berbasis android dan akan dibangun menggunakan *Integrated Development Enviroment* (IDE) untuk sistem operasi Android yaitu Android Studio dengan bahasa pemrograman Java. Pada tahap ini, hasil rancangan dan desain sistem yang hampir jadi diuji coba dan mulai diimplementasikan dalam kebutuhan *user*. Sesekali akan dilakukan pengujian dan evaluasi sistem sebelum benar-benar digunakan oleh *user* untuk mengetahui letak kemungkinan kesalahan sistem.

Pengerjaan sistem dengan metode *waterfall* secara umum mampu menghadirkan metode yang cepat, efektif, dan efisien dalam pembuatan suatu sistem. Hal ini dikarenakan metode *waterfall* memiliki proses yang urut mulai dari analisa hingga *support*. Selain itu, pada metode ini setiap proses memiliki spesifikasinya sendiri sehingga sebuah sistem dapat dikembangkan sesuai dengan apa yang dikehendaki dan lebih tepat sasaran.

5. Pengujian

Tahap pengujian dilakukan untuk memastikan apakah semua fungsi yang ada pada sistem berjalan sesuai dengan perancangan yang telah dibuat sebelumnya. Pengujian akan dilakukan dengan cara *blackbox testing* menggunakan pengujian fungsional dan evaluasi *user acceptance*. Pada penelitian ini, pengujian *system testing* akan menggunakan metode *blackbox* yang berfokus untuk menguji sistem dari sudut pandang fungsional sistem dan pengujian dengan pengujian *user acceptance* untuk menguji kelayakan sistem yang dilakukan oleh pengguna.

5.1 Pengujian *Blackbox Testing*

Pengujian dengan metode *blackbox* berfungsi untuk menguji sejauh mana sistem yang dibangun berjalan, apakah sistem tersebut berjalan sesuai dengan perancangan dan fungsionalitasnya. Rencana pengujian yang dilakukan dalam pengujian *blackbox* yang akan digunakan adalah pengujian fungsionalitas sistem. Pengujian fungsionalitas sistem yang dilakukan pada penelitian ini adalah menggunakan *Equivalence Partitioning* (EP). Pengujian dengan *Equivalence Partitioning* (EP) dilakukan dengan cara membagi domain. Domain tersebut kemudian dimasukkan ke dalam kelas-kelas sehingga akan didapatkan beberapa *test case* dari sistem. Kriteria pengujian yang digunakan adalah kelas uji, *test case*, *input*, *output* yang diharapkan, dan *output* yang diperoleh. Pada pengujian ini, setiap fungsi harus dipastikan bahwa *output* yang diperoleh akan sama dengan *output* yang diharapkan.

Tabel 4. Tabel Pengujian Oleh Admin

No	Kelas Uji	Daftar Pengujian	Kasus Uji
1	Melakukan <i>login</i> dan <i>logout</i>	Melakukan <i>login</i>	Mengisi <i>field form</i> <i>username</i> dan <i>password</i> dengan baik dan benar
			Mengisi <i>field form</i> menggunakan <i>username</i> yang salah dan masukan <i>password</i> yang benar
			Mengisi <i>field form</i> menggunakan <i>username</i> yang benar dan masukan <i>password</i> yang salah
		Melakukan <i>logout</i>	Menekan tombol <i>logout</i> yang ada di menu admin
2	Kelola Gedung	Menambah Gedung	Mengisi <i>field form</i> yang tersedia dengan baik dan benar kemudian memilih tombol simpan
			Mengisi <i>field form</i> yang tersedia kemudian memilih tombol batal
		Mengubah Gedung	Mengubah data yang ada di dalam <i>field form</i> yang tersedia kemudian memilih tombol simpan
			Mengubah data yang ada di dalam <i>field form</i> yang tersedia kemudian memilih tombol batal
		Menghapus Gedung	Menghapus salah satu data gedung yang ada didaftar gedung

Tabel 4. Tabel Pengujian Oleh Admin (Lanjutan)

No	Kelas Uji	Daftar Pengujian	Kasus Uji
3	Kelola data dan daftar Alat Pemadam Api Ringan (APAR)	Menambah APAR	Mengisi <i>field form</i> yang tersedia kemudian memilih tombol simpan
			Mengisi <i>field form</i> yang tersedia kemudian memilih tombol batal
		Mengubah APAR	Mengubah data yang ada di dalam <i>field form</i> yang tersedia kemudian memilih tombol simpan
			Mengubah data yang ada di dalam <i>field form</i> yang tersedia kemudian memilih tombol batal
		Menghapus APAR	Menghapus salah satu data APAR yang ada didaftar APAR
4	Menyebarkan lokasi kebakaran	Menyebarkan lokasi terjadinya kebakaran	Menekan tombol “sebar” dari bagian notifikasi yang terdapat di menu admin ketika ada yang melaporkan kebakaran.
			Menekan tombol “tutup” dari bagian notifikasi ketika ada yang melaporkan kebakaran.
			Menekan tombol “tutup” dari bagian notifikasi ketika ada yang melaporkan kebakaran.

Tabel 4. Tabel Pengujian Oleh Admin (Lanjutan)

No	Kelas Uji	Daftar Pengujian	Kasus Uji
5	Melakukan panggilan darurat	Melakukan panggilan darurat ke pemadam kebakaran	Menekan tombol dengan nama “pemadam kebakaran” yang ada di menu admin
		Melakukan panggilan darurat kepada RSUD Lampung	Menekan tombol dengan nama “RSUD Lampung” yang berada di menu admin
		Melakukan panggilan darurat kepada Polda Lampung	Menekan tombol dengan nama “Polda Lampung” yang berada di menu admin
6	Mengganti <i>password</i>	Menganti <i>Password</i>	Mengisi <i>field form</i> penggantian <i>password</i> dan pilih tombol simpan
7	Mengunduh Data	Mengunduh Data	Melakukan unduh data dengan memilih tombol “unduh data” yang terdapat pada menu admin

Pada Tabel 4 memperlihatkan tabel pengujian yang dilakukan oleh admin. Pengujian ini meliputi melakukan *login*, kelola gedung, kelola Alat Pemadam Api Ringan (APAR), menyebarkan lokasi kebakaran dari laporan pengguna, melakukan panggilan darurat, melakukan penggantian *password*, dan mengunduh data.

Tabel 5. Tabel Pengujian Oleh Pengguna

No	Kelas Uji	Daftar Pengujian	Kasus Uji
1	Melaporkan adanya kebakaran	Melaporkan kebakaran yang terjadi	Melaporkan lokasi dari sebuah kebakaran dengan menggunakan tombol “laporkan kebakaran” di menu utama/beranda
2	Melihat notifikasi kebakaran	Melihat notifikasi terjadi kebakaran	Melihat notifikasi yang masuk pada saat admin menyebarkan lokasi dari kebakaran yang ada kepada pengguna
3	Melihat informasi Alat Pemadam Api Ringan (APAR)	Melihat informasi Alat Pemadam Api Ringan (APAR) yang ada	Memilih Gedung untuk melihat informasi APAR yang berada di gedung tersebut Melihat informasi APAR yang tersedia Melihat gambar dari lokasi APAR yang tersedia Melihat bagian warna yang ada pada indikator tanda kadaluwarsa APAR teraplikasi dengan baik Melihat <i>pop-up</i> detail informasi dari APAR yang tersedia
4	Melihat gambar dari lokasi <i>hydrant</i>	Melihat gambar dari lokasi <i>hydrant</i> yang ada	Melihat gambar lokasi dari <i>hydrant</i> yang ditampilkan ketika pengguna memilih <i>hydrant</i> pada menu pemadam

Tabel 5. Tabel Pengujian Oleh Pengguna (Lanjutan)

No	Kelas Uji	Daftar Pengujian	Kasus Uji
5	Melihat informasi mobil pemadam kebakaran serta petugasnya	Melihat informasi mobil pemadam kebakaran dan petugasnya	Melihat informasi dan data mobil pemadam serta para petugasnya ketika pengguna menekan tombol mobil pemadam yang ada di menu pemadam
6	Melihat jalur-jalur evakuasi yang ada	Melihat informasi berupa gambar jalur evakuasi	Memilih salah satu gedung yang ada dan ingin dilihat gambar jalur evakuasinya Melihat gambar jalur evakuasi yang muncul sesuai dengan gedung yang dipilih
7	Melihat informasi bantuan	Melihat informasi bantuan	Melihat informasi bantuan berupa tata cara penggunaan aplikasi
8	Melihat informasi tentang	Melihat informasi tentang perangkat lunak	Melihat informasi yang ada tentang perangkat lunak

Pada Tabel 5 memperlihatkan pengujian sitem dilakukan oleh pengguna. Pengguna dapat menguji sistem meliputi melaporkan kebakaran, melihat notifikasi kebakaran, melihat informasi Alat Pemadam Api Ringan (APAR), melihat lokasi *hydrant*, melihat informasi mobil pemadam serta petugasnya, melihat informasi jalur-jalur evakuasi, melihat informasi alat-alat P3K, dan melihat menu bantuan.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Dari hasil penelitian yang dilakukan dapat disimpulkan sistem perangkat lunak Tanggap Bencana Kebakaran PT. Pelabuhan Indonesia II (Persero) Cabang Panjang telah berhasil dalam melakukan upaya dalam membantu memberikan informasi ketika terjadi bencana kebakaran dan dapat mempermudah pegawai saat melakukan manajemen alat pemadam api ringan (APAR) pada PT. Pelabuhan Indonesia II (Persero) Cabang Panjang.

B. Saran

Berdasarkan analisis, perancangan, dan implementasi sistem yang dilakukan, maka ada beberapa saran yang perlu diperhatikan dalam mengembangkan sistem ini adalah sebagai berikut:

1. Menambahkan perangkat *smart device* yang dapat terhubung dengan perangkat lunak jika terjadi kebakaran,
2. Menambahkan fitur mengirim gambar foto kebakaran secara langsung ketika melakukan pelaporan kebakaran.
3. Menambahkan fitur jalur evakuasi yang lebih lengkap.

DAFTAR PUSTAKA

- Akil, Ibnu. 2016. Rekayasa Perangkat Lunak dengan Model Unified Process Studi Kasus: Sistem Informasi Journal. *Jurnal Pilar Nusa Mandiri*. 12(1):1-11.
- Alsaad, Saad Najim, & Mahmood Nadia Hussein. 2020. IOT Based Message Alert System for Emergency Situations. *International Journal of Computing and Digital Systems*. 1123-1130.
- Amores, David, Maria Vasardani, & Egemen Tanin. 2019. Smartphone Usability for Emergency Evacuation Applications. *International Proceedings in Informatics, LIPIcs*. 2:1-2:7.
- Aquino, Anthony, Joferson Bombasi, & Eui-Seok Nahm. 2019. Mobile-Based Voice Guidance System for Escape Guide in Case of Emergency. *International Journal of Simulation: Systems*. 30.1-30.4.
- Bishan, Dayal Chauhan, Rana Ajay, & Kumar Sharma Neeraj. 2017. Impact of Development Methodology on Cost & Risk for Development Projects. *International Conference on Reliability, Infocom Technologies and Optimization*. 267-272.
- BNPB. 2018. Pantauan Bencana. geospasial.bnpd.go.id/pantauanbencana. Diakses pada 9 Desember 2019.
- Bo, Jiang, Long Xiang, & Gao Xiaopeng. 2007. MobileTest: A Tool Supporting Automatic Black Box Test for Software on Smart Mobile Devices. *Second International Workshop on Automation of Software Test (AST '07)*. 8-8.
- Detik. 2018. DetikNews. news.detik.com/berita/d-4104767/fakta-fakta-kebakaran-gedung-kemenhub-yang-tewaskan-3-orang. Diakses pada 9 Desember 2019.

- Erik, F Méndez Garcés, M Mafla Gabriela, & Reyes Francisco. 2019. Analysis, Review and Development of a Conceptual Model, based on Class Diagrams as a Component of UML, Focused on Industrial Automation. *International Journal of Control Systems and Robotics*. 4:6-10.
- Fachrul, K, & W Gianto. 2015. *Cepat Menguasai Pemrograman Android*. Universitas Brawijaya Press, Malang. 110 hlm.
- Faisal, Ahmad, & Samuel, HT. 2019. Simulator Sistem Monitoring dan Peringatan Dini Kebakaran Gedung Berbasis Web. *Jurnal Ilmiah Teknik Elektro*.17:57-72.
- Firly, Nadia. 2019. *Android Application Development for Rookies with Database*. Elex Media Komputindo, Jakarta. 260 hlm.
- Gamido, Heidilyn V., & Marlon V Gamido. 2019. Comparative Review of The Features of Automated Software Testing Tools. *International Journal of Electrical and Computer Engineering (IJECE)*. 4473-4478.
- Gerhard, H Steinke, Shams Al-Deen Meshal, & C. LaBrie Ryan. 2017. Innovating Information System Development Methodologies with Design Thinking. *International Conference on Applied Innovations in IT*. 51-55.
- Hanza, Zahra AbdulKarim, & Mustafa Hammad. 2020. Testing Approaches for Web and Mobile Applications: An Overview. *International Journal of Computing and Digital Systems*. 657-664.
- ILO. 2018. *Manajemen Resiko Kebakaran*. Kantor Perburuhan Internasional, Jakarta. 22 hlm.
- Jubilee Enterprise. 2015. *Mengenal Dasar-Dasar Pemrograman Android*. Elex Media Komputindo, Jakarta. 136 hlm.
- Kemen PU. 2009. Pedoman Teknis Manajemen Proteksi Kebakaran. Kemen PU, Jakarta. 125 hlm.
- KEMENHUB. 2016. Peraturan Menteri Perhubungan Republik Indonesia. KEMENHUB, Jakarta. 51 hlm.
- Mariani, Leonardo, Mauro Pezzè, Oliviero Riganelli, & Mauro Santoro. 2011. AutoBlackTest: A Tool for Automatic Black-Box Testing. *International Conference on Software Engineering (ICSE)*. 1013-1015.

- Mori, Hideki, Masaki Ito, & Kaoru Sezaki. 2017. Early Fire Alert System During an Evacuation with Evacuation with Mobile Sensing Technology. *Journal of Disaster Research*. 12(2):320-328.
- Pradiga, Akbar Perdana, Adianto, & Mades Darul Khairunsyah. 2018. Identifikasi Resiko dan Perancangan Sistem Proteksi Kebakaran Android. *Proceeding 2nd Conference On Safety Engineering*. 29-32.
- Rahman, Vinky, & Stephanie. 2017. Passive Fire Building Protection System Evaluation (Case Study: Millennium ICT Centre). *IOP Conference Series: Earth and Environmental Science*. 126:1-6.
- Ramadhan, Wahyu, & Abdi Wahab. 2019. Analisa dan Perancangan Sistem Manajemen Aset Pada Dinas Penanggulangan Kebakaran di Provinsi DKI Jakarta. *Jurnal Ilmu Komputer dan Informatika* 2(3):98-107.
- Ramawangsa, Panji Anom, Debby Seftyarizki, & Dwi OKtavallyan Saputri. 2019. Evaluasi Jalur Evakuasi Bencana Kebakaran Pada Sirkulasi Gedung Serbaguna UNIB. *Jurnal Manajemen Aset Infrastruktur & Fasilitas* 3(1):1-10.
- Rusmawan, Uus. 2019. *Teknik Penulisan Tugas Akhir dan Skripsi Pemrograman*. Elex Media Komputindo, Jakarta. 208 hlm.
- Safaat, N. 2018. *Android Pemrograman Aplikasi Mobile Berbasis Android. (Revisi Ketiga)*. Informatika, Bandung. 392 hlm.
- Satya Putra, Alfa. 2014. *Beginning Android Programming with ADT Bundle*. Elex Media Komputindo, Jakarta. 186 hlm.
- Satya Putra, Alfa, & Eva Maulina Aritonang. 2016. *Let`s Build Your Android Apps with Android Studio*. Elex Media Komputindo, Jakarta. 224 hlm.
- Sugiarti, Yuni. 2018. *Dasar-Dasar Pemrograman Java Netbeans: Database UML dan Interface*. PT. Remaja Rosdakarya, Bandung. 262 hlm.
- Sulaiman, Noraini, Sharifah Sakinah Syed Ahmad, & Sabrina Ahmad. 2019. *Logical Approach: Consistency Rules between Activity Diagram and Class Diagram*. *International Journal on Advance Science Engineering Information Technology*. 9:552-559.
- Supriatin. 2017. Merdeka.com. merdeka.com/peristiwa/tragedi-kebakaran-pabrik-mercon-di-tangerang-tewaskan-49-karyawan.html. Diakses pada 9 Desember 2019.

Yurindra. 2017. *Software Engineering*. Deepublish, Yogyakarta. 158 hlm.

Zufria, Ilka. 2013. Pemodelan Berbasis UML (Unified Modeling Language) dengan Strategi Teknik Orientasi Objek User Centered Design(UCD) dalam Sistem Administrasi Pendidikan. *Jurnal Processor*. 1-15.